A Traceability Attack Against e-Passports

Tom Chothia* and Vitaliy Smirnov

School of Computer Science, University of Birmingham, Birmingham, UK

Abstract. Since 2004, many nations have started issuing "e-passports" containing an RFID tag that, when powered, broadcasts information. It is claimed that these passports are more secure and that our data will be protected from any possible unauthorised attempts to read it. In this paper we show that there is a flaw in one of the passport's protocols that makes it possible to trace the movements of a particular passport, without having to break the passport's cryptographic key. All an attacker has to do is to record one session between the passport and a legitimate reader, then by replaying a particular message, the attacker can distinguish that passport from any other. We have implemented our attack and tested it successfully against passports issued by a range of nations.

1 Introduction

New technologies lead to new threats. Traditionally security protocols have been analysed for a range of security and authenticity goals, however the introduction of small, promiscuous Radio Frequency Identifier (RFID) tags have raised new concerns. For instance, can a person's movements be traced using the RFID tags that have been inserted into the items they are carrying? As RFID tags will respond to any signal broadcast to them, and originally replied with a unique identifier, Benetton's proposal to place RFID tag in clothes caused a public outcry for precisely this reason [BB]; similar traceability concerns have also affected the New York area E-Zpass system [Cal]. Now RFID tags are being placed in passports.

The use of RFID tags in passports was primarily motivated by the desire to provide storage for bio-metric information such as fingerprints or iris scans [ICA06]. A suite of cryptographic protocols protects the data on the tag. Read access to the data on the passport is protected by the *Basic Access Control* (BAC) protocol. This protocol produces a session key by using another key derived from the date of birth, date of expiry and the passport number printed on the document. The aim of this protocol is to ensure that only parties with physical access to the passport can read the data. All data on the tag is signed by a document signing key which is in turn signed by a country key from the state that issued it. The public country verification keys are publicly available from the International Civil Aviation Organisation (ICAO)[1]. This process of

* This work is partly supported by EPSRC grant EP/F033540/1: Verifying Interoperability Requirements in Pervasive Systems.

[1] Currently at https://pkddownloadsg.icao.int

ensuring the integrity of the data is referred to as *Passive Authentication*. A third protocol, *Active Authentication*, ensures that the passport has not been copied by signing a nonce (a new random number) from the reader, using a signing key stored securely on the tag. The verification key, signed by the issuing country, can then be read from the tag and the passport verified by the reader. Both BAC and Active Authentication are specified as optional although BAC seems to be universally used[2]. We only observed Active Authentication on a few of the passports we looked at (e.g. the Irish passport).

In 2006 a second generation of e-passports were announced [ICA06] which included a new *Extended Access Control* protocol that would establish a session key based on a longer secret and would authenticate the reader to the tag using the country signing keys. This protocol would be run after the BAC protocol. A third generation of e-passport protocols are currently under discussion [BG08], although they have not yet been finalised by the ICAO.

The BAC protocol ensures that the data on the e-passport can only be read by someone who knows the key derived from the date of birth, date of expiry and number on the passport. Our attack lets someone who does not know this key trace a passport, i.e., if an attacker can observe a run of a particular passport then they can build a device that detects whenever the same passport comes into range of the reader. RFID tags receive their power via a signal from the reader; FCC regulations [FCC] limit the power of the readers, leading to an effective range of about 9cm. However, if the attacker disregards these regulations, they can power up the tag from a much greater distance, Kfir and Wool calculate that this is possible from a distance of up to 50cm [KW05]. If another reader powers the tag up, messages can be sent to and received from a tag to a range of several meters [Yos04, Han06]. This would make it easy to eavesdrop on the required message from someone as they used their passport at, for instance, a customs post. Furthermore, the RFID tags in passports are “always on” and give no indication to their owner that they are sending data.

A traceability attack does not lead to the compromise of all data on the tag, but it does pose a very real threat to the privacy of anyone that carries such a device. Assuming that the target carried their passport on them, an attacker could place a device in a doorway that would detect when the target entered or left a building. Juels et al. [JMW05] point out, rather melodramatically, that such an attack would make it possible to program a bomb that would explode in the presence of a particular person. More benignly, it could also be used to make a device that would tell a blind person whenever someone they had met before was close by. Such tracing attacks may also apply to other contactless devices. However, we believe that a traceability attack against e-passports is particularly severe because unlike, for instance, Bluetooth devices they cannot be turned off and also because a passport is a government mandated identity document and carrying one is compulsory when crossing a border or when resident in certain countries.

[2] Early US and Belgian e-passports did not have BAC, however BAC is now implemented.

The BAC protocol was closely based on ISO 11770-2 mech. 6 [ISO96]. It sets up a secure session key that the reader then uses to access the data. During a run of the BAC protocol, the passport generates a nonce that the reader must encrypt using the passport's unique encryption key. This ensures that messages are not being replayed to the passport. The reader and passport also generate Message Authentication Codes (MACs) for each message, using the passport's unique MAC key. This guarantees that the messages are received correctly and the MAC is checked before the nonce is looked at. This protocol protects the data on the passport, as any replayed or corrupted message will be rejected.

