Examining Indistinguishability-Based Proof Models for Key Establishment Protocols*

Kim-Kwang Raymond Choo & Colin Boyd & Yvonne Hitchcock

Information Security Institute
Queensland University of Technology
GPO Box 2434, Brisbane, QLD 4001, Australia
{k.choo,c.boyd,y.hitchcock}@qut.edu.au

Abstract. We examine various indistinguishability-based proof models for key establishment protocols, namely the Bellare & Rogaway (1993, 1995), the Bellare, Pointcheval, & Rogaway (2000), and the Canetti & Krawczyk (2001) proof models. We then consider several variants of these proof models, identify several subtle differences between these variants and models, and compare the relative strengths of the notions of security between the models. For each of the pair of relations between the models (either an implication or a non-implication), we provide proofs or counter-examples to support the observed relations. We also reveal a drawback with the original formulation of the Bellare, Pointcheval, & Rogaway (2000) model, whereby the Corrupt query is not allowed. As a case study, we use the Abdalla & Pointcheval (2005) three-party password-based key exchange protocol (3PAKE), which carries a proof of security in the Bellare, Pointcheval, & Rogaway (2000) model. We reveal a previously unpublished flaw in the protocol, and demonstrate that this attack would not be captured in the model due to the omission of the Corrupt query.

1 Introduction

Key establishment protocols are used for distributing shared keying material in a secure manner. However, despite their importance, the difficulties of 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 (Bao 2003, Pereira & Quisquater 2003, Wan & Wang 2004, Wong & Chan 2001, Zhang 2005). The treatment of computational complexity analysis adopts a deductive reasoning process whereby the emphasis is placed on a proven reduction from the problem of breaking the protocol to another problem believed to be hard. Such an approach for key establishment protocols was made popular by Bellare & Rogaway (1993) who provide the first formal definition for a model of adversary capabilities with an associated definition of security (which we refer to as the BR93 model in this paper). Since then, many research efforts have been oriented towards this end which have resulted in numerous protocols with accompanying computational proofs of security proposed in the literature.

The BR93 model has been further revised several times. In 1995, Bellare and Rogaway analysed a three-party server-based key distribution (3PKD) protocol (Bellare & Rogaway 1995) using an extension to the BR93 model, which we refer to as the BR95 model. A more recent revision to the model was proposed in 2000 by Bellare et al. (2000), hereafter referred to as the BPR2000 model. Collectively, the BR93, BR95, and BPR2000 models will be referred to as the Bellare–Rogaway models. In independent yet related work, Bellare et al. (1998) built on the BR93 model and introduced a modular proof model. However, some drawbacks with this formulation were discovered and this modular proof model was subsequently modified by Canetti & Krawczyk (2001), and will be referred to as the CK2001 model in this paper.

* The abridged version of this paper is going to appear in the proceedings of Asiacrypt 2005, LNCS 3788/2005 (pp. 585–604).

Proof Models. There are several important differences between the BR93, BR95, BPR2000, and CK2001 models (which have a significant impact on the security of the models), as follows:

1. the way partner oracles are defined (i.e., the definition of partnership),
2. the powers of the probabilistic, polynomial-time (PPT) adversary,
3. the modular approach adopted in the CK2001 model, and
4. the provable security goals provided by the models.

DIFFERENCE 1: Security in the models depends on the notions of partnership of oracles and indistinguishability of session keys. The BR93 model defines partnership using the notion of matching conversations, where a conversation is a sequence of messages exchanged between some instances of communicating oracles in a protocol run. Partnership in the BR95 model is defined using the notion of a partner function, which uses the transcript (the record of all Send oracle queries) to determine the partner of an oracle by providing a mapping between two oracles that should share a secret key on completion of the protocol execution. However, such a partner definition can easily go wrong. One such example is the partner function described in the original BR95 paper for the 3PKD protocol (Bellare & Rogaway 1995), which was later found to be flawed (Choo et al. 2004).

The BPR2000 model and the CK2001 model define partnership using the notion of session identifiers (SIDs). Although in the BPR2000 model, the construction of SIDs is suggested to be the concatenation of messages exchanged during the protocol run, protocol designers can construct SIDs differently. There is no formal definition of how SIDs should be defined in the CK2001 model. Instead, SIDs are defined to be some unique values agreed upon by two communicating parties prior to the protocol execution. We observe that the way SIDs are constructed can have an impact on the security of the protocol in the model.

DIFFERENCE 2: The CK2001 model enjoys the strongest adversarial power (compared to the Bellare–Rogaway models) as the adversary is allowed to ask the Session-State Reveal query that will return all the internal state (including any ephemeral parameters but not long-term secret parameters) of the target session to the adversary. In contrast, most models only allow the adversary to reveal session keys for uncorrupted parties. In the original BR93 and BPR2000 models, the Corrupt query (that allows the adversary to corrupt any principal at will, and thereby learn the complete internal state of the corrupted principal) is not allowed.

In this paper, we consider the BR93 model which allows the adversary access to a Corrupt query because later proofs of security in the BR93 model (Al-Riyami & Paterson 2003, Blake-Wilson et al. 1997, Blake-Wilson & Menezes 1997, Chen & Kudla 2003, MacKenzie & Swaminathan 1999, McCullagh & Barreto 2005, Wong & Chan 2001) allow the Corrupt query. However, we consider the original BPR2000 model without Corrupt query because the basic notion of BPR2000 freshness restricts the adversary, $\mathcal{A}$, from corrupting anyone in the model (i.e., effectively restricting $\mathcal{A}$ from asking any Corrupt query). However, we show that the omission of such a (Corrupt) query in the BPR2000 model allows an insecure protocol to be proven secure in the model.

DIFFERENCE 3: A major advantage of the CK2001 model is its modular approach whereby protocols may be proven secure in an ideal world (AM) model in which the passive adversary is prevented from fabricating messages coming from uncorrupted principals, and translating such a protocol proven secure in the AM into one that is secure in the more realistic real world model (the UM). As Boyd et al. (2004) have pointed out, the CK2001 modular approach facilitates an engineering approach to protocol design, where protocol components may be combined by "mix and match" to tailor to the application at hand (analogous to a Java API library).

DIFFERENCE 4: Both the BR93 and BPR2000 models provide provable security for entity authentication & key distribution, whilst the BR95 model provides provable security for only the key distribution. Intuitively, protocols that provide both entity authentication and key distribution are "stronger" than protocols that provide only key distribution. In this paper, we refer to the BR93 and BPR2000 models that provide provable security for only key distribution as BR93 (KE) and BPR2000 (KE) respectively, and the BR93 and BPR2000 models that provide provable security for both entity authentication & key distribution as BR93 (EA+KE) and BPR2000 (EA+KE) respectively.

Motivations. We are motivated by the observations that no formal study has been devoted to the comparisons of relations and relative strengths of security between the Bellare–Rogaway and the Canetti–Krawczyk models. Although Shoup (1999) provides a brief discussion on the Bellare–Rogaway models and the Canetti–Krawczyk model, his discussion is restricted to an informal comparison between the Bellare–Rogaway model and his model, and between the Canetti–Krawczyk model and his model. To the best of our knowledge, no distinction has ever been made between the Bellare–Rogaway proof model and its variants shown in Table 1.

Bellare–Rogaway (Bellare et al. 2000, Bellare & Rogaway 1993, 1995)			
$\swarrow$		$\downarrow$	$\searrow$
BR93		BR95	BPR2000
$\swarrow$	$\searrow$	$\swarrow$	$\searrow$
BR93 (KE)	BR93 (EA+KE)	BPR2000 (KE)	BPR2000 (EA+KE)

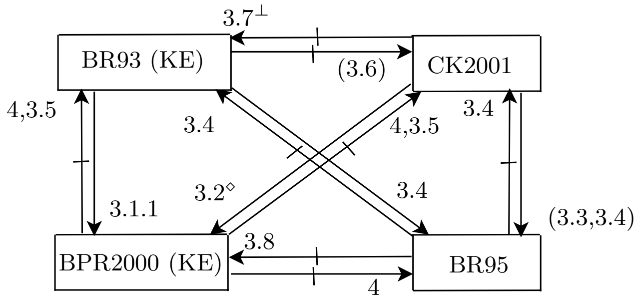
Table 1. The Bellare–Rogaway proof model and its variants

Contributions. We regard the main contributions of this paper to be three-fold:

1. contributing towards a better understanding of the different flavours of proof models for key establishment protocols by working out the relations between the Bellare–Rogaway proof model (and its variants) and the Canetti–Krawczyk proof model,
2. demonstrating that the Bellare–Rogaway (and its variants) and the Canetti–Krawczyk proof models have varying security strength by providing a comparison of the relative strengths of the notions of security between them, and
3. identifying a drawback in the BPR2000 model (not identified in any previous studies) which allows an insecure protocol to be proven secure in the BPR2000 model, as presented in Section 4.

