

# Federation

## Ubilogin Whitepaper

---



Copyright © Ubisecure Solutions, Inc., All rights reserved.

|           |  |           |
|-----------|--|-----------|
| <b>1.</b> | <b>Introduction</b> .....                                      | <b>3</b>  |
| <b>2.</b> | <b>Federation</b> .....  | <b>4</b>  |
| 2.1.      | The term – Federation .....                                    | 4         |
| 2.2.      | Identity Providers (IDP).....                                  | 5         |
| 2.3.      | Service Providers.....   | 5         |
| 2.4.      | Provider roles .....   | 5         |
| 2.5.      | Domains .....  | 6         |
| 2.6.      | Trust .....  | 6         |
| <b>3.</b> | <b>Identity</b> .....  | <b>8</b>  |
| 3.1.      | Static Identity information.....                               | 8         |
| 3.2.      | Dynamic Identity information.....                              | 8         |
| <b>4.</b> | <b>Federation use cases</b> .....                              | <b>9</b>  |
| 4.1.      | B2G federation – transient type of federation.....             | 9         |
|           | <i>Transient Federation in real life</i> .....                 | 10        |
| 4.2.      | Account mapping – B2B, B2C federation.....                     | 10        |
| 4.3.      | Account linking – B2C, B2B federation .....                    | 11        |
| 4.4.      | Account linking and account mapping in real life.....          | 12        |
|           | <i>Creating value chains in B2C</i> .....                      | 12        |
|           | <i>Partnership federation between business partners</i> .....  | 13        |
| 4.5.      | Attribute federation .....                                     | 13        |
|           | <i>Attribute federation in real life, B2B federation</i> ..... | 14        |
| 4.6.      | Combined federation .....                                      | 14        |
| <b>5.</b> | <b>Conclusion</b> .....  | <b>15</b> |
| <b>6.</b> | <b>Contact Information</b> .....                               | <b>16</b> |

# 1. Introduction

The Internet has become a place of business. Majority of even the most traditional services are moving on-line. There are thousands and thousands of services available for us in the Internet. Business ecosystems grow as companies are tied to each other as a provider in a value chain or a node in a mesh of businesses co-operating. We just need to login and start using these services and continue develop our business contacts and create new links between existing and new companies.

But, when we accumulate dozens and dozens of services for our disposal, we run into problems with the multiple accounts across the services. Some of these services that we use, are temporary in nature - we need them only once or twice and never again. Some services, such as on-line banking, is used on a daily bases.

As time goes by, the identity information we leave behind grows. There are identities out there that we don't use any longer, accounts that we don't need, etc. Different services store different information about us, and usually this information is useless in the next service. This is not just inefficient use of resources, such as administrative work and disk space, but also a security and privacy risk. We may have forgotten to close our account in a service that we used to order DVDs from, and our credit card information just sits there waiting to be (mis)used. Or we may have had an account in our business partners' extranet service, but as we are now employed by another company, perhaps a competitor, we shouldn't have any kind of access to those services.

Instead of this multiplying security problem and hassle, we should be able to transfer our identity information from a handful of services to other services on a need-to-know (or need-to-do) basis. This identity information should never be stored anywhere else, but at the (one) location we want and we should have to maintain our electronic identities only in these one or very few identity services. For this scenario and purpose, federation standards have been created. Another clear need is to link our existing identity information so that transitions from one service to another goes smoothly.

This paper looks at federation basics, such as terminology and outlines a few basic use cases for federation. Federation technologies are developing all the time. Currently the latest federation standard is based on Security Assertion Markup Language v2.0 specification (SAML 2.0) and WS-Federation 1.1. This document is not a technical description of the specification, but more like a tutorial on federation and how it could benefit Your organization.

Ubilogin SSO is a full federation suite which supports multiple federation standards:

- OASIS SAML 2.0
- WS-Federation
- Liberty Alliance ID-WSF 2.0

## 2. Federation

Federation is one of the most interesting technologies currently emerging in the field of identity management. Aside from terminology federation is a collection of concepts, technologies and interacting entities. In order to describe federation, let us first start with a few definitions on what it exactly does, or tries to achieve.

### 2.1. The term – Federation

The term federation has a wide variety of meanings. The essential meaning of the ‘federate’ term stems from identity information transfer between two different entities / domains.

