Annex A

CONNECTING EUROPE FACILITY (CEF) - TELECOMMUNICATIONS SECTOR

AGREEMENT No INEA/CEF/ICT/A2016/1271569

Technical Report

Table of Contents

Table of Figures

Introduction

This appendix to the main report for the Connecting Europe Facility – Opening a bank account across borders – aims to describe the technical solution, the considerations and delivery of the prototype to demonstrate how a digital identity can be used across borders. This technical report is not intended to be a set of instructions on how to build an equivalent solution but is intended to give the reader a good understanding of the components used to achieve the end goal. Other solutions could be built to achieve the same result which are worthy of further consideration.

The context of the technical elements used in the project was to take the current eIDAS framework as the main specification for transporting identities and marry this to existing ID handling technologies. This would minimise the need for specially engineered components and accelerate the delivery. The adherence to recognised standards was paramount in the definition of the solution to achieve a cost effective target solution.

High level architecture

This section details the final architecture that was agreed amongst the consortium and the considerations made. The architecture has seven main components.

- The Prototype website branded as 'AnyBank'
- ID federation hub
- eIDAS component
- France Connect
- Mobile Connect infrastructure
- Attribute exchange solution
- Database structure for attributes

Details of these components are set out in the sections below.

Design considerations

The following considerations were made when deciding on the architectural solution:

1) A web browser user interface was required and a mobile responsive variant to be considered in a next phase.
2) A fictitious bank brand and user interface design would be created for the front end to be based on the current ForgeRock banking demonstration solution. This represents a Relying Party (RP) in the ID consumption construct.
3) The key component in the architecture will be an identity federation hub service. This provides access to Identity providers with in the federation for relying parties.
4) Connectivity to the eIDAS outbound node in France needs to be developed to move a French digital Identity into another jurisdiction's notified scheme. It was decided that this component would be integrated into the federation hub so that it would allow any other RP joining the federation to access the eIDAS service. Additionally the federation hub will have other services attached that could be useful in testing; for example the Idemia hub has a test Gov.UK Verify compliant IDP.
5) France Connect and Orange provided the French digital identities and the mechanism for citizens to authenticate themselves against their digital identity.
6) It was considered that the prototype would not handle live identities and integrate into the live systems of Barclays and HSBC. This decision was taken because of other priorities competing for resources within those banks.
7) The integration test systems of France Connect and Orange will be used to deliver authentications back into the architecture to be passed by the outbound eIDAS node. The integration test systems accurately reflect the systems that are live and handling real identities.

Functional components:

This section gives details about the main building blocks in the architecture to give the reader a view of the components involved their owners and interconnection; the diagram below illustrates this.

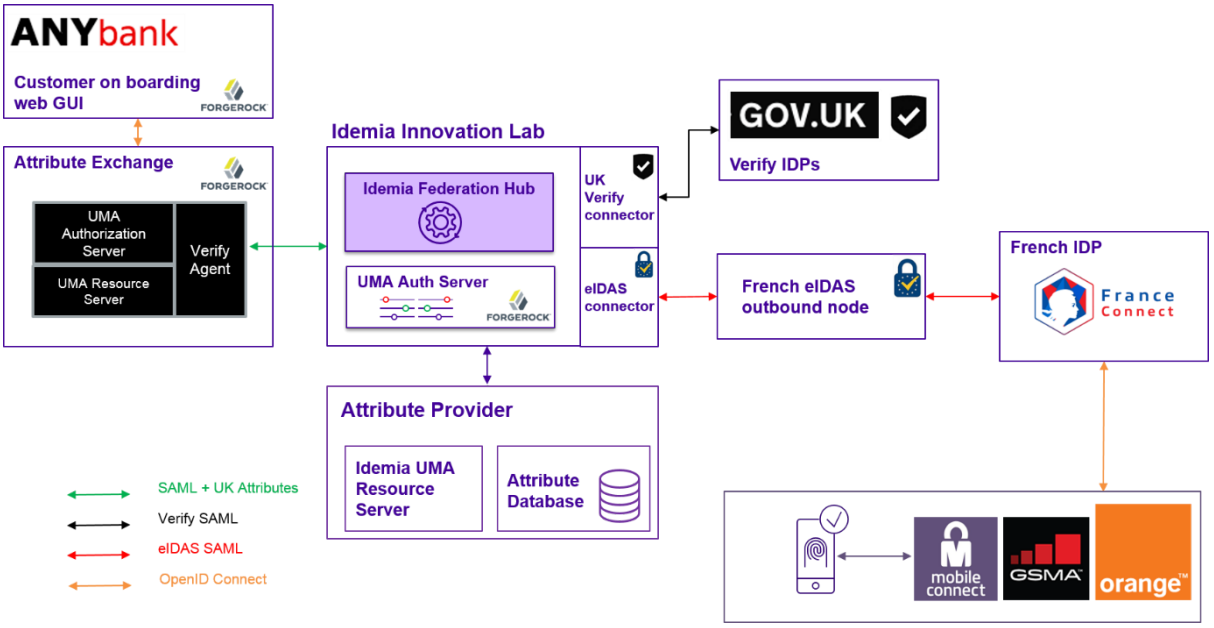


Figure 1 - High Level Architecture

Summary of components

The following sections describe the blocks in the architecture delivered, so that the reader can understand their functionality. Additional details are contained in the Exhibits attached to this Appendix to the main report.

Prototype website - branded as AnyBank.

The AnyBank web front end has been set up to simulate the graphical interface that a citizen from France would end up at whilst searching for a banking services provider in the UK. This frontend was created as an anonymous brand to illustrate how Barclays and HSBC could in future deploy a live service. It is intended that other banks may adopt this web journey but within their own style guides.

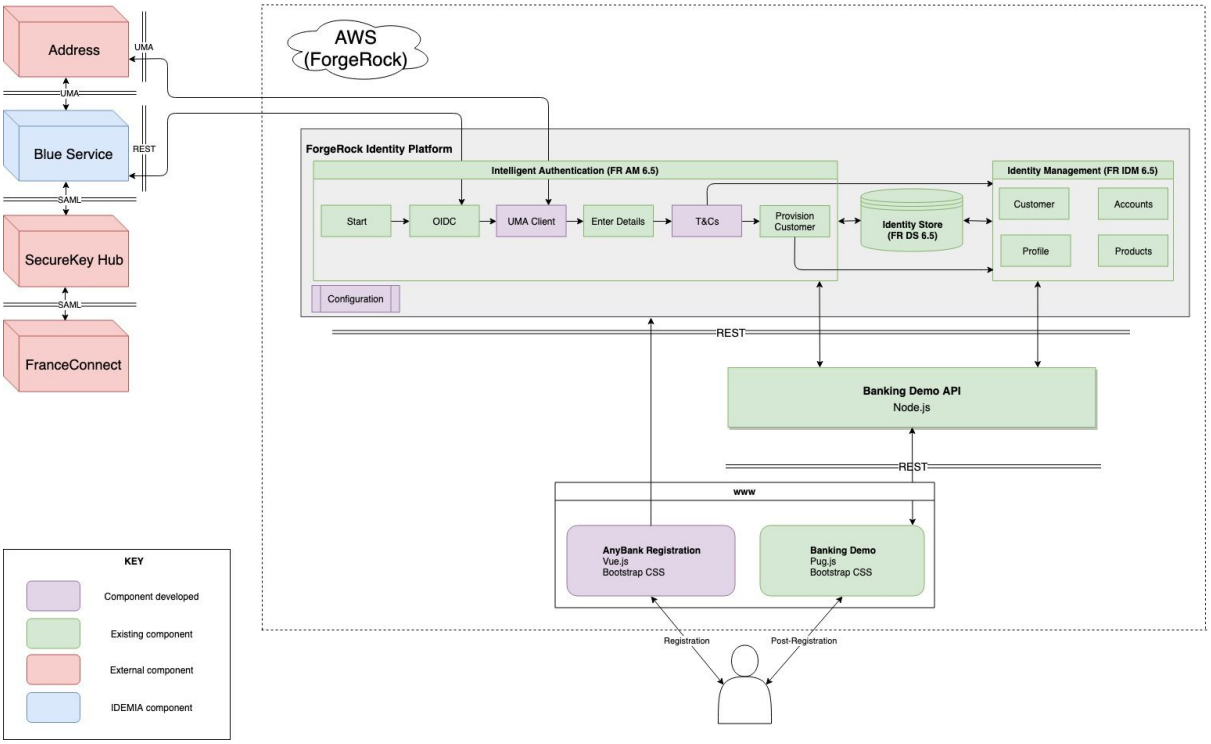
For the delivery of the web solution HSBC engaged ForgeRock to use their standard banking demonstrator to deliver this component. This banking demonstrator has been customised to deliver the delivered prototype user interface within the constraints of what it is capable. The customisation

