Dynamic inter-organizational cooperation setup in Circle-of-Trust environments

Latifa Boursas
Munich Network Management Team
Technische Universität München, Germany
http://www.mnm-team.org/~boursas/

Vitalian A. Danciu
Munich Network Management Team
Ludwig-Maximilians-Universität München, Germany
http://www.mnm-team.org/~danciu/

***Abstract*—The need for collaborative service provisioning across different providers' domains is being addressed by Circles of Trust (CoT), whose members adhere to the same policies and expose the same interfaces for collaboration. Today's CoT specifications require a high initial effort on behalf of enrolling members, thus obviating quick or even ad hoc setup of business cooperation with entities outside a CoT.**

We explore a procedure that complements the static aspects of the CoT by a dynamic assessment of trust levels between organizations. Its benefit lies in the shortened setup time for a business interaction, which can be achieved by automating the assessment process. The appraisement of a potential partner outside a CoT leverages existing CoT members' experience. We propose algorithms suitable for calculating trust values and discuss alternative solutions to be used where reputation-based assessment within the CoT is impossible.

I. Introduction

Delivery of the emergent pervasive services requires flexible and quick setup of cooperations between geographically and organizationally distributed providers on the global market. Security and privacy constitute important aspects, as they are key enablers of today's services landscape. For users to feel comfortable with a service they must have confidence that the services they use are trustworthy and secure. However, with services facing the user, usability of the service must not be diminished by security measures.

In particular, Single Sign-On (SSO) [18] schemes are a popular means for reducing repeated authentication challenges faced by service users. While SSO within single administrative domains hardly poses any difficulty, in inter-organizational settings, Federated Identity Management (FIM) techniques constitute a prerequisite for SSO realization. The increase in cross-domain service provisioning has supported the proliferation of standardization in the FIM area, taking into account identity management, security as well as privacy aspects [13].

Creating inter-organizational, trustworthy service provisioning raises the issue of trust between the participating organizations. FIM makes it possible for a user authenticated by his account partner organization (e.g. employer, school, ISP), known as an Identity Provider (IDP), to make use of personalized services, provided across multiple domains by partner organizations (Service Providers). For example, a subscriber of an Internet Service Provider (ISP) may be eligible to purchase and download multimedia content from a content provider cooperating with the ISP—without having to create an account. Such arrangements are due to contractual agreements between the two provider organizations, governing the realization of everything from privacy policies (e.g. regarding the handling of users' personal data) to discounts. The above example implies a certain *level of trust* between providers being supported by the contractual binding.

Circle-of-Trust fundamentals

Technical realization of such cooperation agreements is facilitated by *Circle of Trust* (CoT) frameworks which specify a common set of policies, procedures and collaboration interfaces within a group of organizations. A CoT is a federation of identity and service providers whose purpose is to facilitate business relationships with regard to security and privacy concerns.

Instead of 1:1 relationships between principals, the CoT offers an association—a club, if you wish—where enterprises (and other organizations) can apply for membership.

To become a CoT member, an organization is compelled to adhere to the specification, in particular to procure and operate prescribed software packages, and to demonstrate that CoT policies are respected and enforced. In return, the setup of cooperation with another CoT member organization is accelerated by the common base of interfaces and by an initial level of trust—the enrollment process supplies a form of *certification* of a fellow member. Accordingly, *trust* in this context is built on a common set of rules, responsibilities, and commitments set forth in the *CoT Foundational Documents*. In practice, the CoT may be a group of service providers that operate the same software package and that share identity information recorded at IDPs. If two of them wish to cooperate, the trust foundation as well as the identity management infrastructure is already in place.

Membership in a CoT accelerates the setup of a business cooperation, e.g. a collaboration between two providers in IDP and SP roles. The participating providers within the CoT have operational agreements, SSO functionalities and an identity management infrastructure (exchange of authentication and authorisation information), such that identity providers and customers can transact business with any of all these service providers in a secure manner. As an *operative benefit*, this allows integration of the services facing the members' customers, while ensuring that user data is shared according to published privacy policies [24].

For the purposes of this work, the CoT ensures an *implicit initial level of trust* that can be exploited in reasoning about the trustworthiness of principals within and outside the CoT.

Taking into consideration the globalization of service provisioning together with shortened setup time until delivery (real-time/ad-hoc, at worst), current CoT specifications may be too rigid. The contractual framework together with a specification of duties for members render the application process slow. In addition, the benefits a CoT offers are only useful when *both* partners in a potential cooperation (e.g. to provide service to a traveling user's location) are members.

For a member, however, the CoT does effectively provide a trust base that can be leveraged in order to instantly estimate a trust value for a hitherto unknown potential cooperation partner. In this paper, we present an approach to deriving trust assessment for entities outside the CoT. As a non-CoT organization may have business relations with some of the CoT members, a member can consult her CoT peers with regard to a non-CoT organization that requests cooperation. Their appraisal, formulated as a level of trust indication based on the requesting organization's conduct, can be employed as base for the initial level of trust for the cooperation.

After reviewing related work in Section II we explore the alternatives for extending trust relationships outside the CoT by means of a scenario in Section III. We consolidate the necessary concepts for a trust model in Section IV. In Section V, we present workflows and algorithms that leverage the initial trust level within a CoT to allow assessment of organizations outside the CoT. The realization of the approach, including a discussion of software components, is described in Section VI. A cooperation based on recommendation does bear hazards, as we show in Section VII in addition to a discussion of open issues.