*”The agreements, standards and technologies that make identity and entitlements portable”*

- Identity federation according to Burton Group

*“A principal's identity is said to be federated between a set of Providers when there is an agreement between the providers on a set of identifiers and/or attributes to use to refer to the Principal.”*

- Federated identity according to OASIS (SAML glossary)

One noticeable common fact of these two definitions is the acknowledgement of an Agreement. There are technical standards that are designed to help software companies implement federation solutions, but for the end user organization this technology is a vehicle which is based on an Agreement between organizations. Federation relationships are based on agreements between two or more entities. An agreement may be legal or contractual in nature or based on the requirements that the technology poses. The agreement may be a mutual contract in which the parties agree that identity information can and will be exchanged, or in more technical terms, which are the identity attributes that are exchanged, or how identity information will be mapped between these two parties.

Although these both are good explanations for the term *federation*, they still need a bit of refinement. Therefore, by extending these two and perhaps combining their meaning, we’ve come to the conclusion that the term *federation* actually means:

*”Federation is a transfer of user identity between two separate domains. A domain is a self contained system that maintains a repository of identity information about its users”*

- Federation according to Ubisecure

The glossary in the SAML standard mentions Providers as entities in action when a federated identity is created. A provider is an essential term to understand when speaking of federation. A Provider is simply an entity which provides services. There’s no distinction what kind of services are offered to the end user (identity). In federation there are however Providers that can be defined.

## 2.2. Identity Providers (IDP)

An identity provider (IDP) is a service that hosts and/or provides identity information to other services. When the IDP hosts identity information, it usually means that the identity information is stored to a database that can be accessed by the IDP. When the IDP hosts the identity information, it normally means that the IDP or another application will handle the actual identity management. The IDP may be connected to a central repository of identity information or the identity information can be scattered to several different repositories, meaning that the IDP acts as a central hub for identity information, acting as a meta-directory.

## 2.3. Service Providers

Another important entity in federation is a Service Provider (SP). The SP is responsible for offering the services, such as on-line applications, to the end users. Most SPs are application servers or applications that are running in the application servers. The SP must support SAML / WS-Federation for the federation to work. The nature of the service as such is irrelevant, the most important thing is that the service requires user authentication.

The SAML specifications use terms such as *Assertion Producer* and *Assertion Consumer*. These terms can be tied to the providers, where the identity provider is usually the assertion producer and the service provider is the assertion consumer. But in federation between two IDPs, the trusting party is the assertion consumer and the trusted party is the assertion producer. Respectively, using the terminology of WS-Federation, *Account Partner* and *Resource Partner* are typically used in a similar context.

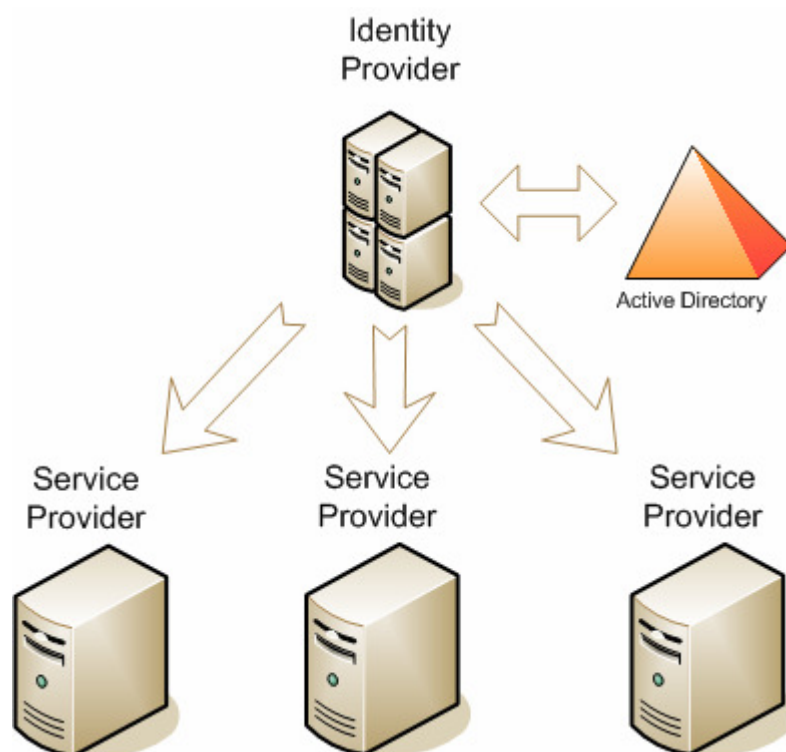


Figure 1 IDP providing identity services to Service Providers

## 2.4. Provider roles

In SAML federation, the entity that sends the identity information is typically an Identity Provider (IDP). The receiving party normally assumes the role of a Service Provider (SP). When federation happens between two domains that are controlled by independent identity providers,

the other one will always act in the Service Provider role, or has functionality that enables it to act as a service provider. The service provider functionality is usually embedded to the SAML capable identity provider product, which makes it possible to receive SAML messages from a trusted identity provider, but this is highly dependent on how the IDP product has been constructed either through functions (code) or licenses. UbiLogin IDP includes both the IDP and SP functionalities in one package.

