Hybrid Mobile Security Protocol: Formal Verification using a New Modal Logic

T. NEWE & T.COFFEY
Department of Electronic & Computer Engineering,
University of Limerick,
Ireland.
E-Mail: (Thomas.Newe@UL.ie) (Tom.Coffey@UL.ie)
Fax: 353-61-338176

ABSTRACT

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 protocol. 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 the Boyd-Park hybrid mobile security protocol is formally verified using a new extended modal logic technique. This logic is based on the Coffey-Saidha logic [3], which was designed to verify public key based protocols only. The logic presented in this paper is a substantial extension to the original. As a result of this verification a modification to the original Boyd-Park protocol is suggested.

Key-Words: Formal methods, verification logic, hybrid security protocol, 3G mobile

1. Introduction

Security communication protocols in general 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. These goals are typical of standard communication systems but they are not all that is needed for a secure mobile system. 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 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 formal verification logic is expanded to include symmetric cryptographic system verification capabilities to provide the necessary language and rule set to formally verify hybrid security protocols. The new logic is then used to formally verify the Boyd-Park hybrid mobile security protocol, and as a result of this verification a modification to the original protocol is suggested.

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.

2.1.2 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_Σ), encryption function, encryption of *x* using key k_Σ
- **d**(x,k_Σ^{-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.

2.1.3 Inference Rules

The logic incorporates the following rules of inference:

R1: From $\vdash$ p and $\vdash$ $(p \rightarrow q)$ infer $\vdash$ q

R2: **(a)** From $\vdash$ p infer $\vdash$ $K_{\Sigma,t}p$

(b) From $\vdash$ p infer $\vdash$ $B_{\Sigma,t}p$

R1 is the *Modus Ponens* and states that if schema *p* can be deduced and $(p \rightarrow q)$ 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 $(p \wedge q)$ infer p

R4: From p and q infer $(p \wedge q)$

R5: From p infer $(p \vee q)$

R6: From $\neg(\neg p)$ infer p

R7: From (from p infer q) infer $(p \rightarrow q)$

2.1.4 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.

2.1.4.1 Logical Axioms

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

A1: **(a)** $\exists t \exists p \exists q(K_{\Sigma,t}p \wedge K_{\Sigma,t}(p \rightarrow q) \rightarrow K_{\Sigma,t}q)$

(b) $\exists t \exists p \exists q(B_{\Sigma,t}p \wedge B_{\Sigma,t}(p \rightarrow q) \rightarrow B_{\Sigma,t}q)$

A2: $\exists t \exists p(K_{\Sigma,t}p \rightarrow p)$

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)** $\exists t \exists x \exists i, i \in \{ENT\}(L_{i,t}x \rightarrow \forall t', t' \geq t\ L_{i,t'}x)$

(b) $\exists t \exists x \exists i, i \in \{ENT\}(K_{i,t}x \rightarrow \forall t', t' \geq t\ K_{i,t'}x)$

(c) $\exists t \exists x \exists i, i \in \{ENT\}(B_{i,t}x \rightarrow \forall t', t' \geq t\ B_{i,t'}x)$

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: $\exists t \exists x \exists y(\exists i, i \in \{ENT\}L_{i,t}y \wedge C(y,x) \rightarrow \exists j, j \in \{ENT\}L_{j,t}x)$

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:

2.1.4.2 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: $\exists t \exists x(\ S(\Sigma,t,x) \rightarrow L_{\Sigma,t}x \wedge \exists i, i \in \{ENT/\Sigma\} \exists t', t' > t\ R(i,t',x)\ )$

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: $\exists t \exists x(\ R(\Sigma,t,x) \rightarrow L_{\Sigma,t}x \wedge \exists i, i \in \{ENT/\Sigma\} \exists t', t' < t\ S(i,t',x)\ )$

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)** $\exists t \exists x \exists i, i \in \{ENT\}(\ L_{i,t}x \wedge L_{i,t}k_{\Sigma} \rightarrow L_{i,t}(e(x,k_{\Sigma}))\ )$

(b) $\exists t \exists x \exists i, i \in \{ENT\}(\ L_{i,t}x \wedge L_{i,t}k_{\Sigma}^{-1} \rightarrow L_{i,t}(d(x,k_{\Sigma}^{-1}))\ )$

