STRONGKEY

StrongKey FIDO Server (SKFS) Sample FIDO Android App

Build Instructions

Copyrights and Notices

StrongAuth, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights—Commercial software. Government users are subject to the StrongAuth, Inc. standard license agreement and applicable provisions of the Federal Acquisition Regulations and its supplements.

This distribution may include materials developed by third parties.

StrongAuth, StrongKey, StrongKey Lite, StrongKey CryptoCabinet, StrongKey CryptoEngine, the StrongAuth logo, the StrongKey logo, the StrongKey Lite logo, the StrongKey CryptoCabinet logo and the StrongKey CryptoEngine logo are trademarks or registered trademarks of StrongAuth, Inc. or its subsidiaries in the U.S. and other countries.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED “AS IS” AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

StrongKey Sample FIDO Android App Preview Release

Building the Application

Once you've downloaded and verified the distribution, unzip SampleFIDOAndroidApp.tgz in a location where you normally work with your Android projects.

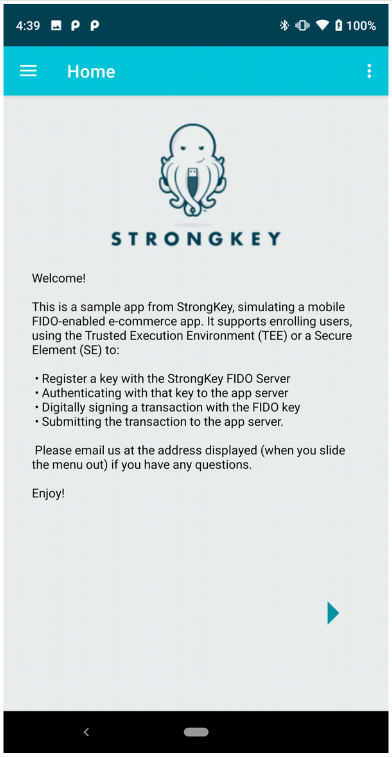
Open the *SampleSACLFidoEcoApp* project. The `settings.gradle` file for the *SampleSACLFidoEcoApp* project must be updated: the *projectDir* variable must reflect where the SACL folder can be found locally on the PC, and then Gradle will connect. Do not update the biometric API; it should be set to 1.0.1.

Wait for Android Studio to synchronize all files it needs. When completed, rebuild the app.

Connect your Android 9+ phone to your development PC; this assumes your phone is already setup in Developer Mode.

Launch the app on your phone. Make sure Android Studio is showing Logcat messages and is filtered on the SFAECO app. You should see the following screens on your phone if all goes well:

1



Welcome!

This is a sample app from StrongKey, simulating a mobile FIDO-enabled e-commerce app. It supports enrolling users, using the Trusted Execution Environment (TEE) or a Secure Element (SE) to:

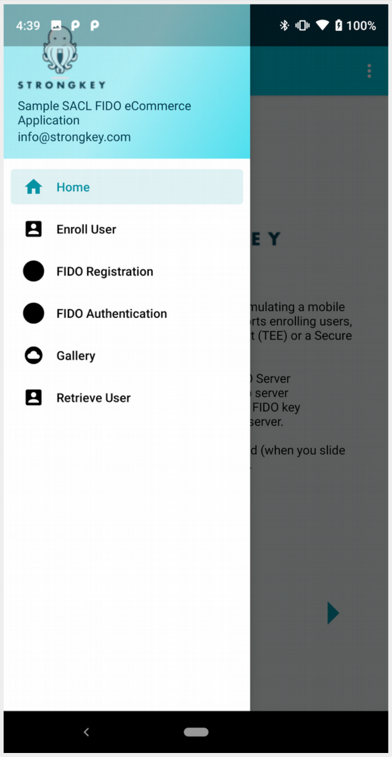
- Register a key with the StrongKey FIDO Server
- Authenticating with that key to the app server
- Digitally signing a transaction with the FIDO key
- Submitting the transaction to the app server.

Please email us at the address displayed (when you slide the menu out) if you have any questions.

Enjoy!

The **Home Page** of the app with a Welcome message.

