Digital Scotland
Digital Identity Scotland

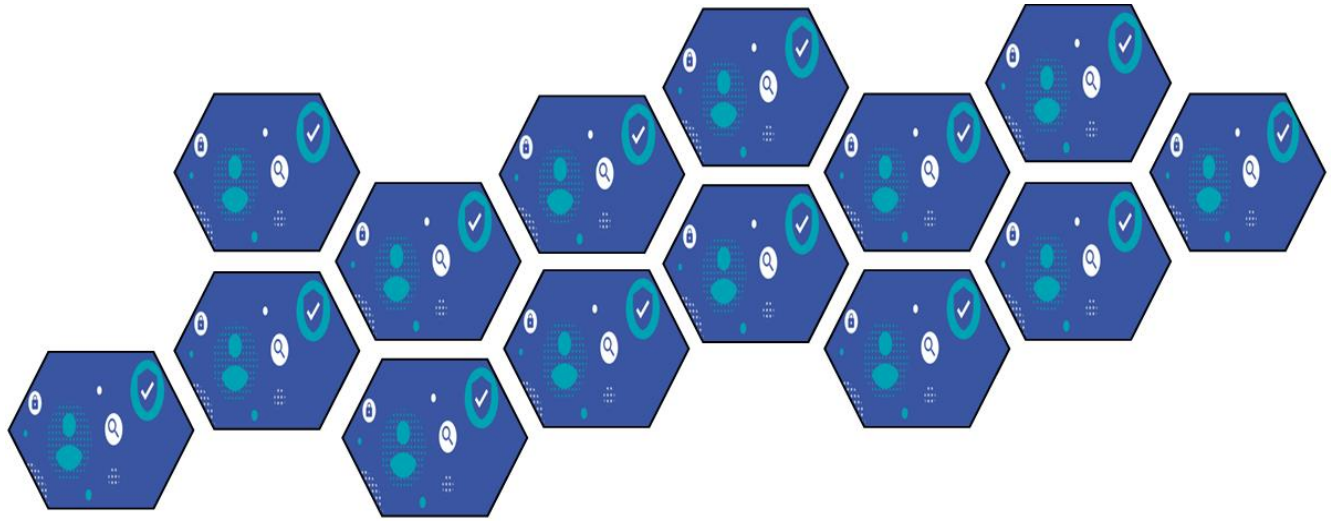
Scottish Government
Riaghaltas na h-Alba
gov.scot

Scottish Government

Digital Identity Scotland

Market Engagement

Indicative Requirements



6 October 2020

Table of Contents

Document History	2
Authorisation	2
Related Documents	2
Introduction	3
SAPS Components High-Level Indicative Requirements	4

Document History

Author/Reviewed By	Date	Version
Author Claire Lumsdaine	1/10/2020	V1.0
Reviewers **Mike Crockart Liza McLean**	5/10/2020	V1.0

Authorisation

No	Name	Title	Date
1	**Mike Crockart**	**Service Manager DIS**	**06/10/2020**

Related Documents

Number	Title	Version/Date
1	DIS SAPS Market Engagement Questions	**October 2020 v1.0**
2	DIS SAPS Market Engagement Presentation Slides	**October 2020 v1.0**
3	DIS SAPS Market Engagement Presentation Script	**October 2020 v1.0**
4	DIS SAPS Technical Brief for Industry	**October 2020 v1.0**
5	DIS SAPS Strategy	**June 2020 v1.0**

Introduction

This note is for the market / industry recipients participating in a non-binding PIN (Ref: AUG392892*)* in Q4 2020. It supports a higher level Technical Brief (Ref. 04), providing **indicative non-binding high-level requirements** for technical capabilities which might deliver the needs of the Digital Identity Scotland, Scottish Attribute Provider Service (SAPS) programme. These indicative requirements are provided to enable industry responses to the engagement **in an informed and fully contextualised way**.

