BISHOPFOX

METHODOLOGY

MOBILE APPLICATION ASSESSMENT

MOBILE APPLICATION ASSESSMENT

Bishop Fox’s mobile application assessment methodology identifies security weaknesses in mobile applications and infrastructure. These zero-, partial-, or full-knowledge assessments begin with the enumeration and analysis of applications deployed within an organization’s infrastructure.

Next, the assessment team uses industry-standard and internally developed tools in conjunction with expert-guided testing techniques to locate mobile-application security deficiencies. After identifying vulnerabilities, the team conducts manual exploitation of the catalogued weaknesses with the intent to compromise sensitive data, credentials, and systems on both the client device and server sides of a mobile deployment.

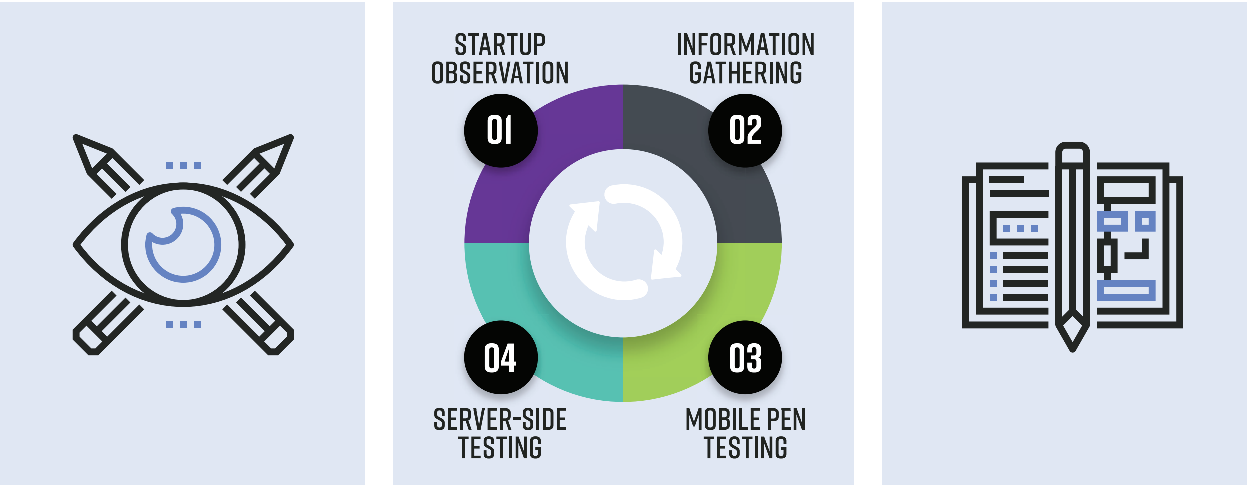
The assessment concludes with a detailed reporting of all security issues discovered within the target environment alongside comprehensive remediation recommendations and steps.

PROCESS OVERVIEW

PHASE 1
PRE-ASSESSMENT

PHASES 2 & 3
DISCOVERY & TESTING

PHASE 4
ANALYSIS & REPORTING



PHASE 1: PRE-ASSESSMENT

The following assessment requirements must be met to ensure the timely and successful completion of the project.

PRE-ASSESSMENT REQUIREMENTS

ASSETS

The assessment team requires the following information prior to the start of a mobile application assessment:

- Client and server architecture design documentation
- Application documentation and dataflow diagrams
- Third-party libraries used in the application, including their respective versions
- Mobile version(s) and hardware (e.g., phone, Kindle, etc.) supported
- Platform used to distribute the application, or the APK and IPA binary file if none
- The Android Studio project or Xcode project and any dependencies, if the mobile application assessment is part of a hybrid application assessment
  > Server-side source code if applicable
- Protocols and processes configured for use within the organization's mobile infrastructure
- Two sets of credentials for each application role

OBJECTIVES

Before the start of fieldwork, Bishop Fox works with the client's team to determine the engagement's scope, restrictions, and primary goals, which often include the following:

- Escalating privileges vertically/horizontally
- Compromising trophies, such as restricted resources, account credentials, or sensitive customer data

DUE CARE

The assessment team performs a review of all pre-assessment information and proposed testing activities to determine their potential for adverse impact against the mobile application and its underlying API and infrastructure.

This review includes the identification of all primary and secondary targets.

PHASE 2: INFORMATION GATHERING

The assessment team conducts in-depth information gathering to assess the application as well as the corresponding services used within the organization's environment.

