Towards Practical Attribute-Based Identity Management: the IRMA Trajectory

Bart Jacobs
Institute for Computing and Information Sciences – Digital Security
Radboud University Nijmegen

Keynote, April 9, 2013, Royal Holloway, Univ. of London
irmacard.org

Outline

- Attributes instead of identities
- Practical realisation issues
- Demo
- Organisation of attributes
- Governance issues
- Conclusions

Identities versus attributes

- Identity management seems to revolve around identities
  - In practice this means uniquely identifying numbers, like social security number, or passport number
  - high-value targets for profiling & identity fraud

- But a more flexible identity ecosystem uses attributes
  - ‘student’, ‘doctor’, ‘president’, ‘top secret clearance’
  - ‘NL-citizen’, ‘resident of Nijmegen’
  - ‘home address’, ‘owner of bankaccount nr. . . .’

Your identity is the collection of attributes that hold for you

Key idea in attribute-based IdM

- Each transaction only requires a subset of your attributes for authentication
  - the subset should be small & proportional: data minimisation
  - this also offers some protection against identity fraud

- Attributes support contextual privacy
  - an essential aspect of privacy is being able to reveal different aspects of yourself in different contexts
  - attributes support such “partial identities” or “personas”

Identifying and non-identifying attributes

- In the attribute literature/tradition it is often (implicitly) assumed that attributes must be non-identifying
  - like: ‘female’, ‘over 18’, ‘UK citizen’ etc.
  - strong emphasis on privacy-friendly usage

- In our “IRMA” approach we deliberately also allow identifying attributes
  - like: “bank account nr.”, “social security nr.”, “client nr.”, or even “Facebook ID”
  - this greatly enlarges the usage & relevance & acceptance

- But this identifying usage is controversial
  - it enables tracking & tracing — which the technology is supposed to prevent
  - proportionality requirements need to be enforced — see later

- Non-identifying attributes good enough for many transactions:
  - a cheaper hair-cut for a student, or cheaper public transport for senior citizens
  - participation in local referendum for locals
  - buying games/books/videos online (over 16, or over 18)
  - participation in chatbox for minors (under 12, or 15)

- Attribute-based extends role-based access control
  - the captain of the ship can turn the ship’s wheel
  - very relevant in the medical sector (access to files)
  - in the military (or elsewhere): hierarchies/compartments/roles

- Typical transactions involve a combination of attributes
  - address, possibly with bank account, for pizza delivery
  - age + bank account for online gambling / XXX / . . .
  - “doctor” status + medical registration number for write-access to medical record
User-centric attribute issuance-usage model

Three main (cryptographic) systems

- **U-Prove** (based on blind signatures)
  - developed by Stefan Brands (Credentica), bought by Microsoft
  - specification available, under the Open Specification Promise
  - open source reference toolkits in C# and Java
  - multiple attributes in single (traceable) token, selective disclosure

- **Idemix** ("Identity Mixer", based on zero-knowledge proofs)
  - developed by Camenisch & Lysyanskaya, IBM Research Zürich
  - specs & sources also openly available
  - most properties, including revocation (by users, not by issuers)
  - most complicated (even "over-engineered")

- **Self-blindable certificates**
  - developed by Eric Verheul and others
  - uses bilinear pairings on elliptic curves
  - open implementation available

Nijmegen’s contribution

- Fast(est) smart card implementation for all three approaches, by Pim Vullers — see his own IdMan paper/talk
- **Practical realisation** initiative "IRMA", based on Idemix
  - not all Idemix features, emphasis on selective disclosure
  - with several (semi-public) partners: Surfnet, TNO, SIDN
  - active role in discussion about next eID in NL

- **Middleware development** to create eco-system for attributes
  - attribute verification, issuing, management; registration
  - integration in websites, NFC phones & tables, POS terminal
  - experimental attribute issuing via government website

- Small pilot for own "Kerckhoffs" master students (±100), starting soon.

