FIM shortcomings revealed

Michal Procházka, Luděk Matyska

CESNET, Czech Republic

21-22/6/2012

Outline

- Introduction
- Drawbacks of Identity Federations
- Experiences with SP in many Identity Federations
- Problems to be addressed
- Aditi—a solution?
- Conclusion

Introduction

- We target only on SAML based identity federations
  - Widely used in academic environment
  - Provides information about verified users
  - Own experience
- FIM deployment in production environment reveals problems
  - Technical (installation, configuration, ...)
  - Organizational (why we need an IdP?, support, ...)
  - Legal (what kind of information are we allowed to release, policies, ...)
- We would like to have answers on these questions:
  - Can we combine attributes from more IdPs?
  - Can user influence attribute release?
  - Does IdP really needs to maintain trust with SP?
  - Do we really need to take care of current legal issues?
  - Can we hide user activity from the IdP?

Drawbacks I.

- Authorization at the SP needs more information about user
  - Information in form of attributes are sent to SP
  - The attributes can reveal information that is considered private—issue with privacy laws
- Therefore, attribute release is controlled by IdPs
  - Users are not in full control of what is made available (they may restrict, but not selectively restrict and cannot extend the set of attributes released by IdP)
  - IdP must have some kind of formal (legal) relationship with SP

Drawbacks II.

- No proper scaling
  - Each IdP must know all SPs it serves or have third party which takes care of this relationship
  - Each SP must negotiate with each IdP, or with third party which is legally responsible, before users are allowed in
- Legal implications
  - IdP is legally responsible, because it really releases information about the user to the SP, IdPs don't want to play this role
  - User's consent given during authentication is considered not sufficient legally to free IdPs from legal responsibility of revealing personal information about a user
- User tracking
  - IdP has complete track of user's activity (which SP he/she accesses)

Drawbacks for users

- Users have to wait until negotiation between SP and IdP is finished
  - Usually takes several weeks
- Users cannot access SP when IdP is in error (not responding)
  - Also a problem for SP, as now their interaction with users is influenced by third party
- Information from IdPs cannot be combined
  - Insufficient attributes for a proper authorization
- Result: not user nor IdP/SP friendly

Our own experience

- 4 years running a service connected to 15 national identity federations
  - Atlases of pathology (`http://atlases.muni.cz`)
  - More than 25 000 registered users (more than 1 700 from IF)
- Huge administrative overhead
  - Check rules and conditions for each IF
  - Fill request (info about the service) per IF
  - Sign each agreement, keep track of changes (re-sign if necessary)
- Technical overhead as well
  - Each IF has different rules for keep data (selective cleaning, anonymization rules, ...)
  - Server certificates maintenance (different root CAs)
  - Needs to maintain local accounts at the IF
  - Compatibility issues
- In majority cases the SP needs to go through the test environment, which is good but time consuming
- Not a one-shot, but continuous maintenance, even if the only attribute needed is the eduPersonTargetedId

Problems to be addressed

- User control
  - Over information released by the IdP
  - Combining information about him/her
  - Being able to use service without waiting for decision of other parties
- Trust only where needed
  - Actually, only SP has to trust IdP, not vice versa
  - Remove legal liability of IdP
  - Trust between SP and user
- Support for automatic trust assessment
  - Trustworthiness of IdPs
  - (Eventually trustworthiness of SPs interesting for users)
- Hide user's activity from the IdP
- If possible, leverage current implementations as much as possible (SAML)

