Towards Automated Trust Establishment in Federated Identity Management

David W. Chadwick, Mark Hibbert

Towards Automated Trust Establishment in Federated Identity Management

David W. Chadwick, Mark Hibbert

School of Computing
University of Kent

{d.w.chadwick, m.j.m.hibbert}@kent.ac.uk

Abstract. We present the Federation Semantic Attribute Mapping System (F-SAMS), a web services based system which enables a semi-automated dynamic trust establishment mechanism for managing identity federations. We present the conceptual model which allows current members to dynamically introduce new members into the federation in a trustworthy manner, using a web of trust model. F-SAMS enables existing members to interact securely with previously unknown new members of a federation and allows them to retrieve policy and semantic information about them.

Keywords: Semantic access control, trust establishment, trust management, identity federation

1 Introduction

In an ideal world, users would be able to access content on the Internet by simply using different subsets of their authorisation credentials (certified identity attributes) that are globally known by all service providers (SPs). In reality the only way that a user can access content from multiple SPs is for the user to authenticate herself either directly with every SP or indirectly via an identity provider (IdP) that the SP trusts, which then asserts the user's identity attributes to the SP. One problem that arises in the latter case is how can the SP trust the assertions made by the IdP. The current solution is for SPs and IdPs to form identity federations, which create circles of trust, where users from an IdP can access the shared resources of the federation SPs.

Identity federation establishment is usually a manual process, whereby the members, or federation operator, will agree on a set of federation terms. Any IdPs or SPs who do not agree to support the federation terms will not be allowed to join the federation. In addition the federation members must establish trust relationships with each other (so that the digitally signed requests and assertions can be verified and trusted). To do this, the interacting parties typically exchange their public key certificates (PKCs). An alternative method is that the PKCs of all the members are regularly distributed to everyone by the federation operator in the form of signed metadata. This semi-automated method enables members to interact with each other without having

pre-established relationships. However, current federations are relatively static entities, whose memberships change relatively slowly, and all changes have to be agreed by the federation operator.

1.1 Motivating Use Case

Our research is driven by the following motivating use case. The number of international students studying postgraduate courses at UK universities has been rising year on year - almost half of all taught postgraduate students were international in 2010-11 [3]. University admissions officers are inundated with applications from students across the world, with each student required to produce evidence that they have attained the prerequisite qualifications for their chosen course. Currently each student presents his/her paper qualification certificates, which have been issued by one of the many different worldwide educational institutions. These have different grading schemes, different pass marks and different levels of attainment, yet it is the job of the admissions officer to determine whether the applicant holds the necessary qualifications to be accepted on to the course. To do this, the admissions officer must map the applicant's qualifications into ones that are locally recognized by his university (as specified in the course prospectus). In addition, he must also check the name of the certificate holder against that of the applicant, which can usually be done from the applicant's passport, unless there has been a name change in the intervening period. He also needs to be sure that the issuing institution performed similarly strong authentication of the applicant to ensure that the certificate is bound to the same person.

Currently, the UK university admissions officers have assistance from UK Naric (http://www.ecctis.co.uk/naric/default.aspx), which provides a directory of foreign educational institutions, the qualifications they offer and an estimate of their equivalent UK counterpart. However, the information it provides is only advisory and each UK admissions officer must make his own decisions about the trustworthiness and mapping of the foreign qualification attribute. Germany on the other hand, has a more advanced system, in that UniAssist (http://www.uni-assist.de/index_en.html) acts as a trusted third party for the university admissions officers and provides validated mappings of the foreign qualifications into their German equivalent. UniAssist validates the authenticity of the paper certificates by asking the applicant to take it to the German embassy in its country of issuance to be validated and stamped.

As electronic qualification certificates and electronic ID cards become more prevalent, in the not too distant future university admissions officers (playing the role of SP) will require an electronic system that can perform the certificate validation for them (which is equivalent to validating attribute assertions from unknown IdPs in a federation). This has led us to define the Federation Semantic Attribute Mapping System (F-SAMS), which allows SPs to securely map unknown attributes from unknown IdPs into the known attributes used in its access control policy, without having a pre-existing trust relationship with the IdP. Similarly an IdP can reliably accept attribute requests from an unknown SP, knowing that the privacy of its attributes will be preserved. Our F-SAMS system enables IdPs to trust attribute requests from unknown SPs and SPs to verify and interpret attribute assertions from unknown IdPs.

The remainder of this paper is structured as follows: section 2 describes the F-SAMS system, whilst section 3 describes the F-SAMS trust model. Section 4 applies F-SAMS to the motivating use case described above, whilst section 5 discusses related work. Finally section 6 concludes and considers future work.

2 The F-SAMS System

The Federation Semantic Attribute Mapping System (F-SAMS) is a web services based system that is added to federations to enable them to dynamically grow so that SPs and IdPs can reliably identify and trust one another. F-SAMS serves two purposes. Firstly it provides semantic mappings between (unknown) attributes that are asserted by IdPs and a standard set of federation attributes, and secondly it provides information about the trustworthiness of the federation members. The standard set of federation attributes are defined by the federation root of trust (FRoT). The FRoT is the overall authority within the federation, e.g. this could be the organization which created the federation. This standard set of attributes is relatively stable, but it can evolve over time to meet the changing needs of the federation members. The trustworthiness of the federation members is based on a web of trust model which is similar to the PGP trust model [16]. This allows existing federation members to dynamically introduce new federation members based on their collective recommendations.

