

# **FINANCIAL INCLUSION GLOBAL INITIATIVE (FIGI)**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

11/2019

Security, Infrastructure and Trust Working Group

## **Implementation of Secure Authentication Technologies for Digital Financial Services**

Report of the Security Workstream





## **FOREWORD**

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

A new global program to advance research in digital finance and accelerate digital financial inclusion in developing countries, the Financial Inclusion Global Initiative (FIGI), was launched by the World Bank Group, the International Telecommunication Union (ITU) and the Committee on Payments and Market Infrastructures (CPMI), with support from the Bill & Melinda Gates Foundation.

The Security, Infrastructure and Trust Working Group is one of the three working groups which has been established under FIGI and is led by the ITU. The other two working groups are the Digital Identity and Electronic Payments Acceptance Working Groups and are led by the World Bank Group.

© ITU 2019

This work is licensed to the public through a Creative Commons Attribution-Non-Commercial-Share Alike 4.0 International license (CC BY-NC-SA 4.0).

For more information visit <https://creativecommons.org/licenses/by-nc-sa/4.0/>

# **Implementation of Secure Authentication Technologies for Digital Financial Services**

*Security Workstream*

## **About this Report**

This report was written by Andrew Hughes, Abbie Barbir. The authors would like to thank the following contributors and reviewers: Arnold Kibuuka, Vijay Mauree, Harm Arendshorst, Tiakala Lynda Yaden, Mr. Mayank, Vinod Kotwal, Jeremy Grant, Brett McDowell, Adam Power, Sylvan Tran, Ramesh Kesanupalli, Chunpei Feng, Hongwei (Kevin) Luo, David Pollington, Matthew Davie, Wycliffe Ngwabe, Salton Massally and Mathan Babu Kasilingam.

If you would like to provide any additional information, please contact Vijay Mauree at [tsbfigisit@itu.int](mailto:tsbfigisit@itu.int)

<b>1</b>	<b>Executive Summary .....</b>	<b>1</b>
<b>2</b>	<b>Acronyms .....</b>	<b>3</b>
<b>3</b>	<b>Background .....</b>	<b>5</b>
<b>4</b>	<b>Introduction .....</b>	<b>6</b>
4.1	<i>Implementations examples section .....</i>	7
<b>5</b>	<b>The requirement for strong authentication – standards and regulations.....</b>	<b>7</b>
5.1	<i>ITU-T Recommendation X.1254 .....</i>	7
5.2	<i>NIST Special Publication 800-63-3 .....</i>	8
5.3	<i>eIDAS Regulation .....</i>	9
5.4	<i>Payment Services Directive .....</i>	9
5.5	<i>The ID2020 Alliance .....</i>	9
5.6	<i>Standardization Objectives .....</i>	10
<b>6</b>	<b>Strong Authentication Technologies and Specifications .....</b>	<b>10</b>
6.1	<i>Characteristics of Advanced Authentication Systems .....</i>	10
6.2	<i>FIDO Alliance Specifications.....</i>	12
6.2.1	Universal Authentication Framework (UAF) .....	12
6.2.2	Universal Second Factor (U2F) .....	13
6.2.3	Client to Authenticator Protocol (CTAP) .....	13
6.2.4	Web Authentication (WebAuthn) .....	14
6.2.5	FIDO Registration Flow .....	14
6.2.6	FIDO Authentication Flow .....	15
6.3	<i>Mobile Connect Specifications .....</i>	16
6.3.1	Mobile Connect for eIDAS.....	17
6.3.2	Mobile Connect for PSD2 .....	20
6.4	<i>IFAA Specifications.....</i>	21
6.4.1	IFAA Biometric Authentication – Local Model .....	22
6.4.2	IFAA Biometric Authentication - Remote Model .....	25
6.5	<i>Aadhaar Authentication .....</i>	25
6.5.1	APB Process Steps .....	27
6.5.2	Types and modes of authentication for Aadhaar .....	28
6.5.3	Aadhaar authentication security concerns .....	28
6.5.4	Security measures introduced recently to address those threats .....	29
6.6	<i>Cognitive Continuous Authentication .....</i>	30
6.7	<i>Decentralized Identity and Distributed Ledgers.....</i>	31
6.7.1	Decentralized Identity Definition of Terms.....	31
6.7.2	Decentralized Identity System Infrastructure Layers.....	32
6.7.3	Verifiable Credential and Decentralized Identifier Draft Standards .....	33
6.7.4	Verifiable Credentials.....	33
6.7.5	Decentralized Identifiers.....	34

6.7.6	DID Authentication .....	36
6.7.7	DID Resolution .....	36
6.7.8	Decentralized Identity Wallets.....	36
<b>7</b>	<b>Implementation examples of Strong Authentication Systems .....</b>	<b>37</b>
7.1	<i>Use case: Enrolment and Account opening.....</i>	38
7.1.1	Example: Aadhaar eKYC .....	38
7.1.2	Example: Sierra Leone National Digital Identity and Credit Platform – Kiva .....	39
7.1.3	Example: K-FIDO Enrolment example .....	41
7.1.4	Example: Zug eID – Ethereum Blockchain-based Digital ID .....	44
7.1.5	Example: FIDO Enrolment example .....	44
7.1.6	Example: Healthcare provider user enrolment.....	46
7.2	<i>Use case: Authentication to access a digital financial service .....</i>	47
7.2.1	Example: IFAA use case – Alipay fingerprint/face payment .....	47
7.2.2	Example: Aadhaar authentication .....	49
7.2.3	Example: K-FIDO authentication.....	50
7.2.4	Example: Healthcare provider customer authentication.....	51
7.2.5	Example: SK Telecom – Mobile Connect.....	52
<b>8</b>	<b>Conclusion .....</b>	<b>54</b>

## List of Figures

<b>Figure 1 – The Digital Financial Services Ecosystem .....</b>	<b>6</b>
<b>Figure 2 – Recommendation ITU-T X.1254 .....</b>	<b>8</b>
<b>Figure 3 – Universal Authentication Framework Architecture.....</b>	<b>13</b>
<b>Figure 4 – FIDO Registration of new keys.....</b>	<b>14</b>
<b>Figure 5 – FIDO Authentication .....</b>	<b>15</b>
<b>Figure 6 – Mobile Connect Portfolio of Services .....</b>	<b>16</b>
<b>Figure 7 – eIDAS level of assurance mapping with Mobile Connect .....</b>	<b>18</b>
<b>Figure 8 – Mobile Connect and eIDAS reference architecture.....</b>	<b>19</b>
<b>Figure 9 – Mobile Connect and eIDAS technical flow .....</b>	<b>19</b>
<b>Figure 10 – Mobile Connect PSD2 Use Cases .....</b>	<b>20</b>
<b>Figure 11 – High Level Reference Architecture for PSD2.....</b>	<b>20</b>
<b>Figure 12 – Mobile Connect Strong Customer Authentication - Server Initiated .....</b>	<b>21</b>
<b>Figure 13 – Mobile Connect Strong Customer Authentication - Device Initiated .....</b>	<b>21</b>
<b>Figure 14 – IFAA biometric authentication – local model .....</b>	<b>22</b>
<b>Figure 15 – IFAA biometric authentication – local model – Registration .....</b>	<b>23</b>
<b>Figure 16 – IFAA biometric authentication – local model – Authentication .....</b>	<b>24</b>
<b>Figure 17 – IFAA biometric authentication – local model – Deregistration.....</b>	<b>24</b>
<b>Figure 18 – IFAA biometric authentication – remote model .....</b>	<b>25</b>
<b>Figure 19 - Aadhaar Enabled Payment System Transactions.....</b>	<b>Error! Bookmark not defined.</b>
<b>Figure 20 - Aadhaar Payments Bridge Process .....</b>	<b>Error! Bookmark not defined.</b>
<b>Figure 21 – Accepto’s Cognitive Continuous Authentication™ .....</b>	<b>30</b>
<b>Figure 22 – Accepto-FIDO high level architecture.....</b>	<b>Error! Bookmark not defined.</b>
<b>Figure 23 – Sovrin Infrastructure Layers.....</b>	<b>32</b>
<b>Figure 24 – Roles and Relationships of Verifiable Credentials .....</b>	<b>34</b>
<b>Figure 25 – Universal DID Resolver .....</b>	<b>36</b>
<b>Figure 26 – Decentralized Identity Wallet with Verifiable Claims.....</b>	<b>37</b>
<b>Figure 27 – NCRA Identity Infrastructure .....</b>	<b>40</b>
<b>Figure 28 – Digital Credit Reporting Ecosystem Architecture .....</b>	<b>41</b>
<b>Figure 29 – Ecosystem Architecture .....</b>	<b>41</b>
<b>Figure 30 – National ID and i-PIN in Korea.....</b>	<b>42</b>
<b>Figure 31 – Registration process of K-FIDO service.....</b>	<b>43</b>
<b>Figure 32 – Registration process of FIDO .....</b>	<b>45</b>
<b>Figure 33 – Healthcare provider user enrolment.....</b>	<b>46</b>
<b>Figure 34 – IFAA use case: Alipay fingerprint/face payment.....</b>	<b>48</b>



<i>Figure 35 – IFAA use case: Alipay fingerprint/face payment – Technical framework .....</i>	<i>48</i>
<i>Figure 36 – Technical process of Authentication &amp; e-KYC services.....</i>	<i>49</i>
<i>Figure 37 – Authentication Process of K-FIDO Service .....</i>	<i>51</i>
<i>Figure 38 – User Journey to Authenticate to a Gaming Account Using T-Auth .....</i>	<i>52</i>
<i>Figure 39 – Authentication process of FIDO .....</i>	<i>53</i>

## List of Tables

<i>Table 1 – NIST SP 800-63-3 Authenticator Assurance Levels.....</i>	<i>8</i>
<i>Table 2 – Advanced Authentication System Characteristics .....</i>	<i>11</i>
<i>Table 3 – Digital Financial Services Use Case Examples.....</i>	<i>37</i>



## **1 Executive Summary**

This Report is the result of contributions and deliberations of the Financial Inclusion Global Initiative Security, Infrastructure and Trust Working Group Authentication work stream.

The Digital Financial Services (DFS) ecosystem requires standardized, interoperable, strong authentication technologies as enablers to reduce risk and protect assets. Weak authentication approaches based on web browsers and passwords are no longer sufficient to support safe DFS use. This report is focused on implementation. It describes technologies and standards that can be used to implement strong authentication systems for DFS and provides examples of implemented strong authentication systems.

Previously, the ITU Focus Group on Digital Financial Services, a multiparty consultative body for fostering the development of safe DFS ecosystems, produced recommendations on security, identification and authentication for DFS. This report addresses several of the Focus Group recommendations.

A primary goal of authentication systems is to increase confidence that a previously-enrolled user is actually that user. Access control and authorization policy can then be applied to that authenticated user.

Design decisions and technology choices for each authentication system element affect how ‘strong’ an authentication system is: how resistant to attack and compromise due to common threats. ‘Strong’ authentication systems are designed to mitigate threats that ‘weak’ authentication systems do not.

Typical authentication systems in use today were designed for the pre-mobile-device internet. They are based on a single authentication event, typically performed at application start up, and assume that the user, device and session do not change after that single authentication event. These elements have proven to introduce weaknesses into authentication systems.

In addition to ‘strong’ authentication system elements, advanced authentication systems are designed to address today’s threat models and design patterns. Compared to ‘strong’ authentication systems, there is an increased emphasis on detection and authentication of human users versus the client software used by people through environmental and behavioral analysis. New approaches are being implemented to minimize friction for mobile and multi-factor use cases: many systems are now built with ‘mobile first’ designs. Authentication now happens at many points during a user-system interaction: at identification time, at times when increased privileges are invoked (so-called ‘step-up’ authentication), and even continuously during the entire session.

This report describes several widely-adopted technical and policy standards that support strong authentication mechanisms.

The examples of strong authentication and advanced authentication systems are categorized as either enrolment or authentication for the use of DFS. These two use case categories primarily impact users of DFS.

The examples presented for the Enrolment use case describe how previously-established identity information can be used to create new service accounts and to satisfy KYC requirements. The key aspect in the examples is that the person has been enrolled previously with an authority: their identity information collected, verified and stored. This stored identity information is then available for later presentation to service providers, controlled by the person’s authorization to release that identity information.

The examples for the Entity authentication use case describe how next generation authentication mechanisms are used to authenticate an individual for authorization to consume services.

The report describes several examples of strong and advanced authentication systems for access to financial services. Further standardization work is needed to ensure that technologies are made to be fit for purpose and that different approaches can be evaluated for relative strengths and capabilities.

In conclusion, it is clear that there exist effective solutions addressing today's enhanced threats to DFS. Through careful planning, strong direction and sustained effort, access to DFS can be safe, low-barrier and effective.

## 2 Acronyms

AAGUID	Authenticator Attestation GUID
ASPSP	Account Servicing Payment Service Providers
AUA	Authentication User Agency
API	Application Programming Interface
APB	Aadhaar Payments Bridge
AEPS	Aadhaar Enabled Payment System
CIDR	Central Identities Data Repository
CTAP	Client to Authenticator Protocol
DFS	Digital Financial Services
eKYC	Electronic Know-Your-Customer
eIDAS	electronic IDentification, Authentication and trust Services
FAR	False Acceptance Rate
FIPS	Federal Information Processing Standards
FIDO UAF	Fast Identity Online User Authentication Framework
FIDO U2F	Fast Identity Online User Second Factor
FRR	False Rejection Rate
FTE	Failure to Enroll
GUID	Globally Unique Identifier
HSM	Hardware Security Module
ID GW	Identity Gateway
IdP	Identity Provider
IFAA	Internet Finance Authentication Alliance
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
IFAA	Internet Finance Authentication Alliance
ITU FG DFS	ITU Focus Group on Digital Financial Services
KUA	KYC User Agency
MGNREGA	Mahatma Gandhi National Rural Employment Guarantee Act
MFA	Multi Factor Authentication
MSISDN	Mobile Station International Subscriber Directory Number
NPCI	National Payments Corporation of India
NFC	Near Field Communication
NIST	National Institute of Standards and Technology
OIDC	OpenID Connect

OOB	Out of Band
OTP	One Time Password
PSD2	Payment Services Directive 2
RP	Relying Party
RTS	Regulatory Technical Standards
SCA	Strong Customer Authentication
SSB	Standards Setting Bodies
SIM	Subscriber Identity Module
TPP	Third Party (Payment Service) Providers
U2F	Universal Second Factor
UAF	Universal Authentication Framework
UIDAI	Unique Identification Authority of India
UPI	Universal Payments Interface
VPA	Virtual Payment Address

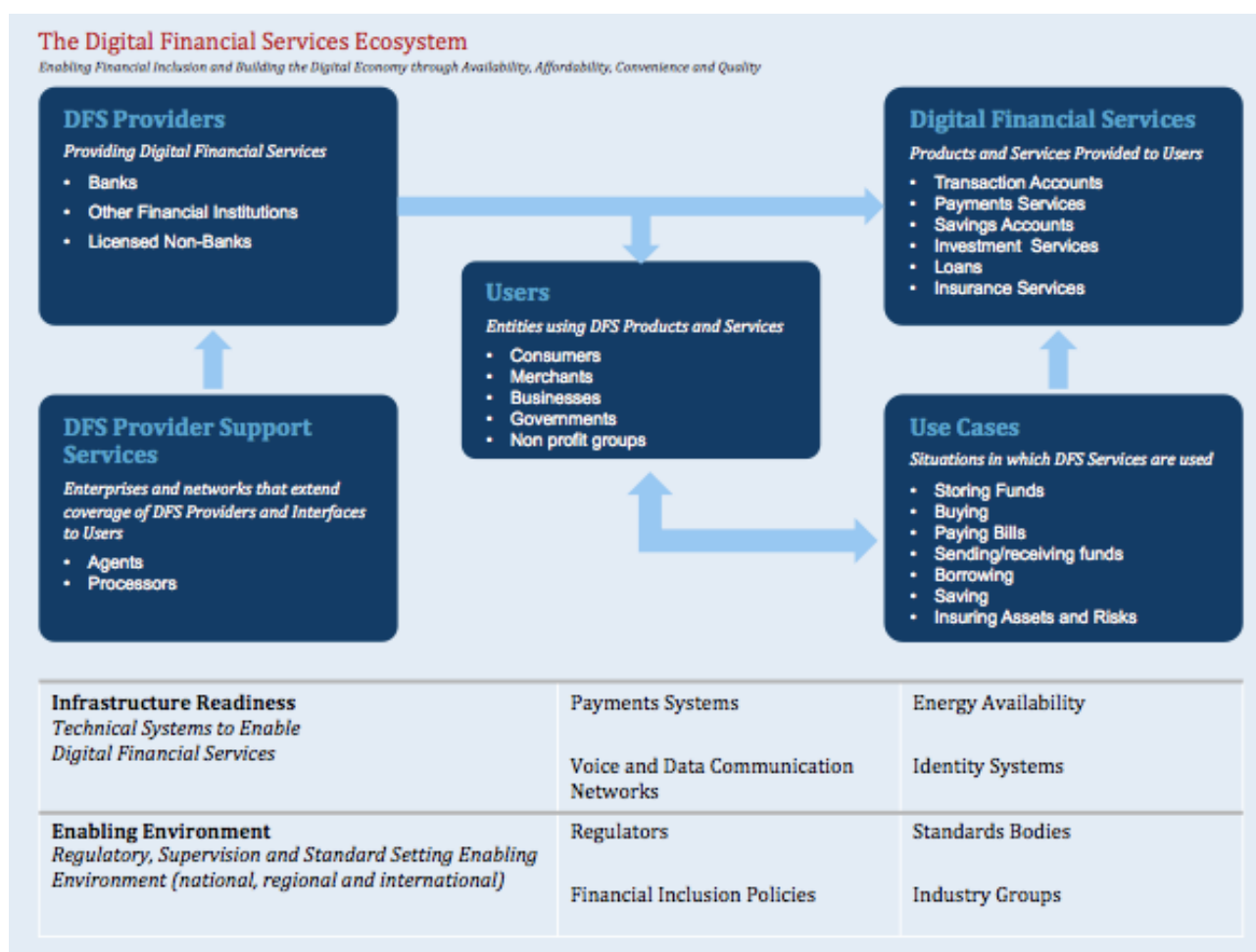
### 3 Background

The Financial Inclusion Global Initiative was formed as a follow-on activity of the ITU Focus Group on Digital Financial Services (hereinafter, “ITU FG DFS” or “Focus Group”) which was established as a multiparty consultative body for fostering the development of safe, enabling DFS ecosystems. The overall objectives of the Focus Group were to: (i) increase and formalize the collaboration between financial and telecommunications authorities with respect to DFS; (ii) identify key issues limiting the development of safe, enabling DFS ecosystems; (iii) analyse how these issues have been addressed in practice and exchange information on best practices; and (iv) develop policy recommendations for authorities and other stakeholders on how to approach these issues in their countries. The Focus Group brought together financial and telecommunications authorities, private-sector stakeholders, consumer advocates, DFS technical experts, development partners, and other key DFS stakeholders to collaboratively explore these issues and develop consensus recommendations. [1]

This report addresses several of the Focus Group recommendations, including [1]:

- The use of mobile devices that allow for the use of strong authentication mechanisms to demonstrate ownership of the device is recommended.
- At time of registration, a DFS operator should create a digital identity for its customers, for use in both DFS transactions and (where relevant) in identity assertion with external service providers.
- DFS Operators should ensure an intuitive and straightforward customer experience for registration and subsequent authentication.
- Policy makers and regulators are encouraged to use national identity systems, or other market-wide identity systems, to help with opening transaction accounts, addressing payments, and, in some instances, improving transaction security.
- App developers should ensure that DFS applications are designed and implemented in accordance with industry and Standards Setting Bodies (SSB) best practices for secure software development, including encrypted and authenticated communication and secure coding practices.
- Regulators should standardize digital identity registration, and ensure interoperability between DFS operators and service providers relying on the digital identity.

The Digital Financial Services ecosystem consists of users (consumers, businesses, government agencies and non-profit groups) who have needs for digital and interoperable financial products and services; the providers (banks, other licensed financial institutions, and non-banks) who supply those products and services through digital means; the financial, technical, and other infrastructures that make them possible; and the governmental policies, laws and regulations which enable them to be delivered in an accessible, affordable, and safe manner. [2]



**Figure 1 – The Digital Financial Services Ecosystem**

This report describes aspects of the Identity Systems infrastructure that enable digital financial services: account opening (eKYC) and strong electronic credential authentication.

## 4 Introduction

The Digital Financial Services (DFS) ecosystem requires standardized, interoperable, strong authentication technologies as enablers to reduce risk and protect assets.

Regulators are increasing the requirement for robust identification of clients to combat money laundering and other misuses of financial systems.

Along with the increase of mobile-only and remote-only clients, financial institutions are facing new kinds of fraud, impersonation and security threats that older password-based authentication systems were never designed to address.

The systems, technologies and approaches described in this report have been designed for use in mobile computing environments, blending well-established techniques such as public key cryptography with new techniques such as generation and storage of cryptographic keys on-device instead of centrally. The move towards mobile devices has made the already weak password-based security less usable while the increasing availability of widespread fingerprint and other biometric sensors makes the shift to password-less and multi-factor authentication technologies feasible.

Technologies and approaches that use continuous and adaptive authentication to minimize the time required to detect impostors are emerging. Technologies that securely shift the storage location for



personal data out of centralized storage that might be limited by network infrastructure, to user-controlled mobile devices are advancing. These new approaches will become widely available within the next several years, and will help to address new threats that emerge over time.

#### **4.1 Implementations examples section**

Section 7 of this report contains descriptions of implemented systems covering two DFS use cases: Enrolment/Account Opening and Authentication for accessing a DFS. Both use cases deal with identification of an individual: the former handles the situation where the DFS system sees the individual for the first time; the latter authenticates the individual using previously-issued credentials. To effectively manage mis-identification risks, DFS providers must ensure that both enrolment and credential authentication are robust and use standardized methods and technologies.

### **5 The requirement for strong authentication – standards and regulations**

A primary goal of authentication systems is to increase confidence that a previously-enrolled user is actually that user. Access control and authorization policy can then be applied to that authenticated user.

Entity authentication assurance is needed in order to comply with various stages of an identity management system. In particular, identity vetting is required as part of the credentialing process. The assurance of achieved in the vetting process determines the nature of the issued credential and eventually can be used to perform access control decisions by the relying party.

Initial work from NIST, ITU and ISO focused on defining four levels of entity assurance. The levels included identity vetting and credentialing. Experience in implementations revealed some limitations of combining authentication assurance and identity vetting assurance which resulted in limiting cases where all what is needed to ensure that the same entity is requesting access as opposed to who is the real requester. As such newer versions of NIST 800-63 separated the identity vetting assurance levels from the credentialing levels and promoted the use of three levels as opposed to the initial four levels. ITU X.1254 and ISO 29115 are being updated to reflect NIST work.

A recent report from the Financial Action Task Force (FATF) [3] provides a comprehensive overview of solutions that can be used to fulfil identity vetting requirements.

This section describes standards that cover strong authentication and authentication technologies that support strong authentication mechanisms.

#### **5.1 ITU-T Recommendation X.1254**

Recommendation ITU-T X.1254, *Entity authentication assurance framework* [4] describes an authentication assurance model which can be used by service providers and authentication providers to communicate about expectations and available authentication mechanisms. The authentication assurance model currently includes four levels of increasing assurance. There are many inputs used to determine the level of assurance achieved by an authentication method. ITU-T X.1254 is currently under revision and will align its assurance model with the 3-level model of NIST Special Publication 800-63-3. It is important to note that ISO 29115 [5] is equivalent to ITU-T X.1254. ISO 29115 is being revised at this time to include latest updates from NIST 800-63.

In the entity authentication phase, the entity uses its credential to attest its identity to a Relying Party (RP). The authentication process is concerned solely with the establishment of confidence in the claim or assertion of identity, and it has no bearing on or relationship with the actions the relying party may choose to take based upon the claim or assertion.

Technical			Management and organizational
Enrolment phase	<ul style="list-style-type: none"> <li>• Application and initiation</li> <li>• Identity proofing and identity information verification</li> </ul>	<ul style="list-style-type: none"> <li>• Record-keeping/recording</li> <li>• Registration</li> </ul>	<ul style="list-style-type: none"> <li>• Service establishment</li> <li>• Legal and contractual compliance</li> <li>• Financial provisions</li> <li>• Information security management and audit</li> <li>• External service components</li> <li>• Operational infrastructure</li> <li>• Measuring operational capabilities</li> </ul>
Credential management phase	<ul style="list-style-type: none"> <li>• Credential creation</li> <li>• Credential pre-processing</li> <li>• Credential issuance</li> <li>• Credential activation</li> <li>• Credential storage</li> </ul>	<ul style="list-style-type: none"> <li>• Credential suspension, revocation, and/or destruction</li> <li>• Credential renewal and/or replacement</li> <li>• Record-keeping</li> </ul>	
Entity authentication phase	<ul style="list-style-type: none"> <li>• Authentication</li> <li>• Record-keeping</li> </ul>		

**Figure 2 – Recommendation ITU-T X.1254**

ITU-T X.1254 section 10.3 describes threats to and controls for the authentication phase.

## 5.2 NIST Special Publication 800-63-3

NIST Special Publication 800-63B *Digital Identity Guidelines Part B* [6] addresses how an individual can authenticate using an authentication system. Similar to ITU-T X.1254, the NIST document uses levels of assurance to indicate relative effectiveness of authenticators and authentication protocols.

