SDSIrep:

A reputation system based on SDSI

Ahmed Bouajjani, Javier Esparza,

Stefan Schwoon, Dejvuth Suwimonteerabuth

Université Paris 7,

Technische Universität München

Plan of the talk

Discretionary access control: from ACLs to SDSI.

A brief introduction to reputation systems

Message I : current reputation systems are as simplistic as ACLs.

Message II:

$$\frac{\mathsf{SDSIrep}}{\text{current reputation systems}} = \frac{\mathsf{SDSI}}{\mathsf{ACLs}} = \frac{\mathsf{prob.\ PDS}}{\mathsf{restr.\ FA}}$$

A brief introduction to discretionary access control

Access control systems

Systems with shared resources

Examples: file server, conference management system, . . .

Issue: access control to files and other objects (e.g. peripherals).

Ownership

Each object is owned by some user, who controls access to the object.

The authorization problem:

Given a user and an object, may the user access the object?

More generally, may the user perform a given operation (read/write/etc) on the object?

Access control lists (ACLs)

Popular approach: Attach to each object a list specifying which users or groups of users have which access rights (read, write, execute, . . .).

Example: ACL of file `assignments.txt`:

alice: read, write
students: read

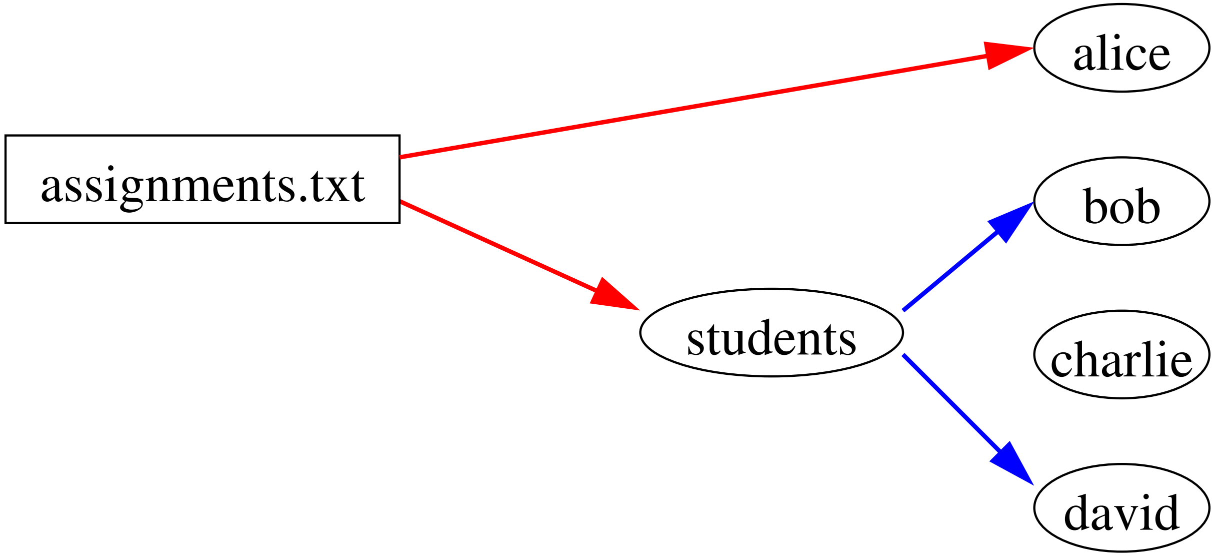
Some means of assigning users to groups, e.g.

students = bob, david, ...

The system contains access-control information and descriptive information (the world “as it is”).

Access control with ACL and groups

Read permissions as a graph:



Authorization problem: Is there a path (of length one or two) from object to user?

→ Efficient decision problem, but very rigid design

Extensions

Subset relations, e.g. staff $\longrightarrow$ profs (standing for staff $\supseteq$ profs).

Delegation: user “passes on” authority to others, e.g. professor Alice delegates the grading of homework to her staff: Alice $\longrightarrow$ staff.