2.1 F-SAMS Protocol Flow

Fig 1 shows the protocol flow between a user's agent (such as a browser) and the SP, which involves assertions made by an unknown IdP to an unknown SP. The F-SAMS service initially publishes two (dynamically changing) sets of (IdP and SP) federation metadata that IdPs and SPs will subsequently use to automatically refresh their metadata, say once per day (step 1 in fig 1). The current list from F-SAMS may contain many previously unknown services that the retriever has previously not interacted with. When a user wishes to access a federation service, she first navigates her client to the F-SAMS service to identify federation SPs (steps 2 and 3). Once identified the user agent navigates to her chosen SP and requests access to its service (step 4). The SP tells the user agent the standard set of federation attributes it requires for the service along with the latest list of federation IdPs (many of which may be unknown to it). The user chooses her preferred IdP and the user agent is redirected to it with an authentication and attribute request signed by the SP (step 7). The IdP can verify the authentication and attribute request from the possibly unknown SP using the metadata it has automatically retrieved from F-SAMS, thereby allowing it to trust the SP with its attributes (see section 3.1). The IdP authenticates the user and may ask the user for consent to release her locally held attributes (steps 8 & 9). The IdP responds to the SP's attribute request and provides the user's attributes in a signed response (step 10). Note that the IdP sends its own attributes to the SP, after checking its local mappings to see which ones are equivalent to the requested attributes. It does this for reasons of trustworthiness, since the SP relies on trusted introducers to do the mapping from an

IdP's attributes to the standard set, rather than relying on the IdP itself (see section 2.2). The SP receives the signed response from the possibly unknown IdP containing possibly unknown attributes about the user. It verifies the IdP's signature using the metadata (PKC) from F-SAMS, then queries F-SAMS to identify the unknown attributes (step 11). F-SAMS looks up the relationships of the unknown attributes to identify suitable federation attributes that they can be mapped to. The federation attributes are returned to the SP (step 12), which can then make an access decision and return a response to the user's agent (step 13).

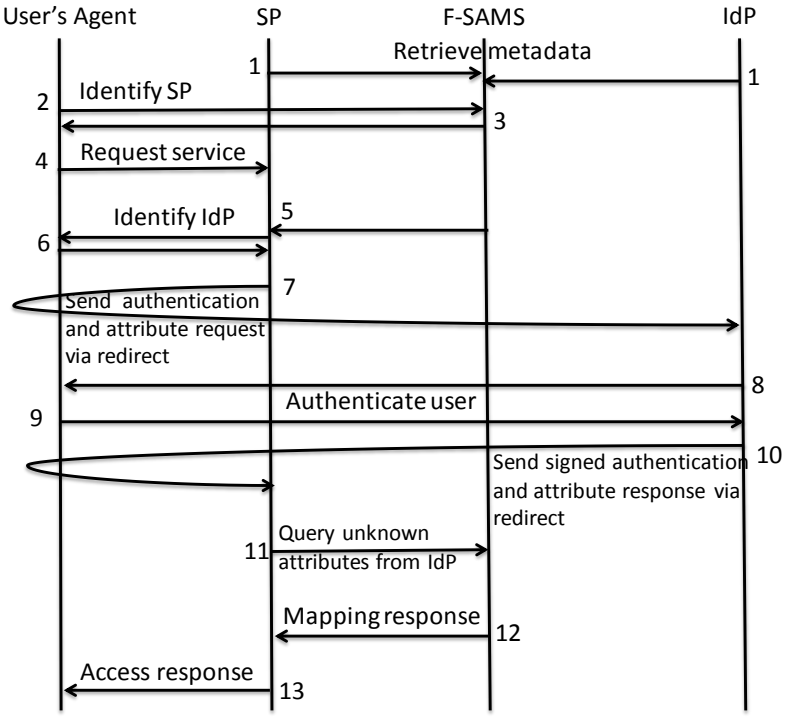

Fig. 1. – F-SAMS Protocol Flow

2.2 F-SAMS Service

The central part of the F-SAMS service revolves around trust and vocabulary expression (TruVEx) documents, which are published on the web by each member of the federation and the federation root of trust (FRoT). To cater for the different types of federation members, there are three types of TruVEx document; one published by the FRoT, one published by SPs and one published by IdPs. Table 1 lists the elements of each type of TruVEx document. Each TruVEx document is an RDF/XML document [17] comprising a set of RDF triples.

The elements of the TruVEx documents are defined as follows:

- ***the PKC part*** holds the X.509 PKC of this federation member (or candidate

Table 1. – Elements of each TruVEx document

TruVEx Document Publisher	PKC Part	IdP Policy Part	Friends Part	Federation Vocabulary Part	Federation Privacy Policy Part	SP Privacy Policy Part
FRoT	X		X	X	X	
IdP	X	X	X			
SP	X		X			X

member). This can be a self-signed certificate, or one issued by a CA. The only restriction is that it must contain the uniformResourceIdentifier component of the subject alternative name (SAN) extension and hold the URI of the web location storing the detached signature of this member's TruVEx document. The (candidate) member determines this URI at the time its PKC is issued, even though the location will initially be empty.

