The unreasonable ineffectiveness of security engineering

Dusko Pavlovic
Kestrel Institute and Oxford University

SEFM/FAST
Pisa, September 2010

Ages of software

Ancient times
- platform: Computer
- applications: Quicksort, compilers, Julia set, ...
- requirements:
  - correctness
  - termination
- tools: programming languages

Middle ages
- platform: Operating System
- applications: MS Word, Oracle, Reason, ...
- requirements:
  - good things happen
  - bad things don't happen
- tools: programming and specification languages
  - refinement, composition
  - verification

Modern times
- platform: Network
- applications: Web, Google, botnets, ...
- requirements:
  - good information flows happen
  - bad information flows don't happen
- tools: Ajax, mashups...
  - ???
  - ???

Question
- Are some formal tools useful for modern software?
### Question

- Are some formal tools useful for modern software?
  - Are they in security engineering?

### Outline

#### Problem: All protocols are insecure
- Life cycle of security
- Adverse selection of trust

#### Background: Security is a process

#### Method: Learning security

#### Discussion: Formal methods for networks

---

#### The Unreasonable Effectiveness of Mathematics in Natural Sciences

E. Wigner (1960)

- Why is nature made in the measure of our mind?

---

#### The Unreasonable Ineffectiveness of Engineering in Security

- Why are we not becoming more secure from more security technologies?

---

#### Verified protocols fail
Ineffectiveness of security
D. Pavlovic

Problem
Life cycle of security
Adverse selection
Background
Method
Discussion

Verified protocols fail

Bull’s protocol
  - Isabelle: secure for $E(k, m; n)$
  - Ryan & Schneider: not for $E(k, m; n) = n \oplus H_k(m)$

IPSec GDoI
  - IETF MSec WG: secure (7 drafts), verified (3 times)
  - Cathy & Dusko: GDoI PoP attack

MQV
  - NSA: "MQV is critical for national security of US"
  - Krawczyk: MQV insecure

Security is an adversarial process
Protocols
Attack

Protocol

MQV
  - NSA: "MQV is critical for national security of US"
  - Krawczyk: MQV insecure, HMQV proven secure

Menezes: HMQV insecure
Security is an adversarial process

Adverse selection of trust

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Attack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory</td>
<td></td>
</tr>
<tr>
<td>Counter-model</td>
<td></td>
</tr>
</tbody>
</table>

Adverse selection of trust

<table>
<thead>
<tr>
<th>TRUSTE-certified</th>
<th>uncertified</th>
</tr>
</thead>
<tbody>
<tr>
<td>honest</td>
<td>94.6%</td>
</tr>
<tr>
<td>malicious</td>
<td>5.4%</td>
</tr>
<tr>
<td>97.5%</td>
<td>2.5%</td>
</tr>
</tbody>
</table>

Table: Trustworthiness of TRUSTE [Edelman 2007]

Adverse selection of trust

<table>
<thead>
<tr>
<th>Google</th>
<th>sponsored</th>
<th>organic</th>
</tr>
</thead>
<tbody>
<tr>
<td>top</td>
<td>4.44%</td>
<td>2.73%</td>
</tr>
<tr>
<td>top 3</td>
<td>5.33%</td>
<td>2.93%</td>
</tr>
<tr>
<td>top 10</td>
<td>5.89%</td>
<td>2.74%</td>
</tr>
<tr>
<td>top 50</td>
<td>5.93%</td>
<td>3.04%</td>
</tr>
</tbody>
</table>

Table: Malicious search engine placements [Edelman 2007]

Adverse selection of trust

<table>
<thead>
<tr>
<th>Yahoo!</th>
<th>sponsored</th>
<th>organic</th>
</tr>
</thead>
<tbody>
<tr>
<td>top</td>
<td>6.35%</td>
<td>0.00%</td>
</tr>
<tr>
<td>top 3</td>
<td>5.72%</td>
<td>0.35%</td>
</tr>
<tr>
<td>top 10</td>
<td>5.14%</td>
<td>1.47%</td>
</tr>
<tr>
<td>top 50</td>
<td>5.40%</td>
<td>1.55%</td>
</tr>
</tbody>
</table>

