6.033 Computer System Engineering

Spring 2009

Preparation for Recitation 22

Read *Why Cryptosystems Fail*. You may wish to skim the abstract, introduction, and conclusion first, because they will help you to focus on the parts of the paper that support the author's main claims. As always, you should read critically and be on the lookout for additional gems, and for arguments that are missing or whose framing de-emphasizes certain points.

This paper is about a philosophy of cryptosystem design, with a focus on their use in financial institutions, and particularly in ATM (Automated Teller Machine, not Asynchronus Transfer Mode) networks. Although it may not be immediately obvious, this paper is closely related to other papers we have read, such as the Therac-25 paper. Think about these connections as you read.

Over half of the paper is devoted to examples of ways in which ATM networks could fail or have failed. This part of the paper is very entertaining, but it can be difficult to keep the big picture in mind while reading about the individual exploits and problems. Pay attention to the section headings (which you may wish to skim before diving into the text) in order to keep your bearings. For each incident, before moving on, spend a few moments thinking about the lessons that it teaches, and how the problem could have been avoided.

Here are some specific issues to think about while you are reading:

- What is a cryptosystem? What elements (machine, communication, and human) does it encompass? How do its components make the concerns of this paper similar to those of the Therac-25 paper, and dissimilar to certain other papers we have read?
- What are the end-to-end requirements of a cryptosystem? (Be specific; don't just say "security", because then that term itself requires a definition.) Can those requirements be achieved by composing modules with certain characteristics? Where and how is the end-to-end check performed, if one is required?
- How is achieving security similar to achieving reliability in networking or safety in other systems?
- Suppose you have built a cryptosystem from a set of components plus a way of composing them. How could you compute a quantitative measure for the security of the system or of some component? Isn't this what standards organizations have to do when certifying a component? Are Anderson's suggestions applicable to this issue?
- How can an organization test the security of a system? Isn't this an important part of the process that Anderson omits?