Paper III: Lessons from the Norwegian ATM system

Lessons from the Norwegian ATM system

Kjell J. Hole, Vebjørn Moen, and André N. Klingsheim

Abstract

This paper first analyzes a Norwegian court case involving a stolen ATM card misused by unknown person(s) who somehow knew the card's PIN. A fictitious attack scenario shows that as long as the Norwegian ATM system used DES to verify a PIN, it was possible for a skilled cracker to determine the PIN belonging to any Norwegian ATM card. It's then discussed how new and open development processes can lead to improved security and usability in future banking systems, as well as better legal protection for the bank customers.

1 Introduction

The authors study a court case from 2004 concerning a Norwegian citizen whose ATM (Automatic Teller Machine) card was stolen and later misused by unknown person(s) [1]. These thieves had somehow obtained the correct PIN (Personal Identification Number) associated with the card.

According to the judge's verdict [1], the Norwegian ATM system employed *single* DES (Data Encryption Standard) [2], or simply DES, to verify a PIN at the time the card was stolen. Despite this assumption, the bank responsible for issuing the card claimed it was impossible for the thieves to ascertain the PIN from the information on the card's magnetic strip during the hour it took from the card was stolen until it was first misused.

We'll introduce a simple model of an ATM system that uses DES to verify PINs and describe a theoretical attack scenario utilizing a two-step attack strategy. The first step is time-consuming and can be carried out before an ATM card is stolen. The second step can then ascertain the PIN belonging to the ATM card in a matter of seconds. In fact, after the first step is completed, the second step can determine the PIN belonging to *any* card issued by the bank in our model.

The Norwegian card owner lost his case because the judge decided it was impossible to establish the PIN during the available time. The attack scenario shows that the judge based his decision on wrong information.

After citing additional information concerning the Norwegian Internet banking systems during 2003 and 2004 [3], the authors assert that the Norwegian bank community's refusal to provide adequate security information is a threat to the citizens' legal protection. We also explain why this secrecy causes the security of banking systems to deteriorate over time. Finally, we make clear why new and open development processes can lead to both improved security and better legal protection in the future. These processes are discussed in some detail.

While we only study Norwegian banking systems, our findings are applicable to many other commercial systems. We leave it to the reader to apply our insights to other systems. The reader should note that the fictitious attack scenario is no longer a threat since today's Norwegian ATM system is based on triple DES.

2 The court case

In 2001, unknown person(s) stole two shoulder bags from a Norwegian citizen—we'll call him Mr. A—at an airport in Spain [1]. Mr. A lost his wallet with six Norwegian payment cards, while his wife lost four cards. Unidentified thieves later misused two of the cards.

Most payment cards issued by Norwegian banks are ATM ready and consist of two parts, one Visa/MasterCard part used abroad and one BankAxept part used in Norway. Since the *same* PIN is associated with both parts, it's theoretically possible to calculate the PIN from the BankAxept part and misuse the Visa/MasterCard part and vice versa.

We'll concentrate on the court case concerning the misuse of one of Mr. A's stolen cards. Since the court didn't consider the Spanish ATM system at all during the trial, we'll only consider the Norwegian ATM system. The stolen card was a MasterCard/BankAxept card issued by a particular Norwegian bank. Using this ATM card, unknown criminals were able to withdraw more than 9,000 Norwegian kroner (NOK) about an hour after Mr. A's bags were stolen. The ATM card was misused four times during a period of about 6 minutes. Each time, the right PIN was entered on the first attempt according to the verdict [1].

In Norway there is a national committee called "Bankklagenemda", established to solve disputes between Norwegian banks and their private customers. "Bankklagenemda" didn't believe the unidentified criminals had obtained the correct PIN by looking over Mr. A's shoulder, because the last time he had used the card was at the airport in Norway before leaving for Spain. This argument is strengthened by the fact that one of his wife's stolen cards with a different PIN—not used at the Norwegian airport—was also misused in Spain.

While Mr. A claimed the only written copy of his PIN was kept in a safe at home, "Bankklagenemda" ruled that he must have kept the PIN together with the card in the stolen wallet. With reference to the relevant Norwegian law, "Bankklagenemda" then made Mr. A responsible for 8,000 NOK of the loss.