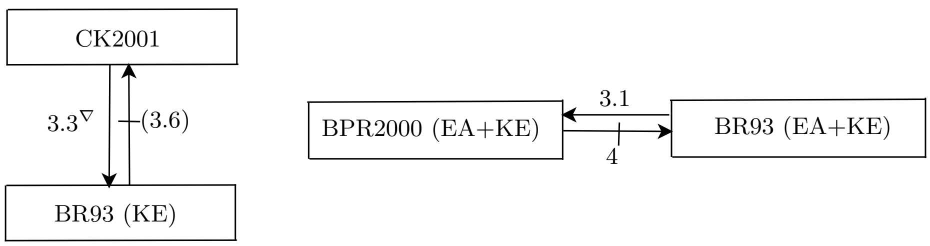
This work may ease the understanding of future security protocol proofs (protocols proven secure in one model may be automatically secure in another model), and protocol designers can make an informed decision when choosing an appropriate model in which to prove their protocols secure. Our main results are summarized in Figures 1 and 2. We observe that if SIDs in the CK2001 model are defined to be the concatenation of messages exchanged during the protocol run, then the implication CK2001 $\rightarrow$ BR93 holds, and the CK2001 model offers the strongest definition of security compared to the BR93 model.

The notation $x \rightarrow y$ denotes that protocols proven secure in model x will also be secure in model y (i.e., implication relation where x implies y). The notation $x \nrightarrow y$ denotes that protocols proven secure in model x do not necessarily satisfy the definition of security in model y. The numbers on the arrows represent the sections in which the proof is provided, and the numbers in brackets on the arrows represent the sections in which the implication relation is proven.



◇ holds if SIDs are constructed in the same manner in both models.
⊥ holds if SIDs are not defined to be the concatenation of messages exchanged during the protocol run.

Fig. 1. Notions of security between the Bellare–Rogaway and Canetti–Krawczyk key establishment proof models



∇ holds if SIDs are defined to be the concatenation of messages exchanged during the protocol run.

Fig. 2. Additional comparisons

Organization. Section 2 provides an informal overview of the Bellare-Rogaway and Canetti–Krawczyk models. Section 3 provides the proofs of the implication relations and counter-examples the for non-implication relations shown in Figures 1 and 2. In these counter-examples, we demonstrate that these protocols though secure in the existing proof model (in which they are proven secure) are insecure in another "stronger" proof model. Section 4 presents the drawback in the original formulation of the BPR2000 model by using a three-party password-based key exchange protocol (3PAKE) due to Abdalla & Pointcheval (2005) as a case study. Section 5 presents the conclusions.

2 The Proof Models

In this section, an overview of the Bellare–Rogaway (Bellare et al. 2000, Bellare & Rogaway 1993, 1995) and Canetti–Krawczyk models (Bellare et al. 1998, Canetti & Krawczyk 2001) is provided primarily for demonstrating the gaps in the relations and the relative strengths of security between the variants of the Bellare–Rogaway and the Canetti–Krawczyk models.

Adversarial Powers. In the Bellare-Rogaway and Canetti–Krawczyk models, the adversary $\mathcal{A}$ is defined to be a probabilistic machine that is in control of all communications between parties via the predefined oracle queries described below:

Send: This query computes a response according to the protocol specification and decision on whether to accept or reject yet, and returns them to $\mathcal{A}$.

Session-Key Reveal(U_1, U_2, i): Oracle $\Pi^i_{U_1,U_2}$, upon receiving a Session-Key Reveal 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 Reveal(U_1, U_2, i) query in the Bellare–Rogaway models.

Session-State Reveal: Oracle $\Pi^i_{U_1,U_2}$, upon receiving a Session-State Reveal(U_1, U_2, i) query and if it has neither accepted nor held some session key, will return all its internal state (including any ephemeral parameters but not long-term secret parameters) to $\mathcal{A}$.

Corrupt: Corrupt(U_1, K_E) query allows $\mathcal{A}$ to corrupt the principal U_1 at will, and thereby learn the complete internal state of the corrupted principal. The corrupt query also gives $\mathcal{A}$ the ability to overwrite the long-lived key of the corrupted principal with any value of her choice (i.e. K_E).

Test: Test(U_1, U_2, i) 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.

Table 2 provides a comparison of the types of queries allowed for the adversary between the various BR93, BR95, BPR2000, and CK2001 models.

Oracle Queries	BR93	BR95	BPR2000	CK2001
Send	*Yes*	*Yes*	*Yes*	*Yes*
Session-Key Reveal	*Yes*	*Yes*	*Yes*	*Yes*
Session-State Reveal	*No*	*No*	*No*	*Yes*
Corrupt	*Yes*	*Yes*	*No*	*Yes*
Test	*Yes*	*Yes*	*Yes*	*Yes*

Table 2. Summary of adversarial powers

Definition of Freshness. The notion of freshness of the oracle to whom the Test query is sent remains the same for the Bellare–Rogaway and Canetti–Krawczyk models. 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 1 describes freshness, which depends on the respective partnership definitions.

Definition 1 (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 (or* Session-State Reveal *in the CK2001 model), and*
3. *both A and B have not been sent a* Corrupt *query.*

The basic notion of freshness (i.e., does not incorporate the notion of forward secrecy) in the BPR2000 model requires that no one (including A and B in requirement 3 of Definition 1) in the model has been sent a Corrupt query. This effectively restricts $\mathcal{A}$ from asking any Corrupt query in the (BPR2000) model.

Definition of Security. Security in the Bellare–Rogaway and the Canetti–Krawczyk models is defined using the game $\mathcal{G}$, played between a malicious adversary $\mathcal{A}$ and a collection of $\Pi^i_{U_x,U_y}$ oracles for players $U_x, U_y \in \{U_1, \ldots, U_{N_p}\}$ and instances $i \in \{1, \ldots, N_s\}$. The adversary $\mathcal{A}$ runs the game $\mathcal{G}$, whose setting is explained in Table 3.

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 $\mathsf{Corrupt}$ and/or $\mathsf{Session\text{-}Key\ Reveal}$ and/or $\mathsf{Session\text{-}State\ Reveal}$ queries (depending on the individual proof model) 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.

Table 3. Setting of game $\mathcal{G}$

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, $\mathcal{A}$'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

$$\mathsf{Adv}^{\mathcal{A}}(\mathsf{k}) = 2 \times \mathsf{Pr}[\mathsf{b} = \mathsf{b}'] - 1.$$

2.1 The Bellare-Rogaway Models

2.1.1 The BR93 Model Partnership is defined using the notion of matching conversations, where a conversation is defined to be the sequence of messages sent and received by an oracle. The sequence of messages exchanged (i.e., only the Send oracle queries) are recorded in the transcript, T. At the end of a protocol run, T will contain the record of the Send queries and the responses. Definition 2 describes security for the BR93 model.

Definition 2 (BR93 Security) *A protocol is secure in the BR93 model if for all PPT adversaries $\mathcal{A}$,*

1. *if uncorrupted oracles $\Pi^i_{A,B}$ and $\Pi^j_{B,A}$ complete with matching conversations, then the probability that there exist i, j such that $\Pi^i_{A,B}$ accepted and there is no $\Pi^j_{B,A}$ that had engaged in a matching session is negligible, and*
2. $\mathsf{Adv}^{\mathcal{A}}(\mathsf{k})$ *is negligible. If both requirements are satisfied, then $\Pi^i_{A,B}$ and $\Pi^j_{B,A}$ will also have the same session key.*

Requirement 1 of Definition 2 implies entity authentication, whereby entity authentication is said to be violated if some fresh oracle terminates with no partner.

2.1.2 The BR95 Model Partnership in the BR95 model is defined using the notion of a partner function, which uses the transcript (the record of all Send oracle queries) to determine the partner of an oracle. However, no explicit definition of partnership was provided in the original paper since there is no single partner function fixed for any protocol. Instead, security is defined predicated on the existence of a suitable partner function.

Definition 3 describes security for the BR95 model.

Definition 3 (BR95 Security) *A protocol is secure in the BR95 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, and*
2. *for all PPT adversaries $\mathcal{A}$,* $\mathsf{Adv}^{\mathcal{A}}(\mathsf{k})$ *is negligible.*

2.1.3 The BPR2000 Model Partnership in the BPR2000 model is defined based on the notion of 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 4 describes partnership in the BPR2000 model.

Definition 4 (BPR2000 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 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.*

In the BPR2000 model, security is described in Definition 5. The notion of security for entity authentication is said to be violated if some fresh oracle terminates with no partner.

Definition 5 (BPR2000 Security) *A protocol is secure in the BPR2000 model under the notion of*

- *key establishment if for all PPT adversaries $\mathcal{A}$, $\mathsf{Adv}^{\mathcal{A}}(\mathsf{k})$ is negligible.*
- *mutual authentication if for all PPT adversaries $\mathcal{A}$, the advantage that $\mathcal{A}$ has in violating entity authentication is negligible.*

2.2 The Canetti-Krawczyk Model

In the CK2001 model, there are two adversarial models, namely the unathenticated-links adversarial / real world model (UM) and the authenticated-links adversarial / ideal world model (AM). Let $\mathcal{A}_{\mathrm{UM}}$ denote the (active) adversary in the UM, and $\mathcal{A}_{\mathrm{AM}}$ denote the (passive) adversary in the AM. The difference between $\mathcal{A}_{\mathrm{AM}}$ and $\mathcal{A}_{\mathrm{UM}}$ lies in their powers, namely $\mathcal{A}_{\mathrm{AM}}$ is restricted to only delay, delete, and relay messages but not to fabricate any messages or send a message more than once. Prior to explaining how a provably secure protocol in the AM is translated to a provably secure protocol in the UM with the use of an authenticator, we require definitions of an emulator and an authenticator, as given in Definitions 6 and 7.