has been based on the wireframes that were created to illustrate a user journey of an end state solution which is fully integrated into a bank's online presence.

This web front end is supported by ForgeRock's UMA Authorisation Server, UMA resource server and a Verify Agent to support Verify SAML connections to the Federation hub.

AnyBank Architecture

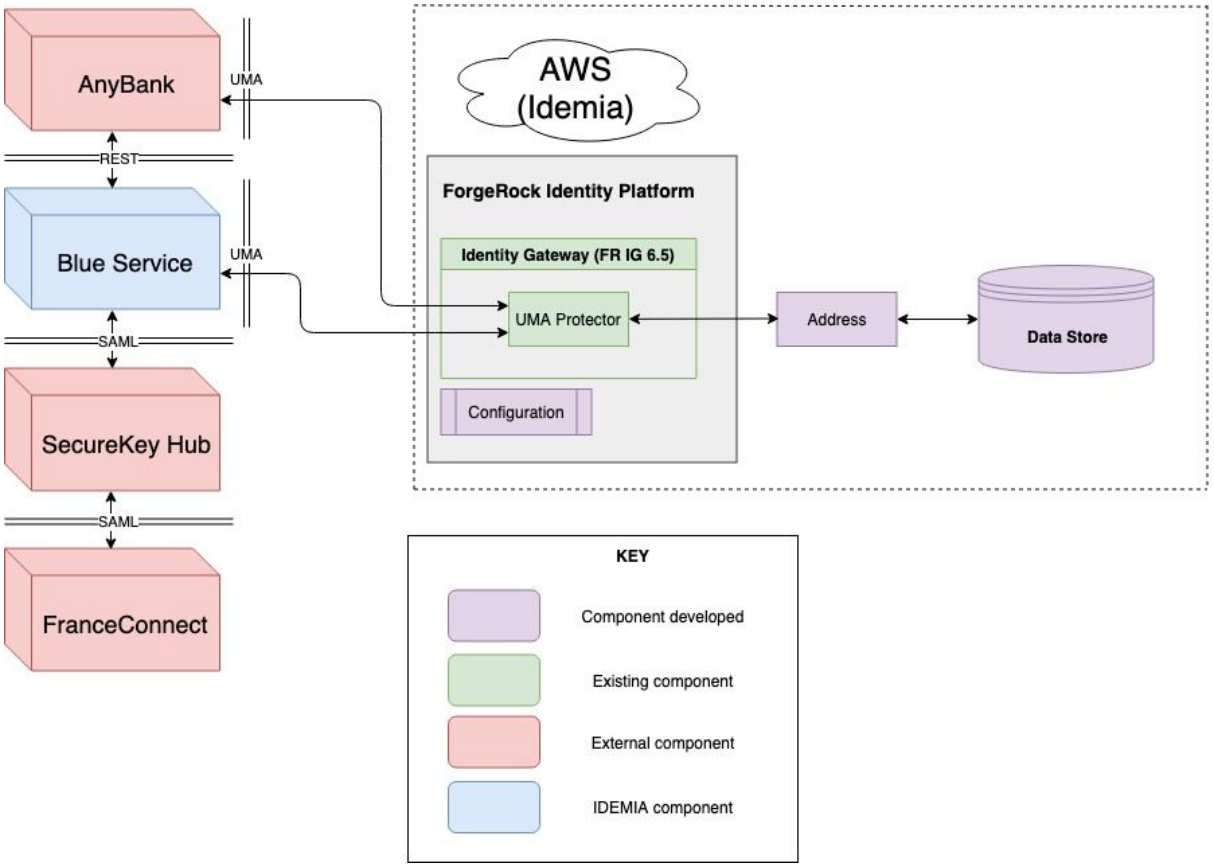
The following diagram shows the AnyBank architecture implemented in this project and its interactions with the other components:



Stub Address Attribute Provider

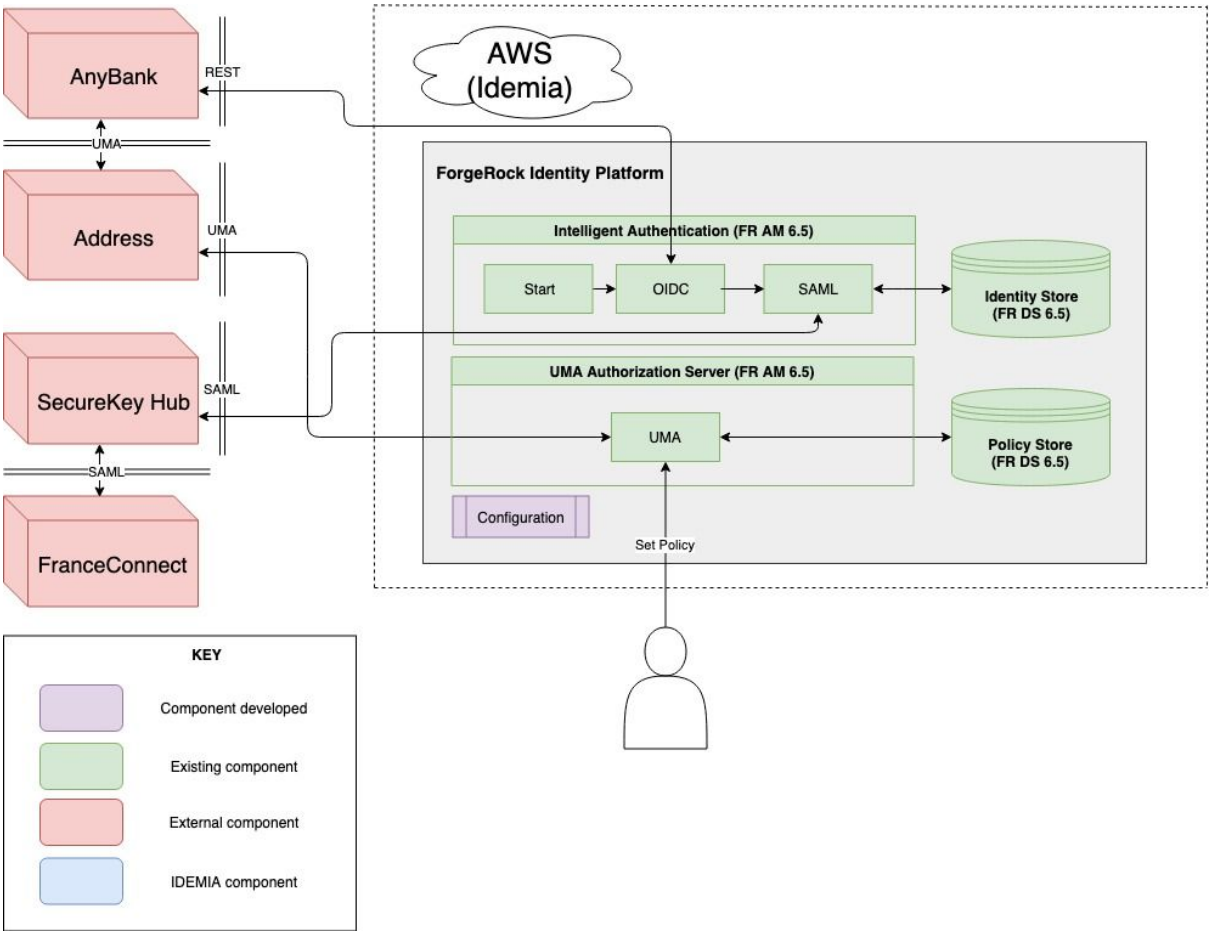
An attribute provider was required for the project. This needed to be an UMA Resource Server that served the address attribute for the identity opening the AnyBank account. A stub service was developed, by IDEMIA, to host the address data, and was fronted by an UMA Resource Server implemented in ForgeRock's Identity Gateway.

