Non-Interactive Conference Key Distribution and Its Applications

Reihaneh Safavi-Naini and Shaoquan Jiang*
Department of Computer Science
University of Calgary
{rei,sqjiang}@ucalgary.ca

ABSTRACT

A non-interactive conference key distribution system (or, a NICKDS for short) allows conference members to calculate a shared key without interacting with each other. NICKDSs have been studied in unconditional and computational settings. In both cases security has been evaluated against an adversary who can corrupt participants. In this paper we consider an adaptive adversary who can both corrupt participants and also access the keys of conference of his choice. We re-visit security of a number of known NICKDSs in this new model and present characterizations and conditions that guarantee security of the system in the new model. We also give a generic construction for computationally secure (in the new model) NICKDSs, from unconditionally secure ones in corruption only model.

To show the usefulness of the new security model, we consider two composition constructions. First, we compose a secure NICKDS with a secure MAC by using the key obtained from the NICKDS as the MAC key, and show that this results in a ring authentication that guarantees authenticity of the received message while the sender remains anonymous and this anonymity is unconditional. The security theorem for the composition guarantees security for unconditional and computational settings, both. We also consider composition of a NICKDS with a secure (CCA2 secure) encryption system and show this results in a broadcast encryption system (BES) that is CCA2 secure. This is the first CCA2 secure BES in symmetric key setting. We discuss future works and open problems.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General—*Security and Protection*

General Terms

Security

Keywords

Conference Key Distribution, Unconditional Security, Ring Authentication, Broadcast Encryption

1. INTRODUCTION

Securing applications such as a shared whiteboards, teleconferencing and collaborative authoring system, requires group members to share a secret key. A *conference (group) key distribution system (CKDS)* is a multiparty protocol that allows a group of users to obtain a shared secret key that is only known to group members.

In most applications including those above, the group that needs a shared secret key is *dynamic* and changes over time. A *dynamic conference key distribution protocol* provides a shared secret key for members of dynamically formed conferences. Common approaches to constructing dynamic CKDS are, (i) to provide two operations *join* to add new members and *revoke* and remove existing members, hence allowing the conference group to change, (ii) run a new instance of the conference key distribution protocol among the new conference members, and (iii) during initialization provide sufficient key material to users so that users can locally calculate the conference keys for all future conferences that they will be a member of. For examples of the three approaches, see [24], [7] and [10], respectively. The first two approaches require *interaction* among conference members and so we refer to them as *Interactive CKDS (ICKDS)*. The last one is *non-interactive*, and in this paper will be referred to as NICKDS. In this paper we are interested in NICKDS.

In computational setting, the celebrated key agreement protocol of Diffie and Hellman [16] can be seen as a *non-interactive* conference key distribution systems. In this system Alice uses her secret key k_A and Bob's public key K_B to calculate a shared secret key with Bob, who will be using a similar calculation on k_B and K_A to obtain the same key. For conferences of size larger than 2, the only known scheme is due to Joux [20] and uses bilinear pairing to non-interactively establish a common key among groups of size three. In both schemes each user is independently initialized and has access to the public key of other users. Construction of NICKDS for conferences of size greater than 3 has been a long standing open problem [8].

In information theoretic setting, NICKDSs were first introduced by Blom [6]. Blundo et al [7] used multi-variable

*Acknowledgement. Shaoquan Jiang has been supported as a postdoctoral fellow by two Informatics Circle of Research Excellence grants on Algorithmic Number Theory and Cryptography, and Information Security.

ASIACCS '08, March 18-20, Tokyo, Japan

symmetric polynomials to construct a dynamic NICKD with information theoretic security against a corruption only adversary. Their scheme allows any group of size at most r to share a key and the system is secure against up to w corrupted users. We denote this scheme by $SymPoly(r, w)$.

Security model for ICKDS [10] is an extension of Bellare-Rogaway model [4] and considers a fully adaptive adversary that can eavesdrop the communication, and has access to a number of oracles including `Corrupt` oracle that allows him to interactively corrupt users of his choice and obtain their secret keys, and `Reveal` oracle that allows him to obtain the keys of conferences of his choice. NICKDSs' security in information theoretic and computational models both, have been considered against a corruption-only adversary, i.e. an adversary that can only corrupt users. An adaptive corruption-only adversary can choose the next corrupted user after accessing the key information of users that are corrupted so far. Security in this model is not sufficient for many scenarios. For instance, if a member of a group $\mathcal{P}$ wants to send a private message M to the group, a natural approach is to use a NICKDS to obtain a conference key $k_{\mathcal{P}}$ for the group, compute a ciphertext $E_{k_{\mathcal{P}}}(M)$ using a strong encryption scheme E (e.g., with CCA2 security), and broadcast it. The adversary in this scenario may have access to ciphertexts $\{E_{k_{\mathcal{P}_i}}(M_i)\}$ w.r.t. receiver groups $\{\mathcal{P}_i\}$. Even if E is CCA2 secure, there is no guarantee for the secrecy of $\{M_i\}$. A similar problem occurs when we use a corruption-only secure NICKDS to agree on a shared key and with it to compute MACs to authenticate messages from the group.

1.1 Our Results

1.1.1 Secure NICKDS.

We propose a strong security model for NICKDS that is in line with the security model of ICKDSs in [4, 10]. We start with the model for ICKDS and remove the oracles that are not applicable in non-interactive case. We allow the adversary to have adaptive access to three types of queries: `Corrupt`, group key `Reveal`, and group key `Test`. The security goal is stated in terms of the key indistinguishability of a target conference selected by the attacker. The model and definitions can be used in computational and information theoretic settings both, with the distinction being through a parameter T that measures the computation time of the adversary.

We use this model to examine security of a number of known NICKDs. We start with the *unconditionally secure setting* and consider two widely considered NICKDSs due to Fiat and Naor [7, 19] and Blundo et [7]. Both these systems are information theoretically secure in corruption-only model. (We use the term *corruption-only* to emphasize that the adversary cannot make `Reveal` queries.) Although not explicitly mentioned, in both systems security is against adaptive corruption. We show that neither of these protocols remains secure when the adversary can access `Reveal` queries. Specifically, $SymPoly(r, w)$ [7] is provably secure in our model *if and only if* the number of group key `Reveal` queries t, plus the number of `Corrupt` queries c is not more than the corruption threshold w. In other words, the information leaked through a `Reveal` query is effectively the same as a `Corrupt` query. We note that this is intuitively true in one direction, i.e. one extra corrupted user reduces the corruption threshold by one. However it is less clear that a `Reveal` query has the same effect. In fact one would expect that a `Reveal` query to be less harmful since it only reveals one conference key while a `Corrupt` query enables the adversary to see the keys of all conferences that contain the corrupted user. We also show that if $t + c > w$, then $SymPoly(r, w)$ will not be secure in the new model.

The scheme in [19] allows conferences of any size to calculate a shared key and is secure against up to w corrupted users. In the rest of this paper we denote this scheme by $FN(\geq 2, w)$ where " ≥ 2" is used to emphasize any conference size at least 2. We show that in the new model and assuming w `Corrupt` queries, a single `Reveal` query will compromise security of the $FN(\geq 2, w)$ and will allow the adversary to completely determine the challenge conference key (note that the adversary chooses the challenge conference). However, if the size of the conferences (and hence the size of the queried conferences) is restricted to bigger or equal to $n - w$, then $FN(\geq n - w, w)$ will be provably secure against w `Corrupt` queries and any number of group key `Reveal` queries. Here n is the total number of users and "any" means all conference keys except the challenge (test) conference key. We denote this variant scheme by $FN(\geq n - w, w))$.

In the *computational setting*, we first show that the non-interactive DH protocol [16] is secure in the new model and the security relies on the bilinear decisional Diffie-Hellman assumption. Note that the proof for interactive DH protocol (See [11]) does not imply the security of the non-interactive DH in our model since a user's exponent in the latter will be re-used for different conferences while in the former, a user will choose independent exponents for different conferences. We then examine security of a pairing based two and three party protocol which is obtained by a modification of the tripartite protocol in [20]. We prove that this protocol is secure against any number of `Corrupt` and `Reveal` queries as long as the test conference key is not revealed.

One of the main results of this paper is providing a solution to the open problem of constructing a computationally secure NICKDS for conferences of size larger than 2. We give a general method of constructing a NICKDSs that is computationally secure in the random oracle model and assuming an adaptive adversary, from a NICKDS that is information theoretically secure in corruption-only model. The construction applies a hash function to the key derived in the information theoretically secure NICKDS to obtain the key for the computationally secure one. See Theorem 6 for a formal statement. This method can be used to obtain computationally secure NICKDS for arbitrary size conferences from $SymPoly(r, w)$ and $FN(\geq 2, w)$.