The programme expects responses to its needs and challenges, and wishes to avoid general marketing or pitches based on generalised / hypothesised needs. **These documents take the position of making public significant levels of detail of our considerations to date, and transparently seeking industry comment, proposition and counter-proposal.**

SAPS Components High-Level Indicative Requirements

This paper covers the key components we believe are required to deliver the service, as described (Ref. 04):

- Credential Provider (CP), potentially separated into capabilities – authorisation service (AZ) and/or authentication methods (AM)
- Attribute Store (AtS) and its incorporated capability, the Consent Manager (CM) which includes the authorisation service of the AtS
- Light-weight Broker (BK) (stating functional requirements for the SAPS features which the broker supports rather than specifying product characteristics of middleware products).

For each of these capabilities, we state indicative high-level requirements. The highest level of requirements is 'level 1', for example denoted CP.1.nn, where nnn is a unique number for that requirement. Level 1 requirements trace to material in *(Ref. 04).* Sub-requirements are denoted by their level, e.g. *CP.2.nn in italics,* and traced to the higher-level requirements which they extend or clarify or make more specific. Each requirement is classified as part of a capability (AZ/AM, AT/CM) to assist in the possible separation of the capabilities, e.g. separation of authentication service from authentication methods see (*Section 4.6, 4.7 of Ref. 04)*.

This document focuses on **functional technical requirements**, it intentionally does not include many non-functional requirements, privacy, usability characteristics or otherwise, since the focus of this market engagement is on technical responses.

For level 1 requirements 'trace' will cross reference material in *(Ref. 04).*

This document does not address alternative designs or architectures: it presents information to enhance the contextualised response to the market engagement based on SAPS core needs.

ID (Capability)	Requirement	Rationale	Notes	Trace
CP	**Credential Provider High-level Requirements**			
CP.1.02 (AM)	The CP shall implement strong authentication of the bound user from account initiation.	Strong credential is the root of incremental trust for the ecosystem, so there must be no delay in establishing a strong credential.	At no time is a credential used to authenticate to a RP or AtS without being provably 'strong'. (It is accepted that this means for accounts with *only* postal loop 2nd factors, the account cannot be used until the second factor is activated.)	2.3 2.4
CP.1.03 (AZ)	Account Identifier shall be unique, near random, high entropy, persistent and never-reused.	Account identifier is the anchor for correlation of user account with verified attributes and the AtS.		
CP.1.04 (AZ)	Authentication assertion/token contains minimal information, limited to account identifier.	Near anonymous authentication – no personal biographic / biometric data is exposed with the authentication.		3.2
CP.1.05 (AZ)	Authentication context shall carry information related to the authentication event and recent history (e.g. IP, geo-IP, velocity, retries, authenticators used, device characteristics).	Ecosystem risk management – protecting from account takeover and detecting attacks. SAPS will perform a level of security and risk monitoring which includes authentication context data from the CP.		4.2
CP.1.06 (AM)	Minimal personal information must be captured and/or retained	Avoiding aggregation and minimising value of attack.		2.4

	over time, keeping only that which is necessary to support the credential lifecycle.			
CP.1.11 (AM)	User has a choice of authentication methods, including smart phone push apps, third party authenticator apps, 'offline' telephony, postal grid cards and others as technology advances.	Many types of user capability, technology and form factors supported to assist different user types, move towards no passwords and to maintain pace with attacks.		2.4 4.2
CP.1.12 (AM)	Users can change their authentication method(s) as they wish.	Users' needs change over time. Different routes also minimise the risks of being unable to recover the account.		2.4
CP.1.13 (AM)	If the user has configured more than one combination of authenticators, the user shall be offered a choice of authentication method during an authentication activity.	Convenience of use based on the context of a specific event. E.g. I might usually use push auth to my smartphone, but I want to use an offline grid card when in a remote location.		2.4
CP.1.14 (AM)	Users should have an option for a 'user name' which is *not* their email address.	For users who do not wish to use email address: either for enhanced privacy and security, or because they do not have one.	Note connections with credential lifecycle management. Email might be offered by the user *only* for password reset. Password reset may be provided via another channel.	2.4
CP.1.15 (AM)	Credential recovery must be self-service across whichever (preconfigured) channels the customer	e.g. I have lost my smartphone, so I select my offline backup codes, or my grid card, or landline telephony code.	Resetting the second factor through the same channel (e.g. email) as the first factor is	

