# FIDO Alliance White Paper:

## PSD2 Support: Why Change to FIDO

July 2020

**Editors:**
Marc Findon, Nok Nok
Bernard Joly, Onespan
Alain Martin, Thales

# Contents

# Tables

# Figures

# 1. Introduction

Banks in Europe have deployed customer authentication solutions for several years. These solutions have served their purpose well and enabled customers to safely log in to their bank accounts. In the world of e-commerce, these solutions, when used, have been successful in combatting online payment fraud.

The Second Payment Services Directive (PSD2) and its associated Regulatory Technical Standards (RTS) dramatically change the payment landscape. It is interesting to consider if legacy authentication solutions are still adequate considering:

- The mandate for strong, multi-factor authentication,
- The emergence of Third Party Providers (TPP) accessing accounts via open APIs, which impacts the customer journey.

User convenience will determine the success of PSD2 as well as the continued growth of e-commerce in the context of PSD2. Balancing user convenience with security obligations, while maximizing reach, is a challenge; banks may want to evaluate how well their legacy authentication solutions meet this new need.

FIDO authentication standards provide an answer to this question, but is the change from a legacy method to FIDO worthwhile? This paper proposes guidance to banks to help them decide.

First, this paper will provide an overview of FIDO authentication principles and then examine the pros and cons of the main existing authentication methods used to access an account or secure an online payment, comparing them with the FIDO approach.

# 2. FIDO Essentials

## 2.1 FIDO Authentication

The figure below illustrates the basic two-step user authentication mechanism provided by the FIDO standards. The figure also maps PSD2 terminology with terminology used in the FIDO standards:

