Comput Syst Sci & Eng (2003) 1: 17–25

International Journal of
Computer Systems
Science & Engineering

Formal verification logic for hybrid security protocols

T Newe and T Coffey

Dept of Electronic and Computer Engineering, University of Limerick, Ireland
Emails: Thomas.Newe@UL.ie, Tom.Coffey@UL.ie

Hybrid cryptographic security protocols find applications in many areas of communications, none more demanding than in the mobile security sector. In recent years a number of hybrid cryptographic security protocols have been proposed for use with 3G mobile systems. These include the ASPeCT [1] and Boyd-Park [2] security protocols. These protocols use a public key algorithm to exchange a secret session key for use with a symmetric algorithm, thereby removing the need for ultra reliable key servers. In order to provide assurance that these protocols are verifiably secure and trustworthy it is necessary to perform a formal verification on their design specifications. In this paper a formal modal logic is presented to enable hybrid cryptographic security protocols to be formally verified. This logic is based on the Coffey-Saidha logic [3], which was designed to verify public key based protocols only. The logic presented here is a substantial extension to the original. The new extended logic is used to formally verify the well-documented Needham-Schroeder key-distribution protocol in order to demonstrate the effectiveness and usefulness of the logic in the protocol verification arena. The Boyd-Park hybrid mobile security protocol is then formally specified and verified using this logic, and as a result of this verification a modification to the original protocol is suggested.

Keywords: Formal methods, verification logic, hybrid security protocol, 3G mobile

1. INTRODUCTION

Security communication protocols generally start with an authentication phase where the identities of the involved parties are established. Next a session key is generated for use with some symmetric cryptographic algorithm to secure the actual transmission. In addition to the above, a mobile system may also require location privacy, link-security and/or end-to-end security. These extra requirements have been incorporated into the ASPeCT and Boyd-Park 3G security protocols. While these protocols are designed for use with the third generation UMTS mobile services their trustworthiness and security have yet to be formally verified. To provide this formal verification a modal logic can be used.

Many of the current modal logical techniques are based on logics of belief [4, 5], which are considered useful in evaluating trust which may be placed in a security protocol. Other techniques are based on logics of knowledge [6,7], which are suitable for proving protocol security. If trust and security are both to be evaluated then a logic that combines the logics of knowledge and belief is required. One such formal logic is the logic of Coffey-Saidha.

In this paper the Coffey-Saidha logic is expanded to include symmetric cryptographic protocol verification capabilities. The necessary language and rule set is provided enabling the verification of hybrid security protocols. A formal verification of the well-known Needham-Schroeder key-distribution protocol demonstrates an application of the expanded logic. This shows that the logic can detect a flaw in the protocol, which leaves it susceptible to a replay attack. The new logic is then used to formally verify the Boyd-Park hybrid mobile security protocol. As a result of this verification a modification to the original protocol is suggested and the modified version is verified.

2. FORMAL LOGIC FOR HYBRID CRYPTOGRAPHIC SECURITY PROTOCOL ANALYSIS

Traditionally, cryptographic protocols have been designed and verified using informal and intuitive techniques. However, an absence of formal verification has been proven [3,4] to lead to flaws and security errors remaining undetected in a protocol. Formal verification aims at providing a rigid and thorough means of testing the correctness of a cryptographic protocol so that even subtle defects can be uncovered. A number of formal techniques have been developed for this purpose. This section discusses the Coffey-Saidha logical technique and proposes language extensions, and additional axioms to enable it to verify hybrid security protocols.

2.1 The Coffey–Saidha formal logic

This logic provides a means of verifying public-key cryptographic protocols. The logic can analyse the evolution of both knowledge and belief during a protocol execution and is therefore useful in addressing issues of both security and trust. A belief operator and two knowledge operators is provided by the logic. One knowledge operator is propositional and deals with the knowledge of statements or facts. The other knowledge operator is a predicate and deals with the knowledge of objects (e.g. cryptographic keys, ciphertext data, etc.). The inference rules provided are the standard inferences required for natural deduction and the axioms of the logic are sufficiently low-level to express the fundamental properties of public-key cryptographic protocols, such as the ability of a principal to encrypt/decrypt based on knowledge of a cryptographic key.

These axioms reflect the underlying assumptions of the logic, which are as follows:

- The communication environment is hostile. That is, the data communication system itself is assumed to be reliable so that message loss and transmission errors cannot occur without some interference from a hostile party.
- The public-key cryptosystem is ideal. That is, the encryption and decryption functions are completely non-invertible without knowledge of the appropriate cryptographic key and are invertible with knowledge of the appropriate cryptographic key.
- A public key used by the system is considered valid if it has not exceeded its validity period and only its rightful owner knows the corresponding secret key.
- If a piece of data is encrypted/decrypted, then the entity which performed the encryption/decryption must know that data.

This logic is capable of analysing a wide variety of public-key cryptographic protocols because the constructs of the logic are general purpose and therefore provide the user with increased flexibility allowing him to develop his own theorems.

2.1.1 The logic language

The language of a logic is used to formally express the logical postulates and protocol facts. The language for the Coffey-Saidha logic is as follows:

- **a,b,c**,..., general propositional variables
- Φ, an arbitrary statement
- Σ and Ψ, arbitrary entities
- **i** and **j**, range over entities
- **ENT**, the set of all possible entities
- **k**, a cryptographic key. In particular, k_{Σ} is the public key of entity Σ and k_{Σ}^{-1} is the corresponding private key of entity Σ
- **t, t', t''**... time
- **e(x,k$_{\Sigma}$)**, encryption function, encryption of *x* using key kS
- **d(x,k$_{\Sigma}^{-1}$)**, decryption function, decryption of *x* using key k_{Σ}^{-1}
- ***K***, propositional knowledge operator of Hintikka. $K_{\Sigma,t}\Phi$ means Σ knows statement Φ at time *t*.
- **L**, knowledge predicate. $L_{\Sigma,t}x$ means Σ knows and can reproduce object *x* at time *t*.
- **B**, belief operator. $B_{\Sigma,t}\Phi$ means Σ believes at time *t* that statement Φ is true.
- **C**, 'contains' operator. *C*(x,y) means that the object *x* contains the object *y*. The object *y* may be cleartext or ciphertext in *x*.
- **S**, emission operator. *S*(Σ,t,x) means Σ sends message *x* at time *t*.
- **R**, reception operator. *R*(Σ,t,x) means Σ receives message *x* at time *t*.

