CERIAS Tech Report 2005-46

INTEGRATING FEDERATED DIGITAL IDENTITY MANAGEMENT AND TRUST NEGOTIATION

by Abhilasha B. Spantzel and Anna C. Squicciarini and Elisa Bertino

Center for Education and Research in
Information Assurance and Security,
Purdue University, West Lafayette, IN 47907-2086

Integrating Federated Digital Identity Management and Trust Negotiation [*]

Abhilasha Bhargav-Spantzel
CERIAS, Purdue University
bhargav@cerias.purdue.edu
Anna C. Squicciarini
Universita degli Studi di Milano
squiccia@dico.unimi.it
Elisa Bertino
CERIAS, Purdue University
bertino@cerias.purdue.edu

Abstract

Most organizations today require the verification of personal information pertaining to users in order to provide service to users. Privacy of such information is of growing concern and because organizations often ask for similar information, this process can also be redundant and inefficient. Recent proposals dealing with federated identity management have the potential to alleviate such problems. A federation is a set of organizations that establish mutual trust with each other. This allows them to share client information whenever possible depending on their service disclosure policies and user privacy preferences. This paper addresses such problem by integrating federated identity management with trust negotiation techniques. We focus on a trust negotiation approach suitable for federated environments. Our federated trust negotiation approach relies on the use of special-purpose tickets, that is, signed assertions that are released by the federation members to users upon successful negotiations. The main advantage of such integration is that if a user has already successfully negotiated with a member of the federation, subsequent negotiations with other federation members may require a reduced number of interactions between the client and the service provider.

[*] The work reported in this paper has been partially sponsored by NSF under the ITR Project 0428554 ”The Design and Use of Digital Identities” and by the sponsors of CERIAS.

1 Introduction

In today's increasing competitive business environment, more and more leading organizations are building web-based infrastructures to gain the strategic advantages of collaborative networking. However, to facilitate collaboration and to fully exploit such infrastructures, organizations need to identify each network user and which resources each user is authorized to access. User identification and access control must however be carried out in way that maximizes user convenience and privacy assurance and at the same time does not increase the operational costs for organizations.

Recent efforts in the area of federated digital identity management are trying to address some of those issues, in particular with respect to user authentication [3, 4, 6]. A federated identity is a digital credential analogous to a country passport. Just the way a passport is issued in one country and is accepted as a valid identification in other countries, such digital credential allows users to access multiple domains with a single, recognized identity. A federation is traditionally a set of organizations which establish trust relationships within which the federated identity information is considered valid. Federations can be classified into different categories, according to different criteria. With respect to identity management issues, it is interesting to distinguish among federations where most of the interactions are internal to members and federations where interactions occur between the federation and third-party users.

With single sign-on users can currently use the same username and password for a seamless access to federated services, within one or multiple organizations. The notion of federated identity should however be extended to include not only user's login names, but also user properties, often referred to as *user attributes*. Such requirement is motivated by the fact that in an increasing number of situations access control policies are based on security-relevant properties of users. Thus authorizations to a given resource are not any longer expressed only in terms of user login ids. Rather, they are expressed in terms of requirements and conditions against user properties. Achieving federated management and single sign-on for credentials containing several user attributes is very promising in both business market and academia. A business market study showed that a saving of more than a million dollar can be achieved by the adoption of federated digital identity and access control management systems [5]. We are however still far from completed solutions to the problem of single sign-on when dealing not only with user's login names but also with user properties.

One problem with current federated identity management systems is the single trusted identity provider, which can be a bottleneck and a single point of failure. If we want to distribute the functionality of the identity provider to different service providers[1] we need a secure and privacy preserving mechanism for retrieving the user attributes from different service providers. We need approaches to give the minimal information about users required to satisfy the requesting service providers' service policy. If not, the privacy of the attributes may be vulnerable as they would reside in multiple locations within a federation some of which might not be trusted by the user. In this respect it is also important to notice that, as shown by a recent survey [1], users have differentiated privacy preferences with respect to the various types of information concerning themselves. For example users may agree to share demographic information with organizations but not credit card or health information. Such requirement calls for a flexible and selective approach to the problem of user attribute sharing in federations.