Definition 6 (Definition of an Emulator (Bellare et al. 1998)) *Let π and π' be two protocols for n parties where π is a protocol in the AM and π' is a protocol in the UM. π' is said to emulate π if for any UM-adversary $\mathcal{A}_{UM}$ there exists an AM-adversary $\mathcal{A}_{AM}$, such that for all inputs, no polyomial time adversary can distinguish the cumulative outputs of all parties and the adversary between the AM and the UM with more than negligible probability.*

Definition 7 (Definition of an Authenticator (Canetti & Krawczyk 2001)) *An authenticator is defined to be a mapping transforming a protocol π_{AM} in the* AM *to a protocol π_{UM} in the* UM *such that π_{UM} emulates π_{AM}.*

In other words, the security proof of the UM protocol in the CK2001 depends on the security proofs of the MT-authenticator used and that of the AM protocol. If any of these proofs break down, then the proof of the UM protocol is invalid.

Definitions 8 and 9 describe partnership and security for the CK2001 model.

Definition 8 (Matching Sessions) *Two sessions are said to be matching if they have the same session identifiers (SIDs) and corresponding partner identifiers (PIDs).*

Definition 9 (CK2001 Security) *A protocol is secure in the CK2001 model if for all PPT adversaries $\mathcal{A}$,*

1. *if two 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*
2. $\mathsf{Adv}^{\mathcal{A}}(\mathsf{k})$ *is negligible.*

3 Relating The Notions of Security

In our proofs for each of the implication relations shown in Figure 1, we construct a primary adversary, $\mathcal{PA}$, against the key establishment protocol in $\mathcal{PA}$'s model using a secondary adversary, $\mathcal{SA}$, against the same key establishment protocol in $\mathcal{SA}$'s model. $\mathcal{PA}$ simulates the view of $\mathcal{SA}$ by asking all queries of $\mathcal{SA}$ to the respective Send, Session-Key Reveal, Session-State Reveal, Corrupt, and Test oracles (to which $\mathcal{PA}$ has access), and forwards the answers received from the oracles to $\mathcal{SA}$. The specification of the simulation is given in Figure 3.
Note that Shoup (Shoup 1999, Remark 26) pointed out that an adversary $\mathcal{A}$ in the Bellare–Rogaway model wins the game if $\mathcal{A}$ is able to make two partner oracles accept different session keys without making any Reveal and Test queries. His findings are applicable to only the BR93 and CK2001 models where the definitions of security requires two partner oracles to accept with the same session key, as described in Definitions 2 and 9 respectively. However, this is not the case for the BR95 and BPR2000 models.

The notation in this section is as follows: $\{\cdot\}_{K^{enc}_{U_1U_2}}$ denotes the encryption of some message under the encryption key $K^{enc}_{U_1U_2}$, the notation $[\cdot]_{K^{MAC}_{U_1U_2}}$ denotes the computation of MAC digest of some message under the MAC key $K^{MAC}_{U_1U_2}$, and $Sig_{d_U}(\cdot)$ denotes the signature of some message under the signature key d_U, $\mathcal{H}$ denote some secure hash function, $||$ denote concatentation of messages, and pwd denote some secret password shared between two users.

3.1 Proving Implication Relation: BR93 (EA+KE) → BPR2000 (EA+KE)

Recall that the Corrupt query is not allowed in the BPR2000 model but is allowed in the BR93 model as shown in Table 2. Intuitively, the model with a greater adversarial power, especially one that allows the adversary access to the entire internal state of a player (i.e., via the Corrupt query), has a tighter definition of security than the model with a weaker adversarial power.

3.1.1 Proof for the key establishment goal: Let the advantage of some PPT adversary, $\mathcal{A}_{00}$, in the BPR2000 (EA+KE) model be $\mathsf{Adv}^{\mathcal{A}_{00}}$, and the advantage of some PPT adversary, $\mathcal{A}_{93}$, in the BR93 (EA+KE) model be $\mathsf{Adv}^{\mathcal{A}_{93}}$.

Lemma 1 *For any key establishment protocol, for any $\mathcal{A}_{00}$, there exists an $\mathcal{A}_{93}$, such that* $\mathsf{Adv}^{\mathcal{A}_{00}} = \mathsf{Adv}^{\mathcal{A}_{93}}$.

Proof (Lemma 1). An adversary $\mathcal{A}_{93}$ against the key establishment protocol in the BR93 (EA+KE) model is constructed using an adversary $\mathcal{A}_{00}$ against the same key establishment protocol in the BPR2000 (EA+KE) model, as shown in Figure 3. In other words, let $\mathcal{A}_{93}$ be the primary adversary and $\mathcal{A}_{00}$ be the secondary adversary where $\mathcal{A}_{93}$ simulates the view of $\mathcal{A}_{00}$. $\mathcal{A}_{93}$ asks all queries by $\mathcal{A}_{00}$ to the respective Send oracles, Session-Key Reveal oracles, and Test oracle (to which $\mathcal{A}_{93}$ has access), and

Queries	Actions
Send	$\mathcal{PA}$ is able to answer this query pertaining to any instance of a server or player by asking its Send oracle.
Session-Key Reveal	$\mathcal{PA}$ is restricted from asking a Session-Key Reveal query to the target test oracle or its partner in its own game. Similarly, $\mathcal{SA}$ faces the same restriction^R. Hence, $\mathcal{PA}$ is able to answer this query by asking its Reveal oracle and is able to simulate the Session-Key Reveal query perfectly.
Corrupt	$\mathcal{SA}$ is disallowed from asking a Corrupt query to the principal of the target test session or whom the target test session thinks it is communicating with in its own game. Similarly, the $\mathcal{PA}$ faces the same restriction. Hence, $\mathcal{PA}$ is able to answer this query by asking its Corrupt oracle and simulates the Corrupt query perfectly.
Test	If the following conditions are satisfied (under the assumption that both $\mathcal{PA}$ and $\mathcal{SA}$ choose the same Test session), then $\mathcal{PA}$ queries its Test oracle. The Test oracle randomly chooses a bit, b_{Test}, and depending on b_{00}, the Test oracle either returns the actual session key or a random key. $\mathcal{PA}$ then answers $\mathcal{SA}$ with the answer received from its Test oracle. Let b_{SA} be the final output of $\mathcal{SA}$ and $\mathcal{PA}$ will output b_{SA} as its own answer. $\mathcal{PA}$ succeeds and wins the game if $\mathcal{SA}$ does. – The Test sessions in both $\mathcal{PA}$'s and $\mathcal{SA}$'s simulations have accepted, and must be fresh. • Since $\mathcal{PA}$ is able to answer all Send, Session-Key Reveal, and Corrupt queries asked by $\mathcal{SA}$ as shown above, if the Test session in $\mathcal{SA}$'s simulation has accepted, so does the same Test session in $\mathcal{PA}$'s simulation. • Since $\mathcal{PA}$ faces the same restriction as $\mathcal{SA}$ of not able to reveal or corrupt an oracle or principal associated with the Test session, if the Test session in $\mathcal{SA}$'s simulation is fresh, so is the same Test session in $\mathcal{PA}$'s simulation.

R: subject to the following requirements:

1. Non-partners in the simulation of $\mathcal{SA}$ are also non-partners in the simulation of $\mathcal{PA}$ so that whatever we can reveal in the simulation of $\mathcal{SA}$, we can also reveal in the simulation of $\mathcal{PA}$. Alternatively, we require that partners in the simulation of $\mathcal{PA}$ are also partners in the simulation of $\mathcal{SA}$ so that whatever we cannot reveal in the simulation of $\mathcal{PA}$, we also cannot reveal in the simulation of $\mathcal{SA}$.
2. A fresh oracle in the simulation of $\mathcal{SA}$ is also a fresh oracle the simulation of $\mathcal{PA}$ so that whatever we cannot reveal in the simulation of $\mathcal{SA}$, we also cannot reveal in the simulation of $\mathcal{PA}$.

Fig. 3. Specification of simulation between the primary adversary and the secondary adversary

forwards the answers received from the oracles to $\mathcal{A}_{00}$. Eventually, $\mathcal{A}_{00}$ outputs a guess bit b_{00} and $\mathcal{A}_{93}$ will output b_{00} as its own answer. $\mathcal{A}_{93}$ succeeds and wins the game if $\mathcal{A}_{00}$ does.