- ***the IdP policy part*** expresses the IdP's authentication policy and attribute mapping policy. The former consists of the highest authentication level of assurance (LoA), as defined in the NIST guidelines [15], that the IdP is capable of performing on its subjects. The latter consists of the relationships between the attributes in the (candidate) IdP member's local vocabulary and those in the federation standard vocabulary including whether the local attribute is either registered or authoritative, along with the registration LoA (RegLoA) for each registered attribute (see section 3.3). See [2] for more information on the semantic mappings. The member will need to keep this information up to date as and when the IdP's local attributes and the standard federation attributes evolve over time.
- ***the friends part*** contains information about each friend i.e. a (candidate) member that this member (the introducer) asserts to be trustworthy. This is used to create the web of trust between federation members and new candidate members. The actual information published for each friend is dependent on the type of friend. If the friend is an IdP, the introducer also states its confidence in the attribute mappings declared by the (candidate) member. The following is published:
  - Friend Type (IdP)
  - Friend's PKC
  - Level of confidence (LOC) that the introducer has in the friend to be a trustworthy federation member. For an IdP friend, this means that it is trusted to authenticate its users as published in its authentication policy and provide an accurate authentication LoA to the federation SPs. The LOC has a value between 0 and 1 (see section 3.2).
  - Hash value of the IdP policy part of the friend's TruVEx document that is being testified to. This is to ensure that the policy that is being attested to, has not been altered since the introducer last validated it.
  - For each attribute mapping in the friend's IdP policy, the confidence that the introducer has in the mapping (AMLOC) and optionally the confidence the introducer has in the way the attribute was registered by the IdP (RegLOC) (see section 3.3).

When a friend assertion is made about an SP, the introducer is also implicitly as-

serting its confidence in the SP to abide by its stated privacy policy. The privacy policy of the SP is published in the privacy policy part of the SP's TruVEx document. The following friend information is published:

- Friend Type (SP)
- Friend's PKC
- The LOC that the introducer has in the friend to be a trustworthy federation member. For an SP friend, this means that it is trusted to abide by its published privacy policy.
- Hash value of the privacy policy part of the friend's TruVEx document. This is to ensure that the policy being attested to has not been altered since the introducer last read it.

- ***the federation vocabulary*** part is published in the FRoT's TruVEx document and is the set of standard federation attributes that other members will map their own local attributes into, either in their attribute mapping part of their TruVEx documents if they are IdPs, or in some local storage if they are SPs. In the latter case other members do not need to know which local attributes SPs use in their authorization decision making). As the federation evolves, the FRoT may dynamically expand the federation vocabulary to include other attributes that are of interest to the federation's members, and members may update their attribute mappings accordingly. This allows finer grained access controls to be introduced.
- ***the privacy policy part*** contains either the privacy policy of the SP member or, in the case of the FRoT, the minimum acceptable privacy policy that all federation SPs must conform to. A candidate SP will not be accepted into the federation by F-SAMs if its privacy policy is below this minimum. When a candidate SP becomes a trusted federation member, its privacy policy will be copied by F-SAMS and stored in its trust base, so that it can be given to IdPs when they request trust information about federation SPs. The privacy policy part contains the following information:
  - The member's name and address, to be used by users in case of a complaint or to retrieve a copy of their attributes stored by the SP.
  - The set of SP's purposes for processing IdPs' attributes (must be a subset of the FRoT's set).
  - A list of the standard attributes that are processed.
  - A list of third party recipients or categories of third party recipient to whom the attributes might be disclosed (must be a subset of the FRoT's set).
  - A list of countries outside of the European Economic Area to where the SP may transfer any of the attributes (must be a subset of the FRoT's set).
  - The list of users' access rights to the attributes held about them, taken from read, update and delete (must be a superset of the FRoT's list).
  - The maximum retention period of the attributes (must be less than the FRoT's period).

Using the private key corresponding to the public key published in the PKC part, each (candidate) member signs its TruVEx document and stores the detached signature at the URI contained in the SAN field of its certificate.

Fig 2 shows the relationships between six members of a federation as contained in their TruVEx documents. Org B, an IdP, has two friends, Org E and Org D, which it

has introduced to the federation, and one friend, the FRoT, who has vouched for it. Org F is currently only a candidate member of the federation, since it has insufficient friends at the moment (see Table 2).