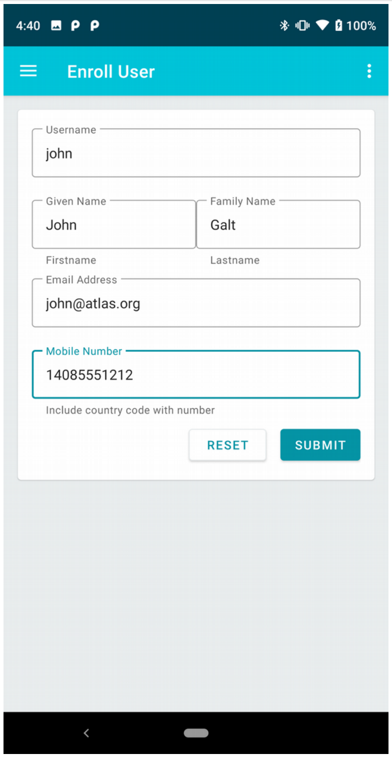
2



Sample SACL FIDO eCommerce Application
info@strongkey.com

The drawer Menu of the app that slides out from the left

3



Username: john

Given Name: John

Family Name: Galt

Firstname

Lastname

Email Address: john@atlas.org

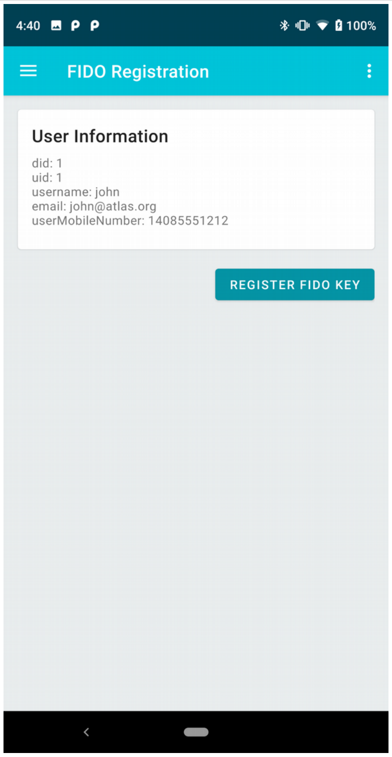
Mobile Number: 14085551212

Include country code with number

The **Enroll User** page where you can create a new user account.

None of the information provided need be real/authentic; the only validation performed is to check the uniqueness of the username, e-mail address, and mobile phone number on the server side—but they can all be fake information.

4



User Information

did: 1
uid: 1
username: john
email: john@atlas.org
userMobileNumber: 14085551212

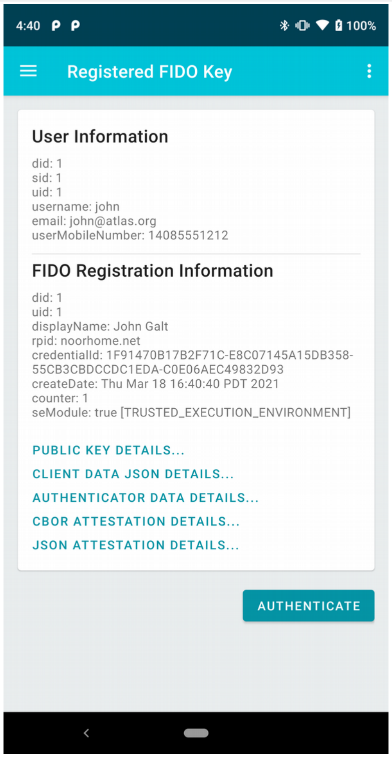
A successfully enrolled user.

Once enrolled, select the **REGISTER FIDO KEY** button to generate your key pair.

If for any reason, you fail to register a key (usually because the network goes to sleep and the TLS connection times out on the app), you can restart the app and choose **FIDO Registration** from the menu. The app will automatically recall the user information last saved on the mobile device.

If by any chance it does have a registered key, information about the registered FIDO key will also be retrieved from RoomDB and displayed.

5


4:40
100%
Registered FIDO Key
PUBLIC KEY DETAILS...
CLIENT DATA JSON DETAILS...
AUTHENTICATOR DATA DETAILS...
CBOR ATTESTATION DETAILS...
JSON ATTESTATION DETAILS...
AUTHENTICATE

User Information

did: 1
sid: 1
uid: 1
username: john
email: john@atlas.org
userMobileNumber: 14085551212

FIDO Registration Information

did: 1
uid: 1
displayName: John Galt
rpid: noorhome.net
credentialId: 1F91470B17B2F71C-E8C07145A15DB358-55CB3CBDCCDC1EDA-C0E06AEC49832D93
createDate: Thu Mar 18 16:40:40 PDT 2021
counter: 1
seModule: true [TRUSTED_EXECUTION_ENVIRONMENT]

A FIDO key successfully registered with the FIDO server.