An approach to address the above problem is to integrate federated identity management with trust negotiation techniques, such as those provided as part of the Trust-χ [2], which is the goal of the work we report in this paper. More specifically, we propose implementing trust negotiation between service providers in a federation, and between users and service providers. This is, to the best of our knowledge, the first attempt to integrate a federated identity management system with a trust negotiation framework. The resulting framework, that we refer to as FAMTN[2], has the key feature that the user does not have to provide a federated attribute[3] more than once to a given federation. Internal users of a FAMTN system will be able to perform negotiations by exploiting their sign-on id without repeating any identity verification process. Further, a FAMTN system supports temporary single-sign on, so that external users can perform different negotiations among the federation taking advantages of the federated framework to reduce the number of information to be exchanged at each process.

The paper elaborates on such an integrated approach. The main contributions of the paper are as follows. We propose an architecture for the main component of the framework which is the service provider. We also specify a trust negotiation approach suitable for federated environments. A key feature of our approach is that it caters to two different types of federation. The first type we refer to is a set of organizations that federate in order to provide some aggregated or complex services to external users. The second type is a set of organizations that need to integrate their own internal purposes. In what follows we refer to the first type as a *coalition* and to the second type as *cooperation*. Thus users of $cooperations$ are internal (or member) users, who need to access resources from the organizations in

[1] We do not differentiate between service providers and organizations in a federation in this paper.

[2] Federated Attribute Management and Trust Negotiation.

[3] Attributes the user is willing to share in a federation are called federated attributes.

the federations.
Our federated trust negotiation approach relies on the use of special-purpose tickets, that is, signed assertions that are released by the federation members to users upon successful negotiations. We propose two different types of ticket. The first type, that we refer to as *trust ticket*, encodes the list of federation service providers with which a non-member user has successfully negotiated. The second type, that we refer to as *session ticket*, is used to member users in order to speed up negotiations. We take advantage of the fact that most attributes do not change in a short period of time; thus if a user got a service recently he/she is most likely eligible for the service again. Finally, we extend the XML-based language χ-TNL [2] used in Trust-χ to represent additional privacy options for users accessing services from the federation.

The remainder of the paper is organized as follows. In Section 2 we present related work followed by a general overview of our approach. In Section 4 we describe the architecture of FAMTN framework along with the ticketing system used within FAMTN. In Section 5 we discuss attribute sharing in federations and in Section 6 we present in detail the negotiation algorithm with illustrative examples. In Section 7 we present a preliminary analysis of the FAMTN framework with respect to privacy and efficiency as compared to current federated identity management systems. Finally, in Section 8 we highlight future work and conclude the paper.

2 Related Work

Federated identity management and trust negotiation have both been investigated extensively. The former is currently a business initiative of interest to several companies. In this section we elaborate on the most relevant projects.

In the corporate world there are several emerging standards for identity federation like Liberty Alliance and WS-Federation. Since the projects are very similar we describe the former in more detail below.

Liberty Alliance [3] is based on SAML[4] and provides open standards for single sign-on (SSO) with decentralized authentication. SSO allows a user to sign-on once at a Liberty-enabled site to be seamlessly signed-on when navigating to another Liberty-enabled site without the need to authenticate again. This group of Liberty-enabled sites is a part of what is called a *circle of trust*, which is a federation of service providers and identity providers having business relationships based on Liberty architecture. This approach enables users to transact business in a secure and apparently seamless environment. The identity provider is a Liberty-enabled

[4] Security Assertion Markup Language (SAML).

entity that creates, maintains and manages identity information of users and gives this information to other service providers. The users authenticate themselves to an identity provider in the federation and other service providers obtain authentication information of the user from it. Similarly, FAMTN framework builds on a SSO and, in addition, it provides a flexible decentralized trust management system for registered users.

There may be multiple identity providers in one federation in a Liberty Alliance framework and they could possibly also be service providers. Basically, in a given Liberty circle of trust a user can use multiple identity providers that share his information among them. Trust relationships and access policies between these identity providers are established a priori while forming the *circle of trust* itself. The underlying semantics and related protocols are not dictated by the Liberty protocols. Our belief is that for a truly decentralized identity management we need a more automatic methodology for federating the user information between the identity providers. In the FAMTN framework, indeed, we do not distinguish between service and identity providers: each service provider in the federation can act as an identity provider. The information between service providers is simply exchanged through automatic trust negotiation, in an on-demand dynamic fashion.