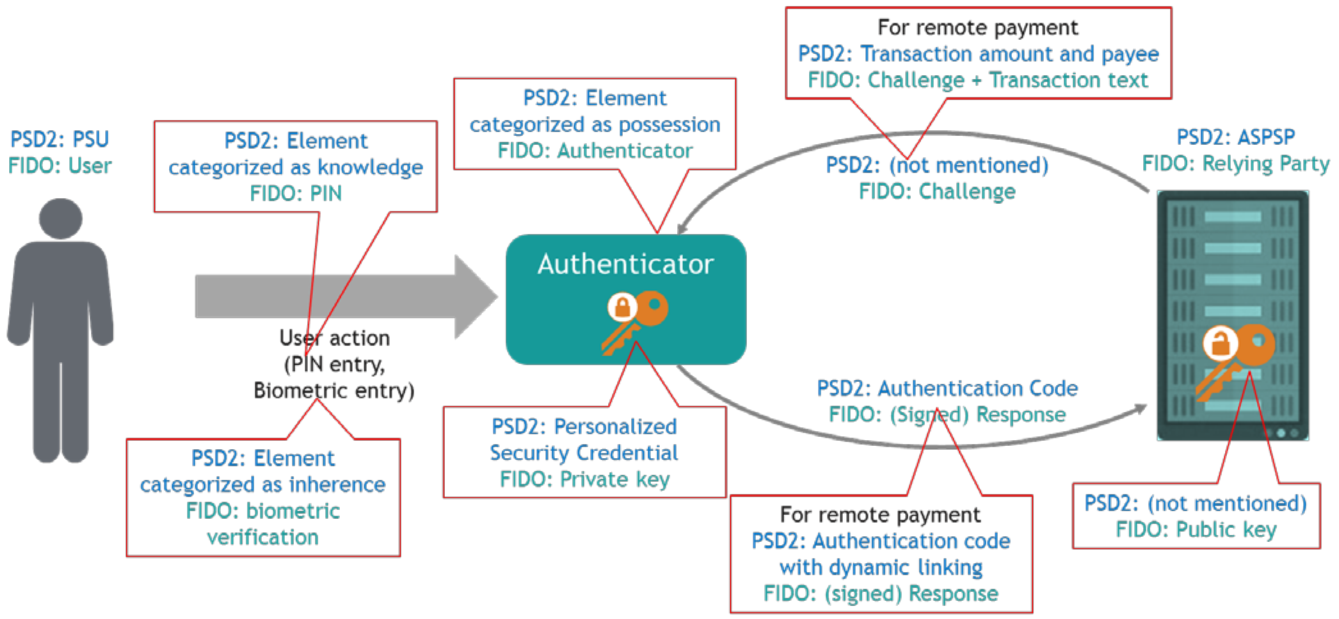

PSD2: PSU
FIDO: User
PSD2: Element categorized as knowledge
FIDO: PIN
PSD2: Element categorized as possession
FIDO: Authenticator
For remote payment
PSD2: Transaction amount and payee
FIDO: Challenge + Transaction text
PSD2: ASPSP
FIDO: Relying Party
PSD2: (not mentioned)
FIDO: Challenge
Authenticator
User action
(PIN entry,
Biometric entry)
PSD2: Authentication Code
FIDO: (Signed) Response
PSD2: Element categorized as inherence
FIDO: biometric verification
PSD2: Personalized Security Credential
FIDO: Private key
For remote payment
PSD2: Authentication code with dynamic linking
FIDO: (signed) Response
PSD2: (not mentioned)
FIDO: Public key

***Figure 1 – How FIDO Works***

To authenticate with FIDO, the Payment Service User (PSU) must have a FIDO authenticator that can either be integrated in a general purpose device (e.g. smartphone, laptop) or be a separate device (e.g. security key, smart card).

## User verification

The first step of FIDO authentication is the user verification step that is performed offline, locally, by the authenticator. This user verification step can be:

- A verification of user presence whereby the user makes a gesture with the authenticator (for example, touches a security key or taps an NFC card on a reader).
- The verification of a PIN code or of biometric data by the authenticator. In this case, the local user verification constitutes one of the authentication factors mandated by the RTS.

The fact that the user verification data (PIN code or biometric data) is stored in the authenticator, verified locally and never transmitted to or shared with servers is a strong asset of the FIDO approach. As such, FIDO implements the privacy design requirement of the General Data Protection Regulation.

The local user verification step is a prerequisite for the online authentication step.

## Online authentication

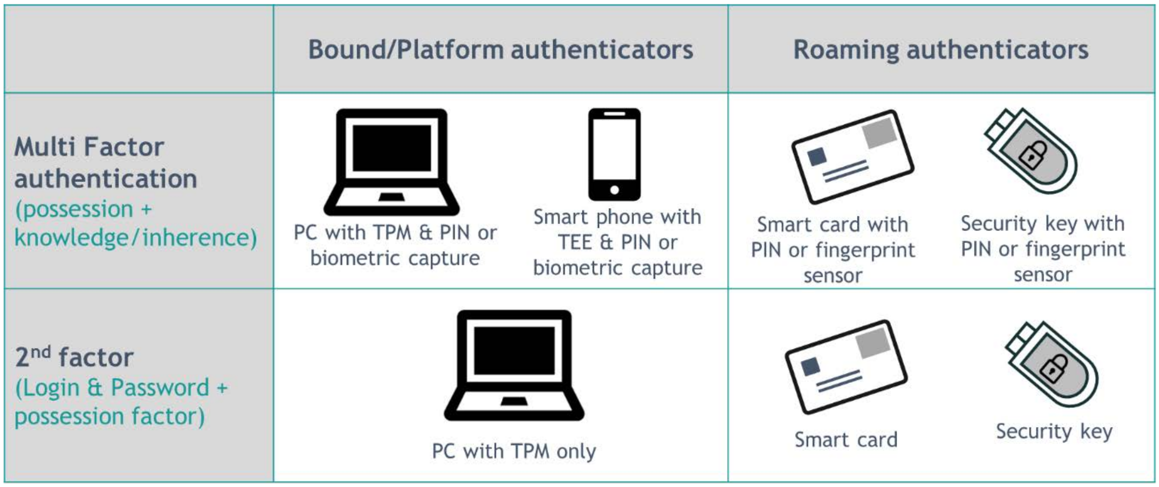
The online authentication step proves the possession of the FIDO authenticator and constitutes a second factor of authentication mandated by PSD2. In this step, the ASPSP (Account Servicing Payment Service Providers) server sends a challenge message to the authenticator which is then cryptographically signed by a private key stored in the authenticator. The signed response is returned to the ASPSP and its positive verification serves as proof of possession.