Table: Malicious search engine placements [Edelman 2007]

Adverse selection of trust

<table>
<thead>
<tr>
<th>Ask</th>
<th>sponsored</th>
<th>organic</th>
</tr>
</thead>
<tbody>
<tr>
<td>top</td>
<td>7.99%</td>
<td>3.23%</td>
</tr>
<tr>
<td>top 3</td>
<td>7.99%</td>
<td>3.24%</td>
</tr>
<tr>
<td>top 10</td>
<td>8.31%</td>
<td>2.94%</td>
</tr>
<tr>
<td>top 50</td>
<td>8.20%</td>
<td>3.12%</td>
</tr>
</tbody>
</table>

Table: Malicious search engine placements [Edelman 2007]

Adverse selection of trust

“Pillars of the society” phenomenon

- social hubs are more often corrupt
- the rich are more often thieves
- higher trust is more often betrayed
Ineffectiveness of security

D. Pavlovic

Problem
-Life cycle of security
-Adverse selection

Background

Method

Discussion

Outline

Problem: All protocols are insecure

Background: Security is a process
-Static and dynamic security
-Trust process
-Fragility of trust

Method: Learning security

Discussion: Formal methods for networks

Security architecture

-secure perimeter (~ cryptography)
-secure access (~ protocols)

Security process

-Attacks on the perimeter: cryptanalysis
-Attacks on the access: breaking protocols
Security threats

- static:
  - attacks on the perimeter: cryptanalysis
  - attacks on the access: breaking protocols

- dynamic:
  - moles: dishonest but trusted
  - defectors: first honest, later dishonest but trusted

Security tools

- static:
  - walls, borders, cryptosystems...
  - locks, passports, protocols, policies...

- dynamic:
  - weapons, police, penal system...
  - trust, privacy, reputation, influence...

What is trust?

Alice trusts that Bob will act according to a protocol $\Phi$.

Examples

- shopping: Bob will deliver goods
- marketing: Bob will pay for goods
- access control: Bob will not abuse resources
- key infrastructure: Bob's keys are not compromised
- software system: Bob will perform his function

Modeling trust

Trust is a labelled relation $u \rightarrow^r j$

- $u$: truster
- $j$: trustee
- $r$: trust rating

(leaving the entrusted protocol $\Phi$ implicit)
Modeling trust

Trust network

\[ M = (R \leftarrow C \rightarrow \mathbb{S} \times \mathbb{O}) \]

where

- \( R = \{r, s, \ldots\} \): ordered set of ratings
- \( C = \{\gamma_1, \gamma_2, \ldots\} \): certificates
- \( S = \{u, v, \ldots\} \): trustors (subjects, users)
- \( O = \{i, j, \ldots\} \): trustees (objects, items)

Sources of trust ratings

- trustor's own experience
  - social networks
- collaborative rating, filtering and recommendation
  - Amazon, Ebay, …
- surveillance and indexing of the network
  - credit card operators, credit rating and direct marketing agencies, Google, Amazon, DHS, NSA, …

Use of trust ratings

\[ X_i(t+1) = i \]

\[ \text{Prob}(X_i(t+1) = i) = C(t)M_{iu}(t) \]

(\text{where } C(t) = \frac{1}{\sum_{i \in O} M_{iu}(t)})

Building and updating trust

\[ M_{iu}(t+1) = \begin{cases} 
M_{iu}(t) & \text{if } i \neq X_i(t+1) \\
0 & \text{if } i = X_i, \text{ not satisfactory} \\
1 & \text{if } i = X_i, \text{ satisfactory, new} \\
1 + M_{iu}(t) & \text{if } i = X_i, \text{ satisfactory, not new} 
\end{cases} \]
Trust distribution

Theorem (FAST 2008)

In the long run, and with a large pool of trustees, the trust processes lead to a power law distribution of ratings.

\[
\{ i \in O \mid M_u = n \} \approx c \cdot n^{-\frac{(1+\gamma)}{\gamma}}
\]

for some constants \( c \) and \( \gamma \).