Mr. A didn't accept the ruling and took the case to court in 2004. The defendant was the bank responsible for issuing the credit card. According to the scenario favored by the court, Mr. A's PIN was first encrypted with DES and then stored on the card's magnetic strip during the production of the card. The bank's two expert witnesses claimed it was impossible to determine the PIN from the information on the magnetic strip during the hour it took from the card being stolen to the first time the card was misused.

The expert witness for the plaintiff on the other hand, explained how most of the cracking could be done in advance if the criminals had prior access to a small number of cards issued from the bank. The judge chose to believe the bank's experts and concluded that the plaintiff most likely had kept a copy of his PIN in the stolen wallet. In the three following sections we'll explain why

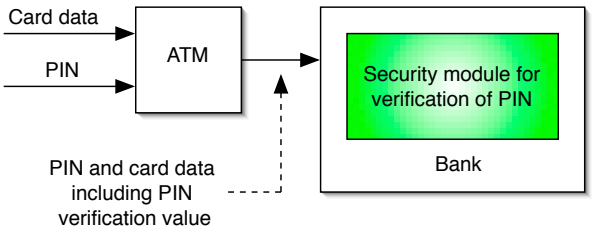

Figure 1: Simplified model of ATM and bank.

the plaintiff's expert witness was correct.

3 Model of ATM system

Figure 1 depicts a simplified model of the Norwegian ATM system during the period it employed DES to verify a PIN. To withdraw cash from the ATM in the model, a customer places her card in the card reader and types the PIN on the keypad. Information on the card's magnetic strip, including a value denoted the *PIN verification value*, is first read and then transmitted over a secure channel to the bank. As we shall see later, the PIN verification value is of particular interest to us.

The bank in Figure 1 employs a (hardware) *security module* [4] to verify the PIN. The verification process is shown in Figure 2. The security module uses DES encryption with a 56-bit secret key protected within the module. The 64-bit block of input data to the DES encryption consists of the customer's PIN and data from the ATM card's magnetic strip. Note that the PIN verification value from the card is not encrypted. Instead, the 64-bit output block from the DES encryption is transformed and compared to this PIN verification value. If the two values are equal then the PIN is accepted by the bank and the customer is allowed to withdraw cash from the ATM.

The transformation in Figure 2 is not the same in all real ATM systems. The exact function used in the Norwegian system is not publicly known. In our simplified model we only assume that the transformation produces a 16-bit result. For simplicity, all possible 16-bit values are assumed to be equally likely.

The same type of security module is used both during the production of ATM cards and the real-time verification of PINs in our model. The part of Figure 2 starting with the input to the DES encryption and ending with the output from the transformation defines how a 16-bit PIN verification value is generated. Clearly, to obtain a match between the pre-calculated PIN verification value and the value generated during a bank's real-time PIN verification, both values must be based on the same DES key. In our model, this DES key is used to verify

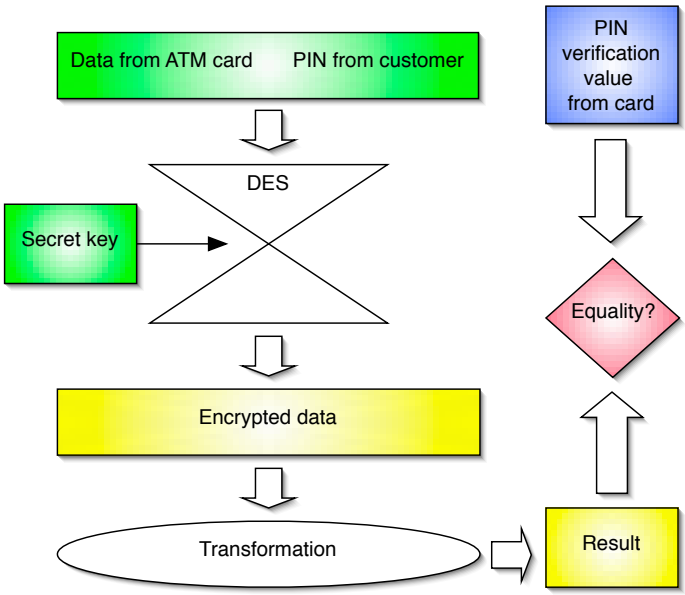

Figure 2: General outline of the PIN verification inside the security module.

the PINs belonging to *all* cards issued by a given bank. Different banks have different DES keys.

