Security Analysis of Australian and E.U. E-passport Implementation

Vijayakrishnan Pasupathinathan and Josef Pieprzyk

Department of Computing, Macquarie University
New South Wales, Australia 2109
{krishnan,josef}@ics.mq.edu.au

Huaxiong Wang

Division of Mathematical Sciences, School of Physical & Mathematical Sciences
Nanyang Technological University
Singapore 639798
and
Department of Computing, Macquarie University
New South Wales, Australia 2109
hxwang@ntu.edu.sg

This paper makes a formal security analysis of the current Australian e-passport implementation using model checking tools CASPER/CSP/FDR. We highlight security issues in the current implementation and identify new threats when an e-passport system is integrated with an automated processing system like SmartGate. The paper also provides a security analysis of the European Union (EU) proposal for Extended Access Control (EAC) that is intended to provide improved security in protecting biometric information of the e-passport bearer.

The current e-passport specification fails to provide a list of adequate security goals that could be used for security evaluation. We fill this gap; we present a collection of security goals for evaluation of e-passport protocols. Our analysis confirms existing security weaknesses that were previously identified and shows that both the Australian e-passport implementation and the EU proposal fail to address many security and privacy aspects that are paramount in implementing a secure border control mechanism.

ACM Classification C.2.2 (Communication/Networking and Information Technology – Network Protocols – Model Checking), D.2.4 (Software Engineering – Software/Program Verification – Formal Methods), D.4.6 (Operating Systems – Security and Privacy Protection – Authentication)

1. INTRODUCTION

Due to an increased risk of terrorism, countries are adopting biometric enabled passports as a preventive measure to monitor and strengthen their border security. In 2005, Australia introduced biometric passports that conform to the e-passport guideline developed by International Civil Aviation Organisation (ICAO), a United Nation body responsible for setting international passport standards. The ICAO established five task forces under the New Technology Working Group (NTWG) to develop a standard for Machine Readable Travel Documents (MRTD) (ICAO, 2006).

Manuscript received: 3 July 2008
Communicating Editor: Ljiljana Brankovic

The ICAO standard DOC 9303 for MRTD describes a contactless smart card microchip that conforms to ISO-14443 (ISO/IEC, 2000) and is embedded within an e-passport booklet. The microchip duplicates the information that appears on a passport's bio-data page and which is recorded in the Machine Readable Zone (MRZ). The e-passport standard provides details not only about establishing a secure communication between an e-passport and an Inspection System (IS), but also about authentication of an e-passport, and provides details about its storage mechanism and biometric identifiers that should be used.

1.1 Related Work

Juels *et al* (2005) discussed security and privacy issues that apply to e-passports. They expressed concerns that, the contact-less chip embedded in an e-passport allows the e-passport contents to be read without direct contact with an IS and, more importantly, with the e-passport booklet closed. They argued that data stored in the chip could be covertly collected by means of "skimming" or "eavesdropping". Because of a low entropy, secret keys stored would be vulnerable to brute force attacks as demonstrated by Laurie (2007). Kc and Karger (2005) suggested that an e-passport may be susceptible to "splicing attack", "fake finger attack" and other related attacks that can be carried out when an e-passport bearer presents the e-passport to hotel clerks. There has been considerable press coverage (Johnson, 2006; Knight, 2006; Reid, 2006) on security weaknesses in e-passports. These reports indicated that it might be possible to "clone" an e-passport.

The "cloning" attack does not compromise authentication at a border security checkpoint, as an e-passport bearer is physically present and is verified against the details available in the e-passport (photograph stored in the chip). To compromise authentication, an attacker needs to modify the details but still maintain the integrity of the data and its corresponding hashes. However, cloning of an e-passport is a major privacy issue as an attacker would not only be able to obtain the passport bearer's details but also his/her biometric details stored in an e-passport. The risk of eavesdropping is increased by the surveillance environment in which border checks occur, particularly as border control processes become more and more automated, as in Australian SmartGate system (Australian Customs Services, 2006). This may ultimately assist an attacker in a covert collection of e-passport data.

To address some of these concerns the NTWG made further discussions about standardizing the next generation of e-passports. They decided to support extended access control (EAC), which is based on the EU proposal (Home Affairs Justice, 2006) for EAC. The primary goal of EAC is to provide a mutual authentication, in particular, an authentication of IS and additional security for biometrics. The first generation e-passports have a single biometric identifier, based on the facial biometric, whereas, the second generation includes both fingerprints and iris scan biometric identifiers.

This paper presents a formal analysis of the first generation e-passport protocols. We have been to formally verify that e-passport protocols do not meet basic security goals like data confidentiality, data integrity, key integrity, mutual and data origin authentication and, are vulnerable to attacks that would compromise both privacy and security of an e-passport bearer. We then provide our analysis and identify security weaknesses in EAC. We believe that, the EAC proposal also fails to provide adequate security and more importantly introduces new security weaknesses and implementation problems that include (1) the failure to prevent the biometric information from being released to a malicious IS in possession of MRZ details, (2) the lack of protection against passport skimming and (3) extensive reliance on the PKI.

1.2 Organisation

In Section 2, we provide a detailed description of current Australian e-passport implementation and the proposed EU mechanism where, the focus being on cryptographic protocols. In Section 3, we define our security goals for a formal verification of e-passport protocols and present our security analysis of the entire protocol suite for the highest level of security as defined by the ICAO guideline. In Section 4, we present our formal verification of the e-passport implementation using CASPER/CSP/FDR and our security analysis of the EU EAC mechanism. Finally, we conclude in Section 5 with a summary of weaknesses and recommendations for a better e-passport implementation.

2. E-PASSPORT SPECIFICATION

2.1 Operation

An e-passport bearer presents his/her document to a border security officer who scans the MRZ information in the e-passport through a MRZ reader and then places the e-passport near an e-passport reader to fetch data from the microchip. The current implementation consists of three protocols:

1. Basic Access Control (BAC) protocol (optional): It provides encrypted communication between the chip and the Inspection System (IS).
2. Passive Authentication (PA) protocol (mandatory): A border security officer reads and verifies the authenticity of e-passport content stored in the chip.
3. Active Authentication (AA) protocol (optional): It provides integrity verification of e-passport's data.

The EU EAC mechanism involves two new protocols that intend to replace active authentication and thus now consists of the following four protocols:

1. Basic Access Control (BAC) protocol (mandatory): It facilitates the e-passport and the IS to establish an encrypted communication channel.
2. Chip Authentication (CA) protocol (mandatory)
3. Passive Authentication (PA) protocol (mandatory): As in first generation passport standard.
4. Terminal authentication (TA).