The language includes the classical logical connectives of conjunction ( ), disjunction ( ), complementation (¬) and material implication (→). The symbols and denote universal and existential quantification respectively. indicates membership of a set and / denotes set exclusion. The symbol |– denotes a logical theorem. The logic does not contain specific temporal operators, but the knowledge, belief and message transfer operators are time-indexed.

2.1.2 Inference rules

The logic incorporates the following rules of inference:

R1: From |– p and |– (p → q) infer |– q
R2: (a) From |– p infer |– $K_{\Sigma,t}$p
(b) From |– p infer |– $B_{\Sigma,t}$p

R1 is the *Modus Ponen*s and states that if schema *p* can be deduced and (p → q) can be deduced, then *q* can also be deduced. R2 consists of the Generalisation rules which state that if p is a theorem, then knowledge and belief in p are also theorems. The logic also includes the following standard propositional rules of natural deduction:

R3: From (p q) infer p
R4: From p and q infer (p q)
R5: From p infer (p q)
R6: From ¬(¬p) infer p
R7: From (from p infer q) infer (p → q)

2.1.3 Axioms

Two types of axioms are used in this logic, *logical* and *non-*

logical. Logical axioms are general statements made in relation to any system, while non-logical are system specific, in this case they apply to public-key systems.

2.1.3.1 Logical axioms

The logic includes the following standard modal axioms for knowledge and belief:

A1: **(a)** $t\ \ p\ \ q(K_{\Sigma,t}p \quad K_{\Sigma,t}(p \to q) \to K_{\Sigma,t}q)$
(b) $t\ \ p\ \ q(B_{\Sigma,t}p \quad B_{\Sigma,t}(p \to q) \to B_{\Sigma,t}q)$
A2: $t\ \ p(K_{\Sigma,t}p \to p)$

The axioms A1(a) and A1(b) are applications of the *Modus Ponens* to the knowledge and belief operators. The axiom A2 is called the knowledge axiom and is said to logically characterise knowledge. If something is known, then it is true. This property distinguishes between knowledge and belief.

A3: **(a)** $t\ \ x\ \ i,i\ \ \{ENT\}(L_{i,t}x \ \to \ t',t' \geq t \ \ L_{i,t'}x)$
(b) $t\ \ x\ \ i,i\ \ \{ENT\}(K_{i,t}x \ \to \ t',t' \geq t \ \ K_{i,t'}x)$
(c) $t\ \ x\ \ i,i\ \ \{ENT\}(B_{i,t}x \ \to \ t',t' \geq t \ \ B_{i,t'}x)$

Axioms A3(a), A3(b) and A3(c) relate to the monotonicity of knowledge and belief, which asserts that knowledge and belief, once gained, cannot be lost.

A4: $t\ \ x\ \ y(\ \ i,i\ \ \{ENT\}L_{i,t}y \quad C(y,x) \to$
$j,j\ \ \{ENT\}L_{j,t}x)$

The final logical axiom is concerned with the *C* predicate. If a piece of data is constructed from other pieces of data, then each piece of data involved in the construction must be known to some entity:

2.1.3.2 Non-logical axioms

The non-logical axioms (A5–A10) reflect the underlying assumptions of the logic. These assumptions relate to the emission and reception of messages and to the use of encryption and decryption in these messages.

A5: $t\ \ x\ (S(\Sigma,t,x) \to L_{\Sigma,t}x \quad i,i\ \ \{ENT/\Sigma\}\ \ t',t'$
$> t\ R(i,t',x))$

The emission axiom (A5) states that: if Σ sends a message *x* at time *t*, then Σ knows *x* at time *t* and some entity *i* other than Σ will receive *x* at time *t'* subsequent to *t*.

A6: $t\ \ x\ (R(\Sigma,t,x) \to L_{\Sigma,t}x \quad i,i\ \ \{ENT/\Sigma\}\ \ t',t'$
$< t\ S(i,t',x))$

The reception axiom (A6) states that: if Σ receives a message *x* at time *t*, then Σ knows *x* at time *t* and some entity *i* other than Σ has sent *x* at time *t'* prior to *t*.

A7: **(a)** $t\ \ x\ \ i,i\ \ \{ENT\}$
$(L_{i,t}x \quad L_{i,t}k_{\Sigma} \to L_{i,t}(e(x,k_{\Sigma}))\ )$
(b) $t\ \ x\ \ i,i\ \ \{ENT\}$
$(L_{i,t}x \quad L_{i,t}k_{\Sigma}^{-1} \to L_{i,t}(d(x,k_{\Sigma}^{-1})))$

Axioms A7(a) and A7(b) refer to the ability of an entity to encrypt or decrypt a message when it has knowledge of a public or private cryptographic key.

A8: **(a)** $t\ \ x\ \ i,i\ \ \{ENT\}$
$(\neg L_{i,t}k_{\Sigma} \quad t',t' < t\ \neg L_{i,t'}(e(x,k_{\Sigma}))$
$\neg(\ \ y(R(i,t,y) \quad C(y,e(x,k_{\Sigma}))\ ))$
$\to \neg L_{i,t}(e(x,k_{\Sigma})))$

(b) $t\ \ x\ \ i,i\ \ \{ENT\}$
$(\neg L_{i,t}k_{\Sigma}^{-1} \quad t',t' < t\ \neg\ L_{i,t'}(d(x,k_{\Sigma}^{-1}))$
$\neg(\ \ y(\ R(i,t,y) \quad C(y,d(x,k_{\Sigma}^{-1}))\ ))$
$\to \neg L_{i,t}(d(x,k_{\Sigma}^{-1})))$

Axioms A8(a) and A8(b) refer to the impossibility of encrypting or decrypting a message without knowledge of the correct key. Axiom A8a states that if an entity does not know k_{Σ} at *t* and does not know, prior to *t*, the encryption e(x,k_{Σ}) and also does not receive e(x,k_{Σ}) at *t* in a message, then the entity cannot know e(x,k_{Σ}) at time *t*. Axiom A8b makes a similar statement for the decryption of a message *x* without knowledge of the decryption key.