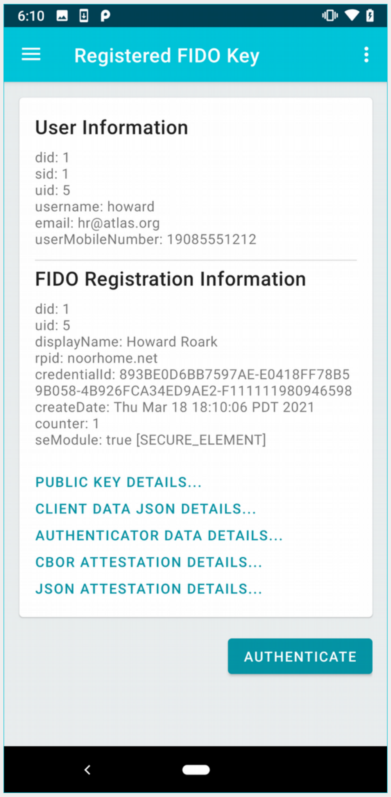
The page displays a fair amount of useful information about the FIDO key and security capability of the mobile device:

- The *relying party ID (RPID)* against which the key is registered
- The credential ID of the FIDO key
- The security capability of the mobile device—in this case, TEE

Each of the highlighted lines in blue provides more detailed information; most of it is not humanly readable, but it is available for developers to see if there is any interest.

Select the **AUTHENTICATE** button to authenticate with the newly registered key to the back-end application.

5A



User Information

did: 1
sid: 1
uid: 5
username: howard
email: hr@atlas.org
userMobileNumber: 19085551212

FIDO Registration Information

did: 1
uid: 5
displayName: Howard Roark
rpid: noorhome.net
credentialId: 893BE0D6BB7597AE-E0418FF78B59B058-4B926FCA34ED9AE2-F111111980946598
createDate: Thu Mar 18 18:10:06 PDT 2021
counter: 1
seModule: true [SECURE_ELEMENT]

PUBLIC KEY DETAILS...
CLIENT DATA JSON DETAILS...
AUTHENTICATOR DATA DETAILS...
CBOR ATTESTATION DETAILS...
JSON ATTESTATION DETAILS...
AUTHENTICATE

This image shows the same information as the previous image—with one difference. Since this app was executing on a phone with a **SECURE ELEMENT (SE)**, the SACL detected this and displays this information when a key is registered.

NOTE: In reality, the SACL does not make this determination. When the FIDO keys are generated, an attestation is also generated by *AndroidKeystore*; this attestation—the Android Key Attestation—generates a chain of X.509 digital certificates rooted in the hardware by the manufacturer of the device (in this case, Google, the manufacturer of the Pixel 3a).

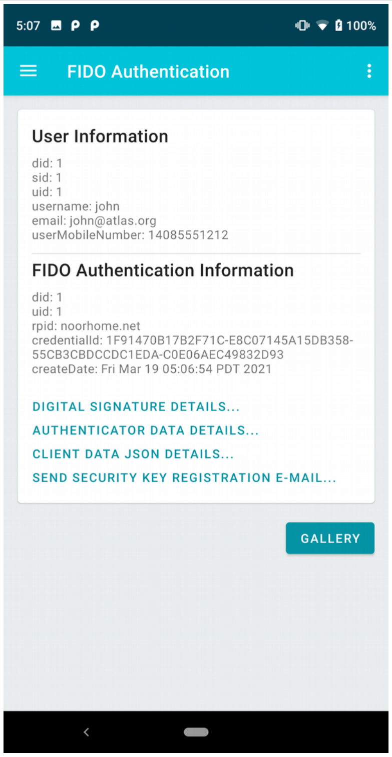
The chain of digital certificates provides information that is parsed by SKFS to determine properties of the public key that was sent for registration and the device in which the key was generated.

That information is relayed back to SACL, which is displayed by the app. This attestation is the assurance a *relying party (RP)* needs to affirm that they are dealing with an authentic device from the manufacturer and the attestation the manufacturer makes about cryptographic keys within that device.

While far more security capabilities are possible based on this technology, this SACL Preview Release is focused on enabling just what the FIDO2 protocol needs to satisfy business requirements.