Parallel initiative: **ABC4Trust**

- **ABC4Trust** is European FP7 research & development project (2010–2014, 12 partners)
- Development & implementation of unified common architecture that supports both U-prove & Idemix
- Two pilots (Söderhamn, Sweden & Patras, Greece) in fixed, educational setting
- Coordination with IRMA ongoing

Requirements for attribute-bases systems

- **Non-transferability**: my little nephew should not be able to get my "over 18" attribute (and go to XXX sites)
  - realised via binding to my private key

- **Issuer-unlinkability**: the issuers should not be able to track where I use which attribute
  - typically realised via blind(able) signature

- **Multi-show unlinkability**: service providers should not be able to connect usage (at different providers)
  - realised via zero-knowledge proofs, or via "self-blinding"

- **Revocation**: rogue attributes (via stolen/lost cards) should be blockable.
  - most difficult, partly in conflict with previous requirements

One may also have multiple issuers (government, banks, isp’s, ...)

- We see growing support for privacy-friendly attributes instead of one or more unique identifiers
  - Of course, the Google / Amazon / Facebook / Apple’s of this world just wish to trace people and don’t want such attributes

- **Hosting of attributes** is an issue in itself
  - usage of smart cards seems obvious, but there are alternatives
  - commercial interests play a substantial role
  - in the end, this matter is highly political ("information is power" and "architecture is politics")

- incentives & legal responsibility/accountability for issuers is unclear and is a delicate separate issue
Architecture II: three models

- **De-centralised / local**
  - attributes are stored under direct control of the user
  - smart card is obvious carrier; possibly also phone (see later)
  - direct interaction with verifier

- **Centralised**
  - attributes are stored in some (central) database; verification proceeds via this central infrastructure
  - single-point of failure, privacy-unfriendly (content & traffic)
  - still requires strong authentication of users
  - but: this allows putting a €-charge on each verification!

- **Pointer-based**
  - attributes remain with attribute-provider
  - slightly more privacy-friendly version: only traffic is visible

Unexpected support for "local", from the military

- Military IT-people expressed interest in attributes on smart cards, such as "colonel", "NL army", "military ID nr." etc.
- They are not interested in privacy; but they do like the decentralised character and robustness of the approach
- Imagine a NL colonel visits a military base in UK; there is no way that NL is allowing UK access to its (LDAP) database for identity/rank/clearance verification.

Not such a good idea!

- There is a personal cryptographic secret involved; cards have protected hardware, phones not yet (like IPT)
- Software on phones is becoming as unreliable as on PCs
- Phones are often changed/lost, or owned by employer
- Extra effort required to take your card out of your pocket for a security/privacy sensitive action is good: it brings you in a higher state of alert.

Pseudonyms or attributes

- The German eID Personalausweis is a high-tech card, distributed since 2010
- Germans take privacy seriously; eg. there is no national social security (identity) number for citizens
- After mutual card-terminal authentication, basic (unsigned) attributes (like name, address) can be exchanged
- There is support for PKI-based signature, but no certificate is loaded by default — the user should do that him/herself
- Domain-specific pseudonyms can be generated.

In comparison:

- attributes are more flexible & general (can contain pseudonyms)
- Idemix also allows domain-specific identifiers, eg. for one-time or long-term usage

Let’s see some running code!

- attribute verification
  - age bound and city, on NFC-enabled tablet
  - age bound for spicy website, using NFC phone as card reader
- Attribute issuing: student status
Example applications

- Junior age boundary: to order games/books/movies online, or to view/play certain content online (e.g., catch-up-TV, games)
  - offline, for buying alcoholic drinks or cigarettes in a shop — or even from a vending machine
- Senior age boundary: to get reductions, e.g., in public transport or in shops
- Both age ≥ 18 and country = NL: privacy-friendly wietpas for buying softdrugs (plans in NL abandoned)
  - the original wietpas was extremely privacy-unfriendly, and thus unpopular

Example credentials in medical sector

- For medical personnel: (IRMA card as staff-pass): credential with medical role (e.g., heart specialist, GP, nurse, pharmacist), registration number, etc. for access control to medical dossier
  Issued by: medical staff registry (BIG in NL)
- For patients: rudimentary medical dossier (known allergies, medicine usage) for ER usage.
  Issued by: e.g., GPs or hospitals

Example credential: address

- Address
  - Country
  - City
  - Street + Number
  - Postal code

Issued by: public authorities (e.g., local, but not in UK)
- Name is not included, stored elsewhere (no credential overlap)
- Expiry info is omitted

Example credential: age boundaries

<table>
<thead>
<tr>
<th>Junior bounds</th>
<th>Senior bounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>≥ 12</td>
<td>≥ 60</td>
</tr>
<tr>
<td>≥ 16</td>
<td>≥ 65</td>
</tr>
<tr>
<td>≥ 18</td>
<td>≥ 70</td>
</tr>
<tr>
<td>≥ 21</td>
<td>≥ 75</td>
</tr>
</tbody>
</table>

Issued by: public authorities
- Note: these attributes never expire, unlike for ≤.
- In Idemix bounds can be derived from the date-of-birth, in a costly, slow manner; they can also be included directly, like above.