A9: $t(\ \ i,i\ \ \{ENT\}L_{i,t}k_i^{-1} \quad j,j\ \ \{ENT/i\}\neg L_{j,t}k_i^{-1})$

The key secrecy axiom (A9) states that the private keys used by the system are known only to their rightful owners.

A10: $t\ \ x(\ \ i,\ i\ \ \{ENT\}L_{i,t}d(x,k_{\Sigma}^{-1}) \to L_{\Sigma,t}x)$

Axiom A10 refers to the fact that a private key owner must know any data which has been decrypted using that private key.

2.2 Language additions

In this section, additions to the language of the Coffey-Saidha logic, which are necessary to enable new axioms to be created, that express the properties of symmetric key protocols are presented.

- **ks**$_{(\Sigma,\Psi)}$ Shared secret key for entities Σ and Ψ.
- **KS**$_{(\Sigma,\Psi)}$ Set of good shared keys for entities Σ and Ψ.
- **ss**$_{(\Sigma,\Psi)}$ Shared secret for entities Σ and Ψ (secret can be fresh).
- **SS{**$_{\Sigma,\Psi}$**}** Set of good shared secrets for entities Σ and Ψ.

The following language additions are the functions necessary to represent encryption using a symmetric key system.

- **E(x, ks**$_{(\Sigma,\Psi)}$**)**, Encryption of plaintext message *x* using the shared secret key of entities Σ and Ψ.
- **D(x, ks**$_{(\Sigma,\Psi)}$**)**, Decryption of ciphertext message *x* using the shared secret key of entities Σ and Ψ.

A new operator is now defined which is called the authentication operator.

- **A**, authentication Operator. **A(∑,t,Ψ)** means that Σ authenticates Ψ at time *t*.

2.3 Additional axioms

As previously mentioned the Coffey-Saidha logic is capable of analysing a wide variety of public key cryptographic protocols. In this section additional axioms are presented for inclusion in the logic, which also enables symmetric key protocols to be analysed.

2.3.1 Axiom 11

Axiom 11 presented here refers to the ability an entity has to encrypt or decrypt a message using a symmetric system, when it has knowledge of a secret key.

A11: (a)

$$t \quad x \quad i,i \quad \{ENT\}$$
$$(L_{i,t}x \quad L_{i,t}ks_{(\Sigma,\Psi)} \rightarrow L_{i,t}(E(x,\, ks_{(\Sigma,\Psi)})))$$

(b)

$$t \quad x \quad i,i \quad \{ENT\}$$
$$(L_{i,t}x \quad L_{i,t}ks_{(\Sigma,\Psi)} \rightarrow L_{i,t}(D(x,\, ks_{(\Sigma,\Psi)})))$$

This axiom states that:

a. if some entity *i* knows and can reproduce *x* at time *t* and *i* knows and can reproduce the shared secret key of entities Σ and Ψ at time *t* then *i* can encrypt *x* using the shared secret key of Σ and Ψ at time *t*.
b. if some entity *i* knows and can reproduce *x* at time *t* and *i* knows and can reproduce the shared secret key of entity Σ and Ψ at time *t* then *i* can decrypt *x* using the shared secret key of Σ and Ψ at time *t*.

2.3.2 Axiom 12

Axiom 12 refers to the inability of an entity to encrypt or decrypt data without knowledge of the appropriate shared secret key.

A12: (a)

$$t \quad x \quad i,i \quad \{ENT\}(\neg L_{i,t}\, ks_{(\Sigma,\Psi)} \qquad t',t'$$
$$< t \neg\, L_{i,t'}(E(x,\, ks_{(\Sigma,\Psi)})) \quad \neg(\quad y(R(i,t,y)$$
$$C(y,E(x,\, ks_{(\Sigma,\Psi)})))) \rightarrow \neg L_{i,t}(E(x,\, ks_{(\Sigma,\Psi)})))$$

(b)

$$t \quad x \quad i,i \quad \{ENT\}(\neg L_{i,t}\, ks_{(\Sigma,\Psi)} \qquad t',t'$$
$$< t \neg\, L_{i,t'}(D(x,\, ks_{(\Sigma,\Psi)})) \quad \neg(\quad y(R(i,t,y)$$
$$C(y,D(x,ks_{(\Sigma,\Psi)})))) \rightarrow \neg L_{i,t}(D(x,\, ks_{(\Sigma,\Psi)})))$$

This axiom states that:

a. if entity *i* does not know $ks_{(\Sigma,\Psi)}$ at *t* and does not know prior to *t* the encryption $E(x,\, ks_{(\Sigma,\Psi)})$ and also does not receive a message containing $E(x,\, ks_{(\Sigma,\Psi)})$ at *t*, then *i* does not know $E(x,\, ks_{(\Sigma,\Psi)})$ at *t*.
b. Same as (a) except this refers to the decryption of a message *x*.

2.3.3 Axiom 13

Axiom 13 states that only the rightful owners of a shared secret key know that key. This implies that this key is a good key.

A13:

$$t((\quad i,i \quad \{ENT/\Sigma,\Psi\}\neg L_{i,t}ks_{(\Sigma,\Psi)}$$
$$j,j \quad \{\Sigma,\Psi\}L_{j,t}ks_{(\Sigma,\Psi)})$$
$$\rightarrow ks_{(\Sigma,\Psi)} \quad \{KS_{\{\Sigma,\Psi\}}\}))$$

2.3.4 Axiom 14

Axiom 14 states that only the rightful owners of a shared secret know that secret. This implies that this is a good secret.

A14:

$$t((\quad i,i \quad \{ENT/\Sigma,\Psi\} \neg L_{i,t}ss_{(\Sigma,\Psi)}$$
$$j,j \quad \{\Sigma,\Psi\}L_{j,t}ss_{(\Sigma,\Psi)})$$
$$\rightarrow ss_{(\Sigma,\Psi)} \quad \{SS_{\{\Sigma,\Psi\}}\}))$$

2.3.5 Axiom 15

Axiom 15 is described as the authentication axiom, and it comes in symmetric and asymmetric form.

