Network Working Group
Internet-Draft
Intended status: Informational
Expires: 16 December 2013

D.W.Chadwick
University of Kent
16 June 2013

A Trust Model for ABFAB Trust Routers
<draft-dwc-abfab-trust-model-00.txt>

Abstract

Trust routers form an integral part of the ABFAB infrastructure for determining routes between IdPs and RPs and determining CoI membership. Since it is inherent in their name that they are to be trusted, this Internet Draft specifies a trust model for their operation, so that IdPs and RPs can rely on them to perform the tasks that they are trusted to perform. These tasks are:

- form a trusted route between an IdP and RP
- ensure that a community of interest (CoI)only has the members that it should have

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

This document is not intended to be an Internet Standards Track specification; it is published for informational purposes.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress".

Copyright Notice

to this document.

Table of Contents

1. Introduction

If trust routers (TRs) are to be trusted by their trustors, then it must be clear to the trustors on what basis this trust is established. A trust model provides such a basis for showing who trusts whom for what purpose. A trust model has been defined as "the defined trust relationships and orientation of the communities participating with (an) organization.." [1].

The trust model described in this document is concerned with trust in trust routers for two purposes:

i) to provide a trusted path between an RP and an IdP/AA, and
ii) to ensure that a community of interest (CoI)only has the members that it should have.

2. Basic Assumptions and Definitions

The authorisation model and assumptions that underlie this document are as follows:

i) The Service Provider, being the Relying Party, is the trustor, and therefore must be in control of, and decide, who it trusts.

ii) The RP uses an attribute based access control model, in which users are granted access to its resources bases on their identity attributes

iii) Attributes may be globally defined, e.g. visa attributes, or locally defined e.g. member of club X. Globally defined attributes are often specified in international standards and may be used in several different CoIs and federations. Their syntax and semantics are fixed, regardless of which Attribute Authority (AA) issues them. Local attributes are defined by their issuing AA and usually are only valid in the CoI or federation in which the AA is a member. For locally defined attributes the attribute authority (issuer) must be globally identifiable (in the CoI or federation). The attribute then becomes globally identifiable through hierarchical naming (AA.attribute). It is therefore assumed that all attributes can be globally recognised.

iv) The RP's software comprises a Policy Enforcement Point (PEP), Credential Validation Service (CVS), and Policy Decision Point (PDP). The PEP is the application specific code which traps users' requests and enforces access control decisions. The CVS is the component which takes as input the user's credentials, claims, attribute assertions (synonymous terms for the purpose of this document) and returns to the PEP a list of validated user identity attributes. The PDP takes the list of user attributes and returns an access control decision.

v) The RP trusts its CVS to correctly validate the user's identity attributes based on its policy and to discard untrustworthy ones.

vi) The RP trusts its PDP to correctly make access control decisions based on its policy and the user's valid identity attributes.

vii) The CVS may have a local trust policy specifying which AAs are trusted to issue which attributes, along with the metadata necessary to validate digitally signed assertions from these AAs. In this case trust routers are not needed. If a router falsely directs the CVS to a fake AA, then the CVS can tell this (by signature validation) and will discard all the attributes it issues.

viii) The CVS may have a local trust policy specifying which attributes to accept from a federation or CoI, but may rely on the federation or CoI to control its membership and ensure that all AAs are trusted to issue these attributes. In this case the CVS trusts the TRs to connect it to trusted AAs, and not to connect it to untrusted AAs. Note that the CVS is not able to validate (digitally signed) assertions from these AAs since it does not have any meta data associated with them. So the CVS has to rely on the TRs

to set up a trusted channel to an AA, via Diffie Hellman key exchange, then it does not need digitally signed assertions. The CVS should be able to trust everything that comes down the trusted path.

The TR trust model described in this document is based on the following assumptions:

A. **Each TR admin is absolutely in charge of his local TR and can** configure it how he wants to (though it may not work properly if he does not follow some set procedures). He does not need to trust anyone else, i.e. he is his own root of trust. He can determine which remote entities to trust, and how much to trust them.

B. **Trust relationships between TRs can only be mutual and not one way.** It makes little sense for one TR to trust messages from another TR but the other TR to not trust any message at all from the former. This is similar to the fact that a mutual trust relationship exists between an SP and an IdP in a federation.

C. **Trust relationships can be symmetrical or asymmetrical. Symmetrical** trust relationships are when each party trusts the other to perform the same set of actions. Asymmetrical trust relationships are when each party trusts the other to perform different sets of actions. An example of an asymmetrical trust relationship is that between an IdP and an SP in a SAML federation. An example of a symmetrical trust relationship is that between two peers in a group.

D. **All TRs are members of the same federation, and this is agreed at** initial configuration time.

E. **CoI membership information is dynamically propagated between TR** federation members.

3. The Trust Model

3.1 TR Roles

A TR may perform one or more of the following roles:

Master - a master TR is responsible for keeping the CoI information up to date in its associated slave TRs. A Master CoI gets all its CoI information from its administrator.

Slave - a slave TR gets all its CoI information from its master TR. A slave TR administrator has no further work to do after initially configuring his/her slave TR.