## 2.5. Domains

A domain is a concept that is important to understand when defining federation. What is usually meant by federation, is that it is an identity transfer between two domains. Therefore it is necessary to outline what a domain actually is from a federation point-of-view.

One definition of a domain, is that it is a self contained system that uses and controls a repository of identity information about its users.

What this means, is that there is usually an identity provider within one domain. This identity provider provides user authentication and possible authorization. A good example of a domain is the Windows Domain concept. A Windows Domain uses Active Directory, usually, to store user identities and it provides authentication services to the clients (workstations and servers).

The Windows domain is a good example in another sense as well. The normal Windows domain is not federation capable alone as such. It requires a dedicated Identity Provider that can interface with other Identity Providers or outside Service Providers. But if you install the UbiLogin Authentication Server to the Windows Domain, you can start federating user identities to other domains.

## 2.6. Trust

One of the key concepts in identity federation is trust. When identities are transferred between domains, the receiving domain should be absolutely sure that it can trust the information that it receives from other domain(s). A trust relationship is not just technology; it's also a process of creating a level of trust through legal agreements and other contractual means that are not based on technology.

Once the trust relationships are formed in the corporate level using agreements, the trust relationship should be formed also technologically. From a SAML point of view the trust relationship is created when the sending domains metadata along with the public key is uploaded to the receiving domains identity provider. Messages that are exchanged are always signed using asymmetric cryptography and this usually means a key pair, public key cryptography. Key pairs can be generated without centralized infrastructure such as PKI, but it would be beneficial to use certificates for message signing in federation when there are several service providers.

Trust in federation has a direction. The direction is opposite to the flow of information, i.e. when Domain A trusts Domain B, the identity information flows from the Domain B to Domain A. So when forming trust relationships with another IDP there are 3 possible trust relationships;

- Domain A trusts Domain B → Identities flow from Domain B to Domain A
- Domain B trusts Domain A → Identities flow from Domain A to Domain B
- Both domains trust each other → Identities flow both ways



**Figure 2 Domain B trusts Domain A, and the Domain B identity provider functions in a Service Provider role**

## 3. Identity

What exactly is our identity in the net? Most of us have dozens of different representations of our electronic identity that we use daily. Those identities are stored in web server identity repositories and we are almost always authenticated with our password. On-line banks utilize perhaps stronger authentication methods to verify our identity. Common for all these services that our identities are verified and after the verification the service knows something about us.

Our electronic identities are basically a collection of attributes that describe us. Different services use different attributes, usually based on business requirements. In order to offer better services for users, on-line services include quite a bit of information about its users. The extra information, not just our user ID, helps these services personalize content for us. The information that is tied to our unique identifier grows in time as we conduct more transactions in the on-line service.

### 3.1. Static Identity information

The information that is used to authenticate us must remain as static as possible. If your identifiers that are used to verify your identities are changed constantly, the authentication becomes actually useless if we loose the connection between the digital identity and the real life person. Static identity attributes are e.g. our user ID, the key pair of our certificate (which should not change even if the certificate is re-issued), our common name (CN). Some of the static attributes are however linked to our own external environment such as employment. If our external environment changes, some of our static identity attributes may change as well.

In federation, static identity attributes are very useful when transferring identity information across domains. This makes it easier to map or link identities between two different domains. When the information stays the same, the process of transferring the user session from one domain to another can be accomplished with simple rules and the IDP configurations doesn't need to change constantly.

### 3.2. Dynamic Identity information

Most of the information that is linked to us and our identities changes over time. New attributes are added; older ones are deleted or changed and so on. This dynamic information grows when time passes and gets distributed across the Internet when we use Internet based services. We may have several different accounts with several different service providers. According to their needs, the service providers collect information about us that is useful for their purposes. We may have several accounts, where e.g. shipping information (address) is stored. If we move, we must change or request change for this data.

