Authorization and Trust Management

Joshua D. Guttman
Worcester Polytechnic Institute
The MITRE Corporation

March 2013
Bertinoro International Spring School

Thanks to the US National Science Foundation, under grant 1116557

guttman@wpi.edu

Authorization

- A is authorized to do X means
  - A is permitted to do X
  - A has the right to do X

Authorization

- A is authorized to do X means
  - A is permitted to do X
  - A has the right to do X
- An authorization policy describes
  - which people (or which roles)
  - are permitted to do actions
  - in an organization or interaction

Authorization

- A is authorized to do X means
  - A is permitted to do X
  - A has the right to do X
- An authorization policy describes
  - which people (or which roles)
  - are permitted to do actions
  - in an organization or interaction
- Traditionally, a per-organization choice

Authorization

- A is authorized to do X means
  - A is permitted to do X
  - A has the right to do X
- An authorization policy describes
  - which people (or which roles)
  - are permitted to do actions
  - in an organization or interaction
- Traditionally, a per-organization choice
- Increasingly, requires cross-organization trust or cooperation

An Example Authorization Policy

in a hospital

- A doctor may
  - Read a patient's chart
  - Make care updates
  - Update diagnosis
  - Allow a patient to be discharged
- A nurse may
  - Read a patient's chart
  - Make care updates
- A bill collector may
  - Read a patient's chart only if determining payment terms
- A patient may
  - Read a chart if it is his own
  - Leave the hospital if a doctor has discharged him

Some Policies are Distributed

EPub gives discount if IEEE member and EPub student

EPub student if ABU accredited(U) and U student

IEEE member if . . .

ABU accredited(U) if . . .

U student if . . .

Some Policies are Distributed

EPub gives discount if IEEE member and EPub student

EPub student if ABU accredited(U) and U student

IEEE member if . . .

ABU accredited(U) if . . .

U student if . . .

The ellipses (. . .) require a query against another principal

Distributed policies now frequent

- Log in via Facebook or Google
- Electronic purchase requires authorization from Visa
- Hospital grants rights to remote doctors
  and especially insurance companies
- Tax service allows me to avoid audit
  if my bank and brokerage confirm my data

A logical approach

- Express authorizations by formulas
  $\mathrm{may}(\mathit{read}, \mathit{guttman}, \mathit{record}12)$
- Express policies by a set of rules
  $\mathrm{may}(\mathit{read}, A, R)$ if $\mathrm{doctor}(A)$ and $\mathrm{patient\_rec}(R)$
- Facts may be hard-coded in policy
  $\mathrm{doctor}(\mathit{guttman})$
  or else obtained by database queries
- Decision is authorized if policy + queries prove it

May be formalized in Datalog

- Datalog is a language for recursive queries
- Sublanguage of first order logic
  - Constants a_i, variables X_i, relation symbols
  - But no function symbols
  - Term t is either constant or variable
- Fact means atomic formula $\mathrm{p}(t_1, \ldots, t_i)$
- Rule means implication

$$\mathrm{p}(t_1, \ldots, t_i) \quad :- \quad \mathrm{q}_1(s_1, \ldots, s_j), \quad \ldots, \quad \mathrm{q}_\ell(r_1, \ldots, r_k)$$

- $\mathrm{p}(t_1, \ldots, t_i)$ called the head meaning consequent; the hypotheses are called the body
- Each t_n should appear in $s_1, \ldots, s_j, \ldots, r_1, \ldots, r_k$

Datalog semantics

- Domain of model:

  constants mentioned in theory T

- Extensions of predicates:

  Least set containing the facts and
  closed under the rules

- Model uniquely determined

  constructible in polynomial time

- Query may($read, guttman, X$)

  retrieves set of tuples matching pattern
  in extension of predicate

Datalog for Local Authorization Policies

- Most authorization policies fit nicely into Datalog
  possibly augmented with constraints,
  e.g. for linear inequalities

  Guttman may enter the Sala Affrescata
  between 9.00 and 19.00

- Lack of disjunctions, universal quantifier in body
  rarely problematic
- Existential quantifier in body implicit

$$\begin{aligned} \text{ancester}(x, y) &:- \text{parent}(x, y) \\ \text{ancester}(x, y) &:- \text{parent}(x, z), \\ &\quad\ \text{ancester}(z, y) \end{aligned}$$

Datalog for Distributed Authorization Policies

- Each principal P has a (Datalog) theory T_P
- P does deduction in T_P
  - Allows ϕ if $T_P \vdash \phi$
  - Communicates "P says ϕ" only when $T_P \vdash \phi$

