Refuting the Security Claims of Mathuria and Jain (2005) Key Agreement Protocols

Kim-Kwang Raymond Choo*
Australian Institute of Criminology, GPO Box 2944 Canberra, ACT 2601 Australia (Email: raymond.choo@aic.gov.au)

(Received June 21, 2006; revised and accepted Oct. 3, 2006 & Dec. 9, 2006)

Abstract

Despite the importance of proofs in assuring protocol implementers about the security properties of key establishment protocols, many protocol designers fail to provide any proof of security. Flaws detected long after the publication and/or implementation of protocols will erode the credibility of key establishment protocols. We examine the class of key agreement protocols (without proofs of security) due to Mathuria and Jain (2005). Using these protocols as case studies, we demonstrate previously unpublished flaws in these protocols. We may speculate that such errors could have been found by protocol designers if proofs of security were to be constructed, and hope this work will encourage future protocol designers to provide proofs of security.

Keywords: Key agreement protocol, key establishment protocol, provable security, security model

1 Introduction

Despite key establishment protocols being the *sine qua non* of many diverse secure electronic commerce applications, the design of secure key establishment protocols is still notoriously hard. The difficulties associated in obtaining a high level of assurance in the security of almost any new or even existing protocols are well illustrated with examples of errors found in many such protocols years after they were published [4, 5, 19, 22, 23, 38, 40, 41, 42]. The many flaws discovered in published protocols for key establishment and authentication over many years, have promoted the use of formal models and rigorous security proofs which led to a dichotomy in cryptographic protocol analysis techniques between the computational complexity approach [7, 16, 39] and the computer security approach [2, 3, 35, 36].

In the computational complexity paradigm for protocols, a deductive reasoning process is adopted whereby emphasis is placed on a proven reduction from the problem of breaking the protocol to another problem believed to be hard. One advantage of protocols proven secure in this approach is that description of protocols security and the goals provided by the protocols are formally defined. For example, we will know whether a proposed attack is valid and what it means to be secure. A complete mathematical proof with respect to cryptographic definitions provides a strong assurance that a protocol is behaving as desired. The history of mathematics is, however, full of erroneous proofs [15]. One such example is illustrated in the *virtuoso* work of Lakatos [31] whereby the many proofs and refutations for Euler's characteristic in algebraic topology are presented as a comedy of errors. Many formulations for Euler's characteristic in algebraic topology, a theorem about the properties of polyhedra, have been tried, only to be refuted and replaced by another formulation.

The difficulty of obtaining correct computational proofs of security is also dramatically illustrated by the well-known problem with the OAEP mode for public key encryption [40]. Although OAEP was one of the most widely used and implemented algorithms, it was several years after the publication of the original proof that a problem was found (and subsequently fixed in the case of RSA). Problems with proofs of protocol security have occurred too, evidenced by the breaking of several provably-secure protocols [19, 41, 42] after they were published.

Despite these setbacks, proofs are invaluable for arguing about security and certainly are one very important tool in getting protocols right. Moreover, having security proofs allow protocol designer to formally state the desirable properties/goals that a protocol offers (giving assurance to protocol implementors).

We advocate the importance of proofs of protocol security and the proposal of any entity authentication and/or key establishment protocol should provide a rigorous proof of security (as we argue that protocols without any computational proofs of security leads one to question the level of trust in the correctness in such protocols). In a recent work, Mathuria and Jain propose a class of key agreement protocols [33] as improvements to the Boyd's class of efficient key agreement protocols [12, 13]. We use

*The views and opinions expressed in this paper are those of the author and do not reflect those of the Australian Government or the Australian Institute of Criminology. Research was performed while the author was with the Information Security Institute/Queensland University of Technology.

the Mathuria–Jain key agreement protocols (which have no proofs of security) as case studies, and demonstrate previously unknown flaws in these protocols. We then propose simple fixes to these protocols. Proof sketches for the fixed protocols are also presented. We work in the widely accepted indistinguishability-based models of Bellare, Pointcheval and Rogaway (hereafter referred to as the Bellare–Rogaway model) [6, 7, 9] and the random oracle model (also known as the ideal hash model) [8][1].

Mathuria and Jain pointed out that if session key is used within the protocol, the definition of security in the Bellare and Rogaway proof model will be violated. Hence, they conclude that the Bellare and Rogaway proof model rules out proofs of protocol that provide key confirmation. However, this is not entirely true as shown in a recent work of Choo and Hitchcock [24] whereby they show that a weaker version of the key confirmation goal is achievable in the setting of the reductionist proof approach for protocols. An example of such a key egreement protocol providing key confirmation, which is proven secure in the Bellare and Rogaway (1993) model is provided by Blake-Wilson, Johnson, and Menezes [10].

The remainder of this paper is structured as follows: Section 2 provides an informal overview of the proof model in which we work in. Section 3 describes the protocols that will be used as case studies, demonstrates previously unpublished attacks on these protocols, and presents the improved protocols. Section 4 presents the proof sketches for the improved protocols. Section 5 presents the conclusions.