The following diagram shows the architecture implemented for the stub address attribute provider and how it interacts with the other components:



Attribute Exchange

The prototype IDEMIA Blue Service, powered by the ForgeRock Identity Platform, was provided as the Attribute Exchange for this project. The following diagram shows the architecture of this hub service and its interactions with the other components:



ID federation hub

The federation hub is the core of the architecture allowing the AnyBank web application to connect to any IDP connected to it. It provides two key actions in the user journey. The first is the routing of the ID authentication request by sending it to the eIDAS agent/connector. This is then passed into the French eIDAS node. The hub then manages the responses, to and from the French outbound node, to the authentication requests.

The second function of the hub is to provide the exchange of the data attributes from the data provider to the Relying Party, AnyBank in this case. This function is dependent on the result of the authentication and would only commence upon the successful authentication of an identity. Further, this function manages the acquisition of the consent of the citizen to have their data released to AnyBank, which is a fundamental requirement to GDPR legislation.

The federation hub has been provided by Idemia who have implemented the SecureKey briidge.net Exchange product. The implementation used for the project is the live Innovation Lab service that has been used by several relying parties to evaluate federation hub services where they wish to test

using UK Government Verify test identities. This SecureKey product has been successfully implemented in Canada as the SecureKey Concierge – Credential Broker Service.

By way of overview, the service allows Canadian citizens to sign in to government services using their online banking credentials. Most citizens can remember their online banking credentials as they use them very frequently, whereas they often only sign into government services once a year meaning that they forget their government logins.

It is not intended to provide a detailed description of the service here but overview information can be found at these web resources:

https://www.canada.ca/en/shared-services/corporate/transparency/access-information-privacy/publications/securekey-concierge-credential-broker-service.html

https://securekeyconcierge.com/resources/securekey-concierge-overview/

High Level Hub internal generic architecture

The diagram below illustrates the main components of the Securekey briidge Exchange solution.

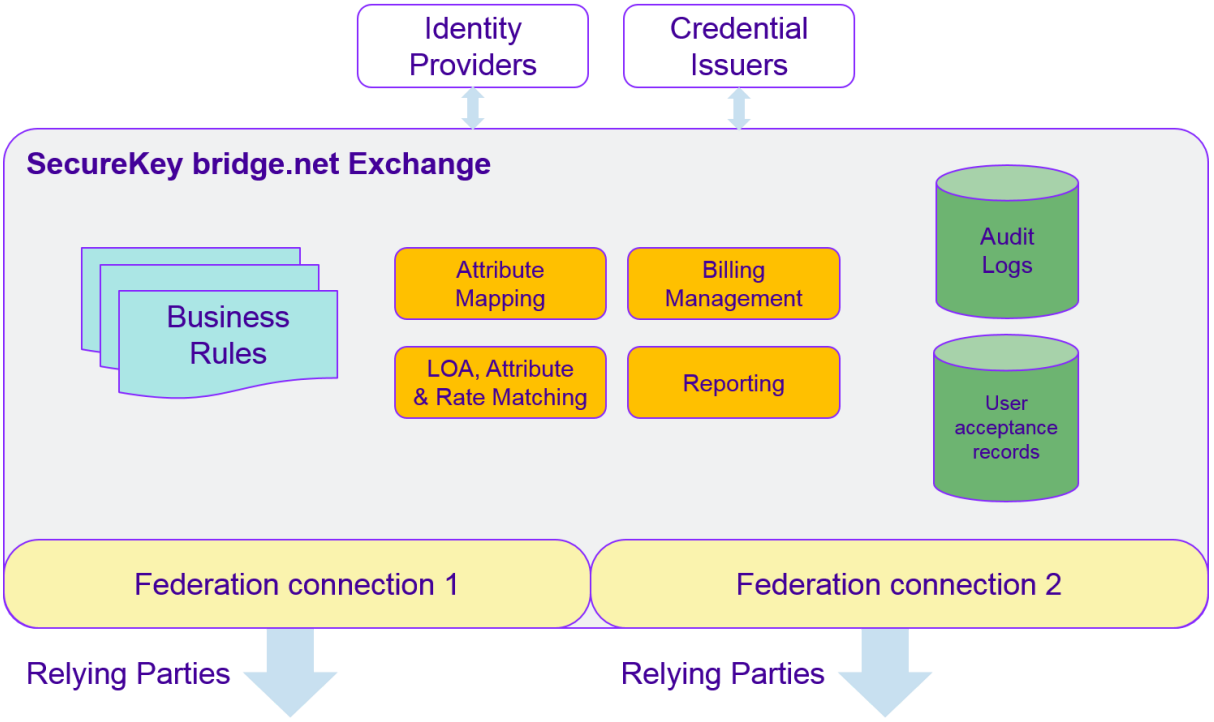


Figure 2 - SecureKey briidge Exchange hub - key components

SecureKey is a leading identity and authentication provider that simplifies consumer access to online services and applications. It provides a secure, privacy-enhancing service that conveniently connects people to critical online services (eGov, Health, Finance) using an approved digital credential they already have and trust.

SecureKey briidge.net exchange core platform delegates authentication to an approved Credential Service Provider chosen by the user. It allows Single Sign-On functionality to allow user access to multiple relying parties without explicitly re-authenticating. Users can transition credentials from one CSP account to another without having to re-enrol to the RP. It consists of various applications and database instances managing auditing, reporting and user records.