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)** $\exists t \exists x \exists i, i \in \{ENT\}(\ \neg L_{i,t}k_{\Sigma} \wedge \forall t', t' < t\ \neg L_{i,t'}(e(x,k_{\Sigma})) \wedge \neg(\exists y(\ R(i,t,y) \wedge C(y,e(x,k_{\Sigma}))\ )) \rightarrow \neg L_{i,t}(e(x,k_{\Sigma})))$

(b) $\exists t\exists x\exists i,i\in\{ENT\}(\neg L_{i,t}k_{\Sigma}^{-1} \wedge \forall t',t'<t \neg L_{i,t'}(d(x,k_{\Sigma}^{-1})) \wedge \neg(\exists y(R(i,t,y) \wedge C(y,d(x,k_{\Sigma}^{-1})))) \rightarrow \neg L_{i,t}(d(x,k_{\Sigma}^{-1})))$

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 $e(x,k_{\Sigma})$ and also does not receive $e(x,k_{\Sigma})$ at *t* in a message, then the entity cannot know $e(x,k_{\Sigma})$ at time *t*. Axiom A8b makes a similar statement for the decryption of a message *x* without knowledge of the decryption key.

A9: $\forall t(\forall i,i\in\{ENT\}L_{i,t}k_{i}^{-1} \wedge \forall j,j\in\{ENT/i\}\neg L_{j,t}k_{i}^{-1})$

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

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

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 to express the properties of symmetric key systems 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(Σ,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 cryptographic protocols, however these protocols must be public key protocols. In this section additional axioms are presented for inclusion in the logic which will enable symmetric key systems to be analysed also.

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)** $\exists t\exists x\exists i,i\in\{ENT\} (L_{i,t}x \wedge L_{i,t}ks_{(\Sigma,\Psi)} \rightarrow L_{i,t}(E(x, ks_{(\Sigma,\Psi)})))$

(b) $\exists t\exists x\exists i,i\in\{ENT\} (L_{i,t}x \wedge L_{i,t}ks_{(\Sigma,\Psi)} \rightarrow L_{i,t}(D(x, ks_{(\Sigma,\Psi)})))$

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

A12: **(a)** $\exists t\exists x\exists i,i\in\{ENT\}(\neg L_{i,t}\ ks_{(\Sigma,\Psi)} \wedge \forall t',t' < t \neg L_{i,t'}(E(x, ks_{(\Sigma,\Psi)})) \wedge\neg(\exists y(R(i,t,y) \wedge C(y,E(x, ks_{(\Sigma,\Psi)})))) \rightarrow \neg L_{i,t}(E(x, ks_{(\Sigma,\Psi)})))$

(b) $\exists t \exists x \exists i, i \in \{ENT\}(\neg L_{i,t}\ ks_{(\Sigma,\Psi)} \wedge \forall t', t' < t\ \neg L_{i,t'}(D(x, ks_{(\Sigma,\Psi)})) \wedge \neg(\exists y(R(i,t,y) \wedge C(y,D(x, ks_{(\Sigma,\Psi)})))) \rightarrow \neg L_{i,t}(D(x, ks_{(\Sigma,\Psi)})))$

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 and this implies that this key is a good key.

A13: $\forall t((\forall i, i \in \{ENT/\Sigma,\Psi\} \neg L_{i,t} ks_{(\Sigma,\Psi)} \wedge \exists j, j \in \{\Sigma,\Psi\}\ L_{j,t} ks_{(\Sigma,\Psi)}) \rightarrow ks_{(\Sigma,\Psi)} \in \{KS_{\{\Sigma,\Psi\}}\}))$

2.3.4 Axiom 14

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

A14: $\forall t((\forall i, i \in \{ENT/\Sigma,\Psi\} \neg L_{i,t} ss_{(\Sigma,\Psi)} \wedge \exists j, j \in \{\Sigma,\Psi\} L_{j,t} ss_{(\Sigma,\Psi)}) \rightarrow ss_{(\Sigma,\Psi)} \in \{SS_{\{\Sigma,\Psi\}}\}))$

2.3.5 Axiom 15

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

A15: **(a)**$\exists x \exists t(A(\Sigma,t,\Psi) \rightarrow (L_{\Sigma,t} ss_{(\Sigma,\Psi)} \wedge ss_{(\Sigma,\Psi)} \in \{SS_{\{\Sigma,\Psi\}}\} \wedge R(\Sigma,t,x) \wedge C(x, ss_{(\Sigma,\Psi)}) \wedge \forall t', t' < t\ \neg S(\Sigma,t',x)) \rightarrow K_{\Sigma,t}(S(\Psi,t',x))))$

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

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*.