Datalog for Distributed Authorization Policies

- Each principal P has a (Datalog) theory T_P
- P does deduction in T_P
  - Allows ϕ if $T_P \vdash \phi$
  - Communicates "P says ϕ" only when $T_P \vdash \phi$
- To incorporate "says" operator
  - For each predicate $\mathrm{q}(X_1, \ldots, X_i)$ add new predicate

    $$\mathrm{says\_q}(P, X_1, \ldots, X_i)$$

    with extra argument P
  - Means P says $\mathrm{q}(X_1, \ldots, X_i)$

Datalog for Distributed Authorization Policies

- Each principal P has a (Datalog) theory T_P
- P does deduction in T_P
  - Allows ϕ if $T_P \vdash \phi$
  - Communicates "P says ϕ" only when $T_P \vdash \phi$
- To incorporate "says" operator
  - For each predicate $\mathrm{q}(X_1, \ldots, X_i)$ add new predicate

    $$\mathrm{says\_q}(P, X_1, \ldots, X_i)$$

    with extra argument P
  - Means P says $\mathrm{q}(X_1, \ldots, X_i)$
- Now each T_P can use both
  old predicates $\mathrm{q}(X_1, \ldots, X_i)$ and
  new predicates $\mathrm{says\_q}(P', X_1, \ldots, X_i)$

What is trust?

- One core meaning of trust:

$$\mathrm{q}(X_1, \ldots, X_i) \quad :- \quad \mathrm{says\_q}(\mathit{alice}, X_1, \ldots, X_i) \tag{1}$$

- If T_{me} contains (11), then I trust Alice on the topic $q()$
  - If I learn that Alice says q holds,
    then I infer that it does hold

What is trust?

- One core meaning of trust:

$$\mathrm{q}(X_1, \ldots, X_i) \quad :- \quad \mathrm{says\_q}(alice, X_1, \ldots, X_i) \tag{1}$$

- If T_{me} contains (11), then I trust Alice on the topic $q()$
  - If I learn that Alice says q holds,
    then I infer that it does hold
- Flexible, e.g.

$$\mathrm{q}(bob, \ldots, X_i) :- \quad \mathrm{says\_q}(alice, bob, \ldots, X_i)$$

- I trust Alice on $q()$, at least about Bob

What is trust?

- One core meaning of trust:

$$\mathrm{q}(X_1, \ldots, X_i) \quad :- \quad \mathrm{says\_q}(alice, X_1, \ldots, X_i) \tag{1}$$

- If T_{me} contains (11), then I trust Alice on the topic $q()$
  - If I learn that Alice says q holds,
    then I infer that it does hold
- Flexible, e.g.

$$\mathrm{q}(bob, \ldots, X_i) :- \quad \mathrm{says\_q}(alice, bob, \ldots, X_i)$$

- I trust Alice on $q()$, at least about Bob
- Allows local ontologies

$$\mathrm{p}(X_1, \ldots, X_i) \quad :- \quad \mathrm{says\_q}(alice, X_1, \ldots, X_i)$$

"Says" and Digital Signature

- Digitally signed message was produced by signer p

$$[\![\text{tag } x, y]\!]_k$$

"Says" and Digital Signature

- Digitally signed message was produced by signer p

$$[\![\text{tag } x,\ y]\!]_k$$

- Translate this into a statement of p

"Says" and Digital Signature

- Digitally signed message was produced by signer p

$$[\![\text{tag } x, y]\!]_k$$

- Translate this into a statement of p
- Conventional rules for msg $\rightarrow$ statement

"Says" and Digital Signature

- Digitally signed message was produced by signer p

$$[\![\text{tag } x, y]\!]_k$$

- Translate this into a statement of p
- Conventional rules for msg $\rightarrow$ statement

$$p \text{ says } \text{pred}(x, y)$$

or

$$\text{says\_pred}(p, x, y)$$

"Says" and Digital Signature

- Digitally signed message was produced by signer p

$$[\![\text{tag } x, y]\!]_k$$

- Translate this into a statement of p
- Conventional rules for msg $\rightarrow$ statement

$$p \text{ says } \text{pred}(x, y)$$

or

$$\text{says\_pred}(p, x, y)$$

- Every signature received by R adds a formula to T_R

"Says" and Digital Signature

- Digitally signed message was produced by signer p

$$[\![\text{tag } x, y]\!]_k$$

- Translate this into a statement of p
- Conventional rules for msg $\rightarrow$ statement

$$p \text{ says } \text{pred}(x, y)$$

or

$$\text{says\_pred}(p, x, y)$$

- Every signature received by R adds a formula to T_R
- All this, assuming $k = \text{sk}(p)$ is p's signature key, and $\text{sk}(p) \in \text{non}$