Trust is like money

▶ The rich get richer

▶ Heavy tails make theft more profitable

▶ Abstract loot is easily transferrable
Fragility of trust

Corollary
The hubs attract attacks as soon as trust is
(a) public
(b) uniform
(c) abstract

Defending trust

Policy
Possible defense strategies are:
(a) non-public: private trust vectors
   recommendations must be public
(b) non-uniform: higher security for higher trust
   complicated; contradicts (a).
(c) non-abstract: retain trust concepts
   qualifying trust: $u \rightarrow j$
   • record the actual feedback (~“marked money”)
   • credit rating, surveillance, profiling

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Defending trust

Policy
Possible defense strategies are:
(a) non-public: private trust vectors
   recommendations must be public
(b) non-uniform: higher security for higher trust
   complicated; contradicts (a).
(c) non-abstract: retain trust concepts
   qualifying trust: $u \rightarrow j$
   • record the actual feedback (~“marked money”)
   • credit rating, surveillance, profiling

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Defending trust

Policy
Possible defense strategies are:
(a) non-public: private trust vectors
   recommendations must be public
(b) non-uniform: higher security for higher trust
   complicated; contradicts (a).
(c) non-abstract: retain trust concepts
   qualifying trust: $u \rightarrow j$
   • record the actual feedback (~“marked money”)
   • credit rating, surveillance, profiling

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion

Ineffectiveness of security
D. Pavlovic

Problem
Background
Statics and dynamics
Trust process
Fragility of trust

Method
Discussion
Example: Small town in World Wild West

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>1.25</td>
<td>1.05</td>
<td>1.12</td>
<td>1.57</td>
</tr>
<tr>
<td>II</td>
<td>.83</td>
<td>1.13</td>
<td>1.02</td>
<td>.35</td>
</tr>
<tr>
<td>III</td>
<td>0</td>
<td>.35</td>
<td>.21</td>
<td>-.56</td>
</tr>
</tbody>
</table>

Outline

Problem: All protocols are insecure
Background: Security is a process
Method: Learning security
Linear algebra of trust
Spectral decomposition of trust
Logic of trust
Discussion: Formal methods for networks

Sequential and parallel composition of trust

Linear algebra

\[ M_u = (2 \times 4) + (3 \times 5) \]

(Alternative: Tropical algebra)

\[ M_u = (2 \times 4) \lor (3 \times 5) \]

Sequential and parallel composition of trust

(Alternative: Resistance algebra)

\[ M_u = \frac{1}{u} + \frac{1}{v} + \frac{1}{w} + \frac{1}{x} \]

Trust correlations

Definition

The subject correlation for \( u, v \in S \) is

\[ S_{uv} = \sum_{i \in O} M_{ui} M_{iv} \]
Ineffectiveness of security
D. Pavlovic

Problem
Background
Method

Algebra of trust
SVD of trust
Logic of trust

Discussion

Trust correlations

Definition
The object correlation for \( i, j \in O \) is
\[
O_{ij} = \sum_{u \in S} M_{iu}M_{ju}
\]

Trust preferences and communities

Definition
- A subject preference is a unit vector \( \nu \in \mathbb{R}^S \).
- A subject community is a subspace \( T \subseteq \mathbb{R}^S \).

Remark
Community is a convex set of preferences.

Definition
- A subject preference is a unit vector \( \nu \in \mathbb{R}^O \).
- A subject community is a subspace \( \Psi \subseteq \mathbb{R}^O \).

Remark
Community is a convex set of preferences.

Trust preferences and communities

Definition
- An object preference is a unit vector \( \psi \in \mathbb{R}^O \).
- An object community is a subspace \( \Psi \subseteq \mathbb{R}^O \).

Remark
Community is a convex set of preferences.

