Erik Sommer
1/11/2012
SE-4930

Article Review 2
Thinking Inside the Box: System-Level Failures of Tamper Proofing

For the second article to review for SE-4930, I decided to read a paper from the *2008 IEEE Symposium on Security and Privac* entitled Thinking Inside the Box: System-Level Failures of Tamper-Proofing. This paper was written by Saar Drimer, Steven J. Murdoch, and Ross Anderson from the University of Cambridge. The main focus of the article is on the PED system used for credit card payments in Europe. It can be found at the following URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4531159&isnumber=4531132

In the U.S., we are used to the traditional credit-card swipe systems. However, while the European credit cards also have the magnetic strip used for swiping, they largely depend on a smart-card system. Their credit cards have a chip embedded in them that contains the information about the card. When a user wants to make a payment, they insert their card into a reader, and then enter their PIN on a pad mounted to the reader. This pin is sent to the card and, if verified, the transaction is authorized. Essentially it operates much like ATMs do, but the card itself verifies the PIN removing the need for a direct contact to the banking institution.

The readers that are used all over Europe are supposedly tested and certified by organizations to incorporate anti-tampering mechanisms. These mechanisms can be divided into 3 different categories: tamper evident, tamper resistant, and tamper response. Tamper evident seals are simply seals that, upon inspection, may reveal that a device has been opened. Tamper resistant devices try to resist unauthorized access. The two models discussed in the paper implement a variety of mechanisms to detect tampering, such as adding switches to detect when the case is open. Some of the more advanced methods used to protected the models include “potting” components (covering them with an “opaque solid compound that cannot easily be cut, dissolved, drilled, or milled”) or embedding a thin grid of wires in the board to detect drilling. Finally tamper response mechanisms take action when tampering is detected, like erasing critical data. In the writer’s experiments, though, they were able to retrieve card and pin data without triggering the tampering mechanisms. One method the used was sticking a paperclip though a hole on the board while the other was to string a thin wire to a connector. With this access they could record thousands of transactions nearly transparently. Obvious this is not a secure system.

In order to protect against these attacks, the writers suggested a few different methods, such as encrypting the PIN or using the iCCV protocol which requires a different reader. One of the more interesting ideas they had for security was to setup a two-way channel of communication. For example,

when a user wants to purchase an item, they put their card in the reader, and then a text message is sent to their registered phone with an authorization code. That code can then be entered to authorize the transaction. Although this would not be suitable in high-traffic situations, it was an extremely interesting concept. The article also touched on the use of near-field-communication (NFC) devices becoming more prominent, and how their use may be the way of the future. Ironically, it mentions that RFID seems to be the way the US is heading, but that was back in 2008 and payment via RFID is not commonplace.

But the vulnerabilities in the hardware are only part of the problem. There are specifications out there, but they are for individual modules. Designers can design their modules to meet the specifications, but they have no idea how the entire system works together, and therefore are unable to find all of the vulnerabilities. Looking closer at the specifications reveals even more issues. The specifications that designers are using are 500+ pages each, with different companies adding an additional 300 pages their own "requirements". The article make the point that designers do not read all of these pages, and therefore not all of the requirements are met. These requirements need to be shortened down into a palatable form that is clear and concise, not surrounded by tons of fluff. The flaws in the systems that are used all over Europe can be traced back to the design specifications and the lackluster oversight. In the end, the writers of the article stress that a single, unified specification is needed, and that a strong organization to certify devices is required. There should be monetary incentives for finding flaws and harsher punishments for not meeting specifications. In essence, Europe needs an SRS that is strictly enforced to ensure that their payment system is secure.

Reading though this article, I realized how important a detailed process is for a project. While they did had some specifications, they were written in manager-speak and unclear. This left enough room for corners to be cut to keep costs down, resulting in an insecure system that processes sensitive data. Millions of people depend on this system every day, but devices that "pass" the test are still exploitable by malicious users. Following a strict SRS will help guide the development and close many of the potential holes. But there are some things that, as a developer, I can't anticipate happening. That is why it's important to have the things that I created tested by a third-part who has some sort of incentive to find exploits, whether it be for a grade or monetary gain. All in all, this paper has taught me that making a product before specifications are laid out is simply a recipe for disaster.