FIDO standards are based on public key cryptography. The private key is the Personalized Security Credential described in the RTS. It is part of a key pair randomly generated by the authenticator itself and is not known to any other party. At the time of generation, the associated public key is sent to the ASPSP in a protected way.

The authenticator maintains dedicated Personalized Security Credentials (private keys) for each ASPSP. For example, if the PSU has accounts at ASPSP1 and ASPSP2, the authenticator would store different Personalized Security Credentials for ASPSP1 and ASPSP2, each being restricted for use with its respective ASPSP.

## 2.2 Authenticators

FIDO authenticators exist in several implementations and are classified as shown in the table below:



| | Bound/Platform authenticators | Roaming authenticators |
|---|---|---|
| Multi Factor authentication (possession + knowledge/inherence) | PC with TPM & PIN or biometric capture<br>Smart phone with TEE & PIN or biometric capture | Smart card with PIN or fingerprint sensor<br>Security key with PIN or fingerprint sensor |
| 2nd factor (Login & Password + possession factor) | PC with TPM only | Smart card<br>Security key |

***Table 1 – Examples of FIDO Authenticators***

## 2.3 FIDO Standards

**FIDO UAF** (Universal Authentication Framework) is a FIDO standard that completely replaces the use of passwords. FIDO UAF-compliant authenticators support the local verification of the user's PIN code or biometric data. Typical UAF implementations are found in smartphones.

**FIDO U2F** (Universal Second Factor) is a FIDO standard that adds an authenticator, the possession factor, to an existing log-in + password authentication method. U2F devices are typically USB security keys.

**FIDO2** is a FIDO standard that consists of **WebAuthn**, a set of web APIs specified by the W3C organization in collaboration with the FIDO Alliance, that are natively incorporated in recent browsers, and **CTAP**, a communication protocol to connect to FIDO authenticators. Much like UAF, FIDO2 enables completely replacing passwords not only on mobile devices, but also on desktops and laptops when configured with appropriate security devices. Additionally, U2F devices are compatible with CTAP. Collectively, WebAuthn and CTAP standardize access from a browser on a platform (a PC or mobile device) to a FIDO authenticator.

A high level description of these standards as well as FIDO specifications can be found at: https://fidoalliance.org/download/.

# 3. SMS OTP Combined with Password

An SMS One Time Passwords (OTP) is a passcode, typically 4 to 6 digits, sent to a user via an SMS service to their mobile phone. SMS OTPs are probably the most common form of second factor authentication used in the banking sector. Their popularity rose in a time where the industry began to realize that passwords and usernames were no longer sufficient to protect user accounts, sensitive data and assets.

The process works as follows:

1. A user registers a mobile phone number with their bank, following some form of KYC process.
2. When the user accesses an online banking service or makes a financial transaction, the bank will generate an OTP and send it to the user's mobile phone by SMS. For payments, the OTP may be generated using transaction amount and merchant identification.
3. The user acknowledges the SMS OTP and enters the digits into the appropriate field in the app or website of the bank.
4. The entered digits are matched with the generated OTP, a successful match authenticating the user.

The SMS OTP will have a limited period of validity, which is usually around ten minutes for most banking applications. Once this time has passed, the OTP will be invalid and the user will need to request the generation of a new one.