II. Related work

The Liberty Alliance Project (LAP) [7] recognizes the concept of a CoT as part of its federated identity vision. They have developed specifications and guidelines to help organizations establish a legally binding CoT, which has enforceable contractual forms between the parties implementing the Liberty specifications [25]. A similar approach is found in the *FIXS* Project [8] that conveys an initial trust to all the participating members. Formally, direct relationships between the participants are established through acknowledgment and agreement to the 'Terms of Use', thus enabling distributed and trusted authentication.

FIM standards: OASIS's Security Assertion Markup Language (*SAML*) [21] or the Web Services Federation Language (*WSFL*) [20] can be employed for reliable authentication within a federation. It is presupposed that principals are bound by contractual relationships, thus deferring the matter of trust to the legal domain.

Trust brokering: The LAP defines a *Brokered Trust Model* for the case, where two entities that wish to cooperate lack direct, mutual business agreements. It is limited to brokering transitive trust between CoT members [15].

Trust models

Public Key Infrastructure (PKI) systems, such as *X.509* [26] and *PGP Web of Trust* [27], provide a certain view on trust. The processing of certificate path meets the needs of deterministic identification and authentication. It can be used to establish access control and authorization and support the attribution to private communities and groups of entities. In contrast to this perception, *trust* deals with assumptions, expectations and risk, and it cannot be quantitatively enforced.

PolicyMaker [2] and its successor *KeyNote* trust management systems [1] complement the traditional certificate frameworks by binding access rights to a principal's public key within the certificate framework. A similar approach is found in the IBM trust e-business system [12]. Certificates can be issued by various bodies, vouching for an entity in a particular role (e.g. the roles of buyer or seller). In addition, it supports the formulation of rules preventing access by means of a Trust Policy Language (TPL).

Beyond validating a principal's certificate and mapping the certificate owner to a role, we need a trust model that dynamically assigns trust levels according to principals' behavior. PGP Web of Trust employs a trust ontology with four trust levels (*untrusted*, *marginal trust*, *complete trust* and *implicit trust*). We use this scale as a starting point for developing a trust metric in Section IV-B.

The *Friend Of A Friend* (FOAF) Project [4] provides machine-readable documents describing persons, the links between them and their activities, by means of an RDF schema. To support the authentication of the FOAF documents, [5] provides a methodology for signing and encrypting those documents with PGP. [14] defines a trust module that describes trust between the FOAF individuals as well as the subject upon which that trust is based. Thus it creates structures similar to CoTs, based on the content of FOAF documents. Though structurally similar, FOAF targets documents located on the web and cannot be leveraged in the context of our approach.

Golbeck and Hendler [10] extended the FOAF Project and developed a trust system for generating-locally reputation rating for semantic web social networks, in such a way that trust and reputation can be expressed on the semantic web using ontologies for descriving entities and the trust relationships that connect them. This trust system deduces trust by node traversal, but the associated schema focuses on social trust, i.e. between people, and exibits the small world behaviour [17]. The reason is that semantic reputation networks are essentially social networks that cover the small world properties, such as the transitiviy and the composability properties [9][19]. However, the properties supported in this ranking approach are exclusively web-specific, allowing a personalization of the way content in the web is presented to the end-user. Therefore, it cannot be applied for the evaluation of the online transaction requirements investigated in our work.

Our approach is similar to the one presented in the *European Project SECURE* [22][6], which proposes a trust and risk framework to secure collaboration between ubiquitous computer systems. It aims at supporting collaborative tasks through an access control manager who grants or denies permission for entities to execute actions according to a certain trust level. While the SECURE Framework gives the access control manager fine-grained control over who they trust, it employs a centralized decision point. Hence, it does not allow distributed decison making over multiple participants in the circle of trust.

III. Scenario and solution alternatives

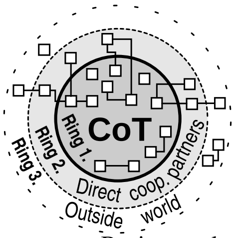

Figure 1. Business relationships

Circles of trust are created to facilitate business cooperation while ensuring that security and privacy requirements are met. Figure 1 illustrates the business relationships (edges) between organizations (rectangular vertices). We differentiate between the relationships within a CoT, those crossing the border of the CoT and those located outside the CoT. From a FIM perspective, every organization assumes the role of identity provider, service provider, or both. In the following, we develop a scenario taking place between a small number of organizations from Figure 1.

A. Scenario description

Due to the commodization of internet access, many internet service providers (ISPs) offer additional (e.g. value-added) services to their subscribers. In particular, entertainment content is offered via the ISPs web portals, either bundled with the access service, or as an additional offering. Frequently, the ISP itself is not owner of the content, but it is being provided by specialized content provider (CP) organizations.

Envision a setting with two kinds of organizations: *ISP/NOs* operate their core and access networks (thus being network operators, NOs) and offer internet access to their customers/users (in the function of internet service providers, ISP). *Content providers* offer streaming content to be consumed by end users.