Table 1: Informal description of the oracle queries

Send (U_1, U_2, i, m)	This query to oracle $\Pi^i_{U_1,U_2}$ computes a response according to the protocol specification and decision on whether to accept or reject yet, and returns them to the adversary $\mathcal{A}$. If the client oracle, $\Pi^i_{U_1,U_2}$, has either accepted with some session key or terminated, this will be made known to $\mathcal{A}$.
Reveal (U_1, U_2, i)	The client oracle, $\Pi^i_{U_1,U_2}$, upon receiving this query and if it has accepted and holds some session key, will send this session key back to $\mathcal{A}$. This query is known as a Session-Key Reveal in the Canetti–Krawczyk model [16].
Corrupt (U_1)	This query allows $\mathcal{A}$ to corrupt the principal U_1 at will, and thereby learn the complete internal state of the corrupted principal.
Test (U_1, U_2, i)	This query is the only oracle query that does not correspond to any of $\mathcal{A}$'s abilities. If $\Pi^i_{U_1,U_2}$ has accepted with some session key and is being asked a Test(U_1, U_2, i) query, then depending on a randomly chosen bit b, $\mathcal{A}$ is given either the actual session key or a session key drawn randomly from the session key distribution.

2 The Proof Model

In this section, an informal overview of the Bellare–Rogaway model [6, 7, 9] is presented. In the Bellare–Rogaway model, the adversary $\mathcal{A}$ is defined to be a probabilistic machine that is in control of all communications between parties by interacting with a set of $\Pi^i_{U_1,U_2}$ oracles (i.e., $\Pi^i_{U_1,U_2}$ is defined to be the i^{th} instantiation of a principal U_1 in a specific protocol run and U_2 is the principal with whom U_1 wishes to establish a secret key). The oracle queries are shown in Table 1.

Note that in the original Bellare–Rogaway model proposed in 1993 [7], the Corrupt query is not allowed. However, we consider the Bellare–Rogaway model which allows the adversary access to a Corrupt query because later proofs of security in the Bellare–Rogaway model [1, 10, 11, 18, 32, 34, 42] allow the Corrupt query. The omission of such a (Corrupt) query may also allow a protocol vulnerable to insider and unknown key share attacks [27] to be proven secure in the model [20].

Security depends on the notions of partnership of oracles and indistinguishability of session keys. The definition of partnership is used in the definition of security to restrict the adversary's Reveal and Corrupt queries to oracles that are not partners of the oracle whose key the adversary is trying to guess.

2.1 Definition of Partnership

Partnership is defined using session identifiers (SIDs) where SIDs are suggested to be the concatenation of messages exchanged during the protocol run. In this model, an oracle who has accepted will hold the associated session key, a SID and a partner identifier (PID). Definition 1 describes the definition of partnership in the Bellare–Rogaway model proposed in 2000 [6]. Note that any oracle that has accepted will have at most one partner, if any at all.

Definition 1 (Definition of Partnership). *Two oracles, $\Pi^i_{A,B}$ and $\Pi^j_{B,A}$, are partners if, and only if, both oracles have accepted the same session key with the same*

[1] Some might argue that a proof in the random oracle model is more of a heuristic proof than a real one. However, despite the criticism, no one has yet provided a convincing contradiction to the practicality of the random oracle model. This model is still widely accepted by the cryptographic community. We remark that recently, the first practical and provable-secure oblivious transfer password-based protocol whose proof of security relies on the random oracle model was published by Gentry, MacKenzie, and Ramzan in ACM CCS 2005 [28]. Moreover, in many applications, a very efficient protocol with a heuristic security proof is preferred over a much less efficient one with a complete security proof [17].

SID, have agreed on the same set of principals (i.e. the initiator and the responder of the protocol), and no other oracles besides $\Pi^i_{A,B}$ *and* $\Pi^j_{B,A}$ *have accepted with the same SID.*

SIDs are unique and known to everyone (including $\mathcal{A}$). Hence, session keys cannot be included as part of SIDs in the protocols.

2.2 Definition of Freshness

Freshness is used to identify the session keys about which $\mathcal{A}$ ought not to know anything because $\mathcal{A}$ has not revealed any oracles that have accepted the key and has not corrupted any principals knowing the key. Definition 2 describes freshness, which depends on the notion of partnership. Note that we do not consider the notion of forward secrecy in this paper, otherwise, the definition of freshness would be slightly different.

Definition 2 (Definition of Freshness). *Oracle* $\Pi^i_{A,B}$ *is fresh (or holds a fresh session key) at the end of execution, if, and only if, (1)* $\Pi^i_{A,B}$ *has accepted with or without a partner oracle* $\Pi^j_{B,A}$*, (2) both* $\Pi^i_{A,B}$ *and* $\Pi^j_{B,A}$ *oracles have not been sent a* Reveal *query, and (3) A and B have not been sent a* Corrupt *query.*

2.3 Definition of Security

Security in the four models is defined using the game $\mathcal{G}$, played between $\mathcal{A}$ and a collection of player oracles. $\mathcal{A}$ runs the game $\mathcal{G}$, whose setting is explained in Table 2.

Table 2: Setting of game $\mathcal{G}$

Stage 1:	$\mathcal{A}$ is able to send any oracle queries at will.
Stage 2:	At some point during $\mathcal{G}$, $\mathcal{A}$ will choose a fresh session on which to be tested and send a Test query to the fresh oracle associated with the test session. Depending on the randomly chosen bit b, $\mathcal{A}$ is given either the actual session key or a session key drawn randomly from the session key distribution.
Stage 3:	$\mathcal{A}$ continues making any oracle queries at will but cannot make Corrupt or Reveal queries that trivially expose the test session key.
Stage 4:	Eventually, $\mathcal{A}$ terminates the game simulation and outputs a bit b', which is its guess of the value of b.