Further details of the SecureKey Exchange briidge.net ID federation hub can be found in Exhibits 4 and 5 at the end of this document

Hub architecture

The following diagram illustrates how the hub is constructed in the Exchange briidge.net solution.

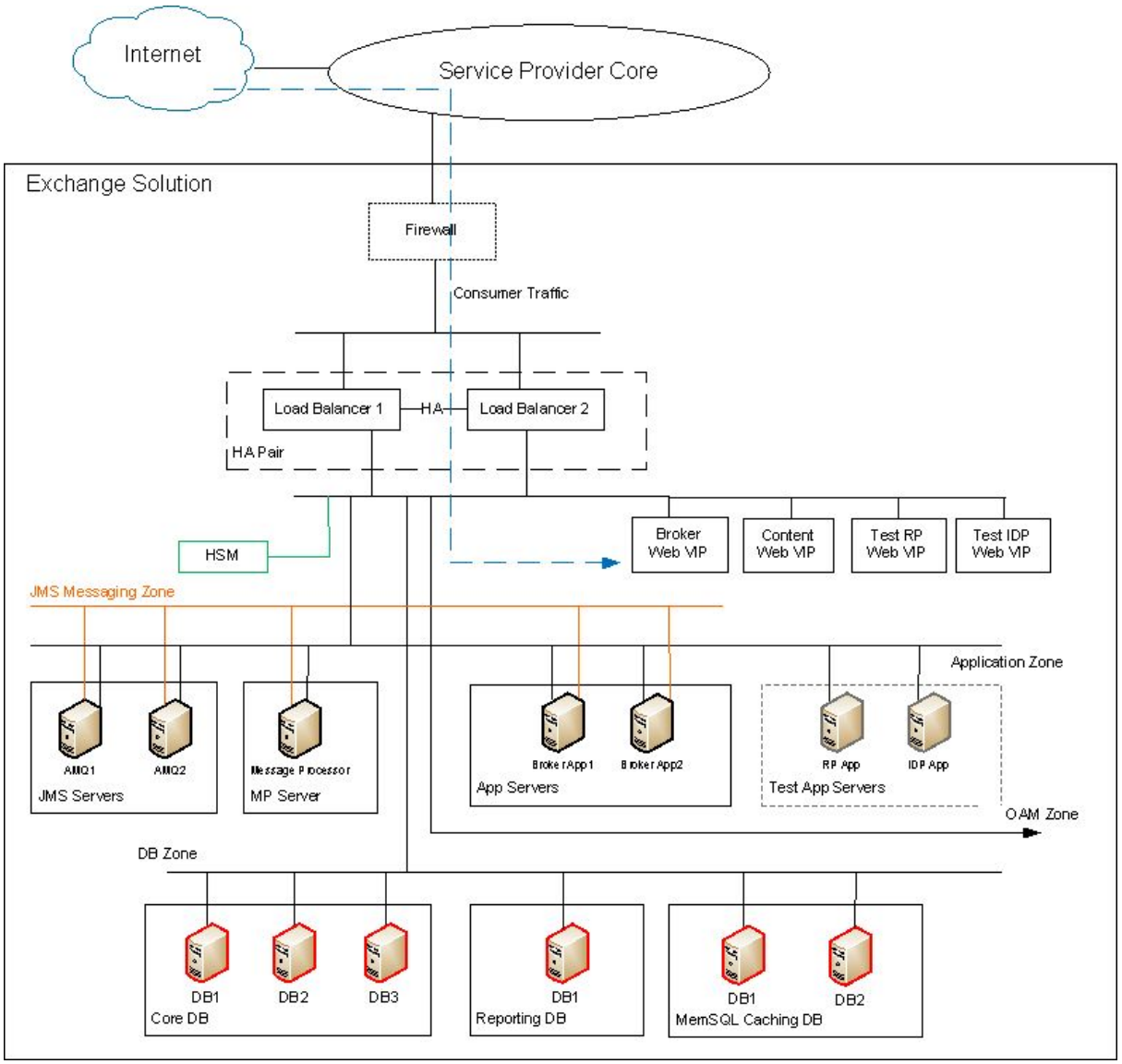


Figure 3 - SecureKey briidge.net architecture

The following sections describe the functions of the components in the ID federation hub.

HSM

An HSM is a physical computing device that safeguards and manages digital keys for strong authentication and provides crypto-processing. SecureKey's Exchange platform is capable of supporting any HSM device that uses a PKCS #11 public-key cryptography standard.

Broker Web VIP

Broker Public URL via https to federated hub, which is resolved with fully qualified domain name i.e. fed1.ukidap.xantav.com and associated port.

Content Web VIP

Static Content URL via https to serve static content, which is resolved with fully qualified domain name i.e. content.ukidap.xantav.com and associated port.

Test RP Web VIP

Test RP URL via https to Service provider unit, which is resolved with fully qualified domain name i.e. testrp.ukidap.xantav.com and associated port.

Test IDP Web VIP

Test CSP URL via https to Identity provider unit, which is resolved with fully qualified domain name i.e. testcsp.ukidap.xantav.com and associated port.

Application Zone

This zone is for Applications where all application instances reside e.g. RP, CSP, Broker, AMQ. The architecture can have more than one instance for single application depending on the usage & purpose.

JMS Server

Apache ActiveMQ Java Message Service (JMS) servers use a standard JEE messaging protocol. Exchange deploys a standard producers (Application Tier) and consumers (Message Processors) model, where each node is a subscriber to a particular queue on an ActiveMQ broker.

MP Server

Message Processor (MP) move all the system and user-related events for Exchange from Java Message Service (JMS) stores into logging and reporting databases for archiving and reporting purposes. Each is a separate, standalone java application that runs under the same Tomcat server instance.

App Servers

The core applications are a set of Java Platform Enterprise Edition (JEE) web and SAML v2 applications called the:

• Broker application

• (Optional for testing) Test RP and Test CSP applications

These applications broker authentication and identity attributes between Token Providers, Identity Providers (CSPs), and RPs.

Database (DB) zone

This zone is used for hosting the database for the application hosted in the App layer for caching, reporting and storage purposes.

Core DB

The Exchange system uses a three-node Percona MySQL cluster for storing persistent system data and is connected to the application along with the Message Processor instances using designated ports.

Reporting DB

Percona MySQL database is used for generating the system reports. Idemia can extract reports for billing, audits and miscellaneous purposes under this particular database.

MemSQL Caching DB

MemSQL is a distributed, in-memory, SQL database management system. SecureKey's Exchange system loads the data in Memsql cache to achieve higher performance and connects with core database for reporting purposes.

Choice of Hub provider

Prior to the inception of this CEF project Idemia had established a strategic partnership arrangement with SecureKey Technologies Inc. and had procured a license from them for the Exchange software and implemented it in its hosting centre in France. Based on this relationship Idemia chose their SecureKey ID federation hub environment to deliver this project.

This implementation as the Idemia Innovation Lab is to support demonstration and proof of concepts of how digital identities can be federated into relying parties. Additionally as Idemia is engaged in many projects with governments around the world, SecureKey's Exchange hub was chosen as it delivered all the main features that Governments needed from a federation hub, and in particular the feature known as triple blinding. This is a feature where an identity provider cannot see which government service RP is requesting an authentication, the government service won't know which identity provider is being used,; i.e. the IDP, the Government Service, and the hub are blind.

