How does PINsentry™ work?

Joss Wright

February 20, 2008

1 Introduction

This text begins by introducing Barclays PINsentry™, its motives and goals. It then discusses the security of the system and the method that it uses to authenticate with the bank. Section 4 gives a very quick introduction to the methods and techniques of cryptography, and can largely be skipped. Section 6 gives a quick overview of how PINsentry™ does what it does.

2 PINsentry™

Barclays PINsentry™ is a pocket-sized device designed to add an extra layer of security for Barclays Online Banking customers. Barclays’ introduction of the device is aimed at reducing the need for customers to remember a password in order to access their services. Such passwords are easily forgotten and consequently are often written down in an insecure location.

Beyond this, the use of a password to access services relies on what is known as “one-factor authentication” — it relies on “something that you know”. As such, anyone who knows the password can access the services.

Barclays PINsentry™, which is part of a scheme designed by APACS1, is designed to extend online banking services to make use of “two-factor authentication”. By requiring use of the customer’s physical card to access services, the system relies not only on “something you know” but also “something that you have”. In order for a third party to access the services, they must not only gain access to the password, they must also have the card itself.

1The Association for Payment Clearing Services — the UK trade association for payments and for those institutions that deliver payment services to customers.
3 Is it secure?

3.1 Short Answer

Yes.

3.2 Longer Answer

... as far as anyone knows.

The Barclays PINsentry™ system is an improvement, in terms of security, over the previous online banking schemes that require only a password. This security has been gained at the cost of decreased usability, by requiring customers to have access to the card reader device whenever they wish to access services.²

The cryptographic algorithm (the series of steps and mathematical operations) used by the PINsentry™ device in order to authenticate itself with the bank is a secret. This secrecy is maintained by the banks to prevent analysis of the method employed.³

Despite not knowing the method used by the card reader, the system has been designed with high levels of security in mind. It is unlikely that the system will be broken at a mathematical level, and even if it were then this would not be a concern to the average customer of Barclays.

In terms of security, the system is perfectly acceptable against the attackers that would be expected.

4 Basics of Cryptography

In order to authenticate with the bank, the PINsentry device and card make use of several cryptographic operations. Without going into too much detail, the basic of cryptography will be explored here.

Cryptography is the science of making information unreadable to someone who doesn’t know a particular secret. This secret is usually referred to as a “key”.

Computer represent all information as streams of numbers. When messages are sent from one computer to another, they are effectively transmit-

²A reduced version of online banking is available for customers without their card reader device, but this is restricted to viewing the details of payments rather than initiating new transactions.

³This is against the advice of the international cryptographic research community. It is commonly held that algorithms used in such devices should be published in order to allow researchers to test their security. Such schemes should be able to function even if the method that they use is well-known. By keeping the method secret, the banks are partially relying on “security through obscurity”. This approach allows mistakes in the algorithm to remain undiscovered, possibly until it is too late for the scheme to be recalled.
4.1 Shared Key Cryptography

In many situations where we wish to use cryptography, we assume that there are two parties that want to send a message to each other. In cryptography, we usually refer to our two communicating partners as “Alice” and “Bob”.

For a simple example, imagine that Alice and Bob have met in person at some time in the past. Between them, they have decided on a “secret” number: 78. This is their shared key.

We can propose a situation in which Alice and Bob want to arrange a secret meeting at one of their two favourite pubs. As it’s a secret meeting, they don’t want anyone else to know which pub they’re going to choose. In order to keep it secret, they have decided that Alice will choose, and will send Bob a secret message when she knows where she wants to meet. They have arranged that if Alice sends Bob a 1 then they will meet in the Red Lion. If Alice sends Bob a 0 then they will meet in the White Horse.

Alice decides on the Red Lion. She sends Bob the number 1 plus their secret number 78, making 79. When Bob receives the message, he simply subtracts their secret number to reveal a 1. If (Malicious) Mallory intercepts the message, he doesn’t know the secret number to take away and so doesn’t know if a 1 or a 0 was sent. Alice and Bob are safe.