Only if all protocols are completed successfully, the e-passport releases sensitive information like secondary biometric identifiers. If an IS does not support EU EAC, the e-passport performs the collection of protocols as specified in the first generation e-passports, therefore providing backward compatibility.

2.2 Data Structure

For interoperability, the ICAO's e-passport guideline provides details on how data should be stored in a microchip. The data elements are grouped together as a data group (DG) and collectively stored in a logical data structure (LDS). The ICAO guideline segregates data elements into 19 data groups and the LDS is categorised into three parts:

1. Data defined by the issuing state or organisation (mandatory). It contains the details recorded in the machine readable zone (MRZ), which includes, the passport number, passport bearer's name, nationality, date of birth, date of expiry, encoded facial biometric image and checksum of individual data elements used to derive the session key.
2. Data defined by the issuing state or organisation (optional). It includes biometric data for identification like finger prints (mandatory in EU EAC), iris scan, displayed identification data

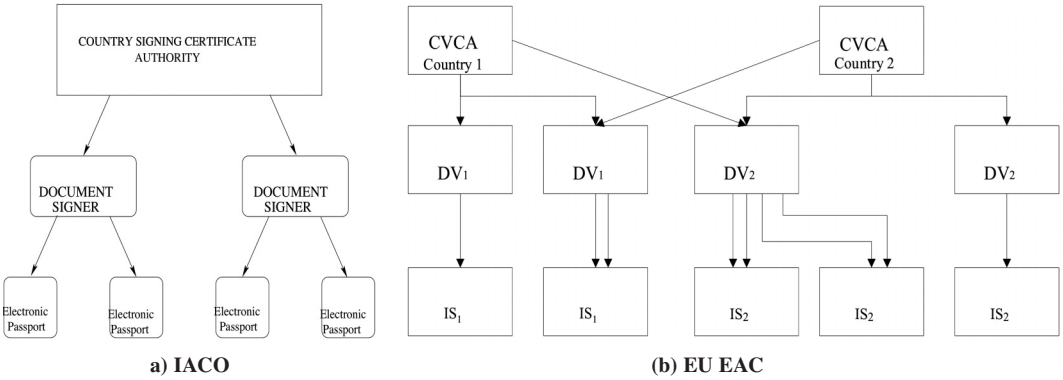
like digitised signature and any additional personal or document details like contact details, proof of citizenship and endorsements.

3. Data defined by the receiving state or organisation (optional). It holds details for automated border clearance, electronic visas and other travel records.

The data groups from 1 to 16 are defined by the issuing state and are read-only, whereas the data groups from 17 to 19 can be modified by authorised states or organisations. The write access is currently not supported, but ICAO plans to implement it in the future generation of e-passports. The LDS is stored in the microchip using the file system as defined in ISO/IEC 7816-4. The dedicated file (DF) in the chip file system hierarchy stores the encryption, MAC (used in basic access control protocol), and private key of the e-passport bearer (used in active authentication protocol). The elementary file (EF) in the chip hierarchy will store security object descriptors (SO_D) and data groups. The SO_D contains the hashes of LDS data elements digitally signed by the issuing organisation (document signer (DS)) and corresponding certificate (C_{DS}). An important security feature is that data groups are individually hashed and collectively signed by the issuing state and stored in SO_D, thus binding the biometric details with the e-passport bearer details.

2.3 E-passport PKI

The PKI section of the ICAO's e-passport document (ICAO, 2006) makes an important distinction between an issuing state and an issuing organisation. The issuing state represents the country of e-passport's origin whereas; the issuing organisation represents a passport issuing office within a country.



a) IACO **(b) EU EAC**

Figure 1: PKI for Current and EU EAC e-passport verification

Each country signing certification authority (CSCA) is required to forward their self-signed certificate ($CERT_{CVCA}$), document signer certificates ($CERT_{DS}$) and certificate revocation lists (CRL) to ICAO to be published at ICAO PKI directory (PKD). ICAO also recommends that issuing states replicate the PKD and CRL both locally and bilaterally among participating states every 90 days.

ICAO suggests the $CERT_{DS}$ be also stored in an e-passport chip, so a border security officer could continue with active authentication in case a PKD is unavailable, but this can compromise security as described later in Section 4.

EU EAC mechanism includes modifications to the ICAO's PKI. CSCA is now required to certify document verifiers (DV) in other countries, which in turn certifies inspection systems (IS) present at a country's border security checkpoint. Figure 1 provides an overview of the modified PKI hierarchy.

2.4 Passive and Active Authentication Protocols

Mandatory passive authentication mechanism provides only a basic level of security, as an e-passport is still vulnerable to skimming or eavesdropping attacks. Currently, USA is the only country that is implementing this level of security. But, due to considerable debate and pressure from researchers and privacy advocates, the US government is considering other optional security measures recommended by ICAO. Passive authentication is used to verify the integrity and to authenticate the data stored in the LDS and SOD, thereby authenticating the e-passport bearer.

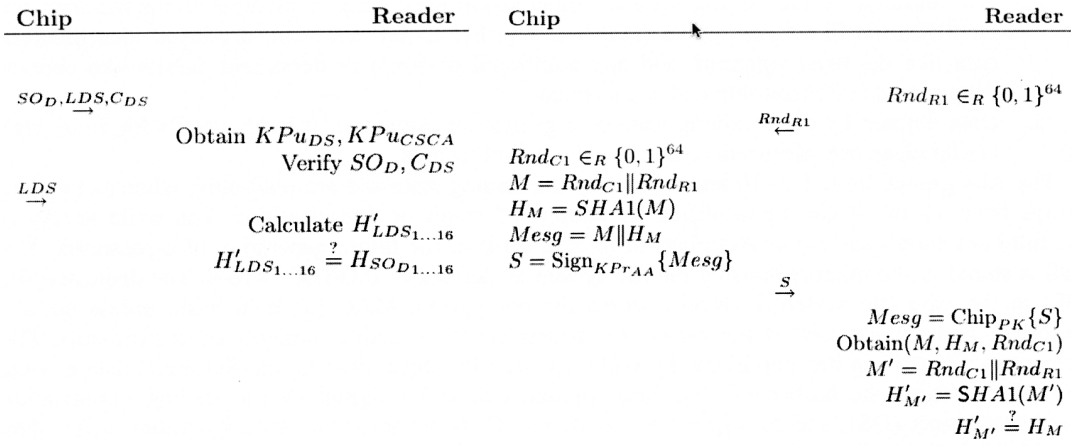


Figure 2: Passive and Active Authentication Protocols

Active authentication is an optional ICAO security feature that relies on public key cryptography and is intended to protect against chip modification and chip cloning. The ICAO guideline uses ISO/IEC 7816 internal authenticate mechanism along with signature computation according to ISO 97986-2 digital signature scheme 1. The reader initiates the protocol by sending an 8 byte random nonce to the e-passport. On receiving a challenge from the reader the chip digitally signs and returns the result. The reader then verifies the signature using KP_{uAA} obtained from SOD.