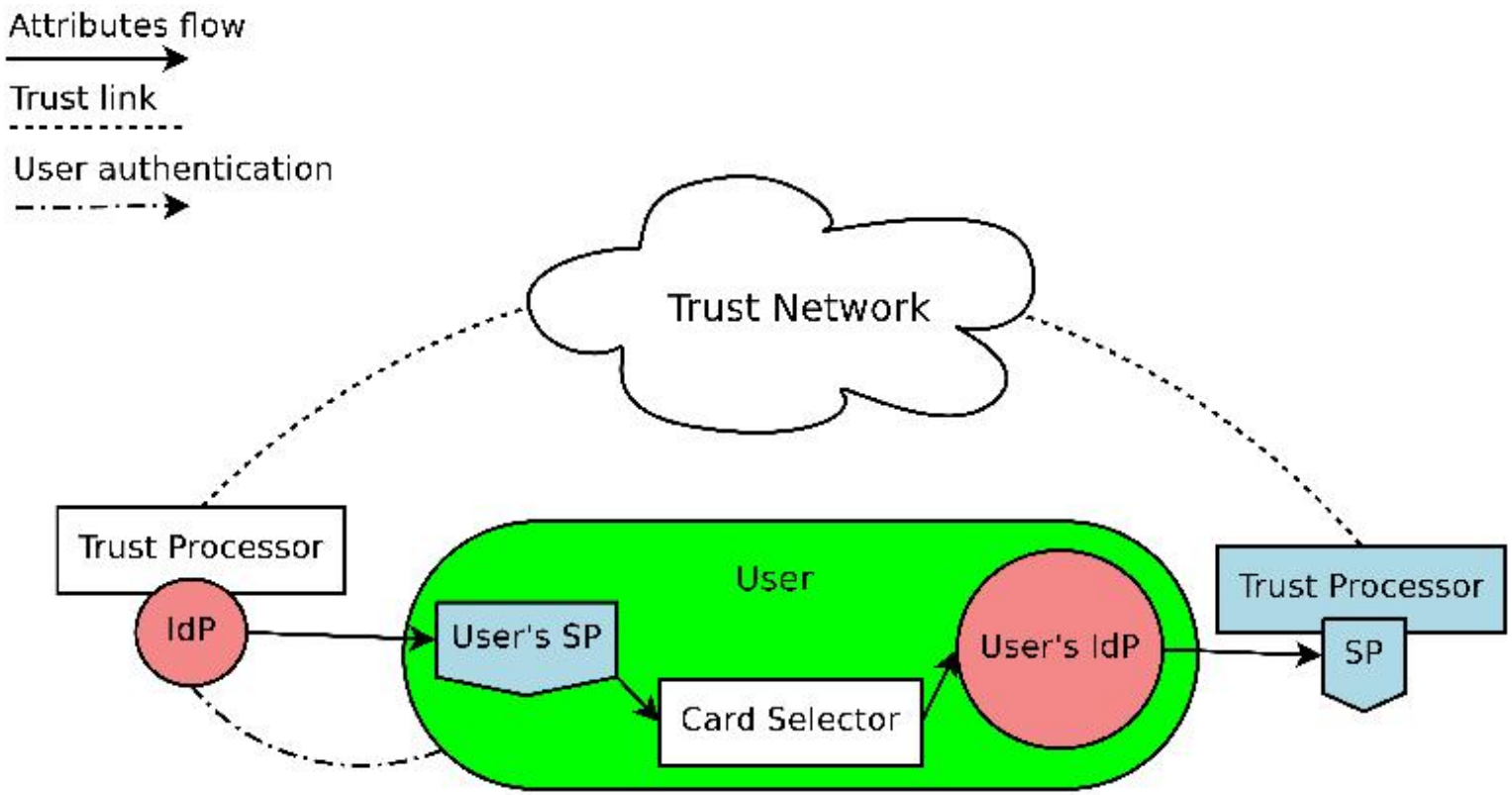
Solution?

- Aditi concept
  - To some extent similar to CardSpace or Higgins
  - Users in the center—they serve both as IdP and SP
  - Collect information from external IdPs (the SP role) into the cards
  - Create their own set of attributes (individual attributes signed by external IdPs or by user him/herself)
  - Serve as IdP to any external SP
- External IdPs communicate only with user (its SP)
  - No direct relationship with other SP
  - User's SP exceptional
    - No privacy issues—user ask about data on him/herself only
    - In fact, this also fulfill the standard requirement of giving user access to data kept about him/her in the institutional user management system
- User decides which attributes are revealed to external SP
  - Everybody is authorized to speak about him/herself
- Leverage current SAML2 based IF