Our examination of actual passports has shown that it is possible to tell the difference between a message that was rejected because of an incorrect nonce and a message that was rejected because of a failed message authentication check. To trace a passport we eavesdrop on a legitimate session between a passport and a reader, and record the encrypted message that contains the passport's nonce. Then, when we want to identify a particular passport, we replay this message. If this replayed message is rejected because the MAC check failed then we know this is not the same passport, as the MAC key is unique to each passport. On the other hand, if the message is rejected because of the nonce check failed, we know that the MAC check using the unique passport key succeeded and therefore we have found the same passport again. In the case of the French passport different error messages are given in response to a failed MAC or an incorrect nonce. In the case of all other nationalities we tested, the rejection messages are the same but a failed MAC check is reported noticeably sooner than a failed nonce.

Many authors (e.g. [JMW05, CLRPS06, AKQ08]) have pointed out that the entropy used to seed the BAC keys is low, and in the case of countries where passport numbers are partly predictable it may be possible to guess the keys. However, passports are now being issued with a passport number made up of letters and numbers, rather than just numbers, which will increase the possible key entropy. It has also been pointed out that once a reader is given access to a passport it cannot be revoked [JMW05]. Richter et al. [RMP08] showed that the error messages issued by a passport were different for each country and so it was possible to uniquely identify the nationality of a passport drawn from a group of 10 European countries Ours is the only attack on e-passports that allows an attacker to remotely trace an individual passport, in real-time, for any passport numbering scheme, without having to know the BAC keys.

Our attack has a relatively simple fix; the error messages issued by the passports must be standardised and response times must be padded so as to remove the information leak. One way to do this would be to make e-passports decrypt messages even if the MAC check fails. For the tens of millions of passports already issued it is too late, however future passports can be made safe.

In the next section we describe the protocols used by e-passports and discuss other analysis of these protocols in Section 2.2. We present a protocol based attack against the French e-passport in Section 3 and extend this to a timing attack against all e-passports in Section 4. We discuss ways in which this attack may be stopped and conclude in Section 5.

2 The e-Passport Protocols

An e-passport[3] is an identification document combining a traditional passport with an RFID tag capable of performing cryptographic operations, storing biometric data and other bearer related information. The specification for e-passports is published by the International Civil Aviation Organization (ICAO) [ICA06] and more than 60 states have started issuing their own e-passports based on this standard.

The ICAO specification requires that passports use the contactless card standard ISO 14443 [ISO01] for hardware level communication. This standard defines how the reader should power up the card and select a particular tag to communicate with; if more than one tag is present, each card broadcasts a unique ID and the reader selects one, with which to establish a session. The ICAO specification recommends that the UID is randomised to avoid the possibility of it being used to trace a particular passport [ICA08, page 22]. If a country chooses to ignore this advice, then a passport will be easily traceable. All the passports we have looked at, so far, use randomised UIDs. ISO 14443 defines two ways in which radio signals can be used to communicate with the cards (Type A and Type B). E-passports may implement either method.

On top of the ISO 14443 communication, the ICAO specification states that the passports should implement some of the commands and error codes defined in the standard for contact-based smart cards ISO 7816 [ISO95]. As well as giving a detailed description of the layout of the data on the passport, it specifies that the passport should support the ISO 7816 commands SELECT FILE and READ BINARY for accessing the data on the tag. The instructions GET CHALLENGE, MUTUAL AUTHENTICATION and INTERNAL AUTHENTICATION are used for BAC and Active Authentication. The passports also use ISO 7816 error codes, such as “6A80: Incorrect parameters” or “6300: No information given”.

2.1 The Passport Protocols

The data on the passport is organised into 16 *data groups*, that can be read using the ISO 7816 SELECT FILE and READ BINARY commands. The ICAO specification defines what each data group should be used for: DG1 and DG2 are compulsory for all passports and store the machine-readable data printed on the passport and the passport photo respectively. DG3 to DG16 are for optional data, such as fingerprints (DG3, which we found on a recent German passport). The contents of some of these data groups have been defined but are not yet used in practice, such as iris scans (DG4), holder's signature (DG7) and the address of someone to contact in an emergency (DG16). Data groups 11 and 12 are for optional additional information depending on the country, for example,

[3] For the rest of this document we will use “passport” to mean “e-passport”, rather than a passport without an RFID tag, and only use e-passport when we want to underline the difference between the two.

