T A
Č R Program Éta TL02000134

Technická dokumentace

Aplikace pro mobilní telefony testující jejich bezpečnost – KonBi

Tomáš Goldmann, Martin Drahanský

Popis aplikace

Jedná se o mobilní aplikaci určenou pro ověřování biometrických vlastností mobilního telefonu s operačním systémem (dále OS) Android. Uživatel aplikace tak může získat informace o tom, jakými biometrickými vlastnostmi mobilní telefon disponuje, které aplikace využívají biometrické rozhraní a jakou verzi operačního systému Android daný mobilní telefon využívá.

Získané informace mohou jednak zvýšit povědomí o biometrii, a především pak uživatele informovat o tom, zdali aplikace, o které „nemá ponětí“ nevyužívá biometrické rozhraní mobilního telefonu. Zjednodušeně lze tvrdit, že hlavním cílem aplikace je zvýšit povědomí o biometrických vlastnostech mobilního telefonu s OS Android a zároveň přinést uživateli informace, které mohou být využité k minimalizaci rizik spojených s využitím biometrické systému, což znamená i zvýšeni bezpečnosti.

Aplikace je pojmenovaná ***KonBi***.

Použité technologie

Aplikace byla napsaná v programovacím jazyce Kotlin[1], který je přímo podporován firmou Google, která vyvíjí operační systém Android. Ačkoliv existují i jiné programovací jazyky pro vývoj aplikací pro OS Android, Kotlin se váže k oficiálnímu SDK pro OS Android. Tím pádem lze naplno využívat biometrické rozhraní.

Biometrické rozhraní operačního systému Android

Ke každé vydané verzi OS Android se váže i specifické API[2], které zprostředkovává nové funkce, mění stávající anebo zprostředkovává funkce dostupné z předchozí verze. Z hlediska biometrického rozhraní došlo k několika výrazným změnám: podpora pro biometrický senzor, konkrétně pro senzor otisků prstů, byla přidána ve verzi API 23, která vyšla společně s OS Android verze 6. Do té doby přicházeli výrobci s vlastními biometrickými podsystémy, které neměly přímou vazbu na struktur OS Android. První verze biometrického API byla především zaměřená na odemykání telefonu pomocí otisku prstů.

V rámci rešerše jsme se zaměřili na zjištění informací o biometrických rozhraních. Vyvinutá aplikace může být přeložena pro OS Android od verze 6.

[1] https://kotlinlang.org
[2] API - Application Programming Interface

Android 6 (API 23) – 2015 – FingerprintManager

Toto Android API je tvořené třídou *FingerprintManager*. Prostřednictvím této třídy lze zjistit, zda je dostupný snímač otisku prstů, zda má uživatel zaregistrovaný otisk prstů a provést autentizaci uživatele. Nutno podotknout, že z hlediska systémové architektury se již využívá zabezpečeného prostředí (TEE).

Z hlediska nastavení biometrických vlastností telefonu nám toto rozhraní poskytuje pouze informaci o tom, zda je přítomen senzor otisků prstů a zda je otisk prstů zaregistrován. Toto lze provést voláním metody `isHardwareDetected()` respektive `hasEnrolledFingerprints()`.

Android 9 (API 28) – BiometricPrompt, BiometricManager

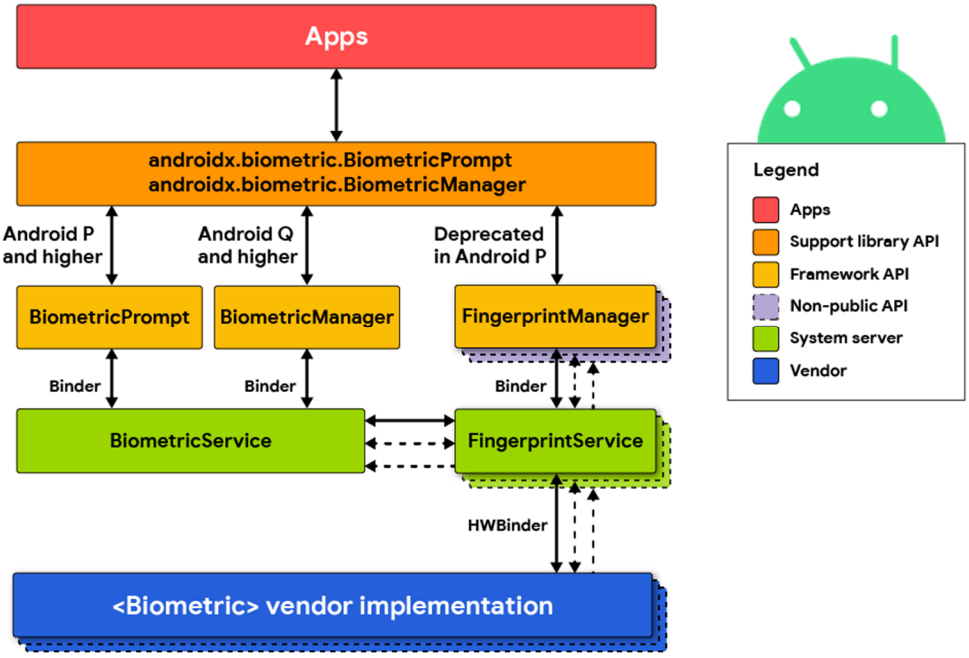
V této verzi API došlo k výrazným změnám v architektuře, a to jak z hlediska bezpečnosti, tak i z hlediska možnosti použití dalších biometrických snímačů. Sada použitelných biometrických snímačů je určena danou verzí API. Například v operačním systému Android 9 výrobci mohli integrovat snímač otisku prstu, senzor duhovky a algoritmy pro rozpoznávání podle obličeje, zatímco ve verzi Android 10 bylo biometrické ověřování omezeno na otisky prstů, duhovku či 3D obličej.