There may be some variance depending on the solution, e.g. if the SMS OTP fails, there may be a voice backup whereby an Interactive Voice Response (IVR) system will call the user with the OTP. IVR OTP has some alternative weaknesses including scammers forwarding phone numbers by accessing the consumer's online account using the phone number and a potentially weak or re-used online password. The scammer receives the OTP at their number and un-forwards the victim's phone number without their knowledge.

## Compliance with PSD2

The European Banking Authority (EBA) issued an opinion[1] confirming that SMS OTPs are an acceptable authentication factor, adding that OTP verification proved possession of the SIM card within the mobile phone; they are a possession factor and alone are not sufficient to comply with the SCA mandate for multifactor authentication.

For example, the use of an SMS OTP alone to authenticate the user when paying online is no longer sufficient; a second authentication factor must be used, typically a password entered by the user.

The OTP may be the authentication code described in the RTS. For online payments, the RTS mandate the support of *dynamic linking*, i.e. the capability to generate an authentication code which digitally signs the transaction amount and the payee identifier. The OTP could be generated using this transaction information so that it meets the digital signing aspect of the dynamic linking requirement. However, it should be noted that due to the weaknesses of the SMS channel, the amount and payee information displayed in the SMS could have been tampered with.

The use of an SMS OTP combined with a password requires multiple manipulations from the user, leading to a poor user experience. This should draw the attention of banks to the RTS requirement on obstacles (article 32.3) when the user journey starts on a TPP interface.

[1] https://eba.europa.eu/eba-publishes-an-opinion-on-the-elements-of-strong-customer-authentication-under-psd2

## Pros of using SMS OTP combined with passwords

SMS OTPs are popular for a number of reasons:

- They are a simple way of adding the possession authentication factor to an existing username + password experience.
- They rely on a device that users generally already have in their possession: their mobile phone. A smartphone is not required; any mobile phone will work.
- They scale easily: no software nor hardware needs to be deployed to users.
- Account recovery, following the loss of the authentication device, i.e. the mobile phone, is easy to implement: a user will ask the mobile operator to block the SIM of the lost phone and ask for re-issuance of a SIM for the new mobile phone with the same telephone number. Account recovery requires no action from the bank.
- No major changes are required to the relying party infrastructure.
- They are easy to administer, i.e. relying parties can simply engage a partner for SMS delivery.

## Cons of using SMS OTP combined with passwords

Shortcomings of the SMS OTP + password authentication method are essentially linked to poor user experience and to its inherent vulnerabilities.

Passwords have well-documented shortcomings in terms of usability (hard to remember, re-used across services, etc.) and security (prone to phishing, data base hacking, etc.).

The addition of SMS OTPs does not resolve the password issues but in effect decreases the user convenience when the user needs to retrieve the OTP and enter it in the user interface. In e-commerce, where every additional user action is a barrier to conversion, SMS OTPs + password present a significant drawback.

From a vulnerability perspective, it is now established that sophisticated attacks on SMS are growing and that SMS OTP are not as secure as expected. Some of the vulnerabilities are explained below:

- Due to the inherent weakness of the international telecoms standard Signaling System No. 7 (SS7), which is vital for the operation of mobile telephone networks, it is possible and relatively inexpensive for hackers to buy equipment that can intercept and read SMS messages.
- The use of a third-party service such as those for SMS delivery increases the attack surface for hackers. In addition, SMS can be read by apps such as Google Hangouts, which have access to a phone users' inbox. This cannot deliver 2FA because it does not guarantee "something you have", i.e. a registered mobile phone.
- SMS OTP are susceptible to phishing attacks because users can be tricked into providing the OTP to a fake website; the OTP is then used by hackers to get unauthorized access to bank accounts.
- A fairly common form of attack against a phone is called SIM Swap. This is where a hacker hijacks a user's phone account by getting their carrier to move the telephone number to a new device. The hacker can then gain access to that user's account and receive SMS OTP in order to conduct fraudulent activity.