Peer - a peer TR gets its CoI information from both its administrator and from its other peer TRs. When it receives any CoI information it propagates this to all its associated peer TRs, unless it already has this information stored in its local database, in which case it does no further propagation.

3.2 Inputting CoI information

All CoI information originates from TR administrators. The trust model allows for one or many CoI members to input this information to their managed TRs. It is then propagated to all other TRs in the federation.

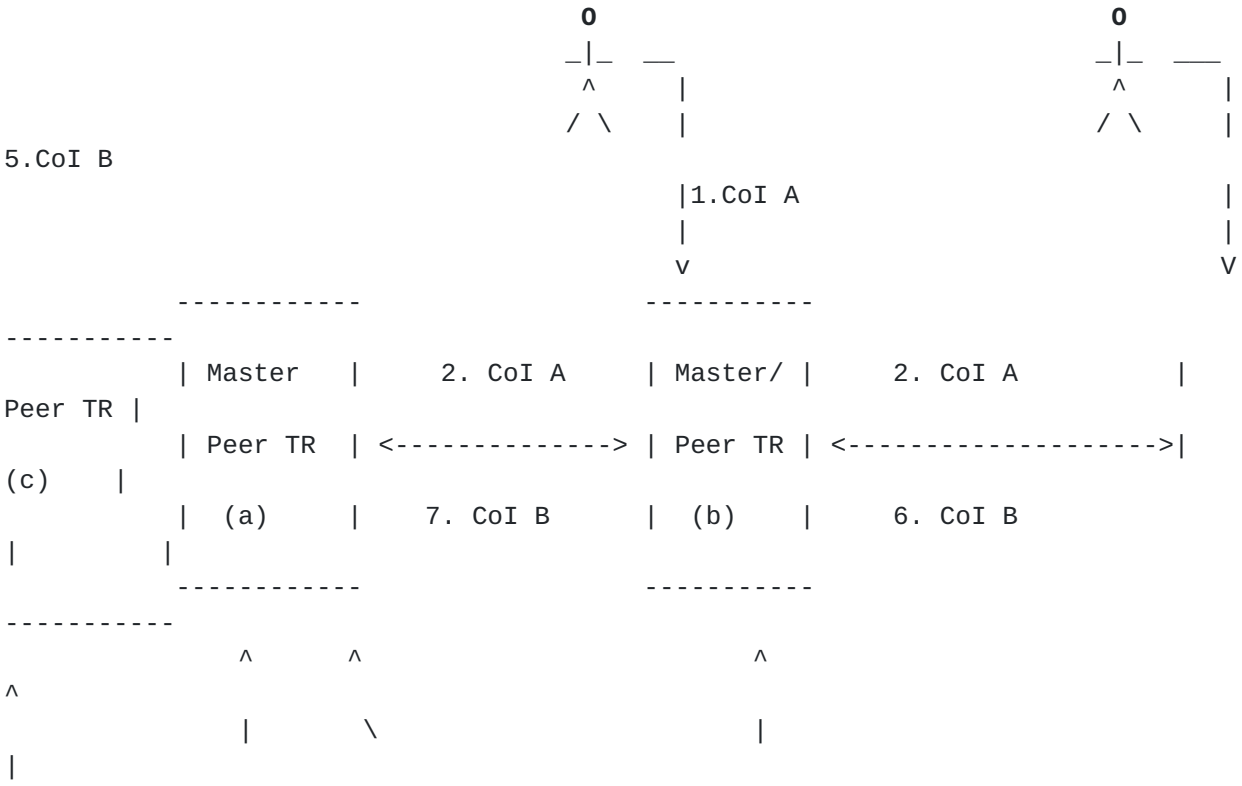
3.3 Handshake Protocol

When a TR is initially configured, it is provided with: the name of its federation, its role, its metadata, and the associated TR(s) along with its (their) metadata. It then establishes an active trust association with its associated TR(s), passing each one: the name of the federation, its role, its metadata, the role of the associated TR and its metadata. The receiving TR checks that the received information matches the information that has been configured into it by its administrator, and if all matches, it acknowledges that the trust association has been established. From then onwards each TR will trust any CoI information received from its associated TR, providing it agrees with the relationship type (master/slave or peer to peer).

3.4 CoI Distribution Protocol

Once an active trust association between two TRs has been established, the CoI CRUD protocol can take place. This allows a TR to create, read, update and delete the CoI information in its associated TRs.

