Fingerprinting Passports

Henning Richter[1], Wojciech Mostowski[2] ⋆, and Erik Poll[2]

[1] Lausitz University of Applied Sciences, Senftenberg, Germany
henning-richter@gmx.de
[2] Radboud University, Nijmegen, The Netherlands
{woj,erikpoll}@cs.ru.nl

Abstract. Passports issued nowadays have an embedded RFID chip that carries digitally signed biometric information. Access to this chip is wireless, which introduces a security risk, in that an attacker could access a person's passport without the owner knowing. While there are measures in place to prevent unauthorised access to the data in the passport, we show that it is easy to remotely detect the presence of a passport and determine its nationality. Although all passports implement the same international standard, experiments with passports from ten different countries show that characteristics of each implementation provide a fingerprint that is unique to passports of a particular country.

1 Introduction

Most passports issued nowadays are e-passports, and have an embedded RFID chip – effectively a contactless smartcard – that carries digitally signed biometric information.

To prevent wireless reading of the passport content without the owner's consent, passports can use a mechanism called *Basic Access Control* (BAC): to access the smartcard one must visually read some information printed in the passport. Subsequent communication between passport and reader is then encrypted to prevent eavesdropping. All EU passports implement BAC.

Weaknesses in the encryption mechanism used in BAC have already been reported [2, 4]: for passports from several countries brute force attacks – which exhaustively try all keys – are feasible. Root cause of this problem is that passport serial numbers are handed out in sequence, meaning that there is not enough entropy in the keys to prevent brute force attacks.

We report on a different issue: the possibility to detect passports and determine their nationality. This turns out to be surprisingly easy to do. Although passports implement the same standard, there are differences that can be detected, especially by sending ill-formed requests, *before* Basic Access Control takes places. In this way we were able to distinguish all passports that we tested. With quite a few foreigners working or studying in our university department, we managed to test passports from 10 different countries: Australia, Belgium, France, Germany, Greece, Italy, the Netherlands, Poland, Spain, and Sweden.

While not an immediate security threat to the passport itself, it could be a concern to the passport holder: this functionality is clearly useful for passport thieves. It strengthens the case for metal shielding in the passport to prevent any communication with the RFID smartcard when the passport is closed (as used in US passport, where it is used instead of Basic Access Control)[3]. More generally, it demonstrates

⋆ Financially supported by Sentinels, the Dutch research programme in computer security.

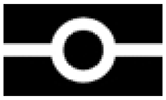
[3] Erratum: US passports now have BAC as well as shielding, according to http://travel.state.gov/passport/eppt/eppt_2788.html

the problems associated with making communication wireless, esp. with something as sensitive as an identification document.

The outline of the rest of this paper is as follows. Section 2 explains the basics of the e-passport, and briefly discusses some of the earlier problems found, Section 3 describes our attempts to fingerprint e-passports from different countries, and Section 4 summarises our conclusions.

2 e-Passports

An e-passport can be recognised by the logo



which is typically printed on the passport cover to indicate the presence of an RFID chip. An e-passport contains an ISO14443 RFID tag, which can also be called a contactless smartcard. ISO14443 is the standard for so-called *proximity* RFID tags, with a designed operating range of 1–10 cm. ISO14443 defines the low level physical communication layer. The higher level logical communication, at the level of bytes sent and received, conforms to the older ISO7816 standard, which was originally developed for contact smartcards. The international standards for e-passports are provided by guidelines [7, 6] of the International Civil Aviation Authority (ICAO), a United Nations body. In particular, these define a set of commands (in ISO7816 format) that the passport can receive and the corresponding responses.

Currently the chips in most e-passports store the personal details of the passport holder and a photo (typically in JPEG2000 format), all digitally signed to prevent fake passports, but it is foreseen that more biometric data, in particular fingerprints, will be added.

