10. Usable Access Control
ENEE 757 | CMSC 818V

Prof. Tudor Dumitraș
Assistant Professor, ECE
University of Maryland, College Park

http://ter.ps/757
https://www.facebook.com/SDSAtUMD

Today's Lecture

• Where we’ve been
  – Passwords
  – Biometrics
  – Kerberos
  – Authorization logic

• Where we’re going today
  – Usability issues in access control

• Where we’re going next
  – Security of Internet protocols
Security and Usability

- Software systems interact with humans
  - Administrators, end users

- Humans are participants in the security protocols
  - Trusted participants => human failures can compromise the system’s security!
  - Security features are no good if humans can’t use them or turn them off
    - Assignment from Lecture 6: have you enabled 2-factor authentication in Gmail?
  - A system isn’t secure if users always make mistakes
    - Example: choosing weak passwords

- In practice, a system cannot be secure unless it is usable

Example: Access Control Lists

UNIX permissions:
- rwx
- rwx
- rwx

AFS permissions
- owner
- group
- other

What are my effective permissions? Who else has access?
The Confused Deputy Problem

• Say I want to give Wei access to the reviews for today’s reading assignment
  – I must set up permissions so that Wei can access the reviews file and the folder where it is located
  – What other files can he access?
    • Can he read confidential information about other students?
    • Can he insert a backdoor in my script to make it look that has submitted a homework even though he did not?

• In this case, the AFS reference monitor is a “confused deputy”
  – I wanted to give Wei access to a single file
  – But the ACL permission model allows him to do additional things
  – I need to think carefully about what permissions I grant proactively
    • This is a usability issue!

2003 Senate Judiciary Committee Data Breach

• Republican clerk discovers that he can access the home directories of Democratic staffers
  – Reports, memos, etc. make their way to Republican staff and press

• Home directories set up with open permissions by confused system administrator
  – Was not instructed on setting the correct permissions
  – Assumed that access was restricted by some other means
Many Other Examples of Incorrect Permissions

Students suspended after seeing topless picture of teacher on school-issued iPad

What College Admissions Don't Like Seeing On Facebook: Vulgarity, Drinking, 'Illegal Activities'

Boss Allegedly Downloaded Nude Photos From Employee's Cell Phone, Showed Them Around The Office

How to Make User Interfaces Usable?

500 OOPS: child died

OK
**Eight Golden Rules of Interface Design**

[Ben Shneiderman]

1. Strive for consistency
   - Prompts, commands, actions
2. Enable frequent users to use shortcuts
   - Experts desire to increase the pace of interaction
3. Offer informative feedback
   - What does “child died” mean to user who never took an OS class?
4. Design dialog to yield closure
   - Confirm outcome of sequence of actions
5. Offer simple error handling
   - Design system so that user cannot make errors
6. Permit easy reversal of actions
   - Undo capability makes user more comfortable with the system
   - How to undo a security breach?
7. Support internal locus of control
   - Users should be initiators of actions rather than responders
8. Reduce short-term memory load
   - Keep displays simple, consolidate multiple pages, reduce window motion

**Usability Problems for ACL Interfaces**

Typically present a list of permission rules for users and groups

- **Scale**
  - Only one rule at a time is visible

- **Hierarchies**
  - Effective permissions may depend on the permissions of the parent folder

- **Groups**
  - Group membership information is not visible

- **Rule conflicts**
  - When rules interact, it isn’t clear what the outcome will be
The Expandable Grid Interface

[Reeder, Bauer, Cranor, Reiter, Vaniea CHI 2008, CHI 2011]

- Shows effective policy instead of policy rules
- Shows both user hierarchies (groups) and file hierarchies (folders)
- Shows entire policy on one screen

---

Tag Based Policy

- Determine access policies to semantic groups of files (specified by tags), rather than individual files and folders
- Alice lets Bob view files with type=photo and album=vacation_photos
  - Minimize policy mismatch: intended vs. specified
  - Allow fine-grained policy
  - Users prefer tags to folders (think iTunes)
  - File hierarchies vs. tags (semantic naming)
Logic-Based Access Control