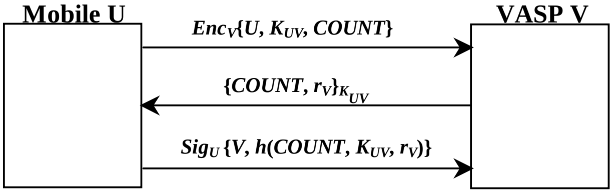
A15 (b) states: If Σ knows the public key of Ψ (k_Ψ) and message x, and if Σ receives a message y containing $e(x, k_\Psi^{-1})$ then Σ knows that Ψ sent message y prior to t.

3. 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.

3.1 Key Transport Protocol

The key transport protocol proceeds as follows:



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

Fig. 1: Boyd-Park Key Transport Protocol

Notation:
COUNT: Used to detect cloning fraud in mobile handsets.
$Enc_V\{X\}$: 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 *COUNT* 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* (id_{TTP}) can be included in the encrypted message of step 1 for this purpose.

3.2 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.

4. Formal Verification of Key Transport Protocol

In the analysis *tn* refers to the time at the end of step *n* of the protocol.

4.1 Goals of the Protocol

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

Goal 1: $K_{\mathrm{V,t1}}(\exists \mathrm{t}, \mathrm{t} < \mathrm{t1}, S(\mathrm{U}, \mathrm{t}, \mathrm{X}) \wedge C(\mathrm{X}, (\mathrm{U}, K_{UV})))$

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

Goal 3: $K_{\mathrm{V,t3}}(\exists \mathrm{t}, \mathrm{t2} < \mathrm{t} < \mathrm{t3}, S(\mathrm{U}, \mathrm{t}, \mathrm{X}) \wedge C(\mathrm{X}, \mathrm{e}((r_v, K_{UV}), \mathrm{K_U}^{-1})))$

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. This step is to correct the apparent weakness that exists with the ASPeCT protocol.

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 r_v and the session key K_{UV}. This provides mobile authentication to the VASP *V* (establishes the definitive ID of the sender in step 1).

4.2 Initial Assumptions

1: $\forall \mathrm{i}, \forall \mathrm{t}, \mathrm{i} \in \{\mathrm{ENT}\}(K_{\mathrm{i,t}}\mathrm{K_V} \wedge K_{\mathrm{i,t}}\mathrm{K_U})$

2: $\forall \mathrm{i}, \forall \mathrm{t}, \mathrm{i} \in \{\mathrm{ENT}\}(K_{\mathrm{i,t}}\mathrm{V} \wedge K_{\mathrm{i,t}}\mathrm{U})$

3: $\forall \mathrm{i}, \forall \mathrm{j}, \forall \mathrm{t}, (\mathrm{i} \in \{\mathrm{U,V}\}L_{\mathrm{i,t}}\mathrm{Count} \wedge \mathrm{j} \in \{\mathrm{ENT/U,V}\} \neg L_{\mathrm{i,t}}\mathrm{Count}) \rightarrow (\mathrm{Count} \in \mathrm{SS}_{\{\mathrm{U,V}\}})$

4: $L_{\mathrm{U,t0}}\mathrm{K_{UV}} \wedge K_{\mathrm{U,t0}}(\forall \mathrm{i}, \forall \mathrm{t}, \mathrm{i} \in \{\mathrm{ENT/U}\}, \mathrm{t} < \mathrm{t1}, \neg L_{\mathrm{i,t}}\mathrm{K_{UV}}) \rightarrow (\mathrm{K_{UV}} \in \mathrm{KS}_{\{\mathrm{U,V}\}})$

5: $K_{\mathrm{V,t0}}(\forall \mathrm{i}, \forall \mathrm{t}, \mathrm{i} \in \{\mathrm{ENT/V}\}, \mathrm{t} < \mathrm{t2}, \neg L_{\mathrm{i,t}}\mathrm{r_v})$