	selects to support their reset.		unlikely to be secure & compliant.	
CP.1.21 (AM)	User support must be offered in appropriate channels. User support should cover advice on use of authenticators and how to self-serve all lifecycle management events.	Usability, assistance and focus on self-service.	Agent-mediated service involving direct management of customer accounts by agents may be desirable, but we note the security issues.	
CP.1.31 (AZ)	The user's account at the CP shall be capable of holding arbitrary information written and read on behalf of the user by user authorised SAPS components (i.e. AtS or broker).	Supporting low friction user journeys, avoiding persistence in broker. Specifically, data related to the AtS account instance and related tokens.	e.g. OIDC 'claims' to/from user profile, note updates by *user* authorised components, *not* sys-admin accounts. Note encryption and sender constrained tokens and other controls will apply.	3.4
CP.1.32 (AZ)	The authentication service shall support silent re-auth (checking for a session without user intervention necessarily required), and incremental auth (specific user check in a session e.g. a confirmation of a specific push message), and forced re-auth (all factors whether or not there is a current session).	Minimising user friction whilst enabling appropriate dynamic re-authentication based on the RP/broker/AtS transactional context.		3.2 3.4 (e.g. of push message on back channel)
CP.1.91 (AZ)	The authentication service will support mechanisms of close collaboration with the AtS/CM to minimise	Back channel authorisation keeps browser and user focus in the RP session.		3.4 (e.g. of push message) 3.5 (potential for common protocols)

	user journey friction and keep users in back channel, such as: ● Custom push authorisation messages ● CIBA, app2app ● Joint mobile app dev for both CP and AtS-CM.	Custom messaging giving specific contextualised message during an authorisation is common best practice. CIBA/app2app are Open banking best practice to minimise user friction in collaborative journeys. Common app dev on behalf of both suppliers is another route to explore.		4.7
CP.1.92 (AM), (AZ)	CPs shall be *certified* GPG44 level Medium.	Standardised certified interpretation of guidance in GPG44.	Ensures independent view of whole of credential management, not just '2FA'.	2.4
CP.1.93 (AZ)	Authentication Server shall implement OIDC Provider with OIDC clients in broker and AtS.	OIDC as basis of *authentication* in SAPS ecosystem.	See and note separate *authorisation* re AtS see AT.1.93.	2.4
AT	**Attribute Store High-level Requirements**			
AT.1.01 (AT), (CM)	All verified attributes and consents are owned, controlled and private to the owner (data subject) and cannot be accessed by the platform, government, or anyone else.	Only the owner (or their delegate) can access. Platform is assumed to encrypt data to a key specific to the owner.	Privileged users of the platform must have appropriate controls AND preferably the platform shall have *zero knowledge* of the encryption key.	2.6 2.7 2.11 5.4 (ZK)
AT.1.02 (CM)	The AtS will manage all aspects of consent and related authorisation to disclose or to update	The consent process is logically part of the AtS CM. In requesting disclosure of attribute(s) an RP should not be able to deduce anything other than		2.8

	verified attributes, keeping the existence or otherwise of attributes as well as their values and metadata confidential unless consent to disclose is given by the owner.	the outcome in the form of a returned attribute(s). In offering update(s) of attributes an RP should not be able to deduce anything at all about an update, its acceptance, the user or their AtS.		
AT.1.03 (AT), (CM)	There shall be mechanisms to recover an AtS account independently of the CP.	Separation of AtS from CP – segregation of concerns, 'firewall' if CP compromised or vendor change, support for users in event of loss of CP credential.		3.4
AT.1.04 (CM)	The AtS shall support federated authentication with SAPS CP.	Ease the user journey minimising authentications by SSO between CP, RP, AtS and broker. Enable the user to maintain their consents independently of a SAPS RP or broker.		3.4 CP.1.31
AT.1.11 (CM)	All disclosure shall occur only when the owner user (or delegate) is in session *and* only when the owner user gives specific consent.	Dynamic consent by owner user. Predetermined consent by owner for a specified delegate for a specified service.	If consented for disclosure the whole of a verified attribute – data and all metadata is disclosed.	2.7
AT.1.12 (CM)	All updates shall occur only when the owner user (or delegate) is in session, *and* only if there is a matching specific consent already given (and not revoked) by the owner user, or if the owner user gives specific consent at the time.	Dynamic consent by owner user. Predetermined consent by owner for a specified delegate for a specified service.	If consented an update is of the *whole* of a verified attribute – data and all metadata. Attribute maintenance over time is simply repeating updates whilst consent to update is in place.	2.6 2.7
AT.1.13 (CM)	Only the owner user can give, revoke or withhold	Only the owner can manage their own consents (including those to delegate).		3.5 3.6