1.1.2 Applications

We motivated our work by arguing inadequacy of corruption only security model of NICKDSs in natural applications such as encryption and authentication of messages in group communication. We will show that the new security model for NICKDS provides exactly the required security property in both applications; the resulting systems have added properties that make them of high interest in practice.

The first application is *ring authentication* as defined in [31]. A ring authentication system is the symmetric key counterpart of ring signatures: a sender wants to send an authenticated message such that it appears as a message that is originated from one of the members of a group. The

group is chosen in an ad-hoc fashion by the sender allowing the sender to 'hide' himself in an *anonymity group* [13] of his choice. The sender's anonymity is unconditional. The difference with ring signature is that this is a symmetric key setting with users only holding secret key information (i.e., no public key) and verification only using the users' secret keys. In this scenario an authenticated message is only verifiable by a privileged group member that holds the required (secret) verification key. The primitive can be seen as authenticating a message for a group, hence similar to broadcast authentication although in this scenario group members are assumed trusted (unlike broadcast authentication). The security properties of ring authentication are *unforgeability* and *sender anonymity* where anonymity is information theoretic. We give formal definitions for these properties and show a generic composition construction that results in a ring authentication system with provable security. See Theorem 7 for a formal statement. In [31], authors considered ring authentication in information theoretic setting assuming corruption only adversaries, and proposed a generic construction that is proved to be secure in that model. Our construction when considered in information theoretic setting is similar to [31] but our security proof gives stronger guarantee, more specifically against an adaptive adversary.

The second application of NICKDS that we consider is sending an encrypted message that is decryptable by members of a group: an application reminiscent of *broadcast encryption* (BES) in symmetric key setting. In BES, a broadcast center wants to securely distribute data to an authorized subset of a receiver set. The authorized subset varies over time and the broadcast center must be able to target the broadcast to a subset of his choice. Broadcast encryption systems are studied in the public and symmetric key settings. In public key setting, Dodis and Fazio [17] formalized the strong notion of CCA2 security using an adaptive adversary with access to encryption and decryption oracles, and gave a construction that is secure in this model. They allowed up to a threshold of w receivers to be corrupted. The security model for BES in symmetric key setting is due to Naor et al [28] and provides CCA1 security. In their model, the adversary can access encryption and decryption queries before the challenge ciphertext is chosen but is not allowed to make decryption queries afterwards- hence CCA1 security. The adversary can however corrupt any number of users.

We consider an alternative model for BES in *symmetric key setting* by limitingthe number of corruption queries of the adversary to w, but allowing access to decryption oracle after seeing the challenge ciphertext, hence CCA2 security. In many scenarios this captures the security threats in practice. For example, in a pay-TV scenario it is unlikely that the adversary can corrupt *all* receivers except the test conference members (as allowed in the model of Naor et al), while it is a reasonable to assume he can corrupt up to w receivers but also has the keys of some (past) conferences.

We prove (Theorem 8) that the composition of a secure NICKDS and a CCA2 secure encryption system will result in a symmetric key BES with provable security in our proposed security model for BES. To our knowledge, this is the first CCA2 secure BES in *symmetric key setting*.

1.1.3 Open problems and extensions

Our work on NICKDS can be extended in a number of directions. In information theoretic setting, [7] obtained bounds on key sizes of users assuming corruption-only adversary. Deriving similar results in our model (i.e., adaptive adversary with access to `Reveal` queries) and proposing efficient and secure NICKDSs in this model are open problems. In computational setting, our construction of NICKDS for arbitrary size conferences is secure in random oracle model. Constructing computationally secure NICKDS in standard model is also an open problem. It will also be interesting to investigate other applications of secure NICKDS for secure group communication.

1.2 Related work

A *key pre-distribution schemes* can be seen as a NICKDS. Blom [6] proposed a key pre-distribution scheme using Maximum Distance Separable (MDS) codes only useable for conferences of size two. The scheme is secure if w users are corrupted. Matsumoto and Imai [25] extended Blom's scheme using general symmetric functions and applicable to conferences of size larger than two. Blundo et al. [7] gave a concrete example of [25] approach using symmetric multivariable polynomials. Their scheme is proven secure against w corrupted users. Fiat and Naor [19] used a combinatorial approach to build key distribution schemes. All these schemes (including those not listed here) are proven secure against corruption-only attack ([25] does not provide a formal security proof). We noted the inadequacy of this security model and proposed a stronger security model for NICDS. The usefulness of our security notions can be clearly seen by the two application areas: ring authentication and broadcast encryption. Related works on these two primitives are outlined below.

Ring signature. Rivest, Shamir and Tauman [30] proposed the notion of ring signature in which a signer chooses a group of users and generates a signature that appears to have been generated by one of the group members, hence providing anonymity for the signer. This anonymity can be shown to be unconditional. In a ring signature, each user has a registered signature scheme and hence a public key. However the system does not need a special setup and the signer can form the anonymity group in an ad-hoc way. Ring signatures have been widely studied in recent years; See [9, 18, 5, 14]. A related signature scheme is *group signature* [13] in which a signer signs on behalf of a group and remains anonymous as a member of that group. However, anonymity in group signatures is usually revocable and the signer's identity can be recovered by the group manager or another designated entity. Group signatures require a system set-up phase and the group is usually fixed at that stage.

Naor [27] introduced deniability in ring authentication by further requiring the communication to be simulateable by the adversary. The protocol is interactive but does not require a receiver to have a secret key. However the security is provable with an assumption on timing of messages, that is assuming that messages must be sent or received within a specific time frame. The protocol provides anonymity for the sender as a member of a ring, and deniability of the communication.

We are only interested in efficient authentication that provides sender anonymity by hiding the sender identity as a member of a ring and do not consider deniability of the communication.

In [31] *unconditionally secure ring authentication* was pro-

posed and a generic construction by composing a NICKDS and an authentication code was given. Authors proved security of the composition in the corruption-only model and against non-adaptive adversaries. Our work strengthens this result by proving security of the construction against an adaptive adversary. We also show a similar construction in computational setting. We refer to this construction as "ring authentication" emphasizing that we do not require simulatability (and hence deniability) of the communication.

Broadcast encryption. Broadcast encryption systems (BES) were first studied by Fiat and Naor [19]. A large body of work (e.g., [32, 33, 26]) has been on finding key assignment schemes that ensure colluders cannot learn the key of the sessions that they do not belong to. This key is used to encrypt a message for the targeted receivers. Security of this message was considered by Dodis and Fazio [17] In public key setting, and a construction that provides CCA2 security was given. In symmetric key setting, Naor et al [28] formalized CCA1 security of BES for a key assignment framework that is called the subset cover framework. We consider CCA2 security of the BES in *symmetric key setting* but unlike Naor et al, assume the number of colluders is bounded by c.

2. PRELIMINARIES

For a set R, $|R|$ denotes the number of elements in R; $e \leftarrow R$ means taking a random element e from R. $a|b$ means concatenation of a and b. PPT refers to *probabilistic polynomial time* and κ is the security parameter. We will use U_i to denote a user and $\mathcal{U}$ to denote a universe of users $\{U_1, \cdots, U_n\}$. For a set $\mathcal{P} \subset \mathcal{U}$ we define $\bar{\mathcal{P}} = \mathcal{U} \backslash \mathcal{P}$. $\mathbb{F}_q$ is a finite field of size q.

2.1 Non-Interactive Conference Key Distribution Systems

A *non-interactive conference key distribution system* (NICKDS) is a cryptographic system defined over a group of users $\mathcal{U}$ and a *Conference Structure* $\Gamma = \{\mathcal{P}_1, \cdots, \mathcal{P}_\gamma\}$ where $\mathcal{P}_i \subset \mathcal{U}, i = 1, \cdots, \gamma$. It includes two phases: a key distribution and a key derivation. In the key distribution phase a trusted authority $\mathbb{T}$ assigns a secret key K_i to each user U_i.

During the key derivation phase a user U_i uses K_i to calculate a a conference key $ck_{\mathcal{P}}$ for a conference group $\mathcal{P}$ that he is a member of. More formally,

DEFINITION 1. *A* (Γ, n)-non-interactive conference key distribution system ((Γ, n)-NICKDS) *for* $\mathcal{U} = \{U_1, \cdots, U_n\}$ *and conference structure* Γ *is a pair* $(\mathsf{KeyDist}, \mathsf{KeyDer})$ *of algorithms.*

- ***Key Distribution*** $\mathsf{KeyDist}^n_\Gamma(1^\kappa)$. *Given a security parameter* 1^κ*, a trusted authority* $\mathbb{T}$ *generates a public key* PK *and a set of secret key* $K_i, i = 1, \cdots, n$. PK *is made public and* K_i *is secretly given to* U_i.

- ***Key Derivation*** $\mathsf{KeyDer}(\mathcal{P}, K_i)$. *On input* $\mathcal{P} \in \Gamma$ *and a secret key* K_i *for* $U_i \in \mathcal{P}$*, compute* $ck_{\mathcal{P}} = \mathsf{KeyDer}(\mathcal{P}, K_i)$.