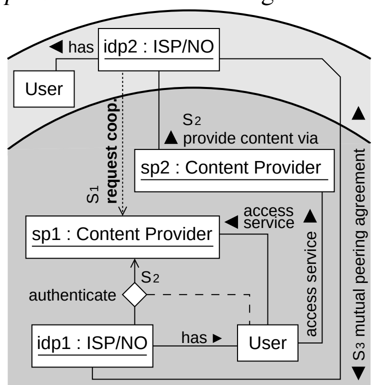

Figure 2. Scenario snapshot

Figure 2 illustrates a scenario where two ISP/NOs and two content providers collaborate. Three of these (`sp1, sp2, idp1`) are members in a CoT, while the remaining actor (`idp2`) is not. Each ISP/NO's users are being given access to a content provider's services, e.g. a video streaming service. In addition, the two ISP/NOs have a peering agreement regarding network traffic. From a FIM point of view, in this setting the ISP/NOs act as Identity Providers, as they have a record of their respective users' account data as well as the possibility to authenticate the users. The content provider is in a Service Provider role. Thus, when a user requests access to the media content, her ISP/NO transmits account information to the content provider in order to facilitate the transaction. Both content providers, as well as one of the two ISP/NOs (`idp1`) are members of a CoT, so that certain guarantees regarding users' privacy may be assumed. In particular, the users authenticated by `idp1` may access content from both content providers, since the transactions all take place within the CoT.

The ISP from outside the CoT, `idp2`, wishes to offer her users video streams provided by `sp1`. A trust relationship must be established between the two providers as a prerequisite for service delivery to users. It must be ascertained that the ISP will provide dependable authentification for the users accessing the CP's service and accurate account data, e.g. for billing purposes. The content provider therefore needs to assess the trustworthiness of the requester entity `idp2`.

B. General requirements

As requirements, we formulate the following constraint on the approach:

a) Short setup time: The setup of a business cooperation should be quick. Delays caused by the setup of trust and security infrastructure cause opportunity cost.

b) Low cost: The cost of the setup of a business cooperation should reflect the benefit, thus making the former dependent on its duration and business volume.

c) CoT integrity: The integrity of the CoT must not be diminished. A CoT, by definition, implies common rules and procedures designed to serve as assurance for CoT members. Therefore, relaxing the standards for the benefit of dynamic cooperation would defeat the purpose of the CoT.

d) No impact on third parties: Cooperations crossing the borders of the CoT must not impact non-participant CoT entities.

C. Solution alternatives

As `sp1` and `idp2` have not conducted business together before, neither of them is in the position to leverage experience values for the trustworthiness of the other. Hence, a number of options are available to secure the level of trust necessary for service delivery.

1) CoT membership: As suggested in Section I, full membership in the CoT may not be an option for a small enterprise, or for an organization seeking a temporary business affiliation with a CoT member; the cost involved would prove prohibitive in the former case, while setup time may obviate the business goal in the latter case.

2) CoT-agnostic cooperation: A cooperation can be setup without taking into account the CoT membership of the queried party. This alternative implies bilateral negotiations between the requesting organization and the CoT member.

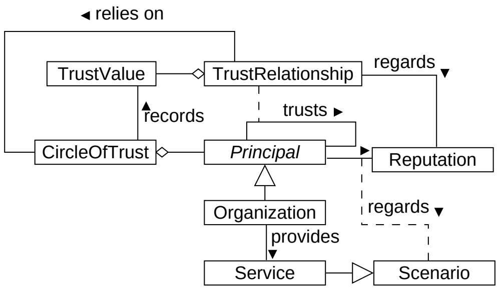

Figure 3. Static model of trust

3) Leverage experience within the CoT: A CoT member can make inquiries within the CoT with regard to the conduct of a potential business partner. In this case, `sp1` could ask the other CoT members about the quality of business relationships with `idp2`.

4) Extend inquiries outside of the CoT: The inquiry domain can be extended to reach outside the CoT. While the CoT provides a structure within which inquiries can be made (e.g. it makes available a directory of members), for inquiries outside the CoT domain similar structures need to be identified. The "initial" trust granted to CoT members as a consequence of the enrollment procedure does not apply outside the CoT.

D. Discussion of alternatives

In many cases, a membership application (solution III-C1) by the external party is impractible, e.g. if requirements *a* and *b* are not satisfied. A cooperation without regard to the CoT (solution III-C2) presupposes knowledge of the external party in order to assert mutual trust.

Following, we focus on the only alternatives not violating any requirements, namely III-C3 and III-C4.

IV. Modeling trust

The aspects of cooperation setup sketched in Section III are described in detail in this section and serve as a foundation of the static trust model sketched in Figure 3.

A. Dimensions

In the following, we identify aspects pertaining to the trust assessment procedure that will be discussed in Section V.

1) Principals: Entities are represented as the set of principals $\{P_1, \cdots, P_n\}$ who participate in the CoT, or have direct and indirect business relationships to the members in the CoT. Thus, principals relevant to our approach can be divided into several groups: members of a CoT (*member*), CoT-external organizations that are known by CoT members, external organizations whose identity can be verified by a certificate issued by a *Certificate Authority CA*, and unknown organizations. These principals can assume different roles.

During setup of a cooperation the role of a *Requester* can be identified as an organization that wishes to cooperate with a CoT member, the *Respondent*.