	consent to disclose/update.			
AT.1.14 (CM)	Consent always relates to an attribute in the context of a service (i.e. disclose to service, update from service).	Owner users control from/to which service, permit updates, services cannot overwrite each other's attributes.	Metadata and data are inseparable, metadata includes origin RP service.	2.5
AT.1.15 (CM)	The owner user can grant a delegate the ability to access (disclose and/or update) in accordance with the owner's consent policy.	Delegates support the owner in use of RP services. (Only owners can manage consents or view AtS contents directly.)	Only the owner, at the time of delegation or later by modifying delegation consent, can specify which attributes can be disclosed to specific services or updated from specific services.	2.11
AT.1.21 (AT)	AtS support derivation of standardised attributes from other attributes, e.g. age>18 from date of birth and 'today', e.g. local authority from residential address. (AtS will implement standard rules, attach metadata from root attributes as per a standard, and sign the resulting derived attribute.)	Minimising disclosure.	Simple standard derived attributes to be defined. (More complex derivations will run in SAPS special processes such as Identity on Demand Service, outside the AtS itself.)	2.9 4.8
AT.1.31 (AT)	AtS manages matching of requested attributes (expressed as a metadata description) against the user's actual store.	Enable the RP to specify what it needs and the user to select and consent.		2.10

AT.1.32 (AT)	Present attributes which match a metadata description for user selection.	Enable user selection for disclosure, consent (including delegation) or deletion.		2.10
AT.1.33 (AT)	Enable the user to specify a metadata description to support search/browsing.	Support user consent or deletion.		2.10
AT.1.34 (CM)	Support delegation to another SAPS user: • Specify delegate user (who provided their details out of band) • Select RP service • Select disclose / update / both • Select attributes / all attributes.	Set up consents for subsequent delegate sessions with the RP.	Assumes a catalog service accessed by the AtS CM at the time of consent to delegate. Catalog of SAPS services and the attributes they consume or which they originate.	2.11
AT.1.92 (CM)	A CM user interface provided as a native mobile app should enable back channel interaction including authorisation.	Back channel will keep the user browser in focus on the RP and provide (for users willing to use apps) simpler, lower friction user journeys.	*<ME01>* only presents flow diagrams for front channel journeys.	3.6 4.6
AT.1.91 (CM)	The AtS CM app should be able to collaborate with an authenticator app (CP) in accord with requirement CP.1.91	See CP.1.91		4.6
AT.2.91 (CM)	AtS will implement an OIDC Client to enable interop with CP.	Federated authentication. Management of profile information from AtS.		AT.1.91 CP.1.93

AT.1.93 (CM)	AtS will implement a mechanism of authorisation of calls to r/w attributes (an Authorisation Server, or federated Authorisation Server).	Broker will r/w attributes on behalf of services, using an API on the AtS. This API should be authorised appropriately.	Two possible mechanisms – standard OAuth, or preferably UMA 2	3.5, 3.6 4.6
AT.2.92 (CM)	The AtS Authorisation server will support structured authorisation tokens.	Mechanisms of token protection and scoping are probably required.	Plain bearer tokens probably not adequate. AtS refresh token sender constrained. Note also connection with UMA.	AT.1.93 CP.1.31 (notes)
AT.2.93 (CM)	(Preferably) AtS may support federated authorisation and fine grained per attribute resource authorisation based on UMA 2.	Coarse-grained OAuth API authorisation by system to system components leaves much of the authorisation decision to be made within the CM Authorisation Server. User Managed Access 2 is preferable providing fine-grained, resource based, user centric policy, which can be provided asynchronously from access.	UMA handles failure cases more gracefully than pure OAuth2, especially relevant to managing subsequent browser redirection after a failed back-channel AtS operation. UMA also for delegation use cases.	AT.1.93
BK	**Broker High-level Requirements**			
BK.1.01	Orchestration of SSO across CP, RP, AtS and broker.	SSO across basic components in the context of a single RP at once.		3.2 4.4
BK.2.01	Broker includes OIDC client for CP.	See diagram in 3.2.		BK.1.01 CP.1.93
BK.1.02	Orchestration of consent, disclosure and update flows across CP, RP, AtS and broker.	Running the protocols to manage these interactions in the front (browser) and / or back channel (native app AtS CM / CP). Handling authorisation failure in the front channel, collaboration with the RP to redirect for front channel authorisation when required.		3.2 4.4