NIST have specifically stated that the use of SMS OTP as an authentication mechanism should be "restricted" because there are inherent security vulnerabilities which increase risk.

# 4. Hardware OTP Generators

Hardware OTP generators are specialized devices that feature a display, sometimes a keyboard and sometimes a small camera to scan a QR code.

They are seeded with a cryptographic key that is used to generate an unpredictable One Time Password that is displayed and entered manually by the user in the bank interface. The OTP verification by the bank server proves that the device is genuine and therefore serves as a possession factor.

Some OTP generators simply require the user to press a button to generate an OTP, but with PSD2, banks favor devices that require the user to enter a PIN code on the keyboard of the device, which is then verified within the device prior to generating and displaying the OTP.

Other OTP generators can capture transaction details when they are used to scan a QR code displayed by the merchant's user interface.

### Compliance with PSD2

Hardware OTP generators comply with the requirements for a possession factor in the authentication process. Those that feature a keyboard and ask the user to enter a PIN comply with the multifactor authentication requirement. For those that do not, the user would have to enter a password (or another element of knowledge or inherence) in a separate step to conform to the knowledge factor requirement.

The OTP generated by the device may be the authentication code mandated in the RTS. It complies with the uniqueness and non-forgeable requirements. To meet the dynamic linking requirement, the user would have to enter the amount as well as a merchant ID on the device, which is rather cumbersome.

Devices that are connected or that allow the user to scan a QR code simplify the implementation of dynamic linking as the transaction details may be automatically captured by the device.

### Pros of using Hardware OTP generators

These devices generally feature a Security Element, i.e. a highly secured hardware component that is used to hold the secret cryptographic key and generate the OTP. Security is a key benefit of using hardware OTP generators.

Moreover, these devices are issued by the bank to their users. The bank is fully in control of the authentication solution it provides; no part is delegated to a third party.

### Shortcomings

- Poor user experience: As shown above, hardware OTP generators require much manipulation to the detriment of a fluid customer journey. In many cases, the user will have to enter both a password and the OTP.

  The fact that the user must take the device and manually enter the OTP and password in the bank interface may be a significant issue for account aggregation services where this manipulation could happen multiple times for as many banks using this type of authentication.
- OTP devices must also be readily available where and when the user needs them. A user leaving their device at home will not be able to access their account at work.
- OTPs are still phishable. OTPs, in spite of their uniqueness, are still shared secrets entered manually in a field of a web page. A user could be tricked, through a clever phishing campaign, into entering an OTP into a fake banking web page, the fraudster capturing this OTP and using it to gain access to the user's account.

- Deployment/scalability: Hardware OTP generators cannot be purchased off the shelf but have to be seeded with a secret key of the bank and then shipped to users. The process is costly and not very flexible.
- Account recovery: The loss of the hardware OTP generator will stop the user from being able to authenticate preventing them from accessing the bank's services. As they are not off-the-shelf devices, the bank has to ship a new one to the user, which is not a fast and simple process.

# 5. CAP Readers

The Chip Authentication Program (CAP) was initiated by Mastercard and adopted by Visa for user authentication based on a dynamic passcode generated by the user's EMV payment smart card.

This EMV payment card is inserted in a dedicated portable CAP reader that also features a display and a numeric keypad. The dynamic code generated by the card, typically 8 digits long, is displayed on the screen and typed by the user in the bank's interface. The readers are often standalone but can sometimes be connected to a computer or mobile phone.

The CAP specifications support several authentication methods: Mode1 (fixed transaction), Mode 2 (transaction data signing) and Mode 3 (challenge/response). The user experience is as follows:

1. A user, which already has an EMV payment card, receives an EMV CAP reader from their bank.
2. The user inserts the smart card in the reader to generate an EMV CAP code to access an online banking service or a banking transaction.
3. The user selects the mode or the application with the dedicated button on the smart card reader.
4. In case of Challenge response mode, or for a transaction validation the user needs to enter the random number displayed by the banking application or the data of the transaction. In case of a transaction, the user may have to manually enter multiple data.
5. The user needs to enter the valid PIN of their EMV card. The PIN entered on the keyboard is submitted to the card, validated within the chip, and the card generates a passcode, which is displayed by the CAP reader.
6. The user manually enters the EMV CAP code in the banking application.