**Table 1 – NIST SP 800-63-3 Authenticator Assurance Levels**

Authenticator Assurance Level	Description
AAL1	AAL1 provides some assurance that the claimant controls an authenticator bound to the subscriber's account. AAL1 requires either single-factor or multi-factor authentication using a wide range of available authentication technologies. Successful authentication requires that the claimant prove possession and control of the authenticator through a secure authentication protocol.
AAL2	AAL2 provides high confidence that the claimant controls authenticator(s) bound to the subscriber's account. Proof of possession and control of two distinct authentication factors is required through secure authentication protocol(s). Approved cryptographic techniques are required at AAL2 and above.
AAL3	AAL3 provides very high confidence that the claimant controls authenticator(s) bound to the subscriber's account. Authentication at AAL3 is based on proof of possession of a key through a cryptographic protocol. AAL3 authentication SHALL use a hardware-based authenticator and an authenticator that provides verifier impersonation resistance; the same device MAY fulfill both these requirements. In order to authenticate at AAL3, claimants SHALL prove possession and control of two distinct authentication factors through secure authentication protocol(s). Approved cryptographic techniques are required.

The publication lists the authenticator types and authentication protocols capabilities that are acceptable at each level of assurance.

### 5.3 eIDAS Regulation

The Regulation (EU) N°910/2014<sup>1</sup> on electronic identification and trust services for electronic transactions (eIDAS Regulation) provides a regulatory environment to European Union members to enable secure electronic interactions between businesses, citizens and public authorities. An important aspect of the eIDAS Regulation is that it describes electronic identification assurance levels. Assurance levels in eIDAS fulfil the same function as those in Recommendation X.1254 and NIST SP 800-63-3.

### 5.4 Payment Services Directive

The Payment Services Directive (PSD2) is in force in Europe, and Strong Customer Authentication (SCA) will be required to access bank accounts for information aggregation or payment initiation. The “Regulatory Technical Standards on strong customer authentication and common and secure communication” (RTS), published by the European Banking Authority, describe the principles and requirements of multi-factor authentication and authentication code generation.

The RTS include the following requirements:

- Users must be authenticated using a minimum of two-factor authentication
- The authentication of a user should result in the generation of an authentication code, a cryptographic signature of the transaction. The authentication code must, in the case of remote payments, be linked to the amount and payee approved by the user
- The user’s cryptographic material must be protected from unauthorized disclosure

### 5.5 The ID2020 Alliance

The ID2020 Alliance [7] is a public-private partnership committed to improving lives through digital identity. The Alliance brings together multinational institutions, non-profits, philanthropy, business, and governments to set technical standards for a safe, secure, and interoperable digital identity that is owned and controlled by the user. It funds high-impact pilot projects that bring digital identity to vulnerable populations, and uses the data generated to find scalable solutions and inform public policy.

The overall objective of the ID2020 Alliance is to empower individuals, enable economic opportunity and advance global development by increasing access to digital identity.

By 2030, the Alliance aims to have facilitated the scaling of a safe, verifiable, persistent digital identity system, consistent with UN Sustainable Development Goal 16.9: “By 2030, provide legal identity for all, including birth registration”. From 2017 to 2020, the Alliance’s work will focus on two areas: developing and testing the best technological solutions for digital identity; and, working with governments and existing, established agencies to implement these solutions.

The ID2020 Certification Mark [8] is an initiative by the ID2020 Alliance to create a Trustmark for digital identities that meet our technical requirements. The Certification Mark is based on the ID2020 Technical Requirements document which is regularly updated by a team of experts to reflect the changing landscape of digital identity. The Certification Mark application form consists of 50 questions across seven focus areas: applicability, identification and verification, authentication, privacy and control, attestations and trust, interoperability, and recovery and redress.

---

<sup>1</sup> [http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv:OJ.L\\_.2014.257.01.0073.01.ENG](http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv:OJ.L_.2014.257.01.0073.01.ENG)

## **5.6 Standardization Objectives**

International standards for strong authentication mechanisms continue to be improved. Areas that need additional focus include:

- Behavioural modeling
- Relative strengths of authenticators
- Requirements for security capabilities of mobile devices relative to authenticator strength
- User experience requirements for strong authentication

ITU-T Study Group 17 is the lead study group on identity management and currently Q10/17 is updating Recommendation ITU-T X.1254 “Entity Authentication Assurance” to reflect recent changes to NIST Special Publication 800-63-3 “Digital Identity Guidelines”.

Additionally, FIDO UAF 1.1 and FIDO CTAP protocols have been standardized in Study Group (SG) 17 as Recommendation ITU-T X.1277 and Recommendation ITU-T X.1278.

The work presented in this report was written with the consideration of being submitted to Q10/17 of ITU-T SG 17 for further standardization as part of the X.1254, X.1277 and X.1278 work.

## **6 Strong Authentication Technologies and Specifications**

Authentication systems involve individuals, credentials issued to those individuals, and authenticators used by the individual to prove they are the original registered credential receiver. Authentication protocols define how each element interacts to authenticate the individual. Each element has observable or measurable behaviors in the environment which can be compared to previously-measured ‘normal’ behavior.

Design decisions and technology choices for each authentication system element affect how ‘strong’ an authentication system is: how resistant to attack and compromise due to common threats. ‘Strong’ authentication systems are designed to mitigate threats that ‘weak’ authentication systems do not.

For example, a weakness for individuals is having to deal with password systems. Passwords are hard to remember, easy to steal, reused across services and very inconvenient to use. Stronger authentication systems might choose to use a biometric to unlock a local secure encryption key vault, which gives the individual a lower-friction, password-less experience.

Authenticators such as SMS-delivered one-time codes that are subject to phishing could be replaced by hardware cryptographic authenticators such as a Secure Element or Trusted Execution Environment in mobile devices.

Authentication protocols that use shared secrets or unencrypted transmissions could be replaced with asymmetric key cryptographic protocols, encrypted channels and different keys for each service. Multi-factor authentication protocols have additional attacker resistance than single-factor protocols.

The threat landscape changes regularly and design decisions must be made to address new or commonly-used threats. For example, threats to web site access from desktop computers are different from mobile-only apps and services which have been invented in recent years.

### **6.1 Characteristics of Advanced Authentication Systems**

Typical authentication systems in use today were designed for the pre-mobile-device internet. They are based on a single authentication event, typically performed at application start up, and assume that the user, device and session do not change after that single authentication event. Authentication tends to be a high friction activity with a poor user experience, especially when password or multi-

factor authentication are used. Current-generation authentication systems are not easy for mobile users to interact with: on mobile devices, authentication events are infrequent and rely on device security locks that may not be effective.

Advanced authentication systems are designed to address today's threat models and design patterns. Compared to 'strong' authentication systems, there is an increased emphasis on detection and authentication of human users versus the client software used by people through environmental and behavioral analysis. New approaches are being implemented to minimize friction for mobile and multi-factor use cases: many systems are now built with 'mobile first' designs. Authentication now happens at many points during a user-system interaction: at identification time, at times when increased privileges are invoked (so-called 'step-up' authentication), and even continuously during the entire session.

Advanced authentication systems do not replace strong authentication systems – the technologies work together to address different threats and vulnerabilities.

The objective of advanced authentication systems is to provide a low-friction experience for users, while reducing risk and increasing security assurance.

**Table 2 – Advanced Authentication System Characteristics**

<b>Characteristics of advanced authentication systems</b>	<b>Description</b>
Elimination or reduced reliance on passwords	Use of passwords to authenticate is hard for users, particularly on mobile devices. Password systems are increasingly vulnerable to database breaches and phishing.
Multi-modal user authentication	The authentication step is designed using more than one authentication mode to minimize user friction. Modes could include push to mobile app, web-based form, device biometric matching, passwords, or voice response.
Real-time analysis of user behavior to detect anomalies	Detection of anomalies that are inconsistent with the mode of access, such as having a user session jump between distant geographical locations, use of an unregistered device, or change in web browser mid-session.
Continuous authentication of user, software and device	Continuous authentication techniques challenge the user, software or device throughout the session, seeking valid responses. Some continuous authentication techniques are invisible to the user, especially at the device and software levels.
Dynamic risk scoring of authentication confidence	Authentication confidence takes several factors into account, such as: device capabilities, the requested transaction, use of weaker or stronger authenticators.
Consistency across all devices and channels a user chooses to use	Authentication systems are designed for user experience and security. Users are connecting to services using whichever channel is convenient for the user. Authentication systems must ensure that the authentication confidence is maintained no matter which channel is used.

See 6.6 Cognitive Continuous Authentication for a description of a solution that embodies these advanced authentication system characteristics.

## 6.2 FIDO Alliance Specifications

The FIDO Alliance protocols use standard public key cryptography techniques to provide stronger authentication. During registration with an online service, the user's client device creates a new key pair. It retains the private key and registers the public key with the online service. Authentication is done by the client device proving possession of the private key to the service by signing a challenge. The client's private keys can be used only after they are unlocked locally on the device by the user called "user verification". User verification can take the form of any number of user-friendly and secure action such as swiping a finger, performing facial recognition, entering a PIN, or speaking into a microphone. Private keys are bound to a device and prove that users are in possession of a specific device (i.e. – the "something you have" of authentication), and their combination with user verification ensures that every authentication is multi-factor authentication.

FIDO protocols are designed to protect user privacy. The protocols do not provide information that can be used by different online services to collaborate and track a user across the services. Biometric information, if used, never leaves the user's device and is only used for user verification to approve the use of a private key.

For implementing authentication beyond a password, companies have traditionally been faced with an entire stack of proprietary clients and protocols.

To enable interoperability between client authentication methods, FIDO standardizes the client and protocol layers. This allows many client authentication methods such as biometrics, PINs and second-factors to be used with a variety of online services in an interoperable manner.

The main FIDO specifications are Universal Second Factor (U2F) [5], Universal Authentication Framework (UAF) [6] and the FIDO2 project which includes both the Client to Authenticator Protocol (CTAP) [7] and W3C's Web Authentication (WebAuthn) [9].

The FIDO2 Project is a set of interlocking initiatives that together create a FIDO Authentication standard for the web and greatly expands the FIDO ecosystem.

FIDO2 is comprised of the W3C's Web Authentication specification (WebAuthn) and FIDO's corresponding Client-to-Authenticator Protocol (CTAP), which collectively will enable users to leverage common devices to easily authenticate to online services — in both mobile and desktop environments.

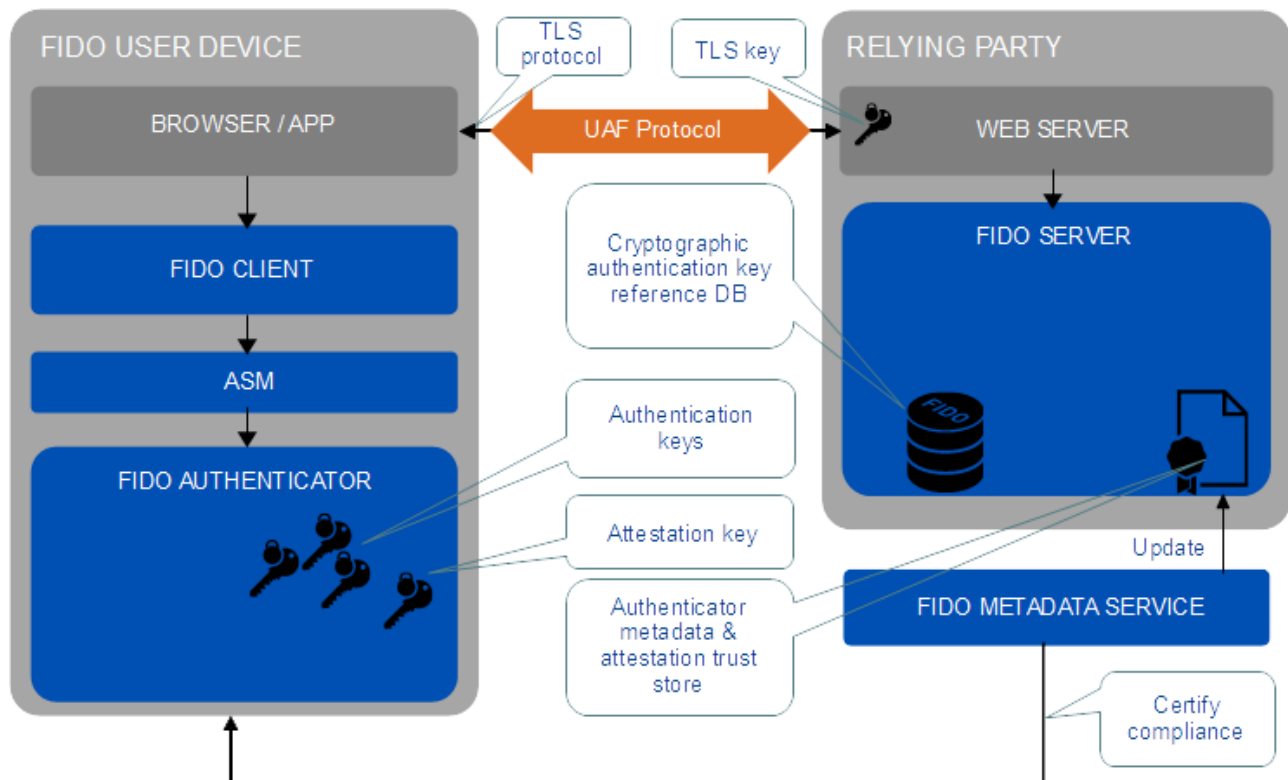
### 6.2.1 Universal Authentication Framework (UAF)

The goal of the Universal Authentication Framework is to provide a broad and comprehensive framework for cryptographically secure multifactor authentication. It includes first-factor (e.g. PIN and biometrics), second-factor, as well as a generalized architecture and protocol that can be extended to any platform or integrated with any system.

The UAF specification standardizes four pieces:

1. The authenticator, which is a device that creates and securely stores the authentication secrets
2. The server, which registers users and subsequently validates authentication requests
3. The client, which acts as a multiplexer and policy enforcer between multiple servers and multiple authenticators.

4. The protocol, which defines the message formats, cryptographic objects, etc. that are carried between the authenticator and the server through the client.



**Figure 3 – Universal Authentication Framework Architecture**

This architecture is re-used by the other FIDO specifications.

### 6.2.2 Universal Second Factor (U2F)

The FIDO U2F specification is focused on the narrow goal of providing second-factor authentication in browsers. It defines a JavaScript API for browsers to perform second factor authentication using JavaScript register() and sign() functions; as well as defining NFC, Bluetooth Low Energy (BLE), and USB communications protocols for registering and authenticating with security keys. These specifications allow better user experience and more secure second factor authentication.

Note: The FIDO U2F JavaScript API has been superseded by WebAuthn and the transport specifications for NFC, BLE, and USB have been merged into the latest FIDO CTAP specifications.

### 6.2.3 Client to Authenticator Protocol (CTAP)

The CTAP specification describes a set of protocols for communication between external authenticator devices and a client/platform, as well as bindings of this application protocol to a variety of transport protocols using different device communication protocols (USB, NFC, Bluetooth). Each transport binding defines the details of how a client (such as a browser or operating system) can make requests to an authenticator to register or authenticate against various services.

CTAP is intended to be used in scenarios where a user interacts with a relying party (a website or native app) on some platform (e.g., a PC) which prompts the user to interact with an external authenticator (e.g., a smartphone).

In order to provide evidence of user interaction, an external authenticator implementing this protocol is expected to have a mechanism to obtain a user gesture. Examples of user gestures include: as a consent button, password, a PIN, a biometric or a combination of these.

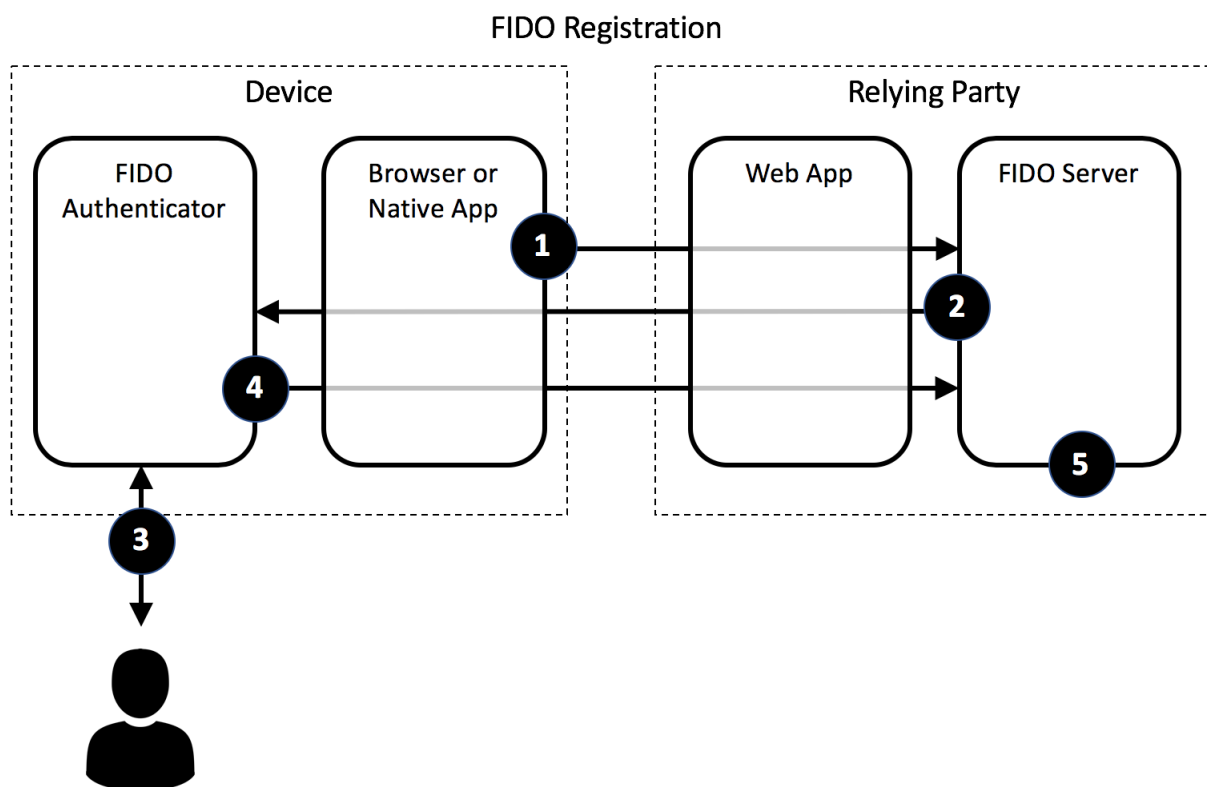
The CTAP specification was created as part of the FIDO2 project, in conjunction with the WebAuthn specification. It contains two distinct protocols: 1) the original U2F transport protocols that enable authenticator devices to perform second factor authentication, retroactively named “CTAP1”; 2) an extended and reformatted set of U2F transport protocols that enable multifactor authentication, named “CTAP2”.

#### 6.2.4 Web Authentication (WebAuthn)

As part of the FIDO2 project, the FIDO Alliance collaborated with the World Wide Web Consortium (W3C) to standardize the browser’s JavaScript APIs for cryptographically strong multifactor authentication – known as Web Authentication. The WebAuthn specification is a Proposed Recommendation of the W3C and includes both browser specific portions of authentication (APIs and browser processing rules) as well as generic message formats (assertions and attestations) that may be reused for non-browser implementations such as servers, operating systems, and authenticators communicating using the CTAP protocol. The WebAuthn specification also defines a series of extensible points, such as the ability to add new attestation formats and the ability to add new extensions to the protocol and define their processing rules.

#### 6.2.5 FIDO Registration Flow

The figure below shows the simplified message flows for registration and authentication. Of note: the public-private key pair is created by the FIDO authenticator, not by the Relying Party. This enables the individual to control how they wish to be known by the Relying Party and also does not disclose any part of the private key to external systems.

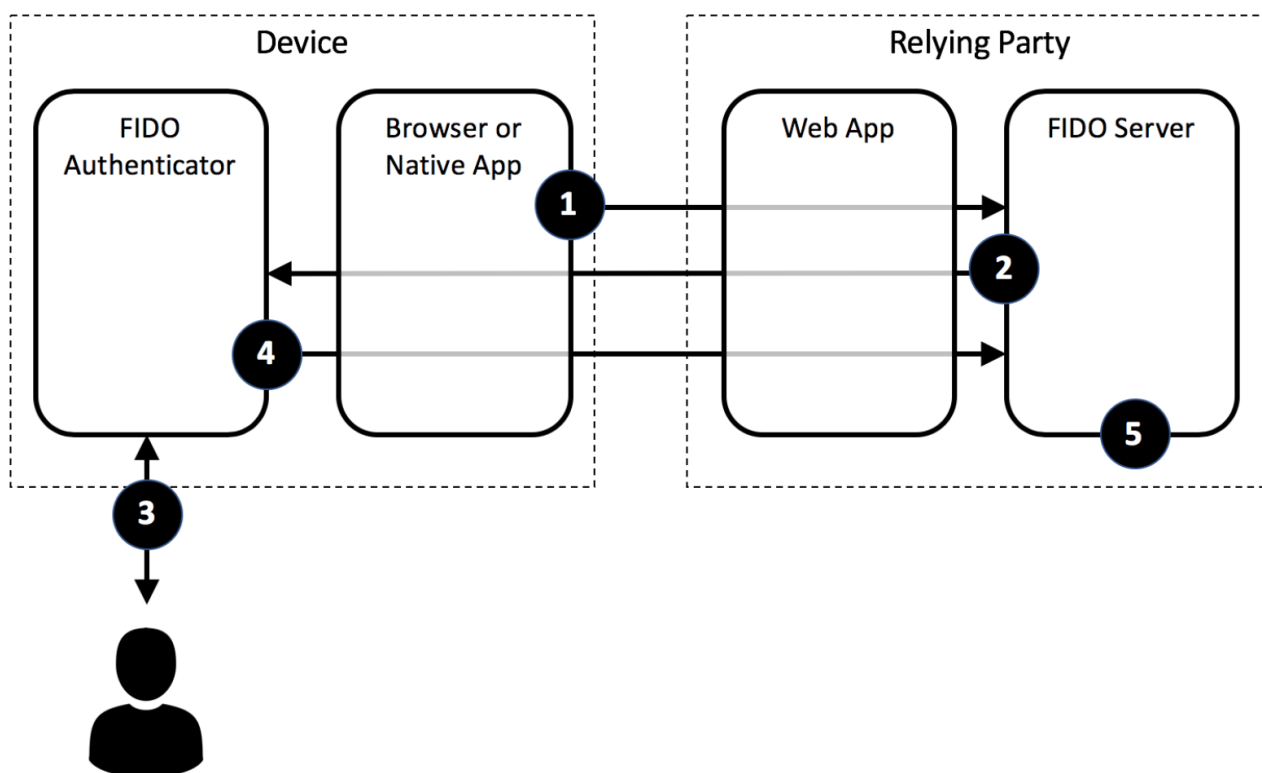


**Figure 4 – FIDO Registration of new keys**



1. Initiate registration with Relying Party
2. FIDO Server sends registration challenge and requested registration options
3. Authenticator performs user verification on device to signal the user's consent to registering with the service
4. Authenticator generates a new key pair for the service and associates the private key with the service's origin. The public key and device model number are signed over by a device model specific (shared across no less than 100,000 devices) attestation private key. The authenticator sends a registration response: device model number + device attestation signature + user's public key
5. Validate response and attestation. The device model number (AAGUID) can be used to look up metadata about the device, such as the attestation public key, the type of user verification being performed (e.g. – biometric, PIN), and the security characteristics of the device (e.g. – how private keys are protected; how biometric templates are protected; third-party security and biometric certifications).
6. The service stores user's public key for future authentication requests.

#### 6.2.6 FIDO Authentication Flow



**Figure 5 – FIDO Authentication**

1. Initiate authentication with Relying Party
2. FIDO Server sends authentication challenge and preferences for the authenticators or credentials to be used
3. Authenticator performs user verification on device to signal the user's consent to authenticate with the service

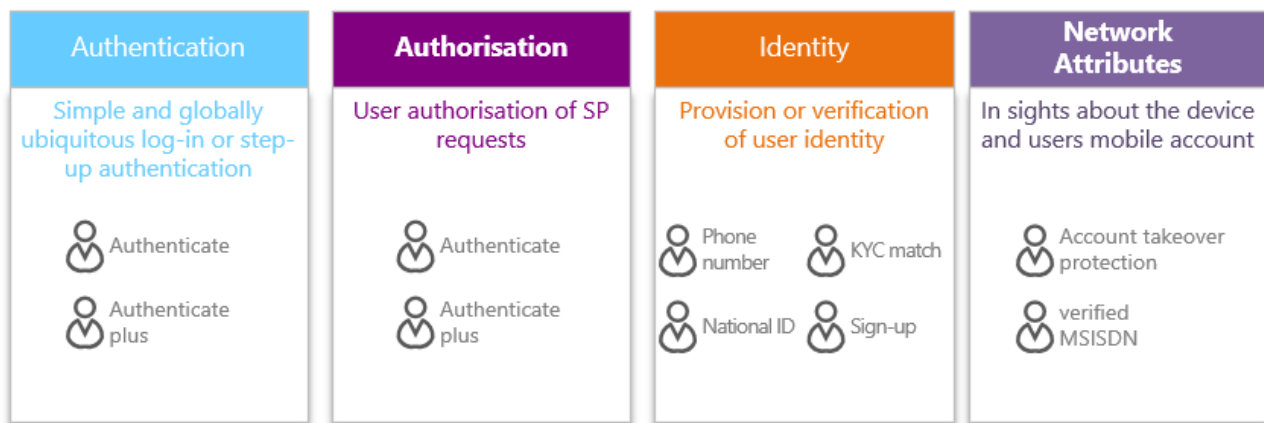
4. The authenticator uses the service's origin to look up the private key for authentication and uses the private key to sign the challenge from the server. The server sends an authentication response: challenge + signature.
5. The server retrieves the public key for the user and validates the signature on the challenge.

