Project proposal

**Background**

The university, along with many other institutions and systems, uses smartcards that use the MIFARE classic chip.

In a presentation by Henryk Plötz and Karsten Nohl at the Chaos Communication Congress in 2007, they described how the security of the MIFARE classic chip relied largely on obfuscation of the implementation details. This flaunts Kerchoff’s principle, which states that encryption security should rely solely on the key and that not on the secrecy of the system. They reverse engineered the MIFARE classic chip and discovered various weaknesses with the chip which won’t be discussed at length here.

Many compromises were made on the security of the tag as a result of the very limited computing power of the chips, but others were unnecessary, and advances in available computing power should allow for better cryptographic methods.

One more general weakness of the MIFARE chip, in common with most other RFID tags, is the use of symmetric cryptography. When used in a large organisation such as the University to enforce door access control (among other things), limited on-chip memory means there are typically only a small number of private keys stored in most cards that are used to open most doors. If any card containing some common key is compromised, that card can open any door responding to that key until either the card is reprogrammed, or the key for the doors are changed – at great inconvenience to all users.

The successor to the MIFARE classic, MIFARE plus, solves many of the problems with the classic chip, but still uses symmetric cryptography so does not solve the stolen key problem.

A University working group is currently exploring a successor system, and a successful implementation of a prototype system could help usefully inform this process.

**The project**

An obvious solution to the stolen smartcard problem would be to use asymmetric cryptography, such as digital signature algorithms, where each card can hold a unique private key and access control devices contain no secret keys. There are many protocols in existence, such as NIST 800-73, OPACITY, and PLAID, and slightly different ways to implement them. The main focus of the project will be to look at these protocols and methods, compare them, and implement one of them.

The MIFARE classic card has other weaknesses too, and I will identify them and assess whether newly implemented protocol addresses these weaknesses, or whether they can be modified to do so.

**Possible extensions**

* Experiment with the timing predictability of the smartcard used, and attempt to implement a distance-bounding protocol similar to that used by the MIFARE plus EV1 chips to prevent relay attacks.
* Implement more than one of the protocols and directly compare their performance and security.
* Either make the card backwards compatible with MIFARE classic, or propose some other smooth rollout strategy, so that the new system could seamlessly replace the old university card system.

Success criteria

The metrics that will be used to judge the success of the project will be determined largely by its suitability as a hypothetical replacement to the current university smartcard system. It should not introduce needless inconvenience, and it should address the primary drawbacks of the MIFARE classic system. The following criteria will be used:

* Should take no more an a second to determine whether a card is valid.
* Access rights of a card should be revocable without having to modify the card itself.
* system should be flexible – unlimited doors, each with different access privileges etc.
* Should be secure – Very resistant to most, or all, likely forms of attack.
* Should be reasonably cheap to produce at a large scale, ideally no more than the current university system.

Timetable

**Weeks 1 and 2**

Do preliminary reading of information regarding MIFARE, asymmetric encryption, digital signature algorithms and specific protocols. Practice using the hardware and development software to create very basic smartcard applications.

**Week 3**

Read through options for protocols, assess their security and how efficient their implementation is likely to be, and make a decision on what will form the basis of the new system.

**Week 4 to 7**

Basic implementation

**Week 8 and 9**

Analyse the implementation by considering likely forms of attack. If there are security flaws that can easily be fixed, apply the fix.

**Weeks 10 and 11**

With a basic implementation completed, write a progress report.

**Week 12**

Tie up loose ends in the implementation so far. At this point, this should be quick work. Assess how well the implementation satisfies the success criteria at this point.

**Weeks 13 to 15**

If work is on schedule at this point, the optional extensions will be implemented.

**Weeks 16 to 20**

Throughout the process, notes will be taken regarding any research, decisions made or work done. During this phase, gaps will be filled in and brief notes will be expanded on, and the dissertation will reach its final form, ready for submission.

Special resources

* Java Card. Costs a few pounds online.
* Development software for Javacard. Can use NetBeans Java Card plugin.
* Contactless ISO 14443 card reader. Available on Amazon for £39.

Supervisor

This project was suggested by Dr. Markus Kuhn, who will also be supervising it.