A15: (a)

$$x \quad t(A(\Sigma,t,\Psi) \rightarrow (L_{\Sigma,t}ss_{(\Sigma,\Psi)}$$
$$ss_{(\Sigma,\Psi)} \quad \{SS_{\{\Sigma,\Psi\}}\} \quad R(\Sigma,t,x)$$
$$C(x,ss_{(\Sigma,\Psi)}) \qquad t',t' < t \neg S(\Sigma,t',x))$$
$$\rightarrow K_{\Sigma,t}(S(\Psi,t',x))))$$

(b)

$$x \quad t(A(\Sigma,t,\Psi) \rightarrow (L_{\Sigma,t}k_{\Psi} \quad L_{\Sigma,t}x \quad R(\Sigma,t,y)$$
$$C(y,e(x,\, k_{\Psi}{}^{-1})))$$
$$\rightarrow (\quad t',\, t' < t,\, K_{\Sigma,t}(S(\Psi,t',y))))$$

A15 (a) states: If Σ knows a secret $ss_{(\Sigma,\Psi)}$ that it shares with Ψ (the secret can be fresh), and this secret is a good secret, and Σ receives a message containing $ss_{(\Sigma,\Psi)}$ at *t* that it did not send, then Σ knows that Ψ sent this message prior to *t*.

A15 (b) states: If Σ knows the public key of Ψ (k_{Ψ}) and message *x*, and if Σ receives a message y containing $e(x,\, k_{\Psi}{}^{-1})$ then Σ knows that Ψ sent message *y* prior to *t*.

3. TESTING OF THE PROPOSED LOGIC

The Needham–Schroeder key distribution protocol is a landmark protocol and many modern day security protocols are based on its design. The Needham–Schroeder protocol however, has a serious weakness in its design. It is not capable of detecting a previously used session key, thereby leaving itself open to a replay attack, where an old compromised session key can be used by an intruder. The BAN logic [4] was the first formal logic to highlight this flaw, and since then the flaw has been used to verify the operation of a number of other formal logics. (Note: In the following analysis tn refers to the time at the end of step *n* of the protocol.) See Figure 1.

3.1 Goals of the Needham–Schroeder protocol

The goals of the protocol are defined in the logic, as follows:

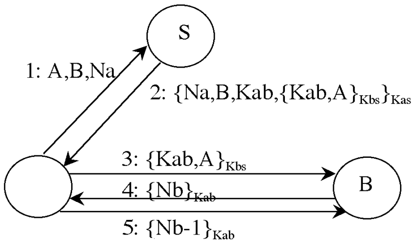

Figure 1 Needham-Schroeder key distribution protocol.

Legend:

S:	Trusted Authentication Server
A, B:	ID of Communicating Entities
Na, Nb:	Random Nonces
Kab:	Secret Session Key
Kas:	Shared secret key between A and S
Kbs:	Shared secret key between B and S

Goal 1: $K_{A,t2}(\ \ t, t < t2, S(S, t, E(X,K_{as}))$
$C(X, (K_{ab}\ \ \ E((K_{ab},A),K_{bs})))$

Goal 2: $K_{B,t3}(\ \ t, t0 < t < t3, S(A, t, E(X,K_{bs}))$
$C(X, K_{ab}))$

Goal 3: $K_{A,t4}(\ \ t, t0 < t < t4, S(B, t, E(N_b, K_{ab})))$

Goal 4: $K_{B,t5}(\ \ t, t0 < t < t5, S(A, t, E(N_b\text{-}1, K_{ab})))$

Goal 1 states that *A* knows it will obtain a message *X* from the trusted server *S* prior to the end of step 2. This message *X* will contain the session key K_{ab} and a secure message $E((K_{ab}, A), K_{bs})$, to be forwarded to *B*.

Goal 2 states that entity *B* knows it will obtain a secure message *X* from entity *A* prior to the end of step 3. This message *X* contains the fresh session key K_{ab}.

Goal 3 states that *A* knows it will receive a random nonce N_b from *B* prior to the end of step 4, and that this nonce is fresh for this run of the protocol.

Goal 4 states that *B* knows it will receive a secure message from *A*, and this message will contain N_{b-1}. Thereby proving the identity of *A* to *B*.

3.2 Initial assumptions of the Needham–Schroeder protocol

1: $\ \ i, i\ \ \{A,S\}L_{i,t0}K_{as}\ \ \ K_{as}\ \ \ KS_{\{A,S\}}$
2: $\ \ i, i\ \ \{B,S\}L_{i,t0}K_{bs}\ \ \ K_{bs}\ \ \ KS_{\{B,S\}}$
3: $K_{S,t0}(\ \ i, i\ \ \{ENT/S\},\ \ t, t < t1, \neg L_{i,t}K_{ab})\ \ \ K_{ab}\ \ \ KS_{\{A,B\}}$
4: $K_{A,t0}(\ \ i, i\ \ \{ENT/A\},\ \ t, t < t1, \neg L_{i,t}N_a)$
5: $K_{B,t0}(\ \ i, i\ \ \{ENT/B\},\ \ t, t < t4, \neg L_{i,t}N_b)$
6: $B_{B,t0}(\ \ i, i\ \ \{ENT\},\ \ t, t < t0, \neg L_{i,t}K_{ab})$

Assumptions 1 and 2 state that the shared keys K_{as} and K_{bs} are good keys and known only to their respective owners. Assumption 3 states that only the trusted server knows the session key K_{ab} prior to step 1 of the protocol, and this key is a good key.

Assumptions 4 and 5 state that the random nonces N_a and N_b are known only to their originating entity prior to their transmission. Indicating their freshness.

Assumption 6 states that entity *B* believes that the shared key K_{ab} is fresh. This is belief rather than knowledge, as *B* has no proof of freshness since time stamping is not employed.

3.3 Message exchanges of the Needham–Schroeder protocol

The following message are exchanged:

Step 1: A → S: A,B,N_a
Step 2: A ← S: $\{N_a,B,K_{ab},\{K_{ab},A\}K_{bs}\}K_{as}$
Step 3: A → B: $\{K_{ab},A\}K_{bs}$
Step 4: A ← B: $\{N_b\}K_{ab}$
Step 5: A → B: $\{N_b\text{-}1\}K_{ab}$

Rewriting each of the above protocol steps in the language of the logic using the new axioms and language additions, we get: For the purpose of this analysis step 1 is skipped as only a plaintext, or unprotected message, is exchanged between entities *A* and *S*.