The FIDO Alliance has produced reports and white papers setting out implementation guidance for several scenarios including for financial services applications. Some relevant reports are listed in Annex B of this report.

### 6.3 Mobile Connect Specifications

Mobile Connect is the mobile operator-facilitated secure universal identity solution developed by the GSMA in collaboration with Mobile Operators. The GSMA represents the interests of mobile operators worldwide, unites nearly 800 of the world's mobile operators, as well as more than 230 companies in the broader mobile ecosystem. To-date there are more than 470 million active Mobile Connect users via over 70 operators covering more than 40 countries and reaching more than 3 billion people.

Mobile Connect is a portfolio of mobile-enabled services that can be integrated into a Service Provider's application to support access to services provided by the Service Provider. Mobile Connect provides strong customer authentication, authorisation, and permissioned access to a user's identity and contextual network attributes. Figure 6 outlines the range of services provided by Mobile Connect.



**Figure 6 – Mobile Connect Portfolio of Services**

Mobile Connect uses a distributed architecture in which each Mobile Operator deploys Mobile Connect services for its particular user base, but with all deployments abiding by a strict set of technical standards to ensure that from a Service Provider's perspective, the experience of consuming Mobile Connect services from any of the Mobile Operators is consistent.

Mobile Connect is based upon the OpenID Connect (OIDC) protocol which provides an identity layer on top of the OAuth 2.0 protocol. It allows Users to be identified by their MSISDN (or a related Pseudonymous Customer Reference) and to be authenticated securely via their mobile device with the SIM providing security. Mobile Connect defines two profiles of OIDC to support Device-Initiated and Server-Initiated requests for authentication, authorisation or permissioned access to User attributes.

The serving Mobile Operator supports and selects an appropriate authenticator to present the authentication and authorisation requests to the user on their mobile device to which the user

responds. The authenticator may also be used to seek user consent for the serving operator to share or validate user attributes with the Service Provider. The authenticator is selected based on operator policy, device capability and the Level of Assurance required.

Mobile Connect authentication factors and insights include:

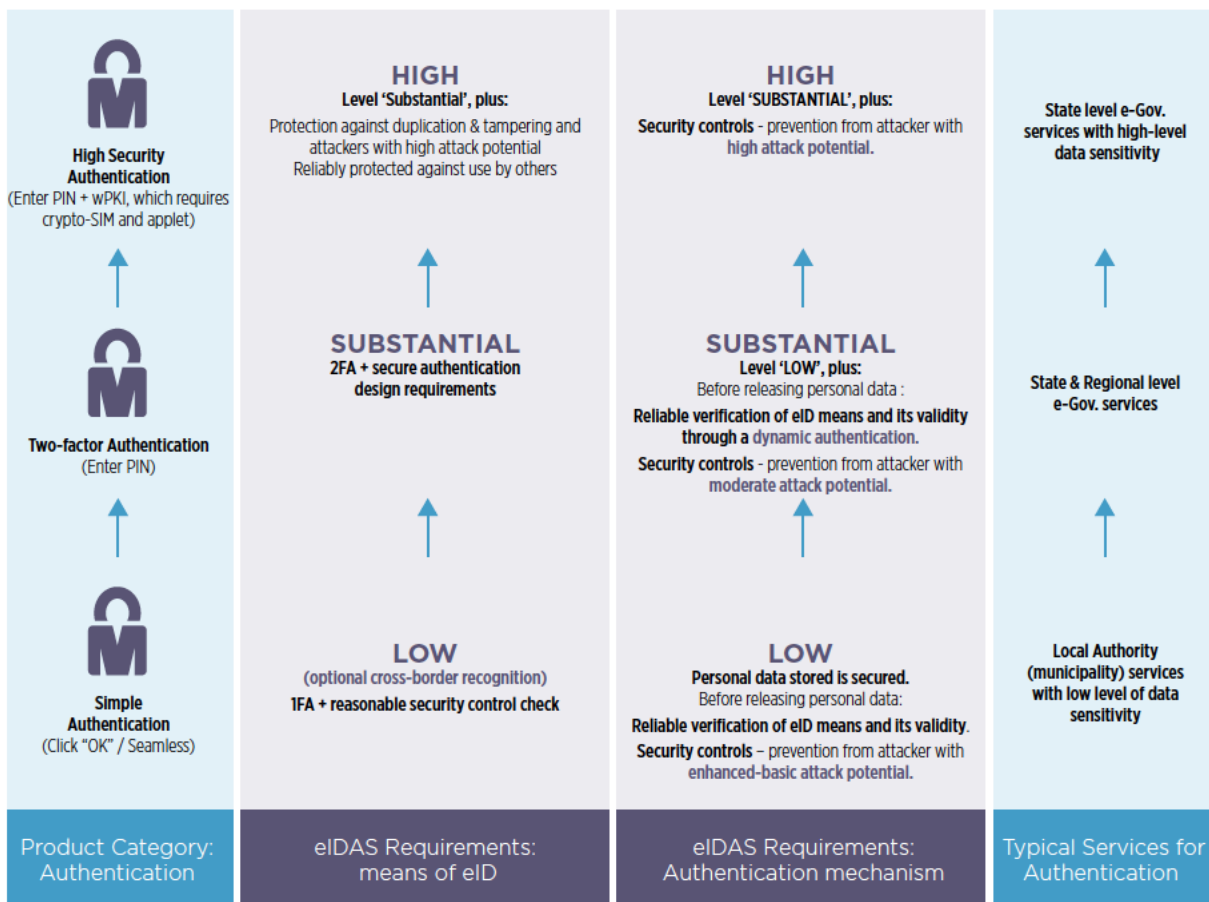
- Possession-based (Something I Have); the possession of the mobile device by the user. This is the first factor used in Mobile Connect Authentication.
- Knowledge/secrecy-based (Something I Know); for example, PIN/Personal Code.
- Active Inherence (Something I Am); for example, biometrics: fingerprints, iris scan, facial biometrics.
- Passive Inherence (Something the Network Knows); Mobile network-based inherence elements, such as usual cell sites (can also be used as “something the user does”) available to the mobile operator. This separation between device and network is vital to fighting fraud and establishing ownership of the device.
- Contextual (Something I Do); for example, supplement the device-based authentication with network-based insights to create a more robust multi-factor authentication mechanism (such as pairing status between IMSI, IMEI and MSISDN).

Mobile Connect levels of assurance are a guide to the degree of confidence in an authentication process. As a critical element within the Mobile Connect ecosystem, the Mobile Connect levels of assurance are used in the Mobile Connect API (OpenID Connect), in the cryptographically-signed Identity Token sent as an authentication proof to the Service Provider, in the authenticator-selection policy and also in the Mobile Connect product-enablement policy.

### **6.3.1 Mobile Connect for eIDAS**

Mobile Connect levels of assurance have been mapped against the minimum technical specification requirements of the Commission Implementing Regulation (EU) 2015/1502 on setting out minimum technical specifications and procedures for assurance levels for electronic identification means and authentication.

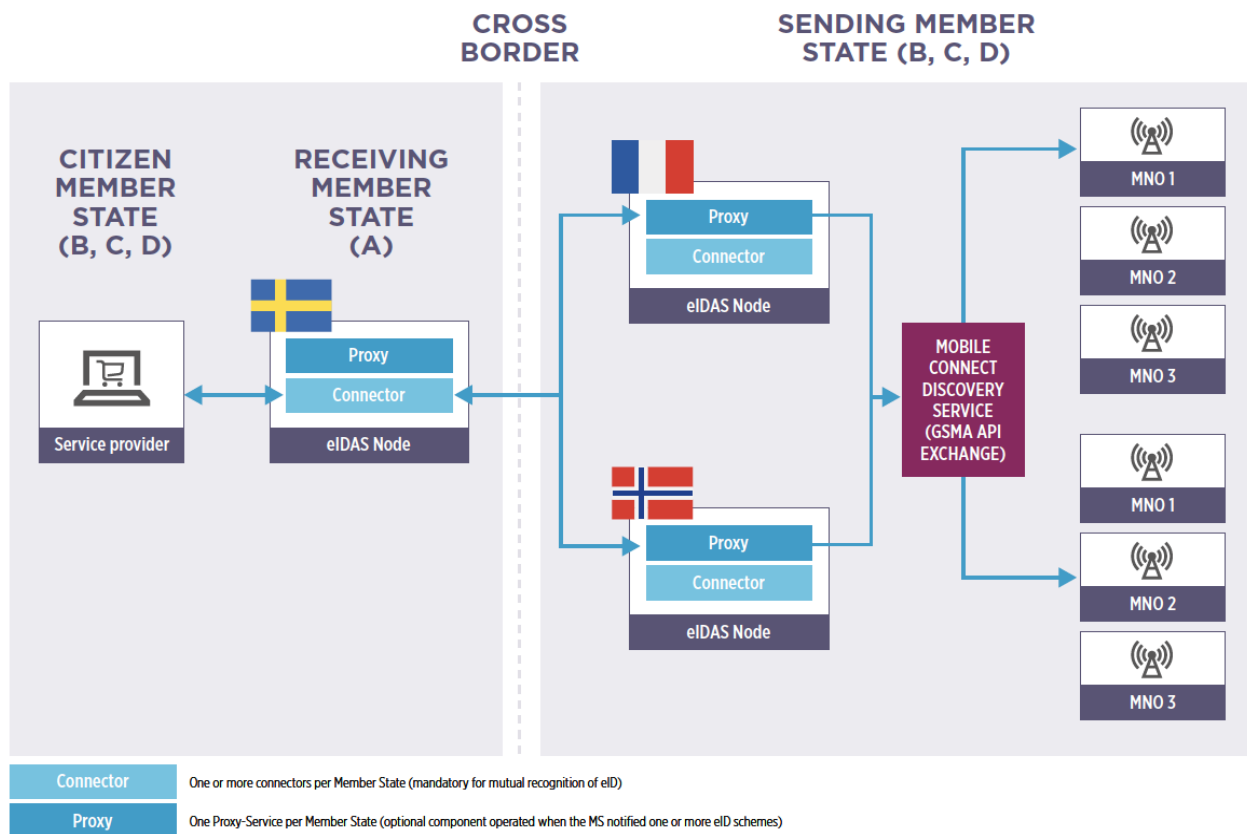
Figure 7 shows a mapping between Mobile Connect and eIDAS level of assurance.



**Figure 7 – eIDAS level of assurance mapping with Mobile Connect**

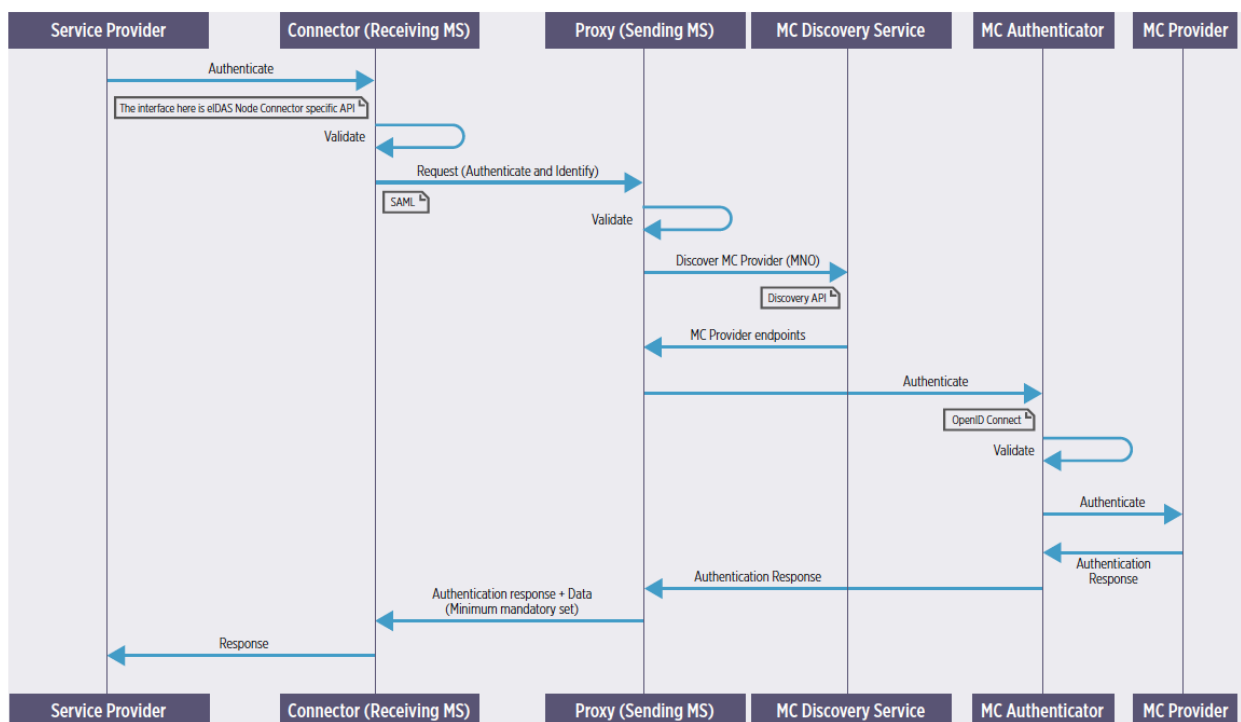
Mobile Connect also meets the eIDAS technical specification and interoperability requirements for integration with national ID as designed by EU Member States eIDAS Nodes in collaboration with the European Commission CEF project<sup>2</sup>. An example reference architecture of eIDAS for the integration with Mobile Connect is shown in figure 8.

<sup>2</sup> <https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eIDAS+eID+Profile>.



**Figure 8 – Mobile Connect and eIDAS reference architecture**

Figure 9 shows a flowchart of Mobile Connect used within an eIDAS deployment.













**Figure 9 – Mobile Connect and eIDAS technical flow**

### 6.3.2 Mobile Connect for PSD2

In relation to PSD2, the Mobile Connect framework uses out-of-band Authentication, such that the Authentication channel is separated from the service request channel and utilises the SIM-enabled Mobile Device along with support from the mobile network in addition to providing dynamic linking to be fully PSD2 compliant. Mobile Connect can support SCA in both decoupled and OAuth modes.

The following figures illustrate the use cases, architecture and flows related to PSD2.

Scenario	Mobile Connect as Second Factor Solution (upgrade from SMS OTP)	Mobile Connect as Two-Factor Solution
Login to account	 authenticate  verified MSISDN  authenticate plus	 authenticate plus
Payment authorisation	 authorise  authorise plus	 authorise plus
Scenario	Mobile Connect providing risk factors into fraud decision engine	
Transaction risk monitoring, incl. for TRA*	 account takeover protection  verified MSISDN  KYC match	
Other products on roadmap (Proximity, Roaming...)		

\*TRA: Transaction Risk Analysis exemption

Figure 10 – Mobile Connect PSD2 Use Cases

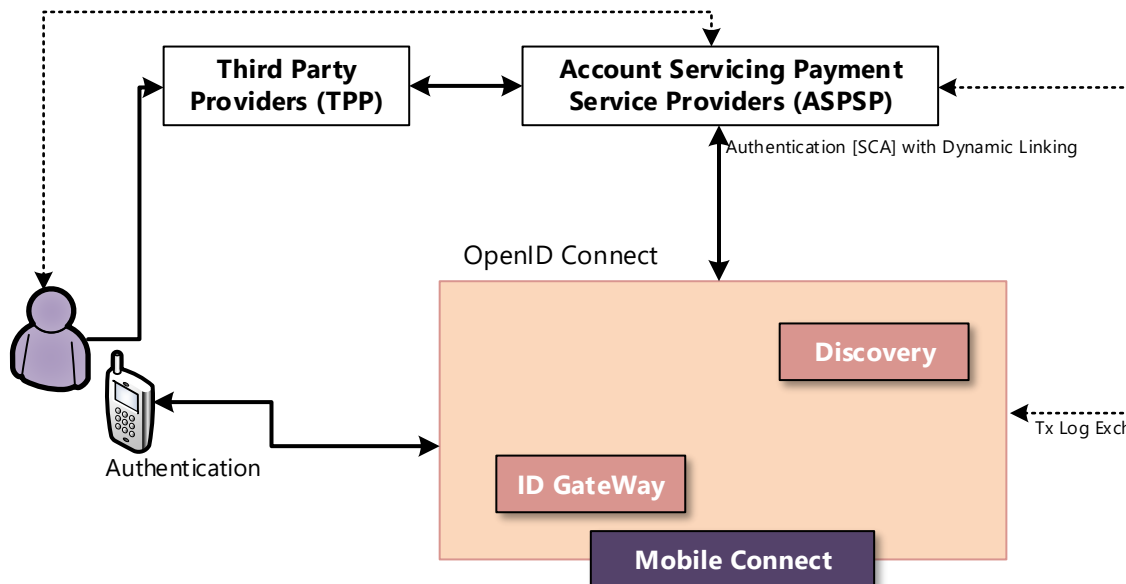
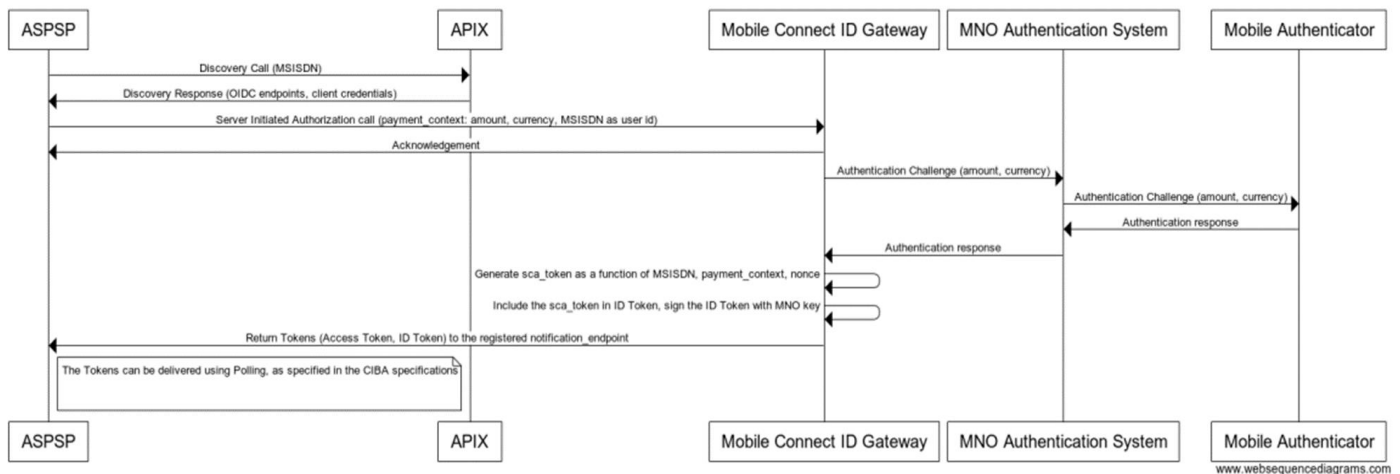
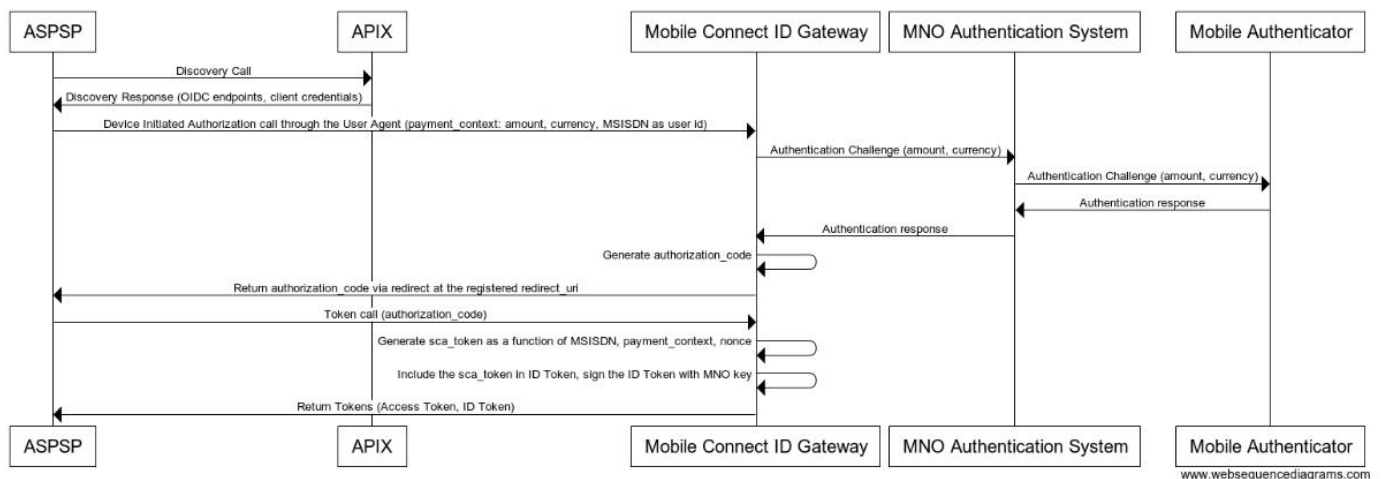


Figure 11 – High Level Reference Architecture for PSD2



**Figure 12 – Mobile Connect Strong Customer Authentication - Server Initiated**



**Figure 13 – Mobile Connect Strong Customer Authentication - Device Initiated**

Additional details to assist in deployment of Mobile Connect can be found in Annex C of this report.

## 6.4 IFAA Specifications

IFAA (Internet Finance Authentication Alliance) was established in June 2015, where around 200 international company and institute members collaborate to innovate authentication scenarios, develop biometrics-based standards, and deliver financial-grade interoperable authentication solutions.

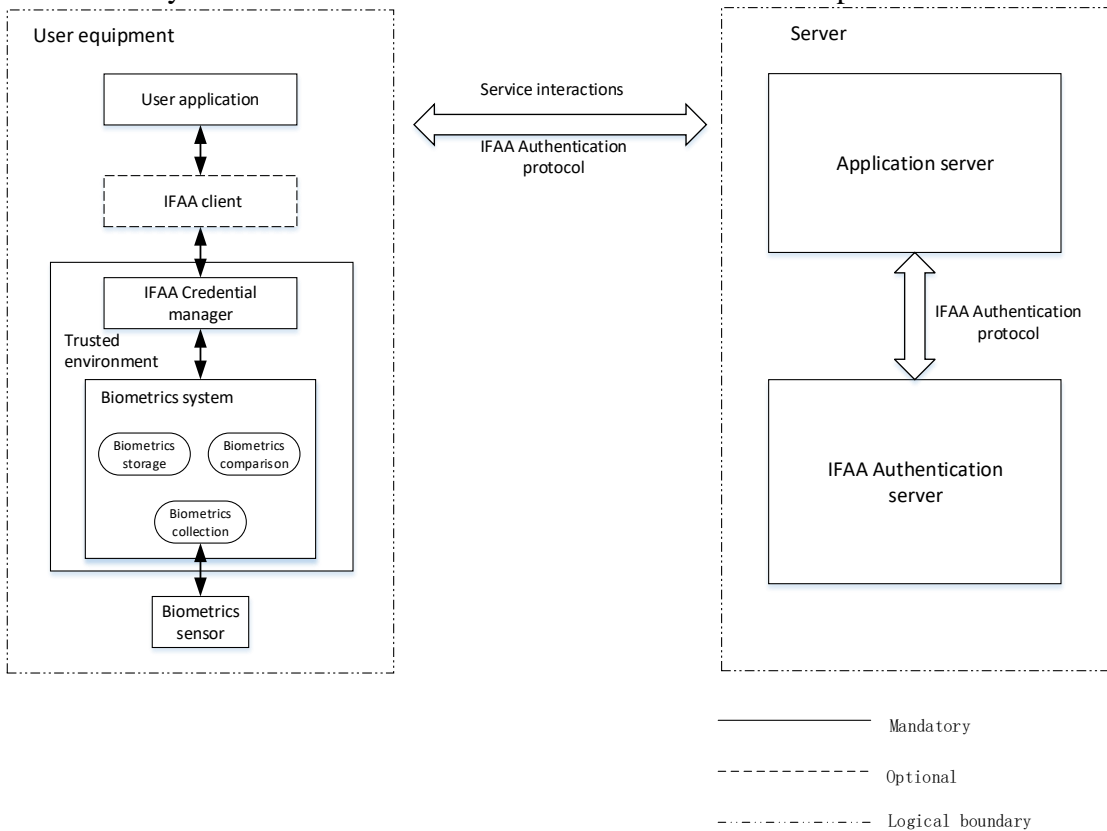
IFAA has been applying continuous focus to address authentication challenges by improving the efficiency while reducing the cost of device adaptation. The main IFAA specification is IFAA Local Passwordless Technical Specification (T/IFAA 0001-2016), which requires strict protection of user data in the trusted execution environment. To date, this specification has been supported by more than 1.2 billion mobile devices and 360 device models. In July 2018, an updated version IFAA Local Passwordless Technical Specification (T/IFAA 0002-2018) was published to describe the optional security-enhanced solution which uses a SE (Secure Element) to protect sensitive applications, keys and data.