More sophisticated solutions, e.g. KeyNote [Blaze, Feigenbaum et al (1999)]

Authorization problem: Given a finite graph representing the access-control and descriptive information, is there some path from object to user?
$\longrightarrow$ non-emptiness of a finite automaton

Problem: Who is entitled to define and update groups in a distributed environment?

Roles

Allow every participant p to (implicitly) define groups, e.g. r.

- Notation: $p.r$
- Meaning: the set of participants that have the role r from p's point of view.

Membership in the group $p.r$ is controlled by p.

Example: Bookstore in Munich offers discount to all students of TUM.
However: The bookstore doesn't know the students of TUM.

TUM is responsible for issuing certificates defining TUM.students:

$$\text{TUM.students} \longrightarrow \text{David}$$

The bookstore then publishes a certificate:

$$\text{Bookstore.discount} \longrightarrow \text{TUM.students}$$

Nested roles

The same bookstore offers a better discount to all PhD students of TUM:

$$\text{Bookstore.cheap} \longrightarrow \text{TUM.phd}$$

PhD students must have an advisor (a professor). This is described by a nested role:

$$\text{TUM.phd} \longrightarrow \text{TUM.profs.phd}$$

Certificates expressing that Alice is a professor and Bob her student:

$$\text{TUM.profs} \longrightarrow \text{Alice} \qquad \text{Alice.phd} \longrightarrow \text{Bob}$$

Nested roles (cont'd)

Roles allow inductive definitions of groups:

$$\begin{aligned}
\text{Alice.friends} &\longrightarrow \text{Charlie}\\
\text{Alice.friends} &\longrightarrow \text{TUM.profs}\\
\text{Alice.friends} &\longrightarrow \text{Alice.friends.friends}\\
\text{Alice.friends} &\longrightarrow \text{TUM.profs} \cap \text{ETAPS.authors}
\end{aligned}$$

Bob proves that he gets the discount by exhibiting a certificate chain that rewrites Bookstore.cheap into Bob:

$$\begin{aligned}
\text{Bookstore.cheap} &\longrightarrow \text{TUM.phd}\\
&\longrightarrow \text{TUM.profs.students}\\
&\longrightarrow \text{Alice.students} \qquad\qquad \text{(prefix-rewriting!)}\\
&\longrightarrow \text{Bob}
\end{aligned}$$

SDSI (SPKI/SDSI) [Clarke, Ellison,. . . since 1999]

Access control in SDSI

A SDSI system is equivalent to a pushdown system.

- Participants ≈ Control states
- Roles ≈ Stack alphabet
- Certificates ≈ Transition rules

The authorization problem reduces to the reachability problem for pushdown systems: given two control states p, q, is q reachable from p?

Theorem [E., Hansel, Rossmanith, S. 00; Jha, Reps 02]:
The authorization problem for n participants and m certificates can be solved in $O(n^2m)$ time and $O(nm)$ space.

A brief introduction to reputation systems

Reputation systems

Open-world systems

Participants do not know each other and change dynamically.

Example: Internet-based systems (auctions, peer-to-peer, etc.)

Issue: trust and reputation (trust $\widehat{=}$ local, reputuation $\widehat{=}$ global)

The reputation problem:

How much trust does the community of participants have in a given participant?

Current reputation systems

Participants recommend each other with a given weight.

The reputation of the participant is extracted from the weighted graph of recommendations.

