Rule based Trust management using RT

Sandro Etalle
Overview

- Rule vs Reputation based TM
- A taste of trust negotiation
- The RT TM system
- Credential Chain discovery
Trust Management, the bottomline

• Typical access control mechanism

subject → shows → ID → lookup → authorization

• TM alternative

subject → has → credentials → infers → authorization
Source of confusion: terminology

• The term Trust Management is heavily overloaded.
  – There are several interpretations in the literature for this

• Two interpretations deal with here are:
  – “Reputation based” TM (which is the most common interpretation)
  – “Rule-based” TM (which is the one we’ll focus on most)

• There are several other interpretation possible:
  – the “trusted computing base”
  – the “trust” you have in data (usually a question of integrity and authentication)

• Bottomline: “trust” is one of the most misused word you are going to run into.

• Next: Reputation-based vs Rule-based Trust Management
Reputation-based TM in the real world by example

- Consider a community of cooks (70 people), you belong to
- On one day, you need to cooperate with someone you don’t know, for instance to prepare a dinner
- How do you find out whether you can trust this guy and what he does?
  - you ask your friends
    - and friends of friends
    - ...
  - some recommendations are better than other
  - you check the record (if any)
- If, at the end of the work the person has done a good job then,
  - trust in him increases
  - also the trust in those who have recommended him increases
Reputation-based TM in the virtual world by example

• Now you belong to a community of open-source developers, or of hackers (hundreds of people, scattered all over the world)
  – people in the community exchange programs & scripts
  – they build programs together

• One day, need to interact with someone you don’t know,
  – e.g. because you need a piece of code and this person can provide it for you.

• Your problem: can you trust the guy’s code?
  – you could analyse the code, but that is time consuming; if you trusted the guy, you would accept his code without many questions.

• Your problem becomes: can you trust the guy?

• difference with concrete community:
  – larger, so trust establishment has to be to some extent automatic
Reputation-based TM: salient features

• It is an open system
  – people come and go, which translated means that there are different security domains

• Trust is a measure
  – (you can trust someone *a bit more* than someone else)

• Trust changes in time

• There is an essential risk component
  – by trusting someone you *take a risk*

• Recommendation based (NOT identity-based)

• Peers are not continuously available

• Some systems:
  – PGP,
  – EigenTrust Algorithm (Stanford)
Rule-based TM in the real world by example

- Bart is a student of – say - GMU
- GMU has a bookstore
- Bart is entitled to a discount at the bookstore
  - because he is a student of the local university
- To obtain his discount, Bart shows his GMU ID
  - that is a credential
- The cashier
  - knows how GMU ID card look like
  - knows that GMU students are entitled to a discount
  - can easily check the photo on the ID
  - (can also check a bit the integrity of the ID, meaning it can distinguish to some extent a forged ID from a genuine one)
Rule-based TM in the virtual world by example

- Bart is about to rent a car at a major international companies.
- Bart might be entitled to a discount.
  - For instance because he is a “silver frequent flyer”, or because he has a “gold americanexpress”
- There are probably hundreds of ways he could be entitled to a discount. Finding them is not easy.
  - I checked e.g. the site of Hertz (14/9/2015) and found no information.
  - I checked the FD site
Zillions of credentials

- There are hundreds of reasons why Bart might be entitled to do something (e.g. get a discount)
  - If he is an employee of any governmental organization
  - If he is a member of the library club
  - If he is a veteran
  - ...

- Too many to mention
  - there is no way you can memorize them

- How can he possibly find out whether he is entitled to a discount?

Problems
  - scalability
  - knowing where to look
Summary: reputation- vs rule-based TM

Reputation-based TM
- open system (different security domains)
- trust is a measure & changes in time
- risk-based
- no delegation
- recommendation based (NOT identity-based)
- peers are not continuously available
- scalability

Rule-based TM
- open system (different security domains)
- trust is boolean & less time-dependent
- no risk
- delegation
- rule (credential) based (NOT identity-based)
- peers are not continuously available
- scalability
Reputation systems: a success and a failure

• Reputation systems in the underground economy.

• In the economy of cybercriminals reputation is everything
  – They cannot call the cops
  – They cannot rely on the mafia-like threats (if you don’t deliver, I’ll eliminate you), because they don’t know where they are

• If the reputation system is badly implemented => the market collapses

• If the reputation system is properly implemented => it florishes
  – L. Allodi, M. Corradin, and F. Massacci. Then and now: on the maturity of the cybercrime markets (the lesson that black-hat marketeers learned). IEEE Trans. on Emerging Topics in Computing, PP(99), 2015. [for a case t
FROM NOW ON WE CONCENTRATE ON RULE-BASED TM

LET’S WORK OUT AN EXAMPLE
Runtime example

• To have a simple example to work with, we use a slightly modified version of the bookstore example.