Some EMV CAP readers support a USB, optical or Bluetooth connection to simplify the customer journey for the user who would not need to manually enter the required information, but these solutions may require the installation of drivers or third party software on the PC; all connected readers increase the complexity of the system.

## Compliance with PSD2

EMV payment cards inserted in CAP readers comply with the SCA requirements in the RTS; the user PIN code required to generate the dynamic passcode is the knowledge factor and the positive verification of the passcode proves the possession of the device.

The passcode may be the authentication code described in the RTS. In this case, the solution also meets the requirement for dynamic linking if the device is used in mode 2 whereby the transaction details manually entered by the user are digitally signed by the card.

### Pros of using CAP readers

EMV CAP readers are popular for several reasons:

- They rely on the EMV payment card that banking customers have already received from their bank. These customers know their PIN code and are familiar with the card usage. Besides, they trust the security offered by the card.
- They provide a PSD2 compliant way of implementing SCA with a level of security, offered by the banking card, approved and certified by banks and schemes.
- No major changes to the relying party infrastructure are required.
- As the secret is stored in the smart card, no administration of the CAP readers is required other than delivery to the end user.

### Shortcomings

CAP readers share some of the disadvantages of Hardware OTP generators:

- Poor user convenience with the need to manually enter the dynamic passcode in the banking interface.
- The solution is susceptible to phishing attacks as the user may be tricked into entering the dynamic passcode in a fake website depending on the integration on the relying party.
- The CAP reader and payment card must be readily available where and when the user needs them. A user leaving them at home will not be able to access their account at work.
- Account recovery: The loss of the CAP reader or the card will stop the user from being able to authenticate, preventing them from accessing the bank's services. The bank would have to ship a new reader or personalize and ship a new card to the user, which is not a fast and simple process.

They also have specific shortcomings:

- The payment application in the smart card must be configured to be able to generate OTPs. If it was not configured in this way when it was personalized, the card will not function in a CAP reader. The issuing bank must anticipate this use case and properly personalize the card before sending it to the user.
- Installation of connected readers increases the complexity of the system and doesn't work out of the box.

## 6. Proprietary Smartphone and Biometrics Based Solutions

A number of banks have implemented solutions using biometric verification for user authentication within their mobile banking app. These solutions often combine the biometric verification of the user with a cryptographic proof calculated by the solution using a key stored in the mobile phone. More and more the biometric verification method is the one supported by the Original Equipment Manufacturer (OEM) and offered as a feature of the device operating system, for example fingerprint verification, facial recognition, or iris scanning. The biometric data is generally stored and matched locally by the device itself.

The user experience is extremely convenient: upon opening their mobile banking app, users are prompted to scan a finger or their face or iris and access their bank account.

When the user is transacting on another device, such as a PC, the smartphone and biometrics solution may be still be used "out of band" by sending a push notification to the phone prompting the user to take action within the mobile banking app. It is important that binding information is shared between the session on the PC and the session on the smartphone.

### Compliance with PSD2

Biometric verification is one factor of authentication defined as inherence in the RTS. To comply with the regulation, a second factor is required, typically a possession factor. Solutions that generate a cryptographic proof using a key stored in the phone meet the possession factor requirement: the verification of the cryptographic proof by the bank's server proves possession of the (key in the) phone.

This proof may be the authentication code described in the RTS. In this case, if the proof is calculated using transaction amount and payee, it meets the dynamic linking requirement.

As the biometric matching is handled locally, and most often delegated to the OEM of the device, it is highly desirable for the bank server to be notified of the matching result. This result should be transmitted securely in a non-forgeable, non-reproducible way to the bank server.