Success of $\mathcal{A}$ in $\mathcal{G}$ is quantified in terms of $\mathcal{A}$'s advantage in distinguishing whether $\mathcal{A}$ receives the real key or a random value. $\mathcal{A}$ wins if, after asking a $\mathsf{Test}(U_1, U_2, i)$ query, where $\Pi^i_{U_1,U_2}$ is fresh and has accepted with the same session key, $\mathcal{A}$'s guess bit b' equals the bit b selected during the $\mathsf{Test}(U_1, U_2, i)$ query. Let the advantage function of $\mathcal{A}$ be denoted by $\mathsf{Adv}^{\mathcal{A}}(\mathsf{k})$, where $\mathsf{Adv}^{\mathcal{A}}(\mathsf{k}) = 2 \times Pr[b = b'] - 1$.

Definition 3 describes the definition of security for the Bellare–Rogaway model.

Definition 3 (BR93 Definition of Security [7]). *A protocol is secure in the Bellare–Rogaway model if both the following requirements are satisfied:*

1) *When the protocol is run between two oracles* $\Pi^i_{A,B}$ *and* $\Pi^j_{B,A}$ *in the absence of a malicious adversary, both* $\Pi^i_{A,B}$ *and* $\Pi^j_{B,A}$ *accept and hold the same session key.*

2) *For all PPT adversaries* $\mathcal{A}$*, (a) If uncorrupted oracles* $\Pi^i_{A,B}$ *and* $\Pi^j_{B,A}$ *complete matching sessions, then both* $\Pi^i_{A,B}$ *and* $\Pi^j_{B,A}$ *must hold the same session key, and (b)* $\mathsf{Adv}^{\mathcal{A}}(\mathsf{k})$ *is negligible.*

For the Bellare–Rogaway model, if both oracles $\Pi^i_{A,B}$ *and* $\Pi^j_{B,A}$ *have accepted, then the probability that oracle* $\Pi^j_{B,A}$ *does not engage in a matching conversation with oracle* $\Pi^i_{A,B}$ *is negligible.*

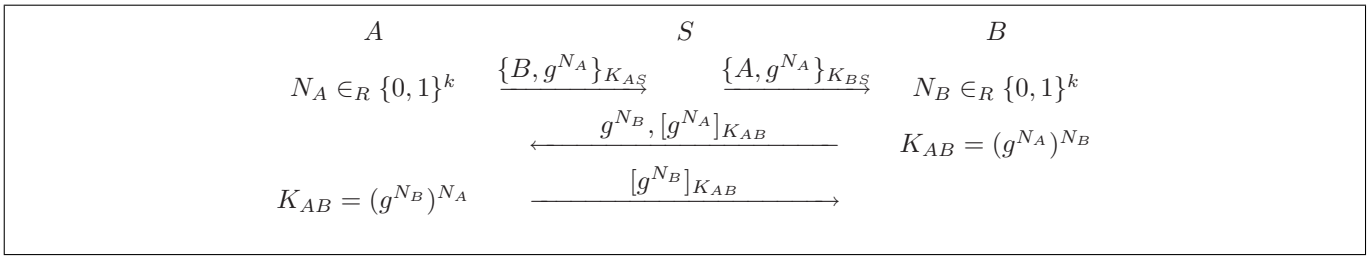
3 Mathuria and Jain (2005) Class of Key Agreement Protocols

Protocols 1, 2, and 3 describe the key agreement protcols of Mathuria and Jain [33]. The notation used throughout this section is as follows: the notation $\{\cdot\}_{K_U}$ denotes an encryption of some message m under U's public key, K_U, $[\cdot]_K(m)$ denotes the computation of MAC digest of some message m under key K, and K_{AB} denote the shared secret session key established by both A and B at the end of the protocols' execution.

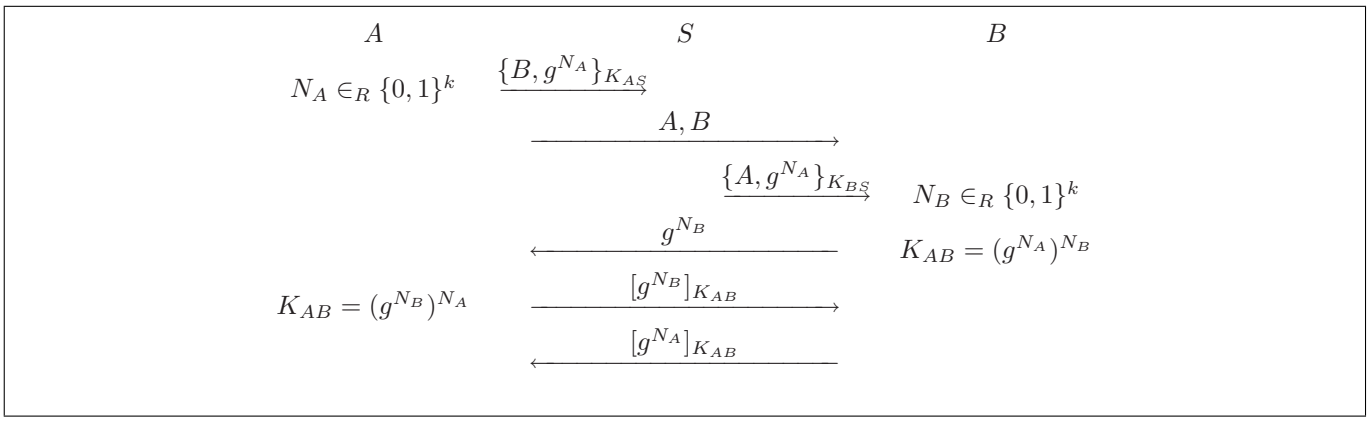
3.1 New Attacks

Attacks 1, 2, and 3 describes example executions of Protocols 1, 2, and 3 in the presence of a malicious adversary, $\mathcal{A}$.