3.5 An Example



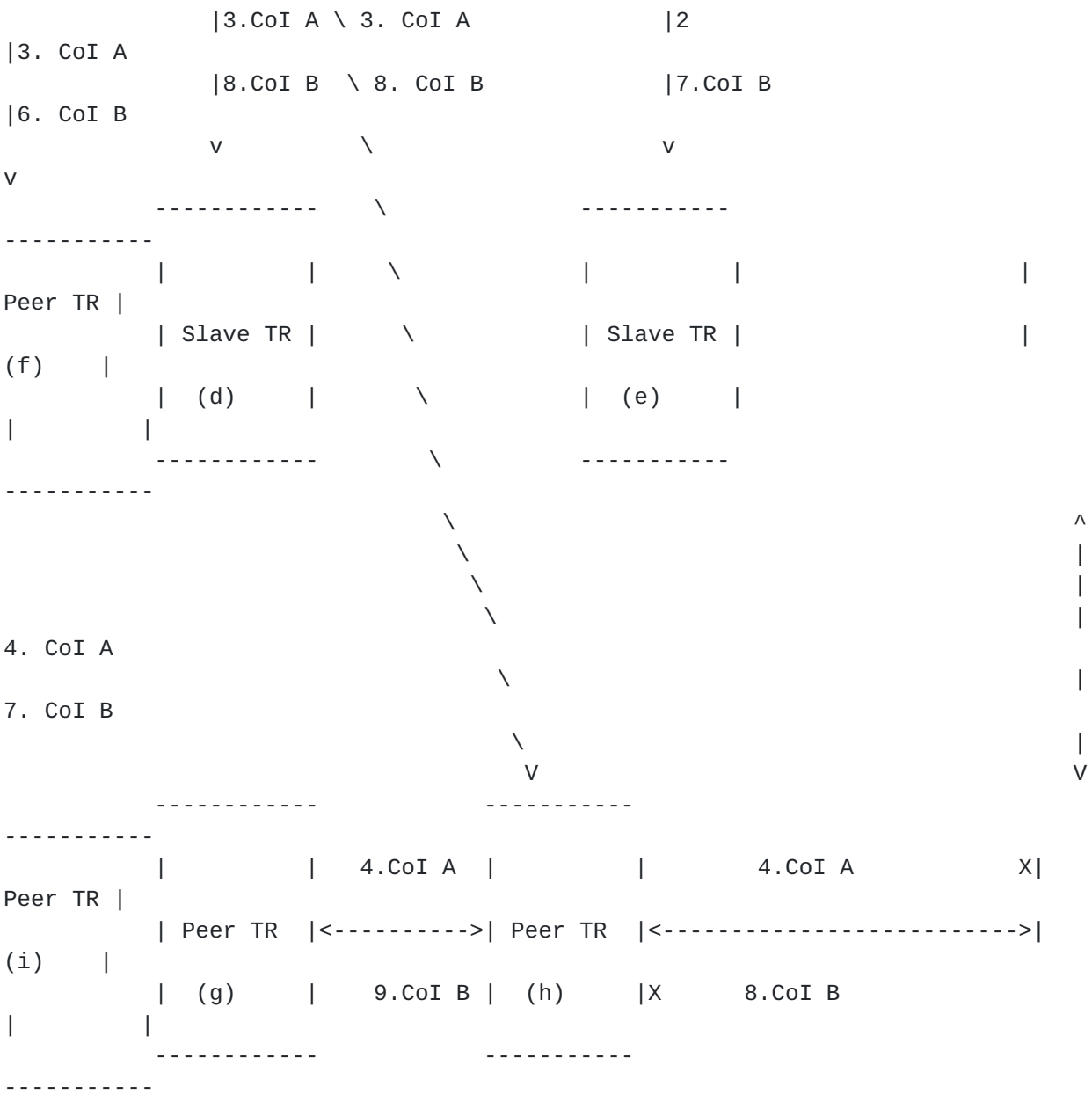


Figure 1. An example TR federation

We assume that a set of TRs in a federation have been configured with the following trust associations, as shown in figure 1:

i) TR(a) is the peer of TR(b) and the master of TR (d)
ii) TR(b) is the master of TR(e) and the peer of TR (c)
iii) TR(c) is the peer of TRs(b) and (f)
iv) TR(d) is the slave of TR(a)
v) TR(e) is the slave of TR(b)
vi) TR(f) is the peer of TRs(c) and (i)
vii) TR(g) is the peer of TR(h)
viii) TR(h) is the peer of TRs(g) and (i)
ix) TR(i) is the peer of TRs(f) and (h)

The administrator of TR(b) configures it with information about CoI A (step

1). TR(b) now propagates this information to all its peers and slaves (step 2). Any peers who receive this information now propagate it to their peers and slaves (step 3). This continues recursively until all TRs in the federation have received this information. Any peer who receives this information more than once from different peers discards the subsequent messages (step 4) (shown with an X in the diagram). The administrator of TR(c) configures it with information about CoI B (step 5). TR(c) now propagates this to all its peers (step 6). All peers who receive this information now propagate it to all their peers and slaves (step 7). This continues recursively until all TRs in the federation have received this information (steps 8 and 9), with peers discarding subsequent occurrences of the same message (shown with an X in the diagram).

4. References

4.1 Normative References

[BCP 78] S. Bradner, Ed. Et al. "Rights Contributors Provide to the IETF Trust" November 2008.

[RFC 2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", March 1997.

4.2 Informative References

[1] http://www.ocio.gov.nl.ca/ocio/im/glossary.html#Trust_Model

5. Author's Address

David W Chadwick
School of Computing
University of Kent
Canterbury
CT2 7NF
England

d.w.chadwick@kent.ac.uk