Od verze Android 11 může programátor využívat libovolné dostupné biometrické podsystémy. Je důležité zmínit, že od této verze operační systém třídí biometrické zabezpečení do dvou kategorií, a to do WEAK (slabé) a STRONG (silné) zabezpečení. Každá tato kategorie musí splňovat metriky, které jsou definované v (1). Především se jedná o metriky *Impostor Accept Rate* (IAR) a *Spoof Accept Rate* (SAR). K tomu se váže i protokol (1), podle kterého je zapotřebí provést samotnou verifikaci biometrické senzoru a algoritmů.

Voláním metody *canAuthenticate()* třídy *BiometricManager* můžeme zjistit:

- `BiometricManager.BIOMETRIC_ERROR_NONE_ENROLLED` – je zaregistrovaná biometrika?
- `BiometricManager.BIOMETRIC_ERROR_SECURITY_UPDATE_REQUIRED` – je zapotřebí provést aktualizaci telefonu? (především bezpečnostní aktualizace)
- `BiometricManager.BIOMETRIC_SUCCESS` – je zaregistrována biometrika.

Dále je pak možné přidáním argumentu `BiometricManager.Authenticators.BIOMETRIC_STRONG` nebo `BiometricManager.Authenticators.BIOMETRIC_WEAK` určit, zda se jedná o slabou biometriku nebo o silnou, toto ovšem platí pro Android od verze 11 (2).



Obrázek 1: Schéma biometrického rozhraní od verze Android 9.

Class	Requirements	Capabilities			Constraints	
		Device Unlock	*App Integration* [1]	*Keystore Integration* [2]	*Fallback Timeout*	*More Constraints*
Class 3 (Strong)	SAR: 0-7%, FAR: 1/50K, FRR: 10%, Secure pipeline	✅	✅	✅	72hr	-
Class 2 (Weak)	SAR: 7-20%, FAR: 1/50K, FRR: 10%, Secure pipeline	✅	✅	❌	24hr	4hr idle timeout or 3 incorrect attempts
Class 1 (Convenience)	SAR: >20%, FAR: 1/50K, FRR: 10%, (In)secure pipeline	✅	❌	❌	24hr	4hr idle timeout or 3 incorrect attempts

Obrázek 2: Přehled požadavků na jednotlivé třídy bezpečnostních kategorií (od Android 11) (2).

Struktura aplikace

Z technického hlediska je aplikace rozdělena do třech hlavních tříd (aktivit) a to na:

- *AppsListActivity* – Seznam aplikací, které využívají biometrického zabezpečení;
- *BiometricSettingActivity* – Provádí zjištění verze operačního systému Android a kontrolu nastavení biometrického rozhraní;
- *MainActivity* – hlavní aktivita aplikace, která provádí zjištění dostupnosti biometrických senzoru.

Aplikace dále obsahuje další pomocné třídy, které slouží k lepší struktuře aplikace (například třídy pro položky seznamu).

