New Zealand Post

INTERNAL AFFAIRS
Te Tari Taiwhenua

RealMe

Consent Service Specification

Version: 0.4

Author:

Date: 03rd December 2012

1 Introduction

1.1 Document Purpose

The purpose of this document is to describe the consent service which will be developed as part of RealMe solution. This document also describes the use cases that the consent service will support.

1.2 Background

The business processes at one service provider require verified or authoritative personal information from other service providers.

In the RealMe context the service providers that store this personal information are known as "identity attribute providers", and RealMe provides the platform for an individual to share their personal information held with the identity attribute providers with consuming service providers ("clients" in the RealMe context), in a secure and privacy centric manner. The RealMe service used to facilitate this transaction is known as the RealMe assertion service.

According to the RealMe privacy design, an individual's explicit consent is required for RealMe to broker sharing of their personal information between identity attribute providers and clients.

To support this requirement RealMe is implementing a centralised consent service, which can also be utilsed for transactions that require user consent where these do not involve the RealMe assertion service..

Because the consent service is intended for use outside the RealMe assertion service context, the remainder of this document does not distinguish between identity attribute providers and clients, and instead refers to these generically as "service providers".

1.3 Audience

The audience for this document is intended to be both IT professionals and business stakeholders from the service providers

1.4 Assumptions

The following assumptions have been made for this specification:

- The business trust and relationship must be established between service providers prior to sharing of personal information.
- The service provider must integrate with the RealMe logon service. The user authentication via the RealMe logon service is at the appropriate strength (low or medium) to gain access to other service providers.
- The specification has been developed in compliance with New Zealand privacy laws.