Step 2:
$K_{A,t2}\ (R(A, t2, E(X,K_{as}))\ \ \ C(X, (K_{ab}\ \ \ E((K_{ab}, A),K_{bs})))$
This states that *A* knows at time t2 that it will receive a secure message *X*. This message will contain the session key K_{ab} and a secure message to be forwarded to entity *B*.

Using Axiom A2: $R(A, t2, E(X,K_{as}))\ \ \ C(X, (K_{ab}$
$E((K_{ab}, A),K_{bs}))$

Apply Axiom A6 and Inference Rule R2:

$L_{A,t2}E(X, K_{as})\ \ \ K_{A,t2}(\ \ i, i\ \ \{ENT/A\},\ \ \ t, t$
$< t2, S(i, t, E(X, K_{as})))$
$C(X, (K_{ab}\ \ \ E((K_{ab}, A), K_{bs}))$

Applying Inference Rule R3:

$K_{A,t2}(\ \ i, i\ \ \{ENT/A\},\ \ \ t, t < t2, S(i, t, E(X,K_{as})))$
$C(X, (K_{ab}\ \ \ E((K_{ab}, A),K_{bs}))$

Applying Axioms A11/A12/A13 and Assumption 1:

> Axioms A11, A12 and A13 establish the fact that only the rightful owners of a good secret key know and can use that key. Assumption 1 states that the key K_{as} is shared by *A* and *S*, this establishes the ID of *S*.

$K_{A,t2}(\ \ t, t < t2, S(S, t, E(X, K_{as})))$
$C(X, (K_{ab}\ \ \ E((K_{ab}, A),K_{bs}))$ **:Satisfies Goal 1**

Step 3:
$K_{B,t3}\ (R(B, t3, E(X,K_{bs}))\ \ \ C(X, (K_{ab}, A)))$
Using Axiom A2: $R(B, t3, E(X,K_{bs}))\ \ \ C(X, (K_{ab}, A))$

Applying Axiom A6 and Inference Rule R2:
$L_{B,t3}E(X,K_{bs})\ \ \ K_{B,t3}(\ \ i, i\ \ \{ENT/B\},\ \ \ t, t$
$< t3, S(i,t,E(X, K_{bs})))\ \ \ C(X,(K_{ab},A))$

Applying Inference Rule R3:
$K_{B,t3}(\ \ i, i\ \ \{ENT/B\},\ \ \ t, t < t3, S(i,t,E(X, K_{bs})))$
$C(X,(K_{ab},A))$

Applying Axioms A11/A12/A13 and Assumption 2 as previously:
$K_{B,t3}(\ \ t, t < t3, S(S,t,E(X, K_{bs})))\ \ \ C(X,(K_{ab},A))$

Given that A’s ID is included in the secure message X we can state that:
$K_{B,t3}(\ \ t, t < t3, S(A,t,E(X, K_{bs})))\ \ \ C(X,K_{ab})$

Using Assumption 6 which reflects belief rather than knowledge about the shared key K_{ab} we get:
$B_{B,t3}(\ \ t, t0 < t < t3, S(A,t,E(X, K_{bs})))\ \ \ C(X,K_{ab})$

This states that Goal 2 of the protocol is not known but believed, indicating that the session key is only believed to be fresh. This is a well-known flaw in the Needham-Schroeder key distribution protocol.

Step 4:
$K_{A,t4}\ (R(A, t4, X)\ \ \ C(X, E(N_b, K_{ab})))$

Using Axiom A2: $R(\mathrm{A, t4, X}) \quad C(\mathrm{X, E(N_b, K_{ab})})$

Applying Axiom A6 and Inference Rule R2:
$L_{\mathrm{A,t4}}\mathrm{X} \quad K_{\mathrm{A,t4}}(\ \mathrm{i, i} \quad \{\mathrm{ENT/A}\}, \quad \mathrm{t, t < t4}, S(\mathrm{i, t, X}))$
$C(\mathrm{X, E(Nb, K_{ab})})$

Applying Inference Rule R3:
$K_{\mathrm{A,t4}}(\ \mathrm{i, i} \quad \{\mathrm{ENT/A}\}, \quad \mathrm{t, t < t4}, S(\mathrm{i, t, X}))$
$C(\mathrm{X, E(N_b, K_{ab})})$

Applying Axioms A11/A12/A13 and Assumption 3:
Axioms A11/A12/A13 establish that only the key owners can know and use the secret key. Assumption 3 tells us that key $\mathrm{K_{ab}}$ is a good secret key and known only to entities S, A and B.
$K_{\mathrm{A,t4}}(\ \mathrm{t, t < t4}, S(\mathrm{B, t, X})) \quad C(\mathrm{X, E(N_b, K_{ab})})$

Using Assumption 5 to give the freshness of $\mathrm{N_b}$ we get:
$K_{\mathrm{A,t4}}(\ \mathrm{t, t0 < t < t4}, S(\mathrm{B, t, E(N_b, K_{ab})}))$:**Satisfies Goal 3**

Step 5:
$K_{\mathrm{B,t5}}\ (R(\mathrm{B, t5, X}) \quad C(\mathrm{X, E(N_b\text{-}1, K_{ab})}))$
Using Axiom A2: $R(\mathrm{B, t5, X}) \quad C(\mathrm{X, E(N_b\text{-}1, K_{ab})})$

Applying Axiom A6 and Inference Rule R2:
$L_{\mathrm{B,t5}}\mathrm{X} \quad K_{\mathrm{B,t5}}(\ \mathrm{i, i} \quad \{\mathrm{ENT/B}\}, \quad \mathrm{t, t < t5}, S(\mathrm{i, t, X}))$
$C(\mathrm{X,E(N_b\text{-}1, K_{ab})})$

Applying Inference Rule R3:
$K_{\mathrm{B,t5}}(\ \mathrm{i, i} \quad \{\mathrm{ENT/B}\}, \quad \mathrm{t, t < t5}, S(\mathrm{i, t, E(N_b\text{-}1, K_{ab})}))$

