

Added in [API level 28](#) (/guide/topics/manifest/uses-sdk-element#ApiLevels)

BiometricPrompt

[Kotlin](#) (/reference/kotlin/android/hardware/biometrics/BiometricPrompt) | **Java**

```
public class BiometricPrompt
extends Object (/reference/java/lang/Object)
```

[java.lang.Object](#) (/reference/java/lang/Object)
↳ android.hardware.biometrics.BiometricPrompt

A class that manages a system-provided biometric dialog.

Summary

Nested classes

class [BiometricPrompt.AuthenticationCallback](#) (/reference/android/hardware/biometrics/BiometricPrompt#authenticationcallback)
Callback structure provided to [BiometricPrompt#authenticate\(CancellationSignal, Executor, AuthenticationCallback\)](#) (/reference/android/hardware/biometrics/BiometricPrompt#authenticate).
.

class [BiometricPrompt.AuthenticationResult](#) (/reference/android/hardware/biometrics/BiometricPrompt#authenticationresult)
Container for callback data from [BiometricPrompt.authenticate\(CancellationSignal, Executor, AuthenticationCallback\)](#) (/reference/android/hardware/biometrics/BiometricPrompt#authenticate).
[BiometricPrompt.authenticate\(android.hardware.biometric.AuthenticationPrompt, android.hardware.biometrics.BiometricPrompt.AuthenticationCallback\)](#)

([reference/android/hardware/biometrics/BiometricPrompt#authenticate](#)

.

class **[BiometricPrompt.Builder](#)** ([reference/android/hardware/biometrics/](#)

A builder that collects arguments to be shown on the system-provided b

class **[BiometricPrompt.CryptoObject](#)** ([reference/android/hardware/biom](#)

A wrapper class for the cryptographic operations supported by Biometri

Constants

int **[AUTHENTICATION_RESULT_TYPE_BIOMETRIC](#)**

([reference/android/hardware/biometrics/BiometricPrompt#AUTHENTIC](#)

Authentication type reported by **[AuthenticationResult](#)** ([reference/a](#)
when the user authenticated by presenting some form of biometric (e.g.

int **[AUTHENTICATION_RESULT_TYPE_DEVICE_CREDENTIAL](#)**

([reference/android/hardware/biometrics/BiometricPrompt#AUTHENTIC](#)

Authentication type reported by **[AuthenticationResult](#)** ([reference/a](#)
when the user authenticated by entering their device PIN, pattern, or pas

int **[BIOMETRIC_ACQUIRED_GOOD](#)** ([reference/android/hardware/biometrics/](#)

The image acquired was good.

int **[BIOMETRIC_ACQUIRED_IMAGER_DIRTY](#)**

([reference/android/hardware/biometrics/BiometricPrompt#BIOMETRIC_](#)

The biometric image was too noisy due to suspected or detected dirt on

int **[BIOMETRIC_ACQUIRED_INSUFFICIENT](#)**

([reference/android/hardware/biometrics/BiometricPrompt#BIOMETRIC_](#)

The biometric image was too noisy to process due to a detected condition.
BIOMETRIC_ACQUIRED_IMAGER_DIRTY
(/reference/android/hardware/biometrics/BiometricPrompt#BIOMETRIC_

int **BIOMETRIC_ACQUIRED_PARTIAL** (/reference/android/hardware/biometric)
Only a partial biometric image was detected.

int **BIOMETRIC_ACQUIRED_TOO_FAST** (/reference/android/hardware/biometric)
The biometric image was incomplete due to quick motion.

int **BIOMETRIC_ACQUIRED_TOO_SLOW** (/reference/android/hardware/biometric)
The biometric image was unreadable due to lack of motion.

int **BIOMETRIC_ERROR_CANCELED** (/reference/android/hardware/biometric)
The operation was canceled because the biometric sensor is unavailable.

int **BIOMETRIC_ERROR_HW_NOT_PRESENT** (/reference/android/hardware/biometric)
The device does not have a biometric sensor.

int **BIOMETRIC_ERROR_HW_UNAVAILABLE** (/reference/android/hardware/biometric)
The hardware is unavailable.