2.5 Basic Access Control Protocol

Basic access control protocol is an optional ICAO security mechanism that uses ISO 11770-2 Key Establishment Mechanism 6 to form a secure communicational channel between a reader and a chip. The protocol uses two secret keys (KENC,KMAC) that are stored in the e-passport chip. The reader derives both these keys using scannable data present in MRZ, namely, e-passport number, date of birth of the e-passport bearer, date of e-passport validity and check digits for those values. The reader initiates the three-pass challenge-response protocol by requesting a challenge from the chip. On receiving the challenge (Rnd_{C2}), the reader creates a checksum according to the ISO/IEC 9797-1 MAC algorithm 3 over the cipher-text that contains the reader's response to the chip's challenge (Rnd_{R2}) and the keying material (KR). On obtaining the reader's response, the chip creates a checksum that includes its keying material (KC). Both the reader and chip verify the MAC obtained and decrypt the encrypted message to reveal both keying materials that form the "key seed" (K_{seed}). K_{seed} is then used to derive a shared session key using the key derivation algorithm described in Section 2.6.

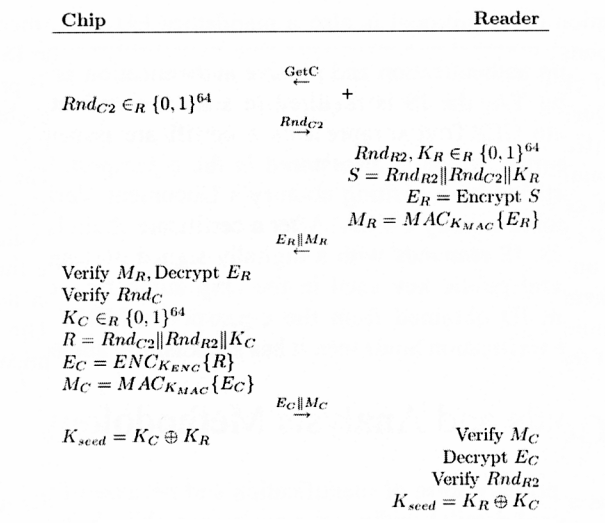


Figure 3: Basic Access Control Protocol

2.6 Key Derivation

The value c is a 32 bit counter that allows to derive multiple keys from a single seed. Depending on whether a key is used for encryption or for MAC, the value c is assigned.

- c = 1 (ie., '0x 00 00 00 01') for encryption
- c = 2 (ie., '0x 00 00 00 02') for MAC

The following steps are performed to derive both encryption and MAC keys that are to be used in 3DES.

a. D = K_{seed}kc
b. $H_{1...20}$ = SHA1(D)
c. $k_a = H_{1...8}$ and $k_b = H_{9...16}$
d. Adjust parity bits for k_a and k_b to form DES keys.

2.7 Chip and Terminal Authentication Protocols

Chip authentication (CA) protocol is a mandatory EU EAC mechanism that replaces active authentication protocol proposed in the first generation e-passports. It involves a Diffie-Hellman key agreement and is followed by the passive authentication protocol. It is performed after a successful BAC and provides both an authentication of the chip and generation of a session key. The chip sends its public key (PK_C) and its domain parameters (D_C) to IS. IS then generates an ephemeral Diffie-Hellman key pair (SK_R,PK_R) using the same domain parameters and sends the newly generated public key to the chip. Both the chip and IS derive a new session key K. The chip authentication is immediately followed by a passive authentication. This allows IS to verify whether PK_C is genuine.

Terminal authentication (TA) protocol is also a mandatory EU EAC mechanism that involves a two-pass challenge-response protocol and allows the chip to authenticate an IS. TA is only carried out after a successful run of chip authentication and passive authentication as it provides only a unilateral authentication of IS. During TA, the IS is required to send a certificate chain ($CERT_{IS}$, $CERT_{DV}$, $CERT_{CVCAH}$). The certificate CERTCVCAH represents a certificate issued by the e-

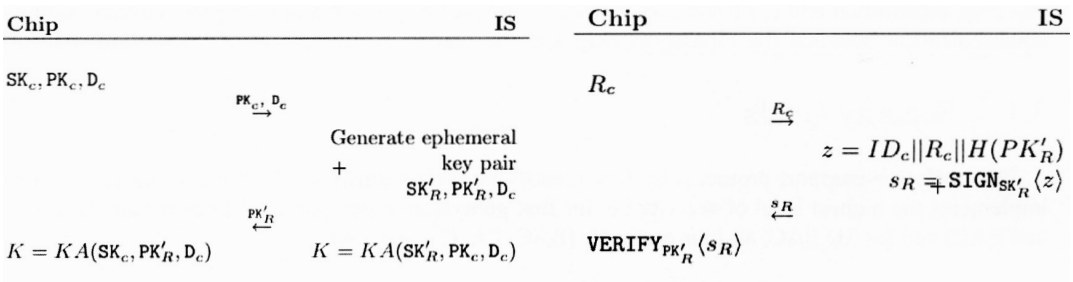


Figure 4: Authentication Mechanism in EU EAC enabled e-passports

passport's home country's certification authority, which is also stored in the e-passport. The chain indicates that the visiting country's IS is certified by a visiting country's Document Verifier (DV), which in turn is certified by a e-passport's home country CVCA. After a certificate chain is validated by the e-passport, it sends a challenge to an IS. IS responds with a digitally signed message that contains the received challenge, the IS's ephemeral public key used in the chip authentication and e-passport ID (ID_c), where, ID_c is the document ID obtained from the e-passport's MRZ. The e-passport verifies the signature received and if the verification holds then it has successfully authenticated IS.

3. SECURITY GOALS AND ANALYSIS METHODOLOGY

Passports are used as a primary form of identification and because of the nature of contents that is stored (biometric and personal details) within an e-passport's chip, it is crucial that the document is tamper-resistant and also maintains secrecy of data.

DOC 9303 (ICAO, 2006) provides a brief description of security goals that are achieved and cannot be achieved by the proposed mandatory and optional security mechanisms. If a country implements only the mandatory security requirement (PA), then authenticity and integrity of both SOD and LDS are provided. It does not, however, prevent data copy, chip substitution or skimming

Method	Security benefits	Vulnerabilities/Weaknesses
Passive Authentication	• Provides authenticity, integrity for SOD and LDS	• Failure to detect chip substitution. • Failure to prevent against chip copy, unauthorized access and skimming.
Active Authentication	• Prevents against duplication of SOD and chip modification	• Implementation complexity as resources (Memory, CPU) are needed.
Basic Access Control	• Prevents against skimming and eavesdropping	• Failure to detect chip substitution. • Failure to prevent against chip copy. • Implementation complexities as extra resources (Memory, CPU) are needed.