• Our bookstore serves all students of all universities, and adopts the following policy
  – “Every student of an accredited university is entitled to a 10% discount”

• Bart wants to prove he is entitled to a discount
  – He shows his GMU student ID
But, is GMU accredited?
Accredited by whom?
The shop needs to specify this, and there are two ways to do so
- (a) the shop could maintain its own list of accredited universities, or
- (b) refer to an external “accreditation bureau”
Of course, (b) is “better” than (a), as it is not a shop’s duty to maintain a list of accredited universities.

So, now to get the discount, Bart needs to show his University ID and “additional evidence” showing that the University he goes to is accredited.
- we now have a “chain of credentials"
The Chain of Credentials

- We have a *chain of credentials*
- The **subject** of one is the **issuer** of the other one
Credential

- A credential is a statement
  - Issued and signed by the **issuer**
  - about a **subject**
  - Containing information about the subject

- Requirements
  - Unforgeable (!)
  - Verifiable (that it belongs to the one asking for the service)
  - Signed (e.g. X509)
  - But most of all....

- **A well-defined semantics**
The Semantics Problem

• **Anyone** may issue credentials.
  – for instance, any University may issue student credentials (as well as employee credentials etc.)

• It is important that there is an agreement on the **semantics of statements** (well-defined semantics)
  – the definition of “student” should be the same for all university issuing student credentials
  – This is by far not trivial. For instance the definition of “professor” changes according to the country. The same applies to the definition of surgeon and to most work-related definitions.
  – Here we assume that there is a “universal” agreement on the meaning of words (otherwise we get crazy).
The Confidentiality problem

• We have seen some standard credentials:
  – credit cards, student IDs, etc
• Other “credentials” you have are more subtle.
  – the medical record is a credential, as it allows you to get the treatment you are entitled to,
  – your criminal record is a credential, because (if clean) it gives you access to jobs
  – your school record....
• These credential are confidential,
Security Monitor and Trust Negotiation

• No predefined security monitor
  – Needs a well-defined semantics
  – Credentials need to be disclosed to a possibly untrusted party
  – **ISSUE 1: Trust negotiation**

• Credentials are *distributed*
  – stored by the subject AND/OR by the issuer
  – **ISSUE 2: credential chain discovery**
Security Monitor and Trust Negotiation

• Question: who is checking the validity of the credential in our bookstore example?
  – the bookstore does
  – each bookstore has its own “security monitor”

• => There is no predefined (central) security monitor we all agree on
  – This is a crucial feature of Trust management.

• But this means that (possibly confidential) credentials needs to be disclosed to possibly untrusted parties.

• This requires Trust Negotiation

• Let us see what we are talking about
A flavor of trust negotiation

- Credentials may contain private information
  - E.g. medical record
- Credentials should be disclosed only according to a given access control policy
  - “I will show my medical record only to accredited surgeons”
  - This is decided by the user
- To disclose a credential one requires to see another credential
  - a credential belonging to the principal that is asking for your own credential
Example of Trust Negotiation

- A: please send me this treatment (request)
- H: I’ll do so if you show me your medical (policy)
- A: I’ll show you my medical if you show me that you subscribe to GoodPrivacyPolicies
- H: Here is a credential showing this.
- A: here is my medical
- H here is the treatment.
Trust Negotiation

- Seamons: “The process of establishing trust between strangers in open systems based on the attributes of the participants”
- Goal: establish trust while maintaining privacy
- How: by iterative disclosure of credentials
- Additional problems:
  - circularities
  - what do you do with the info in a credential after it has been disclosed
Part 2

The RT family language
Policy Language Wish List

- **Decentralize** authority to define attributes
  - Utilize policy and credentials from many sources
  - (in our example: any university may define who is a student)

- **Delegation** of attribute authority
  - To specific principals
  - To principals with certain attributes
  - (in our example: the bookstore delegates to the accreditation bureau the “task” of accrediting a university)

- Some advanced features we may want
  - Intersection of attributes
  - Parameterization, constraints
  - Support for thresholds, separation of duty
Role-based Trust Management (RT)

- A family of credential / policy languages
  - Simplest, $RT_0$, has no parameterization, thresholds, or separation of duty [Li, Mitchell, Winsborough]

- $RT_0$ example: student discount subscription

```
- EPub.studentDiscount ← StateU.student
- StateU.student ← Alice
```