int **BIOMETRIC_ERROR_LOCKOUT** (/reference/android/hardware/biometrics)
The operation was canceled because the API is locked out due to too many failed attempts.

int **BIOMETRIC_ERROR_LOCKOUT_PERMANENT**
(/reference/android/hardware/biometrics/BiometricPrompt#BIOMETRIC_)
The operation was canceled because BIOMETRIC_ERROR_LOCKOUT occurred repeatedly.

int **BIOMETRIC_ERROR_NO_BIOMETRICS** (/reference/android/hardware/biometric)
The user does not have any biometrics enrolled.

| | |
|-------------|---|
| int | <u>BIOMETRIC_ERROR_NO_DEVICE_CREDENTIAL</u> (/reference/android/hardware/biometrics/BiometricPrompt#BIOMETRIC_ |
| | The device does not have pin, pattern, or password set up. |
| int | <u>BIOMETRIC_ERROR_NO_SPACE</u> (/reference/android/hardware/biometric: |
| | Error state returned for operations like enrollment; the operation cannot complete the operation. |
| int | <u>BIOMETRIC_ERROR_SECURITY_UPDATE_REQUIRED</u> (/reference/android/hardware/biometrics/BiometricPrompt#BIOMETRIC_ |
| | A security vulnerability has been discovered and the sensor is unavailabl |
| int | <u>BIOMETRIC_ERROR_TIMEOUT</u> (/reference/android/hardware/biometrics/ |
| | Error state returned when the current request has been running too long |
| int | <u>BIOMETRIC_ERROR_UNABLE_TO_PROCESS</u> (/reference/android/hardware/biometrics/BiometricPrompt#BIOMETRIC_ |
| | Error state returned when the sensor was unable to process the current |
| int | <u>BIOMETRIC_ERROR_USER_CANCELED</u> (/reference/android/hardware/bio |
| | The user canceled the operation. |
| int | <u>BIOMETRIC_ERROR_VENDOR</u> (/reference/android/hardware/biometrics/B |
| | OEMs should use this constant if there are conditions that do not fit und appropriate strings for these errors to the <u>BiometricPrompt.Authent</u> <u>CharSequence</u>). (/reference/android/hardware/biometrics/BiometricPrompt.Authenticatic callback. |
| long | <u>BIOMETRIC_NO_AUTHENTICATION</u> (/reference/android/hardware/biome |
| | Returned from <u>BiometricManager#getLastAuthenticationTime</u> (: (/reference/android/hardware/biometrics/BiometricManager#getLastAu authentication for the given authenticator since boot. |

Public methods

| | |
|--|--|
| void | <u>authenticate</u> (/reference/android/hardware/biometrics/BiometricPrompt#authenticate) cancel , <u>Executor</u> (/reference/android/os/CancellationSignal) |
| | This call warms up the biometric hardware, displays a system-provided prompt, and returns the result. |
| void | <u>authenticate</u> (/reference/android/hardware/biometrics/BiometricPrompt#authenticate) (<u>BiometricPrompt.CryptoObject</u> (/reference/android/hardware/biometrics/BiometricPrompt.CryptoObject), <u>BiometricPrompt.AuthenticationCallback</u> (/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback)) |
| | This call warms up the biometric hardware, displays a system-provided prompt, and returns the result. |
| int | <u>getAllowedAuthenticators</u> (/reference/android/hardware/biometrics/BiometricPrompt#getAllowedAuthenticators) |
| | Gets the type(s) of authenticators that may be invoked by the prompt. |
| <u>CharSequence</u> (/reference/java/lang/CharSequence) | <u>getDescription</u> (/reference/android/hardware/biometrics/BiometricPrompt#getDescription) |
| | Gets the description for the prompt, as set by <u>Builder#setDescription</u> (/reference/android/hardware/biometrics/BiometricPrompt.Builder#setDescription). |
| <u>CharSequence</u> (/reference/java/lang/CharSequence) | <u>getNegativeButtonText</u> (/reference/android/hardware/biometrics/BiometricPrompt#getNegativeButtonText) |
| | Gets the negative button text for the prompt, as set by <u>Builder#setNegativeButtonText</u> (/reference/android/hardware/biometrics/BiometricPrompt.Builder#setNegativeButtonText). |
| <u>CharSequence</u> (/reference/java/lang/CharSequence) | <u>getSubtitle</u> (/reference/android/hardware/biometrics/BiometricPrompt#getSubtitle) |
| | Gets the subtitle for the prompt, as set by <u>Builder#setSubtitle</u> (/reference/android/hardware/biometrics/BiometricPrompt.Builder#setSubtitle). |
| <u>CharSequence</u> (/reference/java/lang/CharSequence) | <u>getTitle</u> (/reference/android/hardware/biometrics/BiometricPrompt#getTitle) |
| | Gets the title for the prompt, as set by <u>Builder#setTitle</u> (/reference/android/hardware/biometrics/BiometricPrompt.Builder#setTitle). |
| boolean | <u>isConfirmationRequired</u> (/reference/android/hardware/biometrics/BiometricPrompt#isConfirmationRequired) |
| | Determines if explicit user confirmation is required by the prompt, as set by <u>Builder#setConfirmationRequired</u> (/reference/android/hardware/biometrics/BiometricPrompt.Builder#setConfirmationRequired). |

