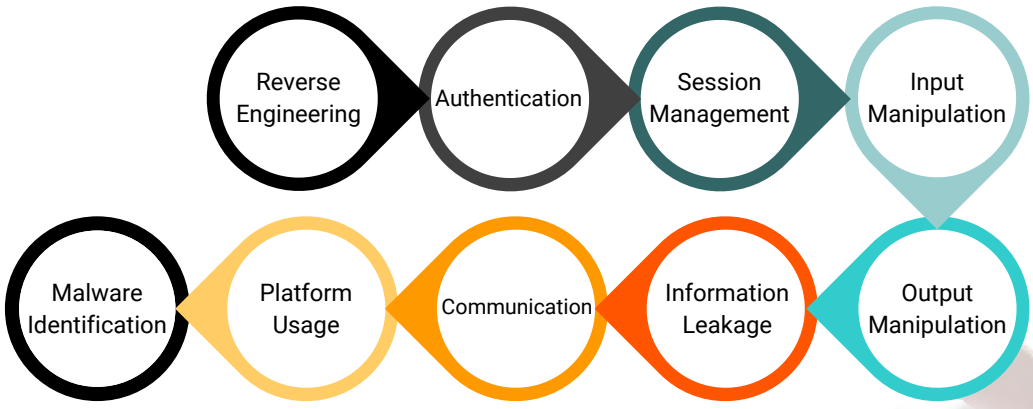
MERIMETSO

Mobile Application Penetration Test

Service Description Introduction

A mobile application penetration test aims to review an entire application. An assessed application will be subjected to a review for vulnerabilities (including those detailed within the OWASP Mobile Top Ten located at https://owasp.org/www-project-mobile-top-10/and the SANS Top 25 list in order to identify any weaknesses that could allow an attacker to compromise the application, the data it interacts with, its users or the hosting environment. Mobile application security testing should be part of all organisation's risk assessment phases. We take mobile application security testing to the highest level, ensuring that a Customer can release their mobile application, knowing it has been extensively scrutinised by industry leaders.

Our Approach



@merimetsocyber
@merimetso
contact@merimetso.net
www.merimetso.net

Engagement Structure

Identified resources will be systematically tested against the OWASP Top 10 and SANS Top 25 list using a combination of automated tools and manual testing. The applications can be developed with any programming languages and technologies. "Black box" and/or "Grey/White box" testing can be used depending on the type of attackers the client wishes to simulate. OS specific issues may be tested for depending on the device e.g. Java exploits in Android and jailbreaking opportunities in iOS etc. There are 9 areas that will be tested against.

Methodology

Reverse-Engineering

Decompose or deconstruct the binary codes using an application appropriate reverse engineering tool. Determine the protocol specification of the server/client application, where appropriate. Determine program logic from the error/debug messages and program behaviour/performance. Attempt to extract any secrets within the application such as API keys, password, private keys etc. Determine if any obfuscation has been used on the application to make reverse engineering more difficult. Identify any debug functionality that maybe present in the mobile application

Authentication

Find possible brute force password guessing access points in the mobile application and server API. Attempt to find valid login credentials by ‘password grinding’ the server API. Check authentication is required on all parts of the server API used by the mobile application Attempt to bypass authentication mechanisms on the server API using spoofed tokens and replying authentication information. Determine application logic to maintain the authentication sessions - number of (consecutive) failure logins allowed, login timeout, etc. Determine the limitations of access control - access permissions, login session duration, idle duration.

Session Management

Determine if session management mechanisms are used by the server API. Attempt to determine the session tokens sequence and format. Determine the session management limitations - bandwidth usages, file download/upload limitations, transaction limitations, etc. Gather excessive information with direct URL, direct instruction, action sequence jumping and/or pages skipping. Inject excess/bogus information with Session-Hijacking techniques.

Input Manipulation

Attempt to inject malformed and malicious input through a range of vectors including fields in the mobile application, local interfaces on the mobile application and directly to the server API. Comprising tests such as: Find the limitations of defined variables, protocol payload - data length, data type, etc. Use long character-strings to find buffer overflows vulnerabilities Concatenate commands in the input strings to find command injection vulnerabilities Inject SQL language in to find SQL injection vulnerabilities in both the mobile application and the server API. Use strings containing path/directory traversal strings to attempt to access unauthorised directory/files with in both the mobile application and the server API. Use a range of differently encoded strings to bypass input filtering and validation on both the mobile application and server API. Attempt to access functionality on the mobile application out of order to fool or modify the logic in the mobile application or server API. Perform requests to the server API out of order and manipulate state sent to the server to fool or modify the logic in server-side web applications if appropriate. Use illogical/illegal input to test the mobile application and sever APU error-handling routines and to find useful debug/error messages.

Output Manipulation

Attempt to modify responses from the server API to the mobile application and from the mobile application to the server API.

Information Leakage

Retrieve valuable information stored in both the mobile application and server API by looking for:

- Incorrect file permissions in the mobile application
- Files stored outside of the mobile application sandbox
- Secrets and credentials stored unencrypted by the mobile application
- Information in server responses and error messages.
- Information contained in the server API banners etc.

Open source intelligence gathering techniques may also be used to discover information on application developers

Communication

Review the communication protocols used between the mobile application and the server Attempt Man-In-the-Middle attacks to gather sensitive information communicated between the mobile application and the server API.

Platform Usage

Look at how the mobile application uses the mobile platform looking for standard types of security issues such as: Incorrectly used APIs Incorrect file permissions

Malware Identification

Decompose or deconstruct the mobile application binary using appropriate reverse engineering tools and analyse the network and I/O activity the any suspicious and possible malware present.