As we continue to use Internet based services and the amount of our dynamic identity information grows, it becomes difficult to maintain for us and the service providers. If we can't consolidate our identity information and transfer relevant attributes from one service to another, the important bits of information can get lost in the sea of trivial identity information. It becomes also very difficult to maintain the life-cycles of the accounts that we have. Most Internet users have obsolete accounts in services that they do not use any more. This requires additional administration efforts from the service providers and we may have sensitive information about our identities lying somewhere, that we have forgotten or can not erase.

## 4. Federation use cases

The basic use case in federation is simply transferring identity data across two independent domains. But there's much more to consider when we think how we can actually achieve this using standards based technology solutions.

### 4.1. B2G federation – transient type of federation

When two independent domains and identity providers trust each other and support the same standards, such as SAML 2.0, federation can be accomplished. The key in transient federation is trust. Trust is of course an issue in all federation use cases, but essential in transient federation.

In transient federation only the user session is transferred from one domain to the other. No additional identity data is transferred, or received by the assertion consumer, i.e. the receiving IDP. When there is no additional user information available, the actual authorization happens in the sending IDP, the receiving IDP just blindly trusts the information sent by the initiating IDP.

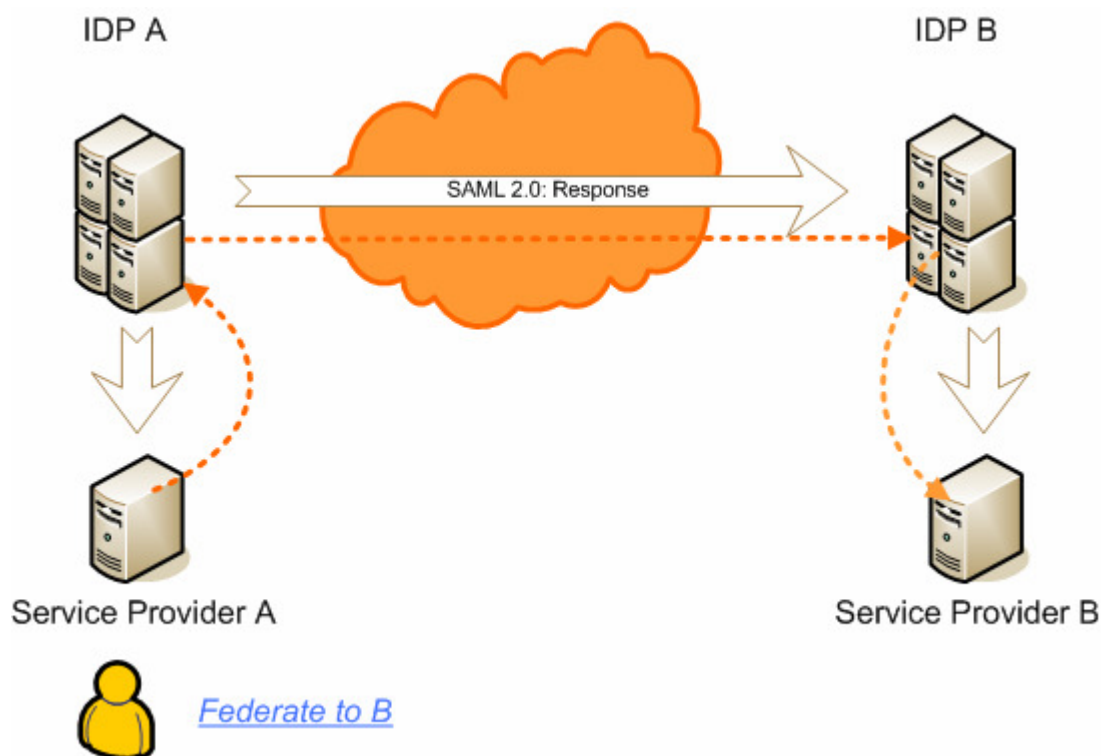


Figure 1. Transient federation

Transient type of federation is one of the simplest ways to implement federation. Thanks to the simplicity, the configuration efforts in IDP A and IDP B are easy, and once accomplished don't have to change. In transient federation the IDP B accepts all incoming response messages (federation requests) from the IDP A. Therefore the IDP B relies on the IDP A, which has to implement necessary access control mechanisms for federated users.

The major benefit of transient federation is its ease of use. The users don't have to know that they have just been federated. The federation can be presented as a simple link in the Service Provider A, and the by clicking this link, the user is automatically and transparently transferred

to the Service Provider B as an authenticated user. Transient federation is also simple and easy to implement between two IDPs.