Application: Certificate Authorities

- A certificate authority issues digitally signed msgs of form

$$[\![\text{cert } P, K, \text{etc}]\!]_{ck^{-1}}$$

- ca endeavors to ensure K is P's signature verification key

$$K = \text{vk}(P)$$

Application: Certificate Authorities

- A certificate authority issues digitally signed msgs of form

$$[\![\text{cert } P,\ K,\ \text{etc}]\!]_{ck^{-1}}$$

- ca endeavors to ensure K is P's signature verification key

$$K = \text{vk}(P)$$

- Principals R may translate $[\![\text{cert } p,\ k,\ \text{etc}]\!]_{ck^{-1}}$ as meaning

$$\text{says\_ver\_key}(ck, p, k)$$

Application: Certificate Authorities

- A certificate authority issues digitally signed msgs of form

$$[\![\text{cert } P,\ K,\ \text{etc}]\!]_{ck^{-1}}$$

- ca endeavors to ensure K is P's signature verification key

$$K = \text{vk}(P)$$

- Principals R may translate $[\![\text{cert } p,\ k,\ \text{etc}]\!]_{ck^{-1}}$ as meaning

$$\text{says\_ver\_key}(ck, p, k)$$

- R trusts ck as signing key for ca if T_R contains

$$\text{ver\_key}(P, K) \quad :- \quad \text{says\_ver\_key}(ck, P, K)$$

Certificate Chains

- ca, using key rk^{-1}, may also issue certificates

$$[\![\, \text{cert\_auth}\ \ P,\ CK,\ \text{etc}\,]\!]_{rk^{-1}}$$

asserting CK is valid to verify certificates from authority P

Certificate Chains

- ca, using key rk^{-1}, may also issue certificates

$$[\![\text{cert\_auth } P,\ CK,\ \text{etc}]\!]_{rk^{-1}}$$

  asserting CK is valid to verify certificates from authority P
- These certificates
  - Allow an organization to issue certificates for employees, e.g.
  - Cost much more, from commercial CAs

Certificate Chains

- ca, using key rk^{-1}, may also issue certificates

$$[\![\,\text{cert\_auth}\;\; P,\;\; CK,\;\; \text{etc}\,]\!]_{rk^{-1}}$$

  asserting CK is valid to verify certificates from authority P
- These certificates
  - Allow an organization to issue certificates for employees, e.g.
  - Cost much more, from commercial CAs
- Interpret $[\![\,\text{cert\_auth}\;\; P,\;\; CK,\;\; \text{etc}\,]\!]_{RK^{-1}}$ to mean

$$\text{says\_certifying\_key}(RK, P, CK)$$

Certificate Chains

- ca, using key rk^{-1}, may also issue certificates

$$[\![\,\text{cert\_auth}\;\; P,\;\; CK,\;\; \text{etc}\,]\!]_{rk^{-1}}$$

  asserting CK is valid to verify certificates from authority P
- These certificates
  - Allow an organization to issue certificates for employees, e.g.
  - Cost much more, from commercial CAs
- Interpret $[\![\,\text{cert\_auth}\;\; P,\;\; CK,\;\; \text{etc}\,]\!]_{RK^{-1}}$ to mean

$$\text{says\_certifying\_key}(RK, P, CK)$$

- Principal R accepts these if T_R contains

$$\begin{aligned} \text{ver\_key}(P, K) &:- \text{says\_ver\_key}(CK, P, K), \text{certifying\_key}(CK) \\ \text{certifying\_key}(CK) &:- \text{says\_certifying\_key}(RK, P, CK), \text{root\_auth}(\text{RK}) \\ \text{root\_auth}(\text{rk}). & \end{aligned}$$

Trust Management Example

EPub gives discount if IEEE member and EPub student

EPub student if ABU accredited(U) and U student

IEEE member if ...

ABU accredited(U) if ...

U student if ...

The theory T_{EPub} contains:

$$\begin{aligned} \text{discount}(\mathit{Buyer}) &:- \text{says\_member}(\mathit{ieee}, \mathit{Buyer}), \\ &\quad \text{student}(\mathit{Buyer}) \\ \text{student}(\mathit{Buyer}) &:- \text{says\_acc}(\mathit{abu}, \mathit{U}), \\ &\quad \text{says\_student}(\mathit{U}, \mathit{Buyer}) \end{aligned}$$

Problem or Limitations

- No treatment of recency
- No private communications
- No conditional disclosure
  - I'll tell you X
    if you convince me that Y
- Needed: integrate trust mgt with protocols