CS 563 - Advanced Computer Security: Human Factors

Professor Adam Bates
Fall 2018
Learning Objectives:
• Discuss the practical consideration of usability of security mechanisms and concepts
• Understand how usability can be incorporated into a broader research agenda

Announcements:
• Reaction paper was due today (and all classes)
• “Preference Proposal” Homework due 9/24

Reminder: Please put away (backlit) devices at the start of class
Why Johnny Can’t Encrypt

• Security mechanisms are only effective when used correctly
  • Invoked? configured?
• This makes security a user interface problem
• Case Study: Investigate PHP 5.0
  • Cognitive Walkthrough
  • Laboratory User Tests
• 2015 USENIX Security “Test of Time” award recipient

“Putting your text in Pig Latin isn't the same as encrypting.”
We can call security software/features “usable” if the people who are expected to use it…

• are made aware of the tasks they need to perform
• are able to understand how to succeed at those tasks
• don’t make dangerous errors while completing tasks
• are comfortable enough to continuously use the software
Usabile Security Challenges

• Lack of Motivation: Users will invest only limited attention/capital to maintain security

• Understanding Abstractions: Abstractions used by domain experts (e.g., security policy) may be obtuse to end users.

• Providing good feedback: How can software guide the user to the security outcome they ‘really want’?

• ‘Barn Door’ Property: Once an asset is unprotected even once, its security may be irrevocably compromised.

• ‘Weakest Link’ Property: Securing assets must be comprehensive; user engagement cannot be intermittent.
• “Pretty Good Privacy”
• Software for encrypting and signing data
• GUI with plug-in for easy (?) use with email clients
Cognitive Walkthrough

• Visual Metaphors:
  • Public vs. Private Keys
  • Signatures & Verification

• Different key types:
  • Compatibility increases complexity
  • Keys listed as users
Cognitive Walkthrough

- **Key Servers:**
  - Vital to using PGP, but buried in menus
  - Connection to remote resource is non-obvious
  - Push for locally revoked keys is not automatic
Cognitive Walkthrough

• Key Management:
  • Unneeded confusion in interface
  • **Validity versus Trust?**

![Example of Key Management Interface]

• Presence of Irreversible Actions (e.g., key deletion)
• Consistency of terminology
• Too much information exposed when not needed
User Tests

- PGP 5.0 with Eudora
- 12 participants all with at least some college and none with advanced knowledge of encryption
- Participants were given a scenario with tasks to complete within 90 min
- Tasks built on each other
- Participants could ask some questions through email
• **Scenario**: Subject is ‘campaign coordinator’ that needs to send private emails to campaign team.

• **Tasks**: Generate a key pair, acquire team’s public keys, type email, sign email using private key, encrypt using team’s public keys (different versions), send result.

• Experimenter posed as team member to send instructions and feedback (sidequest: decrypt message)
User Test Results

• Users sent message in plaintext (3)
• Users used their public key to encrypt (7) and could not recover (5)
• Users could not encrypt at all (1)
• Users could not decrypt messages (2 succeeded)
• Users could not handle legacy keys (1 succeeded)
• Only 3 users completed the basic process of sending and receiving encrypted emails.
Takeaway

If an average user of email feels the need for privacy and authentication, and acquires PGP with that purpose in mind, will PGP's current design allow that person to realize what needs to be done, figure out how to do it, and avoid dangerous errors, without becoming so frustrated that he or she decides to give up on using PGP after all?
If an average user of email feels the need for privacy and authentication, and acquires PGP with that purpose in mind, will PGP's current design allow that person to realize what needs to be done, figure out how to do it, and avoid dangerous errors, without becoming so frustrated that he or she decides to give up on using PGP after all?
Aside: Can we fix the user?

Security Design: Stop Trying to Fix the User

• “The problem isn't the users: it's that we've designed our computer systems' security so badly that we demand the user do all of these counterintuitive things.”

• Usable security does not mean "getting people to do what we want." It means creating security that works, given (or despite) what people do.

• Schneier suggests that solution is not interventions to ‘fix’ user, but the design of systems that work in spite of the user.
• Foundation concept of secure system design and opsec

• What do I want to protect?

• Who do I want to protect it from?

• How bad are the consequences if I fail?

• How likely is it that I will need to protect it?