For a given 64-bit input block to DES, observe that many candidate keys result in the correct 16-bit PIN verification value. Hence, the PIN verification value partly determines the correct key, or, in more formal terms, the PIN verification value constitutes a 16-bit condition on the secret key.

4 How to determine a PIN

Using the ATM system model, we'll show how a fictitious cracker could establish the PIN belonging to any ATM card issued by a particular Norwegian bank during the time DES was used to verify PINs. To do this, the cracker had to be able to execute the internal operations of the security module (see Figure 2). Even if he didn't succeed in buying a module, it's still reasonable to assume he could determine its operations.

Several publicly available ISO (International Organization for Standardization) documents describe the techniques and data formats employed in security modules during the time DES was utilized [5, 6]. Furthermore, it's likely that, throughout the many years the modules were employed, a few of the manufacturers or some of the many banks leaked documents describing the modules' internal operations. Disgruntled former bank employees with intimate knowledge of the ATM system also had the necessary knowledge. Hence, an attacker could obtain the needed information to write a program to carry out all operations of a module. He could also buy a card reader to obtain the information on the magnetic strip of an arbitrary ATM card. The card reader could be connected to a computer to efficiently feed the data from a card into the program

mimicking the operations of the security module.

Suppose an attacker wanted to learn the PIN belonging to a particular ATM card. For now we assume he knew the DES key in Figure 2. (we'll show how to determine this key later on.) The attacker simply had to let the program try different PINs until the transformed value of an encrypted PIN was equal to the PIN verification value (see Figure 2). Note how this very simple technique gives the correct PIN because the PIN verification value is available on the ATM card itself!

Four-digit PINs were employed together with the ATM cards. If all 10,000 possible PIN values were used equally often, then the discussed program had only to try 5,000 PINs on average to determine the correct PIN belonging to a card.

5 How to determine a DES key

Our fictitious cracker had to discover the secret DES key stored in a bank's security module before he could run the program to establish PINs belonging to ATM cards issued by the bank. We'll now study how to determine such DES keys.

5.1 DES crackers

DES [2] is a block cipher encrypting 64-bit blocks of input data. The uppermost block in Figure 2 represents one of these input blocks. The encryption of each new input block is determined by the same secret 56-bit DES key, resulting in a stream of 64-bit blocks of encrypted output data.

EFF (Electronic Frontier Foundation) needed 18 months to design and build a computer to crack DES keys [7]. On July 17 in 1998 EFF determined the DES key used by RSA Data Security to encrypt a secret message. EFF's DES cracker simply tried a large number of different keys during a time period of 56 hours to obtain the correct DES key. Early in 1999 a large network of computers calculated a correct key in 22 hours and 15 minutes [8]. Using programmable logic arrays technology, Bond and Clayton later implemented an efficient DES cracker for 995 US dollars [9].

Cryptographers had already asserted for many years that 56-bit DES keys were too short to offer a high level of security [10]. EFF's DES cracker proved this assertion once and for all, and showed at the same time that many security systems based on DES didn't have the needed level of security. The DES-based security modules in the Norwegian ATM system were examples of such unsafe systems.

5.2 Access to ATM cards

It has been possible for a skilled attacker to crack DES keys since the early '90s. In the following we'll describe how the attacker could learn the DES key belonging to a bank's security module, using a DES cracker and a small number of ATM cards with known PINs. We first introduce a simple attack to establish how many ATM cards the cracker needed to determine a DES key. A more efficient attack is then described.

Assume that the attacker had one ATM card with known PIN, e.g., his own card. He then knew the complete content of the 64-bit input block to the DES encryption in Figure 2. However, the attacker had a problem because he didn't know the corresponding 64-bit output block—only the PIN verification value assumed to consist of 16 bits in this description. The problem for the attacker was that a large number of keys caused the transformation in Figure 2 to produce a value equal to the PIN verification value.