Inherited methods

From class [java.lang.Object](#) (/reference/java/lang/Object)

| | |
|--|---|
| Object (/reference/java/lang/Object) | <u>clone</u> (/reference/java/lang/Object#clone()) () Creates and returns a copy of this object. |
| boolean | <u>equals</u> (/reference/java/lang/Object#equals(java.lang.Object)) (Object (/reference/java/lang/Object) obj) Indicates whether some other object is "equal to" this one. |
| void | <u>finalize</u> (/reference/java/lang/Object#finalize()) () Called by the garbage collector on an object when garbage collection determines that there are no more references to the object. |
| final Class (/reference/java/lang/Class)<?> | <u>getClass</u> (/reference/java/lang/Object#getClass()) () Returns the runtime class of this Object . |
| int | <u>hashCode</u> (/reference/java/lang/Object#hashCode()) () Returns a hash code value for the object. |
| final void | <u>notify</u> (/reference/java/lang/Object#notify()) () Wakes up a single thread that is waiting on this object's monitor. |
| final void | <u>notifyAll</u> (/reference/java/lang/Object#notifyAll()) () Wakes up all threads that are waiting on this object's monitor. |
| String (/reference/java/lang/String) | <u>toString</u> (/reference/java/lang/Object#toString()) () |

Returns a string representation of the object.

final void

wait (/reference/java/lang/Object#wait(long,%20int))
(**long timeoutMillis**, **int nanos**)

Causes the current thread to wait until it is awakened, typically by being *notified* or *interrupted*, or until a certain amount of real time has elapsed.

final void

wait (/reference/java/lang/Object#wait(long)) (**long timeoutMillis**)

Causes the current thread to wait until it is awakened, typically by being *notified* or *interrupted*, or until a certain amount of real time has elapsed.

final void

wait (/reference/java/lang/Object#wait()) ()

Causes the current thread to wait until it is awakened, typically by being *notified* or *interrupted*.

Constants

AUTHENTICATION_RESULT_TYPE_BIOMETRIC Added in API level 30 ([Guide/Topics/manifest/uses-sdk-element#ApiLevels](#))

```
public static final int AUTHENTICATION_RESULT_TYPE_BIOMETRIC
```

Authentication type reported by [AuthenticationResult](#)

(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationResult) when the user authenticated by presenting some form of biometric (e.g. fingerprint or face).

Constant Value: 2 (0x00000002)

AUTHENTICATION_RESULT_TYPE_DEVICE_CREDENTIAL Added in API level 30 ([Guide/Topics/manifest/uses-sdk-element#ApiLevels](#))

```
public static final int AUTHENTICATION_RESULT_TYPE_DEVICE_CREDENTIAL
```

Authentication type reported by [AuthenticationResult](#)

(</reference/android/hardware/biometrics/BiometricPrompt.AuthenticationResult>) when the user authenticated by entering their device PIN, pattern, or password.

Constant Value: 1 (0x00000001)

BIOMETRIC_ACQUIRED_GOOD

```
public static final int BIOMETRIC_ACQUIRED_GOOD
```

The image acquired was good.

