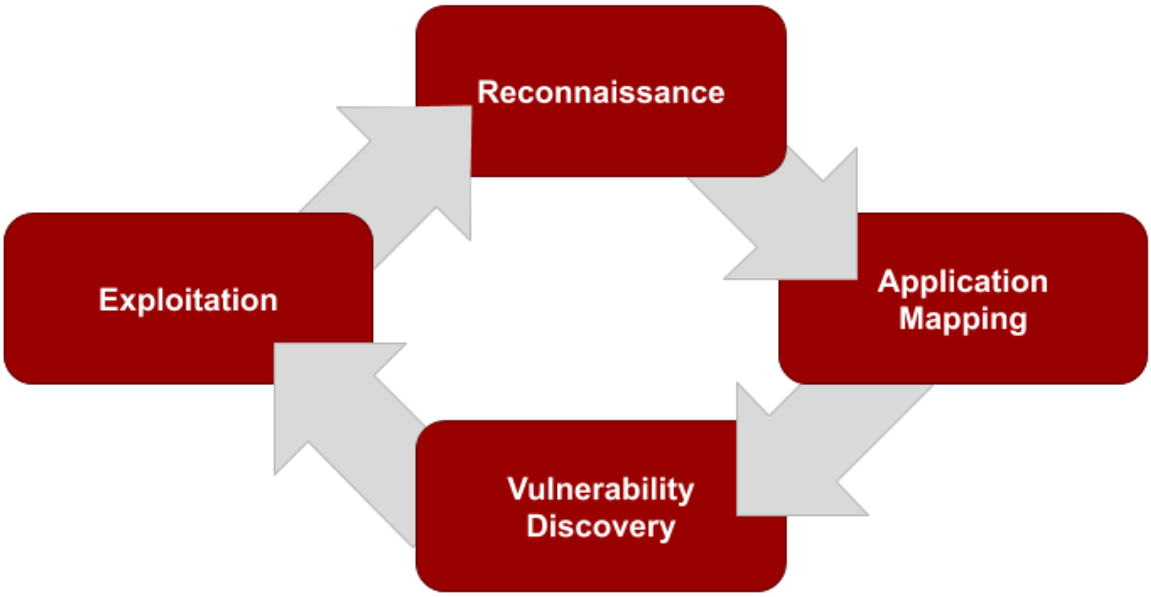
Secure Ideas follows an industry standard methodology for testing the security of web applications. This methodology is a four-step process as follows:



Note that the methodology is cyclical in nature. Successful exploitation may lead to additional iterations through the methodology.

Reconnaissance:
During reconnaissance, Secure Ideas staff will search public sources for information regarding the target organization and target application. Examples of information that is gathered during reconnaissance include:

- Lists of users, for potential use in account harvesting and takeover attacks as well as social engineering if it is in scope
- Host configuration information
- Code, keys, and notes found on public repositories such as github
- Vulnerabilities that have been publicly disclosed in the past

Secure Ideas will include information discovered as part of the final report if it is relevant to any of the exploitable findings found during the penetration testing.

Application Mapping:
The next step in the methodology is mapping, which is where Secure Ideas will study the behavior and data in the target application. Examples of items that are covered during application mapping include:

sales@secureideas.com
+1 (866) 404-7837

- Understanding the application architecture (e.g. traditional vs. single-page app)
- All fields/forms where information is sent to the server
- All types of data received from the server
- Analysis of all HTTP request and response headers used by the application
- Evaluate client-side technologies used in the application, including any third-party Javascript libraries
- Use of websockets
- Use of web services such as REST or SOAP APIs
- Identify sensitive data in any inputs to the application, including encoded data

Note the above list is not exhaustive. If Secure Ideas encounters application behavior that falls outside of the usual, it will be treated as an area of interest and scrutinized accordingly.

Discovery:
During the discovery stage of the web application penetration test, Secure Ideas will use the information gathered in the previous two steps to assist our staff in finding the vulnerabilities within the target applications. Using this information combined with both automated tools and manual testing, Secure Ideas will provide thorough coverage for all common web application vulnerabilities and will also seek out less common vulnerabilities depending on the application architecture and features (e.g. web services or the use of certain client libraries). Testing may be divided into the following categories:

Category	Testing	OWASP Top 10 References
Authentication, Authorization, and Session Management Testing	Most of this category must be tested manually, as automated scanners have difficulty understanding context within the application: • Testing of all authentication features such as login, registration, and forgot password • Testing of authorization across sensitive functionality and data • Seek out horizontal and vertical privilege escalation opportunities • Validate security of session management • Validate existence and effectiveness of Cross-Site-Request Forgery controls	A2, A4, A7, A8
Encryption and Configuration	Scan the application for flaws related to encryption and configuration. This includes: • The use of components (both server and client side) with known vulnerabilities • Use and configuration of SSL • Exploring web services for other vulnerable	A5, A6, A9

sales@secureideas.com
+1 (866) 404-7837

	points such as an XML parser	
Server-side Input Testing	Using a combination of automated and manual techniques, test server-side inputs. This testing includes items such as: • Where appropriate, test inputs for injection flaws such as SQLi, LDAP injection, Command Injection, etc... • Fuzz inputs to determine if application behavior can be influenced	A1
Client-side Input Testing	Using a combination of automated and manual techniques, test inputs on the client-side. This testing includes items such as: • Testing for Cross-Site Scripting (XSS) flaws • Testing for DOM manipulation • JSON-based manipulation • CORS policy testing • Content Security Policy (CSP) testing	A3, A10
Other Testing	Test for other common flaws that fall outside of the OWASP Top-10, such as: • Logic flaws • Insecure use of HTML5 features such as local storage • Insecure use of legacy HTML features • Websockets • Testing of any other application features not otherwise mentioned	N/A

Exploitation:
Exploitation is the final step of the web application penetration test. By exploiting the vulnerability, Secure Ideas is able to provide a realistic understanding of the business exposure from the target. Depending on the flaws discovered, exploitation exercises are often limited to the minimum required to build a proof of concept so that developers and security staff may reproduce the issue. If discovered vulnerabilities allow access to the underlying system, Secure Ideas will begin to reiterate through the methodology by performing reconnaissance on the server and/or network. Further mapping and discovery of the local network will wait until Secure Ideas confirms the scope with [***CLIENT]'s point of contact. Secure Ideas will not conduct exploitation of Denial of Service flaws or other vulnerabilities where a reasonable risk of disruption of business or data loss is perceived without written prior permission.

sales@secureideas.com
+1 (866) 404-7837