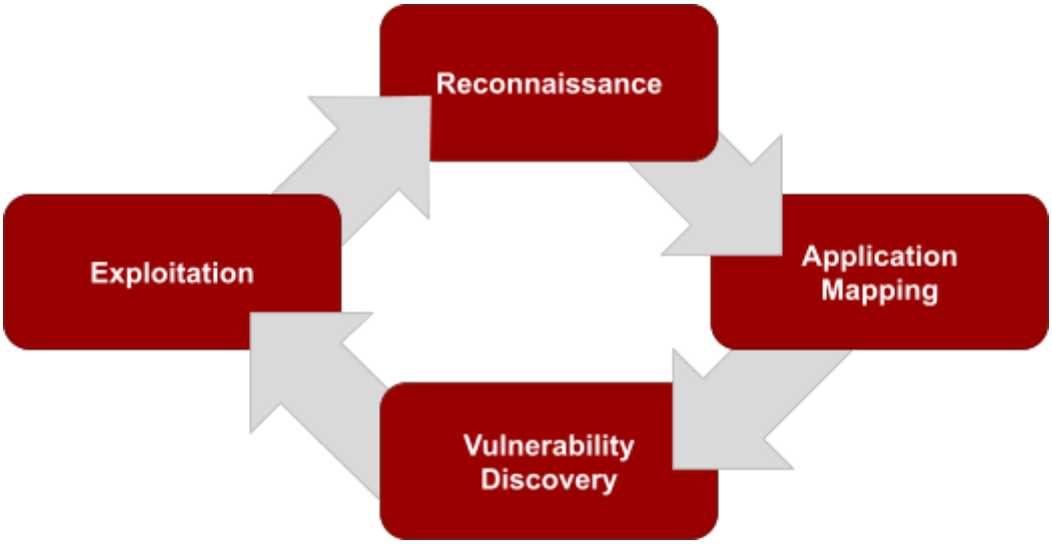
Secure Ideas follows an industry standard methodology for testing the security of mobile applications. This methodology is a four-step process as follows:



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
- Mobile app binaries or source code (e.g. .apk) shared outside of the app stores
- 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:

- Understanding the application architecture (e.g. authentication mechanisms, custom headers, service APIs)

- All fields/forms where information is sent to the server
- All types of data received from the server
- Review of application assets saved to the application's directory on the device, including configuration, local databases, etc...
- Analysis of all HTTP request and response headers used by the application
- Identify custom security measures in addition to platform security controls
- Use of data serialization such as JSON, XML, or Protocol buffers
- 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 mobile 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. Additionally, the application's persistent storage and caching on the device will be examined from multiple contexts including post-logout and after uninstallation of the application. If possible, applications will be decompiled and examined. There are two decoupled portions to the test: the mobile application, and the backing services. Using this information combined with both automated tools and manual testing, Secure Ideas will provide thorough coverage for all common mobile application vulnerabilities and will also seek out less common vulnerabilities depending on the application architecture and features. Testing may be divided into the following categories:

Application

Category	Testing	OWASP Top 10 References
Correct use of Encryption	Scan the application for flaws related to encryption and configuration. This includes: • Use and implementation of encryption for data stored on the device • Use and configuration of SSL	M2, M3, M5
Use of and Integration with Platform Security Features	Using primarily manual techniques, such as analysis of reverse-engineered code to understand platform security API usage. This testing includes items such as: • Use of Platform permissions • Misuse or undermining security of platform features such as biometric identification (TouchID) and credential store (Keychain) • Custom security feature implementations used in place of platform-integrated	M1

	security model	
Code Quality and Protection	Using primarily manual techniques, analyze the application code. This testing includes items such as: • Evaluating code obfuscation • Identifying code flaws presenting exposure to memory corruption flaws • Hardcoded credentials or encryption keys	M7, M8, M9
Injection Flaws	Using automated and manual techniques, analyze the potential for: • SQL injection in uses of local databases • Web app client-side flaws such as cross-site scripting in WebViews • Unsafe handling of IPC parameters	M10, 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

Service API

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	M4, M6, A2, A4, A7, A8
Encrypted Communications and Server Configuration	Scan the application for flaws related to encryption and configuration. This includes: • The use of components with known vulnerabilities	A5, A6, A9

	• Use and configuration of SSL • Exploring web services for other vulnerable points such as an XML parser	
Service API Input Testing	Using a combination of automated and manual techniques, test API call 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 API responses can be influenced	A1

Exploitation:
Exploitation is the final step of the mobile application penetration test. By exploiting the vulnerability, Secure Ideas is able to provide a realistic understanding of the business exposure from the target. 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. Secure Ideas may go beyond a proof of concept if pivoting and post-exploitation is in scope for the test (uncommon). 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.