2) Trust context: Trust between two principals is established for a certain *trust scenario*, in analogy to trust relations between people: trusting someone to cooperate with you on a task may be different from trusting their *opinion* about a third party. Therefore, we differentiate between trust scenarios $S : \{S_1, S_2, \cdots, S_n\}$ by service, and we consider queries with respect to third parties to be just another service, for this purpose. In the setting in Section III, the transmission of account information from `idp1`, the access to the CP `sp1` and `sp2` services as well as the peering agreement between `idp1` and `idp2` regarding network traffic can all be formulated as trust scenarios.

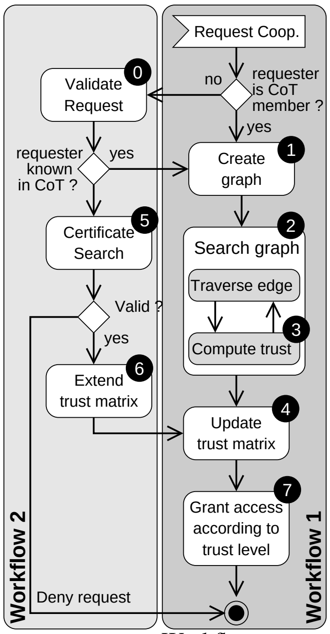

Figure 4. Workflows

3) Query dimensions: As indicated in Section III-C, several *modes of query* may be available: fellow CoT members can be queried regarding previous cooperation with a requester, direct neighbors of CoT members can be queried in the same manner, or CAs outside the CoT can be asked to verify a requester's identity. A *result of a query* depends on the selected mode. The quality of the assessment may be either a *trust value* supplied by queried parties, or a statement regarding an organization's identity, based on its certificate. The latter case spans values of positive, invalid/revoked and unknown.

B. Quantifying trust

Trust values can be assigned or computed. They are employed to quantify the level of trust placed by one principal in her relationship with another principal. A value Set $T : \{T_{1,1}, T_{1,2}, \cdots, T_{n,n}\}$ represents the computed trust relationships. Precise representation of computed trust values requires a continuous scale, while familiarity between principals requires special discrete values to represent established—rather than calculated—trust. We define trust as having values in the range $T \in [0, 1]$, where 1 indicates absolute trust and 0 indicates absolute distrust. They may also be undefined in cases where numerical values cannot be found; this is usually the case when there are no principals with direct relationship to the unknown entity. We assign a value of -1 to indicate an unknown trust level.

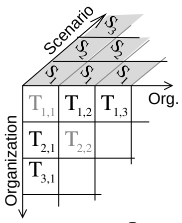

Figure 5. Inter-organizational trust with respect to scenarios

Matrix representation: We have determined that trust relationships are formulated with regard to a pair of organizations in the context of a scenario. Hence, a three-dimensional structure (Figure 5) is necessary to represent the trust relationships between members of a CoT. The height and width of the cube represent the members, while the depth of the structure represents the scenarios.

Note that instead of the three-dimensional structure, the examples in the remainder of the paper illustrate a single scenario S_k in a two-dimensional matrix with the entries $T_{ij} \in [0, 1]$ as shown in Equation 1:

$$M(S_k) = \begin{pmatrix} - & T_{1,2} & T_{1,3} & \dots T_{1,n} \\ T_{2,1} & - & T_{2,3} & \dots T_{2,n} \\ \vdots & & \ddots & \vdots \\ T_{n,1} & & \dots & - \end{pmatrix} \qquad (1)$$

V. Trust assessment process

We differentiate between two kinds of dynamic trust relationships: (i) Those among two principals that are both members in the CoT but have not conducted business together before and (ii) those relationships crossing the borders of the CoT, i.e. those between a CoT member and an organization located outside the CoT. In the following, we will present two workflows for dynamically setting up these two types of trust relationships. Key activities are illustrated by example.

The workflows rely on the trust matrix introduced in Section IV-B. Before business cooperations have been initiated, the values $T_{i,j}(S)$ designating trust between CoT members are set to an arbitrary initial value (we use a value of 0.5 in our examples) to reflect the agreements regarding identity sharing aspects, protocols and privacy policies.

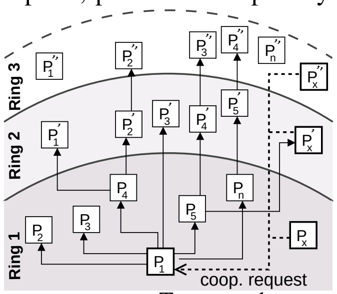

Figure 6. Trust graph

The trust values in the matrix will be updated after each new transaction happening inside the CoT, thus enabling the members to raise or to lower trust values according to an update function.

The workflows are triggered by requests for cooperation directed at a CoT member who assumes the responder role. Workflow 1 is executed when the requester is known to the CoT; Workflow 2 is executed otherwise.

Workflow 1*: Assessment within the CoT*

This workflow (activities 1-4 in Figure 4) computes a trust value between two CoT members. This computation is based on the ratings originating from past business experiences that are stored in the trust matrix.

Activity 1: *Create the acquaintance graph:* Principals both inside or outside the CoT, are represented as nodes in an acquaintance graph, with directed edges representing trust relationships. The graph's structure relies on the (i) the $T_{i,j}(S)$ stored in the trust matrix for known nodes, or (ii) on the information provided by the neighboring nodes about the unknown node, with respect to a trust scenario S.