Deliverables

In-depth report, broken down into 3 main parts:

1. Management Summary
2. Technical Overview
3. Detailed Technical Findings

Highlighting the vulnerabilities:

1. Type of RISK
2. The EFFECT of that RISK
3. Recommendations on how to address and mitigate vulnerabilities
4. Estimate of effort required to remediate any vulnerabilities identified

Project Management

Project Management resource will be assigned throughout the engagement but will not be allocated full time.
Key roles and responsibilities:

- Run engagement kick off workshop.
- Run engagement closure.
- Maintain and communicate engagement risks and issues log.
- Weekly status meetings and engagement tracker.

Prerequisites

The following pre-requisites are required from the client:

- Physical access to the device or exported configuration files
- Make and Model of the Firewalls
- Written permission from the client to test
- Login credentials and digital certificates (if applicable)
- Point of Contact details

Deliverable Acceptance Criteria

Merimetso may present interim deliverables to the customer for review throughout the engagement. Should Merimetso present an interim deliverable, the customer will review, and either accept, or document specific corrective items in writing within three (3) business days. In the absence of any comments, deliverables produced by Merimetso will be deemed accepted after three (3) business days.

Limitations & Exclusions

- This engagement covers only the activities detailed in this document. Additional Merimetso offerings may be purchased as add-ons, otherwise additional consulting work not contained in this engagement is deemed out of scope.
- Working time is defined as Monday to Friday (excluding local bank/statutory holidays), 9am-5pm, with a 60-minute break for lunch.
- Merimetso is not responsible for the installation, configuration, or validation of any third-party software, tools, or utilities.
- Only third-party software and infrastructure identified during the pre-sales phase will be in scope for the engagement. There will be no change of scope where third-party software or infrastructure has been changed by the customer during the course of the engagement, or this will incur additional cost.

Customer Responsibilities

1. The customer must provide appropriate resources during the engagement from commencement date.
2. The customer must provide Merimetso representatives with information and resources to ensure Merimetso is able to successfully execute the engagement. This can include, without limitation, providing access and credentials to systems, completing installation prerequisites, providing resources, and attendance in planning, execution, or training meetings.
3. Customer will ensure resources are available in a timely manner to undertake tasks such as change control and documentation review.
4. Customer must ensure it has the necessary escalation and communication channels to resolve any blockers in a timely manner, including dependencies on third parties customer vendors, suppliers, and consultants.
5. If the consultant should travel to a customer location for the delivery of this engagement, there will be additional travel and expense costs. These travel and expense costs can be paid for prior to the engagement, or at actuals, at engagement completion.
6. The customer shall provide at least seventy-two (72) hours' notice for the cancellation or postponement of any work already scheduled as part of the engagement. In such scenario of a cancellation or postponement Merimetso reserves the right at its sole discretion to charge the customer for additional resources at Merimetso's then current daily rate for the delay period.
7. Customer must have a right to use the software and licenses. This requires that the customer must have purchased, or have converted, the appropriate license rights to the target environment.
8. Hardware requirements are the responsibility of the customer, and it is the customer's responsibility to ensure the hardware requirements are met prior to the engagement.

Merimetso may subcontract all or a portion of the services and/or have the services performed by one of its affiliates.
Note: The services described in this Service Description are subject to the terms and conditions by Merimetso Terms

Our Senior Testing Team

Dr Andrew Blyth - Founder and Director of R&D

Dr Blyth received his PhD in Computer Science in 1995 from Newcastle University, UK. Dr Blyth held the position of Professor of Computer Forensics at the University of South Wales, UK, throughout his illustrious career. Co-authored a textbook on Information Assurance, supervised 15 successful PhD candidates, created several BSc and MSc Computer Forensics and Cybersecurity Programmes. He was instrumental in the University of South Wales being the first University in the UK to behave an MSc in Computer Forensics accredited by the UK Government's National Technical Authority (GCHQ/NCSC).

He was the Cybersecurity Technical Lead at the UK Ministry of Defence's (MOD), Defence Science and Technology Laboratory (DSTL). During that time, he created and ran the Computer Network Defence Technical Demonstrator programme and defined various other research programmes within DSTL. He has over 50 peer-review publications and managed millions of dollars of research funding.

Campbell Murray - Founder and COO

Campbell is an established global leader in the field of information security. With over 25 years of experience ranging from offensive security assessments and security engineering in a vast range of environments providing Campbell with rare detail and insight into the field of cybersecurity. Campbell established himself as a penetration testing consultant in 2000. A firm believer in knowledge transfer, Campbell was a founding director of the TigerScheme in 2007 and the Cyber Scheme in 2013.

The Cyber Scheme is the recognised course used by NCSC to affirm CHECK status on any penetration tester who wishes to work on Government infrastructure in the UK. Campbell is now the chair of the technical panel for the Cyber Scheme and is one of only a handful of GCHQ appointed CHECK Team Leader assessors.

David Mound - Founder and CTO

David is a highly experienced Cyber Security Researcher with a demonstrated history of working in the Computer & Network Security industry and threat intelligence. Skilled in Penetration Testing, Reverse Engineering, Computer Forensics, Red Teaming and Threat Intelligence, he has worked with numerous global agencies within the 5 Eyes community, including GCHQ and the NSA delivering, advising on Red Team methods and engagements to assess Blue Team capabilities.

David's current speciality and focus are cloud-based technologies, and he holds multiple cloud certifications such as AWS Certified Security Specialist. David is an ex-Naval officer and is currently a member of the Army Reserves, where he coordinates threat hunting and assists network defence within the UK MOD and Gov estates. David brings his extensive technical capability to design the innovation to the Merimetso Cyber Technical Training platform.