The fact that e-passports are wireless introduces two new security risks:

active scanning attacks: an attacker communicates with the passport without the owner's consent, by bringing a reader in close proximity to the passport, e.g. when the passport is in a coat pocket or a handbag.

passive eavesdropping attacks: an attacker eavesdrops on the communication when the passport communicates with a legitimate reader with the owner's consent, for instance at passport control at an airport.

Both attacks could potentially reveal sensitive data, such as name and passport number, but Basic Access Control (BAC), discussed below, is a countermeasure to prevent this. Active attacks can also reveal the presence of an e-passport, which in itself is interesting to a potential thief, and the nationality of that passport; this is the topic of this paper. Active attacks could be used in so-called relay attacks, where communication to the e-passport is relayed 'live' to some device, using any means of communication, which can then simulate the passport, say at border control many miles away.

The decision to make the e-passport contactless is a typical design decision favouring convenience over security. Using a contact smartcard as passport would remove the risks above. The traditional forms for contact smartcards – the credit-card format or the smaller format used for mobile phone SIMs – are of course not immediately compatible with that of traditional passports.[4]

[4] Contactless cards do have some security advantages over contact cards, namely when it comes to availability: contacts on smartcards and readers are subject to wear and tear, so they might fail and require maintenance. This also makes readers for contact cards more susceptible to sabotage than readers for contactless cards.

ISO14443 specifies a maximum distance of 10 cm between tag and reader, but typical commercial readers operate only at a maximum distance of 1–2 cm. Still, bigger distances than 10 cm are possible. Eavesdropping on e-passports has been shown to be possible from 9 meters [13]. The maximum distance for an active scan is much smaller than for eavesdropping, because a strong magnetic field is needed to power up the passport. [11] predicts that activating a tag is possible from 50 cm, and [12] built a $100 device achieving 25 cm, which the authors claim could be extended to 35 cm. [3] describes experiments with different antenna sizes and amplifications where a card was activated at 27 cm. [1] claims a reading range of 60 cm using a single antenna and more than one meter using a dual antenna, but this does not appear to be backed up by actual experiments.

Basic Access Control (BAC) One defence mechanism that has been foreseen against eavesdropping attacks is Basic Access Control (BAC). Communication between the passport and reader is encrypted, with an encryption key that is written on one of the passport pages. The idea is that the communication is only after opening the passport and (optically) reading this key, which presumably indicates consent by the passport holder. The key is written in the so-called Machine Readable Zone (MRZ) at the bottom of one of the passport pages; machine readable here means is can be read by OCR (Optical Character Recognition). A passport with such an OCR-readable MRZ is officially called a Machine Readable Travel Document (MRTD). All EU passports support BAC, though the first generation of Belgian passports did not [2].

Attacks on this mechanism have been reported for e-passports of several countries. As described for instance in [4], a flaw in BAC is that the key written in the MRZ, which includes name, passport number and expiry date, may not contain enough entropy to prevent brute force attacks, especially since most countries issue passport numbers in sequence, which means they are predictable and strongly correlated to the expiry date. Feasibility of a brute force attack on BAC was first demonstrated for the Dutch e-passport by Marc Witteman, and later also for English, Belgian and German e-passports.

Anti-collision The first phase of the ISO14443 protocol is an anti-collision protocol, which is needed because more than one RFID tag may be within range of a reader. As part of this anti-collision protocol, each RFID tag sends, upon activation, some number (the UID), so that the reader can detect the presence of multiple tags and select the right one to talk to. This anti-collision protocol has been recognised as a potential security weakness: some e-passports have been reported to always transmit the same number in the anti-collision phase, which is unique to a single passport, so that it can be used to detect individual passports.

This issue is discussed in [8], which leaves it up to the passport issuer to choose for a random UID or a fixed, unique one: "Some issuers of passports wish to implement a unique number for security reasons or any other reason. Other issuers give greater preference to concerns about data privacy and the possibility to trace persons due to fixed numbers." [8, p.21]

For more discussion on the security of e-passports see [4, 10, 2]. In the remainder of this paper, we will focus on the particular attack we investigated, namely if, prior to BAC, the nationality of the passport can be determined by its characteristics at the logical level, i.e. the responses of the passport – in bytes – to specific commands.