In order to demonstrate that the primary adversary, $\mathcal{A}_{93}$, is able to answer the queries asked by the secondary adversary, $\mathcal{A}_{00}$, we need to satisfy requirements 1 and 2 described in Figure 3. Using the example protocol execution shown in Figure 4, B is said to have a matching conversation with A if, and only if, message m'_A received is the same message m_A (i.e., $m'_A = m_A$) sent by A, and A is said to have matching conversation (in the BR93 model) with B if, and only if, message m'_B received is the same message m_B (i.e., $m'_B = m_B$) sent by B. In the context of Figure 4, $sid_A = m_A||m'_B$ and $sid_B = m'_A||m_A$ (in the BPR2000 model), and $sid_A = sid_B$ if message m'_A received by B is the same message m_A (i.e., $m'_A = m_A$) sent by A, and message m'_B received by A is the same message m_B (i.e., $m'_B = m_B$) sent by B. Hence, if both A and B have matching conversations, then $sid_A = m_A||m'_B = m'_A||m_A = sid_B$. If A and B are BR93-secure protocols, then A and B will also accept with the same session key.

A				B
Choose some message m_A	$\xrightarrow{m}$	...	$\xrightarrow{m'_A}$	Receive some message m'_A
Receive some message m'_B	$\xleftarrow{m'_B}$	...	$\xleftarrow{m_B}$	Choose some message m_B

Fig. 4. An example protocol execution

Recall that the BPR2000 definition of partnership requires two oracles to accept with the same SID, corresponding PID, and the same key, in order to be considered partners. Now, if A and B do not have matching conversations, then A and B are not BR93 partners. This also implies that A and B are not BPR2000 partners since $sid_A \neq sid_B$. Since non-partners in the simulation of the secondary adversary, $\mathcal{A}_{00}$, are also non-partners in the simulation of the primary adversary, $\mathcal{A}_{93}$, requirement 1 (described in Figure 3) is satisfied.

An oracle is considered fresh in the BPR2000 model if it (or its associated partner, if such a partner exists) has not been asked a Reveal query. An oracle is considered fresh in the BR93 model if it (or its associated partner, if such a partner exists) has not been asked either a Reveal or a $\mathsf{Corrupt}$ query. It follows easily that a fresh oracle in the BPR2000 model is also fresh in the BR93 model. Both requirements 1 and 2 (described in Figure 3), therefore, are satisfied.

To analyse $\mathsf{Adv}^{\mathcal{A}_{93}}$, we first consider the case in which the Test oracle associated with $\mathcal{A}_{93}$ returns a random key. The probability of $\mathcal{A}_{00}$ guessing the correct b_{00} bit is $\frac{1}{2}$ since it cannot gain any information about the hidden b_{93} bit. We then consider the case where the Test oracle associated with $\mathcal{A}_{93}$ returns the actual session key. In this case, the proof simulation (of $\mathcal{A}_{00}$) is perfect and $\mathcal{A}_{93}$ runs $\mathcal{A}_{00}$ exactly in the game defining the security of $\mathcal{A}_{00}$. Therefore, if $\mathcal{A}_{00}$ has a non-negligible advantage, so does $\mathcal{A}_{93}$ (i.e., $\mathsf{Adv}^{\mathcal{A}_{93}} = \mathsf{Adv}^{\mathcal{A}_{00}}$). This is in violation of our assumption and Lemma 1 follows.

□

3.1.2 Proof for the entity authentication goal: By inspection of Definitions 2 and 5, the definitions for entity authentication in both the BR93 and BPR2000 models are equivalent. Entity authentication is said to be violated if some fresh oracle terminates with no partner. Following from our earlier proofs in Section 3.1.1, we define $\mathcal{A}_{93}$ to simulate the view of $\mathcal{A}_{00}$; that is, $\mathcal{A}_{93}$ does anything that $\mathcal{A}_{00}$

does. Since non-partners in the simulation of $\mathcal{A}_{00}$ are also non-partners in the simulation of $\mathcal{A}_{93}$, if $\mathcal{A}_{00}$ has a non-negligible probability in violating mutual authentication, so does $\mathcal{A}_{93}$. This is in violation of our assumption. The proof for entity authentication follows.

3.2 Proving Implication Relation: CK2001 → BPR2000 (KE)

Recall that one of the key differences between the BPR2000 and the CK2001 models is that the Canetti–Krawczyk adversary is allowed to ask the additional Session-State Reveal and Corrupt queries, as shown in Table 2. Intuitively, the model with a greater adversarial power has a tighter definition of security than the model with a weaker adversarial power. To support our observation, let the advantage of some PPT adversary in the BPR2000 (KE) model be $\mathsf{Adv}^{\mathcal{A}_{00KE}}$, and the advantage of some PPT adversary in the CK2001 model be $\mathsf{Adv}^{\mathcal{A}_{01}}$.

Lemma 2 *For any key establishment protocol and for any $\mathcal{A}_{00KE}$, there exists an $\mathcal{A}_{01}$, such that* $\mathsf{Adv}^{\mathcal{A}_{00KE}} = \mathsf{Adv}^{\mathcal{A}_{01}}$.

Proof. An adversary $\mathcal{A}_{01}$ against the security of a key establishment protocol in the CK2001 (UM) model is constructed using an adversary $\mathcal{A}_{01}$ against the security of the same key establishment protocol in the BPR2000 (EA+KE) model. The primary adversary, $\mathcal{A}_{01}$, runs the secondary adversary, $\mathcal{A}_{00KE}$, and has access to its Send oracles, Session-State Reveal oracles, Session-Key Reveal oracles, Corrupt oracles, and Test oracle.

Recall that we assume in Figure 1 that this relation holds if, and only if, SIDs for both the BPR2000 (KE) and CK2001 model are constructed in the same manner. If A and B are BPR2000 partners, then $sid_A = sid_B$ and A and B will also be partners in the CK2001 model, since $sid_A = sid_B$ implies that both A and B will have matching sessions. Hence, we can say that all CK2001 partners are also BPR2000 partners (under the assumption that SIDs for both the BPR2000 (KE) and CK2001 model are constructed in the same manner) and all partners of CK2001-secure protocols are also BPR2000 partners (recall that in CK2001 security, two partners within a secure protocol must accept the same session key). This implies requirement 1.

An oracle is considered fresh in the BPR2000 model if it (or its associated partner, if such a partner exists) has not been asked a Reveal query and an oracle is considered fresh in the CK2001 model if it (or its associated partner, if such a partner exists) has not been asked either a Reveal or a Corrupt query. Hence, it follows easily that a fresh oracle in the BPR2000 model is also fresh in the CK2001 model. Hence, both requirements 1 and 2 (described in Figure 3) are satisfied.

To analyse $\mathsf{Adv}^{\mathcal{A}_{01}}$, we first consider the case in which the Test oracle associated with $\mathcal{A}_{01}$ returns a random key. The probability of $\mathcal{A}_{00KE}$ guessing the correct b_{01} bit is $\frac{1}{2}$ since it cannot gain any information about the hidden b_{01} bit. We then consider the case where the Test oracle associated with $\mathcal{A}_{01}$ returns the actual session key. In this case, the proof simulation (of $\mathcal{A}_{00KE}$) is perfect and $\mathcal{A}_{01}$ runs $\mathcal{A}_{00KE}$ exactly in the game defining the security of $\mathcal{A}_{00KE}$. Therefore, if $\mathcal{A}_{00KE}$ has a non-negligible advantage, so does $\mathcal{A}_{01}$ (i.e., $\mathsf{Adv}^{\mathcal{A}_{00KE}} = \mathsf{Adv}^{\mathcal{A}_{01}}$ is also non negligible). In other words, if such an adversary, $\mathcal{A}_{00KE}$, exists, so does $\mathcal{A}_{01}$. This is in violation of our assumption and Lemma 2 follows.

□

3.3 Proving Implication Relation: CK2001 → BR93 (KE)

This proof follows on from Section 3.2. Let the advantage of some PPT adversary in the BR93 (KE) model, $\mathcal{A}_{93KE}$, be $\mathsf{Adv}^{\mathcal{A}_{93KE}}$.

Lemma 3 *For any key establishment protocol and for any $\mathcal{A}_{93KE}$, there exists an $\mathcal{A}_{01}$, such that* $\mathsf{Adv}^{\mathcal{A}_{93KE}} = \mathsf{Adv}^{\mathcal{A}_{01}}$.

Proof. We construct an adversary $\mathcal{A}_{01}$ against the security of a key establishment protocol in the CK2001 model using an adversary $\mathcal{A}_{93KE}$ against the security of the same key establishment protocol in the BR93 model. Since we assume that SIDs in the CK2001 model are defined to be the concatenation of messages exchanged during the protocol run (similar to how SIDs are defined in the proof that appears in Section 3.1), the discussion on the notion of partnership between the BPR2000 and BR93 models applies in the discussion on the notion of partnership between the CK2001 and BR93 models. Hence, we can say that all BR93 partners are also CK2001 partners and all CK2001 partners are also BR93 partners (under the assumption that SIDs in the CK2001 model are defined to be the concatenation of messages sent and received during the protocol execution). Therefore, $\mathcal{A}_{01}$ is able to simulate the view of $\mathcal{A}_{93KE}$. Note that since $\mathcal{A}_{93KE}$ is not allowed to ask any Session-State Reveal in the BR93 model, $\mathcal{A}_{93KE}$ will not be asking any such queries in the simulation.