During the attack it was therefore necessary to try all 2^{56} keys and collect the 2^{40} keys resulting in equality in Figure 2. The number 2^{40} follows from the observation that the cracker had a 16-bit condition on the key. Consequently, there are $2^{56}/2^{16} = 2^{40}$ keys giving the correct PIN verification value. However, only one key can result in the correct determination of PINs belonging to all ATM cards issued by the bank.

The correct key was contained in the set of the 2^{40} remaining keys. This set could be reduced to a set of 2^{24} keys by trying all the 2^{40} keys together with a *new* ATM card with a different PIN and PIN verification value. To obtain the new card the attacker could open another account, have another person open an account, or simply steal the card and PIN from one of the bank's customers.

The set of 2^{24} keys could be further reduced to 2^8 keys using a third card. The correct DES key was then determined utilizing a fourth card.

The outlined attack made it necessary to store 2^{40} keys. If the attacker had access to four different ATM cards with known PINs before the attack, he then could test each of the 2^{56} keys against the four cards right away and, thus, remove the need to store a large number of keys. This modification also reduced the number of candidate keys needed to be tried to 2^{55} on average.

It's possible to further simplify the attack when a card owner is allowed to change the PIN. An attacker can then obtain the needed pairs of PIN and PIN verification values from a single card by repeatedly changing the PIN and reading the corresponding PIN verification value from the card's magnetic strip.

5.3 The main point

Even today it takes some time to calculate the key belonging to a security module employing DES. The cracking of a key also requires purpose built hardware or a large collection of regular PCs. The important point is, however, that it's only necessary to determine *one* DES key per bank.

After a DES key is known, the simple program described earlier can ascertain the PIN belonging to any stolen ATM card issued by the bank by simply trying 5,000 PIN values on average. This program can be run on a small laptop today. The same was clearly true during the '90s. In other words, after a skilled attacker had determined the DES key belonging to a security module in a particular bank, an unskilled operator of a simple PC program could establish the PIN belonging to any stolen ATM card from this bank in a matter of seconds.

5.4 Did an attack take place?

According to one line of thought it's unlikely that real attacks took place because this would've lead to a massive number of unexplainable withdrawals from Norwegian bank accounts. Centers would've popped up where criminals could come to determine PINs belonging to stolen cards. Alternatively, criminals could've

"rented out" laptops running phase 2 of the PIN cracking to thieves stealing cards from a particular bank.

On the other hand, it's possible to argue that real attacks would not have lead to massive fraud. According to our model a thief must physically steal each ATM card. Most card owners report card theft to their bank right away. The bank then closes down the account making it impossible to withdraw any cash. The rapid closing of accounts forces a thief to steal no more than a few cards before he tries to withdraw cash from an ATM, which limits how many cards he can misuse during a given time period.

It follows that a group of thieves was needed to steal and successfully misuse, say, five hundreds cards per year. Because the ATM system has had, and still has, limits to how much cash a customer can withdraw from an account during a single week, the group members would've had to operate in different locations to steal enough cards to make the operation profitable. Attacks on different banks would've lead to a distributed geographical pattern of ATMs processing stolen cards, much like the pattern we've seen during the last decade.

The number of misused cards and the geographical pattern of abused ATMs depend on the number of criminal groups. The level of expertise and resources needed to determine one DES key per targeted bank, as well as the actual number of stolen cards each year, make it unlikely that more than a very small number of groups can have existed in Norway.

It's hard both for banks and outside experts to discover whether or not real attacks occurred since there is no simple way to determine if a DES key has been cracked. An uncertainty about how a thief could obtain a PIN remains, making it debatable whether the plaintiff in the cited court case actually kept the PIN together with the stolen ATM card.

6 Too much secrecy is counterproductive

In this section, we'll discuss why too much secrecy causes the security of a banking system to deteriorate over time, and why a bank's refusal to share technical information is a threat to a customer's right to legal protection during a conflict.

6.1 Court case revisited

The bank's expert witnesses seemed not to know about the described attack scenario, or if they did, they didn't acknowledge this during the trial. Their claim that it was impossible to determine a PIN belonging to a stolen ATM card in less than an hour shows a limited understanding of the level of security provided by the DES-based ATM system. The same lack of understanding was observed when we analyzed the Internet banking systems in Norway. Several banks were completely unaware that they were vulnerable to distributed attacks during 2003 and 2004 [3].