Alice

Bob open(vacation_photos)

Prove Bob says open(vacation_photos) → Alice says open(vacation_photos)

Need Bob speaksfor Alice or similar credential

Options presented to Alice:

- Grant Bob credential to access VacationPhotos
- Add Bob to group CloseFriends who already have access
- Give Bob same rights as Charlie, who has access

Proof of Bob says open(vacation_photos) → Alice says open(vacation_photos)

Proof of Alice says open(vacation_photos)

Logic-Based Access Control – cont’d

- Provides high assurance of correctness

- Meaningful audit trail

- Allows fine-grained control, flexible policy
  - Users can create temporary access credentials
    - Useful for controlling access to physical resources, e.g. doors
  - Provide temporary access to email without revealing information that could be used at a later time or to access a different resource
    - Such as passwords
  - Can create complex access control credentials
    - Example: any three colleagues may access the resource when acting together, but at least three must cooperate to gain access
Application Example: The Grey Access Control System  
[Bauer et al., ISC’05]

- Access control system for virtual and physical resources (e.g. doors)
- Deployed in CMU’s Collaborative Innovation Center
  - Approximately 40 Grey-capable doors and 30+ users at the moment
  - Could also log into workstations using Grey-compatible Windows XP and Linux login modules
  - Access-control module for web servers
- Used distributed theorem prover
  - Additional requirements
  - System design led to new requirements
    - Interactive credential creation (to react to access requests)

Application Example: The Perspective File System  
[Salmon, Schlosser, Cranor, Ganger, FAST’09]

- Distributed file system aimed at home storage management
  - Aimed at non-technical users
  - Semantic groups (tags)
  - Peer-to-peer architecture (no master storage server)
- Key concept: view (semantic query + device)
  - Files matching the query are stored on the device
  - Allows users to manipulate file replicas
    - Using expandable grid -like interface
  - Access control
    - To grant access, allow the requester to store a replica on his device
    - Done reactively
  - File replication also provides fault tolerance
    - System computes replication degree automatically
How to Determine If A System Is Usable?

- Quantitative study
  - You have numbers (timing data, ratings of awesomeness)

- Qualitative study
  - You have non-numerical data (thoughts, opinions, types of errors)

Data to Collect In A Study

- Independent vs. dependent variables
- Performance (time, success rate, errors)
- Opinions and attitudes
  
  Please respond to the following statements:
  This user interface was difficult to understand
  1- Strongly disagree 2- Disagree 3- Neutral 4- Agree 5- Strongly agree

  This tool was fun to use
  1- Strongly disagree 2- Disagree 3- Neutral 4- Agree 5- Strongly agree

- Audio recording, screen capture, video, mouse movements, keystrokes
- Formative (initial) vs. summative (validate)
Even More Data To Collect

- Demographics
  - Age, gender, technical background, income, education, occupation, location, disabilities, first language, privacy attitudes, etc.

- Open-ended questions

Study Designs

- Between subjects
  - Each participant tests 1 version of the system
  - You compare the participant groups
  - Groups should be similar (verify!)
    - Avoid systematic bias introduced by gender, age, experience level, etc.

- Within subjects
  - Every participant tests everything
  - Very important to randomize order!
    - Avoid uncontrolled factors that are functions of time, place or experimental units
  - Fewer participants

- The data analysis depends on the study design
Sources

• Various slides from Lorrie Cranor, Lujo Bauer and Michelle Mazurek

Logistics

• On Monday, we are starting the Network Security module

• Survey for improving the course content and format in the future
  – Log into Elms at http://elms.umd.edu
  – CMSC818V > Quizzes
    • Elms calls this a “quiz” but it’s not graded
    • Your responses are anonymized
  – Please answer all 8 questions
Review of Lecture

• What did we learn?
  - Usability challenges for access control lists
  - Potential solutions: expandable grid, semantic tags, logic-based access control
  - System examples: Grey, Perspective
  - Rules for interface design and for user studies

• Paper discussion: “Exploring Reactive Access Control”
  - Discussion lead: Wei
  - Scribe: Ziyun

• What’s next?
  - Security of Internet protocols