Shibboleth [4] is similar to the above project and its goal is to facilitate sharing of resources between institutions. It extends the concept of federating identity information to federating user attributes. When a user at one institution tries to use a resource at another, Shibboleth sends attributes about the user to the remote destination, rather than making the user log in to that destination. The receiver can check whether the attributes satisfy the service providers policy. The identity provider in the Shibboleth architecture has all the user attributes and user privacy preferences which are taken into account when this identity provider gives information to other service providers. We differ with this approach since we do not rely on a central identity provider providing all user attributes. User attributes in our framework are distributed within the different service providers in the federation, each of which can effectively be an identity provider. The ability to negotiate with different service providers adds flexibility to the way a user can define different privacy preferences to different members of the federation which does not exist in Shibboleth. Shibboleth requires trust agreements to define the population, retention, and use of attributes, thus making difficult for external users (who are not affiliated with the federation) to use in an ad hoc fashion the different services offered. In our framework, on the contrary, external users can easily negotiate within the community, due to an ad hoc type of negotiation we have designed.

Concerning the trust negotiation, the trust negotiation system on which the current framework is based is Trust-χ [2], a trust negotiation system specifically conceived for peer to peer environments. Trust-χ is complemented by an ad-hoc

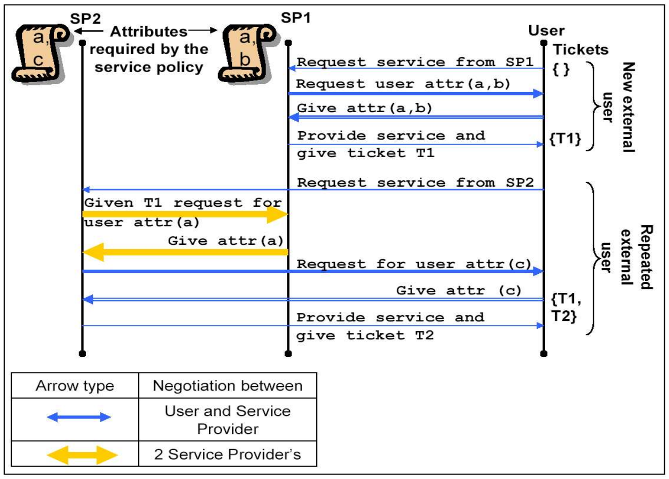

Figure 1: External user negotiating with two service providers of a federation.

XML based language χ-TNL, to encode negotiation policies, digital credentials and security related information. A main difference between Trust-χ and our work is that the negotiation process of FAMTN is much more articulated than the one of Trust-χ and may involve third parties, in addition to the two parties that have initiated the negotiation. We can thus say that FAMTN is characterized by *multi-party* negotiations, whereas Trust-χ only supports two-party negotiations.

3 Overview of the Approach

A federation is a group of organizations which trusts the information from any member of the group to be valid. In this paper we consider an organization to be a service provider. This notion can be easily generalized to organizations consisting of multiple service providers, such that the set of service providers have similar policies and also form a federation.

To be federated means becoming a part of the federation. A FAMTN federation essentially involves two type of entities: FAMTN service providers (FSP) and users. A service provider is an entity providing a service to a user, if the user satisfies the policy requirements of the service. Additionally, FSPs support identity and attributes provisioning, as detailed later in this section.

Users are qualified by means of attributes or credentials. Attributes are logical representation of user properties. In the FAMTN system, we distinguish between two type of attributes. The first type describes an actual value of a user property; the second type describes a conditional property of the actual value. Credentials are digital documents grouping together several attributes according to a pre-defined template. Attributes and credentials are signed by certified authorities and issued to users as certificates. Conditional attributes appear in credentials issued by a service provider, according to its policies, when trust is established the first time. In our work we refer to χ-TNL credential type system [2] to specify the first type of attributes.

Usually, service requirements are expressed in terms of policies requiring user credentials and attributes, so to authenticate valid users.[5] Precisely, service policies specify the credential or attribute requirements which need to be satisfied by the users to gain access to the service. A service policy may be disclosed following different strategies depending on the negotiation process. Each entity interacts with another in the FAMTN system by means of a negotiation protocol. The approach we propose requires two types of negotiation. The first type is between the service provider and the user, and the second is between two service providers in the same federation. Regarding the first type of a negotiation a further distinction is needed. Indeed, the negotiation protocol for negotiations carried out between service providers and users depend, in turn, on the type of the interacting user. Precisely, the distinction is based on the membership of the user with the federation. According to the distinction previously introduced, $cooperations$ are likely to be characterized by negotiations among providers and *member users*. A user is a *member user* of the federation if he/she is affiliated with an organization within the federation. The federation is more likely to have information about a member user even if he/she has not accessed any of its services. This also depends on the policy of the member organization that defines which of its affiliated user attributes are federated. The member will be identified among the federation with a SSO user identification.