3 Fingerprinting Implementations at Logical Level

At the logical level, the e-passport and reader communicate by exchanging so-called *Application Protocol Data Units (APDUs)*, which are just sequences of bytes in a particular format. Communication between reader and e-passport is in Master-Slave mode: it is always the reader that takes the initiative for communication, by sending a so-called command APDU, and the e-passport then replies with a response APDU. The format of these APDUs is specified in ISO7816 (ISO7816-4, to be precise).

A command APDU consists of at least 4 bytes: a class byte CLA, an instruction byte INS, two parameters P1 and P2, a length Lc of the optional additional data, and an optional field Le indicating the expected length of the response:

Mandatory header				**Optional Body**		
CLA	INS	P1	P2	Lc	data	Le

A response APDU consists of at least a two bytes status word – bytes SW1 and SW2 – preceded by an optional response:

Optional body	**Status word**	
data	SW1	SW2

ISO7816 defines many standard values for the instruction byte (INS) and the status word (SW1 SW2), which are in turn used in the ICAO specifications.

According to the ICAO specifications [7, p.22f] all passports must at least support the instructions

- SELECT FILE (A4)
- READ BINARY (B0)

Additional optional instructions (needed for Basic Access Control and other optional features) are

- EXTERNAL AUTHENTICATE (82)
- INTERNAL AUTHENTICATE (88)
- GET CHALLENGE (84)

All these instructions are defined in ISO7816. To distinguish the passports, in addition to the 5 instructions above, we used two other instructions from ISO7816 that are not mentioned in the ICAO specifications, namely

- REHABILITATE CHV (44)
- READ BINARY (B1)

(CHV stands for Card Holder Verification, which refers to the use of PIN codes. PIN codes are not used anywhere on e-passports. Still, different passports reply with different error messages when sent the REHABILITATE CHV instruction.)

Responses of the various passports to these commands included the following standard status words defined in ISO7816:

- No Error (9000)
- Unknown (6F00)
- CLA Not Supported (6E00)
- Instruction Not Supported (6D00)
- Incorrect P1P2 (6A86)
- Command Not Allowed (6986)
- Conditions Not Satisfied (6985)
- Security Status Not Satisfied (6982)
- Wrong Length (6700)

	Commands						
	44	**82**	**84**	**88**	**A4**	**B0**	**B1**
	Rehab. CHV	Ext. Auth.	Get Chall.	Int. Auth.	Select File	Read Binary	Read Binary
Australian	6982	6985	6700	6700	9000	6700	6700
Belgian	—	6E00	—	6700	6A86	6986	6700
Dutch	—	6700	6700	6982	6A86	6982	6982
French	6982	6F00	6F00	6F00	6F00	6F00	6F00
German	—	6700	6700	—	6700	6700	—
Greek	6982	63C0	6700	6982	9000	6986	6700
Italian	—	6700	—	—	—	—	—
Polish	6982	6700	6700	6700	9000	6700	—
Swedish	6982	6700	6700	—	9000	6700	—
Spanish	—	6700	6700	—	6700	6700	—

Table 1. Responses of the various passports to our test commands. A dash (—) means the card responds with 'Instruction not supported' (6D00)

- Counter reached zero (63C0)

Table 1 gives an overview of which status word is given as reply to each command by a passport of a given country[5]. The combinations of these commands and responses provide a unique fingerprint for each of the 10 nationalities, with the exception of the German and Spanish passports, which are identical. However, additional testing with different parameters for instructions did reveal a difference: when giving the EXTERNAL AUTHENTICATE command with the required length of the command data (the Lc byte) of 40 bytes, the German passport answers with the status word "Conditions Not Satisfied" whereas the Spanish passport answers with 6300, meaning "Verification Failed".