Notice how DES cracking only became a threat as time passed and computer technologies improved. Similarly, the attack scenarios against the Norwegian Internet banking systems were not a problem when the systems were new and had few users, but as the number of users grew the systems became more and more vulnerable.

Because Norwegian banks keep all system information secret and don't allow independent experts to analyze their systems, there is a limited number of security experts with an intimate understanding of the Norwegian banking systems. In the long run, the banks' own experts have a tendency to think alike, especially since they cannot freely discuss the banking systems with outside experts. As a result, they have a propensity to overlook slowly developing system vulnerabilities.

As long as the true level of security is hidden from the Norwegian courts, it's difficult for bank customers to win cases against the banks. The case discussed in this paper demonstrates the problem. The international research community had known for many years that DES was unsafe. Still it was difficult for the plaintiff's lawyer to refute the assertion from the bank's experts because the bank didn't have to provide him with any information about the ATM system.

The plaintiff appealed the verdict to a higher court. According to the judge's ruling after the first trial, the ATM system used DES to verify a PIN when the card was stolen. During the appeal process the defendant's lawyer tried to show that the ATM system had utilized triple DES, and not DES, to verify PINs. The fact that this very important information was not established during the first trial only underscores how important it is to have access to correct technical information.

During the appeal process the plaintiff's lawyer asked for more information, but very little was given. In particular, the bank argued that an encryption algorithm developed to do PIN verifications for MasterCard transactions must be kept secret. According to modern security thinking there is no need to keep cryptographic algorithms secret. In fact, it's considered very bad practice. Unfortunately, economical and personal reasons led the plaintiff to withdraw the case before it could be considered by the higher court.

6.2 "Bankklagenemda" revisited

During the last decade the committee has considered a large number of cases involving stolen ATM cards. In nearly all instances the committee concluded that the card owner must have stored the PIN together with the card. The owner typically didn't agree. This state of affairs continues even today. During 2004 there were around 500 cases involving misused ATM cards.

The committee has all along based its decisions on the assumption that the ATM system has had, and still has, a high degree of security. As evidence they initially referred to a note from 1993 penned by the Norwegian Central Bank. According to the Central Bank it was not possible to crack the PIN using the information on the magnetic strip. Our fictitious attack scenario shows the opposite, it was indeed possible to crack PINs during the~'90s.

Later the committee started to refer to a letter from 2002 written by The Financial Supervisory Authority of Norway, a government agency supervising the Norwegian banks. The letter cites a security report first completed in 1997 and then re-evaluated in late 2001. This report written by representatives from the Norwegian bank community isn't available to the public.

Even though some card owners simply forgot they wrote down the PIN, and others lied, it's unfortunate how "Bankklagenemda" has branded a large number of bank customers as liars without a thorough review of the security in the ATM system. Unlike the earlier secret self-evaluations from the Norwegian

bank community, a new review should be carried out by independent security experts and made available to the public.

Even after the upgrade to triple DES there are indications that the level of security may be lower than advertised by the banks. R. Anderson *et al.* [10] have described numerous physical and logical attacks on security modules, including powerful remote attacks on a module's application programming interface. A clever insider attack determining triple-DES keys is described by Bond and Clayton [9].

6.3 Weak authentications

Authentication is the process of establishing confidence in the truth of some claim [11, Ch. 2]. In particular, an authentication process does not *prove* that a particular individual is who she claims to be. The process can only provide a level of confidence in the claim. Unfortunately, a high level of confidence in an authentication method may well be undeserved. The security of a system suffers when developers have high confidence in an authentication technique that turns out to be vulnerable to attacks by crackers.

We define an authentication method to be *weak* if it's susceptible to a practical attack scenario. An attack scenario is practical if it's reasonable to believe it can be carried out by skilled crackers. The fictitious attack scenario in this paper reveals that the customer authentication in the Norwegian ATM system was weak during the last years DES was used. The attack scenarios reported in [3] show the customer authentication in several Norwegian Internet banking systems to be weak during 2003 and 2004.