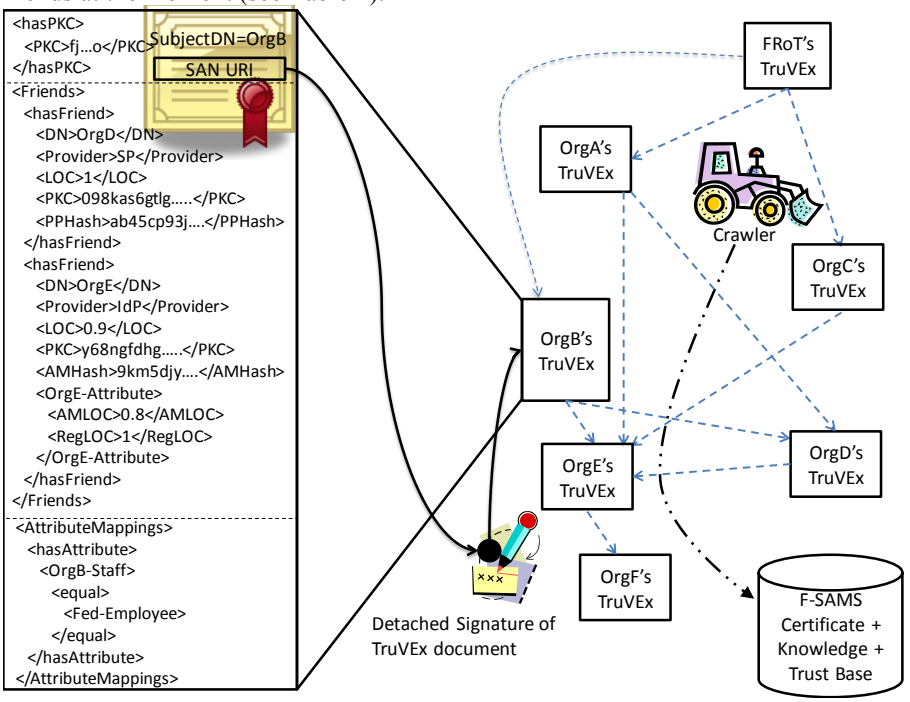

Fig. 2.– Example F-SAMS federation

A web crawler is responsible for identifying, collecting and processing all of the TruVEx documents for the federation. Starting at the FRoT's TruVEx document, it crawls the web picking up the TruVEx documents of other federation and candidate members. Using these documents, it constructs the F-SAMS certificate, trust and knowledge bases, which run as a web service used by the federation members. Once a candidate member passes the trust threshold (described in section 3.2), it becomes a federation member.

The notion of a "friend" is used in the TruVEx document to describe an organization whose IdP policy or privacy policy is trusted to some degree by the introducer asserting the friendship. F-SAMS does not require mutual "friendships", instead it is used as an indication of (partial) trust, or confidence, the introducer has in the subject and its published policy. For a candidate member to become a federation member it must obtain sufficient introductions from organizations that are already members.

The TruVEx documents are signed to both prove the integrity of the document and to verify the friend assertions. This is similar to a signed X.509 PKC, where the issuer asserts that the public key in the certificate belongs to the named subject of the certificate. In the case of F-SAMS, the introducer asserts that it has some level of confidence that the candidate member will honour its published IdP or privacy policy.

Members can update and re-sign their TruVEx documents as often as they wish,

provided they ensure the following:

- whenever the document is altered, it must be re-signed and its detached signature should be stored at the signature URI contained in the SAN field of the signer's PKC. This is to ensure that the signature can be discovered and the document's integrity can be verified.
- if a member changes its PKC, it must inform its introducers, since they have published its PKC (in their friends parts).
- if an IdP member alters its IdP policy part, it must notify its introducers so that they can verify the modified policy and publish the new hash value and their confidence levels in it (in their friends parts).
- if a SP member updates its privacy policy, it must inform its introducers so that they can review the changes and publish an updated privacy policy hash value (in their friends parts) if they accept it.
- All members may update their friends' parts of their TruVEx documents as often as required without notifying anybody.

To assist with the automation of informing their introducers, members should maintain a list of introducers (LOI) that will contain the email addresses of their introducers. The LOI can then be used to inform the introducers when the member updates its policy. The introducers can either confirm the new policy is valid and update their friend's entry accordingly, or if the policy is no longer valid, the friend's entry can be removed as this member is no longer trusted by the introducer. In the event that only some of the attribute mappings are updated, and only some of these are deemed to be still valid by an introducer, the entire attribute mapping part does not have to be invalidated. Instead the introducer simply gives the unsuitable mappings a low (or zero) AMLOC confidence level and the F-SAMS crawler will then decide whether they should still be included in the federation knowledge base or not.

The member's X.509 PKC information is used for verification of the identity and public key of the member. It contains the distinguished name (DN) of the member, the member's public key and the location of the detached signature of the member's TruVEx document (in the SAN extension). Note that a member may assert its own identity, by issuing a self-signed certificate, but its friends will validate this when they add the member to their friends' lists. The certificate is used by the crawler to confirm the subject's DN with the one constructed from the friend assertions made by already trusted members of the federation in their TruVEx documents. The PKC is also used to confirm that the TruVEx document was signed by the correct entity.

3 F-SAMS Trust Model

3.1 Trust in Federation Members

The F-SAMS trust model provides a semi-automated and scalable solution for establishing trust among federation members. The FRoT is a fully trusted entity by all members of the federation, and is the root of the federation trust chains. If an organization does not trust the FRoT, we assume that it would not want to join the federa-

tion. Trusted paths are constructed from the FRoT to all federation members based on the friend assertions within members' TruVEx documents. This creates a web of trust whereby the FRoT trusts its friends directly, and other members indirectly based on its friends' friends, recursively. Hence, all members directly trust the FRoT as the root of trust and indirectly trust other members of the federation based on their friend assertions. A candidate member is given a trust score related to the combined confidence levels of all its introducers, which must reach the membership threshold in order for it to become a member and receive a trust level. The latter is used to calculate the weight of their own friend assertions about the candidates that they introduce.

The implicit trust that F-SAMS confers on each member of the federation diminishes as they move further away from the FRoT in the trust chain. This further diminishes as the confidence of their introducers is lowered. The trust level (*TL*) of a member is computed by F-SAMS based on their position in the trust chain and the confidence levels of their introducers as follows:

$$TL = \frac{1}{(PL+1)} \times LOC_{av} \quad (1)$$

where *PL* is the shortest path length from the federation member to the FRoT and LOC_{av} is the average level of confidence of the introducers as computed in equation (4) below.