### **Transient Federation in real life**

Transient federation is best suited for close partners that have a high level of trust between each other. The services that accept federated users must also be open in nature, and there should not be any company confidential data available through these services. In transient federation the users are just passed through the receiving identity provider and sent to the target service provider without further verification on their identities.

The best real life use case for transient federation comes from government services. Citizens are normally authenticated using strong authentication methods such as certificates in the government eServices. These services do not keep records of identities, and therefore additional identity information is impossible to send when federating the user from one service to another.

Once the user has been authenticated in one domain, the IDP can federate the user to the next domain by sending the unique identity information of the certificate to the next domain. The receiving domain doesn't have to verify this information and the user can access the services with single sign-on.

## **4.2. Account mapping – B2B, B2C federation**

As transient federation sets certain restrictions on trust relationships and information confidentiality, a more secure method should be implemented if the domains do not completely trust each other, or there are some constraints on the information that the federated user can access. The first step to reduce risk on unauthorized access is to create a link between two accounts between the two domains. This requires that the user who tries to federate from Service A to the Service B has an account in both Identity Providers.

From the technical perspective, account mapping is a bit more complicated compared to transient federation, but provides much more control for the receiving Identity Provider. The receiving Identity Provider must be able to receive messages from the sending IDP (assertion producer) that includes the UID element in the response. This UID element is then mapped to an existing identity in the IDP B, and if the UID is found, the user session is transferred and access to Service Provider B granted.

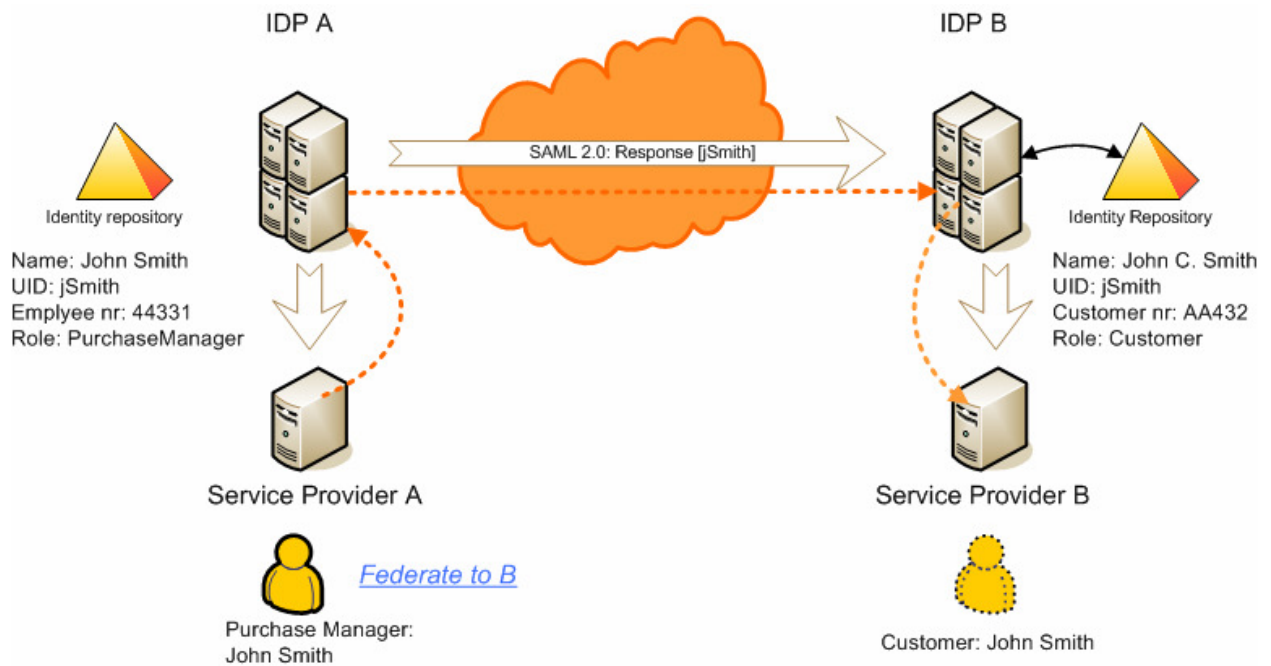


Figure 2. Account mapping. John is federated from A to B based on his UID.

### 4.3. Account linking – B2C, B2B federation

A pure account mapping based federation assumes that the user has accounts with both Identity Providers and there's a common element available for mapping. This may limit the usability of the federation solution and create extra administration tasks in the two organizations that are using federation. Account linking is an extension to the account mapping process, where the existing user account is extended with the IDP A information upon first federation.