At the end of 2016 the Association of British Insurers (ABI) ran a project to build a pension finder service that used a citizen's Gov.UK Verify LOA2 digital identity to locate any pensions that they may have to their name which they could have lost contact with. Idemia participated in the construction of the prototype and delivered the federation hub on the SecureKey Exchange hub deployed in the innovation lab.

During the first quarter of 2017 when the proposal to the INEA was being created it was decided that this existing technology asset should be employed as the federation entity for this project. In doing so many project tasks, common to the ABI project and this project were already completed and would represent a cost reduction to the INEA in the delivery of this component of the architecture.

Further details of this project can be found at: https://pensionsdashboardproject.uk/

Additional information on the SecureKey briidge Exchange can be found in Exhibits one and two:

Exhibit 1: Exchange Whitepaper.pdf

Exhibit 2: Exchange Datasheet.pdf

eIDAS component

To make the hub in the Innovation Lab capable of operating in this architecture the SecureKey Exchange hub would need to have a new connector created to allow it to interface to the eIDAS infrastructure. This was designed and specified in conjunction with the UK Government Digital Services eIDAS implementation team. There were two workshops convened in London in Q1/2 of 2017 to draw up the development requirements for the creation of this connector. SecureKey's senior technical team were present along with members of Idemia's service implementation team and lead subject matter experts form the UK Government Cabinet Office. The output of these

workshops was the requirements documentation submitted to Securekey for development. This is outlined in Exhibit 1.

About Mobile Connect et Moi & Mobile Connect

Mobile Connect et moi is the French Identity Application leveraging Mobile Connect, the secured mobile authentication service. Mobile Connect et Moi and Mobile Connect services are available to any French citizen or foreigners established in France. At time of writing the services are only available to Orange France customers, however more French mobile operators are forthcoming in the next months.

Mobile Connect is based on GSMA set of standards allowing any Mobile Network Operator to provide to business partners or Authorities a strong and simple authentication service and data sharing but with explicit user consent. Users are identified by their mobile number and a secret personal code.

In addition, Mobile Connect conforms to the European eIDAS and PSD2 standards regarding strong authentication as it is a two-factor authentication relying on[1] :

- Possession of the mobile phone embedding the SIM supporting the mobile number;
- Knowledge of a four digits Personal Code.

A typical Mobile Connect et moi use case gathering user authentication and identity attribute sharing is depicted in the following diagram.

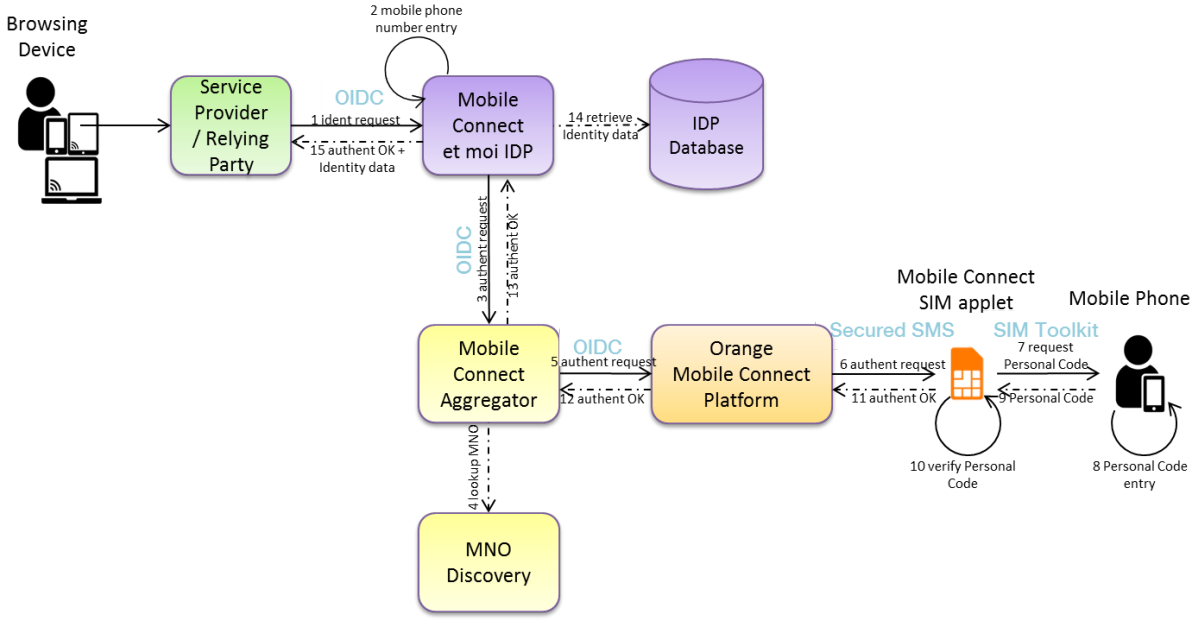


Figure 4 - Mobile Connect authentication flow

[1] Inherent factor such as biometrics is supported by the Mobile Connect standard but not yet deployed in France.

Consent phase is not shown in this diagram because it can vary depending on implementation and commercial agreements between actors and can be required only at the first authentication for a given service provider or after each one. For example in the eIDAS case, consent to share data is collected after each authentication on any French IDP by France Connect which could be seen as the "Service Provider / Relying Party" component in this diagram.

Technically, Mobile Connect strictly follows the OpenID Connect (OIDC) standard for authentication and adds some features such as server-initiated authorization or aliasing of user identifier for user privacy.

Mobile Connect comes with a Discovery mechanism allowing knowing the Mobile Network Operator of the user's mobile number and retrieving the associated authorization endpoint; in the standard GSMA architecture, the Service Provider must request this Discovery first with the Discovery protocol whereas in Mobile Connect et moi, the Discovery mechanism is hidden behind an Aggregator component, which complies with OIDC standard. Other roles of the Aggregator are the mapping of the three eIDAS Levels of Assurance (LoA) to some of the four Mobile Connect LoAs and the routing of the request to the right MNO endpoint.

The implementation of the two-factor authentication on the mobile phone can vary from one mobile operator to the other. For Orange France, the solution is developed and hosted by its Orange Business Services division. It relies on a Javacard applet residing on the user SIM card triggered by secured binary SMS sent by the Mobile Connect Authentication Signature Platform determining the interactions with the user thanks to the SIM Toolkit implementation available on even the oldest feature phones as well as most recent smartphones. The Mobile Connect Personal Code is initiated at registration and stored only in the SIM and nowhere else, strongly reducing the risk of fraud and data leakage. Moreover all transaction results, whether successful or failed, are signed by the applet thus making them non-reputable.

Finally the Mobile Connect et moi component collects identity attributes of the users through a smartphone mobile application (not shown as not part of the project) and stores them in its database. Mobile Connect et moi is also an OIDC endpoint which can provide them to Service Provider after successful Mobile Connect authentication of the user.

All the above components follows the best security practices, the SIM applet has received a "CSPN" certificate from the ANSSI (French IT Security Agency) and the whole solution is being evaluated to gain the eIDAS Substantial level for Electronic Identity.

https://www.gsma.com/identity/mobile-connect

Attribute exchange solution

During the discovery phase of the project it was found that there were no sources of personal data such as credit rating agencies, such as Experian, Lexis Nexis etc, to provide data that could be used by the banks for mitigating risks in the customer on boarding process. To progress the project and to deliver the attribute exchange capability Idemia created a mock up data source that contained address information for the personas created for the prototype. This data was installed in an AWS server along with a ForgeRock Authentication server to manage access and consent for the movement of attributes.