IFAA specifications have powered massive adoption of fingerprint authentication in scenarios of e-commerce, Internet Finance, Banking, Traveling, Mobile Office as well as Internet of Things (IoT). Increased coverage of banks has been seen in the past months.

IFAA identifies two technical models for biometric authentication: local model and remote model. At present IFAA specifications focus on the local model, but the remote model is also on IFAA’s schedule.

**6.4.1 IFAA Biometric Authentication – Local Model**

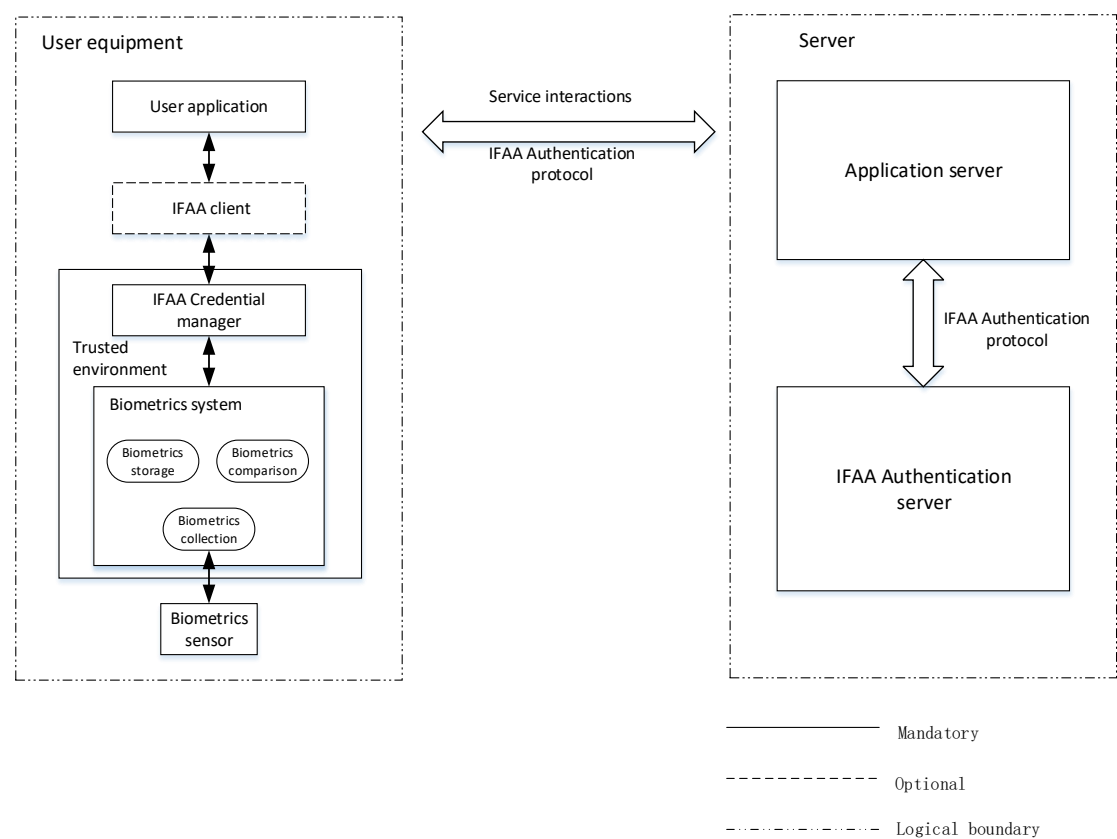
In the local model, the biometrics system resides in the user equipment. The biometric data are collected, stored and compared locally by the biometrics system when called by the user application but the authentication decision is not made locally. A credential will be provided based on the output of the local biometrics system and sent by the user application to the server side. The authentication decision will be made by the authentication server based on the credential provided from the user



equipment.



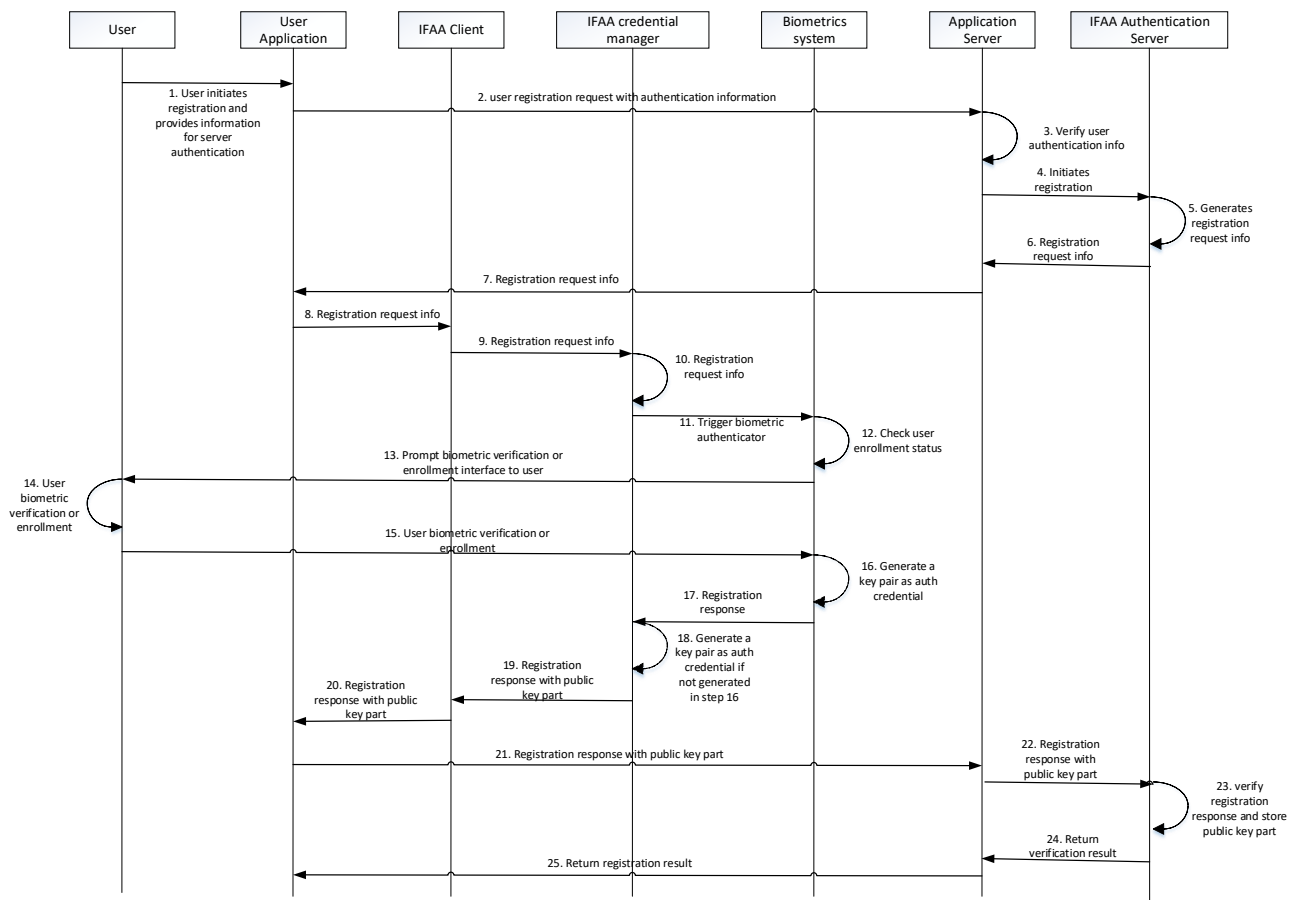
Figure 14 is the technical framework of this model.



**Figure 14 – IFAA biometric authentication – local model**

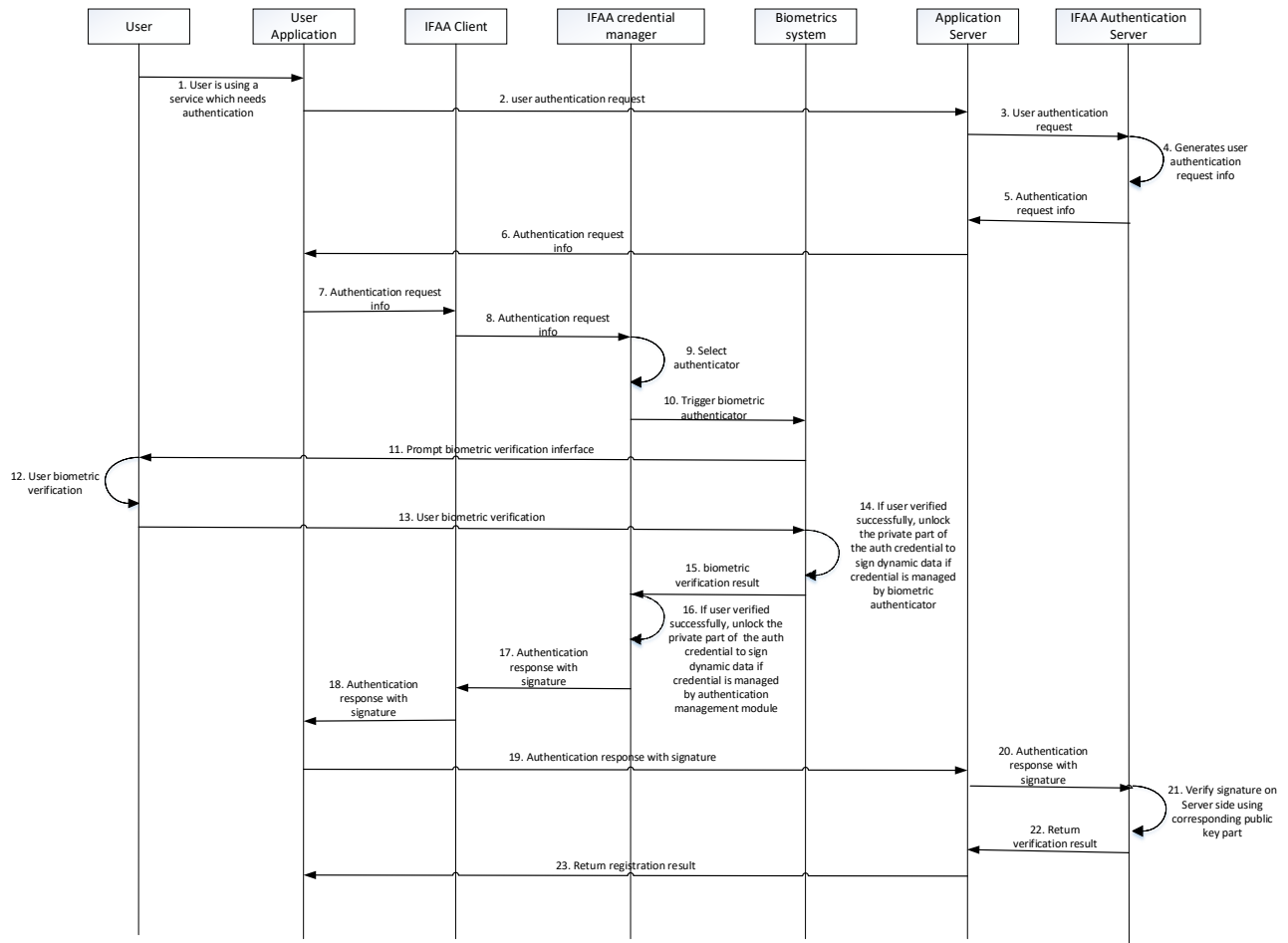
IFAA specifications define three main protocols for the local model: registration, authentication, and deregistration.

Figure 15 is the message flow of the registration protocol:



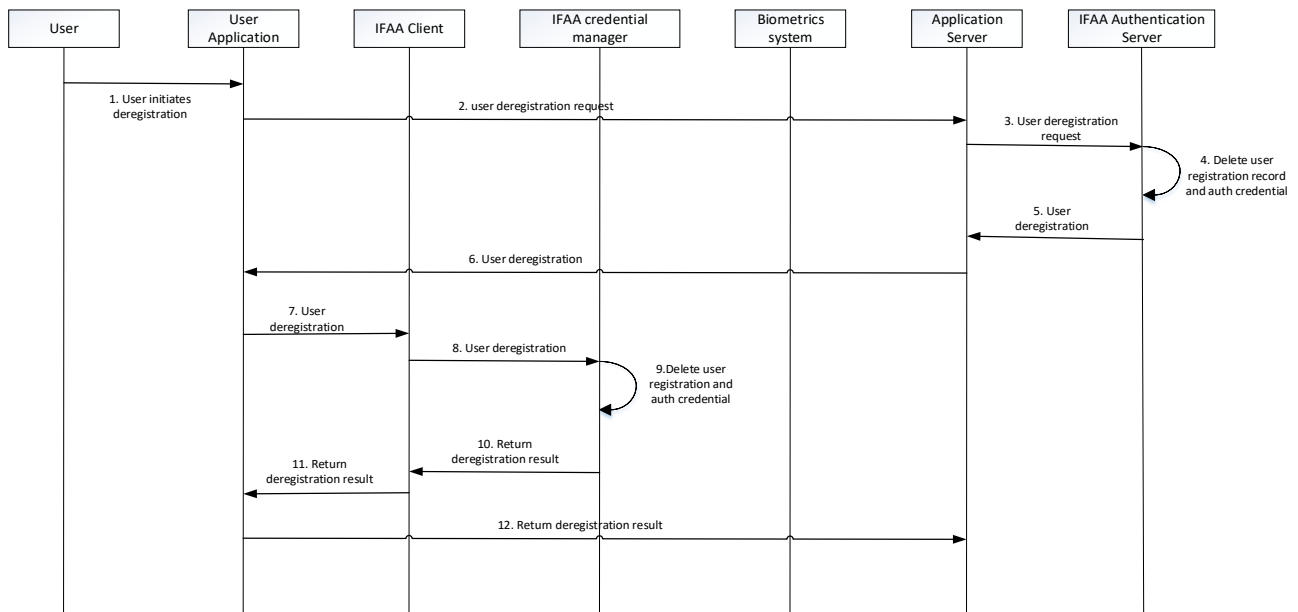
**Figure 15 – IFAA biometric authentication – local model – Registration**

Figure 16 is the message flow of the authentication protocol:



**Figure 16 – IFAA biometric authentication – local model – Authentication**

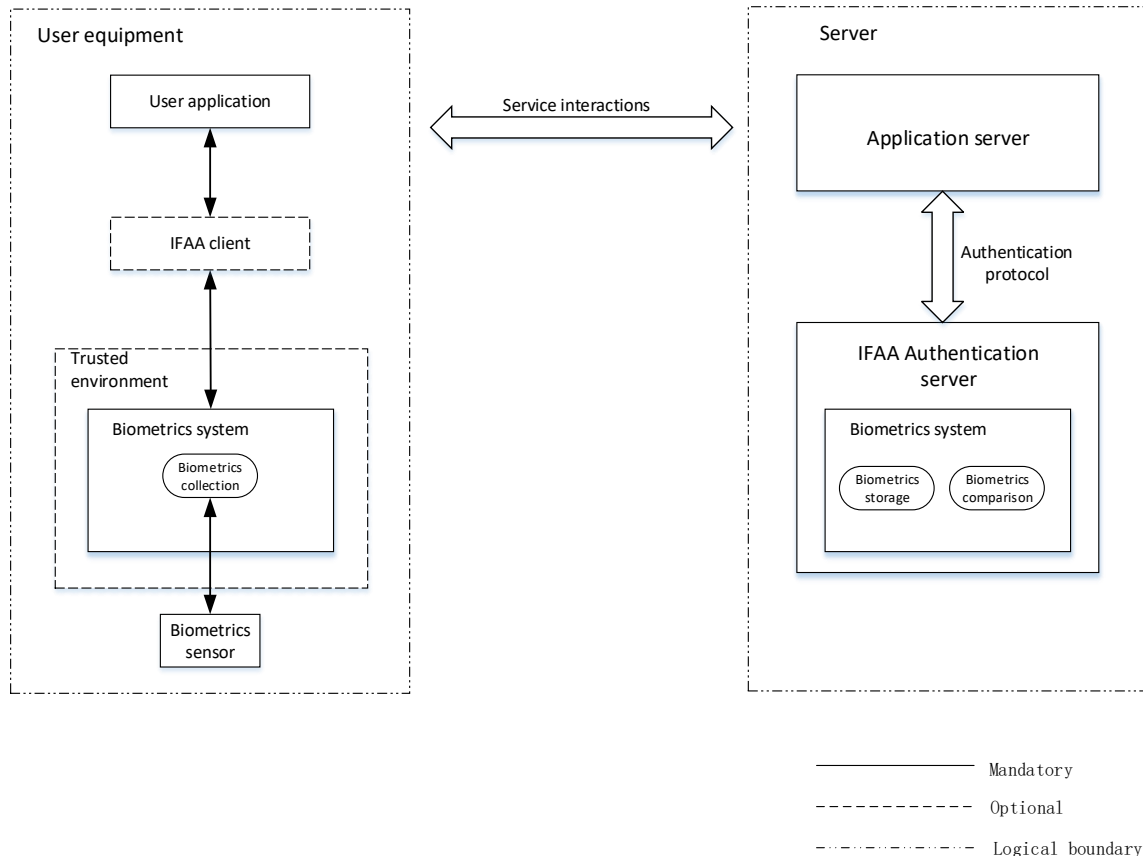
Figure 17 is the message flow of the deregistration protocol:



**Figure 17 – IFAA biometric authentication – local model – Deregistration**

#### 6.4.2 IFAA Biometric Authentication - Remote Model

In the remote model, the biometrics system is divided into two parts: the biometrics collection module resides in the user equipment, but the biometrics storage module and comparison module reside in the authentication server. The biometric data are collected locally but not stored or compared locally, instead they are sent to the server side by the user application and verified by the biometrics system in the authentication server. Figure 18 is the technical framework of this model.



**Figure 18 – IFAA biometric authentication – remote model**

#### 6.5 Aadhaar Authentication

Aadhaar refers to a 12-digit random identification number issued by the Unique Identification Authority of India (UIDAI). It is the largest national biometric database in the world and the Authority has issued more than 1180 million Aadhaar numbers so far.

UIDAI has been tasked with three key functional processes: enrolment, identification, and authentication. Through an extensive network of enrolment agencies, UIDAI collects the demographic (name, date of birth, gender, address) and biometric (fingerprints, iris scan and photograph) information from individuals for the purpose of enrolling them into the Aadhaar system.

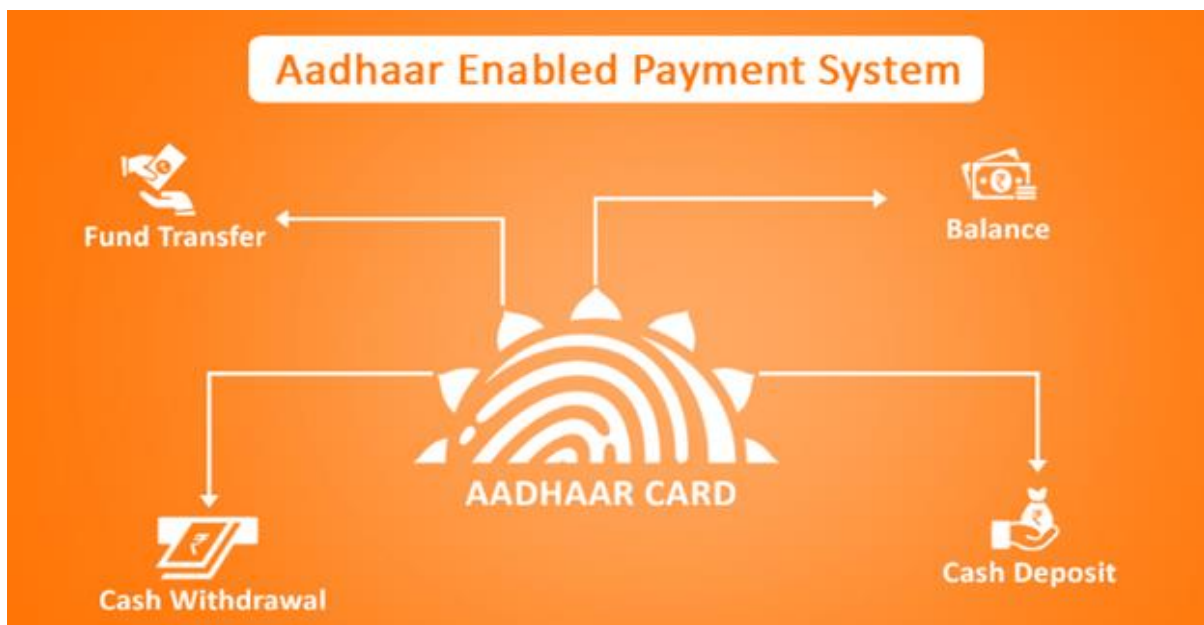
The biometric and demographic data is maintained in a Central Identities Data Repository (CIDR), identity claims and authentication services are provided through open Application Programming Interfaces (APIs) with yes/no answers. Several applications like eSign, digital locker, mobile banking apps etc. use Aadhaar biometric based authentication services.

UIDAI provides the following functional processes to enroll and verify identity of users of Aadhaar

- **Enrolment process:** creating and storing an enrolment data record for an individual who is the subject of a biometric capture process in accordance with the enrolment policy. The subject usually presents his/her biometric characteristics to a sensor along with his/her identity reference. The captured biometric sample is processed to extract the features which are enrolled as a reference in the enrolment database with identity reference.
- **Verification process:** testing a claim that an individual who is the subject of a biometric capture process is the source of a specified biometric reference. The subject presents his/ her identity reference for a claim of identity and biometric characteristic(s) to the capturing device, which acquires biometric sample(s) to be used for comparison with the biometric reference linked to the identity reference for identification. The verification process has a possibility of impacting a subject's information privacy, since this process requires both biometric reference and identity reference. The identification process requires exhaustive search of enrolment database. So, this also has a possibility of impacting on subject's physical privacy. Verification is generally considered to be less privacy intrusive than identification. In Aadhaar system verification is done via online authentication having only a “yes/no” answer.

UIDAI has partnered with various stakeholders including Reserve Bank of India (RBI), National Payments Corporation of India (NPCI), and banks to develop two key platforms:

- **Aadhaar Payments Bridge (APB)** – A system that facilitates seamless transfer of all welfare scheme payments to beneficiary residents' Aadhaar Enabled Bank Account (AEBA).
- **Aadhaar Enabled Payment System (AEPS)** – A system that leverages Aadhaar online authentication and enables AEBA to be operated in anytime-anywhere banking mode by the marginalized and financially excluded segments of society through microATMs.



**Figure 19 - Aadhaar Enabled Payment System Transactions**

Aadhaar Enabled Payment System (AEPS) is a payment service offered by the National Payments Corporation of India (NPCI) to banks, financial institutions using Aadhaar. AEPS is a bank led model,

which allows online financial inclusion transaction at micro-ATM through the business correspondent of any bank using the Aadhaar authentication. This system is designed to handle both ONUS and OFFUS requests seamlessly in an effective way by enabling authentication gateway for all Aadhaar linked account holders.

AEPS empowers the marginalised and excluded segments to conduct financial transactions (credit, debit, remittances, balance enquiry, etc.) through microATMs deployed by banks in their villages.

Four types of transactions are supported by AEPS:

- Balance Enquiry
- Cash Withdrawal
- Cash Deposit
- Fund Transfer
- Aadhaar Pay/Purchase
- Mini Statement
- Aadhaar Status (Bank Linked) through AePS

To make an AEPS, the following information needs to be supplied:

- Transaction Type
- Aadhaar number,
- Bank's Institute Identification Number (IIN)<sup>3</sup>,
- Fingerprint
- Aadhaar number of beneficiary (only in case of Fund Transfer)

The key steps in doing transactions via AEPS are:

- The person provides his/her Aadhaar number, bank name; details of financial transaction sought and fingerprint impression at the microATM device.
- Digitally signed and encrypted data packets are transferred via bank switch to NPCI to UIDAI for user authentication.
- UIDAI processes the authentication request and communicates the outcome in form of *Yes/No*.
- If the authentication response is Yes, the bank carries out the required authorization process and advises microATM on suitable next steps.

The Aadhaar Payments Bridge (APB) is a repository of Aadhaar number of residents and their primary bank account number used for receiving all social security and entitlement payments from various government agencies. It requires using Aadhaar number as the primary key for all entitlement payments. This would maintain the integrity of the system and ensure that the benefits reach the intended beneficiaries. This benefit has an even greater ramification as more and more social security programs are moving from in-kind to in-cash subsidies.