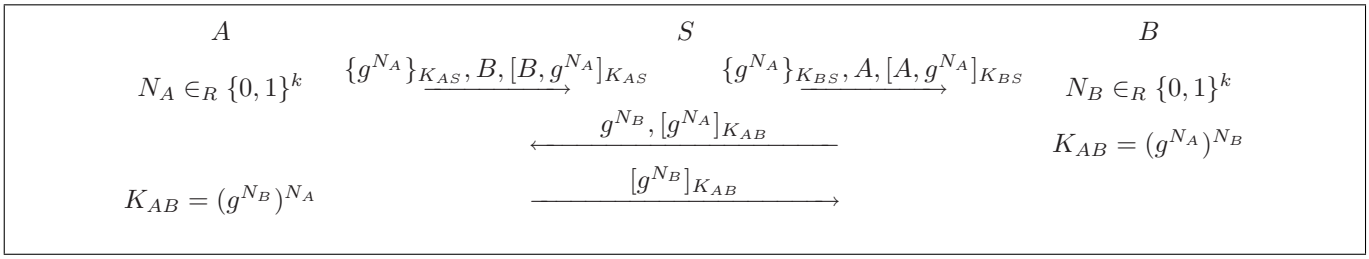
Let the notation $\Pi^i_{U_1,U_2}$ denote the i^{th} instantiation of a principal U_1 in a specific protocol run and U_2 is the principal with whom U_1 wishes to establish a secret key $\Pi^{S1}_{A,B}$. At the end of the protocol executions shown in Attacks 1, 2, and 3, A has completed two concurrent sessions with B with two session keys of the same value, $K_{AB(2)} = (g^{N_A})^{N_{A2}}$, (in which A plays the role of an initiator in the original session and the role of a responder in the second session), when in fact, B knows nothing about any of these sessions. In other words, oracle $\Pi^{S1}_{A,B}$ and oracle $\Pi^{S2}_{B,A}$ have accepted the same session key $K_{AB(2)} = (g^{N_A})^{N_{A2}}$, but they are not partners since they both believe that the key is being shared with some instantiation of principal B. Trivially, the adversary is able to obtain a fresh session key of A by exposing a non-partner of A (or another session of A) using a Reveal query in the Bellare and Rogaway (1993,1995) models [7, 9] or Bellare, Pointcheval, and Rogaway (2000) model [6] or a



Protocol 1: Mathuria – Jain key agreement Protocol 1



Protocol 2: Mathuria – Jain key agreement Protocol 2



Protocol 3: Mathuria – Jain key agreement Protocol 3

1. $A \longrightarrow S:$ $\{B, g^{N_A}\}_{K_{AS}}$
2. $S \longrightarrow B:$ $\{A, g^{N_A}\}_{K_{BS}}$

The adversary $\mathcal{A}$ intercepts message $\{A, g^{N_A}\}_{K_{BS}}$ meant for B. $\mathcal{A}$ impersonate B to start a concurrent session with A.

$1(S2).$ $\mathcal{A}_B \longrightarrow S:$ $\{A, g^{N_A}\}_{K_{BS}}$
$2(S2).$ $S \longrightarrow A:$ $\{B, g^{N_A}\}_{K_{AS}}$
A, upon receiving this message, thinks that B wants to start a concurrent session. A then chooses $N_{A2} \in_R \{0,1\}^k$ and computes $K_{AB(2)} = (g^{N_A})^{N_{A2}}$
$3(S2).$ $A \longrightarrow B:$ $g^{N_{A2}}, [g^{N_A}]_{K_{AB(2)}}$

The adversary $\mathcal{A}$ intercepts message $g^{N_{A2}}, [g^{N_A}]_{K_{AB(2)}}$ meant for A and reflects message back to A, impersonating B.

3. $\mathcal{A}_B \longrightarrow A:$ $g^{N_{A2}}, [g^{N_A}]_{K_{AB(2)}}$
4. $A \longrightarrow B:$ $[g^{N_{A2}}]_{K_{AB(2)}}$

The adversary $\mathcal{A}$ intercepts message $[g^{N_{A2}}]_{K_{AB(2)}}$ meant for B and reflects message back to A, impersonating B.

$4(S2).$ $\mathcal{A}_B \longrightarrow A:$ $[g^{N_A}]_{K_{AB(2)}}$

Attack 1: Execution of Protocol 1 in the presence of a malicious adversary

1. $A \longrightarrow S:$ $\{B, g^{N_A}\}_{K_{AS}}$
2. $A \longrightarrow B:$ A, B

The adversary $\mathcal{A}$ intercepts message A, B meant for B.

3. $S \longrightarrow B:$ $\{A, g^{N_A}\}_{K_{BS}}$

The adversary $\mathcal{A}$ intercepts message $\{A, g^{N_A}\}_{K_{BS}}$ meant for B. $\mathcal{A}$ impersonate B to start a concurrent session with A.

$1(S2).$ $\mathcal{A}_B \longrightarrow S:$ $\{A, g^{N_A}\}_{K_{BS}}$
$2(S2).$ $\mathcal{A}_B \longrightarrow A:$ B, A
$3(S2).$ $S \longrightarrow A:$ $\{B, g^{N_A}\}_{K_{AS}}$
A, upon receiving this message, thinks that B wants to start a concurrent session. A then chooses $N_{A2} \in_R \{0,1\}^k$ and computes $K_{AB(2)} = (g^{N_A})^{N_{A2}}$
$4(S2).$ $A \longrightarrow B:$ $g^{N_{A2}}$

The adversary $\mathcal{A}$ intercepts message $g^{N_{A2}}$ meant for A and reflects message back to A, impersonating B.