  Correctness. *For any* $U_i, U_j \in \mathcal{P}$*, it holds that* $\mathsf{KeyDer}(\mathcal{P}, K_i) = \mathsf{KeyDer}(\mathcal{P}, K_j)$.

The above definition generalizes the definition of NICKDS in [7] in which Γ is a threshold structure (a collection of all subsets of size at most t). In this paper we focus on three types of threshold structures defined below.

- $\Gamma_{\leq r} = \{\mathcal{P} : \mathcal{P} \subset \mathcal{U}, 2 \leq |\mathcal{P}| \leq r\}$
- $\Gamma_{\geq r} = \{\mathcal{P} : \mathcal{P} \subset \mathcal{U}, |\mathcal{P}| \geq r\}$
- $\Gamma_r = \{\mathcal{P} : \mathcal{P} \subset \mathcal{U}, |\mathcal{P}| = r\}$

The definition is applicable to public key and symmetric key setting, both. In public key setting, PK includes the system parameters and the public keys of all the users. In symmetric key setting, PK is only the system parameters.

Security of (Γ, n)-NICKDS in the information theoretic setting has been considered in corruption only model [7] where the adversary can corrupt a set of users $\mathcal{C}$. When a user U_i is corrupted, his secret key K_i will become available to the adversary. With $\{K_i \mid U_i \in \mathcal{C}\}$ at hand, the adversary chooses a challenge conference set $\mathcal{P} \in \Gamma$ and $\mathcal{P} \cap \mathcal{C} = \emptyset$. A (Γ, n)-NICKDS is considered secure against w corruptions, $|\mathcal{C}| \leq w$, if the adversary's view of the distribution of the conference key $ck_{\mathcal{P}}$ is uniform.

In Section 3, we will give a stronger definition of security for NICKDS by considering an adversary who can adaptively ask to see conference keys of his choice. We will use this model for unconditional and computational settings, both. In the following, we recall three constructions of NICKDSs that will be use din the rest of this paper.

2.1.1 A (Γ_r, n)-NICKDS with Perfect Security against w Corruptions

Blundo et al [7] proposed a (Γ_r, n)-NICKDS scheme that is secure against w corruptions. The construction is as follows. Let $f(x_1, \cdots, x_r) \in \mathbb{F}_q[x_1, \cdots, x_r]$ be a random symmetric multi-variable polynomial of degree w, where the degree is in terms of a single variable. For example, $x_1^2 x_2 + x_1 x_2^2$ is of degree 2. Each user $U_j, j = 1, \cdots, n$ has a public label z_j and receives a secret key $K_j = f(z_j, x_2, \cdots, x_r)$. The conference key for $\mathcal{P} = \{U_{i_1}, \cdots, U_{i_r}\}$ is defined as

$$ck_{\mathcal{P}} = f(z_{i_1}, \cdots, z_{i_r}) = K_{i_j}(z_{i_1}, \cdots, z_{i_{j-1}}, z_{i_{j+1}}, \cdots, z_{i_r}).$$

In this scheme the size of a user's key $|K_j|$ is independent of n, the number of users. We denote this scheme by $SymPoly(r, w)$.

2.1.2 A $(\Gamma_{\leq r}, n)$-NICKDS with Perfect Security against any Number of Corruptions

Desmedt and Viswanathan in [15] constructed a $(\Gamma_{\leq r}, n)$-NICKDS in which each conference set $\mathcal{P}$ ($|\mathcal{P}| \leq r$) is associated with an independent random key $ck_{\mathcal{P}}$. The secret key of a user U_i is $K_i = \{ck_{\mathcal{P}} \mid U_i \in \mathcal{P}, |\mathcal{P}| \leq r\}$. The conference key for $\mathcal{P}$ is naturally defined as $ck_{\mathcal{P}}$. It is easy to see that this scheme is secure against any number of corruptions. However, the system is inefficient and each user has to store $\sum_{\ell=1}^{r-1} \binom{n-1}{\ell}$ keys.

2.1.3 Fiat-Naor's Scheme

Fiat and Naor [19] proposed a non-interactive conference key distribution system $(\Gamma_{\geq 2}, n)$-NICKDS for a zero-message broadcast encryption. Their scheme allows any conference set of size two or more to share a common key.

The system works as follows. Let $\mathcal{U} = \{U_1, \cdots, U_n\}$ be the universe of all users and w denote an upper bound on the number of corrupted users. Each set of users, F, of

size at most w, is associated with a *basic key* $k_F \in \mathbb{F}_q$. The secret key for U_i is a subset of basic keys $K_i = \{k_F \mid U_i \notin F, F \subseteq \mathcal{U}, |F| \leq w\}$. For each conference set $\mathcal{P} \subseteq \mathcal{U}$, the conference key $ck_{\mathcal{P}}$ is defined as $\sum_{F \subseteq \bar{\mathcal{P}}, |F| \leq w} k_F$, where $\bar{\mathcal{P}} = \mathcal{U} \backslash \mathcal{P}$. Note that for a user $U_i \in \mathcal{P}$, for a subset $F \subseteq \bar{\mathcal{P}}$ we have $U_i \notin F$ and so $k_F \in K_i$. Hence, U_i can compute $ck_{\mathcal{P}}$. We denote this scheme by $FN(\geq 2, w)$.

It was shown [19] that Fiat-Naor scheme is secure against w corruptions. Indeed, let $\mathcal{P}$ be the test conference set and $\mathcal{C}$ be the corruption set such that $\mathcal{C} \subseteq \bar{\mathcal{P}}$ (otherwise, $ck_{\mathcal{P}}$ is immediately known to a corrupted user). Since $|\mathcal{C}| \leq w$, $k_{\mathcal{C}}$ is in the summation of $ck_{\mathcal{P}} = \sum_{F \subseteq \bar{\mathcal{P}}, |F| \leq w} k_F$. Since from the adversary's view point $k_{\mathcal{C}}$ is random, then it follows that $ck_{\mathcal{P}}$ is also random from his view point.

2.2 Security of Conference Key Distribution Protocols

NICKDS can be regarded as a non-interactive version of *group key exchange protocol* (e.g., [21]). Thus, it would be useful to review the security model of a group key exchange protocol. Our model later for NICKDS is essentially a restriction of this model to a non-interactive setting. In a group key exchange protocol, after a key initialization, a session key for a subset of users $\mathcal{P}$ can be established on demand through interactions. To avoid *replay attack*, the session keys for different sessions must be different even if the group users are the same. In a NICKDS, there is no interaction and so the session key $ck_{\mathcal{P}}$ for a group $\mathcal{P}$ is fixed.

The security of group key exchange protocol is modeled in [4, 10]. In this model, an adversary's power is formalized by access to oracles:

- `Send` oracle. With a call to this oracle, the adversary can send a message m on behalf of user U_i in a particular session to another user U_j. This formalizes man-in-the-middle attack in real world where the adversary can delete, modify, redirect and insert messages over the channel.

- `Corrupt` oracle. With a call to this oracle, the adversary is allowed to adaptively corrupt any party of its choice. As a result, all the information of the party including his long term secret, will become available to adversary.

- `Reveal` oracle. Under a call to this oracle, the key of a conference chosen by the adversary becomes available to him. This models the threat from the revealing of a session key.

- `Test` oracle. The adversary chooses a conference session with participants $\mathcal{P}^*$. He will be given a number that is either the conference key of the selected session, or a random string of the same length. The adversary can continue to issue `Corrupt` and `Reveal` queries subject to the condition that no party in $\mathcal{P}^*$ is corrupted and no `Reveal` query on the test session or its *partnered* sessions is issued. The adversary finally guesses if the provided number is random or the conference key. He is successful if his guess is correct.

2.3 Message Authentication Codes

A message authentication code (MAC) is a shared-key primitive for providing message authentication. The sender and receiver share a secret key $sk \leftarrow \mathcal{K}$, where $\mathcal{K}$ is the range of the secret key. A MAC is a keyed function $F_{sk} : \mathcal{M} \rightarrow \Xi$, that maps a message $m \in \mathcal{M}$ to a *tag*; the domain of possible messages and tags are $\mathcal{M}$ and Ξ respectively.

To authenticate a message m, the sender computes $tag = F_{sk}(m)$ and sends $tag|m$ to the receiver. The receiver accepts it as authentic if $tag = F_{sk}(m)$. The security of the MAC essentially states that an adversary can not forge a tag for a message m^* even if he has seen a number of (message, tag) pairs. Specifically, an adversary $\mathcal{A}$ can launch a chosen message attack by adaptively requesting the tag for messages of his choices. The adversary $\mathcal{A}$ succeeds if he can construct a pair (m^*, σ^*) such that $F_{sk}(m^*) = \sigma^*$ and that m^* was not queried before. Security of the MAC can be considered in the information theoretic or computational setting. In the former, the adversary's runtime is unbounded and in the latter case, it is bounded by a polynomial (in the security parameter).