• How much trouble am I willing to go through to try to prevent potential consequences?
Do threat models improve real-world security?
• Introduce threat modeling to New York City Cyber Command (NYC3)

• Infrastructure accessed by 60 million tourists and 300,000 employees each year

• Introduce 25 NYC3 employees to threat model training (‘Center of Gravity’ framework)

• Monitor their usage at 30, efficacy at 120 days
• In military strategy, CoG is the primary asset(s) needed to achieve mission objective.
• Pilot study to test relevance, clarity, validity of protocol

• Recruit NYC3 employees over company email (25)

• Participants…
  • fill out 29 question baseline survey
  • complete 60 minute training
  • 60 minute individual session
  • fill out 29 question post-training survey
  • complete 30 day follow-up survey

• Long-term evaluation of security incidents at 120 days
### CoG Analysis

**Center of Gravity Worksheet**

1. Please state your work section’s objective/mission:
2. What assets are used to accomplish this mission?
3. What is your center of gravity?
4. Critical Capabilities
5. Critical Requirements
6. Critical Vulnerabilities
7. Threat Capabilities
8. Threat Requirements
9. Defense Plan
Participants

- 25 participants completed study
- 37% of NYC3
- Pre-Intervention Baseline
  - Security assessed through city-specific policies, NIST framework, accreditation process.
  - Participants report that such guidelines were not frequently applies
  - Many were unaware of such programs

<table>
<thead>
<tr>
<th>ID</th>
<th>Duty Position</th>
<th>IT Exp (yrs)</th>
<th>Trng. (yrs)</th>
<th>Educ.¹</th>
</tr>
</thead>
<tbody>
<tr>
<td>P01</td>
<td>Leadership</td>
<td>16-20</td>
<td>6-10</td>
<td>SC</td>
</tr>
<tr>
<td>P02</td>
<td>Data Engr.</td>
<td>16-20</td>
<td>6-10</td>
<td>G</td>
</tr>
<tr>
<td>P03</td>
<td>Sec Analyst</td>
<td>11-15</td>
<td>0-5</td>
<td>SC</td>
</tr>
<tr>
<td>P04</td>
<td>Sec Engineer</td>
<td>11-15</td>
<td>0-5</td>
<td>BS</td>
</tr>
<tr>
<td>P05</td>
<td>Governance</td>
<td>16-20</td>
<td>6-10</td>
<td>SC</td>
</tr>
<tr>
<td>P06</td>
<td>Sec Engineer</td>
<td>6-10</td>
<td>11-15</td>
<td>P</td>
</tr>
<tr>
<td>P07</td>
<td>Sec Engineer</td>
<td>0-5</td>
<td>6-10</td>
<td>G</td>
</tr>
<tr>
<td>P08</td>
<td>Net Admin</td>
<td>21-25</td>
<td>6-10</td>
<td>G</td>
</tr>
<tr>
<td>P09</td>
<td>Sec Engineer</td>
<td>11-15</td>
<td>0-5</td>
<td>SC</td>
</tr>
<tr>
<td>P10</td>
<td>Sec Engineer</td>
<td>11-15</td>
<td>6-10</td>
<td>BS</td>
</tr>
<tr>
<td>P11</td>
<td>Net Admin</td>
<td>16-20</td>
<td>6-10</td>
<td>BS</td>
</tr>
<tr>
<td>P12</td>
<td>Sec Engineer</td>
<td>25+</td>
<td>6-10</td>
<td>G</td>
</tr>
<tr>
<td>P13</td>
<td>Sec Analyst</td>
<td>0-5</td>
<td>0-5</td>
<td>BS</td>
</tr>
<tr>
<td>P14</td>
<td>Sec Engineer</td>
<td>11-15</td>
<td>0-5</td>
<td>BS</td>
</tr>
<tr>
<td>P15</td>
<td>Sec Engineer</td>
<td>16-20</td>
<td>25+</td>
<td>SC</td>
</tr>
<tr>
<td>P16</td>
<td>Support Staff</td>
<td>6-10</td>
<td>0-5</td>
<td>BS</td>
</tr>
<tr>
<td>P17</td>
<td>Sec Analyst</td>
<td>16-20</td>
<td>16-20</td>
<td>G</td>
</tr>
<tr>
<td>P18</td>
<td>Sec Engineer</td>
<td>21-25</td>
<td>16-20</td>
<td>G</td>
</tr>
<tr>
<td>P19</td>
<td>Sec Analyst</td>
<td>21-25</td>
<td>6-10</td>
<td>SC</td>
</tr>
<tr>
<td>P20</td>
<td>Leadership</td>
<td>11-15</td>
<td>6-10</td>
<td>G</td>
</tr>
<tr>
<td>P21</td>
<td>Sec Analyst</td>
<td>0-5</td>
<td>6-10</td>
<td>G</td>
</tr>
<tr>
<td>P22</td>
<td>Leadership</td>
<td>11-15</td>
<td>6-10</td>
<td>G</td>
</tr>
<tr>
<td>P23</td>
<td>Sec Analyst</td>
<td>16-20</td>
<td>6-10</td>
<td>BS</td>
</tr>
<tr>
<td>P24</td>
<td>Leadership</td>
<td>0-5</td>
<td>0-5</td>
<td>BS</td>
</tr>
<tr>
<td>P25</td>
<td>Leadership</td>
<td>0-5</td>
<td>0-5</td>
<td>G</td>
</tr>
</tbody>
</table>