4. $\mathcal{A}_B \longrightarrow A:$ $g^{N_{A2}}$
5. $A \longrightarrow B:$ $[g^{N_{A2}}]_{K_{AB(2)}}$

The adversary $\mathcal{A}$ intercepts message $[g^{N_{A2}}]_{K_{AB(2)}}$ meant for B and reflects message back to A, impersonating B.

$5(S2).$ $\mathcal{A}_B \longrightarrow A:$ $[g^{N_{A2}}]_{K_{AB(2)}}$

Attack 2: Execution of Protocol 2 in the presence of a malicious adversary

Session-Key Reveal query in Canetti and Krawczyk (2001) model [16][2].

3.2 Preventing the Attacks

The countermeasures are well studied and we may adopt the same approach by Choo, Boyd, & Hitchcock [21], who suggest that

- Including the identities of the participants and their roles in the key derivation function provides resilience against unknown key share attacks [14, Chapter 5.1.2] and reflection attacks [29], and
- Including the transcripts in the key derivation function provides freshness and data origin authentication.

[2]Krawczyk [30] termed such an attack as *key replicating attack*.

Hence, we propose to include the sender's and responder's identities and transcripts, $\mathcal{T}_U$ (i.e., concatenation of all messages sent and received), in the key derivation function, which will (effectively) bind the session key to all messages sent and received by both A and B, as shown below:

$$\begin{aligned} SK_{A(Fixed)} &= \mathcal{H}_0(A||B||\mathcal{T}_A||(g^{N_B})^{N_A}) \\ SK_{B(Fixed)} &= \mathcal{H}_0(A||B||\mathcal{T}_B||(g^{N_A})^{N_B}) \\ &= SK_{A(Fixed)}, \end{aligned}$$

where $\mathcal{H}_0$ denotes a secure hash function [26, 37] and $||$ denotes the concatenation of messages. Intuitively, the attacks outlined in outlined in Section 3.1 will no longer be valid as the session key agreed by both the initiator and the responder entities will differ if any of the following changes:

1. $A \longrightarrow S$: $\{g^{N_A}\}_{K_{AS}}, B, [B, g^{N_A}]_{K_{AS}}$
2. $S \longrightarrow B$: $\{g^{N_A}\}_{K_{BS}}, A, [A, g^{N_A}]_{K_{BS}}$

The adversary $\mathcal{A}$ intercepts message $\{g^{N_A}\}_{K_{BS}}, A, [A, g^{N_A}]_{K_{BS}}$ meant for B. $\mathcal{A}$ impersonate B to start a concurrent session with A.

$1(S2)$. $\mathcal{A}_B \longrightarrow S$: $\{g^{N_A}\}_{K_{BS}}, A, [A, g^{N_A}]_{K_{BS}}$

$2(S2)$. $S \longrightarrow A$: $\{g^{N_A}\}_{K_{AS}}, B, [B, g^{N_A}]_{K_{AS}}$

A, upon receiving this message, thinks that B wants to start a concurrent session. A then chooses $N_{A2} \in_R \{0,1\}^k$ and computes $K_{AB(2)} = (g^{N_A})^{N_{A2}}$

$3(S2)$. $A \longrightarrow B$: $g^{N_{A2}}, [g^{N_A}]_{K_{AB(2)}}$

The adversary $\mathcal{A}$ intercepts message $g^{N_{A2}}, [g^{N_A}]_{K_{AB(2)}}$ meant for A and reflects message back to A, impersonating B.

3. $\mathcal{A}_B \longrightarrow A$: $g^{N_{A2}}, [g^{N_A}]_{K_{AB(2)}}$
4. $A \longrightarrow B$: $[g^{N_{A2}}]_{K_{AB(2)}}$

The adversary $\mathcal{A}$ intercepts message $[g^{N_{A2}}]_{K_{AB(2)}}$ meant for B and reflects message back to A, impersonating B.

$4(S2)$. $\mathcal{A}_B \longrightarrow A$: $[g^{N_{A2}}]_{K_{AB(2)}}$

Attack 3: Execution of Protocol 3 in the presence of a malicious adversary

- The identities of the participants and their perceived roles, and
- the transcripts.

The fixed protocols are described by Protocols 4, 5, and 6. Let $\mathcal{H}_0$ and $\mathcal{H}_1$ denote some secure collision-resistant hash functions [25], and $||$ denote the concatenation of messages.

4 Proof Sketches

4.1 Theorem 1

Theorem 1. *Assuming G satisfies the Computational Diffie-Hellman (CDH) assumption, Protocol 1, is a secure key agreement protocol with key confirmation when $\mathcal{H}_0$ and $\mathcal{H}_1$ are modelled as random oracles and if the underlying message authentication scheme and encryption scheme are secure in the sense of existential unforgeability under adaptive chosen-message attack and indistinguishable under chosen-plaintext attack respectively.*

The validity of Protocol 1 is straightforward to verify and we concentrate on the indistinguishability requirement. The security is proved by finding a reduction to the security of the underlying message authentication scheme and the underlying encryption scheme.