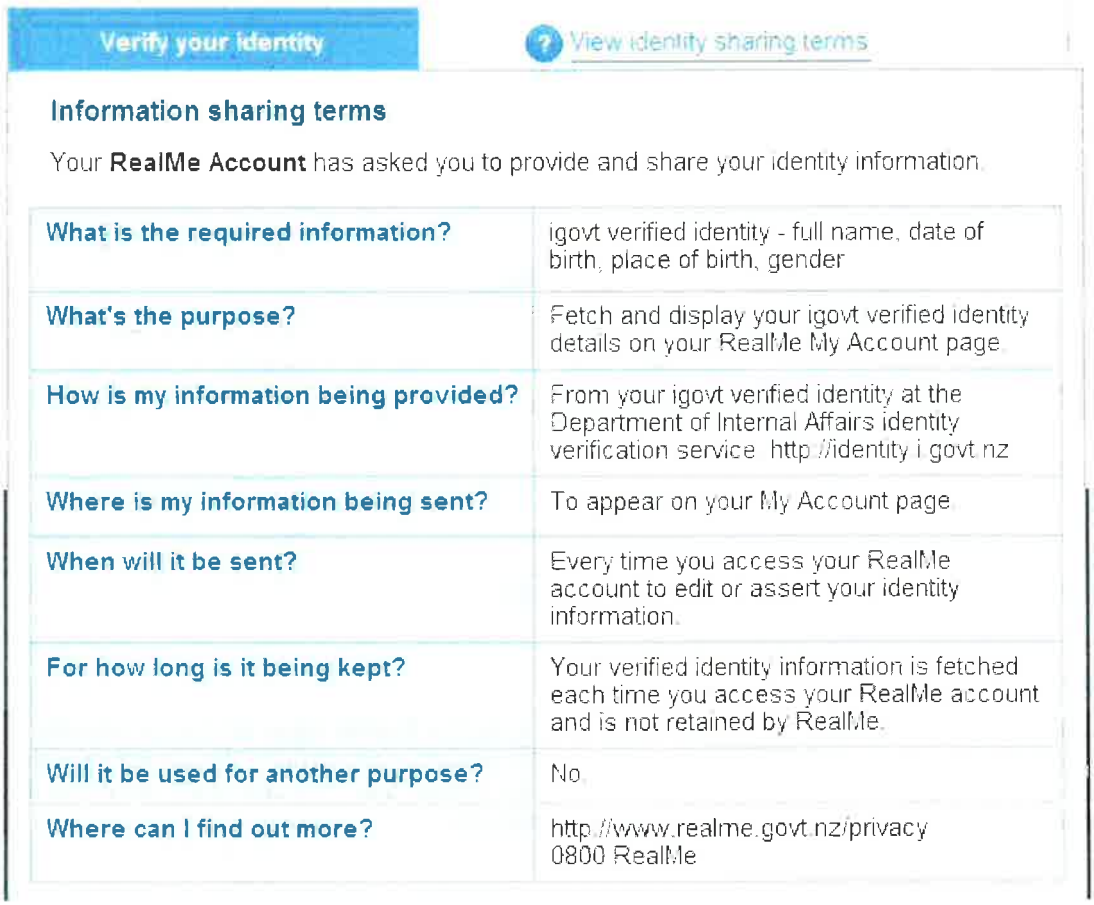
2 Consent Service Use Cases

RealMe's implementation of a shared consent service is a pragmatic solution that aligns with the relevant privacy principles and demonstrates to individuals a significant level of participation and openness in transactions the involves sharing of their personal information . The consent service has a centralised repository that will contain:

- Consent terms for sharing of personal information – The RealMe implementation of Kantara standard information sharing labels. The consent terms will be published into the repository through as part of the integration process with the service provider(s).
- Current and historical consents of an individual – RealMe provides a user interface for an individual to view their current and historical consents.

2.1 Consent terms sharing label

The business process or transaction at the service provider requires personal information from one or more other service providers. The service provider is responsible for displaying the consent terms sharing label and collecting the user consent. The consent terms sharing label is an implementation of the Kantara standard for information sharing labels. See below Figure 1:



Verify your identity

View identity sharing terms

Information sharing terms

Your **RealMe Account** has asked you to provide and share your identity information.

What is the required information?	igovt verified identity - full name, date of birth, place of birth, gender
What's the purpose?	Fetch and display your igovt verified identity details on your RealMe My Account page
How is my information being provided?	From your igovt verified identity at the Department of Internal Affairs identity verification service http://identity.i.govt.nz
Where is my information being sent?	To appear on your My Account page
When will it be sent?	Every time you access your RealMe account to edit or assert your identity information.
For how long is it being kept?	Your verified identity information is fetched each time you access your RealMe account and is not retained by RealMe.
Will it be used for another purpose?	No
Where can I find out more?	http://www.realme.govt.nz/privacy 0800 RealMe

Figure 1 – Consent Terms sharing label

The service providers must provide (or publish) the sharing terms to the consent service's repository. It's clearly beneficial to an individual if they can view the sharing terms for the transactions at the time of consent.

2.2 Use Cases

The following use cases represent the anticipated use of the consent service and are listed in order from information sharing involving one to many service providers.

2.2.1 Consent for single service provider transaction

This use case could cater for an individual providing self-asserted personal information collected by the service where the service provider perceives value in utilising the third-party consent service. Under most conditions, the service provider would typically store the consent within the online service, but this use case is useful for demonstrating the simplest consent service scenario.

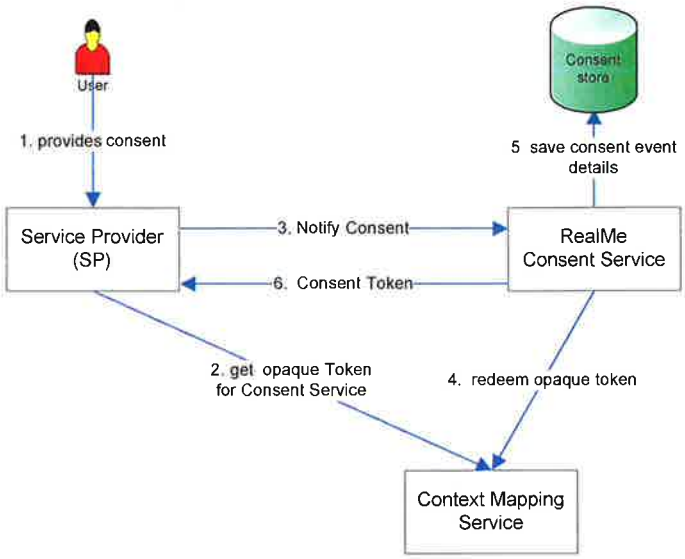


Figure 2 – Consent for single service provider transaction

The steps in Figure 2 are summarised below:

1. The user is authenticated by the service provider using the RealMe logon service (previously known as igovt logon service). The user initiates the business transaction. The service provider displays the consent terms label before initiating the transaction. The user reads the consent terms label and gives explicit consent to the transaction.

2. The service provider requests the context mapping service for the user token that can be passed to the consent service. The context mapping service issues an opaque token to the service provider.
3. The service provider notifies the consent event details to the consent service. The notification request also contains the opaque token.
4. The consent service requests the context mapping service to validate the opaque token. The context mapping service issues a validated token which will contain the user's identifier for the consent service ($FLT_{consent}$) and user's authentication statement.
5. The consent service saves consent event details, with user identifier ($FLT_{consent}$) in its repository.
6. The consent service issues a consent token to the service provider if the consent event details are successfully saved to the repository.

2.2.2 Consent for a transaction involving two service providers

This use case caters for the situation where the initial service provider's online service needs to get or send personal identity data to or from a second service provider – probably during the user's browser session at the initial online service. In this scenario the user's consent might be checked by the second service provider before exchanging the personal information.

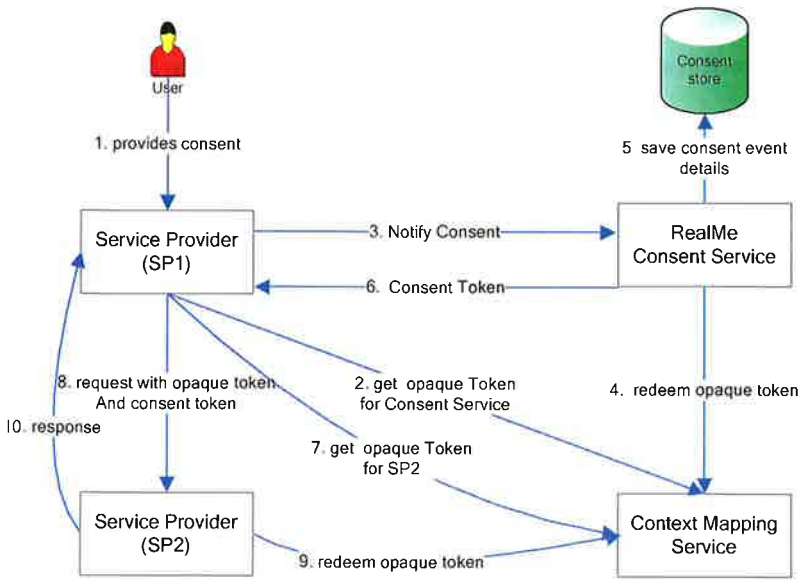


Figure 3 – consent for a transaction involving two service providers

The steps in Figure 3 are summarised below:

1. The user is authenticated by the first service provider (SP1) using the RealMe logon service. The user initiates a business transaction. SP1 displays the consent terms label before initiating

the transaction which involves getting or sending personal information from or to the other service provider (SP2). The user reads the consent terms label and gives explicit consent to the transaction.