To analyse $\mathsf{Adv}^{\mathcal{A}_{01}}$, we first consider the case in which the Test oracle associated with $\mathcal{A}_{01}$ returns a random key. The probability of $\mathcal{A}_{93KE}$ guessing the correct b_{01} bit is $\frac{1}{2}$ since it cannot gain any information about the hidden b_{01} bit. We then consider the case where the Test oracle associated with $\mathcal{A}_{01}$ returns the actual session key. In this case, the proof simulation (of $\mathcal{A}_{93}$) is perfect and $\mathcal{A}_{01}$ runs $\mathcal{A}_{93KE}$ exactly in the game defining the security of $\mathcal{A}_{93KE}$. If $\mathcal{A}_{93KE}$ has a non-negligible advantage, so does $\mathcal{A}_{01}$ (i.e., $\mathsf{Adv}^{\mathcal{A}_{01}} = \mathsf{Adv}^{\mathcal{A}_{93KE}}$ is also negligible), in violation of our assumption. Lemma 3 follows. □

3.4 Discussion on Implication Relation: BR93 (KE) → BR95 and Non-Implication Relations: BR93 (KE) $\not\leftarrow$ BR95 and CK2001 $\not\leftarrow$ BR95

In key establishment protocols proven secure in the BR93 and CK2001 models, two parties in the same session must accept the same session key, which we term a functional requirement. However, this functional requirement is not required in the BPR2000 and the BR95 models. Consider the scenario of an example execution of a BR95 provably-secure protocol in the presence of a malicious adversary that resulted in two partner oracles accepting different session keys. This scenario does not violate BR95 security described in Definition 3. Hence, this protocol is still secure in the BR95 model. However, when two partner oracles accept two different session keys, the BR93 and CK2001 security are violated. Hence, this same protocol is not secure in the BR93 (KE) and CK2001 model. One such example protocol is the Bellare–Rogaway 3PKD protocol proven secure in the BR95 model (Bellare & Rogaway 1995), as shown in Figure 5.

$$
\begin{array}{|l|}
\hline
1.\ \ A \longrightarrow B : R_A \\
2.\ \ B \longrightarrow S : R_A, R_B \\
3a.\ S \longrightarrow A : \{SK_{AB}\}_{K_{AS}^{enc}}, [A, B, R_A, \{SK_{AB}\}_{K_{AS}^{enc}}]_{K_{AS}^{MAC}} \\
3b.\ S \longrightarrow B : \{SK_{AB}\}_{K_{BS}^{enc}}, [A, B, R_B, \{SK_{AB}\}_{K_{BS}^{enc}}]_{K_{BS}^{MAC}} \\
\hline
\end{array}
$$

Fig. 5. Bellare–Rogaway 3PKD protocol

Figure 6 depicts an example execution of the Bellare–Rogaway 3PKD protocol in the presence of a malicious adversary. At the end of the protocol execution, both uncorrupted principals A and B accept different session keys (i.e., A accepts session key SK_{AB} and B accepts session key $SK_{AB,2}$). Both A and B, however, are BR93 partners since they have matching conversations. The BR93 security is violated since A and B accept different session keys. The protocol, therefore, is not secure in the BR93 model.

Similarly, CK2001 security is violated (in the sense of Definition 9) since both uncorrupted principals A and B have matching sessions, according to Definition 8 but accept different session keys (i.e., A accepts session key SK_{AB} and B accepts session key $SK_{AB,2}$). The Bellare–Rogaway 3PKD protocol, therefore, is also not secure in the CK2001 model. A similar observation was made by Choo & Hitchcock (2005), whereby it was shown that the (same) 3PKD protocol proven secure by Tin et al. (2003) in the CK2001 model is, in fact, insecure.

$$
\begin{array}{|l|}
\hline
1.\ \ A \longrightarrow B:\ \ R_A \\
2.\ \ B \longrightarrow S:\ \ R_A, R_B \\
3a.\ S \longrightarrow A:\ \ \{SK_{AB}\}_{K_{AS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB}\}_{K_{AS}^{enc}}]_{K_{AS}^{MAC}}, R_B \\
3b.\ S \longrightarrow B:\ \ \{SK_{AB}\}_{K_{BS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB}\}_{K_{BS}^{enc}}]_{K_{BS}^{MAC}} \\
\quad \mathcal{A} \text{ intercepts and deletes } \{SK_{AB}\}_{K_{BS}^{enc}},\ [A, B, R_B, \{SK_{AB}\}_{K_{BS}^{enc}}]_{K_{BS}^{MAC}}. \\
2.\ \ \mathcal{A}_B \longrightarrow S: R_A, R_B \\
3a.\ S \longrightarrow A:\ \ \{SK_{AB,2}\}_{K_{AS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB,2}\}_{K_{AS}^{enc}}]_{K_{AS}^{MAC}}, R_B \\
\mathcal{A} \text{ intercepts and deletes } \{SK_{AB,2}\}_{K_{AS}^{enc}},\ [A, B, R_A, \{SK_{AB,2}\}_{K_{AS}^{enc}}]_{K_{AS}^{MAC}}, R_B. \\
3b.\ S \longrightarrow B:\ \ \{SK_{AB,2}\}_{K_{BS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB,2}\}_{K_{BS}^{enc}}]_{K_{BS}^{MAC}} \\
\hline
\end{array}
$$

Fig. 6. Execution of Bellare–Rogaway 3PKD protocol in the presence of a malicious adversary

The attack we present on the Bellare–Rogaway 3PKD protocol is similar to the attack on the Otway–Rees key establishment protocol revealed by Fabrega et al. (1999), in which they showed that a malicious adversary is able to make the initiator and the responder agree on a different session key by asking a trusted third party (i.e., server) to create multiple session keys in response to the same message. Although Fabrega *et al.* were perhaps the first to reveal that two communicating parties in a protocol might not agree on the same key in the presence of a malicious adversary, they did not see this as a serious flaw. This, however, is a flaw in the BR93 and CK2001 models where the definitions of security require two partner oracles to accept with the same session key.

3.5 Proving Non-Implication Relation: BR93 (KE) / CK2001 $\not\leftarrow$ BPR2000 (KE)

As a counter-example, we revisit and use the improved (Bellare–Rogaway) three-party key distribution (3PKD) protocol due to Choo et al. (2004) which has a proof of security in the BPR2000 (KE) model. We then demonstrate that this protocol fails to satisfy the functional requirement. Consequently, the protocol is insecure in the BR93 (KE) and CK2001 models. Figure 7 desribes the CBHM-3PKD protocol, which was proven secure in the BPR2000 model. In the protocol, there are three entities, namely: a trusted server S and two principals A and B who wish to establish communication.
Figure 8 depicts an example execution of the CBHM-3PKD protocol in the presence of a malicious adversary. At the end of the protocol execution, both uncorrupted prinicpals A and B have matching sessions according to Definition 8. However, they have accepted different session keys (i.e., A accepts session key SK_{AB} and B accepts session key $SK_{AB,2}$). This violates Definitions 2 and 9, which implies that the 3PKD protocol is not secure under the BR93 (KE) and the CK2001 models. However, according

1.	$A \longrightarrow B : R_A$
2.	$B \longrightarrow S : R_A, R_B$
3a.	$S \longrightarrow A : \{SK_{AB}\}_{K_{AS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB}\}_{K_{AS}^{enc}}]_{K_{AS}^{MAC}}, R_B$
3b.	$S \longrightarrow B : \{SK_{AB}\}_{K_{BS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB}\}_{K_{BS}^{enc}}]_{K_{BS}^{MAC}}$

Fig. 7. Choo, Boyd, Hitchcock, & Maitland provably secure 3PKD protocol

to Definition 4, both A and B are not BPR2000 partners since they do not agree on the same session key and hence, the protocol does not violate the BPR2000 security (i.e., Definition 5).

1.	$A \longrightarrow B : \quad R_A$
2.	$B \longrightarrow S : \quad R_A, R_B$
3a.	$S \longrightarrow A : \quad \{SK_{AB}\}_{K_{AS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB}\}_{K_{AS}^{enc}}]_{K_{AS}^{MAC}}, R_B$
3b.	$S \longrightarrow B : \quad \{SK_{AB}\}_{K_{BS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB}\}_{K_{BS}^{enc}}]_{K_{BS}^{MAC}}$
	$\mathcal{A}$ intercepts and deletes $\{SK_{AB}\}_{K_{BS}^{enc}}$, $[A, B, R_B, \{SK_{AB}\}_{K_{BS}^{enc}}]_{K_{BS}^{MAC}}$.
2.	$\mathcal{A}_B \longrightarrow S : R_A, R_B$
3a.	$S \longrightarrow A : \quad \{SK_{AB,2}\}_{K_{AS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB,2}\}_{K_{AS}^{enc}}]_{K_{AS}^{MAC}}, R_B$
	$\mathcal{A}$ intercepts and deletes $\{SK_{AB,2}\}_{K_{AS}^{enc}}$, $[A, B, R_A, \{SK_{AB,2}\}_{K_{AS}^{enc}}]_{K_{AS}^{MAC}}$.
3b.	$S \longrightarrow B : \quad \{SK_{AB,2}\}_{K_{BS}^{enc}}, [A, B, R_A, R_B, \{SK_{AB,2}\}_{K_{BS}^{enc}}]_{K_{BS}^{MAC}}$

Fig. 8. Execution of CBHM-3PKD protocol in the presence of a malicious adversary

3.6 Proving Non-Implication Relation: CK2001 $\not\leftarrow$ BR93 (EA+KE)

We use the mutual authentication and key establishment protocol (MAKEP) due to Wong & Chan (2001), which was proven secure in the BR93 (EA+KE) model. Note that Figure 9 describes the corrected version of WC-MAKEP, where the computation of $\sigma = (r_A \oplus r_B)$ by A is replaced by $\sigma = (r_A \oplus r_B)||ID_B$. There are two communicating principals in MAKEP, namely the server B and the client of limited computing resources, A. A and B are each assumed to know the public key of the other party. At the end of the protocol execution, both A and B accept with session keys $SK_{AB} = \mathcal{H}(\sigma) = SK_{BA}$. Figure 10 depicts an example execution of MAKEP in the presence of a malicious adversary.