4.2 Public Key Cryptography

A much more complicated example is where rather than having a single (or shared) key, Alice and Bob each have their own pair of keys.

In the previous example, one secret key was used to make the message unreadable (to encrypt it) and the same secret was used to make it readable again (to decrypt it). Such a system causes difficulties with how to get both parties to know the secret in order to send messages.

The solution is for each person to have two keys rather than one. Of each person’s two keys, one can be used only to encrypt messages. The other can be used only to decrypt messages. If Alice wishes to send a message to Bob, he will quite happily send her the public key that can only encrypt messages to him. This doesn’t need to be secret, because anyone who learns it will only be able to make secret messages that go to Bob. Only Bob can read these messages by using the other key, or private key, that he keeps secret.

The maths for this are complicated, but rely on “one-way functions”. These are mathematical operations that are easy to perform in one direction, but very hard to do in the other direction.

A good analogy is of a Chinese to English dictionary (without an English to Chinese section!): if one is told a Chinese word, it is a simple job of two or
three minutes to find the corresponding English word using the dictionary. If one is told an English word, it could take days to find the corresponding Chinese word in the dictionary.

This idea can be used in reverse to prove that a user is who they say they are. The secret key is normally used to decrypt messages encrypted with the public key, but can also be used to digitally “sign” a message. In this situation, it is possible to take a message and uniquely “mark” it with the private key. Anyone with the corresponding public key can verify that the private key was used to make that mark. (The details of this are complicated.) As the private key is known only to one person, this is equivalent to putting a signature on something, but is vastly harder to forge without the secret, private key.

4.2.1 A note on “hard”

The previous section declared that some functions are “hard” to run in reverse. The example given was a Chinese dictionary. It would take days for a person to find the corresponding Chinese entry in a Chinese dictionary if they only knew the English. The quickest way to do this would be simply to start at the beginning and try every word until the answer is found. According to the laws of probability one would expect, on average, to try half of the entries before finding the answer.

Computers are, of course, much faster than humans at finding answers. An electronic Chinese dictionary would give the answer in a few seconds. Luckily, computers are also faster at asking harder questions.

The numbers that are used in cryptography tend to be in the realm of 1024 bits. This means that, on average, a computer would have to try about:

\[161521746670640296426473658228859984306663144318152681524054709-\]
\[07824573659036629724837729808265693930673286493230336261991466-\]
\[93859669107311296862610792148904239628873374506302653492009810-\]
\[62643758258708945395941375496004739918498276676334238241465498-\]
\[030036586063929902368192004233172032080188726965600617167\]

combinations in order to find the answer. A recent estimate showed that if the entire surface of the earth was covered with a layer of top-of-the-range computers one mile deep it would take one million million million million years to try 1% of the available keys.

---

4 Assuming that one doesn’t speak Chinese.
5 A physical signature, on the other hand, can be forged by observing sufficient examples of signed documents. This is not possible with digital signatures.
6 A bit is a 1 or a 0, and is used at a basic hardware level to represent numbers in computers.
Unfortunately, public-key cryptography is relatively intensive to use. As such, public key cryptography is typically used at the start of a set of communications in order to allow the two parties to negotiate a shared key. This key is then used for future messages, and is changed relatively often in case a third party somehow learns it.

4.3 Hashing

A fundamental operation in cryptography is what is known as *hashing*. Hashing can be thought of as “encrypting without keys”. Hashing takes a number and produce another number that is uniquely generated by that first number. In other words, the hash of a given number is always the same number, but a different number will produce a different value. It is therefore easy to check if a hash matches a given number — one simply hashes the number.

The critical part of hashing, however, is that the hash of a number cannot be used to work out what the original number was. In the Chinese dictionary analogy above, the hash of an English word is the Chinese word. If you know the Chinese word (the hash) then you can easily check that the original (the English) matches is. Hashing is used to check passwords, and solves the problem of storing “secret” passwords.