2. SP1 requests the context mapping service for the user token that can be passed to the consent service. The context mapping service issues an opaque token to SP1.
3. SP1 notifies the consent event details to the consent service. The notification request also contains the opaque token.
4. The consent service requests the context mapping service to validate the opaque token. The context mapping service issues a validated token which will contain the user's identifier for the consent service ($FLT_{consent}$) and user's authentication statement.
5. The consent service saves consent event details, with user identifier ($FLT_{consent}$) in its repository.
6. The consent service issues consent token to SP1 if the consent event details are successfully saved to the repository.
7. SP1 requests the context mapping service for the user token that can be passed to SP2. The context mapping service issues opaque token to the SP1.
8. SP1 requests or sends the personal information to SP2. The request also contains the opaque token and the consent token.
9. SP2 requests the context mapping service to validate the token. The context mapping service issues a validated token which contains the user's identifier (FLT_{SP2}) and user's authentication statement.
10. SP2 verifies the consent token, processes the request and issues a response to the SP1 if the consent token is valid.

2.2.3 Long expiry period consent

This use case caters for the situation where the initial service provider's online service needs to share personal information with the second service provider in user transactions that can happen over a period of time. The initial service provider captures the user consent only once and the consent has long expiry period.

The long expiry consent is sufficient for the initial service provider to share personal information with the second service provider without recapturing the user consent for next user transactions.

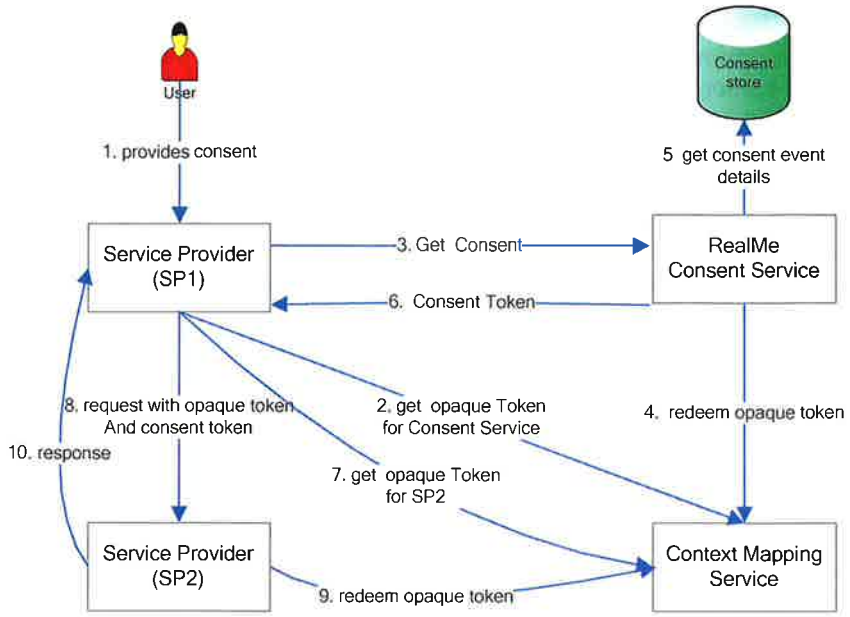


Figure 4 – Long expiry period consent

The steps in Figure 4 are summarised below:

1. The user is authenticated by the first service provider (SP1) using RealMe logon service. The user has already given explicit consent to SP1 in previous transaction for sharing personal information with the second service provider (SP2).
2. SP1 requests the context mapping service for the user token that can be passed to the consent service. The context mapping service issues an opaque token to the SP1.
3. SP1 requests the consent service to verify the consent event details that were captured in previous transaction. The request also contains the opaque token.
4. The consent service requests the context mapping service to validate the token. The context mapping service issues a validated token which will contain the user's identifier with the consent service ($FLT_{consent}$) and user's authentication statement.
5. The consent service retrieves consent event details based on user identifier ($FLT_{consent}$) from its repository.
6. The consent service issues a consent token to SP1 if the previous consent event is valid and not expired.
7. SP1 requests the context mapping service for the user token that can be passed to the SP2.
8. SP1 either requests or sends the personal information request to SP2. The request also contains the opaque token and the consent token.
9. SP2 requests the context mapping service to validate the token. The context mapping service issues a validated token which will contain the user's identifier (FLT_{SP2}) and user's authentication statement.
10. Sp2 verifies the consent token, processes the request and issues a response to SP1 if the consent token is valid.