StrongKey is committed to taking this technology and capability to its fullest potential, delivering some of the strongest security capability with its Tellaro appliance and securing some of the highest-risk business transactions. Please contact us if you would like to learn more.

6


5:07
100%
FIDO Authentication

User Information

did: 1
sid: 1
uid: 1
username: john
email: john@atlas.org
userMobileNumber: 14085551212

FIDO Authentication Information

did: 1
uid: 1
rpid: noorhome.net
credentialId: 1F91470B17B2F71C-E8C07145A15DB358-55CB3CBDCCDC1EDA-C0E06AEC49832D93
createDate: Fri Mar 19 05:06:54 PDT 2021

DIGITAL SIGNATURE DETAILS...
AUTHENTICATOR DATA DETAILS...
CLIENT DATA JSON DETAILS...
SEND SECURITY KEY REGISTRATION E-MAIL...
GALLERY

A successfully authenticated user.

The page displays some useful information about the authentication.

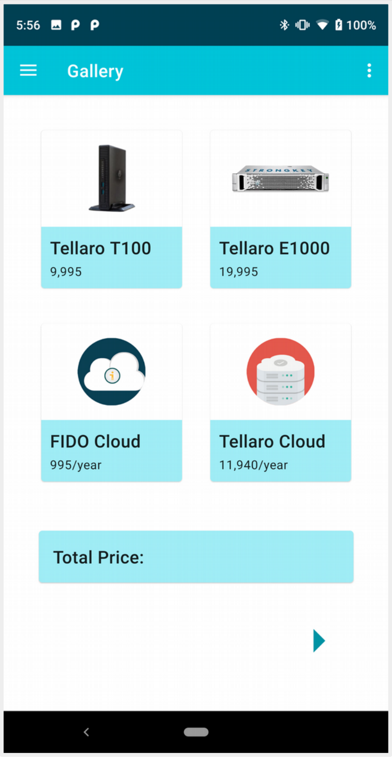
One useful capability (currently not implemented in this Preview) is a prompt to *SEND SECURITY KEY REGISTRATION E-MAIL*.

When a user registers with a site using their mobile device, this FIDO key becomes the strongest way in which the user may interact with the site. Once authenticated, it will be helpful to allow users to email themselves a one-time, time-limited link that allows them to register a second FIDO key using an external *Security Key*.

This has the advantage of allowing the user to use the site from their desktop or laptop, or to allow them to recover their account with a new phone if they lose their current mobile device.

Once authenticated, select the GALLERY button to choose a sample product for purchase.

7

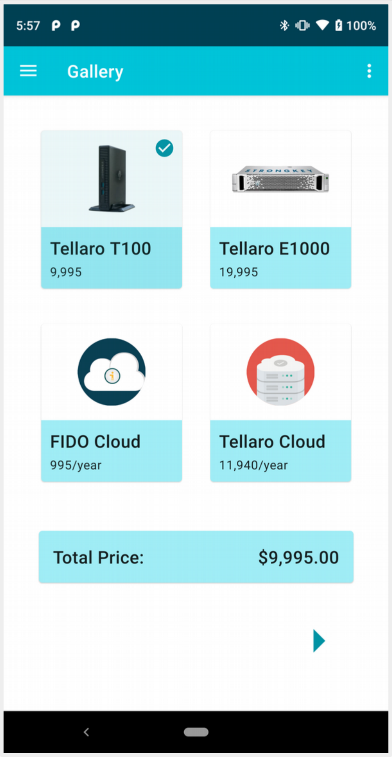


Tellaro T100 9,995	Tellaro E1000 19,995
FIDO Cloud 995/year	Tellaro Cloud 11,940/year

Total Price:

The **Product Gallery** displays 4 sample products in tiles that can be selected (or deselected, once selected) as part of the interaction.