Applying Axioms A11/A12/A13 and Assumption 3:
$K_{\mathrm{B,t5}}(\ \mathrm{t, t < t5}, S(\mathrm{A, t, E(N_b\text{-}1, K_{ab})}))$

Using Assumption 5 to give the freshness of $\mathrm{N_b}$ we get:
$K_{\mathrm{B,t5}}(\ \mathrm{t, t0 < t < t5}, S(\mathrm{A, t, E(N_b\text{-}1, K_{ab})}))$ **: Satisfies Goal 4**

3.4 Summary

The analysis presented above for the Needham-Schroeder key distribution protocol demonstrates the use of the extended logic in analysing symmetric key protocols. In this case it has highlighted the known flaw in the Needham-Schroeder protocol. This flaw, allows an old compromised session key to be replayed.

4. BOYD–PARK PROTOCOL

The Boyd–Park protocol [2,8] was published in an attempt to correct the apparent weakness in the ASPeCT protocol in identifying the participating mobile at the beginning of the protocol. Two versions were published, the first provides key transport and the second key agreement. In this section only the key transport protocol is discussed, as both protocols are quite similar with only minor changes.

4.1 Key transport protocol

The key transport protocol proceeds as follows:

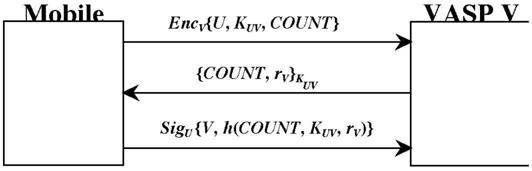
Mobile
VASP V
$Enc_V\{U, K_{UV}, COUNT\}$
$\{COUNT, r_V\}_{K_{UV}}$
$Sig_U\{V, h(COUNT, K_{UV}, r_V)\}$

Figure 2 Boyd-Park Key Transport Protocol

Notation:
COUNT: Used to detect cloning fraud in mobile handsets.
EncV{X}: Denotes encryption of the field X using the public key of V.

Step 1. Here the shared key K_{UV} is generated entirely by the mobile *U* and then sent encrypted under the public key of the VASP *V* along with the mobiles identity *U* and the *COUNT* field.
Step 2: The VASP *V* replies with the *COUN*T field and a random nonce r_V, both are encrypted using a symmetric algorithm and the session key K_{UV}.
Step 3: The mobile *U* now replies with a signed message containing the VASP's identity *V*, and a hash of the *COUNT* field, the session key K_{UV} and the random nonce r_V sent by *V* to *U* in step 2.

Both of the Boyd–Park protocol variations assume that the mobile *U* and the VASP *V* have access to each others public keys. The mobile *U* could receive the networks public key from the broadcast channel of the network, while the VASP V may require a trusted third party (TTP) to obtain the public key of the mobile *U* if it does not already know this key, as is the case with the ASPeCT protocol. The ID of the mobiles TTP ($\mathrm{id_{TTP}}$) can be included in the encrypted message of step 1 for this purpose.

4.2 Summary

The first message in the protocol supplies the identity of the mobile *U*. User anonymity is achieved by encrypting the message with the public key of the VASP *V*. The second message supplies confirmation to the mobile *U* of the session keys arrival and a random nonce for use with step 3. The third message provides key confirmation and key freshness for the VASP *V*. It also provides authentication of the mobile for use in electronic commerce applications.

5. FORMAL VERIFICATION OF KEY TRANSPORT PROTOCOL

The logic is now used to formally verify the Boyd-Park hybrid mobile security protocol. As a result of this analysis a potential weakness in the protocol is highlighted and a modification to the protocol is proposed. In the analysis tn refers to the time at the end of step *n* of the protocol.

5.1 Goals of the protocol

The goals of the key-transport protocol are defined as follows:

Goal 1: $K_{V,t1}(\quad t, t < t1, S(U, t, X)$
$\qquad C(X, (U, K_{UV})))$
Goal 2: $K_{U,t2}(\quad t, t1 < t < t2, S(V, t, X)$
$\qquad C(X, E(r_v, K_{UV})))$
Goal 3: $K_{V,t3}(\quad t, t2 < t < t3, S(U, t, X)$
$\qquad C(X, e((r_v, K_{UV}), K_U{}^{-1})))$

Goal 1 states that the VASP *V* knows it will obtain the session key K_{UV} from *U* prior to the end of step 1.

Goal 2 states that the mobile *U* knows it will obtain the random nonce r_v from the VASP *V* after step 1 but before the end of step 2. This shows the timeliness of the nonce r_v.

Goal 3 states that the VASP *V* knows it will obtain a signed message from *U* containing the random challenge rv and the session key K_{UV}. This provides mobile authentication to the VASP *V* (establishes the definitive ID of the sender in step 1).

5.2 Initial assumptions

1: $\quad i, \quad t, i \quad \{ENT\}(K_{i,t}K_V \quad K_{i,t}K_U)$
2: $\quad i, \quad t, i \quad \{ENT\}(K_{i,t}V \quad K_{i,t}U)$
3: $\quad i, \quad j, \quad t, (i \quad \{U,V\}L_{i,t}Count \quad j \quad \{ENT/U,V\}$
$\quad \neg L_{j,t}Count) \rightarrow (Count \quad SS_{\{U,V\}})$
4: $L_{U,t0}K_{UV} \quad K_{U,t0}(\quad i, \quad t, i \quad \{ENT/U\}, t < t1,$
$\quad \neg L_{i,t}K_{UV}) \rightarrow (K_{UV} \quad KS_{\{U,V\}})$
5: $K_{V,t0}(\quad i, \quad t, i \quad \{ENT/V\}, t < t2, \neg L_{i,t}r_V)$
6: $\quad i, \quad j, \quad t, ((t > t2, i \quad \{U,V\}L_{i,t}r_V)$
$\qquad (j \quad \{ENT/U,V\} \neg L_{j,t}r_V)) \rightarrow (r_v \quad SS_{\{U,V\}})$
7: $\quad i, \quad t, i \quad \{ENT/V\} (\neg L_{i,t}K_V{}^{-1} \rightarrow (t0 < t < t3, \neg L_{i,t}U))$

Assumptions 1 and 2 state that the public keys and ID's of *U* and *V* are known to all entities.