2.2.4 Consent for a transaction involving a broker

This use case caters for the scenario where a service provider can request a broker (in this case the RealMe assertion service) to obtain personal information from multiple identity attribute providers. In this scenario, the user provides "integration consent" at the RealMe "dashboard" to retrieve personal information from each identity attribute provider. The user will later give "release consent" at the Realme assertion service to provide the personal information to the service provider.

The initial identity attribute providers are the identity verification service (IVS) and the address verification service (AVS). Future identity attributes could include tax number, bank account number, car registration, etc. In most instances, the IAP will be the authoritative source of the attribute type, although data could also be sourced from non-verified or self-asserted sources

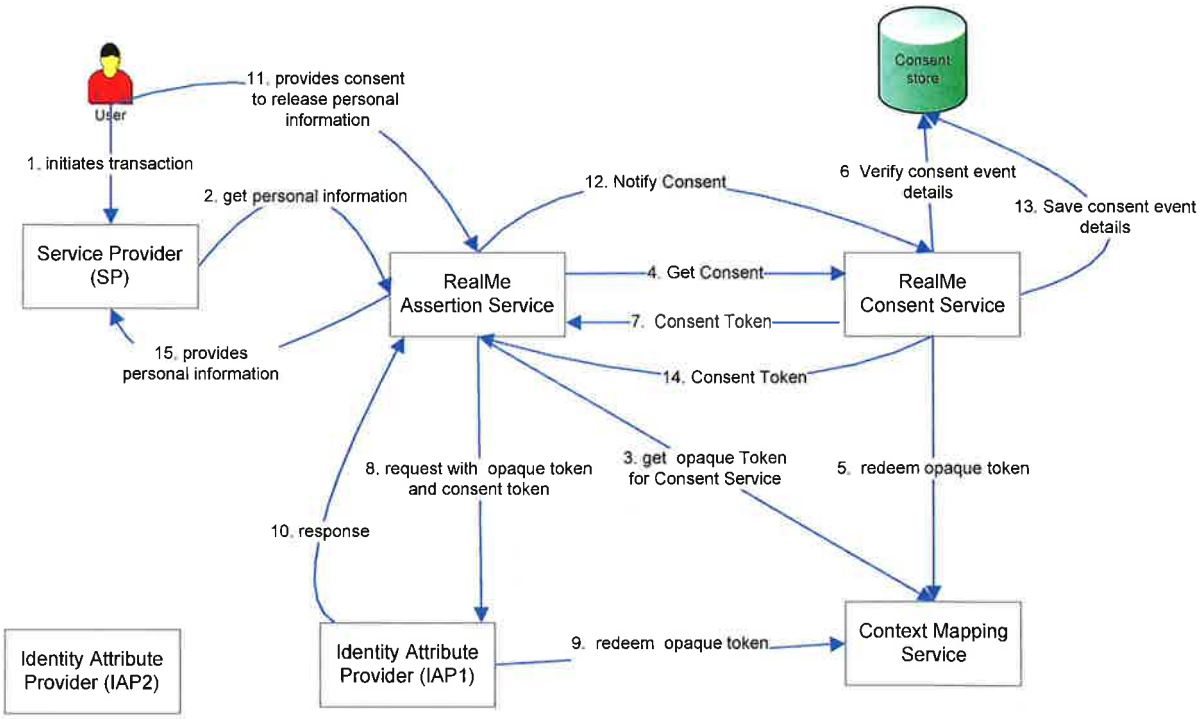


Figure 5 – consent for transactions involving broker

The steps in Figure 5 are summarised below:

1. The user initiates a transaction at the service provider that requires personal information from the identity attribute provider(s).
2. The service provider redirects the user to the RealMe assertion service. The user is authenticated by the RealMe assertion service using the RealMe logon service. The user has already given integration consent to RealMe in a previous transaction to allow the RealMe assertion service to retrieve personal information from the identity attribute provider.
3. The RealMe assertion service requests the context mapping service for the user token that can be passed to the consent service. The context mapping service issues an opaque token to the RealMe assertion service.
4. The RealMe assertion service requests the consent service to verify the integration consent event details that were captured in a previous transaction. The request also contains the opaque token.