On the contrary, $coalitions$ are characterized by negotiations among *external users* and member providers for negotiating aggregated services. External users have to provide all required attributes at their first negotiations. The first negotiation between an external user and a FATMN provider includes identity provisioning, since the provider issues a temporary user-id to be used within the federation. The use of time-limited SSO id for non-members ensures identity linkability even for non-members.[6] Of course, users might have one or multiple identities and choose which one to adopt for requesting access to service. We do not elaborate on multiple identities issues since it goes beyond the scope of this work. By interacting

[5] We currently assume a PKI is in place for basic authentication and key distribution within a federation.

[6] We can reasonably assume that the time interval duration is defined by the federation policy.

further with the federation, the amount of user information disclosed to the federation increases. This information can be linked to the user (who is then called *repeated external user*) and thus reused in the next negotiations. As a result, more efficient negotiations with fewer attributes required to the user can be executed.
An example is given in Figure 1. User(U) requests service from service provider SP1. SP1 requires user attributes *(a,b)* to satisfy its service policy. U provides *(a,b)* and gets the service. Suppose that U, at the end of this successful negotiation, opts for sharing attribute *(a)* within the federation, and suppose that then U requires a service from another provider SP2 in the same federation. The attribute requirements there are *(a,c)*, but U only has to provide the attribute *c* to receive the service.

At the end of a successful negotiation users receive one of two types of ticket. The first is called *trust ticket* and is issued to non-member users, to provide information about the previous services and service providers the user has accessed. Trust ticket is not required for members who have a provider reference storing attributes related to them. The other type of ticket is the *session ticket* issued to non-member users. This ticket is valid for a short period of time, within which, if the user asks for the same service, he/she is given that service without any additional requests. The rationale is that most user attributes do not change within a short time interval, therefore if a user was successfully authenticated recently, with high probability his/her attributes are still valid for accessing the service. This, of course, depends on the policy of the service provider. We show a detailed negotiation process using the described user cases in Section 6.
The second type of negotiation occurs between two service providers. This is useful when a user successfully negotiates a service from one service provider, in fact he/she automatically becomes eligible to receive service from another service provider. As such, when the user asks for a service the FSP providing it can directly negotiate user-related attributes with the FSP holding such data from previous negotiations. Also, negotiations among providers might be required for verifying external user identities. As we do not rely on a single identity provider, a provider might not be aware of the last registered users. When a request from a locally unknown user-id is received, a service provider can directly interact with the provider issuing the claimed user-id to double check its validity.[7]

[7] For simplicity we assume user-id contains service provider information to easily identify the issuer.

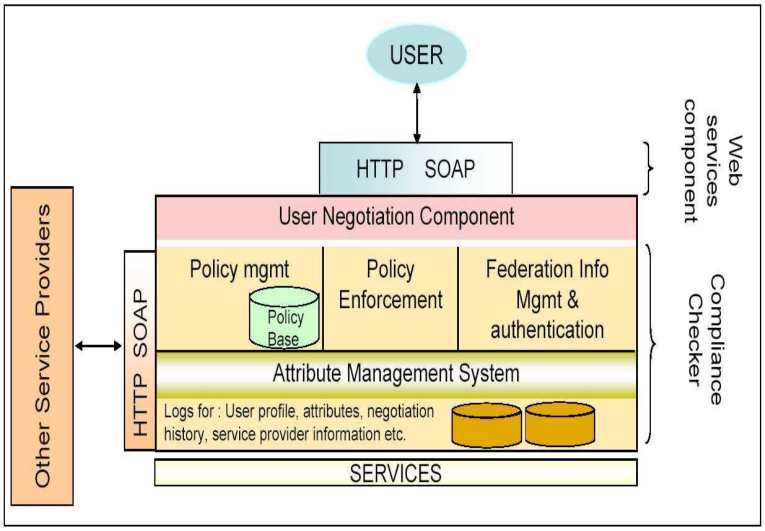
USER
HTTP SOAP
Web services component
User Negotiation Component
Policy mgmt
Policy Base
Policy Enforcement
Federation Info Mgmt & authentication
Compliance Checker
Attribute Management System
Logs for : User profile, attributes, negotiation history, service provider information etc.
SERVICES
HTTP SOAP
Other Service Providers

Figure 2: Architecture for FAMTN Service Provider.