8



Tellaro T100 9,995

Tellaro E1000 19,995

FIDO Cloud 995/year

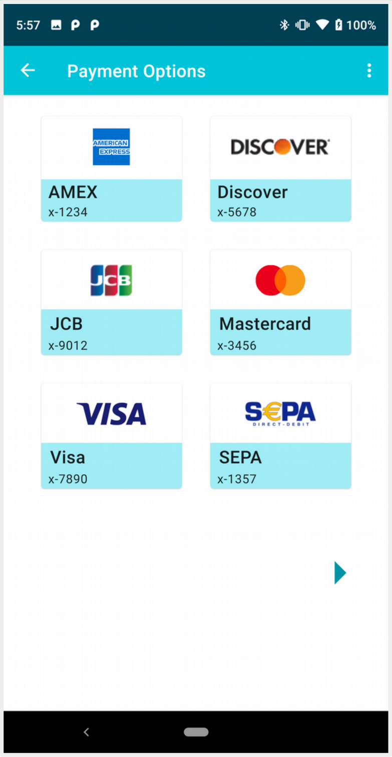
Tellaro Cloud 11,940/year

Total Price: $9,995.00

This page shows one of the tiles selected in this demonstration.

The arrow at the bottom right advances to next page.

9



AMEX x-1234

Discover x-5678

JCB x-9012

Mastercard x-3456

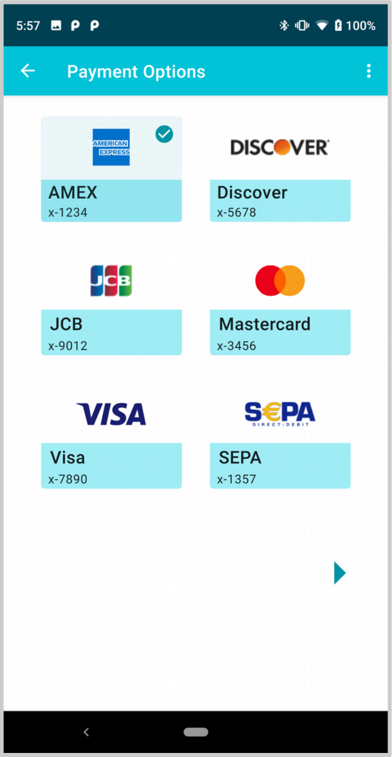
Visa x-7890

SEPA x-1357

The **Payment Options page** displays six sample payment cards in tiles that can be selected\deselected as part of the interaction.

This assumes the app was designed with the capability to store multiple payment options with this site. In this sample app, hard-coded values are used to simulate this function.

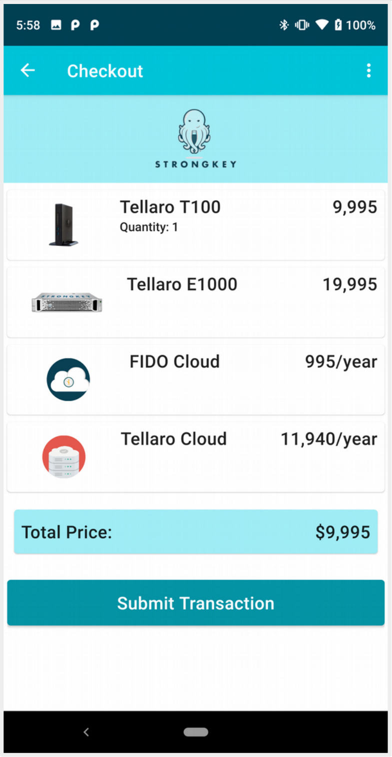
10



This page shows one of the tiles selected in this demonstration.

The arrow at the bottom right advances to next page.

11



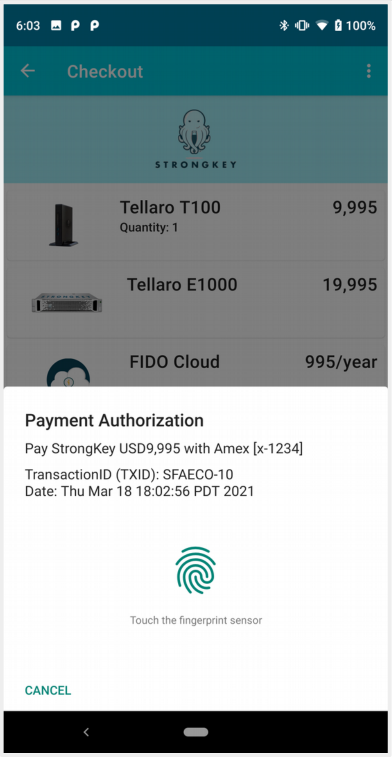
Tellaro T100 Quantity: 1	9,995
Tellaro E1000	19,995
FIDO Cloud	995/year
Tellaro Cloud	11,940/year
Total Price:	$9,995

We finally arrive at the **CHECKOUT** page of the sample app.