7 Toward better development processes

The *architecture* of a banking system defines its conceptual structure and logical organization. The *design* specifies how to create the system and includes the description of communication protocols and cryptographic primitives. Our investigations of the ATM system and Internet banking systems [3] strongly indicate that Norwegian banks didn't perform thorough periodic reviews of the underlying architectural and design assumptions. In fact, according to The Financial Supervisory Authority of Norway ("Kredittilsynet"), no Norwegian bank was instructed to carry out a separate risk analysis of its IT systems before August 2003 [12]. This helps explain why the customer authentications were allowed to become weak over time. Discussions with bank experts have led us to conclude that improved architectural and design processes—better incorporating security and usability aspects—are needed to create more secure banking systems. These processes must produce architectural and design documents understandable not only to the developers of the system, but also to external security experts. The lack of adequate documentation makes it difficult to carry out periodical security reviews of the architecture and design, reducing the likelihood that slowly developing security problems are discovered before they become serious.

In the following, we first outline an alternative to a hierarchical organized development team. We then describe how the team can incorporate security and usability in the architectural and design processes and produce useful documentation. Finally, we discuss how this documentation can also be used during

a conflict between a customer and a bank.

7.1 Development team

A hierarchical organization tends to promote selfish behavior [13]. Leaders wanting to strengthen their own positions in the hierarchy discourage the rank-and-file members from asking critical questions or pointing out mistakes. Groups in an organization where critical thinking is discouraged are unlikely to perform well in the long run. Members only do what the are told, and take little or no personal responsibility for the final outcome of their work. In particular, development teams are unlikely to produce secure systems in environments where critical thinking is discouraged.

A team developing a banking system should be organized as a *heterarchy*—not a hierarchy. Heterarchy means 'multiple rule', a balance of powers rather than the single rule of hierarchy. In a heterarchy, authority is determined by knowledge and function. As the development process progresses, different team members take charge of the process [13].

Members of a successful heterarchy share knowledge and help solve each other's problems. To ensure efficient cooperation, it's important to take the time in the beginning of a project to make sure that all members communicate well and, thus, develop similar mental models. During the project it's vital to build an open culture allowing the team members to utilize all their talents.

In practice, the development team is a small heterarchy in a hierarchical bank organization. Because of the overall hierarchical organization, the team may very well have a formal leader. To ensure that all team members collaborate, the team leader should facilitate free interaction between members and only exercise control during prolonged conflicts.

Open-source collaborative principles [14] capture the most important aspects of a heterarchy. An open dialog between the members is encouraged as well as critical evaluations of ideas. A critical assessment of architectural and design concepts requires the (core) design team to work closely with different kinds of external experts. This is a problem for Norwegian banks since they are used to keep all technical details secret. Unfortunately, secrecy leads to 'groupthink' that discourages creativity and individual responsibility. Bank employees need to learn about openness. See [13, 15] for more information.

7.2 Initial development process

Figure 3 depicts the *initial* stages of a development process producing a system design. (See [16] for a description of the complete process.) Note that we do not not assume a particular software development methodology (such as the Rational Unified Process or the waterfall method). If the development team utilizes an iterative process, then it will cycle through the diagram multiple times.

The first step is to describe the functionality of the new banking system. Often the functionality is described in terms of use cases. Similarly, abuse cases are developed to describe the system's behavior under attack. Together, the use and abuse cases form the basis for the security and usability requirements. The architecture is then developed from these requirements.

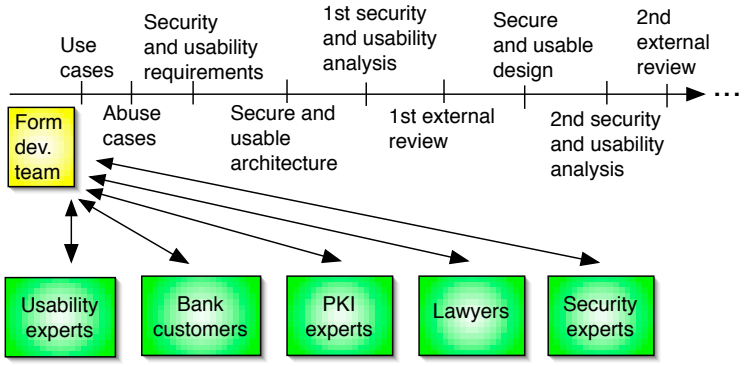

Figure 3: Initial stages of development process for (PKI-based) banking applications where a core development team collaborates with external groups.