In FIM environments this information can be usually collected by means of SAML-Assertions. The resulting graph (Figure 6) may be divided into three *rings* reflecting members, direct neighbors and remainder (compare Section III, Figure 1).

Activity 2: *Breadth-first graph search:* To find the trust relationship between requester and respondent, we progress breadth-first [23] through the graph. The search, as illustrated in Alg. 1 begins at the respondent P_1 (root node) in Ring 1 and traverses the graph recursively (outward) and evaluates the trust values on the path between nodes P_1 and P_x (by means of the `computeTrust` function shown in Alg. 2). It continues until the sought requester node P_x is found or until it fails to find an edge.

Algorithm 1 Breadth-first search for requester P_x

```
traverseGraph(P_1, P_x, S)
(P_2, ..., P_n) := getNeighbors(S, P_1, P_x)
if getEdges(T_{P_2}, ..., T_{P_n}) then
    T_{P_1 P_x} := ComputeTrust(T_{P_2}, ..., T_{P_n})
    return T_{P_1 P_x}
    if (!(T_{P_1 P_x})) then
        for j = P_2 to P_n do
            traverseGraph(j, P_x, S)
        end for
    end if
else
    break
end if
```

In the current case, i.e. when the requester is a CoT member, only the direct neighbors need to be assesssed; they are identified by the function `getNeighbors()`.

Activity 3: *Computing trust values:* In previous work [3] we have laid the foundation for a function for computing T, by integrating the aspects of *Reputation Management* [16]. This involves the tracking of an entity's behavior within a federation and other entities rating that behavior.

Alg. 2 illustrates a function `computeTrust` : $P \times P \times S \mapsto T$ computes the prospective trust relationship between two principals P_1, P_x that are connected by direct neighbors.

EXAMPLE : Recall the example given in Section III, where the CoT includes 3 members `sp1` (in the respondent role), `sp2` and `idp1`, and `idp2` (in the requester role) is located outside the CoT. We start with a freshly initialized trust matrix ($T_{i,j} = 0.5$) and assume that trust value increments of 0.1 to be used. Similarly

Imagine that the content provider `sp1` finds his cooperation with `idp1` satisfactory. `sp1` manually raises `idp1`'s trust value $T_{sp_1 idp1}$ from 0.5 (the arbitrary, initial value) to 0.6 by rating the business relationship. Similarly for the relationship between `sp2` and `sp1`.

As a consequence, the trust value of the relationship between `sp2` and `idp1` is updated automatically. `computeTrust` is used to determine the new trust value. In this simple case we have only one neighbor, `sp1`. According to Alg. 2 (line 12ff), the value $T_{sp_2 idp1}$ is raised from 0.5 to 0.6, since $T_{sp_2 sp1} \geq T_{sp_1 idp1}$: $T_{sp_2 idp_1}(S_2) = \frac{T_{sp_2 sp_1} \cdot T_{sp_1 idp_1}}{T_{sp_2 sp_1}} = \frac{0.6 \cdot 0.6}{0.6} = 0.6$

Algorithm 2 `computeTrust`: Compute $T_{P_1 P_x}$

S_i := `EvaluateRequest`(P_x)
(edge, $T_{P_1P_x}[S_i]$) := `DirectTrans`(P_1, P_x, S_i)
if (edge) **then**
  $T_{P_1P_x}[S_i]$:= $T_{P_1P_x}[S_i]$
else
  $(P_2, \ldots, P_n) =$ `getNeighbors`(S_i, P_1, P_x)
  $((T_{P_1P_2}, \ldots T_{P_1P_n})(T_{P_2P_x}, \ldots T_{P_nP_x}))$:= `getEdges`$(T_{P_2}, \ldots, T_{P_n})$
  $T_{P_1P_x}[S_i]$:= 0
  M := 0
  N := 0
  for $j = 1$ to n **do**
    if $T_{P_1P_j} \geq T_{P_jP_x}$ **then**
      $T_{P_1P_x}[S_i]$:= $T_{P_1P_j}[S_i] \cdot T_{P_jP_x}[S_i]$
    else
      $T_{P_1P_x}[S_i]$:= $T_{P_1P_j}[S_i]^2$
    end if
    M := $T_{P_1P_x} + M$
    N := $T_{P_1P_j} + N$
  end for
  $T_{P_1P_x}[S_i]$:= $\frac{M}{N}$
end if
function `DirectTrans`(P_1, P_x, S_i)
if $\exists\ T_{P_1P_x}$ **then**
  directEdge := 1
  return (directEdge, $T_{P_1P_x}[S_i]$)
else
  directEdge := 0
  return (directEdge)
end if

*1) **Activity 4:** Update the Trust Matrix:* As illustrated in the example above, freshly computed trust values are stored in the matrix. Once one value has changed, all the other trust values will be recomputed accordingly. Though we model the update itself as an activity, it is in fact realized by the `computeTrust` function (see Alg. 2).