Recall that the security of Protocol 1 is based on the CDH problem in the random oracle model. Informally, there are only two ways an adversary, $\mathcal{A}$, can get information about a particular session key $K_{ij} = \mathcal{H}_0(i||j||SID_i^k||g^{N_iN_j})$ either:

Case 1. the value SID_i^k has repeated at some point during the experiment (for the same pair of users), or

Case 2. $\mathcal{A}$ queries the random oracle on the point $i||j||SID_i^k||g^{N_iN_j}$.

Case 1 happens with probability upper bounded by $\frac{q_s}{q^2}$ (where q is the size of the group G and q_s is the upper bound on the number of the sessions in the game simulation, $\mathcal{G}$). Case 2 allows us to solve the CDH problem with probability related to that of $\mathcal{A}$'s success probability. The notation q_p denotes the upper bound of the number of parties in $\mathcal{G}$, and q_h denotes the upper bound of the number of hash queries that $\mathcal{A}$ ask in $\mathcal{G}$.

The proof concludes by observing that $\mathsf{Adv}^{\mathcal{A}}(k)$ is negligible when $\mathcal{H}_0$, and $\mathcal{H}_1$ are modelled as random oracles and if the underlying message authentication scheme and encryption scheme are secure in the sense of existential unforgeability under adaptive chosen-message attack and indistinguishable under chosen-plaintext attack respectively, and therefore Protocol 1 is also secure.

4.2 Theorems 2 and 3

Theorem 2. *Assuming G satisfies the Computational Diffie-Hellman (CDH) assumption, Protocol 2, is a secure key agreement protocol with key confirmation when $\mathcal{H}_0$ and $\mathcal{H}_1$ are modelled as random oracles and if the underlying message authentication scheme and encryption scheme are secure in the sense of existential unforgeability under adaptive chosen-message attack and indistinguishable under chosen-plaintext attack respectively.*

Theorem 3. *Assuming G satisfies the Computational Diffie-Hellman (CDH) assumption, Protocol 3, is a secure key agreement protocol with key confirmation when $\mathcal{H}_0$ and $\mathcal{H}_1$ are modelled as random oracles and if the underlying message authentication scheme and encryption scheme are secure in the sense of existential unforgeability under adaptive chosen-message attack and indistinguishable under chosen-plaintext attack respectively.*

The proof follows that of Section 4.1, and the only difference is the way that additional Send queries required to be simulated. However, the $\mathsf{Adv}^{\mathcal{A}}(k)$ is negligible when $\mathcal{H}_0$,

A	S	B
$N_A \in_R \{0,1\}^k$	$\{A, B, g^{N_A}\}_{K_{AS}} \longrightarrow$	$\longleftarrow \{A, B, g^{N_A}\}_{K_{BS}}$
		$N_B \in_R \{0,1\}^k$
		$MACKey_{AB} = \mathcal{H}_1(A\|\|B\|\|(g^{N_A})^{N_B})$
	$g^{N_B}, [\text{“1”}, B, A, g^{N_A}, g^{N_B}]_{MACKey_{AB}} \longleftarrow$	
		$K_{AB} = \mathcal{H}_0(A\|\|B\|\|(g^{N_A})^{N_B})$
$MACKey_{AB} = \mathcal{H}_1(A\|\|B\|\|(g^{N_B})^{N_A})$		
$K_{AB} = \mathcal{H}_0(A\|\|B\|\|(g^{N_B})^{N_A})$	$[\text{“2”}, A, B, g^{N_A}, g^{N_B}]_{MACKey_{AB}} \longrightarrow$	

Protocol 4: Improved Protocol 1

A	S	B
$N_A \in_R \{0,1\}^k$	$\{A, B, g^{N_A}\}_{K_{AS}} \longrightarrow$	
	$A, B \longrightarrow$	
	$\longleftarrow \{A, g^{N_A}\}_{K_{BS}}$	
		$N_B \in_R \{0,1\}^k$
		$K_{AB} = \mathcal{H}_0(A\|\|B\|\|(g^{N_A})^{N_B})$
	$\longleftarrow g^{N_B}$	
		$MACKey_{AB} = \mathcal{H}_1(A\|\|B\|\|(g^{N_A})^{N_B})$
$MACKey_{AB} = \mathcal{H}_1(A\|\|B\|\|(g^{N_A})^{N_B})$		
$K_{AB} = \mathcal{H}_0(A\|\|B\|\|(g^{N_B})^{N_A})$	$[g^{N_B}]_{MACKey_{AB}} \longrightarrow$	
	$\longleftarrow [g^{N_A}]_{K_{AB}}$	

Protocol 5: Improved Protocol 2

A	S	B
$N_A \in_R \{0,1\}^k$	$\{g^{N_A}\}_{K_{AS}}, B, [B, g^{N_A}]_{K_{AS}} \longrightarrow$	$\longleftarrow \{g^{N_A}\}_{K_{BS}}, A, [A, g^{N_A}]_{K_{BS}}$
		$N_B \in_R \{0,1\}^k$
		$MACKey_{AB} = \mathcal{H}_1(A\|\|B\|\|(g^{N_A})^{N_B})$
$MACKey_{AB} = \mathcal{H}_1(A\|\|B\|\|(g^{N_A})^{N_B})$	$\longleftarrow g^{N_B}, [g^{N_A}]_{MACKey_{AB}}$	
		$K_{AB} = \mathcal{H}_0(A\|\|B\|\|(g^{N_A})^{N_B})$
$K_{AB} = \mathcal{H}_0(A\|\|B\|\|(g^{N_B})^{N_A})$	$[g^{N_B}]_{MACKey_{AB}} \longrightarrow$	

Protocol 6: Improved Protocol 3

and $\mathcal{H}_1$ are modelled as random oracles and if the underlying message authentication scheme and encryption scheme are secure in the sense of existential unforgeability under adaptive chosen-message attack and indistinguishable under chosen-plaintext attack respectively, and therefore Protocol 2 and Protocol 3 are also secure.