DEFINITION 2. *Let $sk \leftarrow \mathcal{K}$. A message authentication code $F_{sk} : \mathcal{M} \rightarrow \Xi$ is said to be (t, T, ϵ)-*secure *under chosen message attack if no adversary $\mathcal{A}$ with runtime bounded by T, and issuing at most t queries to a tag oracle* Tag *described blow, can have a better chance than ϵ in constructing a pair (tag^*, m^*) such that $tag^* = F_{sk}(m^*)$ and that m^* was not issued as tag query.*

- $\mathsf{Tag}(m)$. *The tag oracle receives a query $m \in \mathcal{M}$ and returns an authentication tag $tag = F_{sk}(m)$.*

2.4 Computational Assumptions

Let p, q be large primes with $q \mid p + 1$. Let $\mathbb{G}$ be a group of order q over an elliptic curve and $\mathbb{G}_1$ be a prime group of order q in a finite field $\mathbb{F}^*_{p^2}$. Assume $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_1$ is a non-degenerate bilinear pairing.

DEFINITION 3. *Let $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_1$ be a bilinear pairing map and g be a generator of $\mathbb{G}$. The (ϵ, T)-*bilinear decisional Diffie-Hellman ((BDDH) *assumption requires that for any adversary $\mathcal{A}$ of runtime bounded T,*

$$|\Pr[\mathcal{A}(g^a, g^b, g^c, \hat{e}(g, g)^{abc}) = 1] - \Pr[\mathcal{A}(g^a, g^b, g^c, \gamma) = 1]| \quad (1)$$

*be at most ϵ; here a, b, c are chosen uniformly from $\mathbb{Z}_q$, γ is uniform in $\mathbb{G}_1$ and the probability is taken over random variables a, b, c, γ, coin flips of sampling $\mathbb{G}, \mathbb{G}_1, g$ and the internal coins of $\mathcal{A}$. We call $\mathcal{A}$ a (ϵ, T)-*adversary for BDDH.

In this definition, we usually say $(g^a, g^b, g^c, \hat{e}(g, g)^{abc})$ is a **bilinear Diffie-Hellman** (BDH) tuple while (g^a, g^b, g^c, γ) is a **random** tuple, where $a, b, c \leftarrow \mathbb{Z}_q, \gamma \leftarrow \mathbb{G}$.

3. A SECURITY MODEL FOR NICKDSS AGAINST ACTIVE ADVERSARIES

In the security model of group key exchange, the adversary is *active* and has access to `Send`, `Corrupt`, `Reveal` and `Test` oracles. In NICKDSs there is no interaction among users. This means that to model active adversaries for NICKDS `Send` oracles need not be considered. Hence, an active adversary in this setting has access to three types of oracles: `Corrupt`, `Reveal` and `Test` oracles. A formal definition follows.

DEFINITION 4. *Consider a* (Γ, n)-NICKDS *scheme. Let* $\mathcal{A}$ *be an adversary with adaptive access to the following oracles.*

- `Corrupt`(i). *Upon this query, the secret key* K_i *of* U_i *is provided to* $\mathcal{A}$.
- `Reveal`$(\mathcal{P})$. *Upon this query, the conference key* $ck_{\mathcal{P}}$ *for group* $\mathcal{P}$ *is provided to* $\mathcal{A}$.
- `Test`$(\mathcal{P}^*)$. *This query can be issued only once. Also it is assumed that* `Reveal`$(\mathcal{P}^*)$ *was not issued and that no user in* $\mathcal{P}^*$ *was corrupted. If these conditions are satisfied, choose* $rand \leftarrow \{0,1\}^{|ck_{\mathcal{P}^*}|}$ *and let* $b \leftarrow \{0,1\}$. *If* $b = 0$, $ck_{\mathcal{P}^*}$ *is provided to* $\mathcal{A}$; *otherwise* $rand$ *is provided to him. After this query,* $\mathcal{A}$ *can continue to issue* `Corrupt` *query and* `Reveal` *queries as long as no user in* $\mathcal{P}^*$ *is corrupted and* $\mathcal{P}^*$ *is not issued a* `Reveal` *query.*

At the end of the interaction, $\mathcal{A}$ *outputs a guess bit* b' *for* b. *The adversary is successful if* $b' = b$. *A* (Γ, n)-NICKDS *is said to be* (w, t, T, ϵ)-`secure` *if an adversary* $\mathcal{A}$ *with at most* w `Corrupt` *queries,* t `Reveal` *queries, and computation time bounded by* T *has a success probability (denoted by* $\Pr[\mathsf{Succ}(\mathcal{A})]$*) at most* $\frac{1}{2} + \epsilon$.

We use $\mathsf{Adv}(\mathcal{A}) := 2\Pr[\mathsf{Succ}(\mathcal{A})] - 1$ to quantify $\mathcal{A}$'s attack advantage in the system and use (w, t, T, ϵ)-secure NICKDS, $\Pr[\mathsf{Succ}(\mathcal{A})] \leq \frac{1}{2} + \epsilon$ to say $\mathsf{Adv}(\mathcal{A}) \leq \epsilon$.

Remark 1. Note the above definition is applicable to both unconditional and computational security framework. In the case of an unbounded adversary $T = \infty$, there is no limit on the computational power of the adversary. An (w, t, ∞, ϵ)-secure NICKDS ensures that the adversary's advantage is at most ϵ. If $\epsilon = 0$, the NICKDS scheme provides `perfect security`. In computationally secure case the adversary's success probability can approach 1 when the computational time T goes to ∞.

Remark 2. Note that if $t = 0$, (that is, the adversary does not have access to `Reveal` queries), the security model reduces to security against corruption only attack.

In Sections 4 and 5, we will evaluate the security of some NICKDSs against an active adversary in the unconditional model and computational model, respectively.

4. UNCONDITIONAL SECURE NICKDS AGAINST ACTIVE ADVERSARY

4.1 Security of $SymPoly(r, w)$ against Active Adversary

In the following, we show that $SymPoly(r, w)$ is secure against an adaptive adversary as long as $c + t \leq w$. That is, if there are c corrupted users and $c \leq w$, the NICKDS remains secure as long as the number of `Reveal` queries is at most $w - c$. Moreover, we show that this scheme will become completely insecure if the number of `Reveal` queries is increased by 1 (i.e., $w + 1 - c$ in total).

Note that according to this result the effect of a corrupted user on the security of the system is the same as the effect of a `Reveal` query.

THEOREM 1. *Consider a* $SymPoly(r, w)$ *scheme. Let* $c \leq w$. *Then the scheme is* $(c, w - c, \infty, 0)$*-secure but it is not* $(c, w + 1 - c, \infty, 0)$*-secure.*

Proof. In [7], it was shown $SymPoly(r, w)$ is $(w, 0, \infty, 0)$-secure. Now we show that if there exists an adversary $\mathcal{A}$ that violates the $(c, w - c, \infty, 0)$-security, then we can build an adversary $\mathcal{D}$ to violate $(w, 0, \infty, 0)$-security of it.

$\mathcal{D}$ acts as follows. On the one hand, he interacts with his own challenger in the $(w, 0, \infty, 0)$-security game in Definition 4. On the other hand, he simulates a $(c, w-c, \infty, 0)$-security game with $\mathcal{A}$. To do this, he uses the interaction with his challenger in the former game to help respond to the interaction with $\mathcal{A}$ in the latter game. Initially, when $\mathcal{D}$ receives the system parameter $(\mathbb{F}_q, r, n)$ from his challenger, run $\mathcal{A}$ with it as the system parameter in the simulated game. Then he interacts with $\mathcal{A}$ following the following rules.

$\mathcal{D}$ randomly selects a conference set $\mathcal{P}^*$ from all $\binom{n}{r}$ possible sets of size r and hopes that $\mathcal{A}$ will take $\mathcal{P}^*$ as his test set. After this, $\mathcal{D}$ answers the oracle queries of $\mathcal{A}$ as follows. Whenever $\mathcal{A}$ corrupts a user $U \in \mathcal{P}^*$ or queries the conference key for $\mathcal{P}^*$, $\mathcal{D}$ aborts with failure (this means that the guess of $\mathcal{P}^*$ was wrong since it is impossible for $\mathcal{A}$ to choose $\mathcal{P}^*$ as a test set); otherwise, whenever $\mathcal{A}$ corrupts a user U_j, $\mathcal{D}$ corrupts user U_j too and forwards the obtained K_j to $\mathcal{A}$; also whenever $\mathcal{A}$ makes a `Reveal` query for the key for $\mathcal{P} \neq \mathcal{P}^*$, $\mathcal{D}$ corrupts a user $U_\ell \in \mathcal{P} \backslash \mathcal{P}^*$, obtains his secret key K_ℓ, and uses it to compute and forward $ck_{\mathcal{P}}$ to $\mathcal{A}$; when $\mathcal{A}$ takes $\mathcal{P}^*$ as a test conference, $\mathcal{D}$ takes it as his own test conference. It then will receive the test number α (α is either $ck_{\mathcal{P}^*}$ or a random number), which it forwards to $\mathcal{A}$ as the response to the test query of $\mathcal{A}$.