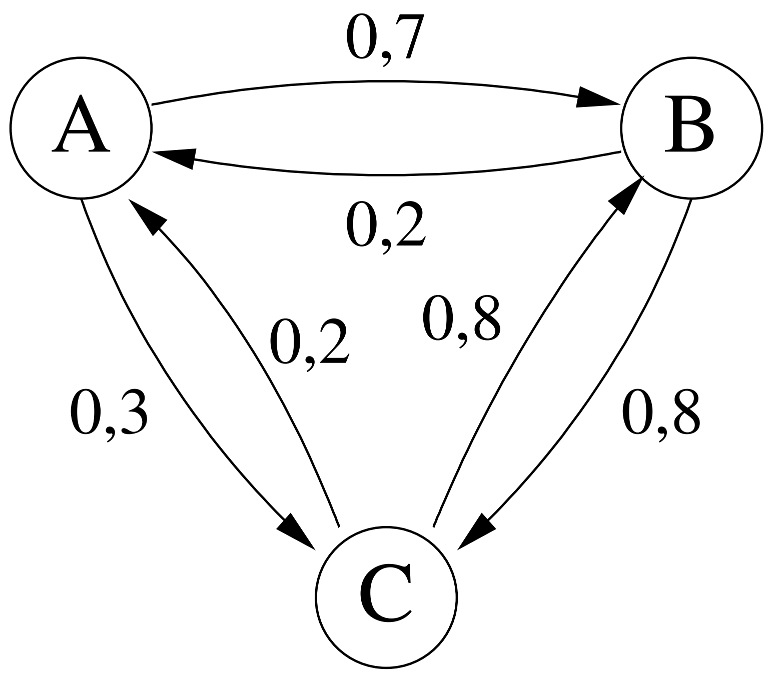
eBay: Reputation = average weight of incoming arcs.

PageRank, Eigentrust [Kamvar, Schlosser, Garcia-Molina 03]:

- Reputation computed using a probabilistic interpretation.
- Construct a Markov chain with
  - the participant as nodes;
  - an edge from A to B labelled by $x \in [0, 1]$ if A recommends B with (relative) weight x.

Reputation of a participant: its value in the stationary distribution.

An example



Reputation: A= 0.16 B= 0.44 C= 0.40.

Reputation values are relative!

Criticism and new idea

Our thesis: Current reputation systems are as rigid as ACL-lists

- No possibility to define groups or recommend groups.
- Peer-to-peer trust expressed directly

Our idea: design a reputation system based on SDSI.

- Trust assigned to individuals or to groups
- Peer-to-peer trust given through certificates

SDSIrep: a reputation system based on SDSI

Weighted certificates

Equip certificates with numerical weights.

Certificate A.r $\xrightarrow{x}$ B.s : the members of B.s belong to A.r with degree x.

Example: ICALP.authors $\xrightarrow{x}$ Esparza

x = fraction of the ICALP papers having Esparza as (co-)author

Certificate A $\xrightarrow{x}$ B.s : A recommends the members of B.s with weight x.

Example: Bouajjani $\xrightarrow{y}$ ICALP.authors

y = Bouajjani's estimation of ICALP's relative quality

Recommendations are actually relative recomendations.

Probabilistic interpretation

Assume Bouajjani issues another certificate Bouajjani $\xrightarrow{z}$ Esparza.

With which total weight does Bouajjani recommend Esparza?

> Normalize the weights of certificates with the same left-hand side so that they add up to 1.

Bouajjani recommends Esparza because of

Bouajjani $\xrightarrow{y}$ ICALP.authors and ICALP.authors $\xrightarrow{x}$ Esparza
Bouajjani $\xrightarrow{z}$ Esparza

"Summarize" this as: Bouajjani $\xrightarrow{y \cdot x + z}$ Esparza.

Semantics

A SDSIrep system is equivalent to a probabilistic pushdown system.

Participants $\approx$ Control states

Roles $\approx$ Stack alphabet

Weighted certificates $\approx$ Probabilistic transition rules

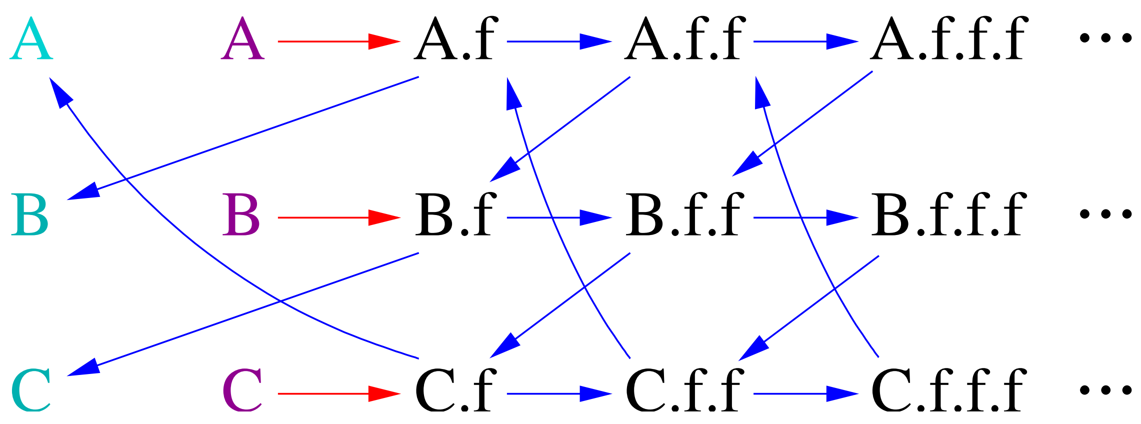
Problem: the Markov chain associated to a SDSIrep system can be infinite.