Account linking produces an extra step to the account mapping process, where the IDP B requires user authentication when it receives a federation message from the IDP A and does not find the user identity from its own repositories based on the identity attribute it received from IDB A. If the user is successfully authenticated in the IDP B, the extra information that can be used in account mapping in the future, is stored to the IDP A identity.

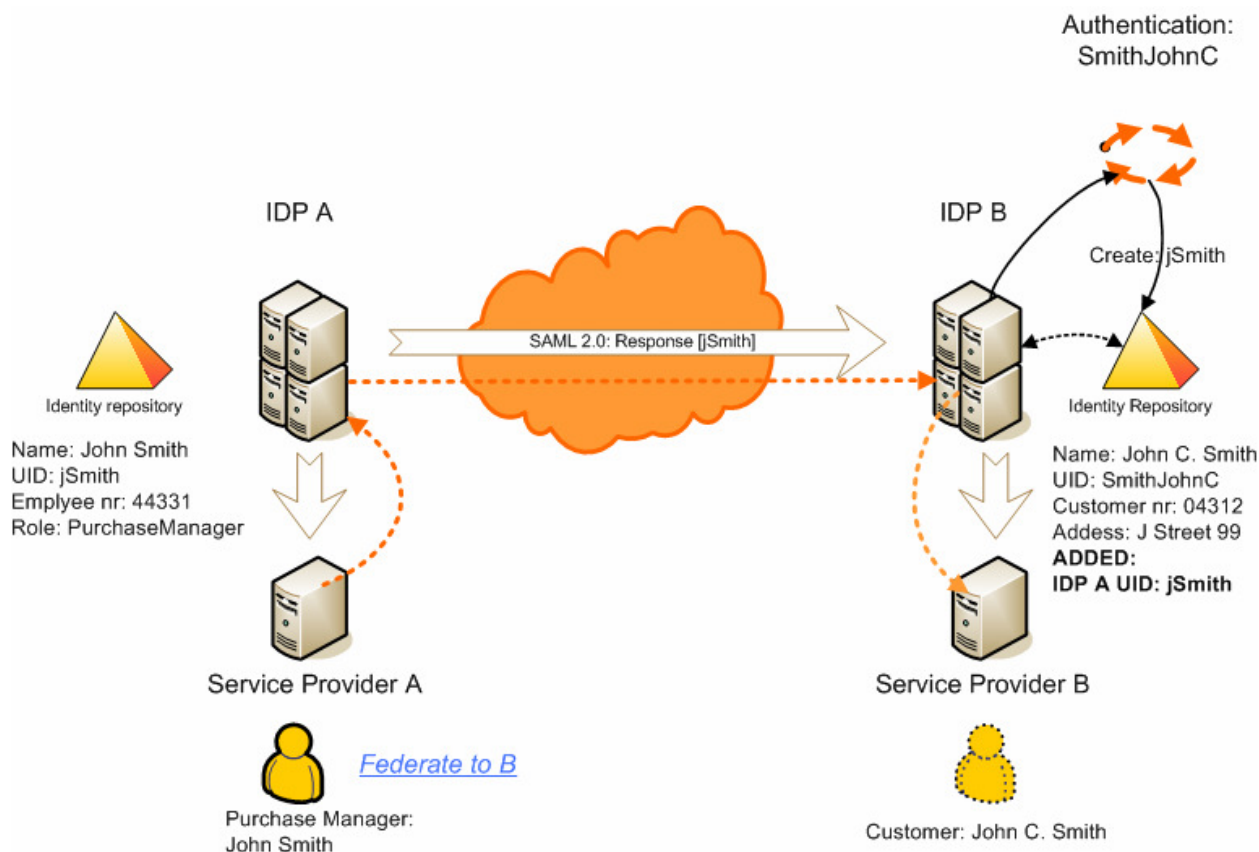


Figure 3. Account linking. John Smith federates to the IDP B and authenticates using his credentials in IDP B and the IDP A UID is stored to his IDP B identity.

If the user does not have an account at all in the IDP B, the account linking can be extended so that the user is able to create an account when he first federates to the Identity Provider. After the account creation in the IDP B, account mapping procedure can be used when the user tries to federate between these two services. Account linking can be seen as an extension to account mapping.

Account linking does not require any administration efforts in the receiving Identity Provider as the account creation process is outsourced to the end user and these accounts and links do not have to be created by the administrators of the receiving identity provider. If account creation is combined to account linking, the end user can automatically create an account in the receiving identity provider domain.