Constant Value: 0 (0x00000000)

BIOMETRIC_ACQUIRED_IMAGER_DIRTY

```
public static final int BIOMETRIC_ACQUIRED_IMAGER_DIRTY
```

The biometric image was too noisy due to suspected or detected dirt on the sensor. For example, it's reasonable return this after multiple [BIOMETRIC_ACQUIRED_INSUFFICIENT](#) (/reference/android/hardware/biometrics/BiometricPrompt#BIOMETRIC_ACQUIRED_INSUFFICIENT) or actual detection of dirt on the sensor (stuck pixels, swaths, etc.). The user is expected to take action to clean the sensor when this is returned.

Constant Value: 3 (0x00000003)

BIOMETRIC_ACQUIRED_INSUFFICIENT

```
public static final int BIOMETRIC_ACQUIRED_INSUFFICIENT
```


The biometric image was too noisy to process due to a detected condition or a possibly dirty sensor (See [BIOMETRIC_ACQUIRED_IMAGER_DIRTY](#) (/reference/android/hardware/biometrics/BiometricPrompt#BIOMETRIC_ACQUIRED_IMAGER_DIRTY)).

Constant Value: 2 (0x00000002)

BIOMETRIC_ACQUIRED_PARTIAL

```
public static final int BIOMETRIC_ACQUIRED_PARTIAL
```

Only a partial biometric image was detected. During enrollment, the user should be informed on what needs to happen to resolve this problem, e.g. "press firmly on sensor." (for fingerprint)

Constant Value: 1 (0x00000001)

BIOMETRIC_ACQUIRED_TOO_FAST

```
public static final int BIOMETRIC_ACQUIRED_TOO_FAST
```

The biometric image was incomplete due to quick motion. For example, this could also happen if the user moved during acquisition. The user should be asked to repeat the operation more slowly.

Constant Value: 5 (0x00000005)

BIOMETRIC_ACQUIRED_TOO_SLOW

```
public static final int BIOMETRIC_ACQUIRED_TOO_SLOW
```

The biometric image was unreadable due to lack of motion.

Constant Value: 4 (0x00000004)

BIOMETRIC_ERROR_CANCELED

```
public static final int BIOMETRIC_ERROR_CANCELED
```

The operation was canceled because the biometric sensor is unavailable. For example, this may happen when the user is switched, the device is locked or another pending operation prevents or disables it.

Constant Value: 5 (0x00000005)

BIOMETRIC_ERROR_HW_NOT_PRESENT

```
public static final int BIOMETRIC_ERROR_HW_NOT_PRESENT
```

The device does not have a biometric sensor.

Constant Value: 12 (0x0000000c)

BIOMETRIC_ERROR_HW_UNAVAILABLE

```
public static final int BIOMETRIC_ERROR_HW_UNAVAILABLE
```

The hardware is unavailable. Try again later.

Constant Value: 1 (0x00000001)

BIOMETRIC_ERROR_LOCKOUT

```
public static final int BIOMETRIC_ERROR_LOCKOUT
```

The operation was canceled because the API is locked out due to too many attempts. This occurs after 5 failed attempts, and lasts for 30 seconds.

Constant Value: 7 (0x00000007)

BIOMETRIC_ERROR_LOCKOUT_PERMANENT

```
public static final int BIOMETRIC_ERROR_LOCKOUT_PERMANENT
```

The operation was canceled because BIOMETRIC_ERROR_LOCKOUT occurred too many times. Biometric authentication is disabled until the user unlocks with strong authentication (PIN/Pattern/Password)

Constant Value: 9 (0x00000009)

BIOMETRIC_ERROR_NO_BIOMETRICS

```
public static final int BIOMETRIC_ERROR_NO_BIOMETRICS
```

The user does not have any biometrics enrolled.