6: $\forall \mathrm{i}, \forall \mathrm{j}, \forall \mathrm{t}, ((\mathrm{t} > \mathrm{t2}, \mathrm{i} \in \{\mathrm{U,V}\}L_{\mathrm{i,t}}\mathrm{r_v}) \wedge (\mathrm{j} \in \{\mathrm{ENT/U,V}\} \neg L_{\mathrm{j,t}}\mathrm{r_v})) \rightarrow (\mathrm{r_v} \in \mathrm{SS}_{\{\mathrm{U,V}\}})$

7: $\forall \mathrm{i}, \forall \mathrm{t}, \mathrm{i} \in \{\mathrm{ENT/V}\} (\neg L_{\mathrm{i,t}}\mathrm{K_V}^{-1} \rightarrow (\mathrm{t0} < \mathrm{t} < \mathrm{t3}, \neg L_{\mathrm{i,t}}\mathrm{U}))$

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.

4.3 Message Exchanges

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

Step 1: $\mathrm{U} \rightarrow \mathrm{V}: Enc_v\{U, K_{UV}, Count\}$

Step 2: $\mathrm{U} \leftarrow \mathrm{V}: \{Count, r_v\}_{K_{UV}}$

Step 3: $\mathrm{U} \rightarrow \mathrm{V}: Sig_U\{V, \mathrm{h}(Count, K_{UV}, r_v)\}$

Rewriting each step in the language of the new logic we get: For the purpose of this analysis the *Count* value is ignored.

Step 1:

$K_{\mathrm{V,t1}}\ (R(\mathrm{V,t1,X}) \wedge C(\mathrm{X}, \mathrm{e}((\mathrm{U}, K_{UV}), K_V))$

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:

$R(\mathrm{V},\mathrm{t1},\mathrm{X}) \wedge C(\mathrm{X}, \mathrm{e}((\mathrm{U}, K_{UV}), K_V)$

Applying Axiom A6 and Inference Rule R2:

$L_{\mathrm{V,t1}}\mathrm{X} \wedge K_{\mathrm{V,t1}}\ (\exists \mathrm{i},\ \mathrm{i} \in \{\mathrm{ENT/V}\},\ \exists \mathrm{t},\ \mathrm{t} < \mathrm{t1},\ S(\mathrm{i},\mathrm{t},\mathrm{X})) \wedge C(\mathrm{X}, \mathrm{e}((\mathrm{U}, K_{UV}), K_V))$

Applying Inference Rule R3:

$K_{\mathrm{V,t1}}\ (\exists \mathrm{i},\ \mathrm{i} \in \{\mathrm{ENT/V}\},\ \exists \mathrm{t},\ \mathrm{t} < \mathrm{t1},\ S(\mathrm{i},\mathrm{t},\mathrm{X})) \wedge C(\mathrm{X}, \mathrm{e}((\mathrm{U}, K_{UV}), K_V))$

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}.

$K_{\mathrm{V,t1}}\ (\exists \mathrm{t},\ \mathrm{t} < \mathrm{t1},\ S(\mathrm{U},\mathrm{t},\mathrm{X}) \wedge C(\mathrm{X}, (\mathrm{U}, K_{UV})))$ ***(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:

$B_{\mathrm{V,t0}}\ (\exists \mathrm{t},\ \mathrm{t} < \mathrm{t3},\ S(\mathrm{U},\mathrm{t},\mathrm{X}) \wedge C(\mathrm{X}, (\mathrm{U}, K_{UV})))$

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

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

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:

$K_{\mathrm{U,t2}}\ (R(\mathrm{U},\mathrm{t2},\mathrm{X}) \wedge C(\mathrm{X}, \mathrm{E}(r_v, K_{UV})))$

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: $R(\mathrm{U},\mathrm{t2},\mathrm{X}) \wedge C(\mathrm{X}, \mathrm{E}(r_v, K_{UV}))$

Applying Axiom A6 and Inference Rule R2:

$L_{\mathrm{U,t2}}\mathrm{X} \wedge K_{\mathrm{U,t2}}\ (\exists \mathrm{i},\ \mathrm{i} \in \{\mathrm{ENT/U}\},\ \exists \mathrm{t},\ \mathrm{t} < \mathrm{t2},\ S(\mathrm{i},\mathrm{t},\mathrm{X})) \wedge C(\mathrm{X}, \mathrm{E}(r_v, K_{UV}))$

