Useless Metaphors? Why Specifying Security is So Hard

DIMACS Workshop on
Useable Privacy and Security Software
Patrick McDaniel - AT&T Research
July 8th, 2004

A story …

What is security policy?

- Statement of expected or desirable behavior within some defined scope
- A *policy system* is a collection of abstractions, representations, interfaces, and implementations used to *specify* and *enforce* policy
  - Realization of underlying model (metaphors)
  - RBAC, B-LP, P3P, Keynote, Antigone, IE Privacy
- Problem: Why don't we have effective interfaces for security policy?

Goals

- A policy system is effective if
  - Allows users to state (interface)
  - what they want (intent)
  - in terms they understand (vocabulary) …
  - … and the system meets that specification. (enforcement)
- Examples:
  - IE Cookie Management Policy : no TP cookies
  - Systrace Policy: `ls` process cannot open network connections

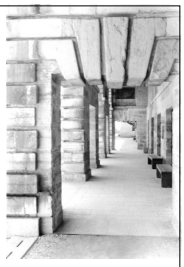
Clearly, we are not there …

- Policy is to CISCO as security is to Microsoft

```
interface Tunnel0-1a67sd
  description Tunnel to router at 1b67sd
  ip address 192.68.23.22 31
  tunnel source sdf01orat22
  tunnel destination sd02forat23
  exit
crypto isakmp policy 10
  authentication pre-share
  encryption 3des
  group 3
  hash sha
```

- Moreover, Security is to Microsoft because of default (open functionality) policy, and no clear way to see or change default policy

One Perspective



- Hypothesis: *Security Policy Systems largely fail because designers fail to present a clear narrative* to the user*
- Experiment: Look at guidelines for fiction and non-fiction writing
  - S&W, my 6th grade primer, ARMY handbook, Harlequin Romance, BBC, web style guides …

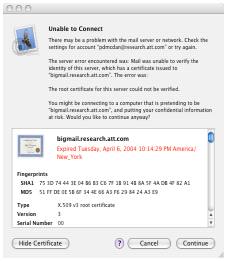
Axioms/Guidelines

- What do these stylebooks and guidelines tell us about effective communication?
  - Themes emerge about good (and bad) writing style (axioms)
  - Do they apply to design of policy systems?
- Policy uses metaphors/abstractions to communicate
  - This is not only interface, but modeling …
- So, lets see what axioms (from the guidelines) apply to policy design ….

Axiom 1: Know audience (vocabulary)

"She grew on him like he was e coli and she was room temperature Canadian beef."

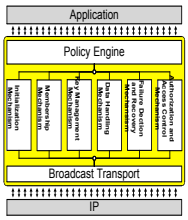
- Policy that fails to speak the users' language has no chance of success
- Moreover, any policy that requires decisions about topics outside users scope of experience has little chance of success



Axiom 2: Focus …

"The knife was as sharp as the tone used by Rep. Shelia Jackson Lee (D-Tex) in the first several points of the parliamentary procedure made to Rep. Henry Hide (R-Ill.) in the House Judiciary Committee hearings on the impeachment of Present William Jefferson Clinton."

- Seperation of concerns
  - Policy should focus on the topics of user interest
  - Be only as flexible as necessary (e.g., Ismene)
- However, needs to be complete (enough)

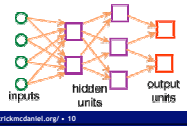


Axiom 3: Simplicity

(intent)

"The plan was simple like my brother-in-law Phil. But unlike Phil, this plan just might work."

- Complexity is the enemy
  - Abstractions work to clarify meaning
  - and simplify tasks or policy structures, i.e, roles
- … but so is simplicity
  - Oversimplification also problematic
  - e.g., high/med/low privacy

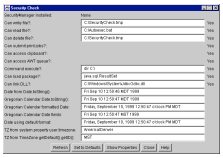


Axiom 4: Structure/tone

(interface)

"Her vocabulary was as bad as, like, whatever."

- A confounding interface, no matter how clear the underlying model, is fatal …
- Interface should be all those things we hope to see from HCI community
  - Intuitive
  - Easy to navigate
  - Targeted to *task*
  - (focused, simple, …)

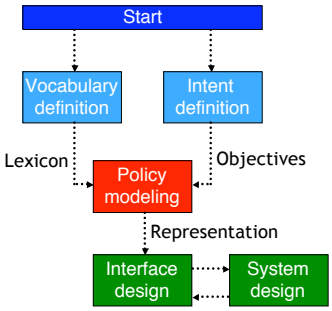


What does this all mean?

- Idea: we want to apply these axioms to drive design of apply?

Narrative Driven Policy Design

A (new) policy design workflow …

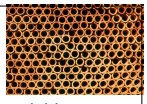


a) Apply axioms to policy design

b) Interact with user community to determine requirements

c) Separation of mechanism from meaning

Conclusions



- Security policy design *is* hard
  - Lots of ways to make mistakes, some unavoidable
  - Policy rarely a factor in systems/interface design
- Community needs to spend more time looking at intent, and less about form and enforcement
  - Most of the problem is no longer about technology, it is about providing meaningful interfaces
  - Separation of the *how* from the *what*
- Idea: narrative driven policy design
  - Not new: storyboarding, etc. is common in HCI
  - Apply to distributed systems security Policy
  - Use tenets of HCI to analysis and modeling

Thank you …

Patrick McDaniel
pdmcdan@research.att.com

"Every minute without you feels like 60 seconds."