A candidate member of the federation only becomes trusted to join the federation when its trust score (*TS*), computed by adding together the confidence adjusted trust levels of all its introducers, reaches the membership threshold (typically set to 1). Having a membership threshold enables federation trust to remain strong and excludes any candidate members with a trust score below the threshold until they attain the desired trust score. Only then will an IdP's individual attributes be assessed as to whether they are deemed trustworthy enough to be entered into the federation knowledge base. The only way a candidate member can become trusted is to increase the number of trusted introducers to increase its trust score to meet the threshold. A candidate member will require more introducers if it is further away from the FRoT or its current introducers have less confidence in it. The *TS* of a candidate member is computed as follows, based on the set of trusted introducers (*int*):

$$TS = \sum_{i:int}(TL_i \times LOC_i) \quad (2)$$

$$TS \geq threshold \quad (3)$$

where *TS* calculates the confidence adjusted trust score contribution for each introducer, giving the sum of all introducers' trust score contributions, where *TS* must be greater than or equal to the membership threshold. By combining the trust and confidence levels of the introducers, the system can decide whether a candidate member can be trusted enough to be accepted into the web of trust, thus eliminating the possibility that a rogue member can autonomously introduce new (potentially rogue) members into the federation.

When an introducer refers a candidate member to the federation, it also specifies its level of confidence in its assessment of the candidate. This is in the form of a *LOC* variable which takes values between zero and one, where zero means no confidence at all and one means full confidence. The *LOC* in an SP candidate member is the introducer's confidence in it adhering to its published privacy policy. The *LOC* in an IdP candidate member is the introducer's confidence that the IdP will adhere to its pub-

lished authentication policy and will supply SPs with correct authentication LoAs during the authentication exchanges.

When a candidate member becomes fully trusted (i.e. has a $TS \geq$ threshold), F-SAMS computes LOC_{av}, which is used to calculate the trust level TL of a member in equation (1). The calculation for LOC_{av} is shown below:

$$LOC_{av} = \sum_i \frac{TL_i \, x \, LOC_i^2}{TS} \quad (4)$$

where LOC_{av} is a weighted average of the confidence levels of the introducers, weighted by their relative contribution to the total trust score. Introducers with lower confidence levels (and higher trust levels) will ensure that LOC_{av} does not get too high. Members whose trust scores fall below the threshold at any time will become candidate members again and their TL will be removed.

Fig 3 shows an example web of trust that is created by six members plus the FRoT. It results in ten trust chains[1]: FroT → OrgA, FroT → OrgB, FroT → OrgC, FroT → OrgA → OrgD, FroT → OrgA → OrgE, FroT → OrgB → OrgD, FroT → OrgB → OrgE, FroT → OrgC → OrgE, FroT → OrgA → OrgD → OrgE, FroT → OrgB → OrgD → OrgE. Whilst OrgE is trusted within the federation, it alone cannot introduce a new member (say OrgF) into the federation regardless of it being 100% confident in its assessment, as OrgF's trust score is relative to OrgE's own trust level, whichwas computed using the average LOC of its introducers. Table 2 shows the trust scores and trust levels computed for each organization and its corresponding TruVEx document. The TL calculation for OrgE's TruVEx document illustrates the LOC_{av} computations from equation 4 above, and shows that compared to OrgD, even though OrgE's trust score is higher, its trust level is lower because its introducers have less confidence in their ratings of OrgE compared to OrgD. If OrgC were to subsequently introduce OrgD with a low confidence level (as for OrgE) then OrgD's trust level would be slightly reduced as a consequence of this.

Candidate members wishing to join the federation have to publish a TruVEx document and get trusted members to verify their IdP policy or privacy policy and introduce them. Once enough introducers have been found to give the candidate member a $TS \geq$ threshold, they can be trusted and a TL is automatically calculated for them. It is the responsibility of each introducer to verify that the policy is correct and sensible. However, the burden placed on each introducer is reduced the further they are from the FRoT as their TL will be lower, meaning that the responsibility of verifying the mappings falls to more introducers. For multiple, intersecting trust chains, each introducer's confidence will only be added to the candidate's TS once, while the TL will be calculated based on the shortest path to the FRoT and the average confidence of the introducers.

3.2 Trust in an IdP's Attributes

Once trusted, the IdP's attribute mappings may then be processed. The IdP's attribute

[1] Note that OrgF's TS is less than the threshold of 1, and therefore it does not have a trust chain from the FRoT

mapping part contains all of the IdP's asserted attributes and semantic information describing how they relate to the standard federation attributes. There are two types of asserted attribute: registered and authoritative. Registered attributes are ones the IdP obtained from an external attribute authority when registering the user in its system.