Constant Value: 11 (0x0000000b)

BIOMETRIC_ERROR_NO_DEVICE_CREDENTIAL

```
public static final int BIOMETRIC_ERROR_NO_DEVICE_CREDENTIAL
```

The device does not have pin, pattern, or password set up. See

[BiometricPrompt.Builder#setAllowedAuthenticators\(int\)](#)

([/reference/android/hardware/biometrics/BiometricPrompt.Builder#setAllowedAuthenticators\(int\)](#)),

Authenticators#DEVICE_CREDENTIAL

([/reference/android/hardware/biometrics/BiometricManager.Authenticators#DEVICE_CREDENTIAL](#)), and BiometricManager#canAuthenticate(int).

([/reference/android/hardware/biometrics/BiometricManager#canAuthenticate\(int\)](#)).

Constant Value: 14 (0x0000000e)

BIOMETRIC_ERROR_NO_SPACE

```
public static final int BIOMETRIC_ERROR_NO_SPACE
```

Error state returned for operations like enrollment; the operation cannot be completed because there's not enough storage remaining to complete the operation.

Constant Value: 4 (0x00000004)

BIOMETRIC_ERROR_SECURITY_UPDATE_REQUIRED

```
public static final int BIOMETRIC_ERROR_SECURITY_UPDATE_REQUIRED
```

A security vulnerability has been discovered and the sensor is unavailable until a security update has addressed this issue. This error can be received if for example, authentication was requested with Authenticators#BIOMETRIC_STRONG ([/reference/android/hardware/biometrics/BiometricManager.Authenticators#BIOMETRIC_STRONG](#)), but the sensor's strength can currently only meet Authenticators#BIOMETRIC_WEAK ([/reference/android/hardware/biometrics/BiometricManager.Authenticators#BIOMETRIC_WEAK](#)).

Constant Value: 15 (0x0000000f)

BIOMETRIC_ERROR_TIMEOUT

```
public static final int BIOMETRIC_ERROR_TIMEOUT
```

Error state returned when the current request has been running too long. This is intended to prevent programs from waiting for the biometric sensor indefinitely. The timeout is platform and sensor-specific, but is generally on the order of 30 seconds.

Constant Value: 3 (0x00000003)

BIOMETRIC_ERROR_UNABLE_TO_PROCESS

```
public static final int BIOMETRIC_ERROR_UNABLE_TO_PROCESS
```

Error state returned when the sensor was unable to process the current image.

Constant Value: 2 (0x00000002)

BIOMETRIC_ERROR_USER_CANCELED

```
public static final int BIOMETRIC_ERROR_USER_CANCELED
```

The user canceled the operation. Upon receiving this, applications should use alternate authentication (e.g. a password). The application should also provide the means to return to biometric authentication, such as a "use " button.

Constant Value: 10 (0x0000000a)

BIOMETRIC_ERROR_VENDOR

```
public static final int BIOMETRIC_ERROR_VENDOR
```

OEMs should use this constant if there are conditions that do not fit under any of the other publicly defined constants, and must provide appropriate strings for these errors to the

[BiometricPrompt.AuthenticationCallback#onAuthenticationError\(int, CharSequence\)](#).

[\(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback#onAuthenticationError\(int,%20java.lang.CharSequence\)\)](#)

callback. OEMs should expect that the error message will be shown to users.

Constant Value: 8 (0x00000008)

BIOMETRIC_NO_AUTHENTICATION

```
public static final long BIOMETRIC_NO_AUTHENTICATION
```

Returned from [BiometricManager#getLastAuthenticationTime\(int\)](#)

[\(/reference/android/hardware/biometrics/BiometricManager#getLastAuthenticationTime\(int\)\)](#) when there has been no successful authentication for the given authenticator since boot.

Constant Value: -1 (0xffffffff)

Public methods

authenticate

Added in [API level 28](#) ([/guide/topics/manifest/uses-sdk-element#ApiLevels](#))

```
public void authenticate (CancellationSignal (/reference/android/os/CancellationSignal) cancellationSignal,
                          Executor (/reference/java/util/concurrent/Executor) executor,
                          BiometricPrompt.AuthenticationCallback (/reference/android/hardware/bio
```

This call warms up the biometric hardware, displays a system-provided dialog, and starts scanning for a biometric. It terminates when

[BiometricPrompt.AuthenticationCallback.onAuthenticationError\(int, java.lang.CharSequence\)](#)