EXAMPLE The matrix in our example will be updated with automatically recomputed trust values as follows:

$$M(S_1) = \begin{pmatrix} - & T_{sp_1sp_2} & T_{sp_1idp_1} \\ T_{sp_2sp_1} & - & T_{sp_2idp_1} \\ T_{idp_1sp_1} & T_{idp_1sp_2} & - \end{pmatrix} = \begin{pmatrix} - & \mathbf{0.6} & \mathbf{0.6} \\ \mathbf{0.6} & - & \mathbf{0.6} \\ 0.5 & 0.5 & - \end{pmatrix}$$

Workflow 2: *Assessing requesters external to the CoT*

Consider the case when a requester P_x outside the CoT, having already transacted a business relation with a CoT member, for instance with P_2, wishes to cooperate with a CoT member, the repondent P_1. This workflow investigates, whether it is possible to use the first relationship (P_x–P_2) to support the second one (P_x–P_1). As can be seen in Figure 4, Workflow 2 is initially based on Workflow 1. It begins with Activity 0 in which the request from P_x is verified. In such a request, the requester has to specify his credentials (e.g. public key) as well as the trust scenario (i.e. the service) in the focus of the request.

If the requester is already recorded in the trust matrix, i.e. a direct edge between the principal P_x and one of the CoT members (e.g. in this case P_2) has been found, we continue with Activity 1 in Workflow 1. Activity 2 will then be applied recursively in *Ring 1* and *Ring 2* to compute the desired trust relationships (since P_x is outside the CoT).

In the final case, no edge exists between a requester P_x and a CoT member exists. Hence, by definition, P_x is located in *Ring 3*. We handle this situation in Activity 5 by means of fall back on authentication of the requester. Note that in this case no information about P_x's reputation is provided.

***Activity 5:** Search by Certificates:* This function can be helpful to get more information about the requester when all the neighbors do not possess a direct reputation about him but have some indirect relationships via certificates. This solution may imply trust for a specific service provisioning [11]. In this activity the following components are needed:

A value Set $C : \{C_1, C_2, \cdots, C_n\}$*:* of possible certificates of principals issuing requests to access the CoT.

`Cert_Search`*:* $P \times C \rightarrow T$ The request presented to the CoT that contains the requester public key, will be verified by `Cert_Search` to ensure this public key is signed by a third party *Certificate Authority*, who might be known to the CoT. This verification will be proceeded by means of the public key system used in the CoT, which has a list of *trusted CAs* together with the corresponding public keys, so that the digital signature can be verified. Some CAs are so known that they are included by default in many public-key systems [26].

The Algorithm 3 iterates until the algorithm identifies the nearest CA (in the certificate path) and computes its reputation. As a proper trust value pertaining to the principal P_x cannot be deduced, P_1 is provided with a confirmation of P_x's identity, the identity of P_x's nearest (in the key path) CA. The reputation of this CA may be computable by means of Workflow 1. Obviously, in this case the respondent has a much weaker basis for deciding about a cooperation with the requester.

Algorithm 3 `ExternSearch`: Estimate $T_{P_1P_x}$

for all P_{CA_i} such that $P_{CA_i} \neq P_{CA_{root}}$ **do**
  $i := 0$ and $j \in [1..n]$
  if $\exists\ T_{P_jP_x}$ **then**
    $T_{P_1P_x}$:= `ComputeTrust`$(T_{P_2}, \ldots, T_{P_n})$
    return $(P_x, T_{P_1P_x})$
  else
    P_{CA_i} := `Cert_Search`(P_x)
    if $\exists\ T_{P_jP_{CA_{i(P_x)}}}$ **then**
      $T_{P_1P_{CA_{i(P_x)}}}$:= `ComputeTrust`$(T_{P_2}, \ldots, T_{P_n})$
      return $(P_x, P_{CAi}, T_{P_1P_{CA_{i(P_x)}}})$
    else if $P_{CA_i} \in$ (Known_CA_List) **then**
      `identitfied`(P_x)
      return (P_x, P_{CAi})
    else
      $P_{CA_{i+1}}$:= `Cert_Search`(P_{CA_i})
    end if
  end if
end for

EXAMPLE In our scenario (see Figure 2) `idp2` has a relationship to `sp2`. We set $T_{sp_2idp_2} = 0.3$. Alg. 2 will compute to

$$T_{sp_1idp_2} = T_{sp_1idp_2}(S_1) = \frac{T_{sp_2idp_2} \cdot T_{sp_1sp_2}}{T_{sp_1sp_2}} = \frac{0.3 \cdot 0.6}{0.6} = 0.3$$

Now, imagine that the requester `idp2` from our previous example has no ties to CoT members. As before, `idp2` requests a cooperation with `sp1` by providing his public key as well as the targeted service (content provisioning). In this case, however, the reputation search method fails to provide a response due to the lack of reputation edges from `sp1` to `idp2`. In consequence, the function `Cert_Search` will be performed to identify the CA who signed `idp2`'s public key. The `computeTrust` function (see Alg. 2) facilitates the reputation search regarding the CA among the neighbors of `sp1`.

Activity 6: *Extend the Trust Matrix:* The extension of the trust matrix covers two cases: (i) Adding a new requester: a new column containing the computed trust value of the requester in the corresponding scenario is added. (ii) Adding a new scenario, as derived from the `Evaluate_Request` activity.

