Session Summary
Approaches to Usable Security

Lorrie Cranor
Don Norman
Approaches to Usable Security Session Notes

A. Characteristics of U-Diff in U:
   e.g., there are predators
   How to enable Us to make meaningful decisions,
   How do we provide timely meaningful feedback
   to users related to informed security decisions?
   How do we provide systems to prevent users from making unsafe decisions (in back or to get their work done)?
   If you back out, how much can you recover if how do you know?

B. How can we develop secure defaults that enable functionality users want?
   How can we give users choices without overwhelming them?
   How do we inform users about risks in a meaningful way?
   How do we inform users so they can make the decisions that matter to them?
   Can we develop guidelines/toolkits/APIs for usable security—based on today’s best practices?

B3. Can we build adaptive interfaces that respond appropriately to various security settings?
   How far down the protocol stack do we have to redesign to get usable security?
   Can we leverage existing security standards?
   How can we improve existing security standards to account for usability?
   What would it be like to work through privacy/security intermediaries rather than do it yourself or automation?
Approaches to Usable Security Session Notes

How could we manage security for users who don't want to do it themselves (and other user types)?
- What level of feedback do users want/need?
Can we develop consistent security indicators for the web, applications?
- usable, understandable, 101-friendly

Why is this a difficult problem to solve?
What are the right objective metrics for usable privacy/security?

What does a user need to know to perceive a system as safe when it is safe and how should that be communicated?

How do we break the silos between sec/priv/usability?
- design/UI/test/dev/pm?

How do we change the role of security people so they are perceived as allies?

How can we educate end users?
- Should we mandate education? (high school)
- Is such education effective?

How can we improve end-user computer/security literacy?

How can we build in security so it happens automatically:
- reflecting end user expectations
  - ex. closing laptop or walking away from computer

How can we build a usable encryption system for laptops?
  - renewable data...

Where are we suffering greatest losses due to lack of usable security?
- What should we focus our attention on?

Where are the greatest leverage points for advances?
Approaches to Usable Security Session Notes

- When we protect users automatically, how much/when do we tell users?
- How can we make security configuration usable?
  - With appropriate feedback so users understand what they are getting
    - Task relevant
- How can we meet organizational needs without sacrificing user needs?
  - Who makes tradeoffs?
  - How do we identify costs & benefits for all parties involved?

- What can systems do when they fail in known and unknown ways? How do they recover?
  - What does it mean to recover in different domains?
  - How do I know the problem started so I go far enough back?
  - Usable and secure recovery?

- How can users safely determine the security status of their machines?

- What are the dimensions of usable security?

  - How can we collaborate:
    - Across companies?
    - Across geographical boundaries?
    - Share data sets?

  - How can we change the business/delivery models for security software so users stay up to date?
  - How can we make security maintenance effortless?

End of Session
Approaches to Usable Security Session Notes

Why is usable security different?
- Secondary task
- How do you create ecologically-valid risk experience in user studies?
- How do you capture changes in behavior over time?
- How can we study reasons organizations do or do not adopt technologies?

How can we change attitudes so security is not perceived as damage?
- How can we make it so security doesn't get in the way?
- Keeping in mind interests of each constituency

What are the gaps between what people think their level of security protection is & reality?
Why is there a gap?
How can we assure that security is appropriately set?
What is appropriate?

What are better methods for understanding user needs & prototyping?
Can we come up with simple sets of heuristics/design guidelines?
for USP?

How do we take bad practices and figure out guidelines for replacing them with something better?
- Short term & long term guidelines
How do we validate proposed guidelines?
How do we operationalize guidelines?
- Tools, models, methods
How do we balance competing goals?
- Organization, admin, user
Is there a science of security?
How do we design usable & secure systems?
Knowing there may be an active adversary?

What is good enough?
What is the role of training?
- Division of labor
What can you achieve through training? Limits of training?
Approaches to Usable Security Session Notes

How do we identify tradeoffs and ensure we pay attention to them throughout the design process?
- Include the right people?
- Train people from different disciplines to understand and work with each other on design teams?

Can we develop quantitative models to evaluate tradeoffs and inform decision-making?

Does the inherent complexity of security technology, user interface, etc., affect usability?
- Security systems for different kinds of users?
- Dynamic threats, needs change fast, as normal product cycle.

How do you design to deal with dynamic threats?

How can we best update user experiences in a rapidly changing threat environment?

Are there good measures of USP?
Business models

What are dimensions of usable security?

Is USB special

Metrics

Human behavior

Feedback about security decisions

Training and education

Prioritization

Design process

Recovery

Automation

Costs and benefits: tradeoffs

Internet/infrastructure/OS redesign

Standards and regulations

communication

Developer support

Administrative tools

Bundles of settings

Human-centered design

Shared data sets
Is USP special
Business models
Prioritization
Costs and benefits: tradeoffs
Standards and regulations
What are dimensions of usable security?
Metrics
Shared data sets

Developer support
Design process
Bundles of settings
Recovery
Administrative tools
Internet/infrastructure/OS redesign

Human-centered design
Human behavior
communication
Feedback about security decisions
Automation
Training and education
Understanding USP

• What are the dimensions of usable security and privacy?
  – Prioritization – where can we have greatest impact?

• Is usable security and privacy special?
  – How? (active adversary, secondary, hard to study)
  – How is it similar to other usability domains and what can we learn from them?

• How to influence development of more usable security?
  – Change business models so security subscriptions never expire
  – Standards and regulations

• How to evaluate usability and security?
  – Metrics
  – Shared data sets
  – Costs and benefits of security, tradeoffs
Approaches to USP

• Automation
  – Ex: How can we build usable encryption systems?
• User education
  – Role of education, limits of education
  – Should it be mandated?
  – What do users already know and how did they learn it?
• Human-centered design
  – Feedback and communication
  – Explicit conceptual model
  – Informed by observation of human behavior
    • Demographic/cultural differences
    • What tradeoffs will people make and how do we influence them?
Processes and tools

• Developer support tools
  – Guidelines, principles, design patterns, heuristics
  – Short term, based on what we know now
  – Long term research questions
  – How do we validate?

• Design process
  – Removing silos, training people from different disciplines to work together
  – How to deal with dynamic threats that develop between design iterations?

• Recovery
  – How do we recover from security problems in different applications?
  – How can we make recovery secure and usable?

• Administrative tools
  – Help administrators work on behalf of users

• Simplify user decisions
  – Bundles of settings
  – Secure defaults

• Internet/infrastructure/OS redesign