Table 1: DOC9303 security benefits and drawbacks

and also does not prevent against an unauthorised access to e-passport. For a greater security the ICAO recommends the implementation of other security mechanisms like: (1) active authentication to prevent copying of SOD and chip substitution and (2) basic access control protocol to prevent skimming and eavesdropping on communication between the e-passport chip and the reader. An overview of DOC 9303's security benefits and drawbacks is given in Table 1.

3.1 Security Goals

We analyse e-passport protocols by first identifying their security goals. We assume that a country implements the highest level of security i.e, for first generation e-passports, all three protocols (PA, AA and BAC) and for EU EAC, all four protocols (BAC, PA, CA and TA).

1. Data Confidentiality: Data confidentiality ensures the privacy of e-passport details and encryption is the common technique that provides confidentiality. In the case of e-passport, encryption is used to create a secure channel between the e-passport reader and the microchip. Note that the cryptographic keys used for encryption have to be guarded against unauthorized access (data elements within the LDS or keys stored in the DF).
2. Data Integrity: Data integrity prevents against illegal modifications of information exchanged between the e-passport reader and the microchip. Also the DF, SOD and LDS should be secure against any unauthorised modifications, i.e., any data tampering should be easily detectable by the border security centre.
3. Data Origin Authentication: Data origin authentication ensure that the source of the transmission in a protocol is authentic, i.e., the data on the chip should be bound to information on MRZ and to the data that appears in the e-passport bio-data page currently being examined by a border security officer.
4. Non-Repudiation: Non repudiation provides the ability to prove an action or an event that has taken place, such that protocol participants cannot later deny having processed that data. E-passports have an advantage, as the e-passport bearer will be physically present at the border security checkpoint. Nevertheless, it would be important to obtain an undeniable digital data from the e-passport for future processing, e.g., in case of an aftermath of a terrorist attack to validate the entry of the e-passport bearer at a particular security checkpoint.
5. Mutual Authentication: Mutual authentication is the process where both participants prove their identities to each other. As in the goal 3, where the e-passport reader authenticates an e-passport, this goal protects the e-passport bearer, as it is crucial for an e-passport to authenticate the e-passport reader before divulging any personal information. This prevents an unauthorised e-passport reader from obtaining biometric and personal details from an e-passport.
6. Certificate Manipulation: Certificates acts as an off-line assurance from a trusted authority that the certified public key really does belong to the principal who is in possession of corresponding secret key. However, it is the responsibility of the protocol to validate that the corresponding secret key is actually held by the principal claiming ownership of the public key. The e-passport reader should have a guarantee that certificates presented by the e-passport are valid and match the data on the e-passport. ICAO has implemented a PKI (Tom A. F. Kinneging for ICAO-NTWG, 2004) which would store signature certificates from issuing state and organisations.
7. Key Freshness and Key Integrity: Key freshness and key integrity protects against replay attacks. The e-passport reader and e-passport must have satisfactory proof that, a nonce generated in protocols is fresh and the integrity of the derived session key is preserved. Both parties should also have undeniable proof that the other party is in possession of a valid session

key. Any previous compromised key should be easily detected and the protocol run should terminate.

8. Forward Secrecy: Forward secrecy is concerned with protecting information that was not compromised before the long term key was lost. In an e-passport protocol, loss of session key or key used to generate a session key (KENC and KMAC) should not compromise any future communication.

3.2 Formal Representation

Model checking approach has been very successful in finding faults in many protocols (Dang and Kemmerer, 1997; Heintze and Tygar, 1994; Lowe, 1996; Lowe and Roascoe, 1997; Mitchell *et al*,1997; Pasupathinathan *et al*, 2006; Schneider, 1997). The approach is based on modelling a protocol as a finite state system by specifying its properties and then using a model checker to verify the systems properties. The advantages of using model checkers are that the verification process is usually automated and if a verification fails on a required property, the model checker lists the sequence of events that led to the failure.

FDR2 (Formal Systems (Europe) Ltd, 2003) is a model-checking tool for state machines, based on Communicating Sequential Processes (CSP) (see Hoare, 1985)). The verification technique is based on the method of establishing whether a property holds by testing for refinement of a transition system and the ability to check determinism of a state machine that is primarily used for checking security properties. Casper (Lowe, 1999) developed by Gavin Lowe, is a compiler, which converts a high level specification of the protocol to a CSP script. The CSP script can then be run on a model checker like FDR2 designed to automate the process of carrying out refinement checks and to verify if the protocol meets its security requirements.

An apparent limitation of this approach is that the verification of a complex protocol suite can lead to an exponential state-space explosion causing the checker to breakdown. Thus a formal model does not cover all possible states of a protocol. Normally the underlying cryptographic functions (like encryption, hash functions, etc.) are assumed to be true. The verification of the simplified protocol that was formalised does not necessarily mean that the full version of the protocol is secure against attacks but only suggests that the requirements of the simplified protocol are satisfied. Nevertheless, it does provide an assurance to users and designers about the relevant security goals that are met by the protocol.

3.3 Modelling E-passport Protocols

The ICAO e-passport is a complex protocol suite that consists of three sub protocols namely, BAC, PA and AA. Such a protocol suite is not only difficult to formalise, but also verification of such systems more often leads to an exponential state-space explosions. We model the flow of e-passport protocol according to the following stages:

1. When an e-passport is presented at a border security checkpoint, the chip and the e-passport reader execute the BAC protocol, in order to establish a secure (encrypted) communication channel between them..
2. On successful completion of BAC, the e-passport reader performs PA.
3. On successful completion of PA the chip and the e-passport reader execute the AA protocol.

The e-passport authentication heavily relies on PKI. We model only one level of certification hierarchy, up to the document signer and we assume that document signer public key is certified by its root (country signing authority) and, is valid and secure. This does not weaken the verification process of the e-passport protocol suite, but only indicates that the model assumes the “ideal” PKI

implementation. We also suppose that cryptographic primitives used in the system like hash functions, MAC, and generation of keys (3-DES) are secure. Details of our modelling of e-passport protocols using Casper are presented in Appendix A.