Whenever $\mathcal{D}$ aborts, he outputs 0/1 randomly; otherwise, he outputs whatever $\mathcal{A}$ outputs. Note $\mathcal{P}^*$ is guessed correctly (denoted by event Good) with probability $1/\binom{n}{r}$. Note when $\neg\mathsf{Good}$ occurs, $\mathcal{D}$ outputs 0/1 randomly and thus succeeds in this case with probability $\frac{1}{2}$. Therefore, as a summary, if $\mathcal{A}$ succeeds in the real game with probability $\frac{1}{2} + \epsilon$, then $\mathcal{D}$ succeeds with probability $(\frac{1}{2} + \epsilon) \cdot \Pr[\mathsf{Good}] + \frac{1}{2} \cdot \Pr[\neg\mathsf{Good}] = \frac{1}{2} + \epsilon/\binom{n}{r}$. Since $\epsilon = 0$ [7], we conclude that $SymPoly$ is $(c, w - c, \infty, 0)$-secure.

Now we prove the second part: there exists an adversary $\mathcal{B}$ that uses c `Corrupt` queries and $w + 1 - c$ `Reveal` queries, to distinguish the test conference key. $\mathcal{B}$ does as follows.

$\mathcal{B}$ corrupts c users U_j, $j = 1, \cdots, c$ and obtains their corresponding K_j's. Now consider the following $w - c + 2$ conferences, $\mathcal{P}_i = \{U_{c+1}, \cdots, U_{c+r-1}\} \cup \{U_{c+r-1+i}\}$, $i = 1, \cdots, w+2-c$. $\mathcal{B}$ issues `Reveal` queries to the first $w-c+1$ conference sets and thus receives $ck_{\mathcal{P}_i}$ for $i = 1, \cdots, w+1-c$. He then takes $\mathcal{P}_{w+2-c}$ as the test conference. Next, $\mathcal{B}$ finds a symmetric polynomial $f^* \in \mathbb{F}_q[x_1, \cdots, x_r]$ of degree w such that $f^*(z_j, x_2, \cdots, x_r) = K_j$ $(1 \leq j \leq c)$ and that f^* is $ck_{\mathcal{P}_i}$ $(i = 1, \cdots, w + 1 - c)$ when evaluated at $\mathcal{P}_i$. This can be done by brute force search (recall that $\mathcal{B}$ has an infinite computational power) over all possible symmetric polynomial in $\mathbb{F}_q[x_1, \cdots, x_r]$ of degree w such that these constraints are satisfied.

Let f be the polynomial used by the trusted authority. Define $\Delta = f - f^*$. We have that Δ is zero when evaluated at $x_1 = z_j, j = 1, \cdots, c$ and is zero when evaluated at conference $\mathcal{P}_i, i = 1, \cdots, k - c + 1$. The first condition implies that $\Pi_{j=1}^{c}(x_1 - z_j) \mid \Delta$. By the symmetric property, $\Pi_{j=1}^{c}(x_\ell - z_j) \mid \Delta, \ell = 2, \cdots, r$. Thus, we have $\Pi_{\ell=1}^{r}\Pi_{j=1}^{c}(x_\ell - z_j) \mid \Delta$. Noting that Δ is of degree w, we have $\Delta = \Pi_{\ell=1}^{r}\Pi_{j=1}^{c}(x_\ell - z_j)Q$ where Q is a symmetric polynomial of degree $\leq w - c$ which is zero when evaluated at conferences $\mathcal{P}_i, i = 1, \cdots, w + 1 - c$. This is true because $\mathcal{P}_i$

does not contain any of the corrupted users U_j for $1 \le j \le c$. Let $Q = C_0 + C_1 x_r + \cdots + C_k x_r^k$, where $C_j \in \mathbb{F}_q[x_1, \cdots, x_{r-1}]$ of degree w. Let $\alpha_j = C_j(z_{c+1}, \cdots, z_{c+r-1})$. Since Q is zero when evaluated at $\mathcal{P}_i$ for $i = 1, \cdots, w+1-c$, we have $\alpha_0 + \alpha_1 z_{c+r-1+i} + \cdots + \alpha_{w-c} z_{c+r-1+i}^{w-c} = 0$. Or,

$$\begin{pmatrix} 1 & z_{c+r} & \cdots & z_{c+r}^{w-c} \\ 1 & z_{c+r+1} & \cdots & z_{c+r+1}^{w-c} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & z_{r+w} & \cdots & z_{r+w}^{w-c} \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_w \end{pmatrix} = 0$$

This is a Vandermond matrix and so is invertible. Thus, $(\alpha_0, \cdots, \alpha_w)$ is a zero vector and f^* and f are equal on the test conference $\mathcal{P}_{w+2-c}$. This means that the test key is determined. ■

It should be noted that our proof of $(c, w-c, \infty, 0)$-security does not depend on the structure of $SymPoly(r, w)$. Thus, generally we have if a NICKDS is $(w, 0, \infty, 0)$-secure then it is also $(c, w-c, \infty, 0)$-secure for any $c \le w$.

4.1.1 Security of Desmedt et al's Scheme against Active Adversaries

In this scheme, all conference keys are independent and so it is $(r, \infty, \infty, 0)$-secure. Note that $t = \infty$ means that there is no bound on the number of `Reveal` queries and the adversary may ask any number of such queries as long as conditions for the `Test` query are not violated.

4.2 Security of Fiat-Naor Scheme against Active Adversary

In the following, we consider security of Fiat-Naor scheme against an active adversary. We will show that the basic $FN(\ge 2, w)$ is insecure against $(w, 1, \infty, 0)$-adversary. We then show that a variant of $FN(\ge 2, w)$ by restricting the conference size to be $\ge n - w$, is $(w, \infty, \infty, 0)$-secure.

THEOREM 2. *Consider an $FN(\ge 2, w)$ on n users. If $w \le n-3$, then this scheme is not $(w, 1, \infty, 0)$-secure.*

Proof. We construct an adversary that uses w `Corrupt` queries and one `Reveal` query to determine the key of a `Test` conference of his choice. The attacker strategy is as follows: issue `Corrupt` queries on users $U_1, \cdots, U_w$ and issue `Reveal` query on the conference set $\mathcal{U} \backslash \{U_1, \cdots, U_w\}$, where $\mathcal{U} = \{U_1, \cdots, U_n\}$. It issues a `Test` query on a conference set $\mathcal{P}^* = \mathcal{U} \backslash \{U_1, \cdots, U_{w+1}\}$. Here $\mathcal{P}^*$ is an allowable conference set since $|\mathcal{P}^*| \ge 2$ for $w \le n-3$. Note $ck_{\mathcal{P}^*} = \sum_{F \subseteq \{U_1, \cdots, U_{w+1}\}} k_F$. Note if F does not include $\{U_1, \cdots, U_w\}$, then k_F is known to the attacker. On the other hand, if F in the formula for computing $ck_{\mathcal{P}^*}$ has a size no more than w, then k_F is known to the attacker unless $F = \{U_1, \cdots, U_w\}$. Since $k_{\{U_1, \cdots, U_w\}}$ can be obtained through key query $ck_{\mathcal{U} \backslash \{U_1, \cdots, U_w\}} = \sum_{F \subseteq \{U_1, \cdots, U_w\}} k_F$, it follows that $ck_{\mathcal{P}^*}$ is known to the attacker. Indeed, if $U_i \notin F$ $(i = 1, \cdots, w)$, then k_F is known to the attacker since k_F is known to U_i. Thus, from $ck_{\mathcal{U} \backslash \{U_1, \cdots, U_w\}}$, the attacker can derive $k_{\{U_1, \cdots, U_w\}}$. This completes the proof. ■

In the following, we shows if the conference size is lower bounded, then the scheme is secure against active adversary.

THEOREM 3. *Let $FN(\ge n-w, w)$ be Fiat-Naor scheme with the conference set structure being specified by $\Gamma_{\ge n-w}$. Then $FN(\ge n-w, w)$ is $(w, \infty, \infty, 0)$-secure. That is, it is unconditionally secure against w `Corrupt` queries and any number of `Reveal` queries.*

We note that the number of possible `Reveal` queries in $FN(\ge n-w, w)$ is bounded and so $t = \infty$ effectively means all possible `Reveal` queries.