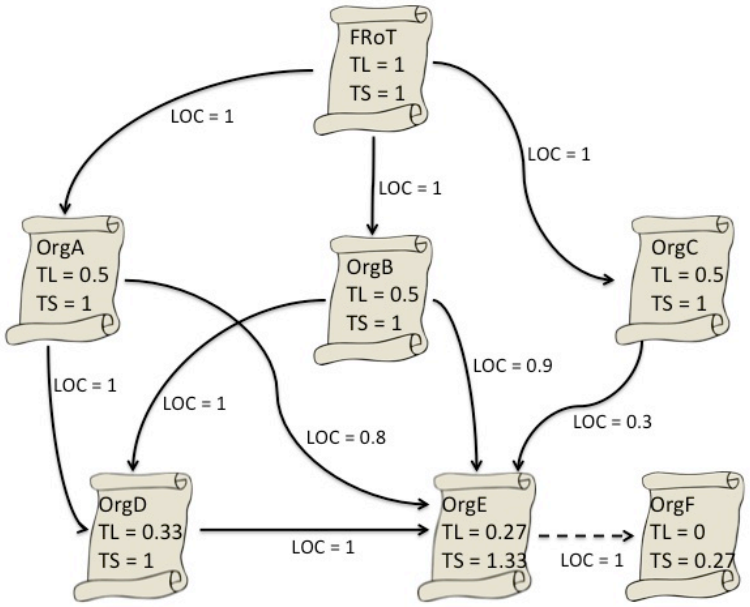

Fig. 3. – F-SAMS's trust chains

Table 2. – *TS*s and *TL*s of TruVEx documents from Fig 2.

TruVEx Document	$\sum TL_i \times LOC_i$	**TruVEx Document TS**	**Is document trusted?**	**TruVEx Document TL**
FRoT	n/a	1	Yes	1
OrgA	$(1x1)_{FRoT}$	1	Yes	$(1/(1+1))*((1x1^2)/1)=0.5$
OrgB	$(1x1)_{FRoT}$	1	Yes	$(1/(1+1))*((1x1^2)/1)=0.5$
OrgC	$(1x1)_{FRoT}$	1	Yes	$(1/(1+1))*((1x1^2)/1)=0.5$
OrgD	$(0.5x1)_{OrgA}$ + $(0.5x1)_{OrgB}$	1	Yes	$(1/(2+1))*$ $(((0.5x1^2)/1)+$ $((0.5x1^2)/1)) =0.33$
OrgE	$(0.5x0.8)_{OrgA}$ $+(0.5x0.9)_{OrgB}$ $+(0.5x0.3)_{OrgC}$ $+ (0.33x1)_{OrgD}$	1.33	Yes	$(1/(2+1))*$ $(((0.5x0.8^2)/1.33)+$ $((0.5x0.9^2)/1.33)+$ $((0.5x0.3^2)/1.33)+$ $((0.33x1^2)/1.33)) = 0.27$
OrgF	$(0.27x1)_{OrgE}$	0.27	No	0

Authoritative attributes are ones the IdP has issued itself and is authoritative for. For example, when a student registers at a university, his name and date of birth are registered attributes, usually taken from his passport, whilst his degree course and marks are authoritative attributes issued by the university. Whilst the authoritative attributes are simply published along with their mappings to standard federation attributes, the registered attributes must contain an extra level of detail regarding how they were verified when they were registered with the IdP. We use the identity proofing aspect of NIST's level of assurance (LoA) [15] to indicate the registration level of assurance (RegLoA). For example, if the IdP asserts a name attribute that has not been verified and was self-asserted by the user during registration, the IdP's name attribute will have a RegLoA of 1 associated with it. However if the user had to provide a driver's licence and passport in person when registering, then the RegLoA would be 4.

The introducers of the IdP will assert certain levels of trust about the candidate IdP in their friends part of their TruVEx document. These various trust metrics (with values between 0 and 1) relate to the confidence they have in three aspects of the candidate IdP, these are:

- LOC: The confidence the introducer has in the candidate IdP to abide by its policy to authenticate a user and provide an accurate authentication LoA to the SP, as described previously.
- AMLOC: The confidence that the introducer has in each attribute mapping published in the candidate IdP's attribute mapping part. Stating its confidence in each attribute mapping allows introducers to indicate any suspect attribute mappings, without penalising the acceptable mappings.
- RegLOC: The confidence that the introducer has in the RegLoA of each registered attribute of the candidate IdP (published in its attribute mapping part). This gives the introducer the chance to confirm that the candidate IdP makes the correct provisions to ensure that their users' registered attributes are verified to the level that they assert.

For each of the IdP's attributes F-SAMS calculates an attribute confidence score (*ACS*) from the introducers' AMLOC values, adjusted in accordance with the introducers' *TL*. The calculation for an attribute's *ACS* is shown below:

$$ACS = \sum \delta ACS \tag{5}$$

$$\delta ACS = TL_i \times AMLOC_i \tag{6}$$

where TL_i and $AMLOC_i$ are the trust level and attribute mapping level of confidence provided by introducer i, respectively. Only attributes with an ACS that reach the ACS threshold (typically 1) will be entered into the F-SAMS knowledge base to be used to provide mappings to SPs when the attribute is unknown to the SP. Those attributes that do not meet the threshold will not be added to the knowledge base. If an SP receives these attributes, it will discard them as unknown/untrustworthy.