7.3 Collaborative architectural process

Norwegian banking systems with weak authentication have been described. To significantly strengthen the authentication in a future banking system, the authentication should be based on a Public-Key Infrastructure (PKI) [17].

We assume that a PKI is needed to fulfill the functionality description since this helps us illustrate the need for collaboration. While it's not difficult to understand the principles of public-key cryptography, it's very hard to design a good PKI. The PKI literature is large and it can be both time consuming and hard for a development team, especially a small team, to obtain a good overview of all critical PKI issues. It's therefore advantageous for the team to collaborate with PKI experts (as indicated in Figure 3). Even if the team decides to buy a PKI solution, it's valuable to have independent PKI experts evaluate the different PKI offers.

A PKI can offer the service of digital signatures, a service analogous to handwritten signatures [17]. A PKI user may deny that a digital signature came from him. This denial is referred to as *repudiation* of the signature. Outside PKI experts are particularly useful if the banking system is to offer digital signatures with a high degree of *non-repudiation*. The goal of the non-repudiation service is to be able to provide credible evidence to a third party that a the signature came from a particular person. This third party is often a judge, jury, or independent arbitrator [18].

Non-repudiation cannot be achieved by technical means alone. To provide a third party with convincing evidence, the development team must collaborate with a team of lawyers to understand what constitutes good evidence according to the relevant local and/or international laws. The architecture should be designed to facilitate the presentation of non-repudiation evidence in court. This evidence must be understandable to people with absolutely no prior knowledge of security. The lawyers can also draw up legally binding contracts to further support the needed degree of non-repudiation.

If people are unable to use a banking system in a secure way, they'll use it in an insecure way. Security must therefore be combined with usability [15]. In fact, usability is king in a banking system. If the usability is poor then many

customers will not use the system at all. Usability goals should therefore be described right after the use cases are developed. The usability goals are particularly important when developing a new banking system for a large number of customers whose computer skills vary widely. Many developers believe there is an inherent trade-off between security and usability. However, this isn't necessarily true when both security and usability are designed into the system from the very start.

The development team should develop the usability goals together with usability experts and bank customers. Important goals in our PKI example is to make certificate generation, installation, and revocation painless for users, and make it easy for users to inspect digital signatures and verify the validity of certificates.

Once the first version of the architecture is finished, the security and usability must be analyzed. Most designers have problems "attacking" their own work. Of course independent outside experts have no such problems. An external review of outside experts (not involved in the development of the system) can therefore be highly advantageous. Note that it's much better to discover serious problems with the architecture at an early stage when it's still possible to remove the problems without incurring large costs.

If a security vulnerability or a usability problem is found, then the architecture must be modified and the altered architecture must be analyzed for new problems. This process must be repeated until no serious problems are found. Any remaining hidden security or usability flaws may lead to an expensive redesign some years into the future.

7.4 Collaborative design process

To see why the design process should also be a collaborative undertaking, we return to our PKI example. The development team should again collaborate with the PKI experts to avoid a design which violates well established PKI principles [17, 18]. The experts make sure that different pairs of public-private keys are used for authentication and digital signatures. If a high degree of non-repudiation is needed, then they suggest a design where a user's signature key is both generated and stored locally on the user's own device.

The usability aspects must be developed further. Unfortunately, users don't understand the difference between a public key and a private key. They also have problems understanding the role of public-key certificates. Finally, users don't understand the connection between the PKI and the goal they are trying to achieve by using a system [15, Ch. 16]. The usability experts can help the development team overcome these issues. Together they can design a banking specific PKI, where most of the PKI functionality is transparent to the users.

Once the initial design is finished, the security and usability must be evaluated. The development team must continue the development until no more serious problems are found.