With this, the combinations of commands and responses provide a unique fingerprint for each of the 10 nationalities. Using at most 4 commands we can distinguish the 10 nationalities as follows:

- the response to command 82 will identify Australian, Belgian, French, and Greek passports;
- otherwise, the response to command A4 will identify Dutch and Italian passports;
- otherwise, the response to command 88 will identify Polish and Swedish passports;
- finally, the response to command 82 with Lc parameter 40 will identify German and Spanish passports.

The methodology in coming up with these commands as the ones to use to fingerprint passports was simple. We started with a basic test application which logged the responses of passports to all possible 256 instruction bytes (and some combinations of additional parameters). The resulting log files were compared to see which commands revealed the most interesting differences and the other commands were removed from the test suite, in the end leaving us with a test suite containing just the 7 instructions in Table 1.

It is not so surprising that we manage to obtain unique fingerprints per country, given the freedom that the standard leaves. Just taking the 9 possible responses

[5] The table shows that some passports reply 'Instruction Not Supported' to the two instructions that must be supported by all passports, namely SELECT FILE and READ BINARY; only after Basic Access Control has been completed will these passports respond to these instructions as required.

for the 7 commands we considered, it is in theory possible to come up with 9^7 distinguishable implementations that all meet the standard, without even considering the possibilities for different parameters for the same command. Of course it will be highly unlikely that any passport implementations will give 7 different responses to these 7 different instructions, but still, the chance of two independently written passport implementations having the same fingerprint will be small. The chance that some other RFID application has the same fingerprint as one of the passports will be even smaller.

The ICAO technical report [7, p.21] states: "A MRTD chip that supports Basic Access Control MUST respond to unauthenticated read attempts (including selection of (protected) field in the LDS) with 'Security Status not satisfied' (6982)". One could argue that all the instructions in Table 1 are 'unauthenticated read attempts', and that the passports should respond with 'Security Status not satisfied' to all of them.

It is a well-known and recurring problem that error messages can leak useful information for attackers. In fact, it is listed as one of the deadly sins of software security (Sin 13) in [5]. The standard example is a website that displays SQL error messages when subjected to strange input, which can provide useful information for crafting SQL injection attacks.

3.1 Other Ways of Fingerprinting Passports

In Section 2 we already discussed the possibility of fingerprinting passports based on the numbers used in the anti-collision phase of the ISO14443 protocol. In addition to this, and the method described above, there may be other ways of fingerprinting passports, as explained below.

Using FCI Data An alternative way to try to distinguish passports is to use the File Control Information (FCI) data. Normally upon selection of the applet the card will just return the success status word 9000. Some of the passports, however, will upon selection also return (when prompted) the File Control Information (FCI) data in the Tag Length Value (TLV) format. The TLV format is essentially a hierarchical tree-like data structure. A single TLV structure can either be a leaf, in which case it contains a value associated with a given tag, or a node, where the value of the tag are further TLVs.

The file information data can be marked with three different tags [9, Section 5.3.3]: File Control Parameter (FCP) tag 62, File Management Data (FMD) tag 64, or File Control Information (FCP+FMD) tag 6F. Among other things, this FCI data can contain manufacturer specific proprietary data (tags 85/A5 [9, Section 5.3.3]). Since selection of the applet happens before BAC is performed, this proprietary data (or any other differences in the FCI data) can give us hints about the source of the passport.

For example, the German and the Spanish passports do not give any FCI data, the FCI tag is empty: 6F 00. Polish and Swedish passports give almost the same data, but there is still a difference:

```
Polish:                  Swedish:
6F 24                    6F 25
  81 02                    81 02
    0F 80                    1F 80
  82 01                    82 01
    38                       38
  83 02                    83 02
    3F 00                    3F 00
```

```
84 0C                     84 0C
  F0 43 4F 53 34 36         F0 43 4F 53 34 36
  31 72 65 76 41 31         31 72 65 76 41 31
86 00                     86 00
8A 01                     8A 01
  05                        05
A5 04                     A5 05
  C1 00                     C1 00
  C2 00                     C2 01 81
```

The contents of tags 81 (“number of bytes in data file”) and A5 (“proprietary information”) reveal differences. The FCI data for the French passport differs considerably too. For example, the tag representing life cycle status (tag 8A) gives value 03. Then for the Greek passport this value is 07. Moreover, the Greek passport chooses to mark the FCI data with tag 62 instead, i.e. File Control Parameters (FCP) tag.