[\(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback#onAuthenticationError\(int,%20java.lang.CharSequence\)\)](#)

is called, when

[BiometricPrompt.AuthenticationCallback.onAuthenticationSucceeded\(android.hardware.biometrics.BiometricPrompt.AuthenticationResult\)](#)

[\(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback#onAuthenticationSucceeded\(android.hardware.biometrics.BiometricPrompt.AuthenticationResult\)\)](#)

is called, or when the user dismisses the system-provided dialog. This operation can be canceled by using the provided cancel object. The application will receive authentication errors through [AuthenticationCallback](#)

[\(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback\)](#), and button events through the corresponding callback set in

[BiometricPrompt.Builder.setNegativeButton\(java.lang.CharSequence, java.util.concurrent.Executor, android.content.DialogInterface.OnClickListener\)](#).

[\(/reference/android/hardware/biometrics/BiometricPrompt.Builder#setNegativeButton\(java.lang.CharSequence,%20java.util.concurrent.Executor,%20android.content.DialogInterface.OnClickListener\)\)](#)

. It is safe to reuse the [BiometricPrompt](#)

[\(/reference/android/hardware/biometrics/BiometricPrompt\)](#) object, and calling

[authenticate\(android.os.CancellationSignal, java.util.concurrent.Executor, android.hardware.biometrics.BiometricPrompt.AuthenticationCallback\)](#)

[\(/reference/android/hardware/biometrics/BiometricPrompt#authenticate\(android.os.CancellationSignal,%20java.util.concurrent.Executor,%20android.hardware.biometrics.BiometricPrompt.AuthenticationCallback\)\)](#)

while an existing authentication attempt is occurring will stop the previous client and start a new authentication. The interrupted client will receive a cancelled notification through

[BiometricPrompt.AuthenticationCallback.onAuthenticationError\(int, java.lang.CharSequence\)](#).

[\(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback#onAuthenticationError\(int,%20java.lang.CharSequence\)\)](#)

.

Note: Applications generally should not cancel and start authentication in quick succession. For example, to properly handle authentication across configuration changes, it's recommended to use `BiometricPrompt` in a fragment with `setRetainInstance(true)`. By doing so, the application will not need to cancel/restart authentication during the configuration change.

Requires [Manifest.permission.USE_BIOMETRIC](#)

[\(/reference/android/Manifest.permission#USE_BIOMETRIC\)](#)

Parameters

cancel **CancellationSignal**: An object that can be used to cancel authentication. This value cannot be **null**.

executor **Executor**: An executor to handle callback events. This value cannot be **null**. Callback and listener events are dispatched through this **Executor** (</reference/java/util/concurrent/Executor>), providing an easy way to control which thread is used. To dispatch events through the main thread of your application, you can use **Context.getMainExecutor()** ([/reference/android/content/Context#getMainExecutor\(\)](/reference/android/content/Context#getMainExecutor())). Otherwise, provide an **Executor** (</reference/java/util/concurrent/Executor>) that dispatches to an appropriate thread.

callback **BiometricPrompt.AuthenticationCallback**: An object to receive authentication events. This value cannot be **null**.

Throws

IllegalArgumentException If any of the arguments are null.
(</reference/java/lang/IllegalArgumentException>)

authenticate Added in [API level 28](/guide/topics/manifest/uses-sdk-element#ApiLevels) (</guide/topics/manifest/uses-sdk-element#ApiLevels>)

```
public void authenticate (BiometricPrompt.CryptoObject (/reference/android/hardware/bio
    CancellationSignal (/reference/android/os/CancellationSignal) cancel,
    Executor (/reference/java/util/concurrent/Executor) executor,
    BiometricPrompt.AuthenticationCallback (/reference/android/hardware/bio
```

This call warms up the biometric hardware, displays a system-provided dialog, and starts scanning for a biometric. It terminates when

BiometricPrompt.AuthenticationCallback.onAuthenticationError(int, java.lang.CharSequence)

(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback#onAuthenticationError(int,%20java.lang.CharSequence))

is called, when

BiometricPrompt.AuthenticationCallback.onAuthenticationSucceeded(android.hardware.biometrics.BiometricPrompt.AuthenticationResult)