Assumption 3 states that *Count* is known only to *U* and *V* and is therefore a good secret.

Assumption 4 states that *U* generates the shared key K_{UV} and that *U* knows that no other entity knows this key prior to t1, and that the key is therefore a good key.

Assumption 5 refers to the timely revelation of the random nonce r_v by the VASP *V*.

Assumption 6 states that only the entities *U* and *V* involved in the transaction will know r_v after step 2, and this indicates that r_v is a good secret.

Assumption 7 refers to the anonymity property of the Boyd-Park protocol and states that since the private key of the VASP $V(K_V{}^{-1})$ is known only to *V* (axiom 9) then this implies that the identity of the mobile *U* is not known to any entity other that *V* for the duration of the protocol.

5.3 Message exchanges

The following messages are exchanged during the operation of the Boyd-Park key-transport protocol.

Step 1: U → V: $Enc_v\{U, K_{UV}, Count\}$
Step 2: U ← V: $\{Count, r_v\}K_{UV}$
Step 3: U → V: $Sig_U\{V, h(Count, K_{UV}, r_v)\}$

Rewriting each step of the protocol in the language of the logic we obtain: (note: in this analysis the Count value is ignored).

Step 1:
$K_{V,t1} (R(V,t1,X) \quad C(X, e((U, K_{UV}), K_V))$

This states that *V* knows at time t1 that it will receive a message *X* containing the ID of mobile *U* and the shared session key K_{UV} encrypted using the public key of the VASP *V*.

By application of Axiom A2: $R(V,t1,X) \quad C(X, e((U, K_{UV}), K_V)$

Applying Axiom A6 and Inference Rule R2:
$L_{V,t1}X \quad K_{V,t1} (\quad i, i \quad \{ENT/V\}, \quad t, t < t1, S(i,t,X))$
$\qquad C(X, e((U, K_{UV}), K_V))$

Applying Inference Rule R3:
$K_{V,t1} (\quad i, i \quad \{ENT/V\}, \quad t, t < t1, S(i,t,X))$
$\qquad C(X, e((U, K_{UV}), K_V))$

Applying Axioms A7/A8 and Assumption 2:

Axiom A7 states that entity *V* can decrypt message *X* since it knows the secret key $K_V{}^{-1}$ and A8 states that no other entity can decrypt message *X*. This gives the VASP *V* the ID of mobile *U* and the shared key K_{UV}.

$K_{V,t1} (\quad t, t < t1, S(U,t,X) \quad C(X, (U, K_{UV})))$ (***BP1***)

Given that the identity of *U* is not known for certain until *U* is authenticated after step 3. Another initial assumption may be introduced:

$B_{V,t0} (\quad t, t < t3, S(U,t,X)$
$\quad C(X, (U, K_{UV})))$:**Initial Assumption 8**

This new assumption allows expression BP1 to be rewritten as an expression of belief rather than knowledge, and should read:

$B_{V,t1} (\quad t, t < t1, S(U,t,X) \quad C(X, (U, K_{UV})))$

This means that Goal 1 should be an expression of belief rather than knowledge. This presents a possible weakness, similar to that of the ASPeCT protocol that will allow a rogue entity to present itself as mobile *U* in the early stages of the protocol.

The anonymity of *U* is provided due to the encryption of *U* and K_{UV} by *V*'s public key. This encryption by K_V leads to the authentication of *V* in step 2.

Step 2:
$K_{U,t2} (R(U,t2,X) \quad C(X, E(r_v, K_{UV})))$

This states that *U* knows at time t2 that it will receive a message *X* from the VASP *V* and that *X* contains the random nonce r_v, all encrypted using the shared key K_{UV}.

Applying Axiom A2: $R(U,t2,X) \quad C(X, E(r_v, K_{UV}))$

Applying Axiom A6 and Inference Rule R2:
$L_{U,t2}X \quad K_{U,t2} (\quad i, i \quad \{ENT/U\}, \quad t, t < t2, S(i,t,X))$
$\qquad C(X, E(r_v, K_{UV}))$

Applying Inference Rule R3:
$K_{U,t2} (\quad i, i \quad \{ENT/U\}, \quad t, t < t2, S(i,t,X)) \quad C(X, E(r_v, K_{UV}))$

Applying Axioms A11/A12 and A13, which state that entity *U* can decrypt message *X* using K_{UV}, and K_{UV} is a good secret key. Assumption 4 states that K_{UV} is only known to entities *U* and *V*. Therefore:

$$K_{U,t2}\ (\ \ t, t < t2, S(V,t,X) \quad C(X, E(r_v, K_{UV})))$$

Assumption 4 also gives the timely arrival of K_{UV} to entity *V* at time t1, therefore:

$$K_{U,t2}(\ \ t, t1 < t < t2, S(V,t,X) \\ C(X, E(r_v, K_{UV}))):$$ **Satisfies Goal 2**

Step 3:
$K_{V,t3}\ (R(V,t3,X) \quad C(X, e(Mes, K_U^{-1})))$
where Mes = (V, h($Count$, K_{UV}, r_v))

This states that *V* knows at time t3 that it will receive a message *X* from a mobile, and this message will be signed by the secret key of the mobile. The message will contain *V*'s ID, random nonce r_v, *Count* and shared secret K_{UV}.