### **6.5.1 APB Process Steps**

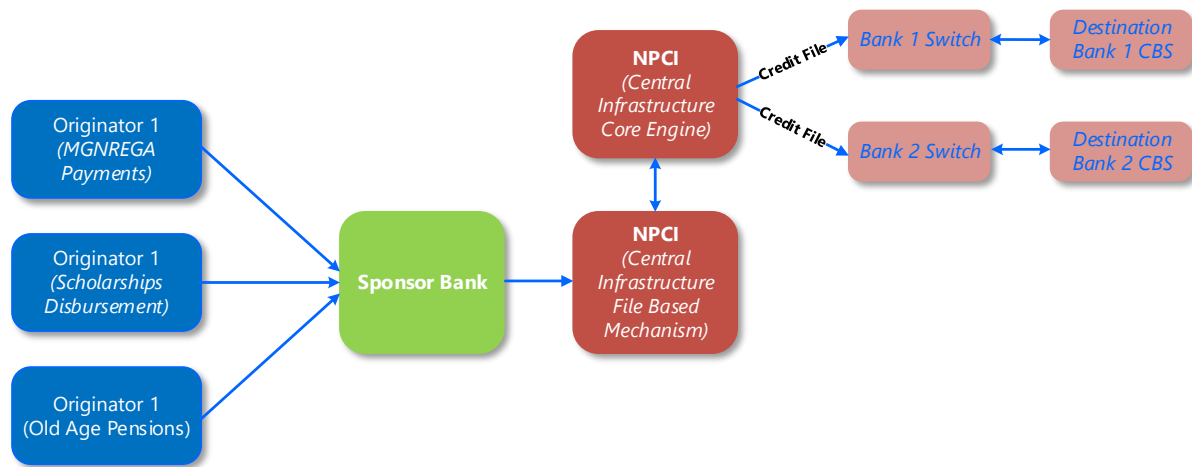
The key steps in posting payments via APB are:

- Service delivery agency that needs to make payments to its beneficiaries (such as wages, scholarships disbursement, old age pension etc.) provides APB file containing details of Aadhaar number, welfare scheme reference number and the amount to be paid to its bank (referred to as a sponsor bank).
- Sponsor bank adds bank's Institute Identification Number (IIN) provided by NPCI to participant banks to the APB file and uploads onto NPCI server.
- NPCI processes uploaded files, prepares beneficiary bank files and generates settlement file

---

<sup>3</sup> IIN - is a six digit number which identifies the Bank with which the person has mapped his Aadhaar number

- Settlement file is posted to bank accounts with Reserve Bank of India.
- Destination banks can download the incoming files for credit processing after the settlement file has been processed.



**Figure 20 - Aadhaar Payments Bridge Process**

### 6.5.2 Types and modes of authentication for Aadhaar

There are two types of authentication, namely—

- Yes/No authentication facility,
- e-KYC authentication facility, which may be carried out only using OTP and/or biometric authentication modes.

The following modes of authentication are supported:

- Demographic authentication:** The Aadhaar number and demographic information of the Aadhaar number holder obtained from the Aadhaar number holder is matched with the demographic information of the Aadhaar number holder.
- One-time pin based authentication:** A One Time Pin (OTP), with limited time validity, is sent to the mobile number and/ or e-mail address of the Aadhaar number holder registered with the Authority, or generated by other appropriate means. The Aadhaar number holder shall provide this OTP along with his Aadhaar number during authentication and the same shall be matched with the OTP generated by the Authority.
- Biometric-based authentication:** The Aadhaar number and biometric information submitted by an Aadhaar number holder are matched with the biometric information of the said Aadhaar number holder. This may be fingerprints-based or iris-based authentication or other biometric modalities based on biometric information stored.
- Multi-factor authentication:** A combination of two or more of the above modes may be used for authentication – chosen by a requesting entity for enhanced security.

e-KYC authentication is carried out using OTP and/or biometric authentication and not demographic.

### 6.5.3 Aadhaar authentication security concerns

Ideally, for any system, identification and authentication without consent should not be possible. In Aadhaar, the single unique identifier, which is needed to identify the user across multiple domains,

has been at the centre of the security issues. For instance, the Aadhaar number is needed at the time of authentication.

Some of the security threats around consumer related information and data privacy in Aadhaar are:

1. Correlation of identities across domains: It may become possible to track an individual's activities using their Aadhaar id. This would lead to identification without consent.
2. Identification without consent using Aadhaar data: There could be risks of unauthorised use of biometrics to illegally identify people.
3. Illegal tracking of individuals: Individuals may be tracked without proper authorisation or legal sanction using the authentication and identification records and trails in the Aadhaar database, or in one or more AUA's databases. Such records will typically also contain information on the precise location, time and context of the authentication or identification, and the services availed.
4. Possible collusion of an attacker with inside personnel can also lead to data breaches under items 2 and 3 above.

#### **6.5.4 Security measures introduced recently to address those threats**

In 2018, the government in India introduced a number of security measures to address these threats:

##### **a) Virtual ID**

UIDAI introduced a system of virtual identification for Aadhaar cardholders, in a bid to prevent a security breach of all the user information from the database. With this 'Virtual ID,' the cardholders can generate a 16 digit temporary number, which can be used to access various platforms such as banks, insurance or telecom service providers. Agencies that undertake authentication would not be allowed to generate the Virtual ID on behalf of Aadhaar holder. The virtual ID is linked to the Aadhaar number but it is not permanent in nature. It is temporary and there are less risks in it being misused. With the virtual ID, there will be no need to share the user's Aadhaar number at the time of authentication. It is revocable and can be replaced with a new one.

- b) **Limited KYC**, which does not return Aadhaar number so that only an agency specific unique UID token is given to eliminate many agencies storing Aadhaar local AUA<sup>4</sup>s and global AUAs. Category of global AUAs will have access to e-KYC with Aadhaar no, while all other will have access to limited KYC for paperless KYC process. Once the UIDAI receives an authentication request from the local AUA, it will lend it a unique identity token, a 72 character alphanumeric string that will work only on the local AUA's system. UID token allows an agency to ensure uniqueness of its beneficiaries, customers etc. without having to store Aadhaar number in their databases while not being able to merge databases across agencies thus enhancing privacy.

##### **c) Biometric locking**

This service is meant to help users protect their biometric details from being misused in one way or the other. It is worth noting that many agencies require applicants to verify their details

---

<sup>4</sup> Authentication User Agency (AUA) provides services to users that are successfully authenticated. Examples of AUAs and services are banks, various state and central government ministries providing services and even private agencies like mobile phone operators. An AUA is required to enter in to a formal contract with UIDAI to be able to use Aadhaar authentication services.



using the Aadhaar biometric authenticate facility. UIDAI may enable an Aadhaar number holder to permanently lock his biometrics and temporarily unlock it when needed for biometric authentication.

All biometric authentication against any such locked biometric records shall fail with a “No” answer with an appropriate response code. An Aadhaar holder shall be allowed to temporarily unlock his/her biometrics for authentication, and such temporary unlocking shall not continue beyond the time period specified by UIDAI or till completion of the authentication transaction, whichever is earlier.

UIDAI can enable Aadhaar holders to remove such permanent locks at any point in a secure manner.

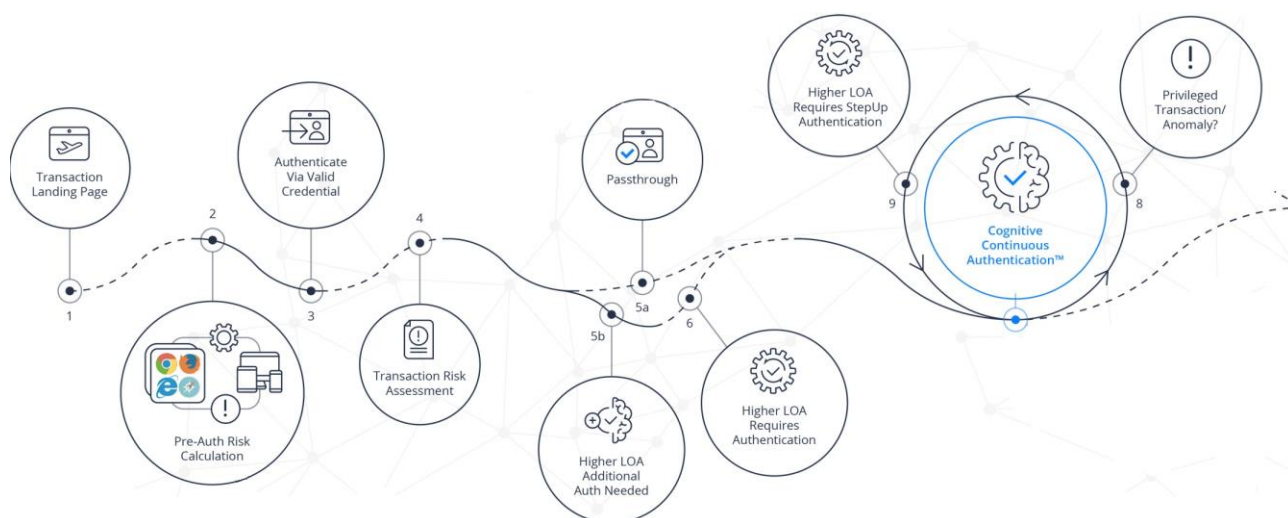
## 6.6 Cognitive Continuous Authentication

The significant security, privacy and usability shortcomings of the current consumer identity management systems in the financial sector require a paradigm shift away from usernames, passwords and other forms of temporal, binary and biometrics controls.

This type of transformation is warranted today through a combination of multi-modal and contextual controls that continuously and accurately protect user identity and privacy even if your online credentials are already compromised. Cognitive Continuous Authentication™ uses AIML and a combination of multi-modal and contextual controls that continuously and accurately protect user transactions, and identity and privacy. Pairing AI with a mixture of Machine learning (AIML) can be used in the background, learning the digital behavior of users within context. By taking a holistic approach to how someone transacts, AI can determine if a bad actor is trying to initiate a fraudulent transaction.

Cognitive Continuous Authentication™ starts collection of intelligence pre-authentication, uses a rich set of contextual data instead of binary authentication to deliver a new state of the art risk-based authentication with lower friction for the good actors and then most importantly a post authorization continuous authentication that detects transaction anomalies leveraging the new controls including the use of AIML.

Figure 21 below shows the users journey in Acceptto's Cognitive Continuous Authentication™ which includes Pre-Auth Intelligence, Context Aware Risk Based Auth, and post-authorization Continuous Authentication



**Figure 21 – Acceptto's Cognitive Continuous Authentication™**

When a user's action seems off, AI can more readily ping the application that something is "off" about the current session. An artificial intelligence system for intrusion and anomaly detection, such as Accepto's Cognitive Continuous Authentication™, applies *machine learning techniques*. These algorithms process and analyze large quantities of previously observed logins and additional contextual data. They *learn* the characteristics and patterns which enable them to classify requests and detect abnormal activity and possible threat actors.

## 6.7 Decentralized Identity and Distributed Ledgers

Traditional identity management systems are built on top of centralized authorities such as corporate directory services, certificate authorities, or domain name registries. Each of these organizational centralized authorities serves as their own root of trust. Identity federation emerged as a stopgap solution that enabled identity management systems to work across systems with different roots of trust.

The emergence of distributed ledger technology (DLT) provides the opportunity for developing a new approach to decentralized identity systems. In a decentralized identity system, entities are able to use any shared root of trust. Distributed ledgers provide a means for managing a root of trust with neither centralized authority nor a single point of failure. In combination, DLTs and decentralized identity systems enable any entity to create and manage their own identifiers on any number of distributed, independent roots of trust [10].

One approach to decentralized identity systems has been labeled "Self-Sovereign Identity". The proponents of this approach have developed a set of design principles [11]:

1. Existence: Entities must have an independent existence
2. Control: Entities must be able to control their identities, they should be able to refer, update or hide it.
3. Access: Entities should have access to their own identity and related data.
4. Transparency: The system and its logic must be transparent in how they function.
5. Persistence: Identities must be long-lived, at least for as long the user desires but it should not contradict the "user" right to be forgotten.
6. Portability: Information about identities must be transportable.
7. Interoperability: Identities should be as widely usable as possible.
8. Consent: Entities must agree to the use of their identities and the sharing of related data.
9. Minimization: Disclosure of claims must be minimized.
10. Protection: The right of entities must be protected, when there is a conflict between the needs of the network and the right of entities, the priority should be the latter.

Most central-authority identity solutions today have limited support for every principle, in particular, control over identity, transparency and portability.

The following sections describe key components of these new decentralized identity systems: verifiable credentials, decentralized identifiers, decentralized identifier authentication and resolution, and personal cryptographic key wallets.

### 6.7.1 Decentralized Identity Definition of Terms

The emerging decentralized identity system standards use modernized, refined definitions of key terms. This section has definitions from the W3C Verifiable Credentials [12] specification. Note that some of these newly-defined terms may conflict with older definitions of the same terms, or definitions in other standards.

#### **subject**

An entity about which claims are made.

## claim

An assertion made about a subject.

## credential

A set of one or more claims made by an issuer. A verifiable credential is a tamper-evident credential that has authorship that can be cryptographically verified. Verifiable credentials can be used to build verifiable presentations, which can also be cryptographically verified. The claims in a credential can be about different subjects.

## decentralized identifier

A portable URL-based identifier, also known as a DID, associated with an entity. These identifiers are most often used in a credential and are associated with subjects such that a credential itself can be easily ported from one repository to another without the need to reissue the credential. An example of a DID is did:example:123456abcdef.

## identity

The means for keeping track of entities across contexts. Digital identities enable tracking and customization of entity interactions across digital contexts, typically using identifiers and attributes. Unintended distribution or use of identity information can compromise privacy. Collection and use of such information should follow the principle of data minimization.

### 6.7.2 Decentralized Identity System Infrastructure Layers

The Sovrin Foundation [13] has created an approach to organizing the technology infrastructure components of their decentralized identity system solution. Sovrin uses a ‘layer’ concept to explain the roles, functions and relationships between infrastructure components as shown in Figure 22.

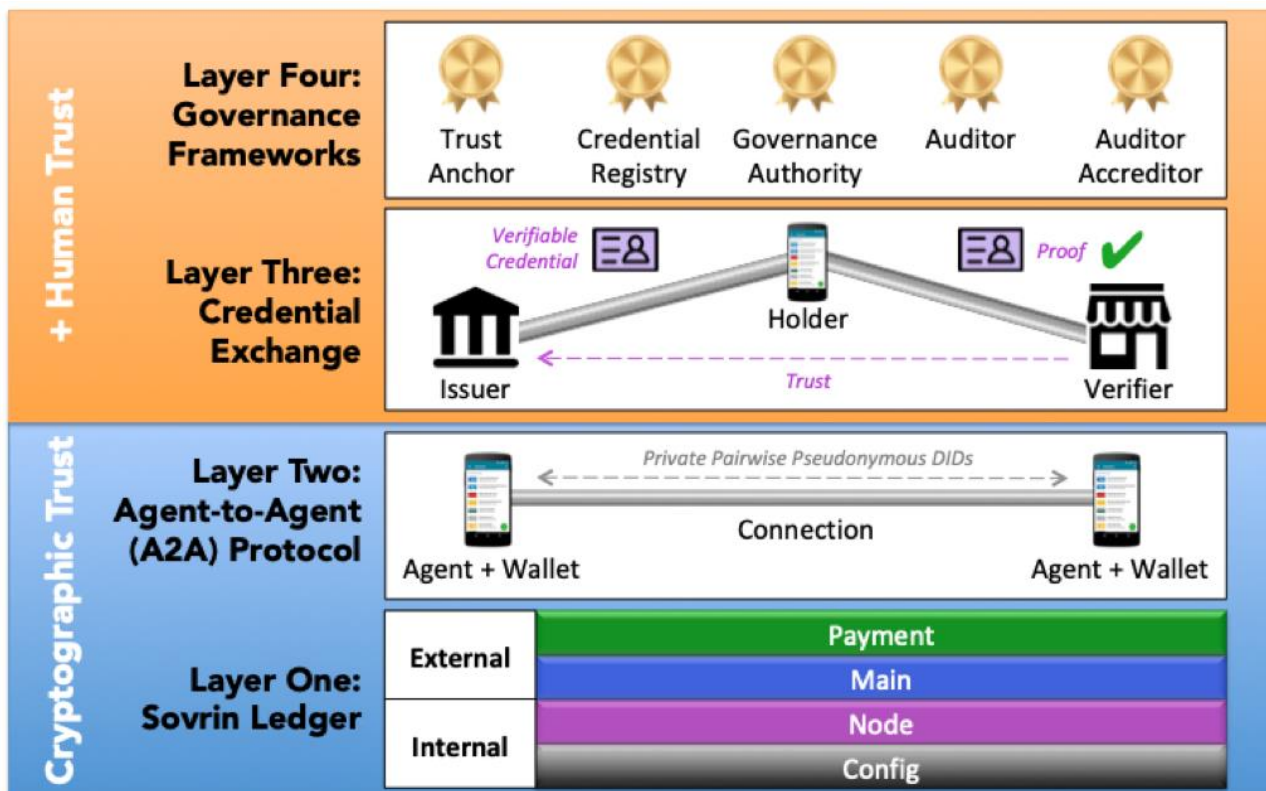


Figure 22 – Sovrin Infrastructure Layers

Layer 1, the Sovrin Ledger layer, contains the component DLTs that underpin the Sovrin solution. Credential issuers who need their credential to be publicly verifiable store their Sovrin identities and decentralized identifiers in these DLTs. Schemas, credential definitions and revocation registries are also located in layer one.

Layer 2, the Agent-to-Agent layer, contains communications protocols to enable direct peer-to-peer credential, agent and cryptographic wallet communications. This layer does not contain a DLT. Together, layer two and layer one provide cryptographic trust between software and hardware components.

Layer 3, the Credential Exchange layer, provides the mechanisms for credential issuers to issue verifiable credentials to holders, and holders to generate proofs to verifiers. Verifiers check the cryptographic proofs to gain certainty that the asserted claim within the credential is valid according to the issuer.

Layer 4, the Governance Frameworks layer, is where business and legal agreements are established to specify the rules that issuers and verifiers must follow.

### **6.7.3 Verifiable Credential and Decentralized Identifier Draft Standards**

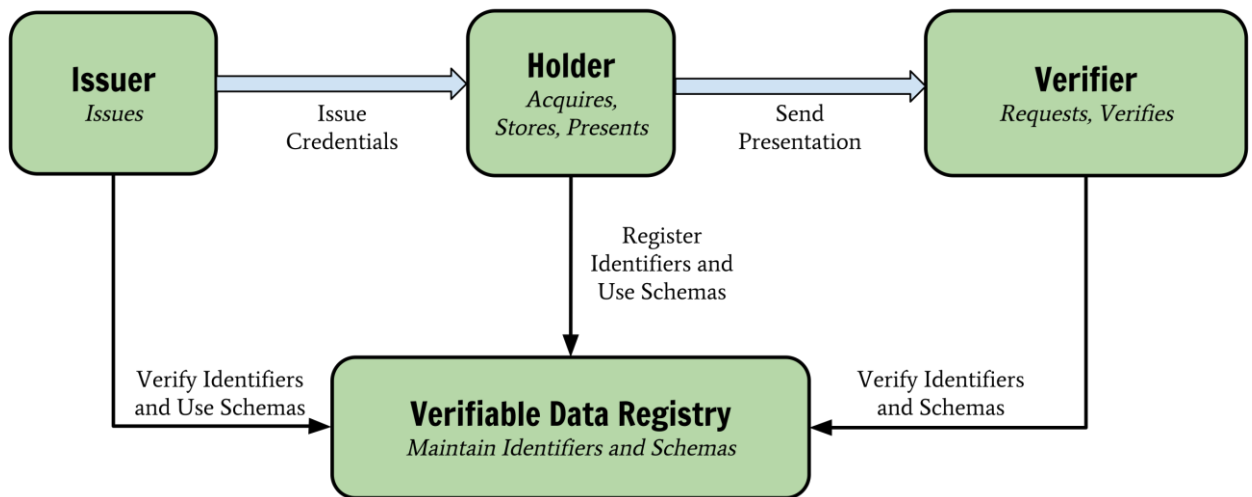
New approaches and technologies are emerging to use distributed ledgers (also known as ‘blockchains’) to establish identity networks that are not dependent on centralized data authorities. These identity networks are described in many different ways by different groups. Two core standards projects are central to these new developments: W3C Verifiable Credentials [12] and W3C Decentralized Identifiers [10]. This group of technologies and standards are still being developed and do not yet have wide adoption.

### **6.7.4 Verifiable Credentials**

From the W3C Verifiable Credentials Data Model specification:

A verifiable credential can represent all of the same information that a physical credential represents. The addition of technologies such as digital signatures makes verifiable credentials more tamper-evident and therefore more trustworthy than their physical counterparts. Holders can generate presentations and share them with verifiers to prove they possess verifiable credentials with certain characteristics. Both credentials and presentations can be rapidly transmitted, making them more convenient than their physical counterparts when establishing trust at a distance.

Figure 23 shows the core roles and concepts of Verifiable Credentials.



**Figure 23 – Roles and Relationships of Verifiable Credentials**

The roles are described in the specification as:

#### **issuer**

A role an entity might perform by asserting claims about one or more subjects, creating a verifiable credential from these claims, and transmitting the verifiable credential to a holder.

#### **verifier**

A role an entity might perform by receiving one or more verifiable presentations for processing. Other specifications might refer to this concept as a relying party.

#### **holder**

A role an entity can perform by possessing one or more verifiable credentials. A holder is usually, but not always, the subject of the verifiable credentials they are holding. Holders store their credentials in credential repositories.

#### **verifiable data registry**

A role a system might perform by mediating the creation and verification of identifiers, keys, and other relevant data, such as verifiable credential schemas and revocation registries, which might be required to use verifiable credentials. Some configurations might require correlatable identifiers for subjects. Example verifiable data registries include trusted databases, decentralized databases, government ID databases, and distributed ledgers. Often there is more than one type of verifiable data registry utilized in an ecosystem.

Verifiable credentials are a central feature in section 7.1.4 Example: Zug eID – Ethereum Blockchain-based Digital ID.

### **6.7.5 Decentralized Identifiers**

The Decentralized Identifier (DID) specifications are being created to establish a cryptographically verifiable, globally-addressable identifier namespace for distributed ledger and blockchain systems. Decentralized Identifiers are the addressing scheme used for Verifiable Credentials.

From the W3C Decentralized Identifier draft specification:

- The emergence of distributed ledger technology (DLT), sometimes referred to as blockchain technology, provides the opportunity for fully decentralized identity management. In a decentralized identity system, entities are free to use any shared root of trust. Globally distributed ledgers (or a decentralized P2P network that provides similar capabilities) provide a means for managing a root of trust with neither centralized authority nor a single point of failure. In combination, DLTs and decentralized identity systems enable any entity to create and manage their own identifiers on any number of distributed, independent roots of trust.
- The entities are identified by decentralized identifiers (DIDs). They may authenticate via proofs (e.g., digital signatures, privacy-preserving biometric protocols, etc.). DIDs point to DID Documents. A DID Document contains a set of service endpoints for interacting with the entity. Following the dictums of Privacy by Design, each entity may have as many DIDs as necessary, to respect the entity's desired separation of identities, personas, and contexts.
- To use a DID with a particular distributed ledger or network requires defining a DID method in a separate DID method specification. A DID method specifies the set of rules for how a DID is registered, resolved, updated, and revoked on that specific ledger or network.
- This design eliminates dependence on centralized registries for identifiers as well as centralized certificate authorities for key management—the standard pattern in hierarchical PKI (public key infrastructure). Because DIDs reside on a distributed ledger, each entity may serve as its own root authority—an architecture referred to as DPKI (decentralized PKI).

In general, DID design goals are the following [10]:

1. Decentralization: DID architecture should eliminate the requirement for centralized authorities or single points of failure in identity management, including the registration of globally unique identifiers, public verification keys, service endpoints, and other metadata.
2. Entity control of identifiers: DID architecture should give entities, both human and non-human, the power to directly control their own digital identifiers without the need to rely on external authorities.
3. PII Protection: DID architecture should enable entities to control the identifiable data of their digital identities, including minimal, selective, and progressive disclosure of attributes or other identity data.
4. Security: DID architecture should enable sufficient security for relying parties to depend on DID records for their required level of assurance.
5. Proof-based: DID architecture should enable an entity to provide cryptographic proof of authentication and proof of authorization rights.
6. Discoverability: DID architecture should make it possible for entities to discover DIDs for other entities to learn more about or interact with those entities.
7. Interoperability: DID architecture should use interoperable standards so DID infrastructure can make use of existing tools and software libraries designed for interoperability.
8. Portability: DID architecture should be system and network-independent and enable entities to use their digital identities with any system that supports DIDs and DID Methods.
9. Simplicity: To meet these design goals, DID architecture should be “as simple as possible but no simpler”.
10. Extensibility: When possible, DID architecture should enable extensibility provided it does not greatly hinder interoperability, portability, or simplicity.

### 6.7.6 DID Authentication

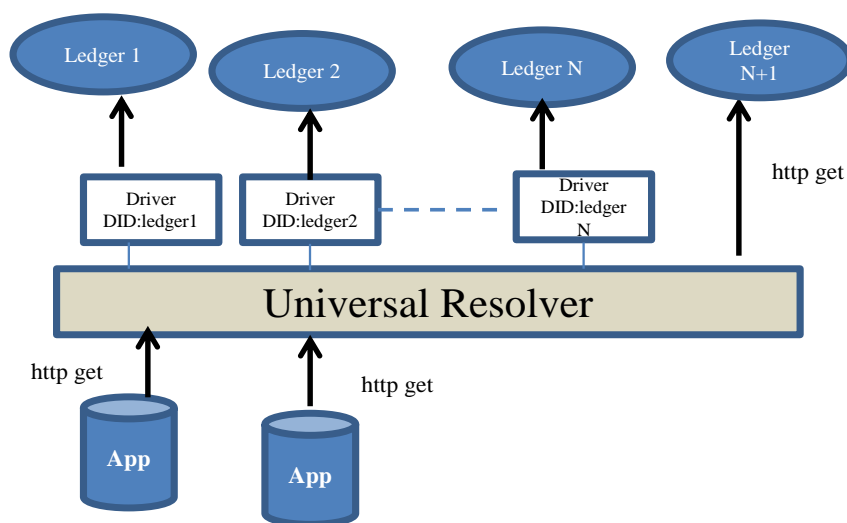
DID Authentication [14] enables a DID Subject to prove control over a DID during its interaction with a relying party. The following general steps to be executed by the relying party include:

1. The relying party retrieves the DID Document associated with the DID Subject
2. The relying party uses the authentication property of the DID Document to determine how to perform DID authentication, for example cryptographic signatures, proving control of a public key or use of an authentication service endpoint
3. The relying party executes the authentication mechanism provided

DID authentication should support web and mobile flows.