Activity 7: *Grant access:* The trust value determined for a requester (or, if appropriate, a mere authetication) serves as a basis for decision regarding his admission to the repondents' resources (e.g. access to a service). A respondent needs to apply local policies in order to make this decision.

EXAMPLE Given the scenario in Section III, imagine a requester `idp2.1` being a subdivision of `idp2`, who already cooperates with the respondent `sp2`. The respondent can determine the identity of the CA on the requester's public key and retrieve the reputation of `idp2` from the trust matrix. `sp2` can correspondingly decide to what extent trust the requester.

VI. Realization

We have developed a prototype that simulates the trust assessment scheme described in this paper, in order to support experiments with different configurations and scenarios. In the following, we discuss the current realization as well as future modifications.

Architectural components

Our architecture encompasses three base components that carry specific responsibilities, as shown in Figure 7: a trust broker, a policy engine and a storage facility for trust values.

a) Trust broker: The trust broker is a central component that implements the algorithms described in this paper as a set of Perl 5 libraries and functions.

When a respondent receives a request and wishes to rank the requester, he asks the trust broker to handle the request on his behalf. The trust broker executes the algorithms presented in Section V on a given CoT model and triggers update of the trust matrix.

b) Trust matrix repository: The trust matrix is stored in a Novell eDirectory Server (an LDAP implementation) due to integration plans with an existing FIM project (see Section VII-C3). The schema specifies attribute triples (scenario, principal, trust value) to represent the trust relationships according to Figure 5. A CoT member receives write permission on her own subcontainer in the directory tree, and read-permissions for other members' subcontainers.

c) Policy engine: A respondent decides on whether to grant access based on the results of the trust assessment workflows.

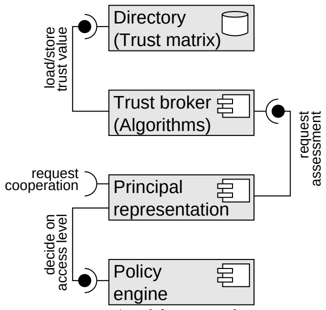

Figure 7. Architectural components

The integration of a policy engine supporting Attribute Release Policies (ARP) for IDPs and Attribute Acceptance Policy (AAP) for SPs is a work in progress.

d) Representation of a principal: A principal can issue requests for cooperation and—if he is a CoT member—act as responder to such requests. As the trust assessment algorithms are encapsulated in the trust broker, and the trust values are stored in a directory, a simulated principal merely needs to be able to issue and redirect requests.

VII. Discussion and conclusion

The approach presented in this paper complements the static nature of Circle-of-Trust environments by a set of new dynamic trust assessment mechanisms. They support the exploitation of the existing trust base within the CoT while avoiding interference with single CoT members' local policies.

The main thrust of the approach is towards a more relaxed trust concept allowing for dynamically deducible trust values. However, a number of constraints must be taken into account when considering dynamic trust systems.

A. Fulfilment of requirements

As discussed in Section III-B, we have sought a quick and cost effective way of setting up cooperations between CoT members and external organizations, without impacting third parties or compromizing the CoT's integrity. While the approach does fulfil the requirements, it does so at the price of incurring several risks in order to allow a trade-off between on the one hand flexibility, speed and degree of automation in the setup of cooperation agreements, and on the other hand the level of security and privacy attained in such cooperations.

B. Hazards

Traditional security mechanisms strive to achieve an unbroken chain of absolute trust. Our approach introduces flexibility at the price of the *quality* of the trust involved. In consequece, a number of hazards may emerge in the context of the approach.

1) Business risk: The level of trust that is necessary in order to effectively cooperate depends, in principle, on the operational risk involved in *trusting* a foreign party. Hence, the quantification of the trust level depends on the quantification of that business risk, which implies a dependence on the type, model and practices characterizing the business conducted by a given organisation.

2) Reputation feedback: In schemes as the one discussed in this paper, certain algorithms employed to refine trust values based on experience and feedback. Assuming these algorithms will not (rather: cannot) be kept secret, it is imaginable that rogue organizations may try to manipulate trust values by exploiting some characteristic of the algorithms. This may be countered by reasoning techniques or, more reliably, by human supervision.

C. Open issues and future work

We recognize several areas where the procedure presented in this paper can be improved. Some of them ensue from fundamental problems, while others would increase the ease of application.

1) Dealing with different trust scale semantics: In a distributed environment, we cannot assume that every participant will use the same software and the same scale of trust. When receiving query responses from peers (see Section V), it is critical to ensure that the level of trust stated by the peer is formulated on the same scale as is used by the requester. To achieve global calibration of the trust scale, we will formulate a *trust protocol* that allows queries and responses to be accompanied by meta-information.

2) Adjustment of trust values: For the regular, computed values, it makes sense to include a certain inertia regarding the change of trust levels: high trust levels should be difficult to achieve, while adjustment of trust levels close to an initial trust value should be more dynamic. Hence, we need a function that describes progressive effort for change in trust values.

3) Areas of application: Sections I, III suggest the support of flexible, pervasive service provisioning as an application of our approach. Another possible application area is within the *Virtual Universities (VU)* for web-based learning and education systems. The members of VUs are currently belonging to different groups of students, tutors, examiners, departments etc. They require assistence by a trust management model for securely sharing their e-learning materials and users' account data. We plan to integrate the approach in a distributed e-learning platform that is being developed in the IntegraTUM-Project at the Technische Uninversität München (TUM) (http://www.tum.de/integratum). More general e-commerce settings, as well as Grid-like environments may also benefit from a-priori trust assessment, as can services in the mobile telecommunications domain.

