PhishSafe SDK

Technical Documentation

Version 1.0

Last Updated: July 20, 2025

Contents

Introduction and Overview

Business Goals and Objectives

PhishSafe SDK is a behavioral authentication engine designed to detect post-login fraud and phishing threats in mobile applications through continuous behavioral tracking and trust scoring. The system aims to:

- Provide real-time fraud detection using behavioral biometrics
- Reduce account takeover incidents by 85% within first year
- Maintain user privacy with on-device processing
- Integrate seamlessly with existing applications
- Provide actionable security insights through comprehensive dashboards

Target Audience

- **Mobile App Users**: End-users who need protection against phishing
- **Security Teams**: Fraud analysts and security professionals
- **Developers**: Mobile app developers integrating the SDK
- **Product Managers**: Stakeholders evaluating security solutions

Core Functionality

The PhishSafe SDK provides the following key features:

Feature	Description
Session Analytics	Monitors complete user session from login to logout
Location Tracking	Verifies geographical consistency during sessions
Screen Recording Detection	Alerts when screen capture is active
Trust Scoring	Generates real-time risk assessment (0-100 scale)
Data Export	Stores session logs in standardized JSON format
Dashboard Integration	Provides visualization tools for security teams

Installation and Setup

System Requirements

Component	Requirements
Development Environment	Flutter 3.8+, Dart 3.0+
Operating Systems	Android 10+, iOS 14+
Hardware	Minimum 2GB RAM, ARM64/x86 CPU
Permissions	Storage, Location (optional)
Dependencies	device_info_plus, shared_preferences, path_provider

Integration Steps

Adding the Dependency

Add the PhishSafe SDK to your Flutter project:

```
dependencies:
  phishsafe_sdk:
    git:
      url: https://github.com/swrjks/phishsafe_sdk.git
      ref: v2.1
```

Android Configuration

Add these permissions to `AndroidManifest.xml`:

```
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

iOS Configuration

Add these entries to `Info.plist`:

```
<key>NSLocationWhenInUseUsageDescription</key>
<string>For fraud detection purposes</string>
<key>UIBackgroundModes</key>
<array>
  <string>location</string>
</array>
```

Trust Score System

Risk Classification

The SDK calculates a trust score based on user behavior patterns:

Score Range	Risk Level	Description
70-100	Safe	Normal user behavior
40-70	Slightly Risky	Unusual patterns detected
Below 40	Risky	Potential fraudulent activity

Table 2: Trust Score Interpretation

Scoring Factors

The trust score is calculated using:

- Tap patterns and durations
- Swipe gestures and velocities
- Screen navigation sequences
- Session duration characteristics
- Device consistency checks
- Location verification (if enabled)

Module-by-Module Breakdown

1. SessionTracker

Tracks session start and end times.

```
// Call on user login
PhishSafeSDK.initSession();

// Call on user logout
PhishSafeSDK.endSession();
```

2. ScreenRecordingDetector

Automatically detects active screen recording during sessions. Shows warning dialog if detected.

3. DeviceInfoLogger

Collects device information (model, OS version) at session end.

4. TapTracker

Captures tap interactions:

```
PhishSafeSDK.onTap("ScreenName");
```

5. SwipeTracker

Automatically tracks swipe gestures (direction, duration, distance).

6. NavigationLogger

Logs screen visits:

```
PhishSafeSDK.onScreenVisit("Home");
```

7. InputTracker

Monitors sensitive inputs:

```
PhishSafeSDK.recordTransactionAmount("5000");
PhishSafeSDK.recordFDBroken();
PhishSafeSDK.recordLoanTaken();
```

8. LocationTracker

Captures device location once per session (if permissions granted).

9. TrustScoreEngine

Computes risk score based on behavior patterns.

10. ExportManager

Saves session data to:

```
/sdcard/Download/PhishSafe/
```

11. LocalStorage

Helper for storing temporary data using SharedPreferences.

12. PhishSafeTrackerManager

Internal manager coordinating all tracking components.

13. PhishSafeSDK

Main public interface for all SDK functionality.

14. RouteAwareWrapper

Tracks screen durations automatically:

```
RouteAwareWrapper(
  screenName: "Home",
  observer: routeObserver,
  child: HomeScreen(),
);
```

Architecture Overview

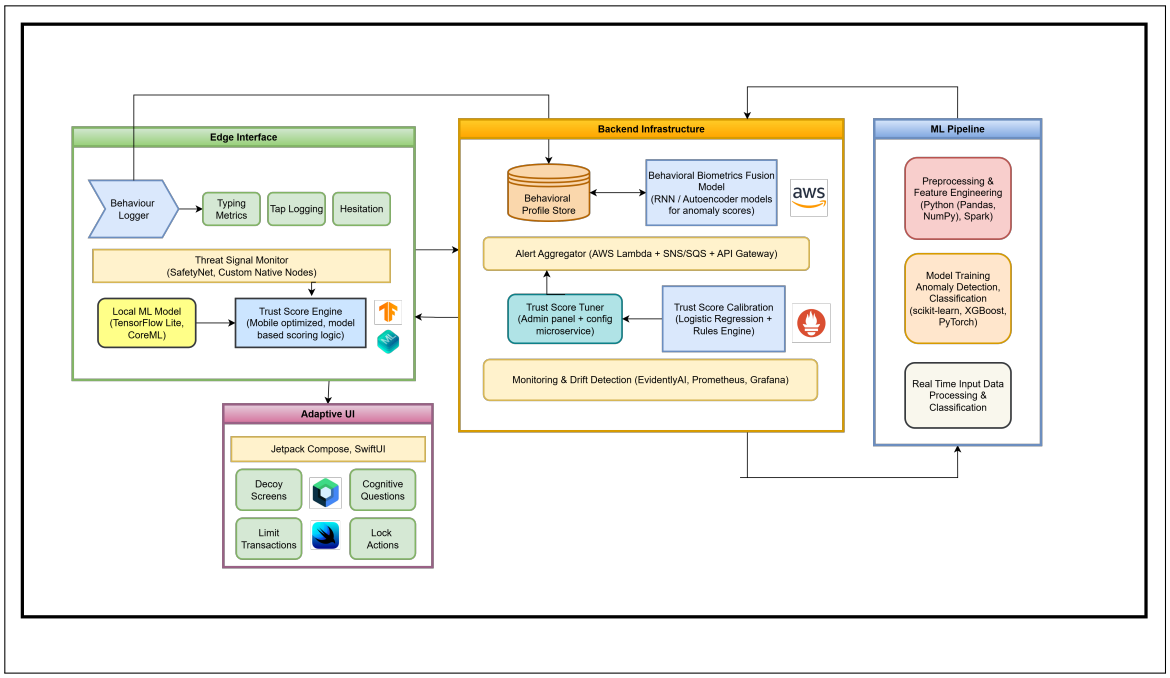

Figure 1: PhishSafe SDK Architecture Overview

The PhishSafe SDK follows a modular architecture with these key components:

- **Data Collection Layer**: Handles user interaction monitoring
- **Analysis Engine**: Processes behavioral patterns
- **Risk Assessment Module**: Calculates trust scores
- **Data Storage**: Manages session logs

Implementation Guide

Basic Integration

Initialize the SDK

Add initialization in your app's main entry point:

```
void main() async {
  WidgetsFlutterBinding.ensureInitialized();

  await PhishSafeSDK.configure(
    PhishSafeConfig(
      enableLocationTracking: true,
      logLevel: LogLevel.info,
    ),
  );

  runApp(MyApp());
}
```

Session Management

Wrap your authentication flow:

```
void loginUser() async {
  try {
    await PhishSafeSDK.initSession();
    // Proceed with login
  } catch (e) {
    // Handle error
  }
}

void logoutUser() {
  PhishSafeSDK.endSession();
  // Clear user session
}
```

Data Storage

All session logs are saved in JSON format at:

```
/sdcard/Download/PhishSafe/
```

Sample log structure:

```
{
  "session_id": "abc123",
  "start_time": "2025-07-20T10:00:00Z",
  "device_info": {
    "model": "Pixel 6",
    "os": "Android 14"
  },
  "trust_score": 85,
  "anomalies": []
}
```