An example

$\text{Alice.frs} \xrightarrow{0.7} \text{Bob}$ $\qquad$ $\text{Alice.frs} \xrightarrow{0.3} \text{Alice.frs.frs}$ $\qquad$ $\text{Alice} \xrightarrow{1} \text{Alice.frs}$

$\text{Bob.frs} \xrightarrow{0.9} \text{Charlie}$ $\qquad$ $\text{Bob.frs} \xrightarrow{0.1} \text{Bob.frs.frs}$ $\qquad$ $\text{Bob} \xrightarrow{1} \text{Bob.frs}$

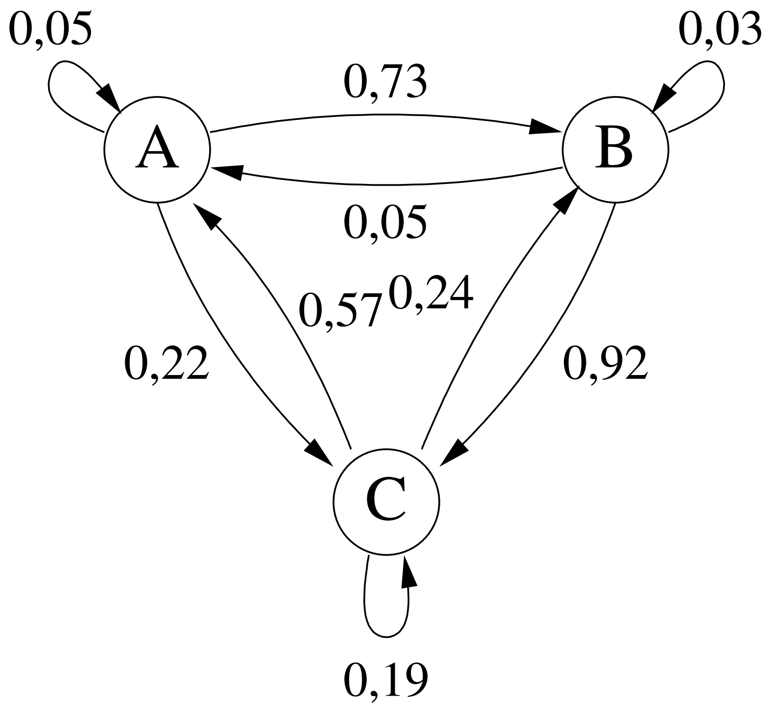
$\text{Charlie.frs} \xrightarrow{0.5} \text{Alice}$ $\qquad$ $\text{Charlie.frs} \xrightarrow{0.5} \text{Charlie.frs.frs}$ $\qquad$ $\text{Charlie} \xrightarrow{1} \text{Charlie.frs}$



Alice's trust in Bob: probability of, starting at A, ending at B.