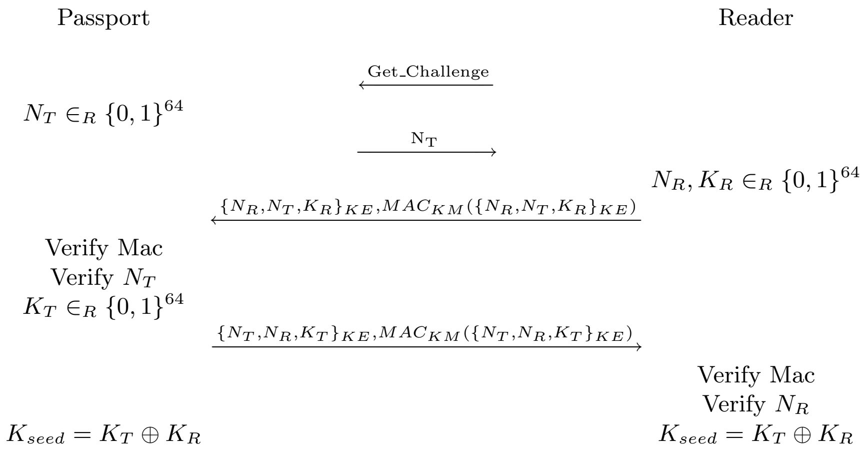

Passport
Reader
Get_Challenge
$N_T \in_R \{0,1\}^{64}$
$\mathrm{N_T}$
$N_R, K_R \in_R \{0,1\}^{64}$
$\{N_R,N_T,K_R\}_{KE}, MAC_{KM}(\{N_R,N_T,K_R\}_{KE})$
Verify Mac
Verify N_T
$K_T \in_R \{0,1\}^{64}$
$\{N_T,N_R,K_T\}_{KE}, MAC_{KM}(\{N_T,N_R,K_T\}_{KE})$
Verify Mac
Verify N_R
$K_{seed} = K_T \oplus K_R$
$K_{seed} = K_T \oplus K_R$

Fig. 1. The Basic Access Control Protocol

the French passport uses these to store the height[4] of the passport holder, their home address and the address of the police station where the passport was issued. According to the specification, the data groups are read-only. The hash of the data groups, which has been signed by the issuing state, is stored on the passport; checking this ensures that the passport is not forged.

Read access to the data on the passport is protected by the *Basic Access Control* protocol (BAC). This protocol uses a key generated from the date of birth, date of expiry and passport number printed on the passport and establishes a new session key to protect all following communication with the reader. The aim of this protocol is to prevent eavesdropping and skimming attacks by ensuring that only someone who has seen the information page of the passport can access the data on the tag. While other authors have criticised this design as less secure than, say, making the reader authenticate to the tag using a certificate, it does have the advantage of allowing moderately skilled users to see what is on their own passport.

BAC is a key establishment protocol, as shown in Figure 1. Here $\{\_\}_K$ denotes Triple-DES encryption with the key K and $MAC_K(\_)$ denotes a cryptographic checksum according to ISO 9797-1 Message Authentication Code Algorithm 3. The passport stores two keys: KE and KM, and the reader derives these keys using the machine-readable information on the passport, which has, in theory, been scanned before the wireless communication begins.

The reader initiates the protocol by sending a challenge to the tag and the tag replies with a random 64-bit string N_T. The reader then creates its own random nonce and some new random key material, both 64-bits. These are encrypted,

[4] We found cases where a French passport overestimated the height of its owner, this seems to be because the height measurement is not checked by the passport issuing organisation and so reflects the height that the passport holder would like to think of themselves as, rather than their true height.

along with the tag's nonce and sent back to the reader. A MAC is computed using the KM key and sent along with the message, to ensure the message is received correctly.

The tag receives this message, verifies the MAC, decrypts the message and checks that its nonce is correct; this guarantees to the tag that the message from the reader is not a replay of an old message. The tag then generates its own random 64-bits of key material and sends this back to the reader in a similar message, except this time the order of the nonces is reversed, in order to stop the reader's message being replayed directly back to the reader. The reader checks the MAC and its nonce, and both the tag and the reader use the xor of the key material as the seed for a session key, with which to encrypt the rest of the session.

This protocol guarantees that only parties who know the keys derived from the machine-readable zone can learn the session key and message freshness is guaranteed by the nonces. However, we observe that this protocol does not guarantee a fresh session key to the reader: as the passport picks its key material after it sees the reader's key material, and the material is xor-ed together, the passport may pick its material in such a way as to force a particular key seed. Although this does not seem to lead to an attack, concatenating the key material would have meant that both parties were guaranteed a fresh key.

Active Authentication is an optional protocol designed to prevent cloning attacks. The protocol is based on public key cryptography; the tag proves the possession of a private key with a straightforward challenge-response protocol. If the passport supports the Active Authentication protocol, the public key is stored in Data Group 15, which is signed along with the rest of the passport data. In 2006, the ICAO proposed a new set of protocols called Extended Access Control (EAC). These protocols are commonly used to protect sensitive biometric data, and require the reader to authenticate itself to the passport using a certificate signed by a country signing key. We observed EAC on a recent German passport, where it was used to protect fingerprints, and information on the EAC parameters was stored in data group 14. Both Active Authentication and EAC are optional and run after BAC, so, as our attack is against BAC, the additional security these protocols provide does nothing to stop our attack.