Trust preference correlations

Definition
A subject preference correlation for \( \nu, \omega \in \mathbb{R}^S \) is the cosine of \( \mathcal{L}(\nu, \omega) = \mathcal{L}(S \nu, \omega) \).

Trust correlations

Definition
The correlation matrices are thus
\[
S = M \cdot M^\top
\]
\[
O = M^\top \cdot M
\]
where \( M^\top \) denotes the transpose.
Trust preference correlations

Definition
A subject preference correlation for \( \nu, \omega \in \mathbb{R}^3 \) is the cosine of \( \angle(\nu, \omega) = \angle(S\nu, \omega) \).

Fact
The members of a subject community \( \Upsilon \) have the strongest correlations if \( \Upsilon \) is an eigenspace of \( S \), i.e., if
\[
Sy = y
\]

Definition
An object preference correlation for \( \psi, \chi \in \mathbb{R}^3 \) is the cosine of \( \angle(\psi, \chi) = \angle(O\psi, \chi) \).

Fact
The members of an object community \( \Psi \) have the strongest correlations if \( \Psi \) is an eigenspace of \( O \), i.e., if
\[
O\psi = \psi
\]

Trust concepts

Lemma
Let
\[
\begin{align*}
\hat{S} &= \text{the set of eigenspaces of } S : \mathbb{R}^3 \to \mathbb{R}^3 \\
\hat{O} &= \text{the set of eigenspaces of } O : \mathbb{R}^3 \to \mathbb{R}^3
\end{align*}
\]
Then the linear operator \( M : \mathbb{R}^3 \to \mathbb{R}^3 \), induced by a matrix \( M = (\delta_{i,j})_{i\in S, j\in S} \) induces a bijection
\[
\hat{M} : \hat{S} \to \hat{O}
\]
where \( \hat{M}(\Upsilon) = \Psi \) iff \( M\Upsilon = \Psi \).

Spectral decomposition of trust

Proposition (SVD)
Any linear operator \( M \) decomposes to
\[
\mathbb{R}^3 \xrightarrow{M} \mathbb{R}^3 \\
\mathbb{R}^3 \xrightarrow{\hat{M}} \mathbb{R}^3
\]
\[
M = \sum_{i=1}^d \delta_i |\psi_i\rangle\langle T_i|,
\]
where \( T \) and \( \psi \) are isometries.

Proposition (SVD)
This decomposition is unique up a permutation on \( \hat{S} \), or equivalently up to a permutation on \( \hat{O} \), lifted via \( \hat{M} \).
Spectral decomposition of trust

**Proposition (SVD)**

This decomposition is unique up a permutation on $\hat{S}$, or equivalently up to a permutation on $\hat{O}$, lifted via $M$.

Equivalently, the decomposition is uniquely determined by

- sets $\hat{S}$ and $\hat{O}$
- bijection $M : \hat{S} \rightarrow \hat{O}$

For example, the decomposition is uniquely determined by

\[
\begin{bmatrix}
1.25 & 1.05 & 1.12 & 1.57 \\
.83 & 1.13 & 1.02 & .35 \\
.00 & .35 & .21 & -.56
\end{bmatrix}
\]

Equivalently, the decomposition is uniquely determined by

\[
\Phi = (T, \Psi)
\]

where

\[
\begin{bmatrix}
.12 \\
.05 \\
.05 \\
.13
\end{bmatrix}
\]

**Application: Small town in World Wild West**

\[
\begin{bmatrix}
1.25 & 1.05 & 1.12 & 1.57 \\
.83 & 1.13 & 1.02 & .35 \\
.00 & .35 & .21 & -.56
\end{bmatrix}
\]

\[
M = \begin{bmatrix}
1.25 & 1.05 & 1.12 & 1.57 \\
.83 & 1.13 & 1.02 & .35 \\
.00 & .35 & .21 & -.56
\end{bmatrix}
\]

**Trust concepts**

\[
\begin{bmatrix}
.12 \\
.05 \\
.05 \\
.13
\end{bmatrix}
\]
Ineffectiveness of security

D. Pavlovic

Problem

Background

Method

Algebra of trust

SVD of trust

Logic of trust

Discussion

Trust predicates

S E O

Singular nodes:

- $2\phi_2 \leq -\phi_1 \leq 0$

- $\Phi_2 \geq 2\phi_1 \geq 0$

Result

The trust concept guidance

- limits the market for "trust lemons" (cf Akerlof)
- minimizes the incentive for adverse selection

Comments

- Trust concepts are intrinsic to the network. The nodes may not be aware of them.
- The traitors may be recognized as singularities, even if their profiles were never seen before.