3.4 Interpreting FDR Output

Casper generates refinement assertions to check for all specifications. It generates one assertion for all secret specifications and one assertion for each agreement and aliveness specification. A CSP script file includes statements making assertions about refinement properties. These statements will typically have the following form:

```
assert Abstract [ X= Concrete
```

Example: `Specification Secret(B, message, [ A] )` specifies that, at the end of a protocol run, entity B expects the value of message to be known only to entity A. Assertion generated for the above specification is:

```
SECRET_M::SECRET_SPEC[ T=SECRET_M::SYSTEM_S
```

The selected assertion is submitted for testing by choosing the Run option in FDR2. FDR2 then attempts to prove the conjecture by compiling, normalising, and checking the refinement. If we find a refinement that is not satisfied, then there might be a weakness in the protocol. To examine the weakness, the FDR2 debugger is invoked, allowing the behaviour of the processes involved to be examined. The information displayed depends on the nature of the counterexample being examined and the contribution made to it by the selected component. The weakness in the protocol is examined by observing a trace leading to divergence.

4. SECURITY ANALYSIS

4.1 Verification of ICAO E-passport Protocols Using Casper/FDR

In e-passports, data confidentiality is provided by the BAC protocol, whereas, the integrity of chip contents of LDS and SOD is verified by the reader using the PA and AA protocols. The keys KENC and KMAC are stored in DF on e-passport and are generated by the reader using the data in MRZ, which includes the e-passport number, date of birth, e-passport validity date, and corresponding check digits. The ICAO e-passport guideline states that the entropy of the key is at most 56 bits. Juels *et al* (2005) show that the entropy of keys used in US e-passports can be reduced to around 52 bits because of specific assignment scheme (first two digits are assigned to 15 e-passport issuing offices) used to identify e-passport issuing offices. Low entropy of cryptographic keys makes them vulnerable to the exhaustive search attacks.

Analysis of the e-passport protocol using the Casper and FDR2 verification software proves that the protocol is vulnerable to the grandmaster chess attack (Desmedt *et al*, 1987) also known as the man-in-the-middle attack. Compiling with security specifications creates corresponding refinement assertions.

The secrecy specification results in an assertion

```
SECRET_M::SECRET_SPEC [ T= SECRET_M::SYSTEM_S
```

and its verification using FDR2 results in an erroneous trace after 30 states with 135 transitions. FDR2 debugger reveals:

```
send.Reader.Chip(Msg1.GETC,<>)
INTRUDER_M::say.GETC
send.Chip.Reader.(MSG2,RNDC2,<>)
INTRUDER_M::say.RNDC2
send.Reader.Chip(Msg3,Sq.<
Encrypt.(KEYE,<RNDR2,RNDC2,KR>),
Encrypt(KeyM,<RNDR2,RNDC2,KR>)>)
INTRUDER_M::say.Sq<
Encrypt.(KEYE,<RNDR2,RNDC2,KR>),
Encrypt(KEYM,<RNDR2,RNDC2,KR>)>
```

which can be intepreted as:

```
1.  Reader -> I_Chip : GETC
1a. I_Chip -> Chip : GETC
2.  Chip -> I_Chip : {RNDC2}
2a. I_Chip -> Reader : {RNDC2}
3.  Reader -> I_Chip :
{RNDR2, RNDC2, KR}{KEYE},
{RNDR2, RNDC2, KR}{KEYM}
3a. I_Chip -> Chip :
{RNDR2, RNDC2, KR}{KEYE},
{RNDR2, RNDC2, KR}{KEYM}
4.  Chip -> I_Chip :
{RNDR2, RNDC2, KC}{KEYE},
{RNDR2, RNDC2, KC}{KEYM}
4. I_Chip -> Reader :
{RNDR2, RNDC2, KC}{KEYE},
{RNDR2, RNDC2, KC}{KEYM}
```

and for assertion

```
AUTH1_M::AuthenticateRESPONDERTo INITIATORAliveness [T=
```

which corresponds to the belief of e-passport that it is involved in a conversation with the reader. Its verifications using FDR2 results in an erroneous trace after 12 states with 35 transitions. FDR2 debugger reveals:

```
send.Reader.Chip.(Msg1,GETC,<>)
INTRUDER_M::hear.GETC
send.Reader.Chip.(Msg3,Sq.<
Encrypt.(KEYE,<RNDR2,KM,KR>),
Encrypt.(KEYM,<RNDR2,KM,KR>)>,<>)
INTRUDER_M::hear.Sq.<
Encrypt.(KEYE,<RNDR2,KM,KR>),
Encrypt.(KEYM,<RNDR2,KM,KR>)>
INTRUDER_M::say.Sq.<
Encrypt.(KEYE,<RNDR2,KM,KR>),
Encrypt.(KEYM,<RNDR2,KM,KR>)>
```

which can be interpreted as:

```
1. Reader -> I_Chip : GETC
2. I_Chip -> Reader : KM
3. Reader -> I_Chip :
{RNDR2, KM, KR}{KEYE},
{RNDR2, KM, KR}{KEYM}
4. I_Chip -> Reader :
{RNDR2, KM, KR}{KEYE},
{RNDR2, KM, KR}{KEYM}
```

The trace from the security assertion can be interpreted as, the lack of mutual authentication. The reader establishes a session key even though it is not sure if a chip is genuine.

Can this weakness be exploited? Once a secure communication is established between reader and chip, the reader retrieves data stored within the LDS and performs an integrity verification using the issuing state certificate. A border security officer on receiving evidence that LDS has not been tampered with would authenticate an e-passport bearer by using the facial biometric image stored in LDS against the person physically present at the checkpoint. Therefore even if the messages are only being replayed the data still has to come from an issuing state certified chip. This weakness can be exploited as facial biometrics is view-dependent and are prone to inter-class similarities within large population groups such as identical twins, similar ethnic groups and certainly possible in case of human cloning. As the probability of uniqueness using facial biometric is low, it is certainly possible that a border security officer might not be able to differentiate between the facial biometric data in the LDS and the person physically present at the checkpoint. Phillips *et al* (2000) pointed out that the false rejection rate could be as high as 43%, as majority of algorithms used in facial biometrics are subject to illumination issues and also depend on the type of camera used to obtain the initial image. Note that e-passports store high-resolution images of the e-passport bearer to make verification independent on the processing algorithms used by various countries. This introduces another serious security weakness – an attacker can manipulate less significant bits of images to find collisions for the hash functions used.

Even with these drawbacks, BAC is important and should be implemented as it prevents against eavesdropping. The protocol is vulnerable to replay attacks but an intruder cannot decrypt values (E_C or E_R) used to form the session key (K_{seed}).

The AA protocol in addition to providing integrity also protects the e-passport against chip modification i.e, it binds LDS with the e-passport bearer's secret key ChipSK and authenticates the e-passport microchip. Our verification of an ideal AA protocol i.e., assuming that the BAC protocol was carried out in a secure way, indicates that there is no security weakness in the protocol.

