# José Alfonso Mora Lores

Professional Android Engineer

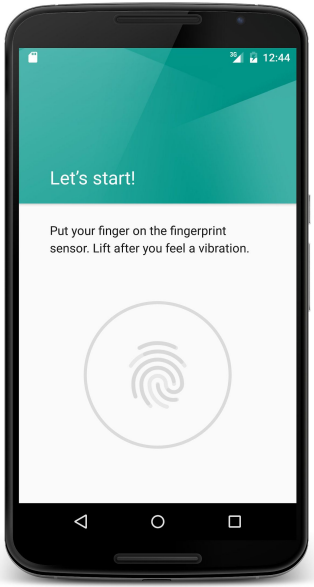

https://twitter.com/josealfonsomora

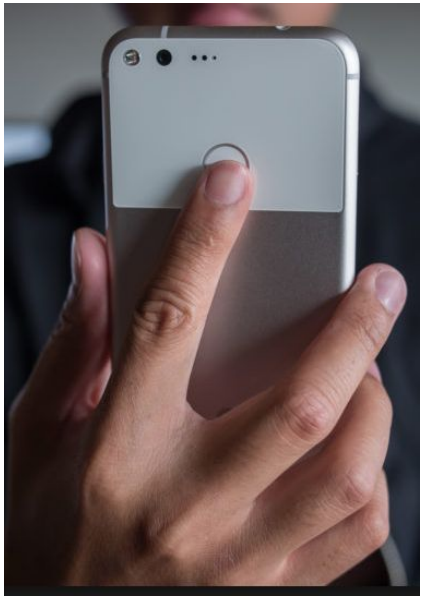
https://www.linkedin.com/in/josealfonsomora