- Role name
- Credential
- Principal
- Role
RT0 Syntax

- A, B, D: principals
- r, r1, r2: role names
- A.r: a role (a principal + a role name)

- Four types of credentials:
  - A.r \leftarrow D
  - A.r \leftarrow B.r1
  - A.r \leftarrow A.r1.r2
  - A.r \leftarrow A1.r1 \cap A2.r2
**Type 1 credentials**

- Epub.discount ← Alice

- Epub states that “Alice belongs to the role Epub.discount”
- Semantics Alice ∈ [[Epub.discount]]
- Issuer: Epub
- Subject: Alice

- Side question: where is this stored? We don’t know. Yet.
Type 2 credentials

- Epub.discount ← StateU.student

- Epub states
  - “If StateU states that X is a student then I state that X gets a discount”
- Operationally:
  - “anyone showing a student certificate signed by stateU gets a discount”
- Epub delegates authority to StateU
- Semantics \([\text{[StateU.student]}] \subseteq [\text{[Epub.discount]}]\)
- Issuer: Epub
- Subject: StateU
Type 3 credentials

• Epub.discount ← AccredBureau.university.student

• Epub states
  – if AccredBureau states that X is an accredited university and
  – X states that Y is a student
  – then I state that Y gets a discount.

• “attribute-based delegation”

• Semantics
  – For every $X \in \{\text{AccredBureau.university}\}, \{X.\text{student}\} \subseteq \{\text{Epub.discount}\}$

• Note:
  – like in SDSI, but links are of length max 2 (does not affect expressivity)
  – In the original RT0 the subject and the issuer are supposed to be the same
Type 4 credentials

- \text{ITbizz.maysign} \leftarrow \text{ITbizz.manager} \cap \text{ITbizz.senior}

- ITbizz states that “… senior managers may sign.”
  - “anyone showing a manager certificate and a senior certificate (both signed by ITbizz)) may ‘sign’”

- Semantics
  - \[[\text{ITbizz.manager}] \cap [\text{ITbizz.senior}]] \subseteq [\text{ITbizz.maysign}]

- Issuer, subject: …
Summary:

- A, B, D: principals
- r, r1, r2: role names
- A.r: a role (a principal + a role name)

- A.r ← D  Role A.r contains principal D as a member
- A.r ← B.r1  A.r contains role B.r1 as a subset
- A.r ← A.r1.r2  A.r ⊇ B.r2 for each B in A.r1
- A.r ← A1.r1 ∩ A2.r2  A.r contains the intersection

- The first 3 statement types: equivalent to pure SDSI
- Notice the higher-order flavour of A.r ← A.r1.r2.
- More complex versions have parameters (RT1), constraints (RTC), and can model thresholds and separation of duty (RTT)
Exercise: find the semantics

- Alice.s ← Alice.u.v
- Alice.u ← Bob
- Bob.v ← Charlie
- Bob.v ← Charlie.s
- Charlie.s ← David
- Charlie.s ← Edward
Solution

- Alice.s ← Alice.u.v
- Alice.u ← Bob
- Bob.v ← Charlie
- Bob.v ← Charlie.s
- Charlie.s ← David
- Charlie.s ← Edward

- [[Charlie.s]] = \{David, Edward\}
- [[Bob.v]] = \{Charlie, David, Edward\}
- [[Alice.u]] = \{Bob\}
- [[Alice.s]] = \{Charlie, David, Edward\}
Other exercise

- The flexible company FC delegates the definition of buyer to any of its territorial divisions FCDiv1,..., FCDivN
- FC uses the role FC.division to list all the territorial divisions.
- FC has accountants (role FC.accountants), which must be approved by Accrinst, and must have a certification as controller given by FedCert.
- Alice is both a buyer and an accountant.
- Write an RT0 set of credentials for this.
Solution

- FC.division $\leftarrow$ FCDiv1
- ...
- FC.division $\leftarrow$ FCDivN
- FC.buyer $\leftarrow$ FC.division.buyer
- FCDiv1.buyer $\leftarrow$ Alice
- FC.accountant $\leftarrow$ Accrinnst.approved $\cap$ FedCert.controller
- Accrinst.approved $\leftarrow$ Alice
- FedCert.controller $\leftarrow$ Alice
Conclusions

• Context:
  – 2 or more parties in an open system.
  – parties are not in the same security domain.

• Goal
  – establish trust between parties to exchange information and services (access control)

• Constraint
  – access control decision is made
    – NOT according to the party identity
    – BUT according to the credentials it has
Biblio


• Marianne Winslett: An Introduction to Trust Negotiation. iTrust 2003: 275-283