#### 4.4. Account linking and account mapping in real life

In real life account linking and account mapping can be seen as a single use case from the business perspective. Account linking just provides the means to create account mappings in an identity provider.

##### Creating value chains in B2C

When companies are offering services to their customers (B2C) and work in a related field, they may want to offer better customer services to the end users. Single Sign-On is one way of improving customer experience and strengthen the business between partnering companies. When a user can simply transfer to other services provided by the business partner, the user experience and ease of use improves, and customer loyalty and satisfaction can be elevated to a new level.

Account mapping and account linking creates possibilities for companies to create webs of interconnected services through federation. For the end user it means improved services and for the business partners increase in their revenues by combining forces. A very traditional, but a very good example is a user that needs to travel to another country for a business trip. If the airline, the car rental company and a hotel are using federation through account mapping, the user can accomplish all the required tasks with one authentication and the flow of the customer process of acquiring the needed services for his trip can be controlled. If account linking is used, or the extension of account linking with account creation, the end user forms linked accounts between the services that are using federation. This way an airline could bring additional business to the other partners, or vice versa.

**Partnership federation between business partners**

Account mapping and account linking use cases provide excellent tools to integrate business partner processes together through identity federation. A company offering services to a prospective buyer can achieve better customer loyalty and satisfaction when they can federate the buyer identities from the buyer organization and therefore offer seamless transitions from the buyer information systems to the service provider system, with Single Sign-On. This is probably the most common use case scenario in B2B identity federation.

**4.5. Attribute federation**

Sometimes it is not necessary to have user accounts in both Identity Providers. The sending Identity Provider can categorize users based on groups, roles and other attribute information. This information can be used in federation to implicate what privileges the user actually has.

In attribute federation, the receiving Identity Provider must be able to read this attribute information from the response sent by the first Identity Provider. This attribute information should then be mapped to authorization information in the receiving end. The attributes that are delivered have a corresponding role or other type of authoritative information in the receiving IDP and this information can create the access control decision.

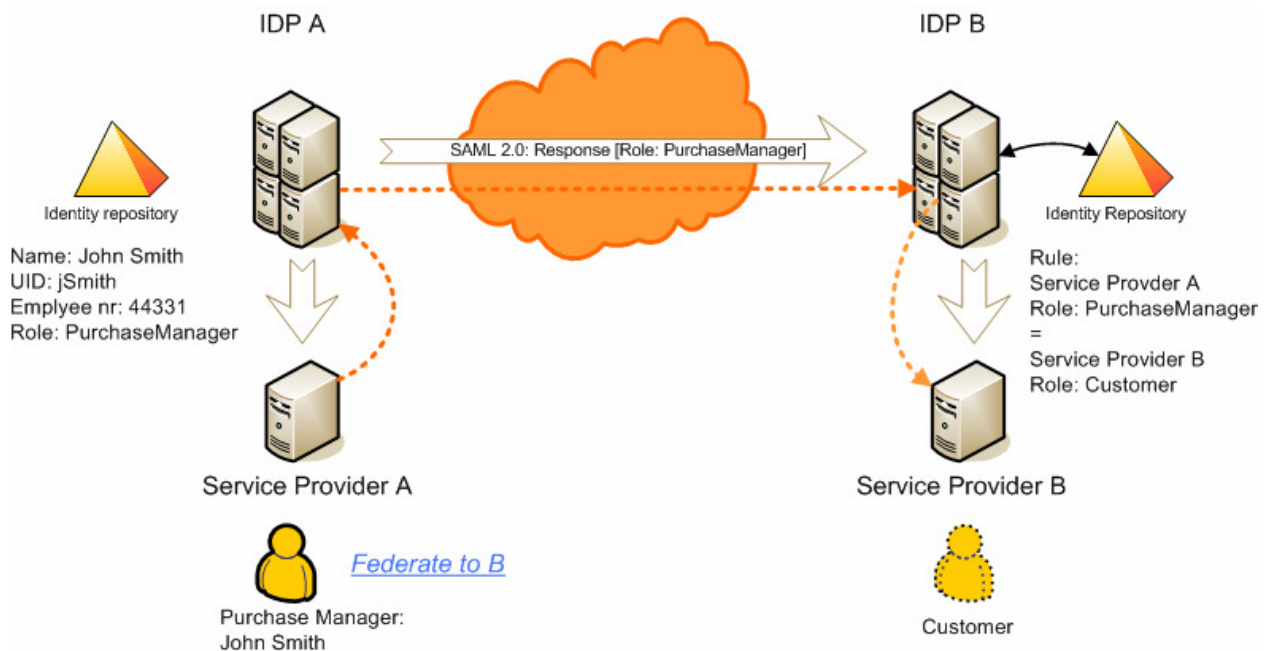


Figure 4. Attribute mapping. John has a PurchaseManager role in the IDP A, which is translated to a role of Customer in the IDP B.