For each registered attribute, its trusted RegLoA must also be determined. The IdP's self-asserted RegLoA needs to be confirmed as reliable before F-SAMS adds it to the knowledge base. To achieve this, an attribute registration score (ARS) is calculated similar to the ACS, based on the *RegLOCs* provided by the introducers and adjusted by the introducers' *TLs* (as with the *ACS*) as follows:

$$ARS = \sum \delta ARS \tag{7}$$

$$\delta ARS = TL_i \times RegLOC_i \quad (8)$$

where, TL_i is the trust level of the introducer i and $RegLOC_i$ is the *RegLOC* provided by introducer i. If the *RegLOC* for the registered attribute reaches the RegLOC threshold (typically 1), the registered attribute is added to the knowledge base with the IdP's asserted RegLoA. If the *RegLOC* is less than the RegLOC threshold, the trusted RegLoA will be level 1, equivalent to a self-asserted (by the user) registered attribute. This is because the combined introducers did not agree with the IdP's asserted RegLoA for the identity proofing used to verify the registered attribute. Therefore it should be treated by the SP as an unreliable self-asserted attribute. Table 3 shows an example set of computations for IdP OrgE. For the registered name attribute, Name (R), F-SAMS will only store the RegLOA of 1, and not the IdP asserted RegLoA 4, due to the introducers' combined confidences falling below the threshold.

Table 3. Computing trust in an IdP's attributes

	Degree Name (A)	**Classification (A)**	**Name (R)**	**Nationality (R)**
IdP RegLoA	-	-	4	4
Org A	AMLOC 0.8	AMLOC 0.8	AMLOC 0.9 RegLOC 0.5	AMLOC 0.8 RegLOC 1
Org B	AMLOC 1	AMLOC 1	AMLOC 0.9 RegLOC 0.6	AMLOC 1 RegLOC 0.9
Org C	AMLOC 0.9	AMLOC 0.8	AMLOC 1 RegLOC 0.5	AMLOC 1 RegLOC 1
Org D	AMLOC 1	AMLOC 0.7	AMLOC 0.6 RegLOC 0.5	AMLOC 0.7 RegLOC 0.8
ACS	1.68	1.53	1.6	1.63
ARS	-	-	(0.5*0.5) + (0.5*0.6) + (0.5*0.5) + (0.33*0.5) = 0.97	1.71
F-SAMS RegLoA	-	-	1	4

3.3 SP's handling of an IdP's attributes

SPs can receive four different types of attribute from an IdP: a known authoritative attribute (the SP can make an authorization decision using this and the authentication LoA from the IdP); an unknown authoritative attribute (this needs to be mapped into a known attribute by F-SAMS before the SP can make an authorization decision using it and the authentication LoA from the IdP); a known registered attribute (the SP needs to obtain its RegLoA from F-SAMS before it can make an authorization decision using it along with the lowest of either its RegLoA or the authentication LoA from the IdP); and an unknown registered attribute (this must be mapped into a known attribute by F-SAMS, which must also provide its RegLoA. The authorization decision is then based on the known attribute and the lowest of the RegLoA and authentication LoA from the IdP).

4 Use Case

Returning to the motivational use case in section 1.1, we now show how F-SAMS can be used to assist admissions officers in handling electronic international postgraduate applications. F-SAMS can utilize the EC's European Credit Transfer and Accumulation System (ECTS) [4], which standardizes a set of qualifications to levels and credits. This allows students with qualifications issued by an ECTS member institution to have their qualifications mapped to the local university equivalent. By extending this model with F-SAMS, the ECTS will act as the FRoT, and will publish a TruVEx document listing the ECTS members as its friends. This basic model of just the FRoT's friends is the same as the current solution. However, when this federation is extended to include the ECTS members' friends, we can then include educational institutions from around the world who have previously had successful interactions with a subset of the ECTS members. This will allow other members who have not had any interaction with them, to validate their qualifications and map them into recognizable ECTS ones. Not only will F-SAMS help to automate the mapping of unknown qualifications through the use of the knowledge base, but it will also help to automate the authentication of the qualifications, as the student will present a digital qualification which has been digitally signed by the issuing institution and can be validated using the trust and certificate base in F-SAMS.

5 Related Work

Josang et al. [1] argue that trust must be pre-established between federation members in order for them to agree upon the mappings of each user identifier. However, most of the literature in the area of dynamic trust in identity management disagrees with this view, arguing that it is not scalable (see for example, [9-14]) since these trust relationships are not only required to be pre-established, but they must be agreed upon by the entire federation, which poses a challenge when a new provider is added.

The authors of [5, 6, 10 and 14] build on the existing SAML [17] identity management framework to create extensions to help automate the process of trust establishment. [5] extends SAML to include the provider's PKC in the metadata and store it at a centralized entity, which can be retrieved when an unknown provider is encountered. The unknown provider's PKC can then be verified using the CA's certificate. F-SAMS provides a web service that can provide trust information to requesting providers, but it does not require one entity to retrieve and verify all of the providers' PKCs. Rather it builds a trust base from the introductions of other providers. [6] requires providers to request reputation information from a known trusted entity about the unknown provider, however, this reputation of trust is built from one trusted third party who must know both providers, while [14] extends this reputation request system to the cloud, where users can set their own privacy policies to govern how their attributes are released. F-SAMS does not require all providers to maintain their own trust engines and compute trust at runtime. Rather it can be queried by providers to obtain a pre-computed trust score for the unknown provider. [10] combines SAML

with trust negotiation to allow unknown providers to attempt to prove that they are trustworthy at runtime. This negotiation requires each entity to provide credentials issued by a trusted third party. [15] presents a model for discovering the trust to place in a SP when releasing user attributes, based on a user feedback system. Though it dynamically computes trust in SPs, it still requires SPs to have pre-established trust in IdPs. [16] allows users to dynamically federate SPs and IdPs, so that IdPs will semi-trust the SPs to retrieve a subset of the user attributes, and the SPs will treat these as user self-asserted attributes.