2.2 Related Work

Many papers have been written about the e-passport specification. One of the most popular themes is the low entropy of the BAC key seed. The original ICAO documentation points out that the ideal entropy of 73-bits is probably closer to 56-bits due to non-random passport numbers. A series of authors have then analysed the passport numbers of particular countries. For instance, Juels et al. [JMW05] pointed out the US passport only offers 54-bits of entropy, Carluccio et al. [CLRPS06] put the German passport's entropy at 55-bits, and Avoine et al. [AKQ08] put the Belgian passport at 38-bits. Most of these authors go on to assume that the attacker knows the birthday of their victim and so subtract another 15-bits from the key entropy. We note that all of these calculations are based on the assumption that the random part of the passport numbers only contain digits. This is no longer true: the passport number on German passports issued since, at least, 2008 include letters as well as numbers. Therefore, the entropy is now likely to be much higher than Carluccio et al. estimate.

The Belgian passports have such low entropy because the passport numbers are mostly numeric and issued sequentially, Avoine et al. show that an eavesdropping attack can find the key in about a second, whereas an online attack against only a passport could take a few weeks, in the worst case. Carluccio et al. [CLRPS06] and Liu et al. [LKLRP07] both present hardware architectures that can speed up the cracking process, however they also assume that the attacker has some previous knowledge about the victim, such as their birthday and has observed a correct run of the protocol. In contrast to this work, our attack is an attack on the protocol itself, rather than an attack against the weak key seed. We do not need to assume that the attacker knows the age of the victim and our attack works, in real-time against any passport numbering scheme.

Hoepman et al. [HHJ$^+$06] also discuss the low BAC entropy and point out that a passport would be traceable if it does not randomise its ISO 14443 UID. All the passports we have looked at do randomise their UIDs, although we have been told that passports from Italy and New Zealand do not.

Perhaps the most similar work to ours is that of Danev et al. [DHBv09] who show that a passport can be identified by its hardware characteristics with an error rate of 2% to 4%. However, to collect their readings they must place the passport in a specially constructed wood frame, therefore they suggest they that their method is better suited to detecting counterfeit passports than it is to tracing people.

2.3 Experimental Framework

To interact with the passports we used an ACR122U reader from Advanced Card Systems Limited. This is one of the cheapest (∼\$50) RFID readers on the market and while more expensive reader could collect more accurate timing data and performed tests faster, using such a reader underlines the fact that our attack does not need specialist hardware.

Adam Laurie's RFID Input/Output Tools (RFIDiot) project [Lau06] has developed a number of tools to make interacting with RFID tags easy. We found these tools very useful when initially experimenting with e-passports, and we have made use of Laurie's libraries when writing the code to perform our attack.

We ran our tests with passports volunteered by members of our lab and their families. We tested 10 passports in total: 3 UK, 2 German, 1 Russian, 2 French, 1 Irish and 1 Greek. We would like to extend our thanks to all of the volunteers that offered their passports for testing, and we were particularly pleased that no country had chosen to make their passports lock up after a set number of failed runs of the BAC protocol.

When taking a large number of time samples from a continuously powered passport we noticed that after around 100 readings in a row the response times from the passport would start to slow down by about 1ms every 20 readings. To

RFID tag	ATR value
UK Passport	3B898001097877D4020000900048
French Passport	3B8E80011177B3A7028091E16577010103FF61
Irish Passport	3B848001043833B1BB
German Passport, (numneric passport number, no fingerprints),	3B8E8001107833D402006404110101313180FFBD
German Passport (alpha-numeric passport number, fingerprints)	3B898001097877C4020000900058
Dubai Metro pass	3B8F8001804F0CA0000003060300030000000068
Mifare (e.g. Oyster card, Univ. Id)	3B8F8001804F0CA000000306030001000000006A

Fig. 2. ATR values from various RFID tags

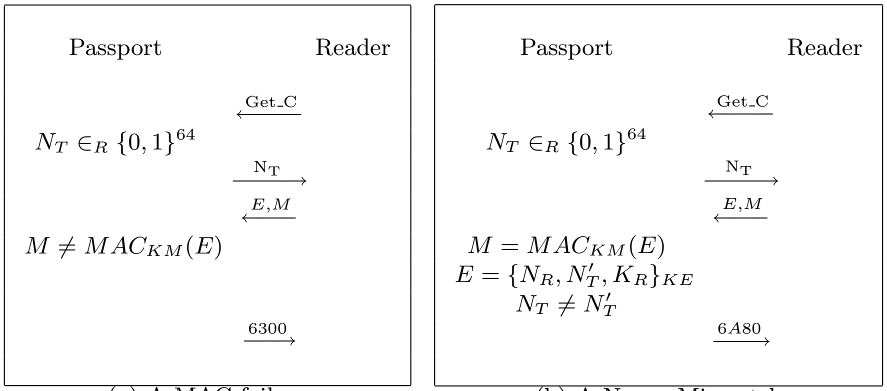
ensure that our sampled data was independent and identically distributed we powered down the tag between each time measurement.