4 FAMTN framework

In this section we provide an overview of the FAMTN framework, showing the provider architecture and illustrating the ticketing system supported by the federation.

4.1 FAMTN provider architecture

A FAMTN framework is composed of FAMTN service providers (FSP) that contain the necessary components required to execute: 1) trust negotiation among users and FSPs; and 2) federation of user attributes. The FSP framework is sketched in Figure 2. FSP is equipped with components deriving from the two underlying frameworks of automated trust negotiations and federated identity management. However, unlike conventional identity management framework we do not have a central identity provider as each FSP can also perform the functionality of an identity provider. Any external user who does not have an identity in the federation registers himself by obtaining a username and password with any FSP of the federation. A member user is implicitly registered with the FSP he is affiliated with. This registration information, required for the user identity verification, is temporary for non-member users and is carried in trust tickets that are issued, as described later.

There are three components in any FSP. The first is the web services component required to enable secure communication within the federation and with the users. The second is the user negotiation component which contains the modules executing the negotiation, depending on the type of user. The third is the compliance checker which is responsible for the access control policies and attributes together with the compliance checking for the policies. To perform such tasks the compliance checker includes components for policy and attribute management and policy enforcement. In addition to these modules, the federation information management component is added, which is responsible for the information related to other FSPs in the federation. Precisely, the federation information management component is in charge of validating certificates and user tickets validation by verifying the FSPs signatures. This module is also responsible for revoking the trust tickets and user credentials which have become invalid due to timeout or user misbehavior.

4.2 Ticketing system in a FAMTN federation

The two types of tickets supported by our framework are temporal with a fixed lifetime. We assume loosely synchronized clocks in the federation to know when the ticket timeouts. We use the SSO id as the user-id in the tickets. Structure and functions of the tickets are discussed in what follows.

Session Ticket

A session between a member user and a FSP is a complete transaction resulting in either service authorization or in service refusal due to lack of authorization. In this paper we primarily consider first case. A session ticket ensures that if the negotiation ends successfully and the same user requests the same service provider for the same service, the service can be granted immediately without unnecessarily having to repeat the trust establishment process. A session ticket therefore contains the following fields: $\boxed{Signed_{SP} < \tau(s_{req}), u, T, R>}$

where $\tau(s_{req})$ denotes the service requested, *u* is the user-id and *T* is the ticket time stamp. Here, *R* denotes the result of the negotiation. R might be either a simple statement or a structured object. The use of structured objects is particularly interesting for tracing intermediate results of negotiations of aggregated services. A session ticket is signed by the service provider, which actually authenticates it giving a receipt of the trust establishment. Since session tickets are encrypted with the service providers private key, they are tamper proof and can be verified. The time-out mechanism depends on the type of user attributes required for the service, and the security level of the service. For example, if the only attribute required is the date of birth, then due to its stable nature, the time-out can be large. Nevertheless, if the service is of high security level, one might require re-confirmation and the freshness of session ticket lasts a shorter time.

Trust Ticket

The purpose of the trust ticket is to determine the list of previous services external users have accessed. Assuming that all the service providers are members of the same federation, the signature of a member provider can be verified by any other member provider. Such a ticket has the following form:

$\boxed{Signed_{SP_{last}} < list\{\tau(s), FSP, T\}, u, T\text{-}I >}$

Every 3-tuple in the list contains the type of service, the corresponding service provider and the timeout. *u* corresponds to the temporary user identification, and T-I is the expiration date for this id. The ticket is signed by the last service provider with which the user had a successful transaction. At the end of a successful transaction, the service provider takes the current user trust ticket, removes all *timed out* entries, appends its information, signs it and sends it to the user. If the policy requirements related to a service are known within a federation, then showing the corresponding item from the trust ticket list for one service would automatically qualify the user for another service. For example if service provider S_1 provides service s_1 only if the user's age is above 25, then the user would automatically qualify for service provider $S_2's$ service s_2, which requires that the age be above 18. This is the main idea behind attribute $subsumption$. It can significantly shorten the trust establishment process and reduce the need for exchange of attribute information.