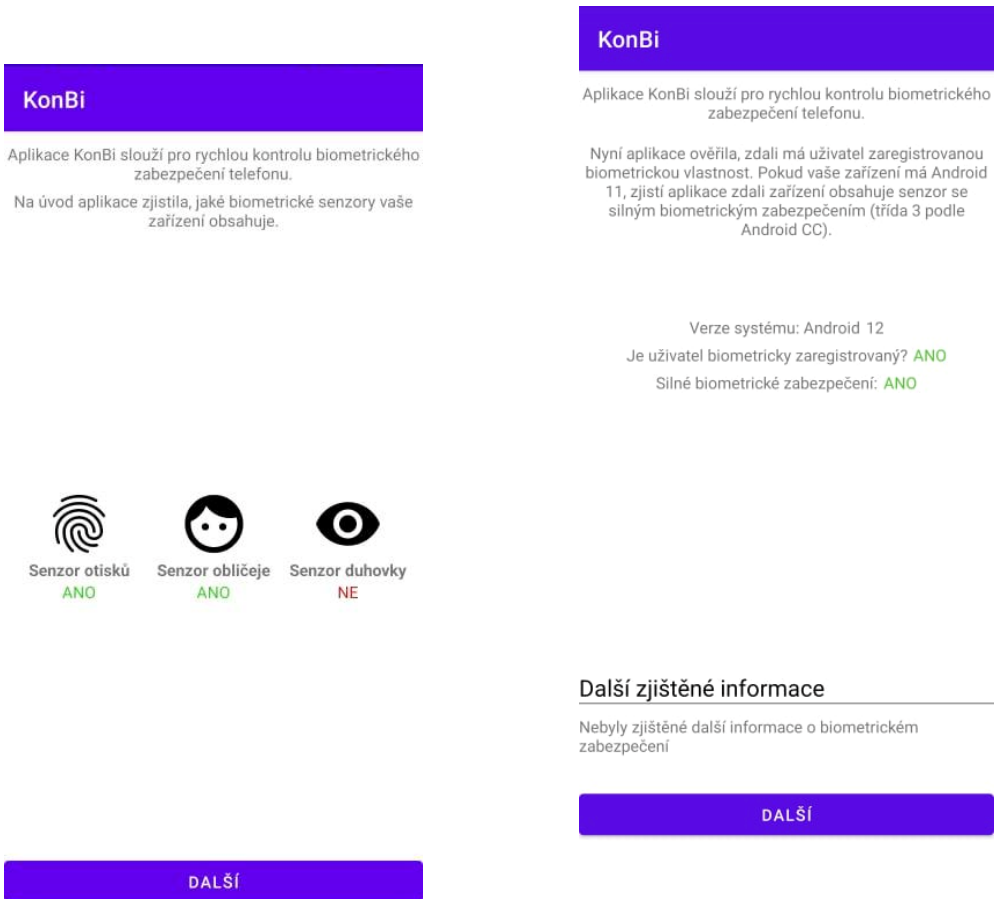
Uživatelské rozhraní

Aplikace se skládá ze tří karet, kdy každá karta obsahuje definovanou sadu informací o biometrickém zabezpečení. Jelikož je aplikace určena pro širokou veřejnost, je kladen důraz na seznámení uživatele se získanými informacemi. Proto je na každé kartě k dispozici krátký textový popis toho, co aplikace zjistila.

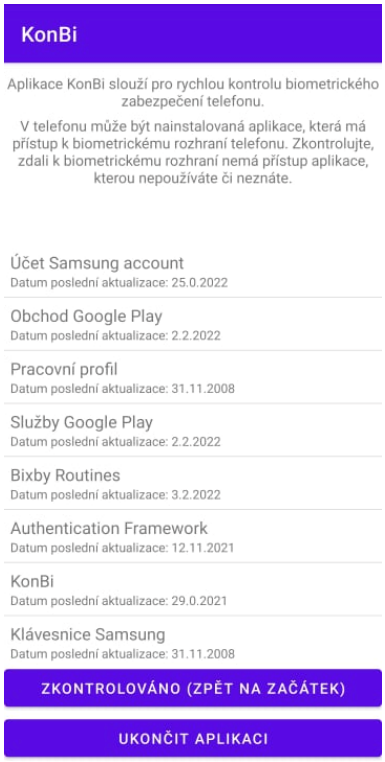
Druhá karta obsahuje informace o tom, zda je zaregistrovaná biometrika a zda telefon disponuje silným biometrickým zabezpečením.

Třetí karta pak obsahuje informace o tom, jaké aplikace využívají biometrické rozhraní.

Celkově je uživatelské prostředí navržené tak, aby k použití aplikace nebyl třeba návod.



Obrázek 3: Karta s informacemi o dostupných senzorech a karta s informacemi o nastavení biometrie.



Obrázek 4: Seznam aplikací využívající biometrické rozhraní.

Instalace

Program je v aktuální verzi dostupný pro Android od verze 9, který používá dle https://www.statista.com/statistics/921152/mobile-android-version-share-worldwide/ drtivá většina uživatelů.

Odkaz na APK: www.fit.vutbr.cz/~igoldmann/projects/precobi/KonBi.apk, příp. na webové stránce produktu https://www.fit.vut.cz/research/product/713/.

Bibliografie

1. **Measuring Biometric Unlock Security : Android Open Source Project.** ***Android Open Source Project.*** **[Online] [Citace: 1. 3 2022.] https://source.android.com/security/biometric/measure.**

2. LOCKSCREEN AND AUTHENTICATION IMPROVEMENTS IN ANDROID 11. [Online] 22. 9 2020. [Citace: 5. 3 2022.] https://android-developers.googleblog.com/2020/09/lockscreen-and-authentication.html.