More explicitly, for $c \le w$ `Corrupt` queries, there are $\sum_{\nu=n-w}^{n-c} \binom{n-c}{\nu}$ conference sets that do not contain a corrupted user (in each conference all users are honest). Excluding the conference set used in the `Test` query, the adversary can query at most $-1 + \sum_{\nu=n-w}^{n-c} \binom{n-c}{\nu}$ conference keys. Thus, the first ∞ should be read as all the remaining conferences excluding the test conference being available for `Reveal` queries.

Proof. W.L.O.G, assume $U_1, \cdots, U_c$ are corrupted. Also assume if a `Reveal` query is issued for a conference set $\mathcal{P}$, then $\mathcal{P}$ does not contain a corrupted user. Under this convention, $\{U_1, \cdots, U_c\} \subseteq \bar{\mathcal{P}}$. Assume $\mathcal{P}_1, \cdots, \mathcal{P}_t$ are the t queried conferences and $\mathcal{P}_{t+1}$ is chosen as the test conference. Let $F_1, \cdots, F_m$ be all the subsets of size at most w that are contained in some $\bar{\mathcal{P}}_i$ (for $i \in \{1, \cdots, t+1\}$) but contain $\{U_1, \cdots, U_c\}$ (i.e., for any j, there exists i such that $\{U_1, \cdots, U_c\} \subseteq F_j \subset \bar{\mathcal{P}}_i$). Define $a_{ij} = 1$ if $F_j \subseteq \bar{\mathcal{P}}_i$, and $a_{ij} = 0$ otherwise. Define $\mathcal{C} = \{U_1, \cdots, U_c\}$. Then,

$$\sum_{j=1}^{m} a_{ij} k_{F_j} = ck_{\mathcal{P}_i} - \sum_{\mathcal{C} \subsetneq F \subseteq \bar{\mathcal{P}}_i} k_F, \quad i = 1, \cdots, t+1. \qquad (2)$$

In other words,

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{(t+1)1} & a_{(t+1)2} & \cdots & a_{(t+1)m} \end{pmatrix} \begin{pmatrix} k_{F_1} \\ k_{F_2} \\ \vdots \\ k_{F_m} \end{pmatrix}$$
$$= \begin{pmatrix} ck_{\mathcal{P}_1} - \sum_{\mathcal{C} \subsetneq F \subseteq \bar{\mathcal{P}}_1} k_F \\ ck_{\mathcal{P}_2} - \sum_{\mathcal{C} \subsetneq F \subseteq \bar{\mathcal{P}}_2} k_F \\ \vdots \\ ck_{\mathcal{P}_{t+1}} - \sum_{\mathcal{C} \subsetneq F \subseteq \bar{\mathcal{P}}_{t+1}} k_F \end{pmatrix}$$

Note that on the right hand side, all terms except $ck_{\mathcal{P}_{t+1}}$, that is all k_F and $ck_{\mathcal{P}_i}$, are known to the attacker. Thus, to show $ck_{\mathcal{P}_{t+1}}$ is independent of the adversary's view, it suffices to show that the matrix (a_{ij}) has full rank equal to $t+1$. We show that the columns and rows of (a_{ij}) can be permuted such that it becomes an upper triangle matrix, and that the element of the permutated matrix in position (i, i) is 1 for all $i \in \{1, \cdots, t+1\}$. Note since $|\bar{\mathcal{P}}_\ell| \le w$, $\bar{\mathcal{P}}_\ell$ must be equal to some F_j. Thus, $m \ge t+1$, which will imply (a_{ij}) has full rank equal to $t+1$. We use induction on t. For $t = 0$, the result is immediate since $a_{1j} = 1$ if $\bar{\mathcal{P}}_1 = F_j$. Now assume the result holds for $t-1$. Consider the case t. There must exist a j such that $\bar{\mathcal{P}}_j$ is not contained in $\bar{\mathcal{P}}_i$ for any $i \in \{1, \cdots, t+1\} \backslash \{j\}$. Since there exists ℓ such that $\bar{\mathcal{P}}_j = F_\ell$, we have that the ℓth column is zero except $a_{j\ell} = 1$ (i.e., $a_{i\ell} = 0$ for $i \ne j$). Exchanging the ℓth column with the $1st$ column, jth row with the first row, we can move $a_{j\ell}$ to the entry $(1, 1)$. For simplicity, assume $j = 1$ and $\ell = 1$. Thus, (a_{ij}) is the following form:

$$\begin{pmatrix} 1 & a_{12} & \cdots & a_{1m} \\ 0 & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{(t+1)2} & \cdots & a_{(t+1)m} \end{pmatrix}. \text{ Thus,}$$

$$\begin{pmatrix} a_{22} & \cdots & a_{2m} \\ \vdots & \ddots & \vdots \\ a_{(t+1)2} & \cdots & a_{(t+1)m} \end{pmatrix} \begin{pmatrix} k_{F_2} \\ \vdots \\ k_{F_m} \end{pmatrix}$$

$$= \begin{pmatrix} ck_{\mathcal{P}_2} - \sum_{\mathcal{C} \not\subseteq F \subseteq \bar{\mathcal{P}}_2} k_F \\ \vdots \\ ck_{\mathcal{P}_{t+1}} - \sum_{\mathcal{C} \not\subseteq F \subseteq \bar{\mathcal{P}}_{t+1}} k_F \end{pmatrix}$$

By induction, the matrix $(a_{ij})_{2 \le i \le t+1, 2 \le j \le m}$ can be transformed into an upper triangle where every (i,i) entry is 1 and so the claim holds for the case t. Therefore, (a_{ij}) has a rank $t+1$. This completes our proof. ■

5. COMPUTATIONALLY SECURE NICKDS AGAINST ACTIVE ATTACKER

In this section we consider the security of NICKDSs in computational setting. That is, assume the adversary's computation time T is bounded.

5.1 Security of Non-Interactive DH Protocol

Numerous Diffie-Hellman based key exchange (*interactive*) protocols have been proposed; for examples see [11, 12] . We are interested in the security of the non-interactive case. See blow.

- DH-NICKDS Let p, q be two large primes and $p = 2q+1$. Let G be a prime group of order q in $\mathbb{Z}_p^*$ and assume g is a generator of G. For a user U_i, let $x_i \leftarrow \mathbb{Z}_q$ be the secret key and compute the public key $PK_i = g^{x_i}$. The system public key is $PK = \{g, PK_1, \cdots, PK_n\}$. For a two-party conference $\mathcal{P} = \{U_i, U_j\}$, the conference key $ck_{\mathcal{P}}$ is defined as $g^{x_i x_j} = PK_j^{x_j} = PK_i^{x_j}$.

Note that in the interactive Diffie-Hellman key exchange, the messages g^x and g^y are independently constructed for each session. In the non-interactive version, x_i will be reused in all conferences $\mathcal{P}$ that contain user U_i and so the security of the former does not imply the security of latter. The following theorem gives a rigorous statement of security of DH-protocol in non-interactive case. The proof is given in the full version of the paper.

THEOREM 4. *The* DH-NICKDS *is* $(n, \infty, T', n(n-1)\epsilon/2)$*-secure, if there is no* (ϵ, T)*-adversary for decisional Diffie-Hellman problem in* G*. Here* $T' = T - n(n+1)t_e/2 - t_i$*,* t_e *is the time required for exponentiation in* $\mathbb{Z}_p^*$*, and* t_s *is the time for sampling two users from* $\mathcal{U}$*.*

Note in the above, by ∞, we mean the adversary can make `Reveal` query to any conference set except the test conference. There are totally $n(n-1)/2-1$ such conferences.

5.2 Secure NICKDS from Bilinear Decision Diffie-Hell man

In this section, we consider the security of a computationally secure NICKDS that uses bilinear pairing and is a variant of the tripartite key agreement of Joux [20].

Let p, q be large primes with $q \mid p+1$. Let $\mathbb{G}$ be a prime group of order q over an elliptic curve and $\mathbb{G}_1$ be a prime group of order q in a finite field $\mathbb{F}_{p^2}^*$. Assume $\hat{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$ is a non-degenerate bilinear pairing.

- BDDH-NICKDS Let $\mathcal{U} = \{U_1, \cdots, U_n\}$ be a group of n users, and $\kappa = |p|$ denote the system's security parameter. Let $g, h \leftarrow \mathbb{G}\backslash\{1\}$ denote two random elements of $\mathbb{G}\backslash\{1\}$. For a user U_i, let $a_i \leftarrow \mathbb{Z}_q$ and compute $A_i = g^{a_i}$. Define the system's public key as $PK = \{A_i\} \cup \{g, h\}$. For a group $\mathcal{P} = \{U_i, U_j, U_z\}$, the conference key $ck_{\mathcal{P}}$ is defined as $ck_{\mathcal{P}} = \hat{e}(g,g)^{a_i a_j a_z}$; also for a group $\mathcal{P}$ of size 2, $\mathcal{P} = \{U_i, U_j\}$ we have $ck_{\mathcal{P}} = \hat{e}(g,h)^{a_i a_j}$.