While [7] highlights the risks involved in allowing cross federation access to resources and identity information, their model for cross federation trust relies on one member of one federation asserting that a provider in another federation is trustworthy. This takes accountability away from the external provider as the introducer is responsible for all their actions. F-SAMS shares this burden between multiple introducers and requires the external provider to become part of the federation, thus making them accountable to the federation for their actions. F-SAMS also removes the possibility of one member introducing multiple rogue providers to the federation.

The authors of [8] suggest extending SP and IdP policies to include a required (or partially required) set of credentials that an unknown SP or IdP must provide to be considered trustworthy. This method of establishing trust still requires a trusted third party to issue the required credentials to the unknown provider. The authors of [9] take a different approach to trust and allow users to federate their attributes with SPs once they have accessed their service. A new SP can request the user's attributes from the previously accessed SP, rather than needing to trust the IdP. The first SP that a user accesses must have an existing trust relationship with the user's IdP.

All of the current solutions to trust establishment in federated identity management require a trusted third party in order to gain trust in an unknown provider. F-SAMS does not. It allows existing trusted federation members to introduce unknown providers, and it then computes a trust score based on these introductions. F-SAMS requires all federation members to trust the FRoT, who is responsible for the management of the federation. But the FRoT does not need to directly trust all of the members, as candidate members may be introduced by existing members of the federation. The trust relationships are thus built dynamically from a network of trust.

6 Conclusions & Future Work

We have presented F-SAMS, which is designed to address the current limitations in dynamic trust establishment in identity management federations. Using the F-SAMS trust model, an organization may join a federation by receiving enough introductions from existing members who have some degree of trust in it and its policies. We have shown how F-SAMS could be used to remove some of the challenges facing university application systems, enabling admissions officers to verify and translate unknown digital qualifications. In future work, we aim to develop a management interface allowing organizations to easily create, sign and publish their TruVEx documents.

References

1. Jøsang, A.; Fabre, J.; Hay, B.; Dalziel, J.; Pope, S. Trust Requirements in Identity Management. Australasian Information Security Workshop 2005, Newcastle, Australia.
2. D.W. Chadwick, M. Hibbert: F-SAMS: Reliably Identifying Attributes and their Identity Providers in a Federation. In: Herrero P et al. (Eds.): On the Move to Meaningful Internet Systems: OTM 2012 Workshops, LNCS 7567, pp. 231-241. Springer, Heidelberg (2012).
3. UK Council for International Student Affairs - http://www.ukcisa.org.uk/about/statistics_he.php
4. European Commission ECTS - http://ec.europa.eu/education/lifelong-learning-policy/ects_en.htm
5. Patrick Harding, Leif Johansson, Nate Klingenstein, "Dynamic Security Assertion Markup Language: Simplifying Single Sign-On," IEEE Security & Privacy, pp. 83-85, March/April, 2008.
6. F. Almenarez, P. Arias, A. Marin, and D. Diaz: Towards Dynamic Trust Establishment for Identity Federation. In proceedings of the Conference of the Euro-American Association on Telematics and Information Systems, Prague, CZ, June, 2009.
7. Uwe Kylau, Ivonne Thomas, Michael Menzel, Christoph Meinel, "Trust Requirements in Identity Federation Topologies," Advanced Information Networking and Applications, International Conference on, pp. 137-145, 2009 International Conference on Advanced Information Networking and Applications, 2009.
8. Hao Gao; Jun Yan; Yi Mu; "Dynamic Trust Model for Federated Identity Management," 4th International Conference on Network and System Security (NSS), pp.55-61, 2010.
9. Abhilasha Bhargav-Spantzel, Anna C. Squicciarini, Elisa Bertino, "Trust Negotiation in Identity Management," IEEE Security & Privacy, pp. 55-63, March/April, 2007.
10. Yicun Zuo, Xiling Luo, and Feng Zeng. “Towards a dynamic federation framework based on SAML and automated trust negotiation.” In Proc. 2010 int conf on Web information systems and mining (WISM'10), Fu Lee Wang, Zhiguo Gong, Xiangfeng Luo, and Jingsheng Lei (Eds.). Springer-Verlag, 254-262, 2010.
11. “Electronic Authentication Guideline”, NIST Special Publication 800-63-1, Dec 2011.
12. “How PGP works”. Available from http://www.pgpi.org/doc/pgpintro/#p20
13. W3C RDF/XML Syntax Specification - http://www.w3.org/TR/rdf-syntax-grammar/
14. Sanchez, R.; Almenares, F.; Arias, P.; Diaz-Sanchez, D.; Marin, A., "Enhancing privacy and dynamic federation in IdM for consumer cloud computing," Consumer Electronics, IEEE Transactions on, vol.58, pp.95,103, February 2012.
15. Félix Gómez Mármol, Joao Girao, Gregorio Martínez Pérez, “TRIMS, a privacy-aware trust and reputation model for identity management systems”, Computer Networks, Volume 54, Issue 16, 15 November 2010, Pages 2899-2912.
16. Ferdous, Md Sadek; Poet, Ron; Dynamic Identity Federation using Security Assertion Markup Language (SAML), To appear in: 3rd IFIP WG 11.6 Working Conference on Policies & Research in Identity Management, 2013.
17. OASIS. “Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V2.0”, OASIS Standard, 15 March 2005