$A\ (a, g^a)$		$B\ (SK_B, PK_B)$
$r_A \in_R \{0,1\}^k, x = \{r_A\}_{PK_B}$		
$b \in_R \mathbb{Z}_q \setminus \{0\}, \beta = g^b$	$\xrightarrow{Cert_A, \beta, x}$	Decrypt x
$\sigma = (r_A \oplus r_B)$		$r_B \in \{0,1\}^k$
$ID_B, y = a\mathcal{H}(\sigma) + b \bmod q$	$\xleftarrow{\{r_B, ID_B\}_{r_A}}$	$g^y \stackrel{?}{=} (g^a)^{\mathcal{H}(\sigma)}\beta$
$SK_{AB} = \mathcal{H}(\sigma)$	$\xrightarrow{y}$	$SK_{BA} = \mathcal{H}(\sigma)$

Fig. 9. Wong–Chan MAKEP

$\mathcal{A}$ intercepts the message sent by A and sends a fabricated message, $Cert_{\mathcal{A}}, \beta \cdot g^e, x$, claiming the message originated from itself ($\mathcal{A}$). B, upon receiving the message, thinks that $\mathcal{A}$ (and not A) wants to establish a session, will respond as per protocol specification. $\mathcal{A}$ is then able to send a $\mathsf{Session-State\ Reveal}$ query to B, and knows the values of both r_A and r_B. Subsequently, A completes the protocol execution and accepts session key, $SK_{AB} = \mathcal{H}(\sigma_A)$, thinking that the key is being shared with B, when in fact, B knows nothing about this session. Since $\mathcal{A}$ obtains the values of r_B, $\mathcal{A}$ is able to compute $\mathcal{H}(r_A \oplus r_{\mathcal{A}})||ID_B) = SK_{AB}$, in violation of the key establishment goal (i.e., Definition 9). Hence, Wong–Chan MAKEP though secure in the BR93 (EA+KE) model, is insecure in the CK2001 model.

$$
\begin{array}{cclc}
A & & \qquad\mathcal{A} & B \\
 & \xrightarrow{Cert_A, \beta, x} & \text{Fabricate} & \xrightarrow{Cert_{\mathcal{A}}, \beta \cdot g^e, x} \\
 & \xleftarrow{\{r_B, ID_B\}_{r_A}} & \text{Intercept} & \xleftarrow{\{r_B, ID_B\}_{r_A}} \\
 & & & \xrightarrow{\text{Session} - \text{State Reveal}} \\
\sigma_A = (r_A \oplus r_{\mathcal{A}})||ID_B & \xleftarrow{\{r_{\mathcal{A}}, ID_B\}_{r_A}} & \text{Fabricate} & \xleftarrow{r_A, r_B} \\
y = a\mathcal{H}(\sigma_A) + b \ mod \ q & & & \\
SK_{AB} = \mathcal{H}(\sigma_A) & \xrightarrow{y} & \multicolumn{2}{l}{SK_{AB} = \mathcal{H}(\sigma_A) \text{ with knowledge of } r_A \text{ and } r_{\mathcal{A}}}
\end{array}
$$

Fig. 10. Execution of Wong–Chan MAKEP in the presence of a malicious adversary

3.7 Proving Non-Implication Relation: BR93 (KE) $\not\leftarrow$ CK2001

Canetti & Krawczyk prove the basic Diffie–Hellman protocol secure in the UM (Canetti & Krawczyk 2001). In order to prove BR93 (KE) $\not\leftarrow$ CK2001, we modified the (Canetti–Krawczyk) Diffie–Hellman protocol to include a redundant nonce N_{BA}, as shown in Figure 11. The modified Diffie–Hellman protocol does not authenticate the redundant nonce N_{BA}. Although N_{BA} is not authenticated, addition of N_{BA} does not affect the security of the protocol.

$$
\begin{array}{ccc}
A & & B \\
x \in \mathbb{Z}_q & \xrightarrow{A, sid, g^x} & y \in \mathbb{Z}_q \\
\text{Verify Signature} & \xleftarrow{B, sid, g^y, Sig_{d_B}(B, sid, g^y, g^x, A), N_{BA}} & y, N_{BA} \in \mathbb{Z}_q \\
SK_{AB} = g^{xy} & \xrightarrow{A, sid, g^y, Sig_{d_A}(A, sid, g^y, g^x, B), N_{BA}} & SK_{AB} = g^{xy}
\end{array}
$$

Fig. 11. A modified (Canetti–Krawczyk) Diffie–Hellman protocol

Figure 12 depicts an example execution of the (Canetti–Krawczyk) Diffie–Hellman protocol in the presence of a malicious adversary. Recall that we assume that the non-implication relation: BR93 (KE) $\not\leftarrow$ CK2001 holds if, and only if, SIDs in the CK2001 model are not defined to be the concatenation of messages exchanged during the protocol run, as shown in Figure 1. Let $\mathcal{A}_U$ denote $\mathcal{A}$ intercepting the message and sending a fabricated message impersonating U.

At the end of the protocol execution, both A and B are partners according to Definition 8, since they have matching SIDs and corresponding PIDs (i.e., $PID_A = B$ and $PID_B = A$). In addition, both uncorrupted A and B accept the same session key, $SK_{AB} = g^{xy} = SK_{BA}$. The CK2001 definition of security is not violated (in the sense of Definition 9). Both A and B, however, did not receive all of each other's messages (recall that messages in message round 2 and 3 are fabricated by $\mathcal{A}$) and neither A's

A	$\mathcal{A}$	A
$\xrightarrow{A, sid, g^x}$		$\xrightarrow{A, sid, g^x}$
$\xleftarrow{B, sid, g^y, Sig_{d_B}(B, sid, g^y, g^x, A), N_{\mathcal{A}}}$	$\mathcal{A}_A$	$\xleftarrow{B, sid, g^y, Sig_{d_B}(B, sid, g^y, g^x, A), N_{BA}}$
$\xrightarrow{A, sid, g^y, Sig_{d_A}(A, sid, g^y, g^x, B), N_{\mathcal{A}}}$	$\mathcal{A}_B$	$\xrightarrow{A, sid, g^y, Sig_{d_A}(A, sid, g^y, g^x, B), N_{BA}}$

Fig. 12. Execution of the modified (Canetti–Krawczyk) Diffie–Hellman protocol in the presence of a malicious adversary

nor B's replies were all in response to genuine messages by B and A respectively. Both A and B are not BR93 partners. $\mathcal{A}$, however, can obtain a fresh session key of either A or B by revealing non-partner instances of B or A respectively, in violation of BR93 security (Definition 2).

3.8 Discussion on Non-Implication Relation: BPR2000 (KE) $\not\leftarrow$ BR95

Recall that security in the models depend on the notion of partnership. No explicit definition of partnership, however, was provided in the BR95 model and there is no single partner function fixed for any protocol in the BR95 model. The flawed partner function for the 3PKD protocol described in the original BR95 paper was fixed by Choo et al. (2004). As Choo *et al.* have pointed out, however, there is no way to securely define an SID for the 3PKD protocol that will preserve the proof of security. Protocols that are secure in the BR95 model, therefore, may not necessarily be able to be proven secure in the BPR2000 (KE) model.

4 A Drawback in the Original Formulation of the BPR2000 Model

4.1 Case Study: Abdalla–Pointcheval 3PAKE

We revisit the protocol 3PAKE due to Abdalla & Pointcheval (2005), which carries a proof of security in the BPR2000 model, as shown in Figure 13. Let A and B be two clients who wish to establish a shared session key, SK, S be a trusted server, pwd_A (and pwd_B) denote the password shared between A and S (B and S respectively), $\mathcal{G}_1, \mathcal{G}_2$, and $\mathcal{H}$ denote random oracles, and l_r and l_k denote security parameters.

4.2 New Attack on Abdalla–Pointcheval 3PAKE

Figure 14 describes an execution of 3PAKE in the presence of a malicious adversary, $\mathcal{A}$. Let C be another client who has a shared password, pwd_C, with the server, S. Prior to the start of the communication initiated by A, $\mathcal{A}$ corrupts a non-related player, C (i.e., static corruption), thereby learning all internal states of C (including the shared password with S, pwd_C).