The security of BDDH-NICKDS is based on bilinear decisional Diffie-Hellman assumption and is stated as follows. The proof will appear in the full version of the paper.

THEOREM 5. *If there does not exist an* (ϵ, T)*-adversary for* BDDH*, then the* BDDH-NICKDS *is* $(n, \infty, T', \epsilon')$*-secure for* $T' \le T - n^3 t_e - t_s$ *and* $\epsilon' = \frac{(n+1)n(n-1)\epsilon}{6}$*, where* t_e *is the time required for exponentiation over* $\mathbb{Z}_p$*, and* t_s *is the time required for sampling a subset of size two or three in* $\mathcal{U}$*.*

Again here ∞ means that, all conference keys except the key for the test conference, can be queried as long as the adversary's running time is bounded by T'.

5.3 Computationally Secure NICKDSs from Unconditionally Secure Ones

In Section 4, we showed that $SymPoly(r, w)$ allows at most w conference key queries. To provide security against more queries we may trade unconditional security and be content with computational security. A natural question is if there exist a generic approach to convert a $(w, 0, \infty, 0)$-secure scheme to a computationally secure NICKDS such that an arbitrary number of `Reveal` queries are allowed. In this section, we answer this question in the affirmative under the random oracle assumption and show that for an unconditionally secure NICKDS $Conf$, a computationally secure variant denoted by $Conf^H$ can be defined, that differs from $Conf$ only in the key derivation function. More specifically, if in $Conf$ the conference key for $\mathcal{P}$ is $ck_{\mathcal{P}} \in \mathcal{K}$, then in $Conf^H$ the conference key for $\mathcal{P}$ is $hck_{\mathcal{P}} = H(ck_{\mathcal{P}})$ where $\mathcal{K}$ is the domain of the conference key in $Conf$ and $H : \mathcal{K} \to \{0,1\}^\kappa$ is a cryptographic hash function.

The following theorem states that if $Conf$ is information theoretically secure in the corruption-only model, then $Conf^H$ is computationally secure in the adaptive model. The proof idea is that for any Reveal$(\mathcal{P})$ query, $hck_{\mathcal{P}}$ is essentially independent of $ck_{\mathcal{P}}$. Thus, the useful information to distinguish the test key is only the secret keys of corrupted users. However because of the information theoretic security of $Conf$, corrupted keys are independent of the test key and so the adversary in the test session in $Conf^H$ has negligible advantage. The complete proof will appears in the full version of the paper.

THEOREM 6. *If* $H : \mathcal{K} \to \{0,1\}^\kappa$ *is a random oracle and* $Conf$ *is a* $(w, 0, \infty, 0)$*-secure NICKDS, then* $Conf^H$ *is* $(w, \infty, T, 3T^2/q)$*-secure NICKDS, where* $|\mathcal{K}| = q$*.*

From Theorems 1, 2, and 6, we immediately have the following corollaries.

COROLLARY 1. *If $H : \mathbb{F}_q \to \{0,1\}^\kappa$ is a random oracle, then $SymPoly^H(r, w)$ is $(k, \infty, T, 3T^2/q)$-secure.*

COROLLARY 2. *If $H : \mathbb{F}_q \to \{0,1\}^\kappa$ is a random oracle, then $FN^H(\geq 2, w)$ is $(k, \infty, T, 3T^2/q)$-secure.*

6. A GENERIC RING AUTHENTICATION FROM NICKDS

In a *ring authentication system* a sender sends a message to a receiver such that the receiver is able to verify authenticity of the message as originated from one of the members of a *ring* chosen by the sender. This allows the sender to 'hide' himself in an *anonymity set* (i.e. the ring) of his choice and his anonymity is information theoretic.

A *ring signature* [30] is a public-key based ring authentication system that in its basic form requires the users to own certified public keys of a signature scheme. We consider symmetric-key ring authentication systems where the authenticated message is only verifiable by users with appropriate secret key. The system requires a *Trusted Authority TA* who sets up the system public parameters and securely distributes secret keys of the users. To authenticate a message m, a user uses the tagging function Tg to generate a tag σ and forms the authenticated messages $(\mathcal{P}, m, \sigma)$, where $\mathcal{P}$ is the anonymity group chosen by the user. A user with appropriate secret key is able to verify authenticity of the message but cannot obtain any information about the identity of the sender other than that the sender is a member of $\mathcal{P}$. Thus, a sender's anonymity is information theoretic. *In the rest of this paper, a ring authentication system refers to symmetric key ring authentication.*

Let $\mathcal{U} = \{U_1, \cdots, U_n\}$ denote the set of users. A formal definition is as follows.

DEFINITION 5. *A $(\leq r, n)$-ring authentication scheme is a triple of PPT algorithms $(\mathcal{G}, \mathsf{Tg}, \mathsf{Ver})$ as described below.*

- ***Initialization*** $\mathcal{G}_r^n(1^\kappa)$. *This algorithm is performed by a* Trusted Authority TA. *TA takes 1^κ as input and outputs the system's public information PK and a set of secret keys K_i, $(i = 1, \cdots, n)$. PK is made publicly accessible but K_i is only provided to user U_i.*
- ***Tag Computation*** $\mathsf{Tg}(m, \mathcal{P}, K_i)$. *A user U_i who wants to send a message m to a receiver U_j does the following: (i) he chooses an anonymity set $\mathcal{P}$, where $U_i, U_j \in \mathcal{P}$, (ii) he uses $\mathsf{Tg}(m, \mathcal{P}, K_i)$ to find a tag $\sigma \in \{0,1\}^\kappa$. The message sent to the receiver is $[m, \mathcal{P}, \sigma]$.*
- ***Verification*** $\mathsf{Ver}([m, \mathcal{P}, \sigma], K_j)$. *A receiver U_ℓ who receives a message $[m, \mathcal{P}, \sigma]$, verifies that $U_\ell \in \mathcal{P}$, and if true uses $\mathsf{Ver}([m, \mathcal{P}, \sigma], K_j)$ to produce a binary output, 0 (for reject) or 1 (for accept).*

  Correctness: *If $\sigma = \mathsf{Tg}(m, \mathcal{P}, K_i)$, then for any $P_j \in \mathcal{P}$, $\mathsf{Ver}([m, \mathcal{P}, \sigma], K_j) = 1$.*

In the above definition, we assume $\mathcal{P}$ has its size restricted to $|\mathcal{P}| \leq r$. However, our definition can straightforwardly be extended to more general cases including $|\mathcal{P}| \geq r$, or $\ell_1 \leq |\mathcal{P}| \leq \ell_2$, or $|\mathcal{P}| = r$.

In the following, we define security of a ring authentication system. We allow an adversary to adaptively corrupt up to w users. When a user U_i is corrupted, his secret K_i is provided to the adversary. In addition, the adversary is allowed to adaptively query the tagging oracle for messages and anonymity sets of his choice. That is, the adversary can choose a message m, a subset of users $\mathcal{P}$ and a user $U_i \in \mathcal{P}$, and ask to compute $\sigma = \mathsf{Tg}(m, \mathcal{P}, K_i)$ for the message sent from U_i.

The ring authentication system is required to satisfy two security properties: unforgeability and anonymity. Informally, in a ring authentication system the adversary should not be able to construct a valid tuple $(m, \mathcal{P}, \sigma, U_i)$ assuming $U_i \in \mathcal{P}$ and $(m, \mathcal{P})$ have not been queried to the tagging oracle, and no member of $\mathcal{P}$ has been the subject of a query to `Corrupt` oracle (i.e. corrupted). A ring authentication is said to provide sender anonymity if given an authenticated message $(m, \mathcal{P}, \sigma)$, the adversary cannot figure out which member of $\mathcal{P}$ has constructed the message (i.e. generated σ). A formal definition follows.