Frameworks

The sections describe the technical components and considerations that were implemented in the prototype.

Physical infrastructure framework

The main consideration for the infrastructure to deliver this project were to decide if the servers should be operated from an Idemia data centre or on another third party provider. The decision was to use AWS for the briidge Exchange and attribute exchange servers.

Initially the Innovation Lab hub server was hosted in an Idemia data centre but as the briidge Exchange software needed to be upgraded in order to run the new eIDAS connector, a decision to move it into an AWS environment during the upgrade was taken.

SecureKey Virtual Private Cloud (VPC)

The diagram below describes the network that the SecureKey hub resides in. It also illustrates the interconnection of the Idemia test IDP that is a replica of the IDP delivered to the UK Government's Verify digital identity scheme. This allows for testing to be conducted using Verify IDs prior to testing IDs through the eIDAS component.

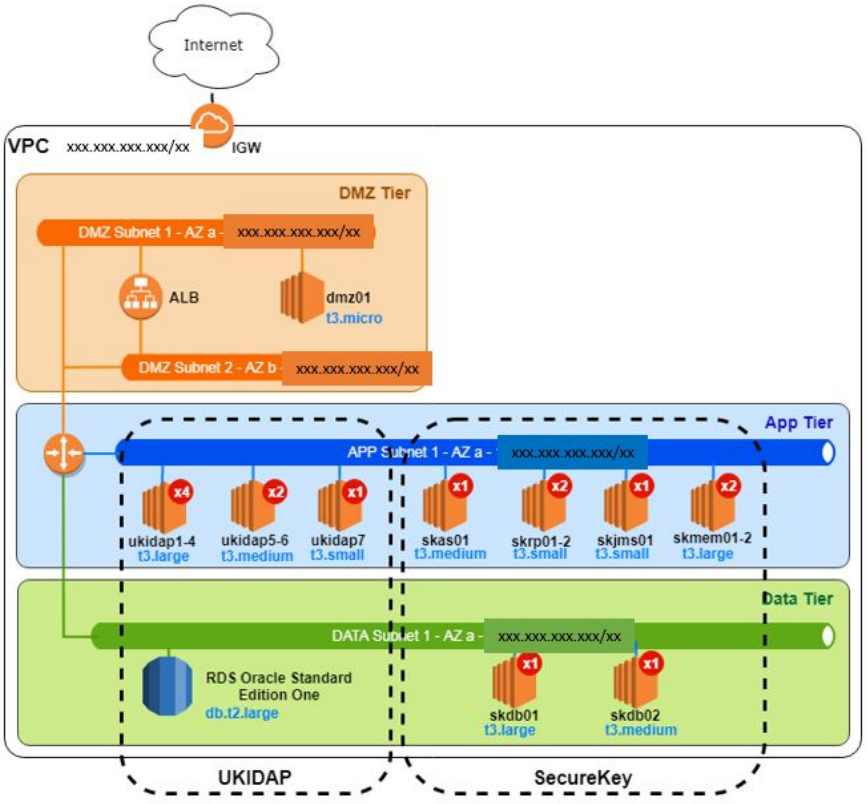


Figure 5 - SecureKey VPC

The following section describes the main components in the SecureKey VPC.

SecureKey VPC architecture has been segregated into three different tiers i.e. DMZ, application zone and database zone with Idemia public IP addresses exposed through a load balancer, which in turn is used to access all applications and database instances.

Flows between applications and database hosts are governed by network and application protocols using dedicated ports.

DMZ Subnet

SecureKey is using two subnets in the DMZ tier, one being exposed to the application and database instances while the second subnet is used by the load balancer internally.

ALB

AWS Application Load balancer has been used to connect application and database instances on designated ports to allow “https://” flows from Idemia public IPs to instances FQDNs (fully qualified domain name)

App Tier

We have application virtual machines hosted on this subnet, which includes the SecureKey application hosts, message processor, active MQs along with versioned software installed for running them. This can connect internally and database instances via bastion host.

Data Tier

We have database virtual machines hosted in this subnet, which includes SecureKey MySQL Percona cluster, reporting & logging MySQL instance along with Memsql for caching purposes. Reports for billing, audits and miscellaneous purpose can be extracted under this particular tier.

Operating systems framework

Federation hub

The following paragraphs outline the operating system and software packages needed to establish the platform built to implement the briidge Exchange software.

Virtual Machine requirements

AWS RHEL 7.5 server instances App/DB server list:

1-2 x VMs (2CPU/4G RAM/20G storage) to host the Application Server;

1-2 x VMs (1CPU/2G RAM/20G storage) to host the Test RP/CSP Application Servers;

1-2 x VMs (1CPU/2G RAM/20G storage) to host the JMS Server(s) (AMQ);

1(or 3) x VMs (2CPU/4(8)G RAM/200G storage) to host the Percona MySQL for core data

Database

2 x VMs (2(4)CPU/8(16)G RAM/200G storage) to host MemSQL caching cluster

1-2 x VMs (1CPU/2G RAM/20G storage) to host the Message Processor Application Server;

1 x VMs (2CPU/4(8)G RAM/200G storage) to host the Percona MySQL for Logging / Reporting Database;

SecureKey Virtual Machines

Host Name (VM)	*Instance type*	*Root disk*	*AMI*	*Purpose*
amq01	t3.small	20GB	CIS Red Hat Enterprise Linux 7 Benchmark Level 1	ActiveMQ Broker 1
csp01	t3.small	20GB	CIS Red Hat Enterprise Linux 7 Benchmark Level 1	Test CSP Application
db01	t3.large	200GB	CIS Red Hat Enterprise Linux 7 Benchmark Level 1	Percona MySQL cluster Node 1
dmz01	t3.micro	20GB	CIS Ubuntu 16.04 Level1	
fed01	t3.medium	20GB	CIS Red Hat Enterprise Linux 7 Benchmark Level 1	Broker Application Node 1
memsql01	t3.xlarge	200GB	CIS Red Hat Enterprise Linux 7 Benchmark Level 1	Caching MemSQL cluster Node 1
mp01	t3.small	20GB	CIS Red Hat Enterprise Linux 7 Benchmark Level 1	Message Processor and Migrator Application
rdb01	t3.large	200GB	CIS Red Hat Enterprise Linux 7 Benchmark Level 1	Percona MySQL Logging and Reporting Database Single Node
rp01	t3.small	20GB	CIS Red Hat Enterprise Linux 7 Benchmark Level 1	Test RP Application

Figure 6 - Server characteristics and functions

Software packages

The main operating systems used was Redhat Enterprise Linux 7.5 along with prerequisite applications installation:

OpenJDK 1.8.0_181 for App/TestRP/TestCSP/JMS AMQ servers

MySQL(Percona) 5.6.40 for Core DB and Logging/Reporting servers

MemSQL OPS 6.5.11 & MemSQL BIN 6.5.17 for MemSQL caching cluster

ActiveAMQ 5.15.5

Tomcat 8.5.33 with APR 1.2.17

HTTPd Apache for Content servers (Application servers)

Protocol framework

The following interconnection protocols were implemented in the architecture.