### 6.7.7 DID Resolution

The DID specification requires each DLT to have a DID Method specification to describe how DID operations are performed. The implication of having many DID Method specifications is that resolving a text string, the DID, to locate the trust root and the associated DID Document is complex. The DID resolution function could become a major impediment to interoperable DIDs. Work has begun on a universal DID resolver architecture and toolset that can take any valid DID as input and resolve it to a DID Document. The universal resolvers are specifically designed to work for decentralized identifiers and support DID resolution over many different types of DLT and networks [15]. The universal resolver approach solves the problem of heterogeneous networks having different method specifications for their own DID. Figure 24 depicts the Universal Resolver concept [15].



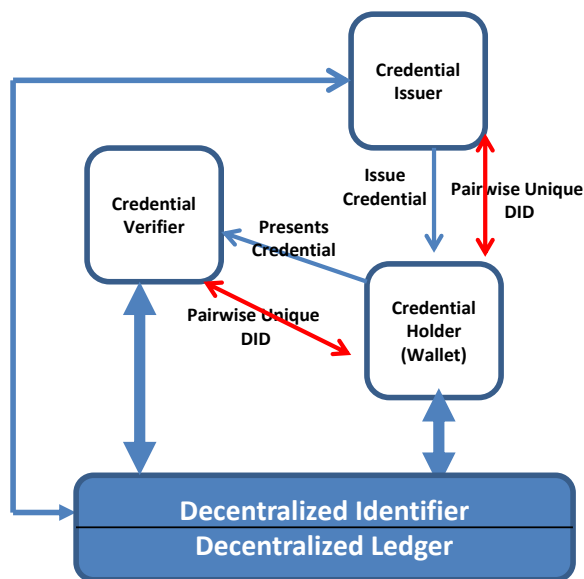
**Figure 24 – Universal DID Resolver**

### 6.7.8 Decentralized Identity Wallets

The individual must have software and/or hardware that enables them to interact with the decentralized identity system. These components are agents and wallets [16].

The primary function of an agent is to communicate with other agents and coordinate DID resolution and authentication. The agent keeps track of DIDs related to other entities in the network. An agent contains or is connected to a wallet where cryptographic secret keys are kept and protected. The wallet contains the essential private keys that allow the individual to prove control over a DID and thus participate in the decentralized identity system. The agent and wallet hold verifiable credentials and proofs belonging to the individual.

The wallet can be entirely on the user’s device or a virtual wallet where one part of the wallet is on the user mobile device and another part in the cloud. The latter configuration enables the creation of agents to act on behalf of the user and perform services without the need for user direct involvement.



**Figure 25 – Decentralized Identity Wallet with Verifiable Claims**

Figure 25 depicts the overall identity interactions in support of an identity-based service. Because the wallet contains all the material needed to assume the identity of the wallet owner, user authentication to the wallet should use a strong, password-less authentication method.

## 7 Implementation examples of Strong Authentication Systems

This section contains examples of strong authentication systems that cover DFS use cases. The examples also illustrate mechanisms related to the authentication assurance phases of ITU-T Recommendation X.1254.

**Table 3 – Digital Financial Services Use Case Examples**

Authentication Assurance Phase	DFS Use Cases	Use case examples
Enrolment	Account opening (Section 7.1): <ul style="list-style-type: none"> <li>▪ eKYC</li> <li>▪ Credit checks</li> </ul>	<ul style="list-style-type: none"> <li>▪ Aadhaar eKYC (Section 7.1.1)</li> <li>▪ Sierra Leone National Digital Identity and Credit Platform – Kiva (Section 7.1.2)</li> <li>▪ K-FIDO Enrolment (Section 7.1.2)</li> <li>▪ Zug eID – Ethereum Blockchain-based Digital ID (Section 7.1.4)</li> <li>▪ FIDO Enrolment (Section 7.1.5)</li> <li>▪ Healthcare provider user enrolment (Section 7.1.6)</li> </ul>



Authentication	Access a Digital Financial Service (Section 7.2): <ul style="list-style-type: none"> <li>▪ Storing Funds</li> <li>▪ Buying</li> <li>▪ Paying Bills</li> <li>▪ Sending/receiving funds</li> <li>▪ Borrowing</li> <li>▪ Saving</li> <li>▪ Insuring Assets and Risks</li> </ul>	<ul style="list-style-type: none"> <li>▪ Alipay fingerprint payment (Section 7.2.1)</li> <li>▪ Aadhaar authentication (Section 7.2.2)</li> <li>▪ K-FIDO Authentication (Section 7.2.3)</li> <li>▪ Healthcare provider Next-Generation Authentication (Section 7.2.4)</li> <li>▪ SK Telecom - Mobile Connect Authentication (Section 7.2.5)</li> </ul>
----------------	--	---

## 7.1 Use case: Enrolment and Account opening

The examples presented for the Enrolment use case describe how previously-established identity information can be used to create new service accounts and to satisfy KYC requirements. The key aspect in the examples is that the person has been enrolled previously with an authority: their identity information collected, verified and stored. This stored identity information is then available for later presentation to service providers, controlled by the person's authentication to release that identity information.

Use of digital sources of identity information for not-in-person KYC and account opening is both convenient for the person but also presents risks for impersonation. Therefore, use of strong authentication mechanisms is recommended.

### 7.1.1 Example: Aadhaar eKYC

eKYC service allows resident to authorize Unique Identification Authority of India (UIDAI) to share electronic version of Aadhaar information (demographic information and photo only) with the explicit authentication of the resident. In eKYC service, UIDAI encrypts the eKYC response data containing resident's latest demographic and photograph information using an e-KYC User Agency (KUA) public key and subsequently forwards the encrypted response to KUA. On receiving the encrypted response, the KUA decrypts the data using their own private key and returns an eXtensible Markup Language (XML) with seven pieces of data - name, address, date of birth, gender, phone number, e-mail address and photograph, this eliminates collecting photocopy of Aadhaar letter from resident.

Some of the benefits of Aadhaar-based eKYC are described below:

- Activation – there is no requirement for filling up of Customer Application Form (CAF) and submission of photograph along with Proof of Identity (POI) and Proof of Address (POA) documents.
- Secure process - customer's data is fetched from central UIDAI server in encrypted format and not stored on any of the Point of Sale (POS) terminals except for the company's server.
- No document copy or photograph is required – this gives additional confidence to the customers as they don't need to submit any documents which can be later misused by the retailers for pecuniary gains.
- Extremely quick activation – as against the traditional process for activation of SIM card which could take between 12-24 hours, the SIM card is activated in very short time once the form gets submitted from the POS terminal to the company's back office. This scores very high on customer satisfaction.

- Apart from the above benefits, this process also helps Telecom Service Providers (TSPs) do away with archaic processes of CAF collection, data entry, document scanning, tele-verification and physical storage and retrieval of CAFs and documents from the warehouse. As an outcome of this, TSPs are able to store the KYC information of their customers in an electronic format which can be retrieved very quickly being an electronic record.
- The process also benefits the government authorities be it the TERM Cells for audits and the law enforcement agencies (LEAs) as this is a highly compliant process and will significantly help in traceability of the customer should there be need for any law enforcement requirement.

### **7.1.2 Example: Sierra Leone National Digital Identity and Credit Platform – Kiva**

A new partnership [15] between Kiva, Sierra Leone and the United Nations (UNDP & UNCDF) is set to bring a nationwide digital identification system to the people of Sierra Leone to provide citizens with formal identity and control over their own credit information.

President Julis Maada Bio of Sierra Leone announced the initiative during his address to the United Nations General Assembly on September 27, 2018.

The centerpiece of the partnership is Kiva Protocol [16], a technology platform which enables a country to create and establish a national digital identification system using distributed ledger technology (DLT). Two of the major barriers to accessing financial services are a lack of formal identification and a lack of verifiable credit history. The ultimate goal of the initiative is to enable national-level financial inclusion initiatives and bring financial products and services to populations currently lacking them.

Kiva Protocol is designed to address these barriers by extending national civil digital identification to all citizens, thus enabling formal and informal financial institutions to perform near-real-time eKYC verification and credit reporting.

Currently, unbanked people cannot leverage financial transactions from the ‘informal economy,’ such as history with a local microfinance institution (MFI), to build their credit histories. Kiva Protocol integrates with a wide range of financial institutions to include their transactions in a person’s credit history—from commercial bank loans to smaller MFI loans—to help people access the financial services they need, including loans for businesses, education, and basic medical services. Kiva is building the system that will record these transactions using open-source DLT frameworks supported by the Linux Foundation.

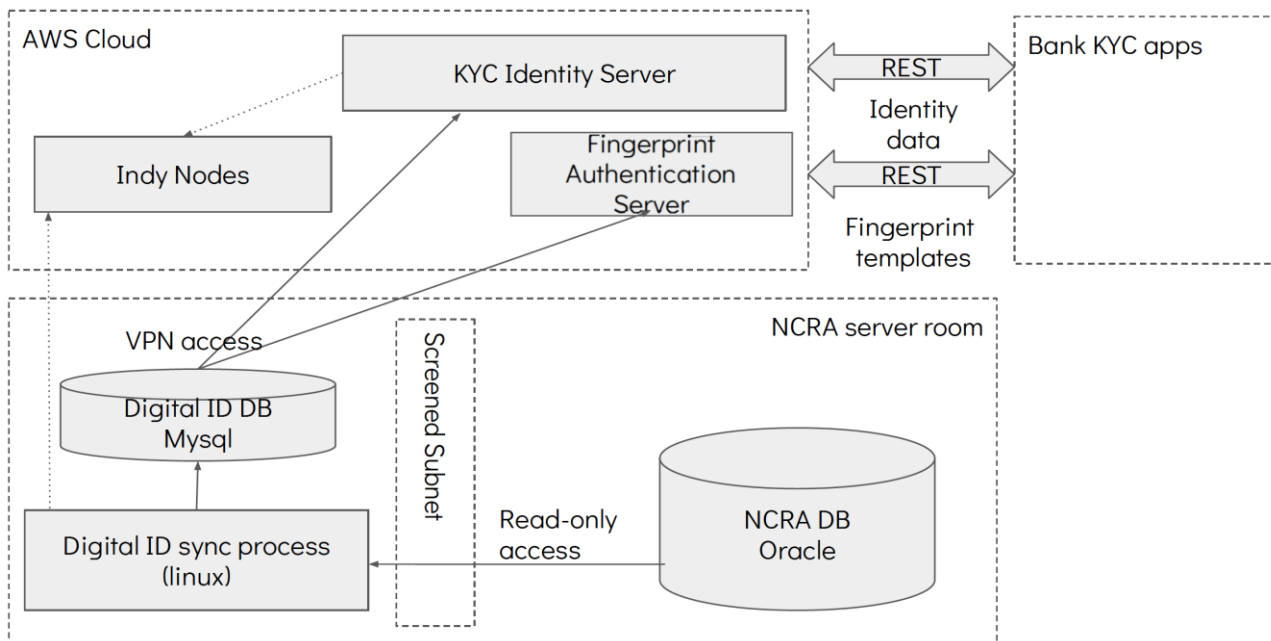
Components of the digital credit reporting ecosystem include:

- **Digital ID & eKYC**  
The National Civil Registration Authority (NCRA)-supported digital ID enables universal unique identification of all citizens, which is critical to accurate credit federation and profiling. Additionally, this digital ID is valid for eKYC verification.
- **Credit Reporting**  
Financial Service Providers report credit data with the Credit Reporting Bureau (CRB), the only national-scale credit bureau in Sierra Leone which is housed at the Bank of Sierra Leone. Such reporting is facilitated by simple API integrations with existing financial service provider platforms.

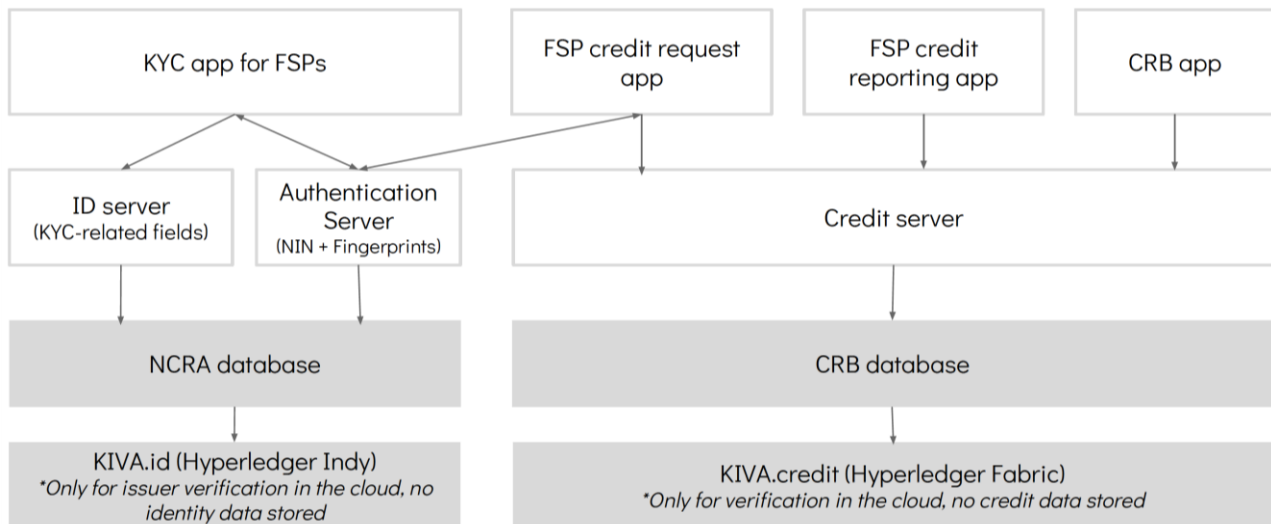
- **Credit Lookup**  
CRB generates credit reports and scores based on the subject's credit history, with consent from the subject.
- **Data-Driven Risk Assessment**  
Comprehensive credit registry enables effective risk assessment and competitive rate setting by financial service providers.

Figure 27, Figure 28 and Figure 29 show high level identity, credit reporting and overall ecosystem architectures.

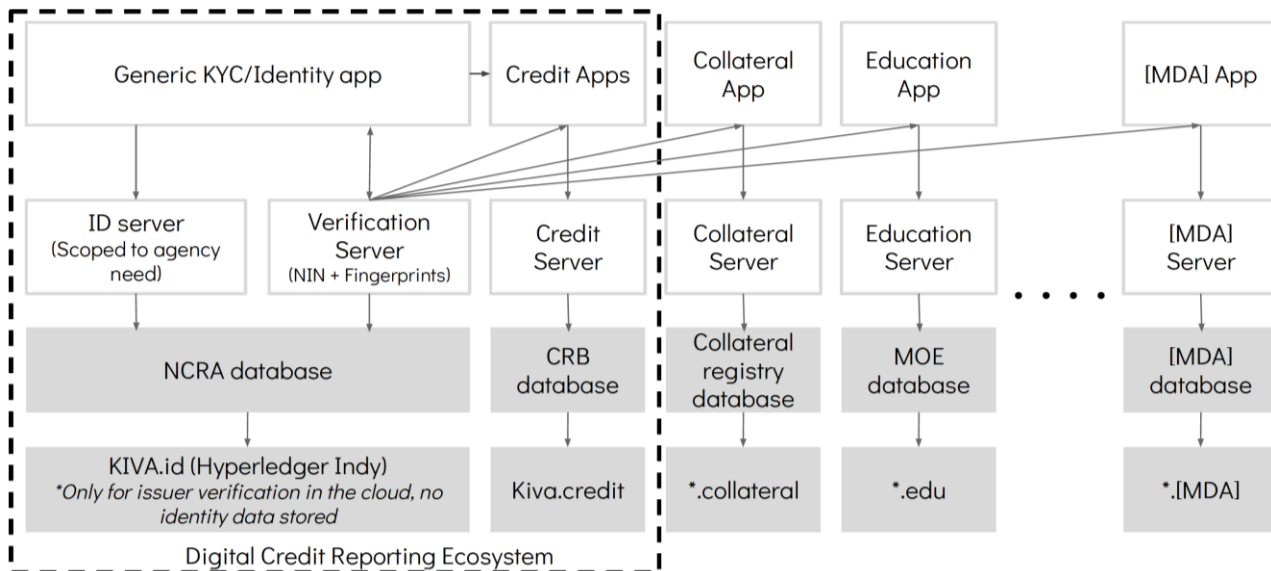
At time of writing, the partners have deployed the identity solution at NCRA, and are integrating eKYC verification and credit reporting with all financial institutions in Sierra Leone. Detailed solution documentation will be published in 2020 as the system goes live and integrations are made available to other financial ecosystem participants in Sierra Leone.



**Figure 26 – NCRA Identity Infrastructure**



**Figure 27 – Digital Credit Reporting Ecosystem Architecture**

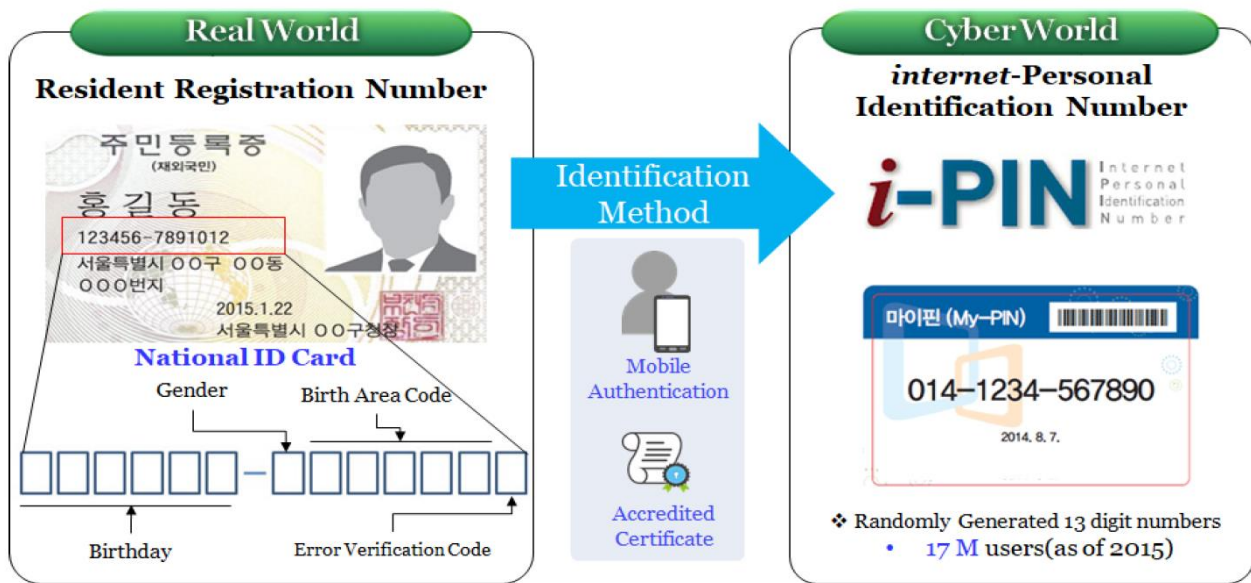


**Figure 28 – Ecosystem Architecture**

### 7.1.3 Example: K-FIDO Enrolment example

This section provides a use case that is based on the FIDO specification. It describes how “K-FIDO” combines FIDO UAF specification and PKI to enable authentication and ID verification at the same time for successful commercial Fintech deployments in Korea. K-FIDO is a specification to be published by KISA (Korea Internet Security Agency), enabling biometric accredited certification services that provide accredited certificates without password using FIDO in Korea.

Korean National ID is used in offline identification and contains a unique Resident Registration Number. To facilitate private and secure online identification and authentication, an i-PIN backed by a PKI certificate issued by a small number of service providers can be generated and associated with the Resident Registration Number. Figure 29 illustrates this relationship.



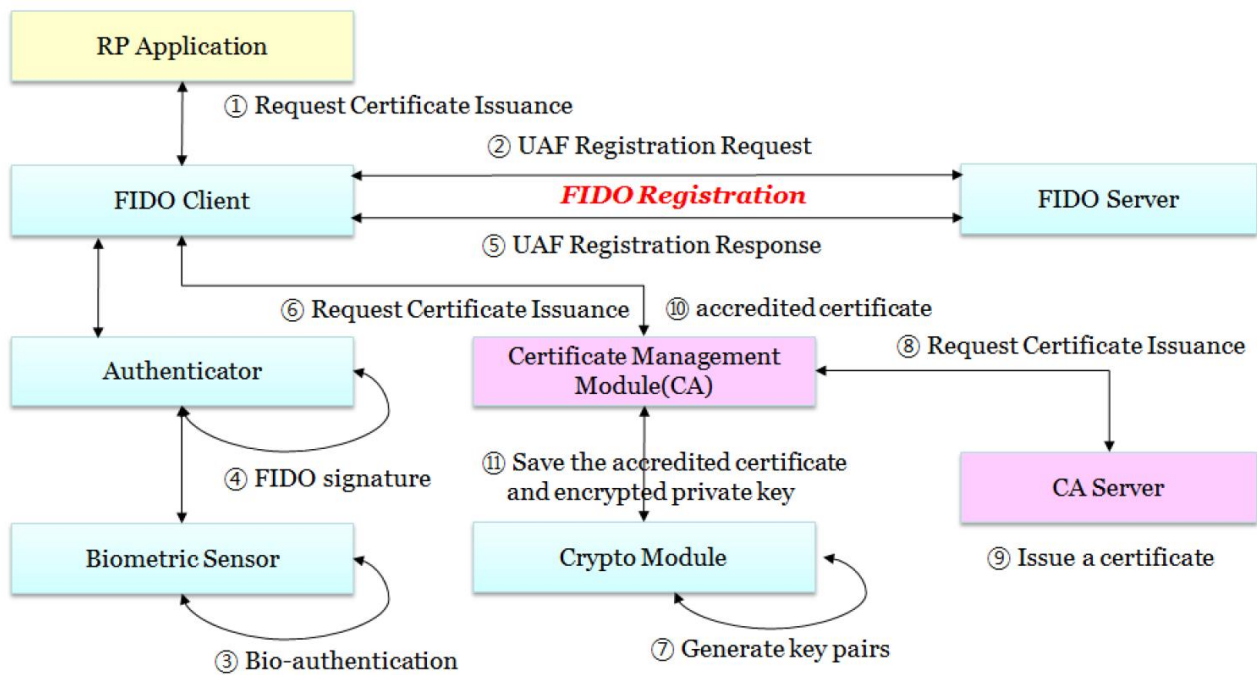
**Figure 29 – National ID and i-PIN in Korea**

The citizen can use many identification methods such as accredited certificates, mobile, bank accounts, and credit cards for internet services that request an online (i.e. non face-to-face) identification method.

Online service providers can choose Identification methods such as Accredited Certificates, Mobile Authentication, i-PIN, K-FIDO, or FIDO depending on the required authentication levels of assurance.

The citizen must register in order to connect their PKI certificate and i-PIN to their FIDO-enabled mobile device. Once registered, the citizen identity data can be provided to other service providers after a strong FIDO authentication.

Figure 30 illustrates the registration process.



**Figure 30 – Registration process of K-FIDO service**

- ① RP App starts bio-registration and requests a user certificate issuance.
- ② The FIDO server triggers a UAF registration request to the FIDO client.
- ③ The user performs a bio-authentication with FIDO authenticators using their respective user verification method, e.g. fingerprint, iris, etc.
- ④ The selected FIDO authenticator generates the FIDO authentication private key. The selected FIDO authenticator generates a FIDO signature using the attestation private key.
- ⑤ The FIDO server verifies the signature using the attestation public and verifies the authentication public key. If verified, the FIDO server trusts the authenticator it is talking to and the authentication public key that was sent from the authenticator in the authentication response. The FIDO server checks FIDO registration message and if passed, the FIDO server stores the authentication public key.
- ⑥ The FIDO client requests the user certificate issuance to the certificate management module.
- ⑦ The crypto module generates a private and public key pair for the user certificate.
- ⑧ The certificate management module requests the user certificate issuance from the certification authority.
- ⑨ The certificate management module stores the user certificate and the private key in the secure element such as USIM, Trustzone, etc. However, the private key should be encrypted by an encryption key in keystore or keychain. The registration process is completed.

Notes on user's identity:

- Before step six happens where the FIDO client requests the user certificate issuance, the user is assumed to have finished user identification using such a mechanism like mobile authentication, accredited certificate, bank account authentication, etc. Thus, the user identity is known at the sixth step.

- The user uses FIDO authentication after the user has finished identification, while it is not tightly coupled. The general scenarios are as follows;
  - 1) A user performs user's identification defined by a service provider.
  - 2) A user uses FIDO or K-FIDO service (the scope of K-FIDO).
- Authenticators decide where the user certificates are stored. KISA recommends secure elements such as keyStore, keyChain, USIM, or Trustzone, etc.