Example credential: student card

- Student card
  - University / College
  - Field of study
  - Student ID
  - Enrolment year

Issued by: universities
- (Again: name is stored elsewhere: no overlap)
Example credential: citizen identity

<table>
<thead>
<tr>
<th>Name</th>
<th>Identity</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Family name</td>
<td>Social security number</td>
<td>Main entrance</td>
</tr>
<tr>
<td>First name</td>
<td>Date of birth</td>
<td>Parking</td>
</tr>
<tr>
<td>Full first names</td>
<td>Place of birth</td>
<td>Vault</td>
</tr>
<tr>
<td>Initials</td>
<td>Gender</td>
<td>Intranet</td>
</tr>
</tbody>
</table>

Issued by: public authorities

Example credential: company access

<table>
<thead>
<tr>
<th>Festival</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Festival name</td>
<td>Main entrance</td>
</tr>
<tr>
<td>Validity date</td>
<td>Parking</td>
</tr>
<tr>
<td>Pre-paid consumptions</td>
<td>Vault</td>
</tr>
<tr>
<td>Ticket number</td>
<td>Intranet</td>
</tr>
</tbody>
</table>

Issued by: e.g. company itself, or third (commercial) party

Example credential: festival/concert ticket

<table>
<thead>
<tr>
<th>Festival</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Festival name</td>
<td>Main entrance</td>
</tr>
<tr>
<td>Validity date</td>
<td>Parking</td>
</tr>
<tr>
<td>Pre-paid consumptions</td>
<td>Vault</td>
</tr>
<tr>
<td>Ticket number</td>
<td>Intranet</td>
</tr>
</tbody>
</table>

Issued by: e.g. festival itself, or third (commercial) party

The role of Issuers

- Users can obtain new/updated credentials from Issuers, either online, or offline
- Issuers first authenticate Users, and then make valid attributes available for download
  - others (“Verifiers”, “Relying Parties”) trust these attributes
  - issuers are thus trusted parties
  - authentication can be based on existing attributes
  - “download” is in fact interactive credential creation, with card presence, by sending a one-time code via texting
  - authentication can be based on existing attributes
- Issuers should publish how they authenticate and why they believe that the attributes they provide are valid (think of Facebook as issuer)

Example issuing: mobile phone number

- Imagine your MNO issues mobile phone number attributes
- A User goes to this MNO issue website (https!) and provides:
  - Name + date of birth, via IRMA card
  - phone number — simply by typing it in
- The MNO checks:
  - these data are consistent with an existing contract
  - phone presence, by sending a one-time code via testing
- Upon seeing the correct code within the same ssl-session, the MNO issues the phone number credential to the IRMA card

Example issuing: bookstore membership

- Imagine a bookstore wishes to issue membership attributes
- Upon presenting an IRMA card in the shop, a credential is issued to the card, stating “member status” (gold / silver / bronze) and “member number” (pseudonym) and “issue date”
- At the checkout (different) reductions can be obtained via the combination of attributes:
  - “member status” or “member number”
  - “member number” + “student”
  - “member number” + “senior citizen”
- This issuing involves no authentication
- The bookstore can build up historical profiles, based on the membership number, which can be used for additional offers (compare to pseudonyms on German card)
Root credentials

**Definition:** A credential is called a root credential if in its issuing process no other credentials are used for user authentication.

**Examples:**
- The MNO phone number credential is not 'root', since it relies on names + date of birth attributes existing on the card
- The bookstore credential is 'root'

- A root credential can be the root of a tree of other credentials that rely on it
- There may be multiple such trees
- Such tree structures should be public, for transparent trust

Trees and spheres

**Interesting question:** should Facebook have its own root credential, or can it depend on others (like name, or date of birth)?

- If NO, then your Facebook credential will be part of an existing 'identity tree'
  - a Facebook credential will then be linked to your real identity
- If YES, then Facebook credentials start a new tree, representing your 'Facebook life'

Which spheres/personas should be (dis)connected?