If the user doesn't need a dedicated account in both domains, attribute federation works like transient federation, but improves risk management processes as we can determine the actual privileges of the user through attribute rules. Attribute federation requires a bit more administration of the Identity Providers, but if the attribute sets can be kept constant, this effort is minimal compared to the ease of use and risk management improvements.

### **Attribute federation in real life, B2B federation**

Attribute federation works very well in B2B environments where the service provider can easily determine the user access privileges to various different services through attribute mapping. It's also lighter and easier to implement than account linking or account mapping as the actual account doesn't have to exist in the receiving Identity Provider repository, just the attribute mapping rules. As with transient federation, trust is an issue, as the receiving domain has to completely rely on the information sent by the other domain (or IDP).

Attribute federation can be used to strengthen partnerships between companies, and offer better services to the partners. Through attribute federation partners can access services in an appropriate role and accomplish their business on-line. Role based federation enables partner companies to verify the users access and transaction authorization levels in the services. Additional benefit in attribute federation is that the user does not have to have accounts in both domains.

## **4.6. Combined federation**

Identities can be transferred between domains without content restrictions. The federation request (or "Response" message) can contain any identity attributes available to the sending Identity Provider. The federation request can include UID, name, address, role, group etc information. The freedom to include basically anything in the message makes federation flexible and complicated at the same time.

The freedom to choose the federation attributes help organizations to implement federation procedures that suit their needs. But it also means that creating such federation and trust relationships requires thorough planning and continuous administration. From the user perspective federation is always transparent, unless account linking is required.

## 5. Conclusion

The Ubilogin Authentication Server supports latest standards based federation. The supported use cases for federation allow the implementation of simple transient federation solutions to combined federation use cases.

Federation can serve B2C, B2B and G2C sectors and bring clarity to the electronic identities that we have. It is no longer necessary to have an account in each service that we need to use to do our on-line purchases, reserve a rental car, book a hotel, file our income tax reports, conduct business between partners, study on-line, buy music on-line etc. Depending on the situation, a suitable federation use case can be constructed to serve customers and business partners.

Federation requires agreements and technical planning and administration, but it also improves services, reduces administration costs, and can make our systems more secure. Federation is based on standards, and these standards are still developing. SAML 2.0 and WS-Federation provide a good base from which to move on as new use cases for federation are created. Ubilogin Authentication Server supports SAML 2.0 and WS-Federation standards as well as many other identity and authentication standards. With Ubilogin companies can build long lasting federation solutions and improve their integration options with business partners or build value chains for customers that can enhance loyalty and provide growing revenues by creating webs of value providers.

## 6. Contact Information

Ubisecure Solutions, Inc.

www.ubisecure.com  
info@ubisecure.com  
support@ubisecure.com

<firstname.lastname>@ ubisecure.com

Tekniikantie 14  
FIN-02150 Espoo, FINLAND

tel. +358-9-2517 7250  
fax +358-9-2517 7070

Registered in Espoo, Finland  
reg. nr. FI17487214

### *About Ubisecure*

*Ubisecure Solutions, Inc. is a leading partner in providing advanced authentication and authorization solution for Internet, Intranet and Extranet services. Ubisecure provides application developers, integrators, solution providers and end-user organizations with IT-security software solutions that maximize the competitive advantage of its customers. The Ubisecure product line consists of UbiLogin solutions for authentication and Web Single Sign On access to Internet and Intranet/Extranet services, UbiPass VPN-authentication and UbiSignature electronic signatures. Ubisecure provided authentication utilizes ordinary GSM handsets, challenge-response SMS-messages, one-time passwords in Java-phones, smart cards, Windows Integrated Authentication as well as various third party vendor services and products. Ubisecure has offices in Finland and Sweden.*

**For more information, visit Ubisecure 's web site at [www.ubisecure.com](http://www.ubisecure.com)**

*Ubisecure, UbiLogin, UbiPass, Ubikey and UbiSignature are trademarks and/or registered trademarks of Ubisecure Solutions, Inc. All other companies and products listed herein are trademarks or registered trademarks of their respective holders.*