5 Attribute sharing in Federation

Secure attribute sharing is one of the main goals of the FAMTN framework. As observed in the previous sections, we achieve a certain degree of efficiency in the negotiation due to the tickets. The rest depends on how well attribute sharing is implemented. Hence the privacy of user information is critical and the sharing of the user information within the federation should be in accordance to the user preferences. We therefore present an extension of the XML based χ-TNL language to describe how users can set privacy preferences. We also demonstrate the concept of different levels of trust between the user and the different service providers of the same federation.

5.1 Privacy Preferences in Attributes

We extend the χ-TNL language to include a new field called **pref** to express users privacy preferences. This field can have two values, namely **canFederate** and **notFederate**. These are included in both the types of attributes namely factual and conditional, as illustrated by Figure 3. As the name points out, factual attribute gives a fact about the user whereas the conditional attribute is a condition that the user attribute satisfies.

```
                         Factual Attribute of a user

<Year-of-birth :pref = notFederate>
                <dob> 1979 </dob>
<Year-of-birth>


                       Conditional Attribute of a user

<Age-Condition :pref = canFederate>
                <age-greater-than> 25 </age-greater-than>
</Age-Condition>
```

Figure 3: Types of User Attributes

The first attribute shown in the Example in 3 cannot be shared since the pref value is set to notFederate, thus giving a choice to the user to *opt-out* from federating the attribute. However the second one can be shared within the federation. Thus, specifying a value for the pref field contributes to support the privacy preferences of users. Note that organizations may have policies which pre-define the federated attributes. In this case the employees and other members directly affiliated with this organization do not have control over the sharing of their attributes. This is normally stated in the agreement they have signed with the company. Thus member attributes can be available to any service provider in the federation because the privacy requirements concerning these attributes are properly set by the affiliated organizations. The attributes need to address the concern of heterogeneity of attribute names for the same property. The attribute name descriptors should not only use the same vocabulary between the attribute name and the policy base of one federation, but with each member of the federation. Otherwise, a translation mechanism is required to map the different values to be able to share the attributes and verify if they satisfy a given policy.

6 Negotiation in Identity Federated System

The negotiation process for trust establishment depends on the type of the involved user and the history of his/her interactions with the federation members. Algorithm 1 shows the complete negotiation process which includes all the user cases. In this section the negotiation algorithm is explained in detail with the help of examples illustrating the different user cases. Here we assume one federation. Multiple federations with non-empty intersection are outside the scope of this paper.

The different user cases give the basis of the design and analysis of the FAMTN negotiation process. The negotiation of user attributes between the user and the service providers, and between the service providers is a subset of this negotiation process. Intuitively a recent user should get service access faster than a new user. Similarly a repeated user, who already received services from different service providers of the federation, should get service access faster than a new external user. Finally, a member user, being internal to the federation thus being more trusted, should have advantages in the negotiation process as compared to a new external (non-member) user.

Considering the following scenario: Alice is a Purdue Student. She needs a health check up from the student health center. The health check up requires her to be of age greater than 25. Following this she would need to go to a Pharmacy for a medicine which requires her to be of age greater than 18 and pay for the medicine. The Pharmacy and Health Center are members of the same federation which is the universities health services department. Alice is not considered to be a part of the federation and hence is an external non-member user. We compare the above with the case when Nora, a nurse in the Health Center, who due to her job is automatically a member user of the federation.

6.1 New External User (non-member)

When Alice requests the health checkup service the message shown in Figure 4 is sent to a federation service. This is the first time she is taking advantage of a federation service. As such, first she is assigned a temporary SSO id. Line 1 to 4 of the algorithm check for the validity of the request. It is not valid if the service provider cannot offer the requested service. The policies of the health center are presented in Figure 4. These policies allow Alice to choose between the two valid credentials to prove her age.