2.4 Passport FingerPrinting via Answer to Reset

While the ICAO defines the specification for e-passports, all of the countries we have looked at have built their own implementations. Richter et al. [RMP08] exploit this fact, to show that it is possible to deduce which country issued a passport by the error messages it gives. They also mention other possible ways to detect the issuing country of a passport including the ISO 14443 "Answer to Select" or "File Control Information" message. We also found that the passports of different nations gave distinctive error messages, however we received different error messages to the ones reported by Richter et al., this may have been due to using different parameters in the ISO 7816 commands.

Contact-based ISO 7816 chips will respond to a reset with an "Answer to Reset" (ATR) message, which includes data on the chip's manufacturer and how the chip should be read. In the interests of compatibility, the Interface Device Handler (the firmware and/or drivers) for contactless card readers construct an ATR message for ISO 14443 tags [Wor07, Sec. 3.1.3.2.3]. These handler constructed ATR messages have a standard prefix, followed by the historical data from the "Answer to Select" for ISO 14443 Type A tags, or the application data and protocol information for ISO 14443 Type B tags. Furthermore, this constructed ATR message is generated when the reader initiates contact with the tag, and is therefore much easier to find than a complete set of error codes.

Out of the passports we tested, we found that each country had its own unique constructed ATR value, we also found that a range of mifare classic cards all issue the same ATR, see Figure 2. The German passport was recently updated to include an alpha-numeric passport number and the fingerprints of the owner. We found that these updated passports had a different ATR to the earlier version. Therefore, the ATR provides an easy way to identify, not just the issuing nation, but also the version of a passport. This is an additional weakness in the passport because if it is possible to narrow down the issue date of a passport it becomes easier to guess the BAC key. Some of the observed ATRs were very close so,



(a) A MAC failure

(b) A Nonce Mismatch

Fig. 3. The Basic Access Control Protocol

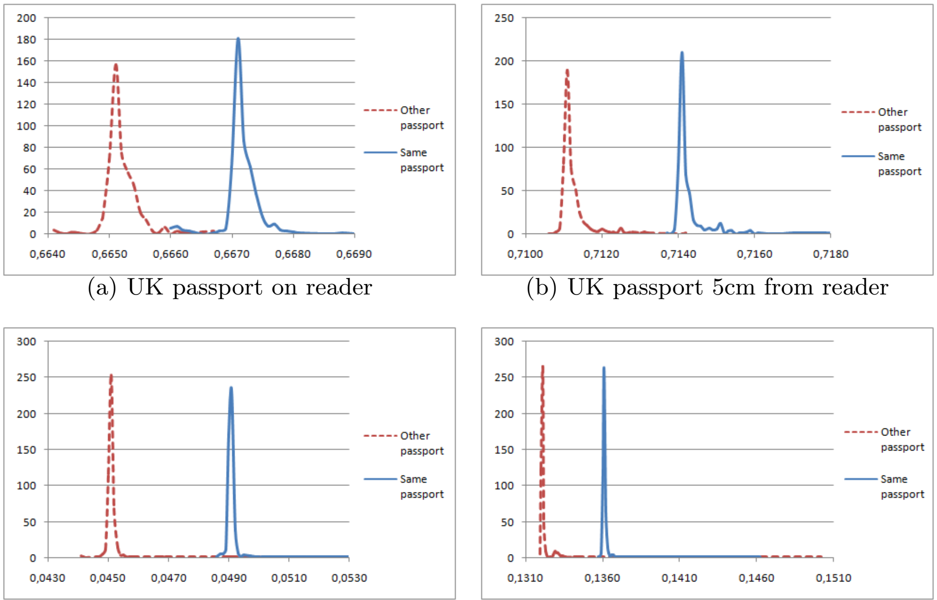
just as with error messages, there is a possibility of two different tags having the same profile. Hence, further research is needed before we can be sure that this is a good identification technique.

3 An Attack Against French e-Passports

The ICAO passport specification states that the passport must always respond to a message, returning an error message if the message was incorrect or unexpected. The fault in the French passport's BAC protocol becomes apparent when we consider the error messages that the passport generates in response to erroneous messages from the reader.

To find these error messages we power up the passport, according to ISO 14443, we then send a GET CHALLENGE message to initiate the BAC protocol to which the passport replies with a nonce. The reader should send the tag's nonce back to the passport, along with some keying material and its own nonce. This message should be encrypted with the passport's unique encryption key and sent with a MAC generated using the passport's unique MAC key. To find the error messages we tried broadcasting a message to the tag with an incorrect MAC, and found that the French passport replied with a "6300: No information given" error (Figure 3(a)). Next we formed a message with a correct MAC but with an incorrect nonce. This message was replied to with a "6A80: Incorrect parameters" error (Figure 3(b)).