OIDC

https://openid.net/connect/

SAML / Verify SAML

https://alphagov.github.io/rp-onboarding-tech-docs/pages/saml/saml.html

eIDAS

https://joinup.ec.europa.eu/document/eidas-technical-specifications-v10

HTTPS

http://www.w3.org/Protocols/rfc2616/rfc2616.html

UMA

https://docs.kantarainitiative.org/uma/rec-uma-core.html

GSMA Mobile connect

https://www.gsma.com/identity/mobile-connect

User Interface

HTML4/5

https://www.w3.org/TR/html52/

JavaScript

https://www.w3resource.com/javascript/introduction/ECMA-and-javascript.php

CSS https://www.w3.org/Style/CSS/specs.en.html

Messaging flow

This section describes the overall messaging flow for the prototype and correlates with the architecture previously described.

For the prototype, there is a simple consent step, but no authentication step for the address attributes exchange because the trust framework would permit the IDP authentication to act as subsequent consent for data release and the user address information are retrieved according to the user's identity information. We assume that for production, the user could be prompted to authenticate itself to the attributes Data Provider and consent with sharing of its address information.

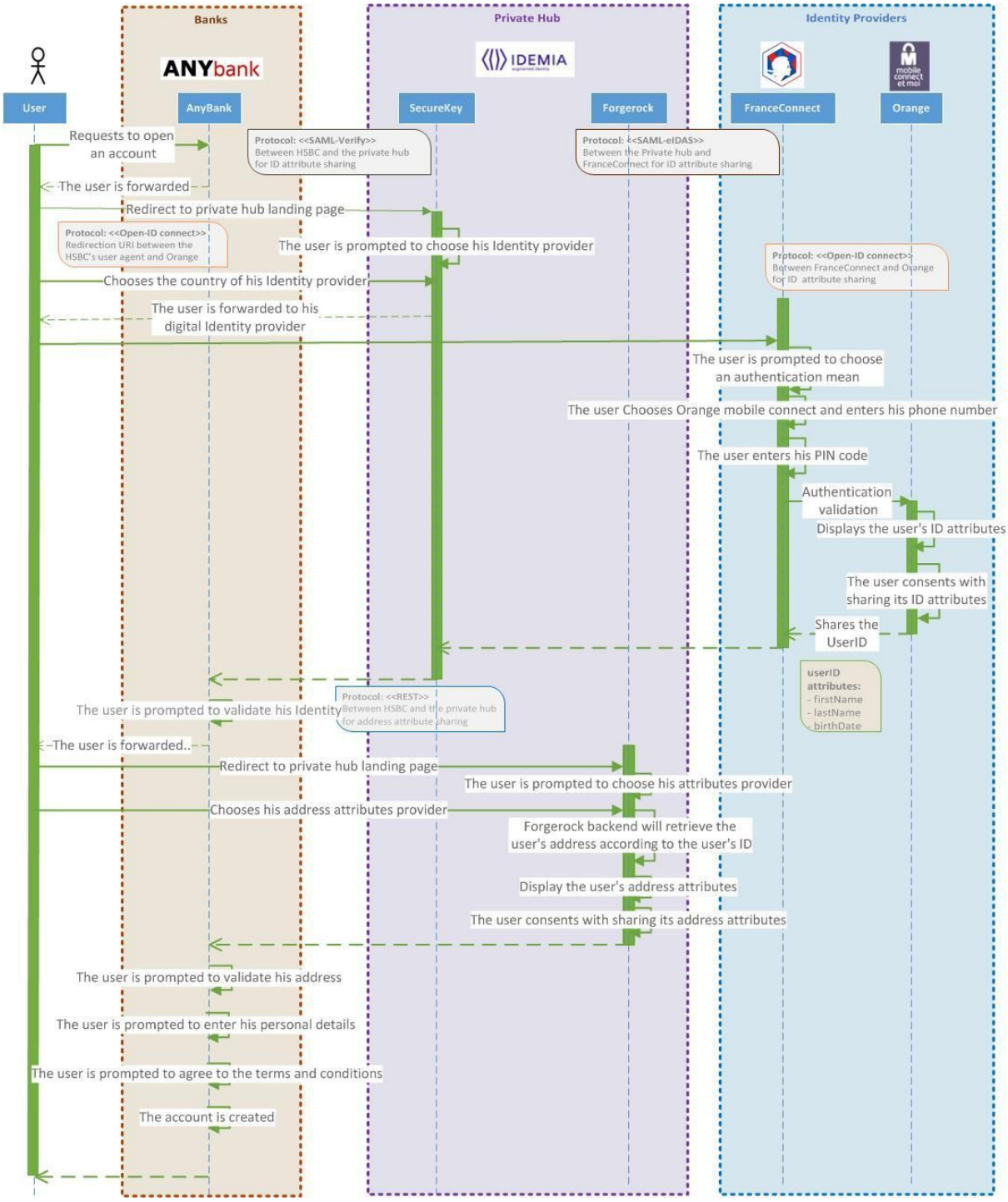


Figure 7 - Overall flow and messaging

This section describes the overall messaging flow for the expected production solution.