In the attack outlined in Figure 14, $\mathcal{A}$ intercepts the first message from A and changes the identity field in the message from A, B to A, C. $\mathcal{A}$ impersonates A and sends the fabricated message A, C, X^* to S. $\mathcal{A}$ impersonates C and sends another fabricated message C, A, E^* to S. S, upon receiving both messages, will respond as per protocol specification. At the end of the protocol execution, A believes that the session key, $SK_A = \mathcal{H}(A, B, S, T, K)$, is being shared with B. However, B is still waiting for S's reply, which will never arrive, since $\mathcal{A}$ has intercepted and deleted the message from the network. However, $\mathcal{A}$ is able to compute the fresh session key of A, since $\mathcal{A}$ is able to decrypt and obtain $K = g^{xre}$ and $SK_A = \mathcal{H}(A, B, S, T, K)$, since parameters A, B, S, and T (T is the transcript of the protocol execution)

A (pwd_A)	S (pwd_A, pwd_B)	B (pwd_B)
$x \in_R \mathbb{Z}_p, X = g^x$	$r \in_R \mathbb{Z}_p$	$y \in_R \mathbb{Z}_p, Y = g^y$
$pw_{A,1} = \mathcal{G}_1(pwd_A)$	$R \in_R \{0,1\}^{l_R}$	$pw_{B,1} = \mathcal{G}_1(pwd_B)$
$X^* = X \cdot pw_{A,1}$	$pw_{A,1} = \mathcal{G}_1(pwd_A)$	$Y^* = Y \cdot pw_{B,1}$
$\xrightarrow{A, B, X^*}$		$\xleftarrow{B, A, Y^*}$
	$pw_{B,1} = \mathcal{G}_1(pwd_B)$	
	$X = X^*/pw_{A,1}, Y = Y^*/pw_{B,1}$	
	$\overline{X} = X^r, \overline{Y} = Y^r$	
	$pw_{A,2} = \mathcal{G}_2(R, pwd_A, X^*)$	
	$pw_{B,2} = \mathcal{G}_2(R, pwd_B, Y^*)$	
$\xleftarrow{S, B, R, Y^*, \overline{Y}^*}$	$\overline{X}^* = \overline{X} \cdot pw_{B,2}, \overline{Y}^* = \overline{Y} \cdot pw_{A,2}$	$\xrightarrow{S, A, R, X^*, \overline{X}^*}$
$pw_{A,2} = \mathcal{G}_2(R, pwd_A, X^*)$		$pw_{B,2} = \mathcal{G}_2(R, pwd_B, Y^*)$
$\overline{Y} = \overline{Y}^*/pw_{A,2}, K = \overline{Y}^x = g^{xry}$		$\overline{X} = \overline{X}^*/pw_{B,2}, K = \overline{Y}^x = g^{xry}$
$T = (R, X^*, Y^*, \overline{X}^*, \overline{Y}^*)$		$T = (R, X^*, Y^*, \overline{X}^*, \overline{Y}^*)$
$SK_A = \mathcal{H}(A, B, S, T, K)$		$SK_B = \mathcal{H}(A, B, S, T, K)$

Fig. 13. Abdalla–Pointcheval 3PAKE

A (pwd_A)	$\mathcal{A}$	S (pwd_A, pwd_B, pwd_C)	$\mathcal{A}$	B (pwd_B)
		$\mathcal{A}$ corrupt C and obtain all internal states of C, including pwd_C		
$\xrightarrow{A, B, X^*}$	Intercept		Intercept	$\xleftarrow{B, A, Y^*}$
			$e \in_R \mathbb{Z}_p, E = g^e$ s.t. underlying value $E \neq 1$	
			$E^* = E \cdot \mathcal{G}_1(pwd_C)$	
	$\xrightarrow{A, C, X^*}$		$\xleftarrow{C, A, E^*}$	
		$pw_{A,1} = \mathcal{G}_1(pwd_A)$		
		$pw_{C,1} = \mathcal{G}_1(pwd_C)$		
		$X = X^*/pw_{A,1}, E = E^*/pw_{C,1}$		
		$\overline{X} = X^r, \overline{E} = E^r$		
		$pw_{A,2} = \mathcal{G}_2(R, pwd_A, X^*)$		
		$pw_{C,2} = \mathcal{G}_2(R, pwd_C, E^*)$		
		$\overline{X}^* = \overline{X} \cdot pw_{C,2}, \overline{E}^* = \overline{E} \cdot pw_{A,2}$		
	Intercept	$\xleftarrow{S, C, R, E^*, \overline{E}^*}$		
	$\xleftarrow{S, B, R, E^*, \overline{E}^*}$	$\xrightarrow{S, A, R, X^*, \overline{X}^*}$		
$pw_{A,2} = \mathcal{G}_2(R, pwd_A, X^*)$				$pw_{C,2} = \mathcal{G}_2(R, pwd_C, E^*)$
$\overline{E} = \overline{E}^*/pw_{A,2}, K = \overline{E}^x = g^{xre}$				$\overline{X} = \overline{X}^*/pw_{C,2}, K = \overline{E}^x = g^{xre}$
$T = (R, X^*, E^*, \overline{X}^*, \overline{E}^*)$				$T = (R, X^*, E^*, \overline{X}^*, \overline{E}^*)$
$SK_A = \mathcal{H}(A, B, S, T, K)$				$SK_C = \mathcal{H}(A, B, S, T, K)$

Fig. 14. Execution of 3PAKE in the presence of a malicious adversary

are public.

Consequently, protocol 3PAKE is insecure. This attack[1], however, cannot be detected in the existing BPR2000 model since Corrupt query is not allowed. Protocols proven secure in a proof model that allows the "Corrupt" query (in the proof simulation) ought to be secure against the unknown key share attack. If a key is to be shared between some parties, U_1 and U_2, the corruption of some other (non-related) player in the protocol, say U_3, should not expose the session key shared between U_1 and U_2 (Choo et al. 2005). Protocol 3PAKE, therefore, will be insecure in the BR93, BR95, and CK2001 models, since $\mathcal{A}$ is able to trivially expose a fresh session key (i.e., $\mathsf{Adv}^{\mathcal{A}}(k)$is non-negligible) by corrupting a non-partner player.

5 Conclusion and Future Work

We examined the Bellare–Rogaway and Canetti–Krawczyk proof models. We analysed some non-intuitive gaps in the relations and the relative strengths of security between both models and their variants. We then provided a detailed comparison of the relative strengths of the notions of security between the Bellare–Rogaway and Canetti–Krawczyk proof models. We also revealed a drawback with the BPR2000 model and a previously unpublished flaw in the Abdalla–Pointcheval protocol 3PAKE (Abdalla & Pointcheval 2005). Such an attack, however, would not be captured in the model due to the omission of Corrupt queries. Our studies concluded that

1. if the session identifier (SID) in the CK2001 model is defined to be the concatenation of messages exchanged during the protocol run, then the CK2001 model offers the strongest definition of security compared to the Bellare–Rogaway model and its variants, and
2. the BPR2000 model is the weakest model.

As a result of this work, we hope to have contributed towards a better understanding of the different flavours of proof models for key establishment protocols (whether protocols proven secure in one model are also secure in another model). While our studies focus only on the Bellare–Rogaway and Canetti–Krawczyk models, it would be interesting to extend our work to other computational complexity proof models (e.g., the proof model due to Shoup (1999)) or other simulation-based proof models (e.g., the universal composability approach and the black-box simulatability approach due to (Canetti 2000, Canetti & Fischlin 2001, Canetti & Krawczyk 2002) and Backes (2004*a*,*b*), Backes & Jacobi (2003), Backes et al. (2002, 2004), Backes & Schunter (2003) respectively).

Acknowledgements

This work was partially funded by the Australian Research Council Discovery Project Grant DP0345775. We would like to thank the anonymous referees of Asiacrypt 2005 for their comments.

[1] Informally, it appears that this attack can be avoided by including the identities of both A and B when computing $pw_{A,2}$ and $pw_{B,2}$.

Bibliography

Abdalla, M. & Pointcheval, D. (2005), Interactive Diffie-Hellman Assumptions with Applications to Password-based Authentication (Extended version available from `http://www.di.ens.fr/~pointche/pub.php`), *in* A. Patrick & M. Yung, eds, '9th International Conference on Financial Cryptography - FC 2005', Springer-Verlag, pp. 341–356. Volume 3570/2005 of Lecture Notes in Computer Science.

Al-Riyami, S. S. & Paterson, K. G. (2003), Tripartite Authenticated Key Agreement Protocols from Pairings, *in* K. G. Paterson, ed., '9th IMA Conference on Cryptography and Coding', Springer-Verlag, pp. 332–359. Volume 2898/2003 of Lecture Notes in Computer Science.

Backes, M. (2004*a*), 'A Cryptographically Sound Dolev-Yao Style Security Proof of the Needham–Schroeder–Lowe Public–Key Protocol', *IEEE Journal on Selected Areas in Communications* **22(10)**, 2075–2086.