When a computer asks for a password, one might assume that it takes the password and compares it to one that has been stored somewhere. This would certainly work, but would allow anyone who could access that computer to read all the passwords.

The answer is to store only the *hashes* of the passwords. When the computer reads a password, it uses that to make a hash. This hash is then compared against a stored list of password *hashes*. If an attacker reads the list of hashes, they cannot learn the passwords.

As an aside, this is why online services will always send a new password rather than supplying the old one — they don’t know the old one! If a service can supply the old password then they are using an insecure method!

5 Encrypted Internet (Web) Connections

The above techniques are used when users connect to a “secure” website, such as all those used by banks in their webpages. The public key method is used firstly to generate a shared key, and this is used for all future transactions.

When a customer connects to a bank, the web browser automatically negotiates an encryption key with the bank. All future traffic is wrapped up in this encryption. A third party who can sit between the customer and the bank can read every single bit of information that is transmitted, but
without the encryption key to decrypt the information this appears as a long string of gibberish.

6 Specifics of PINsentry™

This is the important bit!

So, how does PINsentry™ work? It’s actually rather boring.

The chip on modern credit and debit cards contains a private key unique to that card. When the card is slotted into a reader, such as in a shop, the card immediately asks for the PIN number. The chip on the card compares the hash of this number against a stored hash of the correct PIN. If this is correct then the card “signs” the details of the current transaction with the private key. This is sent to the bank, and the bank knows that the user authorised the transaction: only they know the PIN, and only the PIN can unlock the secret key.

(As an aside, the chip is set to wipe or disable itself if it receives too many incorrect attempts. This stops someone from just trying all PIN combinations.)

The PINsentry™ system uses a very simple extension of this. The bank has the public key corresponding to each user’s card stored, and can therefore check that messages signed with the private key are valid. When the user tries to log on, the card reader asks for the PIN to activate the card.

The card reader contains a clock. When the “Identify” button is pressed, it uses the private key on the card and the current date and time to produce an eight-digit number. The bank knows the current time and the public key, so it can check that the “Identify” key was pressed at that time. When the eight-digit number is sent, the bank can check that it is:

- You, because you know the PIN.
- Your card, because you have made use of the private key stored on the card.
- You right now, because the number that you type relates to the current time.

The number that the card reader produces is valid for, apparently forty seconds from the time of creation. Creating two numbers one second after the other will produce different numbers, but each will be valid for a further forty seconds.

---

7 Described here is a simplified version. The true version is a fair bit more complicated than this, but this is the essence of it.
8 This is untested.
When the PINsentry™ device is first received, there is an activation number that identifies the card with the device. This is likely to use a similar public key encryption system to allow the system to authenticate itself, and adds an extra layer of security.

7 Conclusions

The Barclays PINsentry™ is a two-factor authentication method that makes use of advanced industry-standard cryptography in order to authenticate a user to the bank with high degree of confidence. The card reader device uses the private key stored on the card in combination with the user’s PIN number and the current date to produce a unique eight-digit number every time a connection to the bank is required.

In terms of usability the system leaves something to be desired, as the user must carry the PINsentry™ with them whenever they may wish to access their account, but such a security tradeoff is acceptable when the information is highly important.

In terms of security, the system relies on an attacker learning the PIN, gaining the card, and gaining the appropriate card-reader. Cloning cards is difficult due to the nature of the chip. The most likely avenue for attack is a fake card reader in a shop that stores the user’s PIN number, although making use of the PIN would still require stealing the card itself. A more likely approach would be a thief “shoulder-surfing” the PIN number by reading it over the user’s shoulder in the shop as they make a transaction. The thief would then steal the user’s wallet. (To access online banking, the thief would additionally need to know the user’s online banking account number.)

As a system for allowing home banking online, the PINsentry™ is a convenient system that leans towards a high level of security. The possibility of a user’s account being compromised via this route is extremely low.