Identity Anti-patterns and the Identity Bus

This topic explores the problem of identity anti-patterns and the solution provided by the WSO2 Identity Server in the form of the identity bus.

Spaghetti identity and federation silos

According to research conducted by the analyst firm, Quocirca, many businesses now have more external users than internal ones. Also many businesses are involved in acquisitions, mergers and partnerships, and these result in lots of external users joining the business. Often, these external users have to be integrated with the existing user base in bulk. This sort of integration can be complicated due to the different protocols followed by different identity systems.

How this impacts enterprise identity management

When working with both external and internal users and merging different systems together, you are faced with technicalities related to management of multiple heterogeneous user stores, different types of authentication protocols, legacy systems and many more.

SAML, OpenID Connect, and WS-Federation all support identity federation and cross domain authentication. However, in a real world scenario, not all involved parties in a federation use case will support SAML and OpenID Connect. Most of the federation systems seen today are in silos. It can be a silo of SAML federation or a silo of OpenID Connect federation. While a system that uses SAML as it’s protocol may be able to communicate with other SAML protocol-based systems, they may not be able to communicate with OpenID Connect.

The diagram below illustrates how a silo of SAML federation and OpenID Connect federation interact within the respective silo and how it cannot interact with different silos.

Also consider the scalability of a specific federation silo. Within the SAML federation silo, for example, there can be an increasing number of service providers and identity providers. Each service provider has to trust each identity provider and this leads to the Spaghetti Identity anti-pattern. The following diagram depicts the complexity of this.
The identity bus

Federation Silos and Spaghetti Identity are two anti-patterns directly addressed by the Identity Bus pattern in the WSO2 Identity Server. With Identity Bus, a given service provider is not coupled to a given identity provider and also not coupled to a given federation protocol. A user should be able to log into a service provider that accepts only SAML 2.0 tokens with an identity provider that issues only OpenID Connect tokens. The Identity Bus acts as a middle-man, or an Identity Broker, that mediates and transforms identity tokens between heterogeneous identity protocols.

The following are some of the benefits of using the Identity Bus pattern.

1. **Introducing a new service provider is extremely easy.** You only need to register the service provider in the Identity Bus and pick which identity providers it trusts. It is not necessary to add the service provider configuration to each and every identity provider. Similarly, removing an existing service provider is extremely easy. You only need to remove the service provider from the Identity Bus. It is not necessary to remove the service provider from each and every identity provider.

2. **Introducing a new identity provider is extremely easy.** You only need to register the identity provider in the Identity Bus. It will be available for any service provider. Similarly, removing an existing identity provider is extremely easy. You only need to remove the identity provider from the Identity Bus.

3. **Enforcing new authentication protocols is extremely easy.** Consider a scenario where you need to authenticate users with both the username and password and also Duo Security (SMS based authentication). To do this, you only need to add that capability to the Identity Bus and pick the required set of authentication protocols against a given service provider at the time of service provider registration. In the Identity Bus, each service provider can be configured based on how it authenticates users.

4. **Claim transformations.** Your service provider may read user's email address from the "http://sp1.org/claims/email" attribute id - but the identity provider of the user may send it as "http://idp1.org/claims/email". Identity bus can transform the claims it receives from the identity provider to the format expected by the service provider.

5. **Role mapping.** Your service provider needs to authorize users once they are logged in. What the user can do at the identity provider is different from what the same user can do at the service provider. User's roles from the identity provider define what he can do at the identity provider. Service provider's roles define the things a user can do at the service provider. Identity bus is capable of mapping identity provider's roles to the service provider's roles. For example a user may bring idp-admin role from his identity provider - in a SAML response - then the identity bus will find the mapped service provider role corresponding to this, say sp-admin, and will add that into the SAML response returning back to the service provider from the identity bus.

6. **Just-in-time provisioning.** Since identity bus is at the middle of all identity transactions - it can provision all external user identities to an internal user store.
7. **Centralized monitoring and auditing.**
8. **Introducing a new federation protocol needs minimal changes.** If you have a service provider or an identity provider, which supports a proprietary federation protocol, then you only need to add that capability to the identity bus. No need to implement it at each and every identity provider or service provider.