- who should decide this, and on which grounds ... ??
- the identity ecosystem is of great social/political importance

Control over what?

- All IRMA software is open source, so anyone can put it on smart cards, and start distributing cards
- A closed scheme is possible via cryptographic keys, eg:
  - cards contain public keys of some authority
  - card readers (terminals) need certificates signed by this authority — before cards communicate with them
- Such keys give control, both over issuers and over verifiers
  - if they don’t follow the rules, they can’t participate

IRMA governance

- At this early stage there is no real separation of roles
- Ideally, in the future, an independent foundation runs the scheme
  - commercial interests and public trust don’t mix well
- Independent foundations are probably least controversial for running large, open IT-infrastructures
  - eg. DNS in NL

In the remainder of the discussion I assume there is such an independent IRMA foundation
Attributes instead of identities

Practical realisation issues

Demo

Organisation of attributes

Governance issues

Conclusions

Radboud University Nijmegen

Attribute access for verifiers

- **Recall:** the risk is that verifiers read too many attributes

Governance model

- prospective verifiers register with IRMA foundation, stating their goals & requesting access to certain attributes
- if the request is proportional, access is approved
- verifier obtains certificate capturing access to these attributes
- IRMA cards check such certificates first, before they reveal any attributes

Attribute access for issuers

- **Main risks:** weak attribute validation, or excessive subject verification requirements (unnecessarily linking spheres)

Governance model

- prospective issuers register with IRMA foundation, stating which attributes they wish to issue and how they do the necessary validation & subject verification
- if the attributes are “useful” & reliable, and verification is proportional, read & write access is approved
- issuer obtains necessary certificates

Card-to-cardholder binding

Two mechanisms:
- Photo of cardholder
- PIN

Card view

Front
- There is **only a picture** on the frontside, nothing else
- There is a (random) **card number** on the back, which is:
  - not present inside the (chip in the) card
  - useful for “lost-and-found” scenarios
(The card has a randomised UID)

Back

PIN issues

Each card comes with **two PINs**

- **One for attribute reading**
  - Which attributes should be protected by PIN?
  - Balance between: ease-of-use, ease-of-abuse, confidentiality
  - over 18: yes, medical data: no (restrict read-certificates)

- **One for personal card management**
  - card owner can manage own attributes on card (like apps on phone)
  - also access to card logs

User convenience

- **User convenience is not** an explicit goal
- message: security and privacy require careful behaviour
- users will have to be conscious about what they are doing
- using your IRMA card should give the same alertness as in using your ordinary keys.
Own pilot project plans (this year)

- ±100 security master students can get IRMA card
  - request via national student authentication (Surfnet)
  - requires photo upload and email (for communication only)
  - face-to-face handover, with additional authentication

- Additional attributes: ba/ma/phd, university, study, name, address, town, country, age bounds, . . .

- Application scenarios: free printing, cheaper coffee, goodies on website, . . .

- Students are encouraged to test security and develop additional application scenarios

- Goals: fine-tuning & learning about practical challenges

---

eID developments in NL

- NL is late — through legal fight over earlier tender
  - delay may actually be an advantage
  - existing PKI cards are hardly used in practice (except Estonia)

- NL has social security number
  - but its usage is restricted to the public sector
  - it forms the basis for much-used national authentication system (DigiD), based on password and/or OTP via text message

- Two (main) eID requirements in NL:
  - strong authentication, based on smart cards
  - usable both in the public and private sector

- Two important options:
  - copy German card, using pseudonyms
  - introduce IRMA cards, using attributes

Option 1 is safest bet, but option 2 is most flexible

---

Main points I

- Important step towards practical use is: also allow identifying attributes — even though they go against the spirit

- IRMA work is based on smart & fast Idemix implementation
  - approach offers privacy & security, much user control
  - also development of open middleware

- Not just pushing the technology, but also pushing the management part
  - requires looking at the broader social/political picture

---

Main points II

- Scaling-up attribute use requires carefully designed & controlled identity eco-system
  - attributes form delicate dependency trees
  - governance preferably via independent foundation
  - incentives & accountability for issuing unresolved

- Open character can be innovation motor, leading to many, now unforeseen, applications.

- Recommended next step: organisation of large scale pilot, with ten thousands of users, like in university pass, or city pass.

- This technology gives policy makers & regulators the tools to enforce privacy & security by design!

---

Thanks for your attention. Questions/remarks?