5 Conclusion

Through a detailed study of the class of key agreement protocols due to Mathuria and Jain [33], we demonstrated previously unpublished flaws in these protocols (which do not have proofs of security), and proposed some simple fixes to the protocols. Proof sketches of these improved protocols in the Bellare–Rogaway model were presented providing protocol implementers assurance about the security properties of protocols.

Acknowledgments

The views and opinions expressed in this paper are solely of the author and research was performed while the author was with the Information Security Institute/Queensland University of Technology.

The author would like to thank the anonymous reviewers for providing valuable comments concerning this paper. Despite their invaluable assistance any errors remaining in this paper are solely attributed to the author.

References

[1] S. S. A.-Riyami, and K. G. Paterson, "Tripartite authenticated key agreement protocols from pairings," in *9th IMA Conference on Cryptography and Coding*, LNCS 2898, pp. 332-359, Springer-Verlag, 2003.

[2] M. Backes, *Cryptographically Sound Analysis of Security Protocols*, Ph.D. Thesis, Computer Science Department, Saarland University, Saarbruken, Apr. 2002.

[3] M. Backes and C. Jacobi, "Cryptographically sound and machine-assisted verification of security protocols," in *20th International Symposium on Theoretical Aspects of Computer Science - STACS 2003*, LNCS 2607, pp. 310-329, Springer-Verlag, 2003.

[4] F. Bao, "Security analysis of a password authenticated key exchange protocol," in *6th Information Security Conference - ISC 2003*, LNCS 2851, pp. 208-217, Springer-Verlag, 2003.

[5] F. Bao, "Colluding attacks to a payment protocol and two signature exchange schemes," in *Advances in Cryptology (Asiacrypt'04)*, LNCS 3329, pp. 417-429, Springer-Verlag, 2004.

[6] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks," in *Advances in Cryptology (Eurocrypt'00)*, LNCS 1807, pp. 139-155, Springer-Verlag, 2000.

[7] M. Bellare and P. Rogaway, "Entity authentication and key distribution," in *Advances in Cryptology (Crypto'93)*, LNCS 773, pp. 110-125, Springer-Verlag, 1993.

[8] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *1st ACM Conference on Computer and communications Security (ACM CCS'93)*, pp. 62-73, ACM Press, 1993.

[9] M. Bellare and P. Rogaway, "Provably secure session key distribution: The three party case," in *27th ACM Symposium on the Theory of Computing (ACM STOC'95)*, pp. 57-66, ACM Press, 1995.

[10] S. B.-Wilson, D. Johnson, and A. Menezes, "Key agreement protocols and their security analysis," in *6th IMA International Conference on Cryptography and Coding*, LNCS 1335, pp. 30-45, 1997.

[11] S. B.-Wilson and A. Menezes, "Security proofs for entity authentication and authenticated key transport protocols employing asymmetric techniques," in *Security Protocols Workshop*, LNCS 1361, pp. 137-158, Springer-Verlag, 1997.

[12] C. Boyd, "Towards a classification of key agreement protocols," in *8th IEEE Computer Security Foundations Workshop (CSFW'95)*, pp. 38-43, IEEE Computer Society Press, 1995.

[13] C. Boyd, "A class of flexible and efficient key management protocols," in *9th IEEE Computer Security Foundations Workshop (CSFW'96)*, pp. 2-8, IEEE Computer Society Press, 1996.

[14] C. Boyd and A. Mathuria, "Protocols for authentication and key establishment," Springer-Verlag, June 2003.

[15] A. Bundy, M. Jamnik, and A. Fugard, "What is a proof?," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 363, no. 1835, pp. 2377-2391, 2005.

[16] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *Advances in Cryptology (Eurocrypt'01)*, LNCS 2045, pp. 453-474, Springer-Verlag, 2001. (http://eprint.iacr.org/2001/040/)

[17] D. Catalano, D. Pointcheval, and T. Pornin, "Trapdoor hard-to-invert group isomorphisms and their application to password-based authentication," *Journal of Cryptology*, To Appear, 2006.

[18] L. Chen and C. Kudla, "Identity based authenticated key agreement protocols from pairings," in *16th IEEE ComputerSecurity Foundations Workshop (CSFW'03)*, pp. 219-233, IEEE Computer Society Press, 2003. (http://eprint.iacr.org/2002/184/)

[19] K. K. R. Choo, C. Boyd, and Y. Hitchcock, "Errors in computational complexity proofs for protocols," in *Advances in Cryptology (Asiacrypt'05)*, LNCS 3788, pp. 624-643, Springer-Verlag, 2005.

[20] K. K. R. Choo, C. Boyd, and Y. Hitchcock, "Examining indistinguishability-based proof models for key establishment protocols," in *Advances in Cryptology*

(Asiacrypt'05), LNCS 3788, pp. 585-604, Springer-Verlag, 2005.

[21] K. K. R. Choo, C. Boyd, and Y. Hitchcock, "On session key construction in provably secure protocols," in *1st International Conference on Cryptology in Malaysia (Mycrypt'05)*, LNCS 3715, pp. 116-131, Springer-Verlag, 2005.