These different error messages can be used to trace a passport, even by an attacker that does not have the passport encryption and MAC keys. First the attacker must observe a run of the passport with a reader that knows the passport key, for instance, while going through customs. The attacker records the message from the reader that contains the encrypted and MACed nonces and



(a) UK passport on reader

(b) UK passport 5cm from reader

(c) Greek passport on reader

(d) German passport on reader

Fig. 4. Sampled Times from Replaying a Message to the Same or a Different Passport

keying material. Later, when the attacker comes across another passport, they can use this recorded message to test if it is the same passport as they observed before: the attacker broadcasts a GET CHALLENGE message, to which the tag responses with a nonce. The attacker then replays the message they recorded from the previous run. If the tag responds with a 6300 error message then we know that the MAC check failed, therefore the passport we are currently looking at used a different MAC key from the original passport and is not the same one. If, on the other hand, we get a 6A80 message then we know that the MAC check must have succeeded, and so the current passport is the passport we are trying to trace.

4 A Time-Based Traceability Attack

Out of all the passports we tested, only the French passport responded to a failed MAC check and a mismatched nonce with different error messages; all the other passports issued the same error code, usually “6300”. So it seemed that this attack only affected French passports. However, examining the passports further, we noticed that the time it took for a passport to issue these error messages was not constant.

Figure 4(a) shows the time it took for a UK passport to issue the error message (to 4 decimal places). We sent 500 messages we knew would fail the MAC check (shown in dashed, red) and 500 replayed messages, with the correct MAC key, but with an incorrect nonce (shown in solid, blue). It is clear from this data that a failed MAC elicits a reply more quickly than a failed nonce. Looking at the protocol specification, it seems that this is because the passport rejects a message with an incorrect MAC straightaway, whereas if the MAC is correct, the MAC check is performed, the message is then decrypted and only after that can the nonce be checked. The additional time it takes to reply to a replayed message is the time it takes the passport to decrypt the message and check the nonce. After checking several passports, we found that the exact time difference depended mainly on which country issued the passport. For our particular reader, UK passports took around 2.8 milliseconds longer to respond to a replayed message, German, Greek and Irish passports took 4ms to 5ms and a Russian passport we tested took a sluggish 7ms.

We retested a UK passport, this time placing the passport 5cm away from the reader (Figure 4(b)). This data set clearly shows the time difference between a message replayed to the passport that generated it and a message replayed from a different passport. However, placing the passport away from the reader leads to all the messages taking longer. The time it takes the radio waves to cross the extra distance is of the order of 10^{-10} seconds so this slowdown is most likely explained by less power being supplied to the RFID tag. Such variations in response times mean that it is not possible to trace a passport with a single replayed message. Instead, the attacker must send a message they know will fail the MAC check, then send the replayed message and compare the response times.

The exact attack could be performed in a number of different ways. If a passport is known to be stationary then the attacker could send one completely random message and then replay the message from the passport they wish to trace. If the time difference is more than some value the attacker could decide that it is the same passport as before, and if it is less than that value the attacker could decide that it is a different passport. This test could be repeated for additional accuracy, the attacker could also use different lower and upper bounds, or attempt to work out the nationality of the passports via the ATR (as described in Section 2.4) and then pick the most efficient cutoff for that country. When the passport is moving it is necessary to send a number of different random messages interleaved with a number of replayed messages and then take the average. We find the error rates and efficiencies of these different methods using a statistical analysis of the response times.

Statistical Analysis of Passport Response Times The response times in Figure 4 appear to follow a normal distribution. Due to the limited accuracy of our measuring framework, we round our data to 4 decimal places. This makes our data discrete by placing the results into a number of bins, (e.g. all time measurements between 0.66505 and 0.66515 are placed in the 0.6651 bin). Therefore we can verify that the data is well modelled by a normal distribution using a χ^2 goodness of fit test. This test defines a test statistic:

$$X = \sum_{i=1,\ldots,k} \frac{(O_i - E_i)^2}{E_i} \tag{1}$$

where O_i is the observed number in bin i and E_i is the number predicted by the distribution. The sampled data is well modelled by a normal distribution if the X statistic is consistent with a $\chi^2_{(k-3)}$ distribution (see e.g. [SC89]). We carried out this test and found that the X statistic was within the 95% confidence interval for the British, German, Greek and Irish passports, both when the passport is directly on the reader or when placed 5cm away from it. We note that this does not mean that the distribution is exactly normal, but rather it means that a normal distribution is a reasonable model for the sampled data and is therefore useful in order to estimate the error rates.

The Russian e-passport was not consistent with a normal distribution. The time graphs for a 100 samples are given in Figure 5 (only 100 samples were taken due to limited access to the passport). As well as not following a normal distribution, the passport would not let us access any data after we have performed BAC, which suggests that the passport might not be fully compatible with the ICAO standard (EAC, if used, should only protect bio-metric data). Information on the Russian passport specification is sparse, and mostly in Russian (see e.g. [Min03, Eva05]), so this calls for further study. The time gap between random and replayed messages was the biggest we have seen for any passport and with no overlap at all; therefore our attack would work against Russian passports with a very high degree of certainty.