Assertions

```
SECRET_M::SECRET_SPEC [ T= SECRET_M::SYSTEM_S
AUTH1_M::AuthenticateRESPONDERToINITIATOR
Aliveness [ T= AUTH1_M::SYSTEM_1
AUTH2_M::AuthenticateINITIATORToRESPONDER
Aliveness [ T= AUTH2_M::SYSTEM_2
AUTH3_M::AuthenticateINITIATORToRESPONDER
Agreement_rndr1 [ T= AUTH3_M::SYSTEM_3
AUTH4_M::AuthenticateRESPONDERToINITIATOR
Agreement_rndc1 [ T= AUTH4_M::SYSTEM_4
```

which corresponds to secrecy, authentication of an e-passport to reader and from reader to an e-passport does not yield any erroneous traces. But if we consider that an intruder was able to successfully run the BAC protocol with the reader by obtaining K_{ENC} and K_{MAC} by performing a brute force attack as in (Laurie, 2007) and thus successfully able to compute session key Kseed, then assertions:

```
SECRET_M::SECRET_SPEC [ T= SECRET_M::SYSTEM_S
AUTH2_M::AuthenticateINITIATORToRESPONDER
Aliveness [ T= AUTH2_M::SYSTEM_2
AUTH3_M::AuthenticateINITIATORToRESPONDER
Agreement_rndr1 [ T= AUTH3_M::SYSTEM_3
```

yields erroneous traces which indicates that weakness exists in the protocol.

Assertion

```
SECRET_M::SECRET_SPEC [ T= SECRET_M:: SYSTEM_S
```

yields an error trace after 4 states and 8 transitions and analysis using the FDR2 debugger reveals the following first level trace.

```
send.Reader.Chip.(Msg1,Encrypt.
(KEYCR,<RNDR1>),<RNDR1>)
leak.RNDR1
```

This attack is obviously true, as the intruder is now in possession of the session key and therefore able to decrypt any communication between the chip and the reader. This would compromise the privacy of an e-passport bearer as his/her personal details would be compromised and increase the risk of identity fraud.

Assertion

```
AUTH3_M::AuthenticateINITIATORTo
RESPONDERAgreement_rndr1[ T=AUTH3_M::SYSTEM_3
```

yields an erroneous trace after 8 states and 149 transitions. FDR2 debugger revels the following second level trace:

```
env.Chip.(Env0,Reader,<RNDC1,Reader>)
receive.Reader.Chip.(Msg1,
Encrypt.(KEYCR,<RNDM1>),<RNDM1>)
signal.Commit3.
RESPONDER_role.Chip.Reader.RNDM1
```

From the above traces we can interpret that an attacker is able to successfully authenticate to the reader as a genuine e-passport. This is possible because the session key is compromised. This attack is theoretically possible but practically would not be easy to implement, as the data is protected by digital signature and is computationally impossible to generate a valid signature for a modified data. Nevertheless the attacker in lieu can exploit this weakness with weakness in facial biometric systems as discussed above. The intruder can exploit the combination of weakness in both BAC and AA. An attacker can now make a copy of the e-passport and authenticate successfully, defeating the primary security goals of BAC and AA, to prevent against chip substitution and chip copy.

```
Assertion
AUTH2_M::AuthenticateINITIATORTo
RESPONDERAliveness[ T=AUTH2_M::SYSTEM_2
```

yields an error trace after 3 state and 6 transitions and the FDR2 debugger revels the following second level trace

```
env.Chip.(Env0,Reader,<RNDC1,Reader>)
receive.Reader.Chip.(Msg1,Encrypt.
(KEYCR,<RNDM1>),<RNDM1>)
signal.Commit2.RESPONDER_role.Chip.Reader
```

The above traces points to an important security goal that is not met: mutual authentication between a chip and a reader. The reader believes that it has successfully authenticated the chip but there is no proof that the chip has successfully authenticated the reader. Authentication of reader by the chip depends on the fact that only a genuine reader would be able to obtain KENC and KMAC from MRZ to perform BAC protocol and compute the session key K_{seed} used in AA protocol. We have seen that it is not necessarily true. An attacker who is in possession of the keys

KENC and KMAC (because of low entropy and brute force attacks as in (Laurie, 2007)) will be able to masquerade as a reader and successfully authenticate itself to the chip.

From the above traces it is also clear that the e-passport protocol does not satisfy any key related security goals like freshness and integrity. Key integrity is not satisfied as an attacker is able to successfully run the BAC protocol and obtain the session key K_{seed} used to form a secure communication channel. There are no guarantees provided to either the chip or the reader regarding key freshness. The nonce generated by either reader, chip or both may not contain enough randomness that is necessary for a security protocol. An eavesdropper might be able to collect information about several runs of the protocol and perform a cipher-text with known partial plain-text attack to obtain the session key and/or MRZ information that is necessary to create KENC and KMAC. This would also compromise the security goal of forward secrecy. An e-passport has an average validity of around 10 years. Any loss of KENC or KMAC keys makes the e-passport vulnerable to skimming and snooping attacks.

We were unable to make a formal analysis of non-repudiation and certificate manipulation, but an informal analysis of e-passport protocols suite reveals they may also be prone to PKI based attacks. Public key certificates (for both document signer and country signing certificates) are held by ICAO in a central repository. The ICAO e-passport guideline states that each border security checkpoint should update their certification hierarchy list individually. This is necessary to perform a valid verification during the AA protocol, as the issuing country certifies the secret key of an e-passport. The drawback is that an attacker may be able to mount a denial-of-service attack on the border security checkpoint certificate server before arriving or in co-ordination with others to prevent the certificate server from updating and thus preventing the border security checkpoint from verifying validity of e-passport signature, as the border security checkpoint now relies on CDS that is stored in the chip and will not have an updated revocation list. ICAO e-passport guideline acknowledges this issue and states that in such a case a border security checking officer should rely on conventional methods that were in place before e-passport for verification of the e-passport bearer. But this defeats the entire purpose of introducing e-passports.

4.2 Analysis of EU EAC Mechanism

EU proposal for EAC in e-passports intends to provide better security compared to the first generation e-passports and in this paper, we only make an informal security analysis. Nevertheless, even informally, we have identified that EU EAC proposal does not adequately protect an e-passport bearer and compared to first generation, induces new security problems.

The EAC proposal still relies on BAC to derive the initial session key needed to access e-passport bearer's details including their facial biometric. Because of the inherent weaknesses of BAC as previously described (e.g. keys that have insufficient entropy), the EAC proposal also suffers from the same weaknesses.