Since Alice is a non-member the execution of the algorithm skips to line 29. Since this is the first time Alice is using a service from the federation, her trust ticket is empty, therefore she herself has to provide all the credentials to satisfy the policy requirements. She provides her student ID and driver's license and suc-

Alice → Student Health Center (service request)

```
<ServiceType>Health-CheckUp</ServiceType>
<Token-List>NULL</Token-List>
```

Health Center Service Policy

```
$pol_1$ = ({}, Health-Checkup ⇐ Purdue-Student
       Student-ID-Card (name=Purdue-Student-Name))
$pol_2$ = ($pol_1$, Health-Checkup ⇐
       Divers-License (year-of-birth < 1979))
$pol_3$ = ($pol_1$, Health-Checkup ⇐
       International-Passport (year-of-birth < 1979))
$pol_4$ = ($pol_2, pol_3$,
       Health-Checkup ⇐ SERVICE-GRANTED)
```

Student Health Center → Alice (user tickets)

```
$TrustTicket_1$:
<Trust Ticket>
(HealthCheckUp,StudentHealthCenter, timeout-120-days,121204,Alice)
</Trust Ticket>
```

Figure 4: Example of a new external user service request, service policy and user trust ticket

cessfully gets the service. As given in line 46, she gets a trust ticket as shown in Figure 4.

6.2 Repeated External User (non-member)

Now Alice wants to get a medicine from the Pharmacy store which in addition to the Health Center policy requirements also needs a valid credit card to buy the medicine. The initial validity checks are the same as before, but not at line 29, where Alice's non-empty trust ticket is retrieved. This leads to the Pharmacy and the Health Center negotiating Alice's attributes.

If the Pharmacy knows the policy of the Health Center before then it does not need to interact with the Center. Having the proof that Alice successfully negotiated a service with the Health Center for a health-checkup implies that Alice meets the “Purdue University Student” and “Age $>$ 18” requirement. Thus the policy requirements of the Health Center *subsume* some of the policy requirements of the Pharmacy. If the policy is not known from before, Pharmacy sends a request for attributes, giving its policy to the Health Center. The Health Center replies indicating the policy conditions which are satisfied by the information the health center has, and does not send the actual attribute values.

In this case the main loop of the negotiation algorithm ends after the first iteration since Alice visited only one federated member before requesting the service. Thus the number of iterations is proportional to the number of different federated members the user has got service from. An active member therefore presumably has an advantage, since there are more chances that most of his/her non-critical information is already available within the federation. After completing the main loop of the algorithm, the Pharmacy requests Alice for her credit card number. Thus we see that Alice did not have to negotiate for the attributes already given to the federation.

6.3 General Member User

Now considering Nora, who is a nurse in the Health Center requesting service from the Pharmacy. We assume that member user’s attributes are present with the member’s affiliated organization, therefore Nora’s federated attribute information is available from the Health Center. Her authenticated user identification is a proof of her affiliation with the Health Center. Thus only one negotiation round between the current service provider and Nora’s service provider, that is, the Health Center, is required as given in line 12 of the algorithm. There could also be special policies which apply discounts to members of the federation which will be accounted at this step. The only disadvantage is that Nora may not have as much control over her information as Alice, since Nora has to comply to the policies of the Health Center. If the Health Center’s policy is to share its employees age with the federation, then Nora cannot *opt-out* of this option. Although Nora can potentially allow her private information to be shared, we assume that all the information that can ever be federated is available at her organization. This not only makes the negotiation process faster for all members, but also avoids revelation of attributes which conflict with the interests of her organization. If more attributes are required after negotiating within the federation, Pharmacy directly negotiates with Nora as directed by line 15. When she successfully qualifies for the service she is given a session ticket.

6.4 Recent Member User

Following from above if Nora needs the same service again from the Pharmacy, her session ticket would enable her to bypass any negotiation and she can get the service. As such, Nora gets a seamless access to the system without having to re-submit the attribute information.

7 Key features of the FAMTN framework

The FAMTN framework is characterized by security properties that current federated identity management framework and trust negotiation framework do not have.

Minimal disclosure and no leakage of user attributes.

In a typical federated environment, there is a central identity provider which has all user information. In the FAMTN framework, the information is distributed dynamically among the service providers and each service provider has the minimal information concerning users.

The attributes cannot be leaked because users specify if they want to federate their attributes in their certificates. The policy enforcement component of the compliance checker prevents the service provider from misusing the information. Also, the user profiles stored can only be accessed through the attribute management system which authenticates every request. No outsider nor entity not involved in the negotiation can get any user information since the communication is encrypted. A member of the federation not providing service to the user cannot request for the attribute information of the user. This is because the inter-service provider negotiation requires each federated member to prove validity of the request through user service request token and its own policy statement.

Authorized disclosure of private service provider policies to external users.