#### **7.1.4 Example: Zug eID – Ethereum Blockchain-based Digital ID**

Since November 2017 the Swiss City of Zug has been offering blockchain-based digital IDs to all of its 30,000 citizens. [17]

The Zug eID consists of three parts. First is the digital vault, which is part of the mobile app. This contains the actual digital ID, which is encrypted; it can be unlocked by the owner biometrically or using a PIN code. Second is the Ethereum blockchain where the app creates a unique cryptographic address for its holder. Third is the certification portal used by the officials who check that the applicant is a resident of Zug.

After the applicant's name, address, date of birth, nationality, and passport number or ID card number have been verified, this data is digitally signed by the City of Zug, and the signature is stored as a certificate in the citizen's digital vault. Since the City's public key is publicly available from the Ethereum blockchain, anyone who receives an eID from its holder can readily verify its authenticity.

After a successful residency check, the City of Zug — itself a digital identity on the blockchain, albeit with special privileges — signs the identity contract of the user, for anyone to see and verify on the Internet. The owner of this special identity is the Zug city clerk.

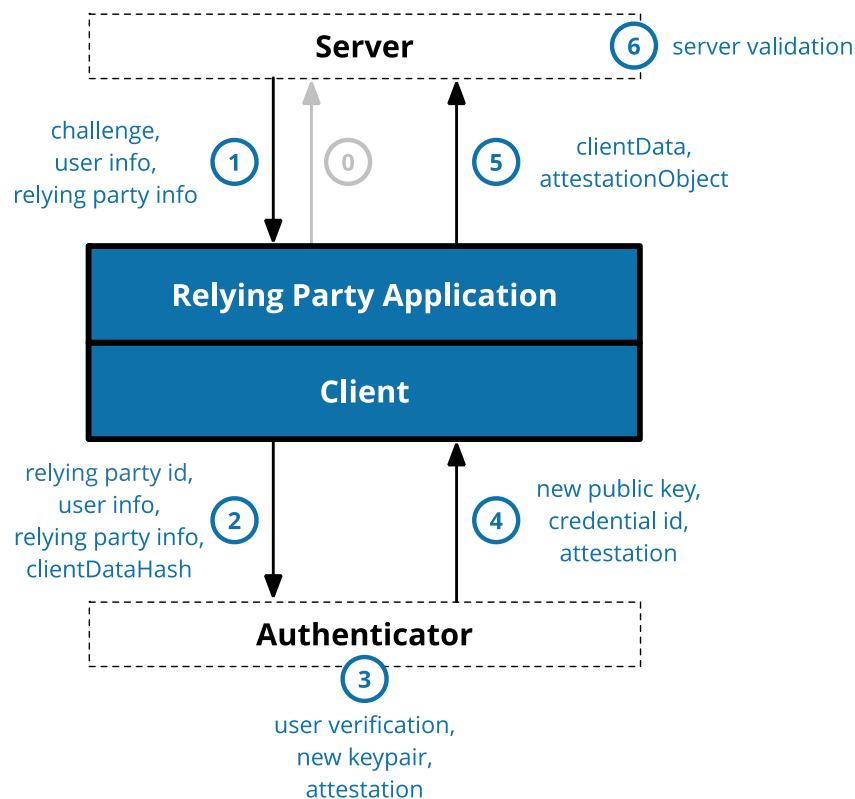
From that moment on, the owner of the eID can use the mobile app to provide identity information. The authenticity of this data can be validated by checking its digital signature on the blockchain.

In the second quarter of 2018 Zug planned to organise a consultation on a specific topic for existing eID holders. Its primary goal was to collect ideas for e-voting based on the new eID.

#### **7.1.5 Example: FIDO Enrolment example**

FIDO specifications have made an explicit and conscientious decision to separate “identity proofing” step from “enrolment” step. The separation allows for a more modular architecture whereby any identity proofing technique can be combined with FIDO enrolment, including Alipay, Aadhaar eKYC, existing PKI credentials (such as K-FIDO above) and various NIST / FIPS LOAs.

A preferential architecture with FIDO is that strong identity proofing is performed once, and then identity is bound to cryptographically and physically secure credentials.



**Figure 31 – Registration process of FIDO**

The enrolment steps are:

- ① **Application Requests Registration** - The application makes the initial registration request after completing identity proofing / KYC and within the same session or trusted environment.
- ② **Server Sends Challenge, User Info, and Relying Party Info** - The server sends a challenge, user information, and relying party information to the Relying Party Application. The Relying Party Application can be a mobile, web, native, or other application and its implementation is outside of the scope of the FIDO specifications. The protocol for communicating with the server is not specified and is also outside of the scope of FIDO. Typically, server communications would be REST over TLS, but they could also be SOAP, RFC 2549 or nearly any other protocol provided that the communication channel is secure. The parameters received from the server will be passed to the client to create credentials, typically with little or no modification.
- ③ **Client Calls authenticatorMakeCredential on Authenticator via CTAP** - Internally, the client will validate the parameters and fill in any defaults, which become the clientData. One of the most important parameters is the origin, which is recorded as part of the clientData so that the origin can be verified by the server later. The parameters to the credentialCreate call are passed to the authenticator, along with a SHA-256 hash of the clientData (only a hash is sent because the link to the authenticator may be a low-bandwidth NFC or Bluetooth link and the authenticator is just going to sign over the hash to ensure that it isn't tampered with).
- ④ **Authenticator Creates New Key Pair and Attestation** - Before doing anything, the authenticator will typically ask for some form of user verification. This could be entering a PIN, using a fingerprint, doing an iris scan, etc. to prove that the user is present and consenting to the registration. After the user verification, the authenticator will create a new asymmetric key pair and safely store the private key for future reference. The public key will become part of



the attestation, which the authenticator will sign over with a private key that was burned into the authenticator during its manufacturing process and that has a certificate chain that can be validated back to a root of trust.

- ④ **Authenticator Returns Data to Client** - The new public key, a globally unique credential id, and other attestation data are returned to the client where they become the attestationObject.
- ⑤ **Client Creates Final Data, Application sends response to Server** – The authenticatorMakeCredential call returns a PublicKeyCredential, which has a rawId that is the globally unique credential id along with a response that is the authenticator’s attestation response containing the clientData and the attestationObject. The PublicKeyCredential is sent back to the server using any desired formatting and protocol.
- ⑥ **Server Validates and Finalizes Registration** - Finally, the server is required to perform a series of checks to ensure that the registration was complete and not tampered with. These include:
  1. Verifying that the challenge is the same as the challenge that was sent
  2. Ensuring that the origin was the origin expected
  3. Validating that the signature over the clientDataHash and the attestation using the certificate chain for that specific model of the authenticator

A complete list of validation steps can be found in the WebAuthn specification [9]. Assuming that the checks pan out, the server will store the new public key associated with the user's account for future use -- that is, whenever the user desires to use the public key for authentication.

### 7.1.6 Example: Healthcare provider user enrolment

For example, a potential use case from healthcare could include a healthcare provider’s online enrolment processes. The process will first attempt to help onboard a new member using customer attribute information and then determine if the attributes presented during enrolment are usable. The strategic goal is to improve the user experience and better identify a member at enrolment time in combination with other internal authentication processes.

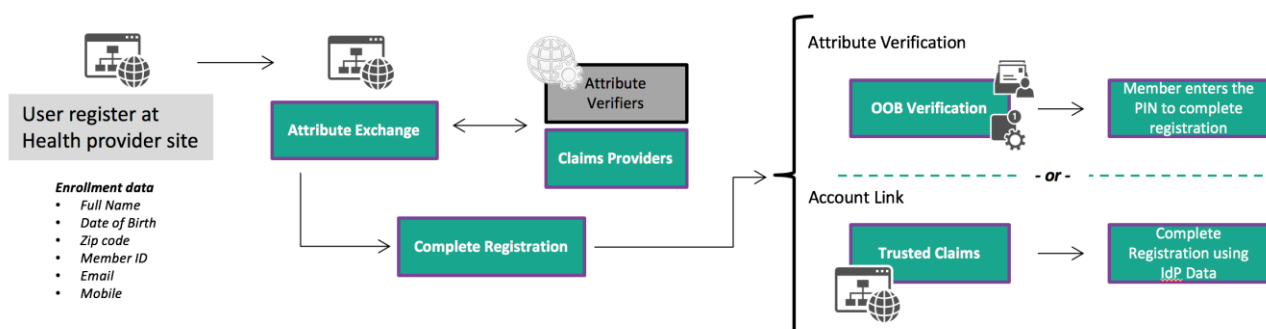


Figure 32 – Healthcare provider user enrolment

### Option 1: Federated Account Linking

During online enrolment, member is allowed to select an option to perform identity verification via a trusted Identity Provider (e.g. a bank).

Member is redirected to bank (IdP or Attribute Provider)) via federation standards. Member authenticates to the IdP. Healthcare provider obtains user information from IdP to compare to initially collected user data.

Member is allowed to complete enrolment with the healthcare provider.

### **Option 2: Attribute Verification**

Instead of the IdP providing attributes to the healthcare provider for consumption and evaluation after authenticating the user, the healthcare provider sends attributes collected in enrolment to the IdP (with user consent).

IdP evaluates, and provides a response indicating the quality or accuracy of the attributes collected during enrolment. The healthcare provider completes member enrolment using OOB verification techniques.

## **7.2 Use case: Authentication to access a digital financial service**

The examples for the Entity authentication use case describe how next generation authentication mechanisms are used to authenticate an individual for authorization to consume services.

### **7.2.1 Example: IFAA use case – Alipay fingerprint/face payment**

Alipay is the most popular mobile payment application in China. It supports fingerprint or face authentication when a user wants to transfer money through mobile devices.

The Alipay payment authentication process adopts the local model of IFAA and is based on the IFAA authentication protocol as illustrated in Figure 16.

Figure 34 is the snapshot from Alipay.

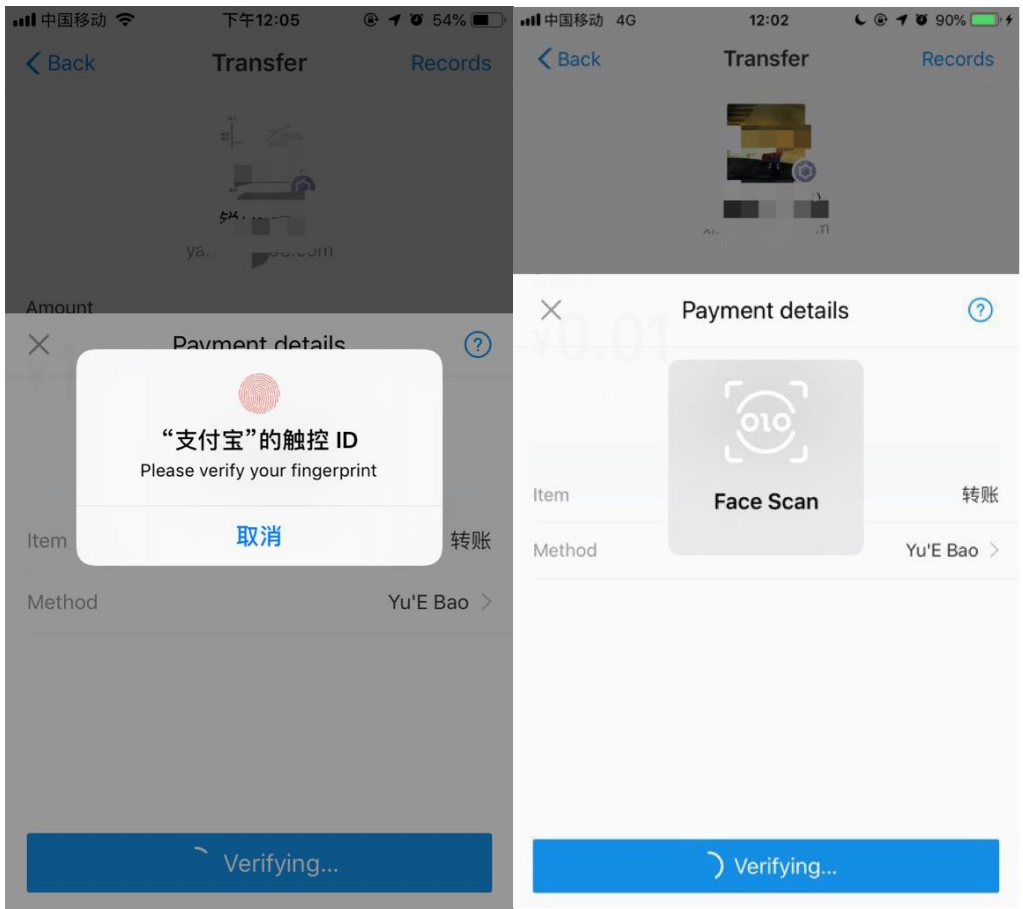


Figure 33 – IFAA use case: Alipay fingerprint/face payment

Figure 35 is the technical framework of Alipay payment authentication system.

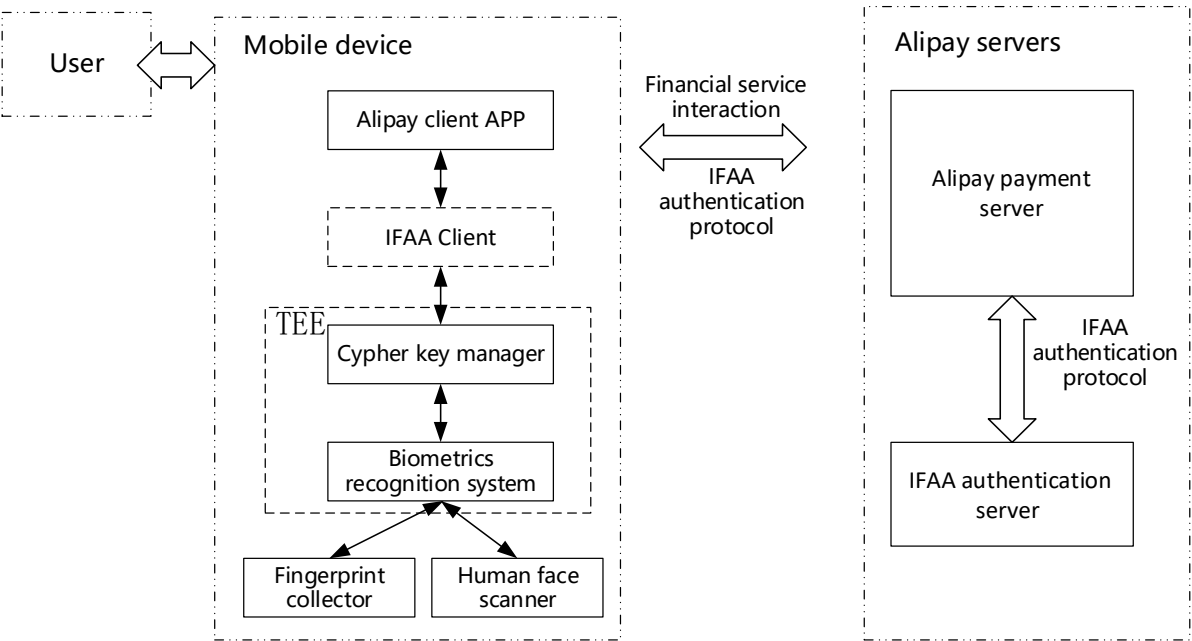


Figure 34 – IFAA use case: Alipay fingerprint/face payment – Technical framework

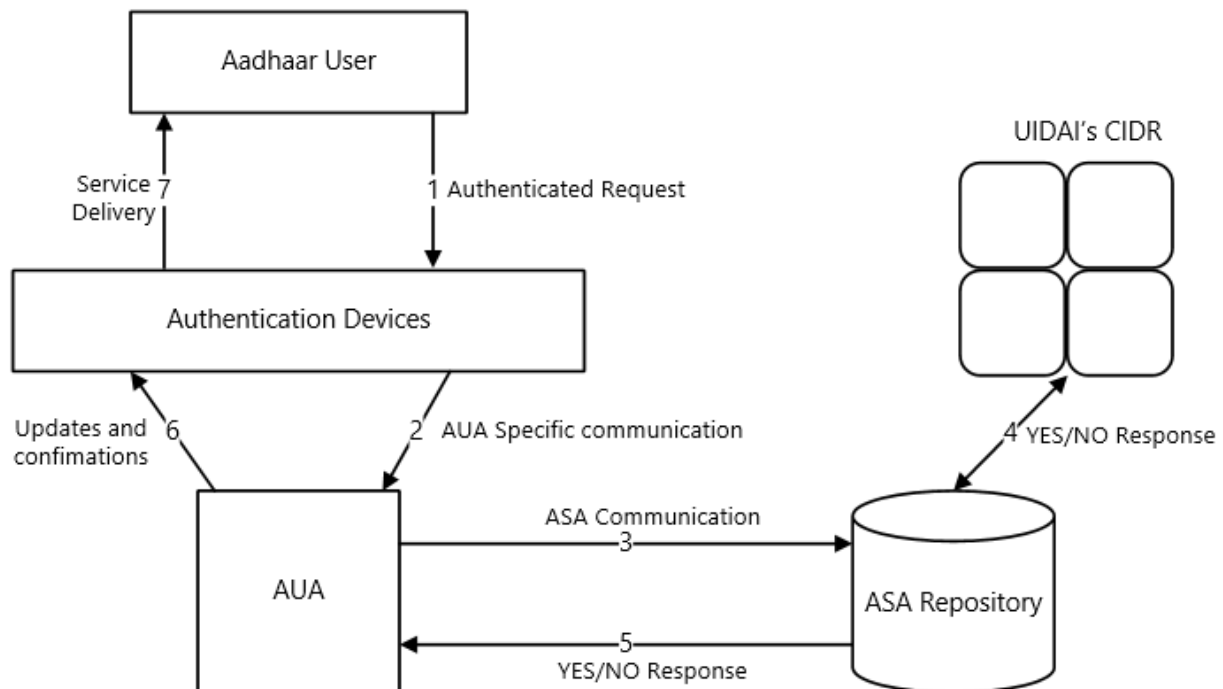
The Alipay user first initiates the registration request through the Alipay client app and runs the registration process as in Figure 15. After a successful registration, the user can initiate a payment request through the Alipay client app.

To begin a payment process, the Alipay client app first interacts with the Alipay payment server to confirm whether mobile payment can be carried out. If yes, the Alipay client app calls the key manager (or optionally, the IFAA client) to authenticate the user as in the following:

- 1) Require the user to perform fingerprint/face authentication based on the local fingerprint/face template.
- 2) After fingerprint/face verification, the key manager invokes the local stored user authentication private key to sign the transaction information, and sends it to the Alipay payment server through the Alipay client app.
- 3) Alipay payment server sends the authentication information to the IFAA authentication server for verification and retrieves the verification results.
- 4) Alipay payment server authorizes the payment after successful verification.

### 7.2.2 Example: Aadhaar authentication

Aadhaar authentication is the process wherein the Aadhaar Number, along with other attributes, including biometrics, are submitted online to the Central Identities Data Repository (CIDR) for its verification on the basis of information or data or documents available with it. During the authentication transaction, the resident's record is first selected using the Aadhaar Number and then the demographic/biometric inputs are matched with the stored data which was provided by the resident during enrolment/update process. Alternatively, authentication can also be carried out on the basis of the OTP. All biometric/OTP authentication schemes are valid for e-KYC service too.



**Figure 35 – Technical process of Authentication & e-KYC services**

The following are the major steps in the Aadhaar authentication process as shown in Figure 35 above:

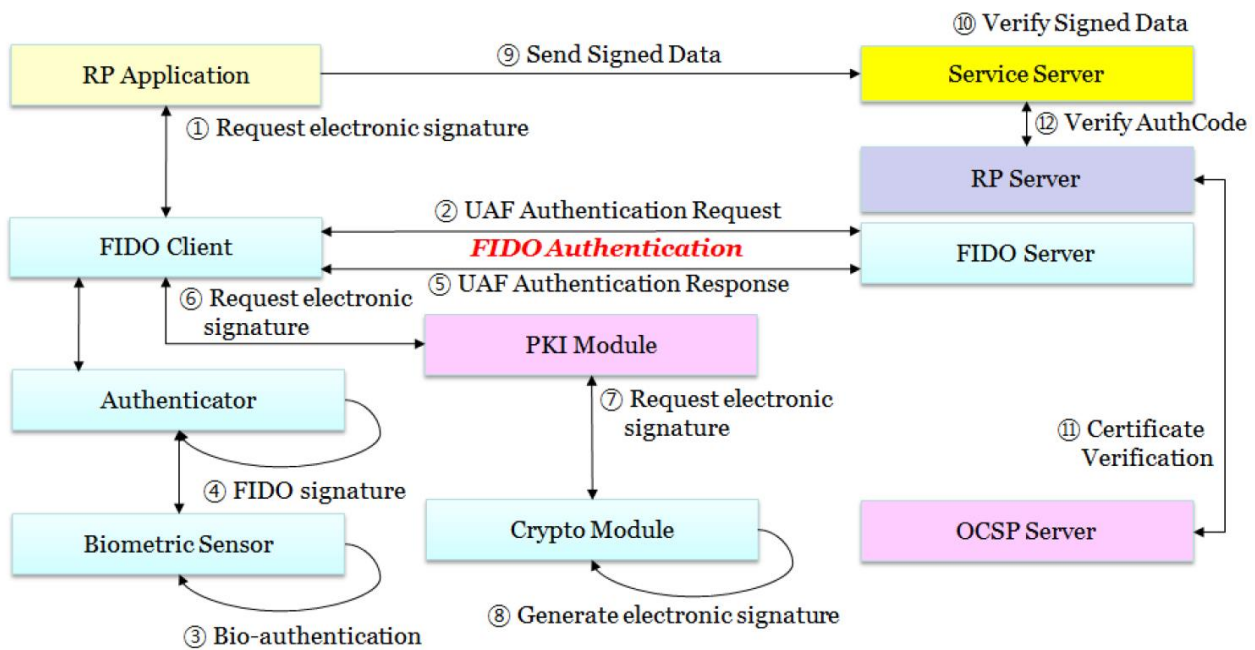
- Aadhaar holder sends the authentication request through the devices
- Aadhaar authentication enabled application software which is installed on the device, encrypts and sends the data to AUA server
- AUA server, after validation, adds necessary headers (AUA specific wrapper XML with license key, signature, etc.), and passes the request through ASA server to UIDAI CIDR.
- Aadhaar authentication server returns a “yes/no” based on the match of the input parameters.
- Based on the response from the Aadhaar authentication server, AUA/Sub-AUA conducts the transaction and Aadhaar holder receives the service.

Additional Security features for Authentication/KYC service:

- To further enhance the security of Aadhaar authentication eco-system, under Regulations 14(n) and 19(o) of Aadhaar (Authentication) Regulations, 2016, it is mandatory to use Hardware Security Module (HSM) for digital signing of Authorised XML and decryption of e-KYC data.
- For digital signing of Authorised XML, Authentication request is digitally signed by the requesting entity (AUA/ KUA) and/or by the ASA using HSM, as per the mutual agreement between them. However, to decrypt the e-KYC response data received from UIDAI, the KUA shall necessarily use its own HSM.
- The HSM to be used for signing Auth XML as well as for e-KYC decryption is FIPS 140-2 compliant.
- All AUA/ KUA/ASA ensure the implementation of HSM in Aadhaar authentication services.
- To eliminate the use of stored biometrics, UIDAI has mandated the use of registered devices by AUA/KUAs and ASAs. The registered devices provide the following key additional features compared to public devices:
- Device identification – every device is required to have a unique identifier allowing traceability, analytics, and fraud management.
- Eliminating use of stored biometrics – biometric data is signed within the device using the provider key to ensure it is indeed captured live. Then the Registered Device (RD) Service of the device provider must form the encrypted PID block before returning to the host application.

### **7.2.3 Example: K-FIDO authentication**

Various user authentication methods used for user authentication for web portals, e-transactions, financial institutions and e-government services are typically supported. Figure 36 illustrates K-FIDO authentication.



**Figure 36 – Authentication Process of K-FIDO Service**

- ① RP App performs bio-authentication and requests electronic signature for a service provider.
- ② FIDO server triggers UAF authentication request to FIDO client.
- ③ A User performs a bio-authentication by the FIDO authenticator using the same method as at registration time.
- ④ The FIDO authenticator generates FIDO signature (using the FIDO authentication private key).
- ⑤ The FIDO client sends UAF authentication response to FIDO server. The FIDO server checks FIDO authentication message and if passed, the RP server generates an Authcode.
- ⑥ The FIDO client requests electronic signature generation to PKI module.
- ⑦ The PKI module requests electronic signature generation to Crypto module.
- ⑧ In case of secure element such as Trustzone, or USIM, the electronic signature will be generated by the private key inside the secure element. However, in case of keystore or keychain, the encrypted private key should be decrypted by a decryption key stored in keystore or keychain and electronic signature will be generated by the private key with crypto module.
- ⑨ RP App sends the signed data to Service server.
- ⑩ Service server verifies the signed data.
- ⑪ Service server or RP Server checks user certificate's verification from OSCP server.
- ⑫ Service server checks the Authcode from FIDO service provider. And Service server sends the result to the user.

#### **7.2.4 Example: Healthcare provider customer authentication**

A large healthcare provider is now in a multi-year process of rolling out its next-generation authentication (NGA) platform across mobile and web applications. With NGA, the healthcare provider is forging new industry best practices for improving healthcare access through a two-

pronged approach to strong authentication. First, they have adopted passwordless FIDO Authentication with biometrics for their customers' online account credentials, reducing their reliance on highly vulnerable "shared secrets," like passwords and one-time-passcodes with strong, unphishable, public key cryptography.