Selecting the default applet on the passport can also give interesting results. While almost all passports select the MRTD application by default, the Dutch passport selects the Global Platform Security domain (AID A000000003000000). This behaviour, being a differentiator by itself, also gives us the hint that this passport is probably implemented on a Java Card.

ATSs Upon activation, an ISO14443 RFID tag replies with an ATS (Answer To Select), a fixed sequence of bytes for a given tag. Some of these bytes describe protocol parameters, e.g. the maximum framesize and the data rates for sending and receiving the card supports, and some can be freely used, e.g. to give the version number of the card’s operating system.

This ATS could be used to identify e-passports, though it is by no means guaranteed that all passports of a country have the same ATS, or that this ATS is not also used on passports from a different country or other RFID applications than e-passports. We have not looked into the possibility of using ATSs to fingerprint passports. Note that the ATS identifies the smartcard hardware and operating systems, whereas our technique for fingerprinting based on APDU responses identifies the code of the implementation. If a government upgrades passports to use different hardware this will change the ATS, but it will not affect the fingerprints we found based on APDU responses, unless the application code is rewritten.

Side-channels Even if passports were to behave identically at the logical level, there is the possibility of detecting differences at the lower, physical level, e.g. by looking at response times or power consumption profiles. Power consumption and timing are a well-known side-channels that can leak information on smartcards. However, the way in which we fingerprint passports, at the logical level, is much simpler and more precise. An advantage of detecting differences at the physical level might be that the country can also be deduced in a passive eavesdropping attack, despite the encryption of the communication provided by BAC, whereas we rely on an active attack.

4 Conclusions

We have described a new way to remotely detect the presence of an e-passport and to determine its nationality. Results in the literature prove the attack is possible at a distance of 25 cm [12, 3] and suggest it may be possible from slightly larger distances.

The detection is done at the logical level, i.e. by looking at the bytes that an e-passport sends as reply in response to some carefully chosen commands from the reader. The attack is able to distinguish e-passports of all countries that we managed to get hold of: Australia, Belgium, France, Germany, Greece, Italy, the Netherlands, Poland, Spain, and Sweden. The ease with which we were able to find differences between these passports, and the huge space of potential fingerprints, suggests that discriminating between additional countries should not pose any problems, unless there are countries that have an identical implementation. This might happen if different countries obtain the implementation from the same supplier, or if several countries reused a common open source implementation, such as the open source Java Card implementation of the passport we provide at `http://jmrtd.sourceforge.net`. But for the countries above it is clear that each country developed its own implementation from scratch.

Given that we can remotely detect the presence of a passport of a particular country, how could this functionality be abused? One abuse case that has been suggested is a passport bomb, designed to go off if someone with a passport of a certain nationality comes close.[6] One could even send such a bomb by post, say to an embassy. A less spectacular, but possibly more realistic, use of this functionality would by passport thieves, who can remotely check if someone is carrying a passport and if it is of a 'suitable' nationality, before they decide to rob them.

In principle, the fingerprint of an implementation can be used as a hidden channel for the passport to communicate some information about the passport holder on purpose. For instance, a government could issue passports with a slightly different fingerprint to say people with a criminal record or suspected of terrorism. But then there are many ways in which a government can include additional hidden information or 'Trojan' functionality in the passports it issues, so this is a more general issue.

A countermeasure against scanning attacks which try to access the passport without the owner's knowledge or consent would be the inclusion of a Faraday cage in the passport: aluminium foil in the cover would prevent access to the chip if the passport is closed. US passports already include this. (US passports do not implement BAC, which makes the need for this countermeasure even stronger.) ICAO considered the use of such a physical protection measure, but because it does not prevent against eavesdropping attacks, BAC was considered a better option [7, Section 2.4].