Backes, M. (2004*b*), A Cryptographically Sound Dolev-Yao Style Security Proof of the Otway-Rees Protocol, *in* P. Samarati & D. Gollmann, eds, '9th European Symposium on Research in Computer Security - ESORICS 2004', Springer-Verlag, pp. 89–108. Volume 3193/2004 of Lecture Notes in Computer Science.

Backes, M. & Jacobi, C. (2003), Cryptographically Sound and Machine-Assisted Verification of Security Protocols, *in* H. Alt & M. Habib, eds, '20th International Symposium on Theoretical Aspects of Computer Science - STACS 2003', Springer-Verlag, pp. 310–329. Volume 2607/2003 of Lecture Notes in Computer Science.

Backes, M., Jacobi, C. & Pfitzmann, B. (2002), Deriving Cryptographically Sound Implementations Using Composition and Formally Verified Bisimulation, *in* L.-H. Eriksson & P. A. Lindsay, eds, 'Formal Methods - Getting IT Right', Springer-Verlag, pp. 310–329. Volume 2391/2002 of Lecture Notes in Computer Science.

Backes, M., Pfitzmann, B. & Waidner, M. (2004), A General Composition Theorem for Secure Reactive Systems, *in* M. Naor, ed., '1st Theory of Cryptography Conference - TCC 2004', Springer-Verlag, pp. 336–354. Volume 2951/2004 of Lecture Notes in Computer Science.

Backes, M. & Schunter, M. (2003), From Absence of Certain Vulnerabilities towards Security Proofs: Pushing the Limits of Formal Verification, *in* '10th ACM Workshop on New Security Paradigms - NSPW 2003', ACM Press, pp. 67–74.

Bao, F. (2003), Security Analysis of a Password Authenticated Key Exchange Protocol, *in* C. Boyd & W. Mao, eds, '6th Information Security Conference - ISC 2003', Springer-Verlag, pp. 208–217. Volume 2851/2003 of Lecture Notes in Computer Science.

Bellare, M., Canetti, R. & Krawczyk, H. (1998), A Modular Approach to The Design and Analysis of Authentication and Key Exchange Protocols, *in* J. Vitter, ed., '30th ACM Symposium on the Theory of Computing - STOC 1998', ACM Press, pp. 419–428.

Bellare, M., Pointcheval, D. & Rogaway, P. (2000), Authenticated Key Exchange Secure Against Dictionary Attacks, *in* B. Preneel, ed., 'Advances in Cryptology – Eurocrypt 2000', Springer-Verlag, pp. 139 – 155. Volume 1807/2000 of Lecture Notes in Computer Science.

Bellare, M. & Rogaway, P. (1993), Entity Authentication and Key Distribution, *in* D. R. Stinson, ed., 'Advances in Cryptology - Crypto 1993', Springer-Verlag, pp. 110–125. Volume 773/1993 of Lecture Notes in Computer Science.

Bellare, M. & Rogaway, P. (1995), Provably Secure Session Key Distribution: The Three Party Case, *in* F. T. Leighton & A. Borodin, eds, '27th ACM Symposium on the Theory of Computing - STOC 1995', ACM Press, pp. 57–66.

Blake-Wilson, S., Johnson, D. & Menezes, A. (1997), Key Agreement Protocols and their Security Analysis, *in* M. Darnell, ed., '6th IMA International Conference on Cryptography and Coding', Springer-Verlag, pp. 30–45. Volume 1355/1997 of Lecture Notes in Computer Science.

Blake-Wilson, S. & Menezes, A. (1997), Security Proofs for Entity Authentication and Authenticated Key Transport Protocols Employing Asymmetric Techniques, *in* B. Christianson, B. Crispo, T. M. A. Lomas & M. Roe, eds, 'Security Protocols Workshop', Springer-Verlag, pp. 137–158. Volume 1361/1997 of Lecture Notes in Computer Science.

Boyd, C., Mao, W. & Paterson, K. (2004), Key Agreement using Statically Keyed Authenticators, *in* M. Jakobsson, M. Yung & J. Zhou, eds, 'Applied Cryptography and Network Security - ACNS 2004', Springer-Verlag, pp. 248–262. Volume 3089/2004 of Lecture Notes in Computer Science.

Canetti, R. (2000), 'Universally Composable Security: A New Paradigm for Cryptographic Protocols', Cryptology ePrint Archive, Report 2000/067. `http://eprint.iacr.org/2000/067/`.

Canetti, R. & Fischlin, M. (2001), Universally Composable Commitments, *in* J. Kilian, ed., 'Advances in Cryptology – Crypto 2001', Springer-Verlag, pp. 19–40. Volume 2139/2001 of Lecture Notes in Computer Science.

Canetti, R. & Krawczyk, H. (2001), Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels (Extended version available from `http://eprint.iacr.org/2001/040/`), *in* B. Pfitzmann, ed., 'Advances in Cryptology - Eurocrypt 2001', Springer-Verlag, pp. 453–474. Volume 2045/2001 of Lecture Notes in Computer Science.

Canetti, R. & Krawczyk, H. (2002), Universally Composable Notions of Key Exchange and Secure Channels. (Extended version available from `http://eprint.iacr.org/2002/059/`), *in* L. R. Knudsen, ed., 'Advances in Cryptology - Eurocrypt 2002', Springer-Verlag, pp. 337–351. Volume 2332/2002 of Lecture Notes in Computer Science.

Chen, L. & Kudla, C. (2003), Identity Based Authenticated Key Agreement Protocols from Pairings (Corrected version at `http://eprint.iacr.org/2002/184/`), *in* '16th IEEE Computer Security Foundations Workshop - CSFW 2003', IEEE Computer Society Press, pp. 219–233.

Choo, K.-K. R., Boyd, C. & Hitchcock, Y. (2005), Errors in Computational Complexity Proofs for Protocols (Available from `http://sky.fit.qut.edu.au/~choo/publication.html`), *in* B. Roy, ed., '(Accepted to appear in) Advances in Cryptology - Asiacrypt 2005', Springer-Verlag. Lecture Notes in Computer Science.

Choo, K.-K. R., Boyd, C., Hitchcock, Y. & Maitland, G. (2004), On Session Identifiers in Provably Secure Protocols: The Bellare-Rogaway Three-Party Key Distribution Protocol Revisited (Extended version available from `http://eprint.iacr.org/2004/345`), *in* B. Carlo & S. Cimato, eds, '4th Conference on Security in Communication Networks - SCN 2004', Springer-Verlag, pp. 352–367. Volume 3352/2005 of Lecture Notes in Computer Science.

Choo, K.-K. R. & Hitchcock, Y. (2005), Security Requirements for Key Establishment Proof Models: Revisiting Bellare–Rogaway and Jeong–Katz–Lee Protocols (Extended version available from `http://sky.fit.qut.edu.au/~choo/publication.html`), *in* C. Boyd & J. M. González Nieto, eds, '10th Australasian Conference on Information Security and Privacy - ACISP 2005', Springer-Verlag, pp. 429–442. Volume 3574/2005 Lecture Notes in Computer Science.

Fabrega, F. J. T., Herzog, J. C. & Guttman, J. D. (1999), 'Strand Spaces: Proving Security Protocols Correct', *Journal of Computer Security* **7**, 191–230.

MacKenzie, P. D. & Swaminathan, R. (1999), 'Secure Network Authentication with Password Identification', Submitted to the IEEE P1363 Working Group.

McCullagh, N. & Barreto, P. S. L. M. (2005), A New Two-Party Identity-Based Authenticated Key Agreement (Extended version available from `http://eprint.iacr.org/2004/122/`), *in* A. J. Menezes, ed., 'Cryptographers' Track at RSA Conference - CT-RSA 2005', Springer-Verlag, pp. 262–274. Volume 3376/2005 of Lecture Notes in Computer Science.

Pereira, O. & Quisquater, J.-J. (2003), 'Some Attacks Upon Authenticated Group Key Agreement Protocols', *Journal of Computer Security* **11**, 555–580.

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

Tin, Y. S. T., Boyd, C. & González Nieto, J. M. (2003), Provably Secure Key Exchange: An Engineering Approach, *in* 'Australasian Information Security Workshop Conference on ACSW Frontiers 2003', Australian Computer Society, pp. 97–104. Volume 21 of Conferences in Research and Practice in Information Technology.

Wan, Z. & Wang, S. (2004), Cryptanalysis of Two Password-Authenticated Key Exchange Protocols, *in* H. Wang, J. Pieprzyk & V. Varadharajan, eds, '9th Australasian Conference on Information Security and Privacy - ACISP 2004', Springer-Verlag. Volume 3108/2004 of Lecture Notes in Computer Science.

Wong, D. S. & Chan, A. H. (2001), Efficient and Mutually Authenticated Key Exchange for Low Power Computing Devices, *in* C. Boyd, ed., 'Advances in Cryptology - Asiacrypt 2001', Springer-Verlag, pp. 172–289. Volume 2248/2001 of Lecture Notes in Computer Science.

Zhang, M. (2005), 'Breaking an Improved Password Authenticated Key Exchange Protocol for Imbalanced Wireless Networks', *IEEE Communications Letters* **9(3)**, 276–278.