[22] K. K. R. Choo, C. Boyd, and Y. Hitchcock, "The importance of proofs of security for key establishment protocols: Formal Analysis of Jan–Chen, Yang–Shen–Shieh, Kim–Huh–Hwang–Lee, Lin–Sun–Hwang, and Yeh–Sun protocols," *Computer Communications*, vol. 29, no. 15, pp. 2788-2797, 2006.

[23] K. K. R. Choo, C. Boyd, Y. Hitchcock, and G. Maitland, "On session identifiers in provably secure protocols: the bellare-rogaway three-party key distribution protocol revisited," in *4th Conference on Security in Communication Networks (SCN'04)*, LNCS 3352, pp. 352-367, Springer-Verlag, 2004.

[24] K. K. R. Choo and Y. Hitchcock, "Security requirements for key establishment proof models: Revisiting Bellare–Rogaway and Jeong–Katz–Lee protocols," in *10th Australasian Conference on Information Security and Privacy (ACISP'05)*, LNCS 3574, pp. 429-442, Springer-Verlag, 2005.

[25] I. Damgård, "Collision free hash functions and public key signature schemes," in *Advances in Cryptology (Eurocrypt'87)*, LNCS 304, pp. 203-216, Springer-Verlag, 1987.

[26] I. Damgård, "A design principle for hash functions," in *Advances in Cryptology (Crypto'89)*, LNCS 435, pp. 416-427, Springer-Verlag, 1989.

[27] W. Diffie, P. C. v. Oorschot, and M. J. Wiener, "Authentication and authenticated key exchange," *Journal of Designs, Codes and Cryptography*, vol. 2, pp. 107-125, 1992.

[28] C. Gentry, P. MacKenzie, and Z. Ramzan, "Password authenticated key exchange using hidden smooth subgroups," in *12th ACM Conference on Computer and Communications Security (ACM CCS'05)*, pp. 299-309, ACM Press, 2005.

[29] H. Krawczyk, "SIGMA: The 'SIGn-and-MAc' approach to authenticated Diffie-Hellman and its use in the IKE-Protocols," in *Advances in Cryptology (Crypto'03)*, LNCS 2729, pp. 400-425, Springer-Verlag, 2003.

[30] H. Krawczyk, "HMQV: A high-performance secure Diffie-Hellman protocol," in *Advances in Cryptology (Crypto'05)*, LNCS 3621, pp. 546-566, Springer-Verlag, 2005.

[31] I. Lakatos, *Proofs and Refutations: The Logic of Mathematical Discovery*, Cambridge University Press, 1976.

[32] P. D. MacKenzie and R. Swaminathan, *Secure Network Authentication with Password Identification*, Submitted to the IEEE P1363 Working Group, 1999.

[33] A. Mathuria and V. Jain, *On Efficient Key Agreement Protocols (Accessed online on 08 March 2005)*, Cryptology ePrint Archive, Report 2005/064, 2005. (http://eprint.iacr.org/2005/064)

[34] N. McCullagh and P. S. L. M. Barreto, "A new two-party identity-based authenticated key agreement," in *Cryptographers' Track at RSA Conference-CT-RSA 2005*, LNCS 3376, pp. 262-274, Springer-Verlag, 2005.

[35] C. Meadows, "Open issues in formal methods for cryptographic protocol analysis," in *DARPA Information Survivability Conference and Exposition*, vol. 2052, pp. 237-250, IEEE Computer Society Press, 2000.

[36] C. Meadows, "Formal methods for cryptographic protocol analysis: Emerging issues and trends," *IEEE Journal on Selected Area in Communications*, vol. 21, no. 1, pp. 44-54, 2003.

[37] R. C. Merkle, "One way hash functions and DES," in *Advances in Cryptology (Crypto'89)*, LNCS 435, pp. 428-446, Springer-Verlag, 1989.

[38] J. Nam, S. Kim, and D. Won, *Attacks on Bresson-Chevassut-Essiari-Pointcheval's Group Key Agreement Scheme*, Cryptology ePrint Archive, Report 2004/251, 2004. (http://eprint.iacr.org/2004/251/)

[39] V. Shoup, *On Formal Models for Secure Key Exchange (Version 4)*, Technical Report RZ 3120 (#93166), IBM Research, Zurich, 1999.

[40] V. Shoup, "OAEP reconsidered," in *Advances in Cryptology (Crypto'01)*, LNCS 2139, pp. 239-259, Springer-Verlag, 2001.

[41] Z. Wan and S. Wang, "Cryptanalysis of two password-authenticated key exchange protocols," in *9th Australasian Conference on Information Security and Privacy (ACISP'04)*, LNCS 3108, pp. 164-175, Springer-Verlag, 2004.

[42] D. S. Wong and A. H. Chan, "Efficient and mutually authenticated key exchange for low power computing devices," in *Advances in Cryptology (Asiacrypt'01)*, LNCS 2248, pp. 172-289, Springer-Verlag, 2001.

Kim-Kwang Raymond Choo is currently a Research Analyst with the Australian Institute of Criminology working on the criminological aspects of high tech crime. He received his Ph.D. from the Information Security Institute at Queensland University of Technology and has presented at several international and local conferences including ASIACRYPT 2005, 2006 IEEE Computer Security Foundations Workshop and 2005 Australasian Conference on Information Security and Privacy. His research has been widely cited, including a citation in a special publication of the U.S. National Institute of Standards and Technology and a citation in the 2005 Cryptographic Technique Monitoring Subcommittee report of the Japanese Government agency, Cryptography Research and Evaluation Committees. He has also served on the program committee for several international conferences and as a reviewer for several international conferences and journals including ASIACRYPT 2005 & 2006 and the ACM Computing Reviews.