It would have been good to include this as an extra defence mechanism right from the start, sticking to the well-known security principle of defence-in-depth. Applying this principle would have been the wise thing to do when introducing the relatively new technology of RFID in such a sensitive application as the e-passport. It is also wise in the light of the history of the ongoing arms race between attackers and defenders in the field of (contactless) smartcards in the past decades, which suggests we can expect some new and as yet unforeseen attacks in the future.

Another countermeasure could be taken at the level of the ICAO specifications. For commands not listed in the ICAO specs, the specs do not prescribe any required response, which leaves the room for so much diversity among implementations. If the ICAO specs would require a standard response for commands not listed in the specs and all malformed requests, this would make distinguishing passport nationalities much harder or even impossible, even though it will not affect the possibility to detect the presence of a passport.

[6] There is even a YouTube movie demonstrating such a device, at `http://www.youtube.com/watch?v=-XXaqraF7pI`.

Acknowledgements

We would like to thank the students and employees at the Radboud University who let us use their passports in our experiments, and Tjark Weber and Henryk Plötz for comments on draft versions of this paper.

The work of Wojciech Mostowski is supported by Sentinels, the Dutch research programme in computer security, financed by the Technology Foundation STW, the Netherlands Organisation for Scientific Research (NWO), and the Dutch Ministry of Economic Affairs.

References

1. Radio frequency identification systems HF antenna design notes. Technical Report 11-08-26-003, Texas Instruments, September 2003.
2. Gildas Avoine, Kassem Kalach, and Jean-Jacques Quisquater. epassport: Securing international contacts with contactless chips. In *Financial Cryptography 2008*, LNCS. Springer-Verlag, 2008. To appear.
3. Gerhard P. Hancke. Practical attacks on proximity identification systems. In *IEEE Symposium on Security and Privacy (S&P'06)*. IEEE, 2006.
4. Jaap-Henk Hoepman, Engelbert Hubbers, Bart Jacobs, Martijn Oostdijk, and Ronny Wichers Schreur. Crossing borders: Security and privacy issues of the European e-Passport. In *Proc. IWSEC 2006: Advances in Information and Computer Security*, number 4266 in LNCS, pages 152–167. Springer, 2006.
5. Michael Howard, David LeBlanc, and John Viega. *19 Deadly Sins of Software Security.* McGraw-Hill, 2005.
6. Development of a logical data structure - LDS for optional capacity expansion technologies, revision 1.7. Technical report, ICAO, May 2004. Available from `http://mrtd.icao.int/images/stories/Doc/ePassports/Logical%AC%AC_Data_S%tructure(LDS)_version1.7.pdf`.
7. PKI for machine readable travel documents offering ICC read-only access, version 1.1. Technical report, ICAO, Oct 2004. Available from `http://www.icao.int/mrtd/download/documents/TR-PKI%20mrtds%20ICC%20read%-only%20access%20v1_1.pdf`.
8. Supplement to doc 9303, version 6 (final). Technical report, ICAO, Sept 2007. Available from `http://mrtd.icao.int/component/option,com_remository/Itemid,256/func,st%artdown/id,26/`.
9. ISO 7816. ISO/IEC 7816 Identification cards – Integrated circuit(s) cards, Part 4: Organization, security and commands for interchange. Technical report, ISO JTC 1/SC 17, 2005.
10. Ari Juels, David Molnar, and David Wagner. Security issues in e-passports. In *SecureComm 2005*. IEEE, 2005.
11. Ziv Kfir and Avishai Wool. Picking virtual pockets using relay attacks on contactless smartcard systems. In *First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM'05)*. IEEE, 2005.
12. Ilan Kirschenbaum and Avishai Wool. How to build a low-cost, extended-range RFID skimmer. In *Proceedings of the 15th USENIX Security Symposium*, pages 43–57. usenix, 2006.
13. Junko Yoshida. Tests reveal e-passport security flaw. *EE Times*, August 2004.