INFORMATION GATHERING ACTIVITIES

APPLICATION REGISTRATION/ STARTUP OBSERVATION	**The assessment team observes the registration and/or first-time startup process for the mobile application. The team takes the following steps:** • Install the application on a rooted device • Pull the entire application installation directory from the device before first-time startup, using adb pull (Android), iFunBox (iOS), or SCP (iOS) • Set up Man in the Middle (MitM) on the device with Burp Suite Pro or another intercepting proxy • Start up the application using Filemon and/or iSpy/Frida on the device • Capture all network traffic from the device with Wireshark and Burp Suite Pro • Observe all file access and creation with Filemon and/or iSpy
SERVER-SIDE DISCOVERY	**The assessment team scans all servers associated with the mobile application from the following vantage points:** • Internet and web services used to support the application • Third-party connections that the application might use **Application scanning activities include the following:** • Burp scanning of interesting URLs • Common TCP/UDP port scanning • Other content discovery

PHASE 3: MOBILE PENETRATION TESTING

After all pre-assessment requirements have been satisfied and a sufficient amount of information about the environment has been gathered, the assessment team performs the following activities to identify and exploit mobile-specific vulnerabilities within the application.

MOBILE PENETRATION TESTING	
RUNTIME PATCHES	By leveraging hooking, debugging, and runtime patching techniques, the team intercepts, rewrites, and bypasses client-side protections (such as anti-jailbreaking and anti-debugging), as well as explores the application's internals. Runtime attacks are custom-designed on a per-application basis.
NETWORK INTERCEPTION	Using an in-house methodology and toolset, the team intercepts and analyzes client/server network traffic. Where necessary, SSL MitM attacks are leveraged to view and tamper with encrypted data streams. The application traffic is analyzed to identify sensitive information disclosure issues (e.g., Social Security numbers or credit card data).
FILESYSTEM STORAGE	The team scans the device's filesystem for fingerprints left by the client application, with particular attention paid to sensitive data such as credentials, personally identifiable information (PII), encryption keys, and other data that may prove useful to an attacker.
DEVICE KEYSTORE STORAGE	Where possible, the team attempts to recover the information stored within the device's keystore/keychain and manually analyze the data to discover sensitive information.
BINARY REVERSE ENGINEERING	As needed, the client application is reverse-engineered and patched at the binary level to defeat client-side security measures, such as anti-jailbreak detection or license-key verification.

SERVER-SIDE TESTING

API

The assessment team carries out industry-standard web API penetration testing techniques against identified application server deployment(s) with which the mobile client application communicates. **Discovered issues may include:**

- Bypassing authentication and authorization controls
- Injecting arbitrary commands
- Exploiting improper session management
- Identifying data security and encryption weaknesses
- Bypassing client-side validation
- Exploiting query injection and input validation
- Leveraging file transfer capabilities
- Circumventing application and service logic

PHASE 4: ANALYSIS AND REPORTING

Bishop Fox reports feature an executive-level summary of the engagement, which includes the assessment's goals, a synthesis of the highest-impact findings, and high-level recommendations. In addition, within each finding, a vulnerability definition is given alongside detailed reproduction steps, a description of the impact as it pertains to the business, tailored recommendations, and applicable resources.

For each finding, the assessment team builds a holistic view of the business risk by performing the following activities.

TECHNICAL ANALYSIS ACTIVITIES	
LIKELIHOOD DETERMINATION	**For each vulnerability, the assessment team determines the likelihood that it will be exercised based on the following factors:** • Threat-source motivation and capability • Nature of the vulnerability • Existence and effectiveness of controls • Whether physical access to a device and/or a jailbreak is required
IMPACT ANALYSIS	For each potentially successful exploitation of a vulnerability, the assessment team analyzes and determines the impact of such an exercise as it affects the organization and its customers in the areas of confidentiality, integrity, and availability.
SEVERITY DETERMINATION	Bishop Fox determines severity ratings using in-house expertise and industry-standard rating methodologies such as the Open Web Application Security Project (OWASP) and the Common Vulnerability Scoring System (CVSS). The severity of each finding is determined independently of the severity of other findings. Vulnerabilities assigned a higher severity have more significant technical and business impact and achieve that impact through fewer dependencies on other flaws.

PHASE 5: REMEDIATION REVIEW (OPTIONAL)

Optionally, the assessment team re-performs scanning and testing of the identified vulnerabilities after the client indicates that the vulnerabilities have been addressed.