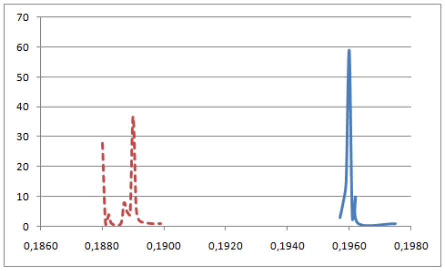


Fig. 5. Russian Sample Times

Looking at the timings that follow a normal distribution, we can calculate the rates of false positives and false negatives for particular tests. We know that the difference between a value from a distribution $\mathcal{N}(m_1, v_1)$ and a value from the distribution $\mathcal{N}(m_2, v_2)$ will come from the distribution $\mathcal{N}(m_1 - m_2, v_1 + v_2)$. Therefore, the difference in response times in milliseconds, for a random message and a message replayed to the same UK passport it came from will come from the distribution $\mathcal{N}(2.8, 0.63)$, whereas the difference in response times for a different passport, one that did not generate the message being replayed, would come from the distribution $\mathcal{N}(0, 0.62)$. The distributions of these differences for all of the different passports are shown in the first 2 columns of Figure 6.

A false positive occurs when we test a different passport and decide that it is the one that generated the message we are replaying, whereas a false negative occurs when we test the passport that generated the message we are replaying but fail to identify it as the same passport. The simplest test is for the attacker to send one random message and one replayed message. Using the distributions in Figure 6 we calculated that if the attacker decides that it is the same passport when the time difference is more than 1.7ms and a different passport when the difference is less than 1.7ms, then the worst false positive probability is 0.084 and the worst false negative rate is 0.084. If the attacker repeats this test, taking

Passport	Same	Different	Prob. False Pos.	Prob. False Neg.
Country	Passport	Passport	at 1.7ms	at 1.7ms
UK	$\mathcal{N}(2.8, 0.63)$	$\mathcal{N}(0, 0.62)$	0.015	0.084
German	$\mathcal{N}(3.9, 0.124)$	$\mathcal{N}(0, 0.52)$	0.009	0.024
Greek	$\mathcal{N}(4.0, 1.57)$	$\mathcal{N}(0, 1.21)$	0.061	0.033
Irish	$\mathcal{N}(5.2, 0.79)$	$\mathcal{N}(0, 1.52)$	0.084	0.00004

Fig. 6. Distribution of Time Differences and the Error Rates

the *best out of 3* the false positive and negative probabilities fall to 0.02 and for the *best out of 5* the error rates are 0.005.

If the attacker decides that the passport is the same when the difference is more than 2.8ms, a different passport when the difference is less than 1.0ms and runs another test when the difference is in between these values, we find that the probability of a false negative is 0.011 and a false positive is 0.012 and the expected number of trials is 4.8. This suggested that, for the passports we tested, the most efficient test, that balances the false positives and false negative, is to use 1.7ms as a cutoff and running extra trails to get the accuracy required. We implemented this test, using the best out three, and wrote a program that tested a passport against a database of replay messages from each of the 10 passports we examined, in turn. In 20 tests our program correctly identified every passport from the time delay of its replay message.

To test the feasibility of our attack against a moving target we tried taking a number of readings from a passport while it was moved across the reader. We averaged these readings, and found that some readings would take up to a few seconds and have a disproportional effect on our averages, therefore we discarded any time measurements that were more than one second. Used the single time cutoff, as described above, we found that with just one test the false positive and negative probabilities where as high as 0.32, however with 50 tests these probabilities fell to 0.21. With 100 observations, taking less than a minute, the error probabilities where as low as 0.1, suggesting that this attack is feasible against a moving target. The reader we used was the cheapest hardware we could find; we expect that more advanced readers with specialised hardware may be able to perform these attacks far more quickly and to a higher degree of accuracy.

5 Conclusion

Our work shows the inherent dangers of using RFID tags in personal items. The e-passport specification was developed by experts over many years and since its publication has been the subject of dozens of academic studies. During this time e-passports have been issued to over 30 million people, all of whom may be at risk of being traced using our attack. As future work we would like to examine more passports and test our attack against other RFID enabled identity documents.

The fix for our attack is relatively simple. First, all e-passports must standardise their error messages. The required error messages in all possible situations should be specified by the ICAO (in e.g. [ICA08]). Second, in the BAC protocol, after a MAC check fails, the passport should try to decrypt the message and check the nonce anyway before sending the error message. Care must also be taken when implementing new protocols, as our attack might work against any protocol that requires an RFID tag to first check a MAC before decrypting and processing some data.

Our attack is only feasible because the e-passports contain an RFID tag. If e-passports used, for instance, a contact based smart card, then such attacks would not be possible. The reasons for making the e-passports wireless is not immediately clear, the ICAO documentation [ICA06] mentions that reasons for choosing RFID include high data transfer rates, reduced wear and tear on the document and that contact based readers do not fit the shape of the passport. However, contact-based smart cards are quite capable of transferring the data on the card in a reasonable amount of time, and the BAC protocol requires the contact based reading of the passport number and date of birth and expiry, so these reasons seem weak.