(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback#onAuthenticationSucceeded(android.hardware.biometrics.BiometricPrompt.AuthenticationResult))

, or when the user dismisses the system-provided dialog, at which point the crypto object becomes invalid. This operation can be canceled by using the provided cancel object. The application will receive authentication errors through **AuthenticationCallback**

(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback), and button events through the corresponding callback set in **Builder#setNegativeButton(CharSequence, Executor, DialogInterface.OnClickListener)**

(/reference/android/hardware/biometrics/BiometricPrompt.Builder#setNegativeButton(java.lang.CharSequence,%20java.util.concurrent.Executor,%20android.content.DialogInterface.OnClickListener))

. It is safe to reuse the **BiometricPrompt**

(/reference/android/hardware/biometrics/BiometricPrompt) object, and calling **BiometricPrompt#authenticate(CancellationSignal, Executor, AuthenticationCallback)**

(/reference/android/hardware/biometrics/BiometricPrompt#authenticate(android.os.CancellationSignal,%20java.util.concurrent.Executor,%20android.hardware.biometrics.BiometricPrompt.AuthenticationCallback))

while an existing authentication attempt is occurring will stop the previous client and start a new authentication. The interrupted client will receive a cancelled notification through **AuthenticationCallback#onAuthenticationError(int, CharSequence)**

(/reference/android/hardware/biometrics/BiometricPrompt.AuthenticationCallback#onAuthenticationError(int,%20java.lang.CharSequence))

.

Note: Applications generally should not cancel and start authentication in quick succession. For example, to properly handle authentication across configuration changes, it's recommended to use BiometricPrompt in a fragment with `setRetainInstance(true)`. By doing so, the application will not need to cancel/restart authentication during the configuration change.

Per the Android CDD, only biometric authenticators that meet or exceed the requirements for **Strong** are permitted to integrate with Keystore to perform related cryptographic operations. Therefore, it is an error to call this method after explicitly calling **Builder#setAllowedAuthenticators(int)**

(/reference/android/hardware/biometrics/BiometricPrompt.Builder#setAllowedAuthenticators(int)) with

any biometric strength other than [Authenticators#BIOMETRIC_STRONG](#) ([/reference/android/hardware/biometrics/BiometricManager.Authenticators#BIOMETRIC_STRONG](#)). Requires [Manifest.permission.USE_BIOMETRIC](#) ([/reference/android/Manifest.permission#USE_BIOMETRIC](#))

Parameters

| | |
|-----------------|--|
| crypto | BiometricPrompt.CryptoObject : A cryptographic operation to be unlocked after successful authentication. This value cannot be null . |
| cancel | CancellationSignal : An object that can be used to cancel authentication. This value cannot be null . |
| executor | Executor : An executor to handle callback events. This value cannot be null . Callback and listener events are dispatched through this Executor (/reference/java/util/concurrent/Executor), providing an easy way to control which thread is used. To dispatch events through the main thread of your application, you can use Context.getMainExecutor() (/reference/android/content/Context#getMainExecutor()). Otherwise, provide an Executor (/reference/java/util/concurrent/Executor) that dispatches to an appropriate thread. |
| callback | BiometricPrompt.AuthenticationCallback : An object to receive authentication events. This value cannot be null . |

Throws

[IllegalArgumentException](#) ([/reference/java/lang/IllegalArgumentException](#)) If any argument is null, or if the allowed biometric authenticators is [Authenticators#BIOMETRIC_WEAK](#) ([/reference/android/hardware/biometrics/BiometricManager.Authenticators#BIOMETRIC_WEAK](#)) and the device is [Build.VERSION_CODES.R](#) ([/reference/android/os/Build.VERSION_CODES.R](#)) or higher, an [IllegalArgumentException](#) is thrown if [Builder#setDeviceCredentialAllowed\(boolean\)](#) ([/reference/android/hardware/biometrics/BiometricPrompt.Builder#setDeviceCredentialAllowed\(boolean\)](#)) is called with `allowed` set to `true`.