BK.2.02	Broker will implement a client for the AtS Authorisation Server.	Broker orchestrates r/w activity and needs explicit authorisation to access AtS. See diagram in 3.2.	UMA RqP or OAuth client.	BK.1.02 AT.1.93
BK.1.03	During a session with user and an AtS, orchestrate maintenance updates based on checking queue of VAs for the owner.	Attribute maintenance is • detecting a session with a user – owner or delegate, • checking a queue of updates for the owner, • and sequencing updates to the in-session AtS.	A 'queue' is located somewhere (TBD, perhaps at the RP, perhaps at a broker instance). Noting that a 'queue' may be a database of encrypted VAs, keyed by owner & delegate ids.	2.7 4.4
BK.1.04	Orchestration of CP, AtS, broker in synchronous session with IAP (Identity Attribute Provider) or other trusted source of attributes.	IAPs may provide credentials for synchronous user session or for direct broker access whilst user is in session.	Variety of IAP integration models possible.	3.8 4.8
BK.1.05	Orchestration of AtS and SAPS services such as IoDS; broker to obtain derived attributes from the AtS.	Supports the user in obtaining special RP dependent or Identity related derived attributes. Also necessary for SAPS as a federated IDP to an external scheme.		3.7
BK.1.06	Broker may expose pages for user interaction during redirection orchestration.	e.g. select a CP platform e.g. select an external IAP or input a correlation code e.g. explain an IoDS transaction		
BK.1.11	Protocol mediation/adaption – protocols between RP and broker are likely to be different from those between broker and AtS and external IAPs; redirection and api	RP integration needs to be isolated from the specifics of all scenarios in the SAPS components. RPs should only see authentication and dialogs to request/receive/write attributes in a standard metadata form. RPs should be unaware of the management of updates (whether or not updates are consented origin RPs will simply push updates to their queue).	Broker will support redirection-based flows e.g. OIDC, OAuth, and apis to support back-channel comms RP-CP/AtS (to include app based CM) and front side access to CP claims and AtS.	3.2 4.4

	protocols supported by broker.			
BK.2.11	Broker may need to invoke SAPS services by API during orchestration.	e.g. a service which transforms inbound IAP attributes into SAPS equivalent metadata and signs the resultant VA before returning to broker for write to the user's AtS. e.g. pulling from VA queue as per BK.1.03		BK.1.11 BK.1.03
BK.1.12	Session management – broker maintains a session state to support SSO, SAPS process orchestration, and logout.			3.2 4.4
BK.1.31	Distribution – broker *may* be distributed /deployed across domains (or even RPs) to support the distributed nature of the SAPS public services.	Avoiding commitment to deployment architecture, keeping state minimal (target is no state other than the current user session). (Keep update queues of verified attributes outside broker see BK.1.03 notes.)	e.g. health, benefit, council, business domains	4.4
BK.2.31	Messaging between broker instances (to seek attribute updates for the AtS when the user authenticates).	In case of distribution of broker, any one instance may be that at which a user session is created (as the user authenticates for an RP), yet updates may be elsewhere in distributed queues, so messaging might be one implementation option in this context.	Probably not generic middleware messaging features. Note: if distributed broker is not required there is no messaging requirement.	BK.1.31