Worryingly, the protocols that are used in e-passports are also to be used in some national identity cards, such as the proposed UK ID card scheme [Bog09]. While we have not been able to confirm if these cards will be RFID or contact based, it is possible that our attack will also work against these. It is quite possible that, at some point in the future, it will become a legal requirement for people to carry such an RFID enabled cards and their use will become common to, for instance, access health care, prove identity at an airport or a bank, prove age at a bar, etc. The use of our attack in such a possible future would make it possible for anyone to trace the movements of anyone else.

Acknowledgments We would like to thank Henryk Plötz for helpful comments regarding how the Interface Device Handler constructs the ATR message.

References

[AKQ08] Gildas Avoine, Kassem Kalach, and Jean-Jacques Quisquater. ePassport: Securing International Contacts with Contactless Chips. In *Financial Cryptography*, volume 5143 of *LNCSe*, pages 141–155, 2008. [cited p. 3, 6]

[BB] Boycott benetton. Retrieved 26/8/2009, http://www.boycottbenetton.com/. [cited p. 1]

[BG08] BSI-Germany. Advanced security mechanisms for machine readable travel documents. Technical report, Federal Office for Information Security, 2008. [cited p. 2]

[Bog09] Steve Boggan. New id cards are supposed to be unforgeable. *Daily Mail*, August 2009. http://www.dailymail.co.uk/news/article-1204641. [cited p. 14]

[Cal] Christopher Caldwell. A pass on privacy? The New York Times, July 17, 2005. [cited p. 1]

[CLRPS06] Dario Carluccio, Kerstin Lemke-Rust, Christof Paar, and Ahmad-Reza Sadeghi. E-passport: The Global Traceability or How to Feel Like an UPS Package. In *Workshop on RFID Security – RFIDSec*, 2006. [cited p. 3, 6, 7]

[DHBv09] Boris Danev, Thomas S. Heydt-Benjamin, and Srdjan Čapkun. Physical-layer Identification of RFID Devices. In *Proceedings of the 18th USENIX Security Symposium – USENIX'09*, 2009. [cited p. 7]

[Eva05] Alexander Evangeli. Biometric passport: the whole world under control, 2005. (In Russian), http://www.pcweek.ru/themes/detail.php?ID=69892. [cited p. 12]

[FCC] Title 47–telecommunication, chapter i–federal communications commission, part 15–radio frequency devices. [cited p. 2]

[Han06] Gerhard P. Hancke. Practical attacks on proximity identification systems. In *Symposium on Security and Privacy*, pages 328–333, 2006. [cited p. 2]

[HHJ^{+}06] 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 *Advances in Information and Computer Security, First International Workshop on Security – IWSEC*, volume 4266 of *LNCS*, pages 152–167, 2006. [cited p. 7]

[ICA06] ICAO. Machine Readable Travel Documents. Doc 9303. Part 1. Technical report, International Civil Aviation Organization, 2006. [cited p. 1, 2, 4, 14]

[ICA08] ICAO. Supplement to doc 9303. Technical report, International Civil Aviation Organization, 2008. [cited p. 4, 14]

[ISO95] *Information technology – Identification cards – Integrated circuit(s) cards with contacts – Part 4: Interindustry commands for interchange*, 1995. ISO/IEC 7816-4. [cited p. 4]

[ISO96] *Information technology – Security techniques – Key management – Part 2: Mechanisms using symmetric techniques*, 1996. ISO/IEC 11770-2. [cited p. 3]

[ISO01] *Identification cards – Contactless integrated circuit cards – Proximity cards*, 2001. ISO/IEC 14443. [cited p. 4]

[JMW05] Ari Juels, David Molnar, and David Wagner. Security and Privacy Issues in E-passports. In *SecureComm*, 2005. [cited p. 2, 3, 6]

[KW05] Ziv Kfir and Avishai Wool. Picking Virtual Pockets Using Relay Attacks on Contactless Smartcard Systems. In *SecureComm*. IEEE, 2005. [cited p. 2]

[Lau06] Adam Laurie. RFIDIOt, 2006. http://rfidiot.org/. [cited p. 7]

[LKLRP07] Yifei Liu, Timo Kasper, Kerstin Lemke-Rust, and Christof Paar. E-Passport: Cracking Basic Access Control Keys. In *OTM Conferences (2)*, volume 4804 of *LNCS*, pages 1531–1547, 2007. [cited p. 7]

[Min03] Victor Minkin. Myths and realities of biometric passport system, 2003. (In Russian), http://www.elsys.ru/review7.php. [cited p. 12]

[RMP08] Henning Richter, Wojciech Mostowski, and Erik Poll. Fingerprinting Passports. In *NLUUG Spring Conference on Security*, 2008. [cited p. 3, 8]

[SC89] George W. Snedecor and William G. Cochran. *Statistical Methods*. Iowa State University Press, 8 edition, 1989. [cited p. 12]

[Wor07] The PC/SC Workgroup. *Interoperability Specification for ICCs and Personal Computer Systems. Part 3*, 2007. [cited p. 8]

[Yos04] Junko Yoshida. Tests reveal e-passport security flaw. *Electronic Engineering Times*, 1336:1, 30 August 2004. [cited p. 2]