### Pros of using smartphone and biometrics based solutions

The main advantage of smartphone and biometrics based solutions is their simplicity and user convenience. Biometric verification is natively supported in most smartphones today, and their integration in mobile banking apps presents no difficulty.

Multifactor authentication and biometric methods could help with the risk assessment to increase the security of transaction.

Also, the smartphone is a device that users are likely to carry and have available where and when needed. It will be the device on which users transact or the means of authentication when transacting on any other device, provided the mobile phone can be reached out of band.

### Shortcomings

Smartphone and biometrics based solutions are proprietary solutions in which the proof of possession is most often based on the use of symmetric key cryptography. In such systems, a secret key has to be provisioned in the device over the air using a secure proprietary protocol and server.

Smartphone biometrics only works for users with adequate smartphones. They are not a universal solution and the bank will necessarily have to deploy another solution to authenticate its unequipped user population. In the absence of standardized protocols, this may require the use of different server solutions.

Account recovery: The loss of the mobile phone will stop the user from being able to authenticate, preventing them from accessing the bank's services. The recovery procedure would require the user to load the mobile banking application in the new phone and re-enroll, following remote identification by the bank.

# 7. So why change to FIDO?

The main reasons to change to FIDO are threefold: full compliance with PSD2 with enhanced user experience, enhanced resistance to phishing attacks, and greater scalability/ease of deployment.

### Compliance with PSD2 with enhanced user experience

Unlike solutions based on OTP/passcodes, FIDO requires no manual entry of authentication codes. The user experience can be as simple as scanning a face or fingerprint. Behind the scenes, a full multifactor authentication nevertheless takes place. This single user gesture offered by FIDO multi factor authentication solutions is fully in line with the European Banking Authority (EBA)'s opinion[2] that states that "the authentication of the [user] with the ASPSP in an AISP/PISP journey, […] should not create unnecessary friction".

FIDO authentication provides for a secure proof of possession thanks to the use of a private key securely held in the device that is used to generate non replicable assertions. When this is combined with on-device user verification, be it a PIN code or biometric verification, FIDO can remove the need for passwords altogether.

The FIDO assertion may be the authentication code mandated by the RTS. For remote payments, FIDO supports the dynamic linking requirement as the authenticator can sign an incoming message that combines a challenge with the transaction details.

With regard to biometrics, FIDO Authentication supports multiple modalities. A bank FIDO server does not need to be adapted to the biometrics handled on the FIDO authenticator. Also, FIDO assertions are generated upon valid biometric verification and digitally sign the result of the verification. This capability also aligns with another EBA recommendation in the above mentioned opinion stating that "[ASPSP should] secure transmission of the ASPSP's app authentication status to the ASPSP (e.g. using a signed proof that the biometric validation has been performed successfully)".

### Security and resistance to phishing

FIDO can offer a level of security similar to hardware OTP generators. FIDO authenticators can be found in Secure Element implementations as well as in Trusted Execution Environments.

The security level of a given FIDO authenticator is attested by its FIDO certification. The FIDO Alliance indeed runs a stringent certification program that not only tests interoperability of solutions but also their level of security.

FIDO protocols have strong measures to prevent phishing attacks: verification of web origin by the FIDO client, cryptographic assertion generated by authenticator that signs web origin and which is verified by the relying party during the authentication step. These measures effectively prevent Man-In-The-Middle attacks and phishing attempts.

### Deployment/scalability

Deployment and reach are facilitated both by the fact that FIDO is a standard natively supported in platforms and thanks to the way registration is handled.

FIDO authenticators can be bought off the shelf. They do not need to be preconfigured to a particular bank and no key seeding is required. Key generation happens at the time the user registers their FIDO device with their bank. A FIDO authenticator used with one bank can be registered for use with another bank; each time a new private/public key pair is generated specific to each bank. As only the public key is uploaded to the bank server, there is no need for secure provisioning servers as with other mobile phone based solutions.