Applying Axiom A2: $R(V,t3,X) \quad C(X, e(Mes, K_U^{-1}))$

Applying Axiom A6 and Inference Rule R2:
$$L_{V,t3}X \quad K_{V,t3}\ (\ \ i, i \quad \{ENT/V\}, \quad t, t < t3, S(i,t,X)) \\ C(X, e(Mes, K_U^{-1}))$$

Applying Inference Rule R3:
$$K_{V,t3}\ (\ \ i, i \quad \{ENT/V\}, \quad t, t < t3, S(i,t,X)) \\ C(X, e(Mes, K_U^{-1}))$$

Applying Axioms A11/A12:
$$K_{V,t3}\ (\ \ t, t < t3, S(U,t,X) \quad C(X, e(Mes, K_U^{-1})))$$

Assumption 5 gives the timely arrival of r_v:
$$K_{V,t3}\ (\ \ t, t2 < t < t3, S(U,t,X) \\ C(X, e(Mes, K_U^{-1})))$$: **Satisfies Goal 3**

The mobile *U* is authenticated at this point of the protocol since only it could have encrypted *Mes* with its secret key K_U^{-1} (Axioms A7/A8/A15(b)) and *Mes* contains the random challenge r_v, and the shared key K_{UV}, therefore:

$$A(V, t3, U) \rightarrow K_{V,t3}\ (\ \ t, t < t3, S(U,t,X) \quad C(X, (U, K_{UV})))$$

This statement does not meet the requirements of goal 1, as mobile *U* is not authenticated until the end of the protocol, t3. Up to t3, the VASP *V* does not know with certainty the identity of the mobile, as only belief is established.

5.4 Summary

From the analysis presented it can be seen that two goals, Goal 2 and Goal 3, of the Boyd-Park protocol are achieved. The third, Goal 1, has not been established. Although V believes the ID of the mobile *U* after step 1, no knowledge is gained until step 3.

6. PROPOSED MODIFICATION TO THE BOYD–PARK KEY-TRANSPORT PROTOCOL AND ITS ANALYSIS

The reason for this modification to the protocol is to allow the authentication of the mobile *U* by the VASP *V* to take place at the start of the protocol. This removes any possible benefits a rogue mobile might obtain by masquerading as *U* in the early stages of the protocol. In the proposed modification the processing requirements do not increase since the signing of the session key can be performed offline by the mobile.

The following messages are now exchanged to reflect the modification to the Boyd–Park key-transport protocol.

Step 1: U → V: $Enc_v\{U, Sig_u\{K_{UV}, Count\}\}$
Step 2: U ← V: $\{Count, r_v\}K_{UV}$
Step 3: *U* → *V*: $Sig_U\{V, h(Count, K_{UV}, r_v)\}$

The following assumption can now be included in the list of assumptions in section 5.2.

$$K_{V,t0}\ (\ \ t, t < t1, S(U,t,X) \\ C(X, Sig_U(U, K_{UV})))$$:**Initial Assumption 8**

This states that VASP *V* knows that it was mobile *U* who sent the message *X* prior to the end of step 1, and this message *X* contains the ID of *U* and the secret session key K_{UV}. The mobile *U* has signed the session key, thereby enabling its identity to be authenticated during step 1. With this additional assumption the analysis of step 1 now becomes:

Step 1:
$K_{V,t1}\ (R(V,t1,X) \quad C(X, e((U, K_{UV}), K_V))$

This states that *V* knows at time t1 that it will receive a message *X* containing the ID of mobile *U* and the shared session key K_{UV} encrypted using the public key of the VASP *V*.

By application of Axiom A2: $R(V,t1,X) \quad C(X, e((U, K_{UV}), K_V)$

Applying Axiom A6 and Inference Rule R2:
$$L_{V,t1}X \quad K_{V,t1}\ (\ \ i, i \quad \{ENT/V\}, \quad t, t < t1, S(i,t,X)) \\ C(X, e((U, K_{UV}), K_V))$$

Applying Inference Rule R3:
$$K_{V,t1}\ (\ \ i, i \quad \{ENT/V\}, \quad t, t < t1, S(i,t,X)) \\ C(X, e((U, K_{UV}),K_V))$$

Applying Axioms A7/A8: Axiom A7 states that entity *V* can decrypt message *X* since it knows the secret key K_V^{-1} and A8 states that no other entity can decrypt message *X*. This gives the VASP *V* the ID of mobile *U* and the shared key K_{UV}.

$$K_{V,t1}\ (\ \ i, i \quad \{ENT/V\}, \quad t, t < t1, S(i,t,X)) \\ C(X, (U, K_{UV})))$$

Now applying Assumptions 2 and 8: Assumption 2 states that the public keys of all protocol entities are known to all entities, and assumption 8 states that the mobile *U* is authenticated during step 1.

$K_{V,t1}\,(\;\; t,\, t < t1,\, S(U,t,X) \quad C(X,\, (U,\, K_{UV})))$: **Satisfies Goal 1**

The analysis of the other protocol steps remains the same.

7. CONCLUSIONS

In this paper a formal modal logic was presented. The logic enables hybrid cryptographic security protocols to be formally verified. The logic was applied to a symmetric security protocol and a hybrid security protocol. The analysis of the symmetric Needham–Schroeder key-distribution protocol identified a know flaw in the protocol. In the case of the hybrid Boyd–Park key-transport protocol, the analysis highlighted a new weakness in timing of the authentication of the mobile unit. A proposed modification to the protocol to correct this weakness was presented and verified.

REFERENCES

1. Advanced Security for Personal Communications Technologies.http://www.esat.kuleuven.ac.be/cosic/aspect/index.html
2. Boyd C. and Park D.G. “Public Key Protocols for Wireless Communications”, *ICISC '98*, pp. 47–57, 1998.
3. Coffey T. and Saidha P. “Logic for verifying public-key cryptographic protocols”. *IEE Proceedings – Comput. Digit. Tech.*, Vol. 144, No. 1. pp 28–32 , January 1997
4. Burrows M., Abadi M., and Needham R.M. “A Logic of Authentication”. DEC Systems Research Report No. 39. February 1990.
5. Gong L., Needham R., and Yahalom R. “Reasoning about Belief in Cryptographic protocols”. *Proceedings: 1990 IEEE Computer Society Symposium on Research in Security and Privacy*. Oakland, CA, USA. pp 234–248. 1990.
6. Syverson P. “A Logic for the Analysis of Cryptographic Protocols”. Naval Research Laboratory Report No. 9305, December 31 1990.
7. Bieber P. “A Logic of Communication in a Hostile Environment”. *Proceedings of the Computer Security Foundations Workshop III*. Washington (IEEE), pp 14–22. 1990.
8. Horn G., Martin K.M. and Mitchell C.J., “Authentication protocols for mobile network environment value-added services”, 2000. http://isg.rhbnc.ac.uk/cjm/APFMNE.zip