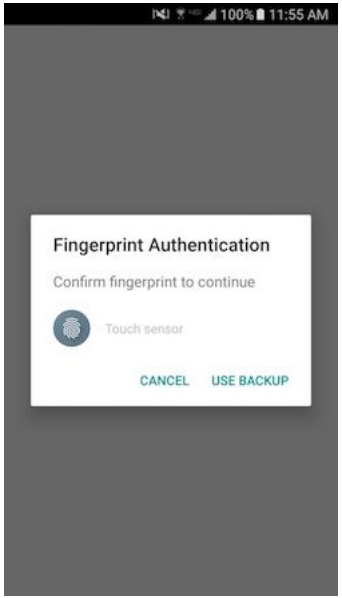
josealfonsomora@gmail.com

# Android P

# Biometrics

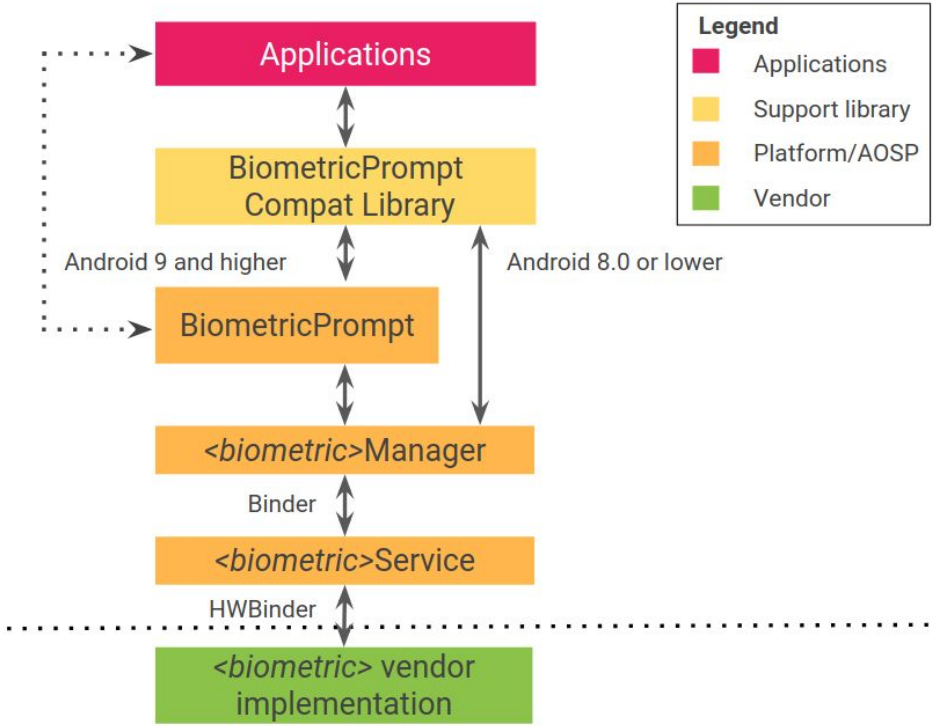





# Fingerprint Authentication

SAMSUNG


Applications
BiometricPrompt
Compat Library
Android 9 and higher
Android 8.0 or lower
BiometricPrompt
<biometric>Manager
Binder
<biometric>Service
HWBinder
<biometric> vendor
implementation
Legend
Applications
Support library
Platform/AOSP
Vendor

**FingerprintManager** added in API level 23
Deprecated since API level 28

**FingerprintManagerCompat** added in version 24.1.0

**BiometricPrompt** added in API level 28

# FingerprintManager

https://developer.android.com/reference/android/hardware/fingerprint/FingerprintManager

# FingerprintManager

https://developer.android.com/reference/android/hardware/fingerprint/FingerprintManager

minSdkVersion 23

# FingerprintManager

https://developer.android.com/reference/android/hardware/fingerprint/FingerprintManager

minSdkVersion 23

```
<uses-permission android:name="android.permission.USE_FINGERPRINT"/>
```

# FingerprintManager

https://developer.android.com/reference/android/hardware/fingerprint/FingerprintManager

minSdkVersion 23

```
<uses-permission android:name="android.permission.USE_FINGERPRINT"/>
```

```
val fingerprintManager = getSystemService(FingerprintManager::class.java)
```

# FingerprintManager

https://developer.android.com/reference/android/hardware/fingerprint/FingerprintManager

minSdkVersion 23

```
<uses-permission android:name="android.permission.USE_FINGERPRINT"/>
```

```
val fingerprintManager = getSystemService(FingerprintManager::class.java)
```

```
fingerprintManager.isHardwareDetected
```

```
fingerprintManager.hasEnrolledFingerprints()
```

# FingerprintManager

https://developer.android.com/reference/android/hardware/fingerprint/FingerprintManager

minSdkVersion 23

```
<uses-permission android:name="android.permission.USE_FINGERPRINT"/>
```

```
val fingerprintManager = getSystemService(FingerprintManager::class.java)
```

```
fingerprintManager.isHardwareDetected
```

```
fingerprintManager.hasEnrolledFingerprints()
```

```
fingerprintManager.authenticate( ... )
```

# FingerprintManager

```
fingerprintManager.authenticate(
        null, // cryptoObject
        cancellationSignal, // cancel listener
        0, // should be 0
        callback, // FingerprintManager.AuthenticationCallback()
        null // Handler to run callbacks
)
```

# FingerprintManagerCompat

https://developer.android.com/reference/androidx/core/hardware/fingerprint/FingerprintManagerCompat

minSdkVersion 23 (or less)

```
<uses-permission android:name="android.permission.USE_FINGERPRINT"/>
```

```
val fingerprintManager = getSystemService(FingerprintManager::class.java)
```

```
fingerprintManager.isHardwareDetected
```

```
fingerprintManager.hasEnrolledFingerprints()
```

```
fingerprintManager.authenticate( ... )
```

# FingerprintManagerCompat

```
fingerprintManager.authenticate(
        null, // cryptoObject
        0, // should be 0
        cancellationSignal, // cancel listener
        callback, // FingerprintManagerCompat.AuthenticationCallback()
        null // Handler to run callbacks
)
```

# BiometricPrompt

https://developer.android.com/reference/android/hardware/biometrics/BiometricPrompt

API 28

```
<uses-permission android:name="android.permission.USE_BIOMETRIC"/>
```

```
private fun biometricPrompt() = BiometricPrompt
    .Builder(this)
    .setTitle("title")
    .setSubtitle("subTitle")
    .setDescription("description")
    .setNegativeButton(
        "Cancel",
        mainExecutor,
        DialogInterface.OnClickListener { _, _ ->
            fingerPrintCallback.text = "description"
        })
    .build()
```

```
biometricPrompt().authenticate(
        cancellationSignal,
        mainExecutor,
        callback
)
```