It shows the which of the four products was chosen from the Product Gallery, and the *Total Price* the user is expected to pay to consummate this transaction.

The **SUBMIT TRANSACTION** is where the magic of FIDO *Transaction Authorization (TXA)* begins within SACL.

12



Payment Authorization

Pay StrongKey USD9,995 with Amex [x-1234]

TransactionID (TXID): SFAECO-10
Date: Thu Mar 18 18:02:56 PDT 2021

Having selected the **SUBMIT TRANSACTION** button, the app takes relevant information about the transaction and sends it to the back office application.

The back office application, in turn, processes that information and acquires a unique challenge from the FIDO Server, which takes this transaction's details into account.

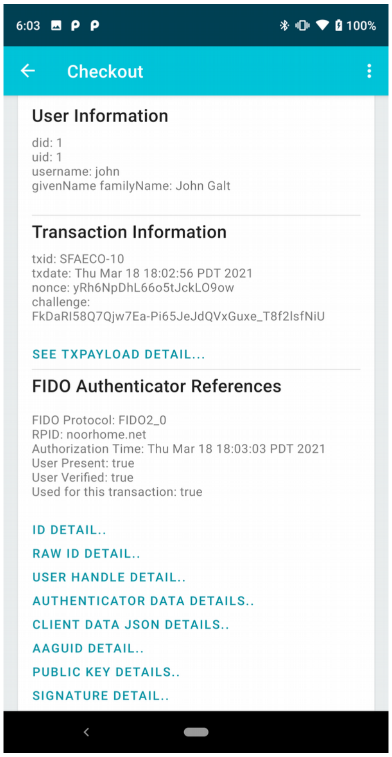
Upon receiving that challenge, the app calls Android's *BiometricPrompt* to display a unique message—a *dynamic link* in PSD2 RTS parlance—that displays necessary information for regulator compliance:

- Payee information
- The amount to be paid, and
- The chosen payment option

The *transaction ID (TXID)* and a Timestamp are added by the app to uniquely identify this transaction within its database.

The user is explicitly prompted to supply their fingerprint to authenticate with the device using the FIDO key to digitally sign the challenge sent by the FIDO server (through the back office application).

13



User Information

did: 1
uid: 1
username: john
givenName familyName: John Galt

Transaction Information

txid: SFAECO-10
txdate: Thu Mar 18 18:02:56 PDT 2021
nonce: yRh6NpDhL66o5tJckLO9ow
challenge:
FkDaRI58Q7Qjw7Ea-Pi65JeJdQVxGuxe_T8f2lsfNiU

FIDO Authenticator References

FIDO Protocol: FIDO2_0
RPID: noorhome.net
Authorization Time: Thu Mar 18 18:03:03 PDT 2021
User Present: true
User Verified: true
Used for this transaction: true

Upon successfully authenticating to the Android phone, the app—using SACL—uses the FIDO key to digitally sign the unique challenge and confirms the transaction displayed on the secure display.

This page shows the successful transaction with 2 interesting pieces of information:

- The SEE TXPAYLOAD DETAILS is Base64-encoded data of a JSON object with the following details:
  - Merchant name
  - Currency type
  - Total price
  - Card brand
  - Last 4 digits of the card
  - The unique transaction ID
  - The date/time of the transaction
- The information at the bottom, referenced as **FIDO Authenticator References**, refers to data elements standardized by the FIDO Alliance and EMVCo to transmit FIDO-authenticated transaction confirmations over 3DS messages to Issuing Banks for authorization. The information displayed here maps to the details specified in the FIDO Alliance's FIDO Authentication and EMV 3-D Secure—Using FIDO for Payment Authentication white paper.

This concludes the demonstration of the SFAECO app and its use of SACL for using FIDO2 protocols.

SACL can be used for applications in finance, healthcare, education, government, gaming, enterprise apps, etc. While mobile devices have made it significantly easier for users to authenticate to websites (by using biometrics), behind the curtain they still use the ancient password-based authentication schemes that are susceptible to attack and are responsible for more than 95% of all data breaches.

With SACL and SKFS, Android apps can now forever leave passwords behind. We hope you find this opportunity exciting to protect your users, as well as your own sites, by eliminating passwords and all the hassles that come with them.

Drop us a note at getsecure@strongkey.com to tell us what you think.

Enjoy!

STRONGKEY

20045 Stevens Creek Boulevard Suite 2A
Cupertino, CA 95014
USA