CSE 484 / CSE M 584 (Autumn 2011)

Human Factors in Security

Daniel Halperin
Tadayoshi Kohno

Thanks to Dan Boneh, Dieter Gollmann, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

CSE 484 / CSE M 584 (Autumn 2011)

Human Factors in Security

Daniel Halperin
Tadayoshi Kohno

Thanks to Dan Boneh, Dieter Gollmann, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

Updates, 11/21

- Second security review & current event due 12/2
  - Extra credit for every week early

Issues with Usability

1. Lack of intuition
   - See a safe, understand threats. Not true for computers
2. Who's in charge?
   - Doctors keep your medical records safe, you manage your passwords
3. Hard to gage risks
   - "It would never happen to me!"
4. No accountability
   - Asset-holder is not the only one you can lose assets
5. Awkward, annoying, or difficult
6. Social issues
7. Usability promotes trust

Issue #7: Usability Promotes Trust

- Well known by con artists, medicine men
- Phishing
  - More likely to trust professional-looking websites than non-professional-looking ones

Response #1: Education and Training

- Education:
  - Teaching technical concepts, risks
- Training
  - Change behavior through
    - Drill
    - Monitoring
    - Feedback
    - Reinforcement
    - Punishment
- May be part of the solution - but not the solution

Response #2: Security Should Be Invisible

- Security should happen
  - Naturally
  - By Default
  - Without user input or understanding
- Recognize and stop bad actions
- Starting to see some invisibility
  - SSL/TLS
  - VPNs
  - Automatic Security Updates

See Dan Simon's slides: http://research.microsoft.com/projects/SWSecInstitute/slides/simon.ppt

Response #2: Security Should Be Invisible

- "Easy" at extremes, or for simple examples
  - Don't give everyone access to everything
- But hard to generalize
- Leads to things not working for reasons user doesn't understand
- Users will then try to get the system to work, possibly further reducing security
  - E.g., "dangerous successes" for password managers

See Dan Simon's slides: http://research.microsoft.com/projects/SWSecInstitute/slides/simon.ppt

Response #3: "Three-word UI:" "Are You Sure?"

- Security should be invisible
  - Except when the user tries something dangerous
  - In which case a warning is given

- But how do users evaluate the warning? Two realistic cases:
  - Always heed warning. But see problems / commonality with Response #2
  - Always ignore the warning. If so, then how can it be effective?

See Dan Simon's slides: http://research.microsoft.com/projects/SWSecInstitute/slides/simon.ppt

Response #4: Focus on Users, Use Metaphors

- Clear, understandable metaphors:
  - Physical analogs; e.g., red-green lights
- User-centered design: Start with user model
- Unified security model across applications
  - User doesn't need to learn many models, one for each application
- Meaningful, intuitive user input
  - Don't assume things on user's behalf
  - Figure out how to ask so that user can answer intelligently

See Dan Simon's slides: http://research.microsoft.com/projects/SWSecInstitute/slides/simon.ppt

Response #5: Least Resistance

- "Match the most comfortable way to do tasks with the least granting of authority"
  - Ka-Ping Yee, Security and Usability
- Should be "easy" to comply with security policy
- "Users value and want security and privacy, but they regard them only as secondary to completing the primary tasks"
  - Karat et al, Security and Usability