This property is assured by any automated trust negotiation where the negotiation process does not allow disclosure of policies until the pre-requisites are met. Thus, a user trying to learn the policies of a federated member cannot do so, until the user attributes available to the server provider satisfy the initial requirements.

Number of attributes required from a user decreases with more interaction with the federation.

The number of attributes required for the FAMTN negotiation decreases with the external users accessing more services from the federation. Assuming negotiation with the user takes more time than negotiation within the system the efficiency of the entire negotiation increases with time. A recent user is not required to provide any attributes for the service he has already received.

8 Conclusion and Future Work

In this paper we have explored the integration of federated digital identity management with trust negotiation. This paper highlights several issues which need to be addressed for a practical implementation of the FAMTN framework. This includes questions regarding policy e.g. policy compliance and subsumption of policies. The language to define the policies should use vocabulary well understood not only between user and organization, but among the whole set of organizations. This may not be a realistic assumption and we would need to look into alternatives. Policy languages supporting the specification of credential sharing within a federation do not exist and will be useful for better privacy control in a federation. Another important problem is the representation of attributes. This is essential for efficient lookup if several users are using the system. Also the meaning of the attribute and its underlying logic can help to infer implications between conditional attributes.

References

[1] D. L. Baumer, J. B. Earp, and P. S. Evers. Tit for Tat in cyberspace: Consumer and Website Responses to Anarchy in the Market for Personal Information. *North Carolina Journal of Law and Technology*, 4(2), 2003.

[2] E. Bertino, E. Ferrari, and A. C. Squicciarini. Trust-χ: A Peer-to-Peer Framework for Trust Establishment. In *IEEE Transactions on Knowledge and Data Engineering*, pages 827– 842. IEEE, July 2004.

[3] Identity-Management. Liberty alliance project. http://www.projectliberty.org.

[4] Internet2. Shibboleth. http://shibboleth.internet2.edu.

[5] Netegrity. Identity and Access Management: The Promise and the Payoff. http://www.hillwriter.com/NetegrityWhitePaper.pdf.

[6] RSA-Security. Rsa Sign-On Manager. http://www.rsasecurity.com/.

Algorithm 1 FAMTN-negotiation process

```
**Require:** $userID, userAuthenticationInfo$
**Ensure:** $IsRegistered(userID)$
$userRequest \Leftarrow getRequest(userID)$
**if** $userRequest \notin Services_{FSP}$ **then**
    $return$ Abort-Negotiation
**end if**
***Comment:** For Members*
**if** $isValidMember(userID) = true$ **then**
    $sessionTicket \Leftarrow getSessionTicket(userID)$
    **if** $sessionTicket \neq NULL \ \wedge sessionTicket.time < timeout$ **then**
        $return$ OK
    **end if**
    $M_{FSP} = getMemberFSP(userID)$
    $remAttrList1 \Leftarrow NEGOTIATE_{FSP}(Curr_{FSP}, M_{FSP}$
                                $userID, userRequest)$
    **if** $remAttrList1 \neq NULL$ **then**
        $remAttrList2 \Leftarrow NEGOTIATE_{User}(Curr_{FSP},$
                                $userID, CurrPolicy_{FSP})$
    **else**
        $send(SessionTicket) \Rightarrow userID$
        $return$ OK
    **end if**
    **if** $remAttrList2 \neq NULL$ **then**
        $return$ Abort-Negotiation
    **else**
        $send(SessionTicket) \Rightarrow userID$
        $return$ OK
    **end if**
**end if**
***Comment:** For Non-Members*
$FSPlist \Leftarrow getTrustTicket(userID)$
**while** $FSPlist \neq EmptyList$ **do**
    $M_i = rmHeadOfList(FSPlist)$
    $remAttrList3 \Leftarrow NEGOTIATE_{FSP}(Curr_{FSP}, M_i$
                                $userID, userRequest)$
    **if** $remAttrList3 = NULL$ **then**
        $send(TrustTicket) \Rightarrow userID$
        $return$ OK
    **end if**
**end while**
**if** $remAttrList3 \neq NULL$ **then**
    $remAttrList4 \Leftarrow NEGOTIATE_{User}(Curr_{FSP},$
                                $userID, CurrPolicy_{FSP})$
**end if**
**if** $remAttrList4 \neq NULL$ **then**
    $return$ Abort-Negotiation
**else**
    $send(TrustTicket) \Rightarrow userID$
```