(/reference/android/hardware/biometrics/BiometricPrompt.Builder#setAllowedAuthenticators(int)) was explicitly set to true.

getAllowedAuthenticators

Added in [API level 30](#) (/guide/topics/manifest/uses-sdk-element#ApiLevels)

```
public int getAllowedAuthenticators ()
```

Gets the type(s) of authenticators that may be invoked by the prompt to authenticate the user, as set by [Builder#setAllowedAuthenticators\(int\)](#).

(/reference/android/hardware/biometrics/BiometricPrompt.Builder#setAllowedAuthenticators(int)).

Returns

int A bit field representing the type(s) of authenticators that may be invoked by the prompt (as defined by [Authenticators](#) (/reference/android/hardware/biometrics/BiometricManager.Authenticators), or 0 if this field was not set. This value may be **null**.

getDescription

Added in [API level 30](#) (/guide/topics/manifest/uses-sdk-element#ApiLevels)

```
public CharSequence (/reference/java/lang/CharSequence) getDescription ()
```

Gets the description for the prompt, as set by [Builder#setDescription\(CharSequence\)](#) (/reference/android/hardware/biometrics/BiometricPrompt.Builder#setDescription(java.lang.CharSequence)).

Returns

CharSequence The description for the prompt, or null if the prompt has no (/reference/java/lang/CharSequence)description.

getNegativeButtonText Added in [API level 30](/guide/topics/manifest/uses-sdk-element#ApiLevels) (/guide/topics/manifest/uses-sdk-element#ApiLevels)

```
public CharSequence (/reference/java/lang/CharSequence) getNegativeButtonText ()
```

Gets the negative button text for the prompt, as set by

Builder#setNegativeButton(CharSequence, Executor, DialogInterface.OnClickListener)

(/reference/android/hardware/biometrics/BiometricPrompt.Builder#setNegativeButton(java.lang.CharSequence,%20java.util.concurrent.Executor,%20android.content.DialogInterface.OnClickListener))

Returns

CharSequence The negative button text for the prompt, or null if no negative button (/reference/java/lang/CharSequence)text was set.

getSubtitle Added in [API level 30](/guide/topics/manifest/uses-sdk-element#ApiLevels) (/guide/topics/manifest/uses-sdk-element#ApiLevels)

```
public CharSequence (/reference/java/lang/CharSequence) getSubtitle ()
```

Gets the subtitle for the prompt, as set by **Builder#setSubtitle(CharSequence)**.

(/reference/android/hardware/biometrics/BiometricPrompt.Builder#setSubtitle(java.lang.CharSequence))

Returns

CharSequence The subtitle for the prompt, or null if the prompt has no subtitle.
(</reference/java/lang/CharSequence>)

getTitle Added in [API level 30](/guide/topics/manifest/uses-sdk-element#ApiLevels) (</guide/topics/manifest/uses-sdk-element#ApiLevels>)

```
public CharSequence (/reference/java/lang/CharSequence) getTitle ()
```

Gets the title for the prompt, as set by [Builder#setTitle\(CharSequence\)](#).
([/reference/android/hardware/biometrics/BiometricPrompt.Builder#setTitle\(java.lang.CharSequence\)](/reference/android/hardware/biometrics/BiometricPrompt.Builder#setTitle(java.lang.CharSequence))).

Returns

CharSequence The title of the prompt, which is guaranteed to be non-null.
(</reference/java/lang/CharSequence>)

isConfirmationRequired Added in [API level 30](/guide/topics/manifest/uses-sdk-element#ApiLevels) (</guide/topics/manifest/uses-sdk-element#ApiLevels>)

```
public boolean isConfirmationRequired ()
```

Determines if explicit user confirmation is required by the prompt, as set by [Builder#setConfirmationRequired\(boolean\)](#).
([/reference/android/hardware/biometrics/BiometricPrompt.Builder#setConfirmationRequired\(boolean\)](/reference/android/hardware/biometrics/BiometricPrompt.Builder#setConfirmationRequired(boolean))).

Returns

boolean true if explicit user confirmation is required, or false otherwise.

Content and code samples on this page are subject to the licenses described in the [Content License](/license) (</license>).
Java and OpenJDK are trademarks or registered trademarks of Oracle and/or its affiliates.

Last updated 2024-04-11 UTC.