Applying Inference Rule R3:

$K_{\mathrm{U,t2}}\ (\exists \mathrm{i},\ \mathrm{i} \in \{\mathrm{ENT/U}\},\ \exists \mathrm{t},\ \mathrm{t} < \mathrm{t2},\ S(\mathrm{i},\mathrm{t},\mathrm{X})) \wedge C(\mathrm{X}, \mathrm{E}(r_v, K_{UV}))$

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:

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

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

$K_{\mathrm{U,t2}}\ (\exists \mathrm{t},\ \mathrm{t1} < \mathrm{t} < \mathrm{t2},\ S(\mathrm{V},\mathrm{t},\mathrm{X}) \wedge C(\mathrm{X}, \mathrm{E}(r_v, K_{UV})))$

: Satisfies Goal 2

Step 3:

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

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: $R(\mathrm{V},\mathrm{t3},\mathrm{X}) \wedge C(\mathrm{X}, \mathrm{e}(\mathrm{Mes}, K_U^{-1}))$

Applying Axiom A6 and Inference Rule R2:

$L_{\mathrm{V,t3}}\mathrm{X} \wedge K_{\mathrm{V,t3}}\ (\exists \mathrm{i},\ \mathrm{i} \in \{\mathrm{ENT/V}\},\ \exists \mathrm{t},\ \mathrm{t} < \mathrm{t3},\ S(\mathrm{i},\mathrm{t},\mathrm{X})) \wedge C(\mathrm{X}, \mathrm{e}(\mathrm{Mes}, K_U^{-1}))$

Applying Inference Rule R3:

$K_{\mathrm{V,t3}}\ (\exists \mathrm{i},\ \mathrm{i} \in \{\mathrm{ENT/V}\},\ \exists \mathrm{t},\ \mathrm{t} < \mathrm{t3},\ S(\mathrm{i},\mathrm{t},\mathrm{X})) \wedge C(\mathrm{X}, \mathrm{e}(\mathrm{Mes}, K_U^{-1}))$

Applying Axioms A11/A12:

$K_{\mathrm{V,t3}}\ (\exists \mathrm{t},\ \mathrm{t} < \mathrm{t3},\ S(\mathrm{U},\mathrm{t},\mathrm{X}) \wedge C(\mathrm{X}, \mathrm{e}(\mathrm{Mes}, K_U^{-1})))$

Assumption 5 gives the timely arrival of r_v:

$K_{\mathrm{V,t3}}\ (\exists \mathrm{t},\ \mathrm{t2} < \mathrm{t} < \mathrm{t3},\ S(\mathrm{U},\mathrm{t},\mathrm{X}) \wedge C(\mathrm{X}, \mathrm{e}(\mathrm{Mes}, K_U^{-1})))$

: 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) → $K_{V,t3}$ (∃t, t < t1, S(U,t,X) ∧ C(X, (U, K_{UV})))

This closely relates to goal 1 but *V* does not obtain this knowledge until time t3, up to this point only trust (belief) is assumed.

5. CONCLUSIONS

From the analysis presented in this paper it can be seen that two goals of the Boyd-Park protocol are achieved and the third (Goal 1) is achieved albeit late, and a session key has been established. The anonymity of the mobile is achieved by encrypting the first message with the public key of the VASP *V*.

Although *V* believes the ID of the mobile *U* after step 1, no knowledge is gained until step 3. This authentication is necessary for non-repudiation based applications. With this analysis it can also be seen that the VASP *V* is authenticated in step 2.

A suggested modification to the protocol is to allow the authentication of the mobile by the VASP *V* to take place at the start of the protocol (Step 1) thereby removing any possible benefits a rogue mobile might obtain by masquerading as *U* in the early stages of the protocol.

This would entail the mobile *U* sending the following message for step 1:

$$Enc_v\{U, Sig_u\{K_{UV}, Count\}\}$$

Now the VASP *V* will need to authenticate *U* at the start of the protocol to recover the session key K_{UV}.

The processing requirements for the protocol do not increase since the new calculation can be performed offline by the mobile.

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