OWASP Top Ten Defenses in Profense

The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Profense™ Professional provides defenses against all OWASP top ten 2007 vulnerabilities.

OWASP Top Ten 2007 summary	Profense defenses	Professional	Base
A1 - Cross Site Scripting (XSS) *XSS flaws occur whenever an application takes user supplied data and sends it to a web browser without first validating orencoding that content. XSS allows attackers to execute script in the victim's browser which can hijack user sessions, deface web sites, possibly introduce worms, etc.*	**Profense detects and blocks Cross Site Scripting (XSS) attacks through validation of user input using either negative or positive security policies.**	✔	✔ *Positive Only*
A2 - Injection Flaws *Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.*	**Profense detects and blocks injection attacks through validation of user input using either negative or positive security policies.**	✔	✔ *Positive Only*
A3 - Malicious File Execution *Code vulnerable to remote file inclusion (RFI) allows attackers to include hostile code and data, resulting in devastating attacks, such as total server compromise. Malicious file execution attacks affect PHP, XML and any framework which accepts filenames or files from users.*	**Profense detects and blocks Malicious File Execution attacks through validation of user input using either negative or positive security policies.**	✔	✔ *Positive Only*
A4 - Insecure Direct Object Reference A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.	**Profense detects and blocks Insecure Direct Object Reference attacks through validation of user input using positive security policies.** **Additionally negative policies can be defined blocking direct access to directories or files (like for instance /admin/).**	✔	✔ *Positive Only*
A5 - Cross Site Request Forgery (CSRF) *A CSRF attack forces a logged-on victim's browser to send a pre-authenticated request to a vulnerable web application, which then forces the victim's browser to perform a hostile action to the benefit of the attacker. CSRF can be as powerful as the web application that it attacks.*	**Profense protects against session hijacking and CSRF attacks by injecting cryptographic validation cookies and parameters to responses from the web system.** **Forms issued by an application in the web system are bound to the session through insertion of a form validation parameter containing a cryptographic token which proves that the action formulator (the application issuing the page containing a form) is in fact part of the web system protected by Profense. This provides very strong protection against CSRF attacks as the attacker, in order to forge a request, have to know the validation token for the form action for the current session.**	✔	

OWASP Top Ten Defenses in Profense

OWASP Top Ten 2007 summary	Profense defenses	Professional	Base
A6 - Information Leakage and Improper Error Handling *Applications can unintentionally leak information about their configuration, internal workings, or violate privacy through a variety of application problems. Attackers use this weakness to steal sensitive data, or conduct more serious attacks.*	**Web server error messages are captured and replaced with configurable error messages.** **Server response rewriting allows for completely configurable policies matching and rewriting confidential data like Payment Card Numbers, Social Security Numbers, etc.**	✔	
A7 - Broken Authentication and Session Management *Account credentials and session tokens are often not properly protected. Attackers compromise passwords, keys or authentication tokens to assume other users' identities.*	**Session cookies are bound to client IPs by issuing a validation cookie containing a cryptographic token (a checksum) which validates that the client IP is the one the session token was originally issued to.** **In order for an attacker to perform session attacks he also has to steal the IP address of the target or give his IP to the target in case of session fixation attacks.**	✔	
A8 - Insecure Cryptographic Storage *Web applications rarely use cryptographic functions properly to protect data and credentials. Attackers use weakly protected data to conduct identity theft and other crimes, such as credit card fraud.*	**Profense does not directly store confidential data.** **It is possible though that confidential data is logged in the deny log.** **Log input data masking capabilities provides for configurable data masking policies rendering the data useless for an attacker.**	✔	
A9 - Insecure Communications *Applications frequently fail to encrypt network traffic when it is necessary to protect sensitive communications.*	**Profense can enable HTTPS access to web resources.** **Additionally HTTP (cleartext) requests can be redirected use HTTPS.**	✔	✔
A10 - Failure to Restrict URL Access *Frequently, an application only protects sensitive functionality by preventing the display of links or URLs to unauthorized users. Attackers can use this weakness to access and perform unauthorized operations by accessing those URLs directly.*	**Access to resources requiring a valid user session from unauthenticated users (users without a valid session) is detected and blocked by Profense.** **Resource access authorization can be enabled for web applications as well as static files like XML and PDF.**	✔	

For more information, contact us
by e-mail at sales@symtrex.com
or call us toll-free at 1.866.431.8972