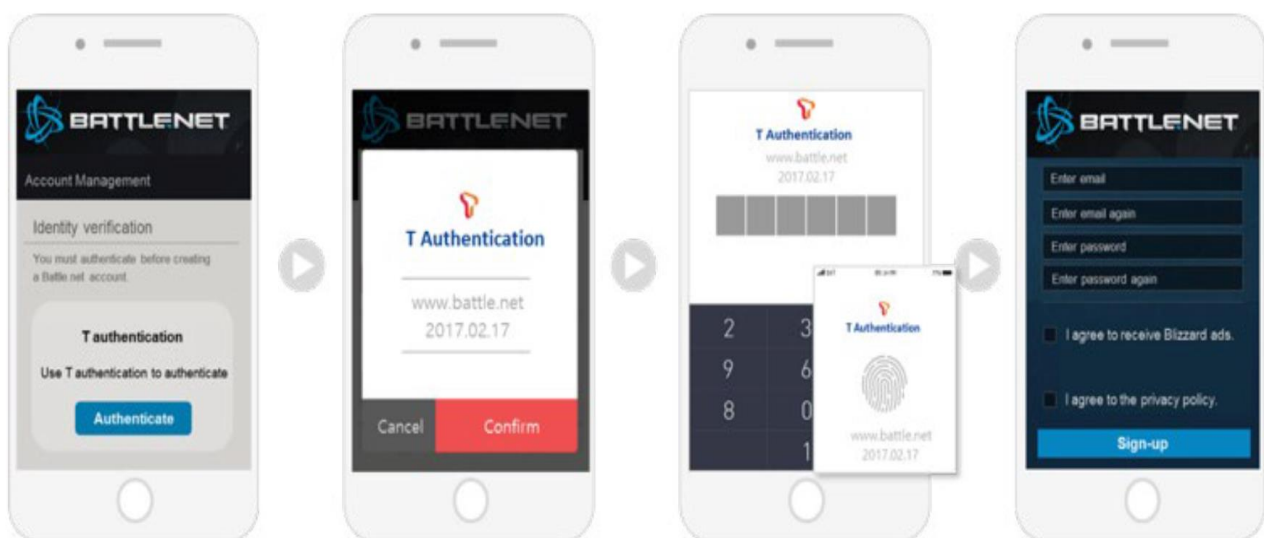
While deploying standards-based strong authentication like FIDO helps resolve many of the authentication problems organizations have faced around security and user experience, healthcare providers still have to contend with risks associated with lost and stolen devices. Thus, the healthcare provider is rolling out the second core component of the NGA platform — continuous, behavior-based authentication — to ensure that the authenticated user is the same person throughout the lifetime of the session. To do this, the healthcare provider looks at several user attributes (such as the way they hold their phone) and assigns risk scores to determine how much access to give a user during a session. If high risk is detected during a session, the healthcare provider may challenge the user for additional information before allowing continued access from that device.

### 7.2.5 Example: SK Telecom – Mobile Connect

SK Telecom is the largest mobile operator in South Korea serving 28 million of the country's 57 million subscribers. SK Telecom has been a pioneer in harnessing the potential of identity services. As early as 2005, it started offering T-Auth, its own mobile identity solution supporting a combination of mobile authentication and attribute matching.

SKT saw an opportunity in Korea's regulations, which require content providers to actively ensure that their customers are authorised to access particular content. Effectively, this means that content providers are responsible for checking that customers wishing to purchase content are over the legal age. SKT realised that its customer account information could help service providers meet this requirement. It designed T-Auth to address this use case with minimal impact on the user experience.

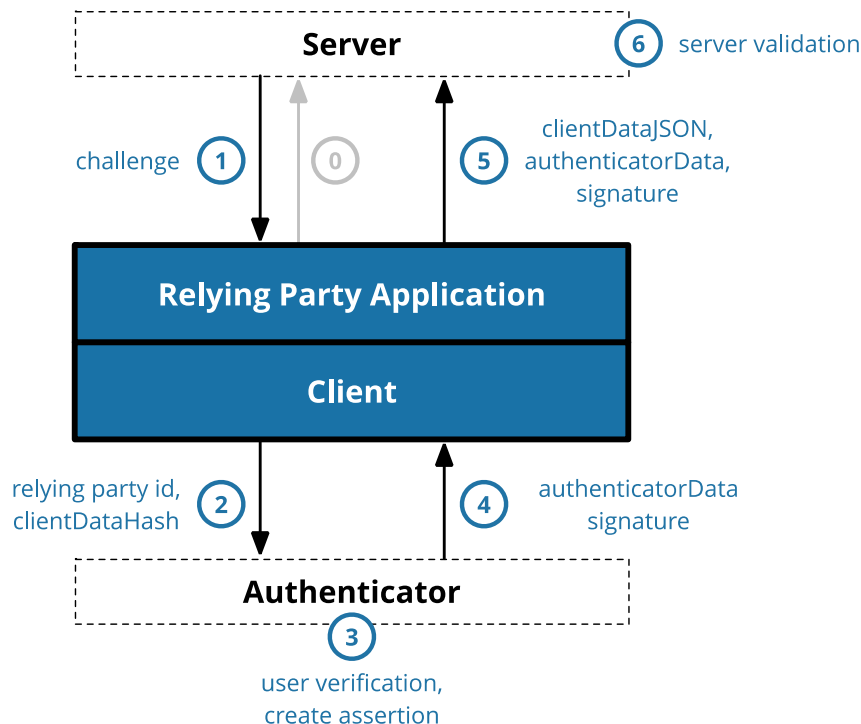
SKT has designed the user journey to minimize user friction during authentication. Figure 37 shows a typical authentication flow: the customer attempts to access a service provider application and is redirected to the T-Auth mobile app for authentication; the customer enters their PIN and biometric sample; on successful authentication T-Auth sends the authentication result and attribute data to the service provider.



**Figure 37 – User Journey to Authenticate to a Gaming Account Using T-Auth**

In early 2017, SKT became compliant with Mobile Connect, the global mobile operator authentication, authorisation and identity framework. As a result, T-Auth is now interoperable with other mobile authentication and identity solutions provided by operators outside of Korea.

### 7.2.6 Example: FIDO Authentication



**Figure 38 – Authentication process of FIDO**

- ① **Application Requests Authentication** - The application makes the initial authentication request. The protocol and format of this request is outside of the scope of FIDO.
- ① **Server Sends Challenge** - The server sends a challenge to the application. The protocol for communicating with the server is not specified and is outside of the scope of FIDO. Typically, server communications would be REST over TLS, but they could also be SOAP, RFC 2549 or nearly any other protocol provided that the protocol is secure. The parameters received from the server will be passed to the credentialGet call, typically with little or no modification.
- ② **Client Calls authenticatorGetCredential on Authenticator via CTAP** - Internally, the client will validate the parameters and fill in any defaults, which become the clientData. One of the most important parameters is the Relying Party ID, which is recorded as part of the clientData so that the Relying Party ID can be verified by the server later. The parameters to the credentialGet call are passed to the authenticator, along with a SHA-256 hash of the clientData (only a hash is sent because the link to the authenticator may be a low-bandwidth NFC or Bluetooth link and the authenticator is just going to sign over the hash to ensure that it isn't tampered with).
- ③ **Authenticator Creates an Assertion** - The authenticator finds a credential for this service that matches the Relying Party ID and prompts a user to consent to the authentication. Assuming both of those steps are successful, the authenticator will create a new assertion by signing over the clientDataHash and authenticatorData with the private key generated for this account during the registration call.



- ④ **Authenticator Returns Data to Client** - The authenticator returns the authenticatorData and assertion signature back to the client.
- ⑤ **Client Creates Final Data, Application sends response to Server** - The authenticatorGetCredential call returns a PublicKeyCredential with the authenticator's assertion response. It is up to the application to transmit this data back to the server using any protocol and format of its choice.
- ⑥ **Server Validates and Finalizes Authentication** - Upon receiving the result of the authentication request, the server performs validation of the response such as:
  - 1. Using the public key that was stored during the registration request to validate the signature by the authenticator.
  - 2. Ensuring that the challenge that was signed by the authenticator matches the challenge that was generated by the server.
  - 3. Checking that the Relying Party ID is the one expected for this service.

A full list of the steps for validating an assertion can be found in the WebAuthn specification [9]. Assuming the validation is successful, the server will note that the user is now authenticated (e.g. – set a flag for the session, set a cookie, etc.).

## 8 Conclusion

This report provides information about two key functions supporting DFS: initial identification of customers at enrolment and strong authentication of customers returning to access DFS services. Both functions are essential to meet regulatory requirements, de-risk service provision and to improve customer experiences.

The implementation examples describe many different system and technology approaches, each dealing with a different set of constraints and environments. The examples build on either centralized authorities and systems or distributed and decentralized systems depending on the capabilities of the constituency, available technologies and societal norms. There is no single optimum solution for all environments. However, implementers can benefit from the experience of the global standards communities by ensuring that their DFS implementations use open international technology standards.

The Financial Action Task Force (FATF) recommends a risk based approach for regulated entities which includes understanding the digital ID systems level/s of assurance for identity proofing and authentication and that these assurance levels are appropriate for consumer due diligence. Strong consumer authentication is based on two or more factors of authentication hence a high level of assurance which is consistent with the FATF recommendation.

As new technologies and approaches appear, standardization becomes critical, both to ensure that existing threats and risks are addressed, and that the new technologies are compatible and fit into existing architectures.

## Annex A – Bibliography

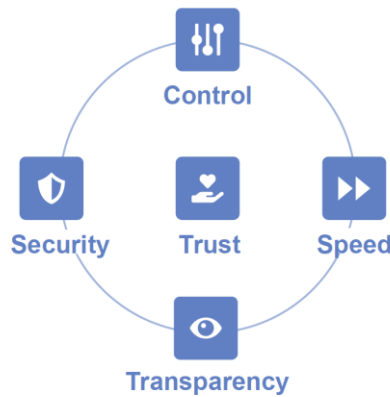
- [1] ITU-T, "ITU-T FG-DFS Main Recommendations," March 2017. [Online]. Available: [https://www.itu.int/en/ITU-T/focusgroups/dfs/Documents/201703/ITU\\_FGDFS\\_Main-Recommendations.pdf](https://www.itu.int/en/ITU-T/focusgroups/dfs/Documents/201703/ITU_FGDFS_Main-Recommendations.pdf).
- [2] ITU-T, "The Digital Financial Services Ecosystem," May 2016. [Online]. Available: [https://www.itu.int/en/ITU-T/focusgroups/dfs/Documents/09\\_2016/FINAL%20ENDORSED%20ITU%20DFS%20Introduction%20Ecosystem%2028%20April%202016\\_formatted%20AM.pdf](https://www.itu.int/en/ITU-T/focusgroups/dfs/Documents/09_2016/FINAL%20ENDORSED%20ITU%20DFS%20Introduction%20Ecosystem%2028%20April%202016_formatted%20AM.pdf).
- [3] FATF, "Digital ID guidance," 20 11 2019. [Online]. Available: <http://www.fatf-gafi.org/publications/fatfrecommendations/documents/consultation-digital-id-guidance.html>.
- [4] ITU-T, "Recommendation X.1254 Entity Authentication Framework," 07 September 2012. [Online]. Available: <http://handle.itu.int/11.1002/1000/11608>.
- [5] "ISO/IEC 29115:2013: Information technology — Security techniques — Entity authentication assurance framework," [Online]. Available: <https://www.iso.org/standard/45138.html>.
- [6] National Institute of Standards and Technology, "Special Publication 800-63B," National Institute of Standards and Technology, Washington, DC, 2017.
- [7] "ID2020 Alliance - Committed to improving lives through digital identity," 2017. [Online]. Available: [https://id2020.org/s/ID2020AllianceDoc\\_UPDATED.pdf](https://id2020.org/s/ID2020AllianceDoc_UPDATED.pdf). [Accessed 27 11 2018].
- [8] ID2020 Alliance, "ID2020 Certification Mark Application form," 2019. [Online]. Available: <https://id2020.org/technical-certification-mark>. [Accessed 27 05 2019].
- [9] D. Balfanz, A. Czeskis, J. Hodges, J. Jones, M. B. Jones, A. Kumar, A. Liao, R. Lindemann, E. Lundberg, V. Bharadwaj, A. Birgisson and H. Le Van Gong, "Web Authentication: An API for accessing Public Key Credentials Level 1," 20 September 2018. [Online]. Available: <https://w3c.github.io/webauthn/>. [Accessed 21 September 2018].
- [10] D. Reed, M. Sporny, D. Longley, C. Allen, R. Grant and M. Sabadello, "Decentralized Identifiers (DIDs) v0.11," 23 August 2018. [Online]. Available: <https://w3c-ccg.github.io/did-spec/>. [Accessed 20 September 2018].
- [11] C. Allen, "The Path to Self-Sovereign Identity," 25 04 2016. [Online]. Available: <http://www.lifewithalacrity.com/2016/04/the-path-to-self-sovereign-identity.html>. [Accessed 27 03 2019].
- [12] M. Sporny and D. Longley, "Verifiable Credentials Data Model 1.0," 18 September 2018. [Online]. Available: <https://www.w3.org/TR/verifiable-claims-data-model/>. [Accessed 18 September 2018].
- [13] Sovrin Foundation, "Sovrin Glossary v2," 27 03 2019. [Online]. Available: [https://docs.google.com/document/d/1gflz5TT0cNp2kxGMLFXr19x1uoZsruUe\\_0glHst2fZ8/edit?usp=sharing](https://docs.google.com/document/d/1gflz5TT0cNp2kxGMLFXr19x1uoZsruUe_0glHst2fZ8/edit?usp=sharing). [Accessed 27 05 2019].
- [14] M. Sabadello, K. Den Hartog, C. Lundkvist, C. Franz, A. Elias, A. Hughes, J. Jordan and D. Zagidulin, "Introduction to DID Auth," 26 07 2018. [Online]. Available:

- <https://github.com/WebOfTrustInfo/rebooting-the-web-of-trust-spring2018/blob/master/final-documents/did-auth.pdf>. [Accessed 15 03 2019].
- [15] M. Sabadello, "A Universal Resolver for self-sovereign identifiers," 01 11 2017. [Online]. Available: <https://medium.com/decentralized-identity/a-universal-resolver-for-self-sovereign-identifiers-48e6b4a5cc3c>. [Accessed 15 03 2019].
  - [16] D. Reed, J. Law, D. Hardman and M. Lodder, "DKMS (Decentralized Key Management System) Design and Architecture," 02 04 2018. [Online]. Available: <https://github.com/hyperledger/indy-sdk/blob/677a0439487a1b7ce64c2e62671ed3e0079cc11f/doc/design/005-dkms/DKMS%20Design%20and%20Architecture%20V3.md>. [Accessed 15 03 2019].
  - [17] A. Offerman, "Swiss City of Zug issues Ethereum blockchain-based eIDs," February 2018. [Online]. Available: <https://joinup.ec.europa.eu/node/700433>. [Accessed September 2018].
  - [18] Open Banking Limited, "Customer Experience Guidelines," September 2018. [Online]. Available: <https://www.openbanking.org.uk/providers/standards/>. [Accessed 09 2018].
  - [19] ITU-T, "ITU-T FG-DFS Executive summary," March 2017. [Online]. Available: [https://www.itu.int/en/ITU-T/focusgroups/dfs/Documents/201703/ITU\\_FGDFS\\_Executive-summary.pdf](https://www.itu.int/en/ITU-T/focusgroups/dfs/Documents/201703/ITU_FGDFS_Executive-summary.pdf).
  - [20] FIDO Alliance, "Frequently Asked Questions," [Online]. Available: <https://fidoalliance.org/faqs>. [Accessed 15 July 2018].
  - [21] FIDO Alliance, "Client To Authenticator Protocol v2.0," 27 September 2017. [Online]. Available: <https://fidoalliance.org/specs/fido-v2.0-ps-20170927/fido-client-to-authenticator-protocol-v2.0-ps-20170927.html>. [Accessed 15 July 2018].
  - [22] FIDO Alliance, "Universal Authentication Framework," 02 February 2017. [Online]. Available: <https://fidoalliance.org/specs/fido-uaf-v1.1-ps-20170202/fido-uaf-protocol-v1.1-ps-20170202.html>. [Accessed 15 July 2018].
  - [23] FIDO Alliance, "Universal Second Factor Overview," 11 April 2017. [Online]. Available: <https://fidoalliance.org/specs/fido-u2f-v1.2-ps-20170411/fido-u2f-overview-v1.2-ps-20170411.html>. [Accessed 15 July 2018].
  - [24] S. N. Yanushkevich, "ENCM 509 Fundamental of Biometric System Design Chapter 1," [Online]. Available: <http://www.ucalgary.ca/btlab/files/btlab/ch1.pdf>. [Accessed 15 July 2018].
  - [25] FIDO Alliance, "Setting the Standard for Healthcare Security: Aetna Rolls out FIDO and Behavioral Authentication.," 19 July 2017. [Online]. Available: <https://fidoalliance.org/aetna-deploys-fido-authentication>. [Accessed 11 04 2018].
  - [26] OpenID Foundation, "OpenID Foundation Specifications," [Online]. Available: <http://openid.net/developers/specs/>. [Accessed 07 August 2018].
  - [27] OpenID Foundation, "OpenID Connect FAQ and Q&As," [Online]. Available: <http://openid.net/connect/faq/>. [Accessed 07 August 2018].
  - [28] FIDO Alliance, "FIDO Deployment Case Study K-FIDO," September 2017. [Online]. Available: [https://fidoalliance.org/wp-content/uploads/FIDO-Deployment-Case-Study-K-FIDO\\_170905.pdf](https://fidoalliance.org/wp-content/uploads/FIDO-Deployment-Case-Study-K-FIDO_170905.pdf). [Accessed September 2018].

- [29] "Press Release," 14 09 2018. [Online]. Available: [https://www.prnewswire.com/news-releases/id2020-alliance-launches-inaugural-pilots-welcomes-new-partners-at-annual-summit-300713089.html?tc=eml\\_cleartime](https://www.prnewswire.com/news-releases/id2020-alliance-launches-inaugural-pilots-welcomes-new-partners-at-annual-summit-300713089.html?tc=eml_cleartime). [Accessed 27 11 2018].
- [30] "Press Release," 14 09 2018. [Online]. Available: <http://www.globenewswire.com/news-release/2018/09/14/1571269/0/en/Everest-ID2020-and-the-Government-of-Indonesia-TNP2K-Secretariat-Announce-Innovative-Identity-and-Blockchain-Pilot-Solution-to-Enhance-the-National-LPG-Subsidy-Program.html>. [Accessed 27 11 2018].
- [31] GSMA, "Mobile Connect - Operator cooperation in South Korea has created a successful identity solution," 2017. [Online]. Available: [https://www.gsma.com/identity/wp-content/uploads/.../mc\\_skt\\_case\\_08\\_17\\_v.1b.pdf](https://www.gsma.com/identity/wp-content/uploads/.../mc_skt_case_08_17_v.1b.pdf). [Accessed 13 09 2018].
- [32] FIDO Alliance, "FIDO & PSD2: Meeting the needs for Strong Customer Authentication," 2017. [Online]. Available: <https://fidoalliance.org/wp-content/uploads/FIDO-PSD2-white-paper-FINAL.pdf>. [Accessed 20 09 2018].
- [33] European Banking Authority, "Final Report on Draft Regulatory Technical Standards on Strong Customer Authentication and common and secure communication under Article 98 of Directive 2015/2366 (PSD2)," 2017. [Online]. Available: <https://eba.europa.eu/documents/10180/1761863/Final+draft+RTS+on+SCA+and+CSC+under+PSD2+%28EBA-RTS-2017-02%29.pdf>. [Accessed 12 09 2018].
- [34] Kiva, "Kiva, Sierra Leone and U.N. agencies partner to implement 'credit bureau of the future'," 2018. [Online]. Available: <https://www.kiva.org/blog/kiva-sierra-leone-and-un-agencies-partner-to-implement-credit-bureau-of-the-future>. [Accessed 27 05 2019].
- [35] Kiva, "Kiva Protocol FAQ," 2019. [Online]. Available: <https://pages.kiva.org/kiva-protocol-faq>. [Accessed 27 05 2019].

## Annex B – Guidance for DFS Providers

DFS Providers engage with consumers and manage most aspects of their user experience. Careful attention to customer journeys and user experience can increase trust in the system and increase operational security. Open Banking has published Customer Experience Guidelines [18] that describe ‘experience principles’ which were developed to assist designers.



**Figure 40 - Open Banking experience principles for user engagement**

- Control: Consumers need to have a sense of control through having the right tools and clear information at the right time.
- Speed: Speed must be appropriate to the specific customer and interaction. Fastest is not always best.
- Transparency: Transparency of choice, action and information about consequences are crucial.
- Security: Fraud and data privacy are top concerns for users of DFS. Messaging and information about security measures must be clear and direct.
- Trust: The combination of the Control, Speed, Transparency and Security principles create a trusted environment for the customer.

## Annex C – Guidance for Authentication System Providers

Authentication standards organizations publish guidance material for authentication system providers to assist with implementation. Guidance material relevant to the standards cited in this report include:

### **NIST Trusted Identities Group:**

<https://www.nist.gov/itl/tig/projects/special-publication-800-63>

#### *Implementation guidance*

- “NIST will work with the community to prepare implementation guidance for the Digital Identity Guidelines. The goal is to give implementers easily deployable guidance and help them meet the requirements.”

### **Fido Alliance:**

<https://fidoalliance.org/white-papers/>

- *FIDO Alliance White Paper: Enterprise Adoption Best Practices – Managing FIDO Credential Lifecycle for Enterprises (April 2018):*  
“This white paper provides guidance to IT and Security professionals on how manage FIDO authentication credentials throughout their full lifecycle.”
- *FIDO Alliance White Paper: How FIDO Standards Meet PSD2’s Regulatory Technical Standards Requirements on Strong Customer Authentication (December 2018):*  
“This document provides a detailed review of the security requirements listed in the Regulatory Technical Standards For Strong Customer Authentication and Common and Secure Open Standards Of Communication under PSD2 (the RTS) and describes how the FIDO standards meet such requirements.”
- *FIDO Alliance White Paper: FIDO & PSD2 – Providing for a Satisfactory Customer Journey (September 2018):*  
“This white paper examines the different authentication models that could apply within the interactions of a Third-Party Provider and an Account Servicing Payment Service Provider. It proposes the FIDO standards as a solution to simplify the user experience, for any of these models, in a way that meets the Strong Customer Authentication requirements of PSD2.”
- *FIDO Alliance White Paper: Enterprise Adoption Best Practices – Integrating FIDO & Federation Protocols (December 2017):*  
“This white paper outlines how the FIDO standards compliment federation protocols. It also provides guidelines on how to integrate the two in order to add support for FIDO-based MFA and replace or supplement traditional authentication methods in federation environments.”

### **OpenBanking:**

<https://www.openbanking.org.uk/providers/standards/>

#### *Customer Experience Guidelines*

- “This document brings together regulatory requirements and extensive customer research to provide customer experience guidelines and examples of customer journeys for third party providers and account providers. They are designed to encourage adoption of Open Banking-enabled products and services.”

## GSMA Mobile Connect:

[https://www.gsma.com/identity/wp-content/uploads/2019/03/mc\\_mwc\\_platforms\\_booklet2\\_web\\_02\\_19-1.pdf](https://www.gsma.com/identity/wp-content/uploads/2019/03/mc_mwc_platforms_booklet2_web_02_19-1.pdf)

*GSMA Platforms & Operations services, February 2019*

- “The GSMA offers a series of technical platforms and services designed to help mobile network operators (MNO) and service providers (SP) deploy Mobile Connect successfully:
- Interoperability Testsuite Portal: Check if your Mobile Connect product complies with the Mobile Connect specification.
- API Exchange: Become part of a Mobile Connect ecosystem with other MNOs to be able to offer seamless cross-operator reach to Service Providers.
- Developer Portal (with Sandbox and SDKs): Comprehensive documentation and tools to facilitate the integration of Mobile Connect into your applications.
- Operator Management Console: Self-service portal to access reports and manage business processes between you and GSMA.
- Service Desk: Single point of contact for all Mobile Connect enquiries.
- Monitoring & Incident Management: Check the health of your Mobile Connect components and the status of any incident affecting these.”

[https://www.gsma.com/identity/wp-content/uploads/2018/02/Mobile-strong-customer-authentication-under-PSD2\\_March2018-FINAL.pdf](https://www.gsma.com/identity/wp-content/uploads/2018/02/Mobile-strong-customer-authentication-under-PSD2_March2018-FINAL.pdf)

*Mobile strong customer authentication under PSD2: comparisons and considerations*

- “This paper looks at major mobile SCA solutions that are generically available in the EU at the time of writing. Whilst there are many proprietary and national solutions that comply effectively with the PSD2 requirements, we have focused on those that are generically available across the European Union. The profiles in this paper are indicative only of high-level considerations that payment service providers will have in mind when implementing mobile SCA. The aim of ASPSPs is to prevent a single point of failure and offer a variety of authentication solutions to their payment service users (PSUs); the solutions selected for consideration in this document, therefore, can be combined and implemented in a complementary way.”

<https://developer.mobileconnect.io/step-by-step-guide>

- “Implementing Mobile Connect involves interacting with a number of different services and technologies. The section shows each of the steps you need to follow and offers guidance on how to complete the steps.”