OWASP ENTERPRISE SECURITY API TOOLKITS

Strong, simple security controls

Just as web applications and web services can be Public Key Infrastructure (PKI) enabled (PK-enabled) to perform for example certificate-based authentication, applications and services can be OWASP ESAPI-enabled (ES-enabled) to enable applications and services to protect themselves from attackers.

Top new features and enhancements

- There are Java EE, .NET, Classic ASP, ColdFusion/CFML, PHP, and Python language versions
- The ESAPI for Java EE version includes a Web Application Firewall (WAF) that can be used to give development teams breathing room while making fixes
- All language versions of ESAPI Toolkits are licensed under the BSD license, which is very permissive and about as close to public domain as is possible. You can use or modify ESAPI however you want, even include it in commercial products.

Don't write your own security controls!
Reinventing the wheel when it comes to developing security controls for every web application or web service leads to wasted time and massive security holes. The OWASP Enterprise Security API (ESAPI) Toolkits help software developers guard against security-related design and implementation flaws. The ESAPI Toolkit architecture is very simple – a collection of classes that encapsulate the key security operations most applications need. ESAPI is designed to make it easy to retrofit security into existing applications, as well as providing a solid foundation for new development.

Plan and prepare, don't react…
Security testing, code reviews, penetration testing and architecture reviews are not ends in themselves. Unless architects and developers are prepared to make fixes, and to guard against vulnerabilities in the first place, the results of security-focused testing and analysis fall on deaf ears. The emphasis needs to be on adding strong, simple security controls into YOUR solution stack, and training your architects and developers to use them from the start, BEFORE undergoing security testing, code reviews, penetration testing and architecture reviews.

How ESAPI Works:
Allowing for language-specific differences, all OWASP ESAPI versions have the same basic design:

- There is a set of security control interfaces. They define for example types of parameters that are passed to types of security controls. There is no proprietary information or logic contained in these interfaces.
- There is a reference implementation for each security control. The logic is not organization-specific and the logic is not application-specific. There is no proprietary information or logic contained in these reference implementation classes. An example: string-based input validation.
- There are optionally your own implementations for each security control. There may be application logic contained in these classes which may be developed by or for your organization. There may be proprietary information or logic contained in these classes which may be developed by or for your organization. An example: enterprise authentication.

Related OWASP projects:

- Learn about the most common web application vulnerabilities: OWASP Top Ten
- What security teams will be testing for after you integrate ESAPI: OWASP Application Security Verification Standard (ASVS)
- What you can do to help ensure that security is being built in, in the first place: OWASP Legal Project

For more information

For more details about OWASP ESAPI, visit http://www.owasp.org/index.php/Category:OWASP_Enterprise_Security_API

How it works out of the box, from a developer's perspective

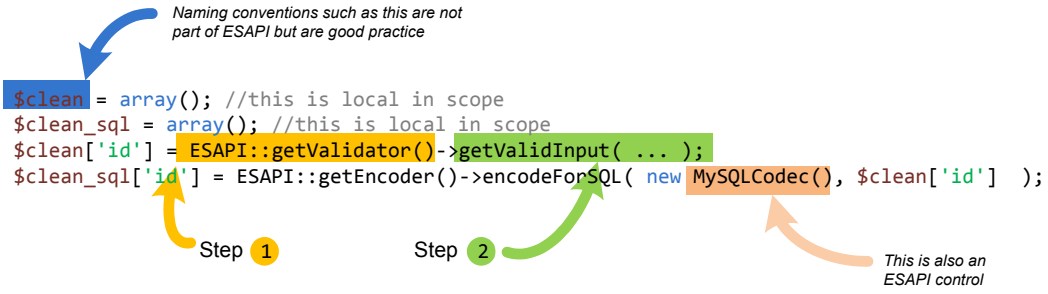
Calling security controls is easy!

The ESAPI security control interfaces include an "ESAPI" class that is commonly referred to as a "locator" class. The ESAPI locator class is called in order to retrieve singleton instances of individual security controls, which are then called in order to perform security checks (such as performing an access control check) or that result in security effects (such as generating an audit record). Below is an example of how input validation and output escaping can be done to guard against SQL injection:

Security controls that are included:

There are reference implementations for each of the following security controls:

- Authentication
- Access control
- Input validation
- Output encoding/escaping
- Cryptography
- Error handling and logging
- Communication security
- HTTP security
- Security configuration



OWASP
The Open Web Application Security Project

The Open Web Application Security Project (OWASP) is a worldwide free and open community focused on improving the security of application software. Our mission is to make application security "visible," so that people and organizations can make informed decisions about application security risks. Everyone is free to participate in OWASP and all of our materials are available under a free and open software license. The OWASP Foundation is a 501c3 not-for-profit charitable organization that ensures the ongoing availability and support for our work.