¹ SC: Some College, BS: Bachelor’s, G: Graduate degree, P: Prefer not to answer
Results

- Participants reported that threat modeling gave them a better understanding of capabilities and requirements (n=12)
- Participants agreed threat modeling was useful in their daily routine (n=23)
- Many report improved ability to monitor critical assets (n=17), mitigate threats (n=16), respond to incidents (n=15)

Figure 3: A cumulative distribution function (CDF) for participant subtask completion times.
Results (30 days later)

- Perceived efficacy of framework decreased only slightly (not significant)
- Still using mitigation strategies from threat modeling (n=21) or incorporating concepts into routine (n=20)
- NYC3 began to institutionalize threat modeling as a result of participant feedback

Figure 4: Perceived efficacy after using threat modeling for 30 days.
Results (120 days later)

- Inspect participants’ threat models to identify actionable defense plans:
  - Testing readiness (test defense plans)
  - Secure account permissions
  - Protect physical network assets
  - Crowdsourcing assessments (bug bounty program?)
  - Increased sensor coverage
  - Segment legacy systems
  - Protect against data corruption
  - Reduce human error (e.g., two person change control)
Results (120 days later)

• Inspect participants’ threat models to identify actionable defense plans:
  • Testing readiness (test defense plans)
  • Secure account permissions
  • Protect physical network assets
  • Crowdsourcing assessments (bug bounty program?)
  • Increased sensor coverage
  • Segment legacy systems
  • Protect against data corruption
  • Reduce human error (e.g., two person change control)
• Inspect participants’ threat models to identify actionable defense plans:
  • **Secure account permissions**: Seven foreign access attempts blocked because of 2FA implemented after training
  • **Crowdsourcing assessments**: Pilot bug bounty program yielded 3 previously-unknown vulnerabilities
  • **Sensor Coverage**: 1331 new endpoint sensors deployed, prevented 541 intrusion attempts (59 critical, 135 high severity).
NYC3 Takeaways

• Threat modeling facilitated adoption of ‘best practices’
• Hands-on learning is effective (n=24)
• Social and organizational support may speed adoption
• Threat modeling improved threat advocacy with leadership
• Knowledge retention (e.g., terminology) is ongoing challenge

• Thoughts? Limitations?
• “Security as a secondary objective” already guides the way we design and evaluate solutions
  • e.g., why do we have performance evaluations?

• Usable security methodologies allow us to measure the human capitol of systems

• If your design interacts with a human, usability should* be as central to your eval as any other benchmark

• Incorporating usable security into your research will give you an “unfair” advantage when publishing
Usable Security: Looking Forward

• Where to look for literature: “Big 4” security conferences (IEEE S&P a.k.a. Oakland, USENIX Security, CCS, NDSS), SOUPS workshop, security track at CHI.

• Hot Topics in Measurement (not exhaustive):
  • User Authentication (passwords, pins, meters, …)
  • Web Security, Social Networks, Secure Messaging
  • Emerging technology (IoT, VR)
  • Risk Perception, Attitude towards Privacy + Security
  • Usability of Security for Developers
  • Real World Testimony & Analysis (Enterprise, Developing World)