[2] EBA Opinion of the European Banking Authority on obstacles under Article 32(3) of the RTS on SCA and CSC, dated 4 June 2020

Native platform support: With the native support of FIDO authentication in Android and iOS, many smartphones come with FIDO authenticators already embedded in the device, which greatly simplifies deployment and scale. FIDO is also now natively supported in Windows 10 so that external FIDO authenticators, including smartphones, can connect to a PC through the embedded CTAP (Client to Authenticator protocol) interface. No driver installation is required.

For non-Windows 10 users, FIDO enabled smartphones may be reached out of band for the purpose of user authentication.

Global reach: The fact that FIDO is a standard facilitates the deployment of authenticators to 100% of the bank's users; while, on average, 70% of a bank's customers will have a smartphone that can be used for FIDO authentication, 30% will not and still have to be addressed in the context of PSD2. For those users, the FIDO vendor community proposes a range of devices that all interoperate with FIDO servers. The investment of a bank in a FIDO server will allow the bank to deploy a variety of compatible FIDO devices.

Account recovery. As with other solutions relying on a device, the loss of a FIDO authenticator may block the user from authenticating and accessing the bank's services. However, where other solutions require the bank to ship a new device, FIDO authenticators may be off-the-shelf devices. For example, the user, having lost their smartphone, could rapidly purchase another one and register with the bank again to regain access. This does assume that the bank has a remote identity verification solution in place; a number of remote KYC solutions exist in the market to facilitate this process.

## 8. Summary Table

Color code: 🟥 Poor | 🟧 Medium | 🟩 Good

| Consideration | SMS OTP + password | Hardware OTP Generators | CAP Readers | Proprietary smartphone and biometrics solution | FIDO |
|---|---|---|---|---|---|
| User convenience | 🟥 | 🟥 | 🟥 | 🟩 | 🟩 When using MFA authenticators |
| PSD2 compliance | 🟩 | 🟩 May require password entry if no on-device user verification | 🟩 | 🟩 | 🟩 |
| Resistance to phishing | 🟥 | 🟥 | 🟥 | 🟩 | 🟩 |
| Security of the solution | 🟥 Passwords and SMS channel insecure. SIM swapping | 🟩 | 🟩 | 🟧 Depending on key storage method | 🟩 |
| Account recovery in case of loss | 🟩 | 🟥 New device to be seeded and deployed | 🟥 New reader to be deployed. New card to be personalized | 🟥 Re enrollment required | 🟥 Re enrollment required |
| Deployment/Scalability | 🟩 | 🟥 Devices need seeding and deploying | 🟧 Readers still need deploying | 🟧 Proprietary solutions. Require key provisioning server | 🟩 |

# 9. Glossary of Terms

| | |
|---|---|
| 2FA/MFA | Two-factor/multifactor authentication. Two or more of authentication factors defined as knowledge, inherence or possession |
| ASPSP | Account Servicing Payment Service Providers |
| CAP | Chip Authentication Program |
| CTAP | Client to Authenticator Protocol |
| EBA | European Banking Authority |
| EMV | Europay/Mastercard/Visa. A standard for payment cards |
| IVR | Interactive Voice Response |
| KYC | Know Your Customer |
| OEM | Original Equipment Manufacturer |
| OTP | One Time Password |
| PSD2 | The Second Payment Services Directive |
| PSU | Payment Service User |
| QR code | Quick Response, two dimensional, code |
| RTS | Regulatory Technical Standard. A technical document with the force of a regulation, in this document the RTS is tied to PSD2 |
| SCA | Strong Customer Authentication |
| SS7 | Signaling System No. 7 |
| TEE | Trusted Execution Environment |
| TPM | Trusted Platform Module |
| TPP | Third Party Provider. A financial services provider, as defined in PSD2 |
| U2F | Universal Second Factor |
| UAF | Universal Authentication Framework |