EAC proposal makes extensive use of PKI. Both chip and terminal authentication protocols requires verification of certificates that involve the entire certification hierarchy. The e-passport initially contains the root level certificate ($CERT_{CVCAH}$) that was written by its document verifier at the time of issue. As the e-passport chips are time-less devices, i.e. they do not have any internal clock, this makes them vulnerable to attacks using expired certificates. Kluger (Klugler, 2005a,b) acknowledges this vulnerability and proposed that the e-passport should write $CERT_{CVCAH}$ with the latest certificate it obtains when it performs a terminal authentication with a visiting country's IS. During the first run of terminal authentication the time of expiry of $CERT_{CVCAH}$ that was initially written is used as a reference time to validate visiting country's IS certificate and after a successful run of the protocol the e-passport will store the $CERT_{CVCAH}$ that is present in the certificate chain received from an IS. But, the protocol is still vulnerable to attacks using expired IS certificates. Validity of IS certificates are considerably shorter when compared to CVCA certificates. A compromised IS even if its certificate was expired would still be able to authenticate itself to an e-passport and obtain access to sensitive e-passport information including finger prints and iris scans, that were intended to be protected by EAC. The attack is more effective for infrequently used e-passports, because they have only the initially written $CERT_{CVCAH}$ which themselves may be expired. As the e-passport uses the time on $CERT_{CV\ CAH}$hi as a reference point, it would accept any certificate, as long as its validity is before the current reference time recorded on the e-passport.

The approach of sending certificate chains can also lead to a denial-of-service attack on an e-passport. Since an IS terminal is not authenticated during or before chip authentication, a malicious terminal could flood the chip by sending lots of public keys and certificates. Because of the limited memory that is available in an e-passport chip, the chip could run out of memory and essentially stop the chip from functioning in a desired manner.

The EAC proposal also has some new weaknesses. The e-passport should now have write access to the chip, to update its $CERT_{CVCAH}$. This could be used by an illegitimate e-passport bearer to update the chip with false information. The EAC proposal does not specify how write access would be controlled by the chip. Another drawback of EAC proposal is the cross certification among countries. Every country implementing EAC would be required to certificate other country's document verifiers. That essentially means that each document verifier that certifies IS will need to be certified by CSVA of every participating country. EAC recommends the validity of document verifier certificates be one third of CVCA certificate's validity period. This becomes an extremely complex undertaking for each country, with respect to certifying other participating country's document verifiers and maintenance of revocation lists. EAC also does not address grandmaster chess attack (Desmedt *et al*, 1987) to which the first generation passports were vulnerable. The BAC protocol is used only to form a session key for an encrypted communication channel between a chip and IS and does not provide authentication. Therefore the chip establishes a session key even though it is not sure if IS is genuine. EU EAC also does not provide any guarantees regarding freshness or origin of messages.

There are also concerns regarding privacy of the e-passport bearer. The chip sends its identification details (public key) during CA, even before it has authenticated the IS. Therefore, this would make very easy for an attacker to track an e-passport bearer, as an attacker is not required to authenticate to an e-passport before obtaining details from an e-passport.

5. CONCLUSION

Formal methods have become an integral part in verification of protocols. We have used the Casper and FDR model checker to verify security of Australian e-passport implementation that is based on ICAO e-passport protocol suite and our analysis have shown that current security measures that are in place are weak. Security techniques implemented in both the first and second generation of e-passports do not adequately protect an e-passport bearer. The first generation e-passport standard is vulnerable to brute force attacks because session keys generated have very low entropy. The second-generation e-passport proposal requires extensive modifications to exiting infrastructure and it still relies on the first generation standards to provide a secure connection to protect primary biometric identifiers. Both the standard have ignored the need to protect e-passports details during setting up a communication, which makes the e-passport bearer vulnerable to identity theft and covert surveillance.

Our formal analysis shows that ICAO e-passport guideline does not meet our security goals.

- The e-passport protocols does not satisfy our goal for data origin authentication as it can be subject to replay and grandmaster chess attacks, and the weakness can be exploited in cases where problems with facial biometric exists.
- Data confidentiality is also compromised when an attacker is able to obtain encryption and MAC keys stored in the e-passport chip using information presented in MRZ.
- We were able to prove that this further affects the security goals for active authentication protocol, namely, mutual authentication, key freshness and key integrity.
- An informal analysis of the e-passport system reveals that it may also be vulnerable to certificate manipulation, as they are dependent on PKI, which is prone to denial-of-service attacks.

Electronic passports are an important step in the right direction. It enables countries to digitize their security at border control and provides faster and safer processing of an e-passport bearer. E-passports introduce facial biometric recognition for verification of an e-passport bearer, which is less intrusive when compared with other biometric systems. But facial biometric are not very secure because of relatively low uniqueness and are prone to inter-class similarities. The risks of identity theft or illegal entries into a country are further increased when e-passports can be used as in Australian Customs Services (2006), that are currently on trial in Australia. Unattended border control check-ins increase the risk of fraudulent facial biometric verifications being undetected and eavesdropping on communication between e-passport and reader.

REFERENCES

AUSTRALIAN CUSTOMS SERVICES (2006): 'Smartgate', http://www.customs.gov.au/ site/page.cfm?u=5555.

DANG, Z. and KEMMERER, R.A. (1997): Using the Astral model checker for cryptographic protocols analysis, in ORMAN, H. and MEADOWS, C. eds, Workshop on design and formal verification of security protocols.

DESMEDT, Y., GOUTIER, C. and BENGIO, S. (1987): Special uses and abuses of the Fiat-Shamir passport protocol, in Advances in Cryptology - CRYPTO '87, Springer Berlin/Heidelberg, 293: 21–39.

FORMAL SYSTEMS (EUROPE) LTD (2003): Failures-divergence refinement, FDR2 User Manual. Available from http://www.fsel.com/.

HEINTZE, N. and TYGAR, J.D. (1994): A model for secure protocols and their compositions, in *1994 IEEE Computer Society Symposium on Research in Security and Privacy*, IEEE Computer Society Press, 2–13.