Theorem: The trust between participants is the least solution of a system of $n^2 \cdot m$ quadratic equations.

$$
\begin{aligned}
[A,A] &= 0.3 \sum_{I\in\{A,B,C\}} [A,I]\cdot[I,A] & [A,A] = 0.05 \\
[A,B] &= 0.3 \sum_{I\in\{A,B,C\}} [A,I]\cdot[I,B] + 0.7 & [A,B] = 0.73 \\
[A,C] &= 0.3 \sum_{I\in\{A,B,C\}} [A,I]\cdot[I,C] & [A,C] = 0.22 \\
[B,A] &= 0.1 \sum_{I\in\{A,B,C\}} [B,I]\cdot[I,A] & [B,A] = 0.05 \\
[B,B] &= 0.1 \sum_{I\in\{A,B,C\}} [B,I]\cdot[I,B] & [B,B] = 0.03 \\
[B,C] &= 0.1 \sum_{I\in\{A,B,C\}} [B,I]\cdot[I,C] + 0.9 & [B,C] = 0.92 \\
[C,A] &= 0.5 \sum_{I\in\{A,B,C\}} [C,I]\cdot[I,A] + 0.5 & [C,A] = 0.57 \\
[C,B] &= 0.5 \sum_{I\in\{A,B,C\}} [C,I]\cdot[I,B] & [C,B] = 0.24 \\
[C,C] &= 0.5 \sum_{I\in\{A,B,C\}} [C,I]\cdot[I,C] & [C,C] = 0.19
\end{aligned}
$$



(Relative) Reputation is the stationary distribution of this Markov chain.

Two stages:

- Markov chain (peer-to-peer values) obtained from quadratic equation system.
- Reputation obtained from Markov chain using linear equation system.

Evaluating the reputation of the PC of TACAS 2008

Participants: the 28 members of TACAS'08 PC, 6 conferences (CAV, ICALP, LICS, POPL, VMCAI, TACAS), the Citeseer author list, the Citeseer impact list, and the list of h-numbers taken from "publish or perish".

Roles: auth, publ, coaut, and circ, with the following intended (fuzzy) meaning

- c.auth: researchers that publish in conference c;
- r.publ: conferences in which researcher r has published;
- r.coaut: r's co-authors;
- r.circ: r's circle, defined as r's coauthors, plus the coauthors of r's coauthors, and so on (degree of membership decreases with "distance" to r)

Certificates

$$\text{TACAS.auth} \xrightarrow{10} \text{KL} \qquad \text{Impact} \xrightarrow{1.24} \text{TACAS.auth}$$

$$\text{KL.publ} \xrightarrow{10} \text{TACAS} \qquad \text{H-number} \xrightarrow{34} \text{KL}$$

$$\text{KL.coaut} \xrightarrow{22} \text{PP} \qquad \text{Citeseer} \xrightarrow{2023} \text{KL}$$

$$\text{KL.circ} \xrightarrow{0.8} \text{KL.coaut} \qquad \text{KL} \xrightarrow{4} \text{KL.publish.auth}$$

$$\text{KL.circ} \xrightarrow{0.2} \text{KL.circ.circ} \qquad \text{KL} \xrightarrow{3} \text{KL.circ}$$

"Delegation:" $\text{KL} \xrightarrow{2} \text{Impact}$

$\text{KL} \xrightarrow{3} \text{Citeseer}$

$\text{KL} \xrightarrow{3} \text{H-index}$

Some experimental results

PB	EB	TB	RC	BC	BD	PG	OG	AG	FH	MH	JJ	KJ	JK
26	18	19	78	45	6	56	60	30	19	45	19	5	23

BK	MK	KL	NL	KN	PP	SR	CR	JR	AR	SS	SS	BS	LZ
10	30	88	26	37	33	64	22	45	6	54	15	80	41

More experimental results

scientists	10	20	30	40	50	60	70	76
variables	627	1653	3089	4907	7126	9752	12777	14779
time (s)	0.47	2.07	6.85	12.55	23.90	44.89	78.35	106.55

	Unflattened	Depth 2	Depth 3	Depth 4	Depth 5	Depth 6
vars	2545	5320	7059	8798	10537	12276
time	5.83	1.23	3.32	6.39	10.34	18.78

Conclusions

SDSIrep increases the flexibility of reputation systems like EigenTrust.

Reputation computable with reasonable resources.

Expectation: between linear and quadratic slow-down compared to EigenTrust.

Numerical solution related to interesting theoretical problems (Esparza/Kiefer/Luttenberger: STOC'07, STACS'08).