Qualifying trust

Each trust relationship has an intrinsic trust concept

$$u = \sum_{i} \phi_i \psi_i$$

Minimize

$$r_j = \lambda_i \psi_i \nu_i$$
Ineffectiveness of security
D. Pavlovic
Problem
Background
Method
Discussion
Social software engineering
Formal methods for networks
Problem 2.0

Outline

Problem: All protocols are insecure

Background: Security is a process

Method: Learning security

Discussion: Formal methods for networks

Social software engineering
Formal methods for networks
Problem 2.0

The ages of software

<table>
<thead>
<tr>
<th>era</th>
<th>ancient times</th>
<th>middle ages</th>
<th>modern times</th>
</tr>
</thead>
<tbody>
<tr>
<td>platform</td>
<td>ancient times</td>
<td>middle ages</td>
<td>modern times</td>
</tr>
<tr>
<td>applications</td>
<td>ancient times</td>
<td>middle ages</td>
<td>modern times</td>
</tr>
<tr>
<td>requirements</td>
<td>ancient times</td>
<td>middle ages</td>
<td>modern times</td>
</tr>
<tr>
<td>tools</td>
<td>ancient times</td>
<td>middle ages</td>
<td>modern times</td>
</tr>
</tbody>
</table>

What is left to formal methods?

"It is the aim of the natural scientist to discover mathematical theories, formally expressed as predicates describing the relevant observations that can be made of some physical system. […]

The aim of an engineer is complementary to that of the scientist. He starts with a specification, formally expressible as a predicate describing the desired observable behaviour of a system or product not yet in existence. Then […] he must design and construct a product that meets that specification."

C.A.R. Hoare,
Programs are predicates

Formal methods could stand on both legs

Science: analyze, learn

Software: refine, synthesize

Specification
Ineffectiveness of security
D. Pavlovic
Problem
Background
Method
Discussion
Social engineering
FM for networks
Problem 2.0

**Problem 2.0**

- secure social software:
  - detect malicious insiders
- do not manipulate social processes:
  - respect the privacy of others

**Problem 2.0: Statistical database filtering**

Dalenius’ Desideratum (1977)

"Anything that can be learned about a respondent from a statistical database should be learnable without the access to the database."

**Problem 2.0: Statistical database filtering**

Dalenius’ Desideratum (1977)

"Anything that can be learned about a respondent from a statistical database should be learnable without the access to the database."

**Impossible**

For every statistical database $A$ with an incomplete record $R_A \subseteq R$, there is a database $B$ with an incomplete record $R_B \subseteq R$ such that the complete record $R$ can be derived from $R_A$ and $R_B$. 

FM for networks = logic + learning
logic: programs are (models of) predicates
learning: processes are (driven by) concepts

FM for networks = logic + learning
logic: programs are (models of) predicates
  - predicates $\Phi_{\geq}$
learning: processes are (driven by) concepts
  - concepts $\Phi \rightarrow Q$
Ineffectiveness of security
D. Pavlovic

Problem
Background
Method
Discussion
Social engineering
FM for networks
Problem 2.0

Upshot

▷ Trust and privacy are two sides of the same coin
  ▷ the same tools secure trust and break privacy

▷ Data privacy is like DRM
  ▷ inference control against the users

▷ Dynamic threats are mitigated by network analysis
  ▷ not data mining, but concept mining

▷ The tradeoff between trust and privacy is not a technical problem
  ▷ resolved through battling out the political powers of the stakeholders