HOARE, C.A.R. (1985): Communicating sequential processes, Prentice Hall International.
HOME AFFAIRS JUSTICE (2006): EU standard specifications for security features and biometrics in passports and travel documents, Technical report, European Union.
ICAO (2006): Machine readable travel documents, Technical report, ICAO.
ISO/IEC (2000): ISO/IEC14443, identification cards – contactless integrated circuit(s) cards – proximity cards.
JOHNSON, B. (2006): Hackers crack new biometric passports, *The Guardian*.
JUELS, A., MOLNAR, D. and WAGNER, D. (2005): Security and privacy issues in e-passports, in *IEEE SecureComm '05*.
KC, G.S. and KARGER, P.A. (2005): Preventing attacks on Machine Readable Travel Documents (MRTDs), Cryptology ePrint Archive, Report 2005/404. http://eprint.iacr.org/.
KNIGHT, W. (2006): Hackers clone radio-chip passports, *NewScientist*.
KLUGLER, D. (2005a): Advance security mechanisms for machine readable travel documents, Technical report, Federal Office for Information Security (BSI), Germany.
KLUGLER, D. (2005b): Security concept of the EU-passport, *Security in Pervasive Computing* 85.
LAURIE, A. (2007): Rfidiot, http://rfidiot.org/.
LOWE, G. (1996): Breaking and fixing the Needham-Schroeder public-key protocol using CSP and FDR, in MARGARIA, T. and STEFFEN, B. eds, Tools and alogrithms for the construction and analysis of systems, Springer-Verlag, 1055: 147–166.
LOWE, G. (1999): Casper – A compiler for the analysis of security protocols, User Manual and Tutorial, Ver1.3.
LOWE, G. and ROASCOE, B. (1997): Using CSP to detect errors in the TMN protocols, in *IEEE Transactions on Software Engineering*, 3.
MITCHELL, J.C., MITCHELL, M. and STERN, U. (1997): Automated analysis of cryptographic protocols using murphi, in *16th IEEE Symposium on Security and Privacy*, IEEE Computer Society Press.
PASUPATHINATHAN, V., PIEPRZYK, J., WANG, H. and CHO, J.Y. (2006): Formal analysis of card-based payment systems in mobile devices, in SAFAVI-NAINI, R., STEKETEE, C. and SUSILO, W. eds, *Fourth Australasian Information Security Workshop (Network Security) (AISW 2006)*, Hobart, Australia, 54: 213–220.
PHILLIPS, P.J., MARTIN, A., WILSON, C.L. and PRZYBOCKI, M. (2000): An introduction evaluating biometric systems, *IEEE Computer* 33(2): 56–63.
REID, D. (2006): E-passports at risk from cloning, BBC.
SCHERZER, H., CANETTI, R., KARGER, P.A., KRAWCZYK, H., RABIN, T. and TOLL, D.C. (2003): Authenticating mandatory access controls and preserving privacy for a high-assurance smart card, in *8th European Symposium on Research in Computer Security (ESORICS 2003)*, Lecture Notes in Computer Science, Springer-Verlag, Gjovik, Norway, 2808: 181–200.
SCHNEIDER, S. (1997): Verifying authentication protocols with CSP, in *10th IEEE Computer Security Foundations Workshop*, IEEE Computer Society Press, 2–17.
TOM A. F. KINNEGING for ICAO-NTWG, P. T. F. (2004): PKI for machine readable travel documents offering ICC read-only access, Technical report. Version 1.1.

APPENDIX A CASPER REPRESENTATION

The Casper script provided below, presents a combined representation of all three protocols and does not represent modifications that are needed when verifying security properties for individual protocols.

```
#Free variables
C,R,DS : Agent
getc : InitializeConv
lds : DataGroups
sod : SecurityObject
rndr2,rndc2,kr,kc,rndr1,rndc1 : Nonce
h : HashFunction
PK : Agent -> PublicKey
SK : Agent -> SecretKey
keyM,keyE,keyCR : SessionKey
InverseKeys = (PK,SK), (keyM,keyM), (keyE,keyE),
(keyCR,keyCR)
#Processes
```

```
INITIATOR(R,C,getc,rndr1,rndr2,kr,keyM,keyE,keyCR)
knows PK,SK(R)
RESPONDER(C,R,rndc1,rndc2,kc,keyM,keyE,keyCR)
knows PK,SK(C)
#Protocol description
0. -> C : R
0a. DS -> C : {C,PK(C)}{SK(DS)} % CERTC
0b. DS -> R : {C,PK(C)}{SK(DS)}
1. R -> C : getc
2. C -> R : rndc2
3. R -> C : {rndr2,rndc2,kr}{keyE},
{rndr2,rndc2,kr}{keyM}
4. C -> R : {rndr2,rndc2,kc}{keyE},
{rndr2,rndc2,kc}{keyM}
---
5. C - > R : {LDS,SOD}{KeyCR},
{C,PK(C)}{SK(DS)} % CERTC
---
6. R -> C : {rndr1}{keyCR}
7. C -> R : { {h(rndc1,rndr1), rndr1,rndc1}
{SK(C)} }{keyCR}
#Specification
StrongSecret(C,kr,[R])
StrongSecret(C,kc,[R])
StrongSecret(R,kr,[C])
StrongSecret(R,kc,[C])
Aliveness(C,R)
Aliveness(R,C)
Agreement(C,R,[kr,kc])
StrongSecret(C,rndr1,[R])
#Actual variables
Chip,Reader,DSigner,Mallory : Agent
GETC : InitializeConv
LDS : DataGroups
SOD : SecurityObject
RNDR2,RNDC2,RNDM2,KR,KC,KM,RNDR1,RNDC1 : Nonce
KEYM,KEYE,KEYCR, KEYMM,KEYEM : SessionKey
InverseKeys = (KEYM,KEYM), (KEYE,KEYE),
(KEYMM,KEYMM), (KEYEM,KEYEM), (KEYCR,KEYCR)
#Functions
symbolic PK,SK
#System
INITIATOR(Reader,Chip,GETC,RNDR1,RNDR2,KR,
KEYM,KEYE,KEYCR)
RESPONDER(Chip,Reader,RNDC1,RNDC2,KC,
KEYM,KEYE,KEYCR)
```

```
CERTAUTH(DS,C,R) knows PK,SK(DS)
#Intruder Information
Intruder = Mallory
IntruderKnowledge = {Chip,Reader,RNDM2,KM,PK,
SK(Mallory),KEYMM,KEYEM}
```

BIOGRAPHICAL NOTES

Vijayakrishnan Pasupathinathan

Vijayakrishnan Pasupathinathan is a PhD Candidate with the Department of Computing Science at Macquarie University, Australia. His research interests are in the area of cryptographic protocols focusing on e-commerce, electronic identification systems and e-governance, and the use of formal methods in protocol verification. His PhD research encompasses the analysis and design of a class of protocols called functional cryptographic protocols. The motivating factor for the research is the need to analyse and design crypto-logic protocols that have a practical significance.

Josef Pieprzyk

Josef Pieprzyk is a Professor with the Department of Computing at Macquarie University, Australia. His research interest includes analysis and design of block ciphers, stream ciphers, public key cryptography and application of cryptographic in areas of database security, copyright protection, e-commerce and e-governance.

Huaxiang Wang

Huaxiang Wang is an Associate Professor with the school of Physical and Mathematical Sciences at Nanyang Technological University. His research interest also includes the field of combinatorics, coding theory, information security and cryptography.