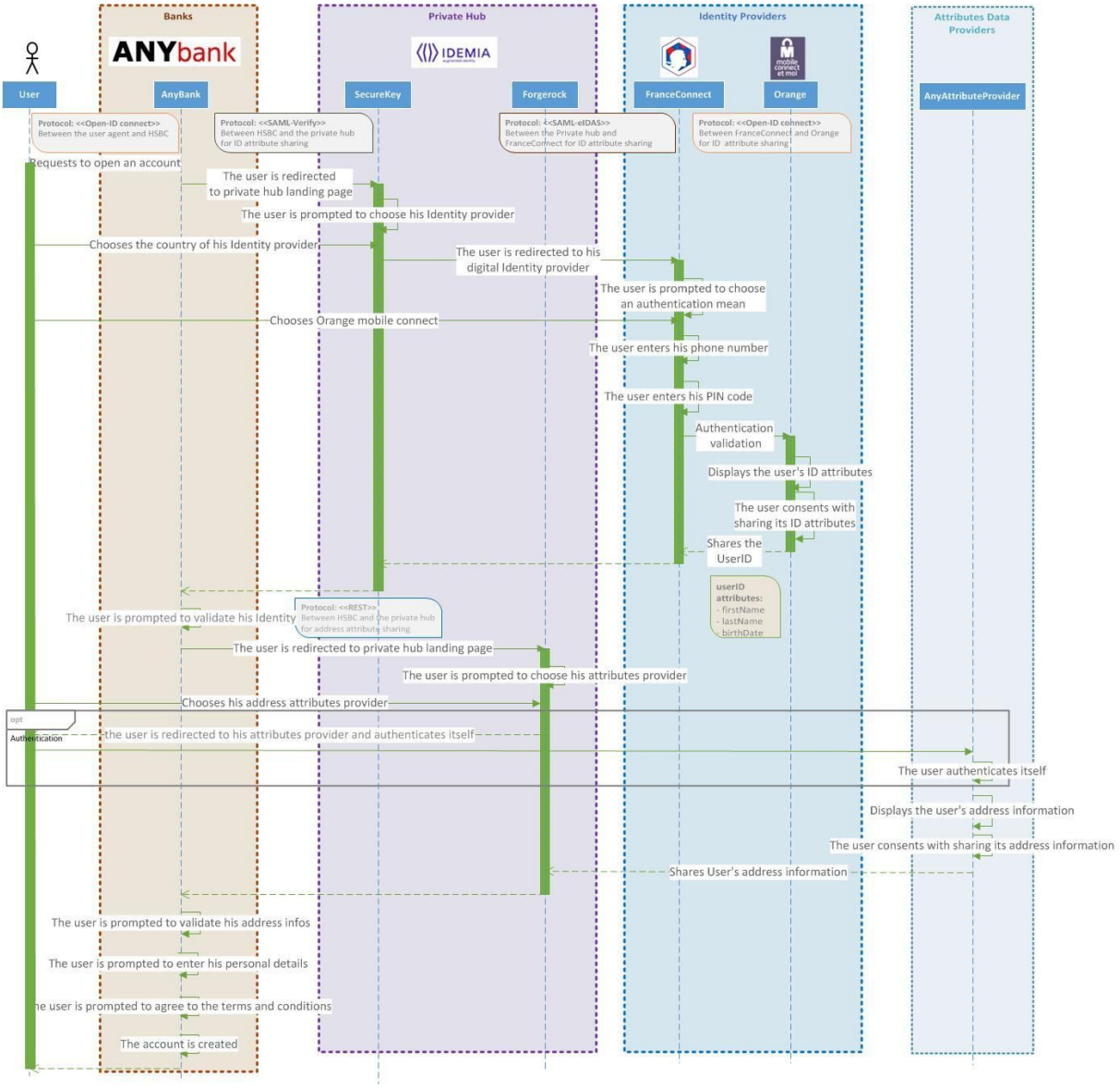


Figure 8 - Expected production flow

Exhibit 1

CEF hub service eIDAS node implementation requirements

Idemia developed the eIDAS components to comply with eIDAS specifications:

Exhibit 2: eIDAS Message Format_v1.1-2.pdf

Exhibit 3: SAML Attribute Profile v1.1_2.pdf

The incremental features to the Idemia ID hub are outlined in Table 1 and 2.

Table 1

#	FEATURES	Feature and Implementation Comments
General Feature		
1	<md:EntitiesDescriptor> Support	Need to support loading of metadata from <md:EntitiesDescriptor> (multiple entities) both RP and credential service provider ("**CSP**") sides. Should be supported according to security assertion markup language ("**SAML**") 2.0 spec.
2	Adding urn:oasis:names:tc:SAML:2.0:name id-format:transient support	Transient flow should be fully supported according to SAML 2.0 spec. Currently support persistent flow.
3	Adding urn:oasis:names:tc:SAML:1.1:name id-format:unspecified support	Unspecified flow should be fully supported according to SAML 2.0 spec.
4	<eidas:SPType> support	Custom extension - general description from the eIDAS profile: For indicating whether an authentication request is made by a private sector or public sector service provider ("**SP**"), the defined element <eidas:SPType> MUST be present either in the <md:Extensions> element of SAML metadata or in the <saml2p:Extensions> element of a <saml2p:AuthnRequest>. If the SAML metadata of an eIDAS-Connector contains a <eidas:SPType> element, SAML authentication requests originating at that eIDAS-Connector MUST NOT contain a <eidas:SPType> element. The <eidas:SPType> element can contain the values "public" or "private" only. Exchange Implementation Requirements: 1>if Received from RP, should be passed to the CSP. 2>Should be definable per CSP (provider files). If the switch is present and value defined should send the parameter with the authentication request to that CSP.

Requesting Attributes		Requesting attributes by an eIDAS-Connector from an eIDAS-Service MUST be carried out dynamically by including them in a <saml2p:AuthnRequest>. Instead of using AttributeConsumerServiceIndex and similar to the way we pass attribute expression from federation to broker eIDAS requests attributes using custom extension with <eidas:RequestedAttributes> element and listing all requested / optional attributes in that extension that resembles attribute statement SAML elements.
5	Adding Attribute Request Extension support with listed required/optional attributes -<eidas:RequestedAttributes>\	Custom SAML extension with optional and required attributes requested dynamically. Exchange implementation requirements: SECUREKEY Exchange should be able to receive eIDAS AuthRequest with requested attributes extension
6	Adding Attribute Request Extension support with listed required/optional attributes -<eidas:RequestedAttributes>	Custom SAML extension with optional and required attributes requested dynamically. Exchange implementation requirements: SECUREKEY Exchange should be able to receive eIDAS AuthRequest with requested attributes extension: Exchange broker should be able to send authentication request to the CSP(based on CSP provider settings) with <eidas:RequestedAttributes> describing the requested attributes
eIDAS SAML Attribute Format		
8	Support Attribute value with custom eIDAS type defintion All verify attributes must be mapped to eIDAS attributes and vice versa (CF. GDS specification)	<saml:Attribute FriendlyName="FirstName" Name=" http://eidas.europa.eu/attributes/naturalperson/CurrentGivenName" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"> <saml:AttributeValue xsi:type="eidas:CurrentGivenNameType"> Sarah</saml:AttributeValue> </saml:Attribute> Exchange implementation requirements: Exchange should be able to pass custom attribute value type for the attribute from CSP to RP
9	Supporting eIDAS SAML Attribute Profile encoding and Attribute Structure requirements	General customization work to support required and optional eIDAS attributes with requested encodings Exchange implementation requirements: pass-through from CSP to RP
10	Supporting basic eIDAS attribute types	
11	Supporting complex eIDAS types with encoding	pass-through from CSP to RP

SecureKey Development Eng and Project Support		
12	SecureKey Development Engineering	builds, package, deploy, performance tests, performance reports, operations handover + Docs
13	Product / Project Management, Integration support, Solution Architecture support	
UK Verify requirements		
14	Create Exchange product extension mechanism(plug-in) that will allow SAML to SAML translation (from one SAML profile to another including attribute format transformation)	SecureKey will provide configuration and interface description for the "SAML to SAML IDP Response translation" plug-in mechanism.
15	eIDAS - UK Verify translator based on plug-In mechanism	Exchange implementation requirements: 1> Part ok UK Verify private sector Exchange based Hub 2> Receives assertion from eIDAS CSP (eIDAS attributes and eIDAS response message format) 3> Generates UK Verify response message with UK verify attribute statement from the received eIDAS assertion 4>Sends the message back to RP that requested the authentication SecureKey will provide "eIDAS - UK Verify Translator" source code as an example of "SAML to SAML IDP Response translation" plug-in implementation.
16	UK Verify - eIDAS Translator based on Plug-In mechanism	Exchange implementation requirements: 1> Receives assertion from UK VERIFY (UK VERIFY attributes and UK VERIFY response message format) 2> Generates eIDAS response message with eIDAS attribute statement from the received UK VERIFY assertion 3>Sends the message back to RP that requested the authentication SecureKey will provide "eIDAS - UK Verify Translator" source code as an example of "SAML to SAML IDP Response translation" plug-in implementation.
17	SecureKey Development Engineering	
18	Product / Project Management, Integration support, Solution Architecture support	

Exhibit 2

eIDAS Message Format

https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eIDAS+Profile?preview=/46992719/47190132/eIDAS%20SAML%20Attribute%20Profile%20v1.1_2.pdf

Exhibit 3

SAML Attribute Profile

This specification defines the SAML attributes to be used for the assertion of natural and legal person identity between eIDAS nodes.

https://ec.europa.eu/cefdigital/wiki/download/attachments/46992719/eIDAS%20SAML%20Attribute%20Profile%20v1.1_2.pdf?version=1&modificationDate=1497252920100&api=v2