5. The consent service requests the context mapping service to validate the token. The context mapping service issues a validated token which will contain the user's identifier for the consent service ($FLT_{consent}$) and user's authentication statement.
6. The consent service retrieves the integration consent event details based on user identifier ($FLT_{consent}$) from its repository.
7. The consent service issues a consent token to the RealMe assertion service if the integration consent event details are valid and not expired.
8. The RealMe assertion service requests the context mapping service for the user token that can be passed to the identity attribute provider. The RealMe assertion service retrieves the personal information request from the identity attribute provider. The request also contains the opaque token and the consent token.
9. The identity attribute provider requests the context mapping service to validate the token. The context mapping service issues a validated token which will contain the user's identifier (FLT_{IAP}) and user's authentication statement.
10. The identity attribute provider verifies the consent token, processes the request and issues a response to the RealMe assertion service if the consent token is valid.
11. The RealMe assertion service displays the personal information to the user. The user gives release consent to the assertion service to provide the personal information to the service provider.
12. The RealMe assertion service notifies the release consent event details to the consent service. The notification request also contains the opaque token.
13. The consent service requests the context mapping service to validate the opaque token. The context mapping service issues a validated token which will contain the user's identifier for the consent service ($FLT_{consent}$) and user's authentication statement. The consent service saves the release consent event details, with the user identifier ($FLT_{consent}$) in its repository.
14. The consent service issues a consent token to the RealMe assertion service if the release consent event details are successfully saved to the repository.
15. The RealMe assertion service provides the user personal information to the service provider.

3 Consent Service Design

3.1 Conceptual Data Model

The following Figure 6 depicts the consent service conceptual data model:

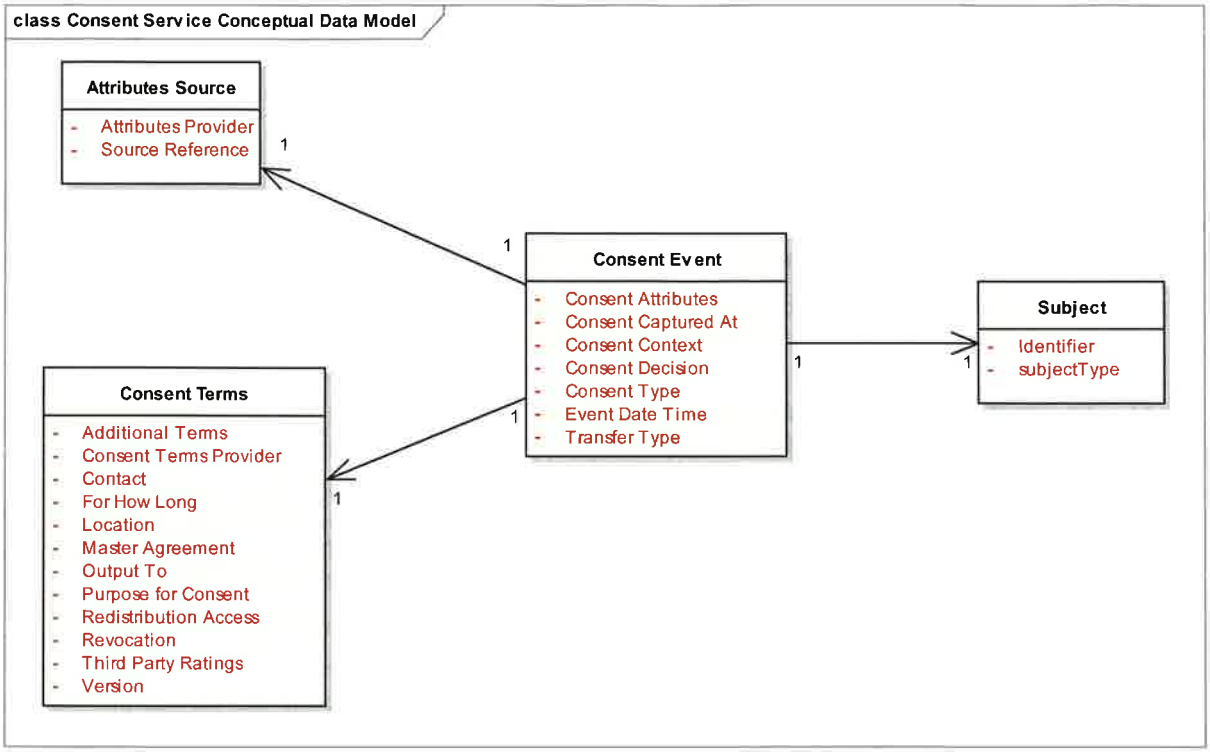


Figure 6 – Consent Service Conceptual Data Model

3.2 Consent Service Interface

A consent service will be implemented as part of RealMe solution which can be utilised in an all-of-government solution for user consent management. The consent service can be extended to work outside the RealMe assertion service context. Because it is a coarse grained web service, the data model is generic and can support various consent contexts. The consent web service has following functions:

- **Notify Consent -** The initial service provider captures the user consent for the transaction(s) which involve sharing of personal information with other service providers. The initial service provider notifies the consent event to the consent service. The notification request contains:
    - Consent Attributes – the personal attributes which will be shared between the service providers

- Consent Context – the business process or context where the user consent is required
- Event Date Time – the date and time of the user consent
- Consent Decision – the user consent decision eg : Accept, Declined etc
- Consent Type – single transactional, long expiry period consent etc
- Consent Terms – the sharing agreement between the user and service provider(s).
- Attributes Transfer Type – Notify or Get Attributes

The consent service saves consent event details in its repository and issues the consent token to the initial service provider. The initial service provider includes the consent token in the request to other service providers.

- **Get Consent:** The consent service issues a consent token to the initial service provider if the user consent details already exist and in valid state. The initial service provider includes the consent token in the request to other service providers.

3.3 Consent Token

The consent service issues the consent token to the service provider on notify or get consent request.

The consent token is a string which consists of seven attributes of a consent event in name-value pair format, delimited by ampersand (&) character and the signature value of seven attribute-value pairs. The following is the structure for the consent token:

CosentType=*value1*&***ConsentAttributes***=*value2*&***ConsentEventDate***=*value3*&***ConsentDecision***=*value4*&***ConsentCapturedAt***=*value5*&***TokenIssueDate***=*value6*&***TokenExpiryDate***=*value7*&***Signature***=*value8*

The following table describes consent token elements:

Consent Token Element	Description
ConsentType	The type of consent event and the possible values are: 1. Integration Consent 2. Single Transactional Consent (i.e. Release Consent) 3. Multiple Transactional Consent
ConsentAttrubutes	A string value to represent the set of attributes that the user has given consent to.
ConsentEventDate	the user consent is captured at that date and time
ConsentDecision	A string value to represent the user's decision for consent event.
ConsentCapturedAt	The name of the service or application where the consent is captured at.
TokenIssuanceDate	The issued date and time of the consent token.

Consent Token Element	Description
TokenExpiryDate	The expiry date and time of the consent token.
Signature	The signature value of the first seven parts (i.e. attribute name-value pairs) of consent token. The consent service signs the token with its private key and appends signature value to the consent token.

Table 1 – Consent Token elements

The initial service provider passes the consent token in the request to the other service providers. The second service provider retrieves the consent token from the request and performs the following validations:

- Signature validation: verify signature value of consent token against remaining part of consent token using consent service's public key.
- The expiry date of the token MUST be with in the limits of current time.
- Verify the consent event attribute values, SHOULD match with the request context.