ACKNOWLEDGMENT

The authors wish to thank the members of the Munich Network Management Team and the IntegraTUM project team for valuable comments on previous versions of this paper. The MNM-Team, directed by Prof. Dr. Heinz-Gerd Hegering, is a group of researchers from the University of Munich, the Munich University of Technology, and the Leibniz Supercomputing Center of the Bavarian Academy of Sciences. The web server of the MNM Team is located at `http://www.mnm-team.org/`. IntegraTUM is a project funded by the German Research Foundation (DFG) aiming at the development and establishment of a seamless and integrated IT infrastructure for the Munich University of Technology (TUM). IntegraTUM is headed by the vice president and CIO of TUM, Prof. Dr. Arndt Bode (see `www.tum.de/iuk/cio/`). This paper was supported in part by the EC IST-EMANICS Network of Excellence (#26854).

REFERENCES

[1] M. Blaze, J. Feigenbaum, J. Ioannidis, and A. Keromytis. RFC 2704: The keynote trust-management system version 2. Technical report, September 1999.

[2] Feigenbaum J. Blaze M. and Lacy J. Decentralized Trust Management. In *IEEE Conference on Security and Privacy*, Oakland, California, USA, 1996.

[3] L. Boursas and H. Reiser. Propagating Trust and Privacy Aspects in Federated Identity Management Scenarios. In *Proceedings of the 14th Annual Workshop of HP Software University Association*, Leibniz Supercomputing Center, Munich, Germany, Juli 2007.

[4] Brickley and Miller. FOAF Vocabulary Specifications 0.1. http://xmlns.com/foaf/0.1/.

[5] Edd Dumbill. Usefulinc FOAF information, 2004.

[6] P. O'Connell E. Gray and C. Yong. Towards a Framework for Assessing Trust-Based Admission Control in Collaborative Ad Hoc Applications, 2002.

[7] Thomas Wason (Ed.). Liberty id-ff architecture overview v1.2. Liberty Alliance Specification, 2004.

[8] The Federation for Identity and Cross-Credentialing System (FIXS) Trust Model. FIXS Specifications (http://www.fixs.org/), March 2007.

[9] J. Golbeck. *Computing and Applying Trust in Web-based Social Networks*. PhD thesis, University of Maryland, College Park, MD, 2005.

[10] J. Golbeck and J. Hendler. Accuracy of metrics for inferring trust and reputation in semantic web-basd social networks. In *Proceedings of EKAW 04*, 2004.

[11] T. Grandison and M. Sloman. Using A Survey of Trust in Internet Applications. In *IEEE Communications Surveys*, pages 2–16, 2000.

[12] A. Herzberg and Y. Mass. Access Control Meets Public Key Infrastructure. In *IEEE Symposium on Security and Privacy*, 2000.

[13] W. Hommel and H. Reiser. Federated Identity Management: Shortcomings of existing standards. In *9th International Symposium on Integrated Management*, Nice, France, May 2005.

[14] Eric Vitiello Jr. A trust module for defining trust realtionships in FOAF. www.perceive.net/xml/foaf.rdf, 2002.

[15] J. Linn. Liberty Trust Models Guidelines V.1.0. Liberty Alliance Specification, 2003.

[16] H.D. McKnight and N.L Chervany. The Meanings of Trust. Technical Report 94-04, Department Carlson School of Management, University of Minnesota, 1996.

[17] S. Milgram. The small world problem. In *Psychology Today*, pages 60–67, 1967.

[18] A. Pashalidis and C.J. Mitchell. *A Taxonomy of Single Sign-On Systems*. Springer Berlin / Heidelberg, 2003.

[19] P. Raghavan R. Guha, R. Kumar and A. Tomkins. Propagating of trust and distrust. In *Thirteenth International World Wide Web Conference*, May 2004.

[20] Brendan Dixon S. Bajaj, G. Della-Libera and M. Dusche. Web Services Federation Language (WS Federation) Version 1.0., 2003.

[21] R. Philpott S. Cantor, J. Kemp and E. Maler. Assertions and Protocols for the Security Assertion Markup Language (SAML) V2.0. OASIS Security Services Technical Committee Standard, 2005.

[22] Secure Environments for Collaboration among Ubiquitous Roaming Entities. European Union Project IST-2001-32486.

[23] Ronald L. Rivest Thomas H. Cormen, Charles E. Leiserson and Clifford Stein. Introduction to Algorithms. pages 531–539. MIT Press and McGraw-Hill, 2001.

[24] Christine Varney, Hogan, and Hartson. Liberty Alliance Privacy and Security Best Practices v. 2.0. Liberty Alliance Specification, 2005.

[25] Hogan Victoria Sheckler and Hartson. Liberty Alliance Contractual Framework Outline for Circles of Trust. Liberty Alliance Specification, 2005.

[26] Internet X.509 Public Key Infrastructure Certificate and CRL Profile. http://www.ietf.org/rfc/rfc2459.txt.

[27] P. Zimmermann. The official PGP User's guide. In *MIT Press*, 1994.