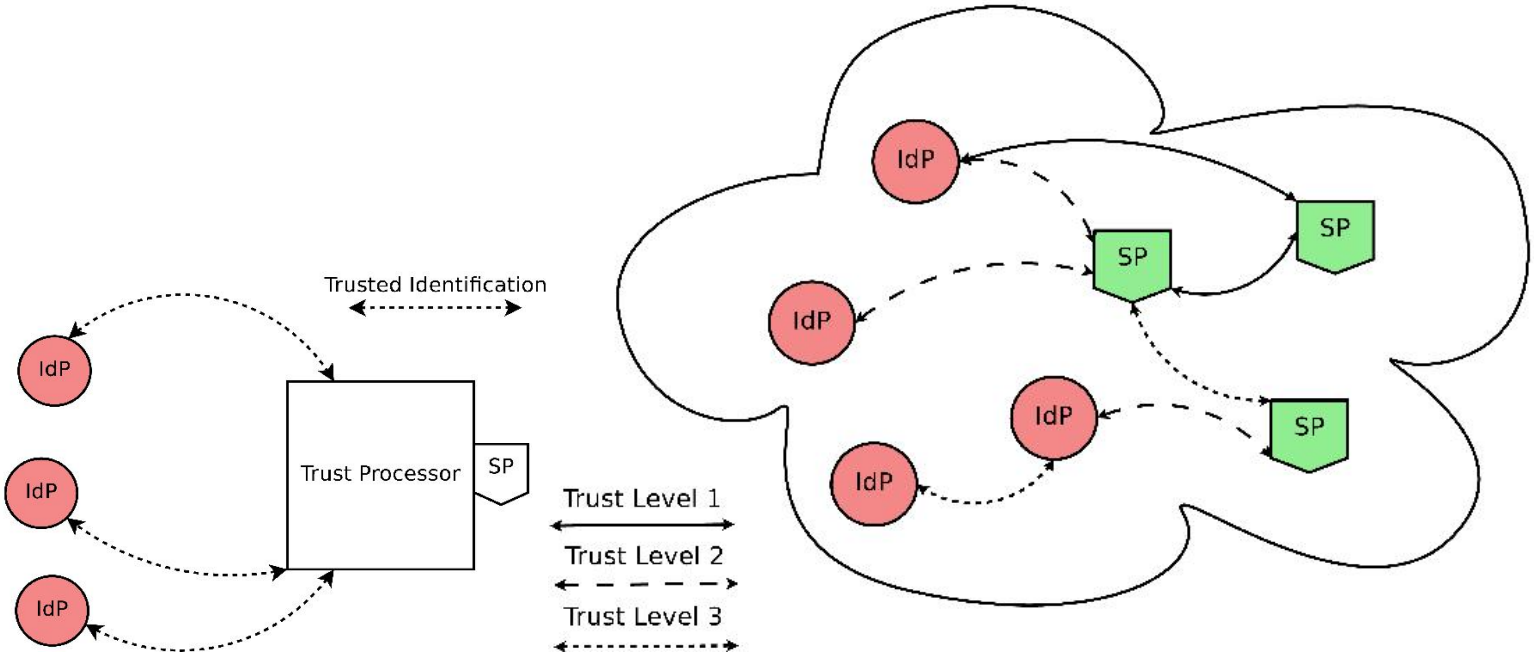
Trust

- User collects attributes from IdPs than know him
- Same attributes are released to external SPs
  - These SPs must trust original IdPs (whose signatures are under each attribute)
- Trust Processor at each SP
  - Maintains trust information about individual IdPs
  - Provides list of trusted IdPs gathered from SAML Metadata, manual configuration, PKI, ... (for its own decisions)
  - Optionally can operate Trust Network node (to reveal such information to other SPs)
- Trust Network
  - P2P based network used to share trust information about its members
  - Reputation based computation of the trust
  - Other models also possible

Aditi Design



Aditi trust and Trust Network



Card Selector

- User's way to keep the attributes
  - Locally or remotely (as a service)
- As SP, it collects attributes from external IdPs and stores them as Cards
- Cards could be combined, reshuffled, new cards can be created
  - Original IdP signatures are kept
  - Additional attributes could be provided by the user directly
    - Equivalent to an answer posed directly to user/customer
- As IdP, user uses the proper card to provide required attributes

Login Process

- User accesses SP (Aditi enabled)
- SP replies with AuthN request, specifying needed attributes
- User selects card which fits SP needs
  - User directly sees which attributes are needed
  - A new card can be created on the fly if none stored is appropriate
  - If some attributes are expired or not cached (never asked for before), proper external IdP (or several of them) is approached by user to get needed attributes
- User sends appropriate set of attributes back to the SP
- SP verifies each attribute independently
  - Uses Trust Network if some used IdP is not know to its Trust Processor

Complementary features

- Support of non-web applications
  - In parallel to operating SP and IdP, user can also operate credential transformation service
  - Federated identity $\Rightarrow$PKI, Kerberos, ...
- No cookies needed
  - Aditi can send full set of user's attributes (the card) with each request to SP
  - Any information has to be stored on the SP side
- Delegation
  - Supported by the Aditi concept
  - Attributes/cards can be signed by user and delegated
  - Limited lifetime for a card/set of delegated attributes lowers risks
  - Other scenarios possible, e.g. encrypting attributes/card by the public key of the target service

Open issues

- Link attributes from different IdPs together
  - SP could issue opaque identifier, which will be added to each released attribute
- Client application required
  - Could be in form of browser plugin or standalone application called by mimetype

Conclusion

- Current SAML based IF has several problems
- Aditi represents a new view on IF principles to overcome identified problems
- Complete user control over information asked and sent
  - User directly decides which SP to trust
- Easy SP provisioning
  - No need to sign any agreement with IdPs
- Trust only where really needed
  - Aditi helps SPs to assess/compute trust of each IdP
  - It's SP unilateral decision which IdP to trust

Thank for your attention. Questions?