7.5 Documents

To further encourage all team members to take part in the decision process and make them feel responsible for the complete system, the architectural and design documents should be the property of the development team rather than

individual members. To emphasis that security and usability must be an integral part of the development process, team members should not develop separate security and usability documents.

The architectural and design documents of future banking systems should be made available to the public. This will further encourage good development processes and, thus, reduce the risk of creating systems with serious security flaws.

During a conflict a customer can easily employ outside security experts to carry out an independent security analysis of a banking system. A good design should of course manage to withstand scrutiny from hostile experts.

8 Final remarks

It's difficult to develop banking systems that remain secure and usable over time. A good architecture and design require the (core) development team to collaborate with outside lawyers, security experts, usability specialists, and customers. Such collaborative development is hampered when the company policy dictates that the development team must keep all information about the system secret.

An open collaborative development process can produce banking (and other commercial) systems with better security and usability than in many current closed systems. The publicly available architectural and design documents produced during the development period can improve the customers' legal protection during conflicts involving the systems.

More work is needed to establish open development processes combining security and usability. Ideas from the open source community on how to collaborate [14] and results from security and usability research [15] provide good starting points.

We've only considered the Norwegian DES-based ATM system because our main goal was to analyze a Norwegian court case. However, we believe that much of our analysis applies to other ATM systems based on DES. It's difficult to verify a security analysis of a closed system. While we've had many discussions with banking experts to try to double-check our work, the authors remain solely responsible for the content of this paper.

References

[1] "Verdict from Trondheim Tingrett, case number 04-016794TVI-TRON," (in Norwegian).

[2] NIST, "Data Encryption Standard (DES)." [Online]. Available: http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf

[3] K. J. Hole, V. Moen, and T. Tjøstheim, "Case study: Online banking security." *IEEE Security & Privacy*, vol. 4, no. 2, pp. 14–20, 2006.

[4] R. Anderson, M. Bond, J. Clulow, and S. Skorobogatov, "Cryptograpic processors—a survey," University of Cambridge, Tech. Rep. 641, 2005. [Online]. Available: http://www.cl.cam.ac.uk/TechReports/UCAM-CL-TR-641.pdf

[5] "ISO 9564–1 Banking—Personal Identification Number (PIN) management and security—part 1: Basic principles and requirements for online PIN handling in ATM and POS systems, 1st ed." 1991.

[6] "ISO 9564–2 Banking—Personal Identification Number (PIN) management and security—part 2: Approved algorithm(s) for PIN encipherment," 1991.

[7] EFF, "Cracking DES," 1998. [Online]. Available: http://www.eff.org/Privacy/Crypto/Crypto_misc/DESCracker/

[8] "RSA DES challenge III," 1999. [Online]. Available: http://www.eff.org/Privacy/Crypto/Crypto_misc/DESCracker/HTML/19990119_deschallenge3.html

[9] M. Bond and R. Clayton, 2001. [Online]. Available: http://www.cl.cam.ac.uk/~rnc1/descrack/

[10] R. Anderson, "Why cryptosystems fail, from communications of the ACM, November, 1994," in *William Stallings, Practical Cryptography for Data Internetworks.* IEEE Computer Society Press, 1996. [Online]. Available: http://citeseer.ist.psu.edu/anderson94why.html

[11] N. R. Council, *Who Goes There?* National Academies Press, 2003.

[12] The Financial Supervisory Authority of Norway, "Risk and vulnerability analysis 2003," 2003, (in Norwegian).

[13] G. Fairtlough, *The Three Ways of Getting Things Done.* Triarchy Press, 2005.

[14] C. DiBona, D. Cooper, and M. Stone, Eds., *Open Sources 2.0.* O'Reilly, 2006.

[15] L. F. Cranor and S. Garfinkel, Eds., *Security and Usability.* O'Reilly, 2005.

[16] G. McGraw, "Software security," *IEEE Security & Privacy*, vol. 2, no. 2, pp. 80–83, 2004.

[17] C. Adams and S. Lloyd, *Understanding PKI*, 2nd ed. Addison-Wesley, 2003.

[18] W. Ford and M. S. Baum, *Secure Electronic Commerce*, 2nd ed. Prentice Hall, 2001.