DEFINITION 6. *Let $(\mathcal{G}, \mathsf{Tg}, \mathsf{Ver})$ be a $(\leq r, n)$-ring authentication scheme and $\mathcal{A}$ be an adversary with adaptive access to the following oracles.*

- `Corrupt`(i). *$\mathcal{A}$ can issue a corruption query. If a user U_i is corrupted, his secret K_i will be available to $\mathcal{A}$.*
- `TAG`$(m, \mathcal{P}, j)$. *$\mathcal{A}$ can issue a tag query $(m, \mathcal{P}, j)$. If the request satisfies the requirements, $|P| \leq r$ and $U_j \in \mathcal{P}$, then a tag $\sigma = \mathsf{Tg}(m, \mathcal{P}, K_j)$ is computed and provided to $\mathcal{A}$.*
- `F-Test`$(m, \mathcal{P}, \sigma, j)$. [Unforgeability Test] *In this test the adversary constructs a tuple $(m, \mathcal{P}, \sigma)$ for $|\mathcal{P}| \leq r$ s.t. no user in $\mathcal{P}$ is uncorrupted and that $(m, \mathcal{P}, *)$ has not been queried to tagging oracle. $\mathcal{A}$ is successful if the verification $\mathsf{Ver}([m, \mathcal{P}, \sigma], K_j) = 1$ holds. This test can be called at most once.*
- `A-Test`$(m, \mathcal{P}, j_1, j_0)$. [Anonymity Test] *In this test, $\mathcal{A}$ chooses a message m, an anonymity set $\mathcal{P}$ and a pair of users U_{j_1}, U_{j_0} where $U_{j_1}, U_{j_0} \in \mathcal{P}$ and will receive $\sigma = Tg(m, \mathcal{P}, K_{j_b})$ for $b \leftarrow \{0,1\}$. The adversary $\mathcal{A}$ can continue to issue* `Corrupt` *and* `TAG` *queries as in previous steps and eventually outputs a guess bit b' for b. The adversary* succeeds *if $b' = b$.*

*A ring authentication system $(\mathcal{G}, \mathsf{Tg}, \mathsf{Ver})$ is (w, t, T, ϵ)-*secure *if the success probability of any adversary $\mathcal{A}$ with runtime bounded by T, and access to at most w* `Corrupt` *queries and t* `TAG` *queries, is at most $\epsilon + \frac{1}{2}$.*

We remark that in the anonymity test, $\mathcal{P}$ may include corrupted users. This essentially requires that an insider can not break the anonymity. The adversary runtime T need not be finite. When $T = +\infty$ the ring authentication scheme is *unconditionally secure*.

Ring authentication in the information theoretic setting and against passive adversaries, has been considered in [31]. Authors proposed a composition construction that uses a non-interactive conference key distribution and an authentication code, both with information theoretic security, and results in a ring authentication that is unconditionally secure in corruption-only model.

In this section, we show a generic construction for a ring authentication system using an (w, t, T_1, ϵ_1)- secure NICKDS

as defined in Section 3, and (ϵ_2, t, T_2)-secure MAC to obtain a (w, t, T, ϵ)-ring authentication. This generalizes [31] in two ways: firstly by considering security against a more powerful adversary, and secondly by providing a construction that is secure in computational setting. Our proof relies on the new security notion of NICKDS introduced in Section 3.

Now we introduce the syntax of our construction.

A Generic Ring Authentication Scheme

Let $F : \mathcal{K} \times \mathcal{M} \to \Xi$ be a message authentication code and $(\mathsf{KeyDist}, \mathsf{KeyDer})$ be a $(\Gamma_{\leq r}, n)$-NICKDS, where the range of KeyDer is $\mathcal{K}$. We construct a $(\leq r, n)$-ring authentication scheme $(\mathcal{G}, \mathsf{Tg}, \mathsf{Ver})$ as follows.

- ***Initialization*** $\mathcal{G}_r^n(1^\kappa)$. Let $(PK, K_1, \cdots, K_n) \leftarrow \mathsf{KeyDist}(r, n, \kappa)$. The public key information is PK. The secret key for U_i is K_i.
- ***Tag Computation*** $\mathsf{Tg}(m, \mathcal{P}, K_i)$. Compute $ck_{\mathcal{P}} = \mathsf{KeyDer}(\mathcal{P}, K_i)$ and $\sigma = F_{ck_{\mathcal{P}}}(m \| \mathcal{P})$. Output σ.
- ***Verification*** $\mathsf{Ver}(m, \mathcal{P}, K_j, \sigma)$. Compute $ck_{\mathcal{P}} = \mathsf{KeyDer}(\mathcal{P}, K_j)$ and check whether $\sigma \stackrel{?}{=} F_{ck_{\mathcal{P}}}(m \| \mathcal{P})$. If no, output 0 (for reject); 1 (for accept) otherwise.

This construction is similar to the one proposed in [31]. The security of this construction is described in the following theorem. The proof will appear in the full version of the paper.

THEOREM 7. *Let* $(\mathsf{KeyDist}, \mathsf{KeyDer})$ *be a* (w, t, T_1, ϵ_1)*-secure* $(\Gamma_{\leq r}, n)$-NICKDS *scheme and* F *be an* (ϵ_2, t, T_2)*-secure* MAC. *Then the generic ring authentication scheme is* (w, t, T, ϵ)*-secure, where* $\epsilon \leq (4\epsilon_1 + \epsilon_2) \sum_{t=2}^{r} \binom{n}{t}$, $T \leq \min\{T_1 - (t+1)t_F - t_s, T_2 - (t+1)t_F - t_i - t_s\}$, t_F *is the time to evaluate* F, t_s *is the time to sample a random conference set in the NICKDS scheme and* t_i *is the time to initialize the ring authentication scheme.*

REMARK 1. *Our composition theorem is applicable to both computationally and unconditionally secure ring authentication systems. Specifically, if we require* T_1 *and* T_2 *to be* ∞*, then the composition theorem states that the resulting ring authentication is secure against an unlimited adversary with* $T = \infty$.

REMARK 2. *The theorem only considers* $(\leq r, n)$*-ring authentication. However, it is easy to extended it to other conference sizes (e.g.,* $|\mathcal{P}| \geq r$*, or* $|\mathcal{P}| = r$*, or* $\ell_1 \leq |\mathcal{P}| \leq \ell_2$*).*

7. CCA2 SECURITY FOR BROADCAST ENCRYPTION

A *broadcast encryption system* (BES) is an encryption mechanism that allows a *Broadcast Center* (BC) to efficiently distribute content to subgroups of a user group $\mathcal{U}$ such that only the members of a target group $\mathcal{P}$ can decrypt the ciphertext. Broadcast encryption was first introduced by Fiat and Naor [19]. A näive solution to securely distribute the content to a target group is to let BC to have a shared key with each user, requiring the keys to be independently chosen from a uniform distribution. To securely broadcast a message m to users in $\mathcal{P}$, the center encrypts the message m using all the keys that it shares with users in $\mathcal{P}$. This is an expensive solution as the length of the broadcast ciphertext is proportional to the size of the group. In the literature, numerous approaches were proposed to obtain an efficient BES with provable security. Naor et al. [28] proposed a subset cover framework and proved security of the scheme. In their security model the adversary can adaptively corrupt a set of users and has adaptive access to encryption and decryption oracles before seeing the challenge ciphertext, hence guaranteeing CCA1 security for the encryption system. This scheme is in the symmetric key setting and the keys used by the BC and receivers are secret.

Dodis and Fazio [17] adapted the framework of [28] to the public key setting and allowed the adversary to have access to the decryption oracle after receiving the challenge ciphertext, hence guaranteeing CCA2 security.

In this section we consider a new security model for BES in the symmetric key setting that provides CCA2 security but the number of corrupted users is limited to w. We believe that this model realistically captures the adversary's power in the real world: the adversary can corrupt a bounded number of users but he can obtain the decryption of ciphertexts of his choice. We then consider a generic composition construction for a BES that is secure in our model, using a secure NICKDS and a CCA2 secure encryption system.

In a *zero message broadcast encryption* [19], the broadcast center uses a non-interactive conference key distribution system(NICKDS) to calculate a common key for the target group (i.e. authorized users) and uses that key to encrypt the message. Our contribution here is to provide a formal model and prove the security of this generic composition if NICKDS and the basic encryption scheme are properly chosen (See Theorem 8 for a formal statement).

We first formally introduce a NICKDS based broadcast encryption systems. Let $(\mathsf{KeyDist}, \mathsf{KeyDer})$ be a $(\Gamma_{\geq r}, n)$-NICKDS. [1] Let $E : \mathcal{K} \times \mathcal{M} \to \Sigma$ be a symmetric encryption scheme and $D : \mathcal{K} \times \Sigma \to \mathcal{M} \cup \{\perp\}$ be the corresponding decryption scheme, where $\mathcal{K}$, $\mathcal{M}$ and Σ are key space, message space and ciphertext space, respectively. A NICKDS based broadcast encryption system has the following stages.

1. In the Key Assignment stage a broadcast center runs a polynomial time algorithm $\mathsf{KeyDist}_r^n(1^\kappa)$ algorithm to get $(PK, K_1, \cdots, K_n) \leftarrow \mathsf{KeyDist}_r^n(1^\kappa)$, where PK is the public information available to all users and K_i is the secret key for user $U_i, i = 1, \cdots, U_n$.

2. In Broadcast and Decryption stage the broadcaster center, who wants to broadcasts a message m to a group of users $\mathcal{P}$, first computes the conference key $ck_{\mathcal{P}}$ of $\mathcal{P}$ and then constructs the ciphertext $C = E_{ck_{\mathcal{P}}}(m)$ and broadcasts $(\mathcal{P}, C)$. A user $U_i \in \mathcal{P}$ who receives C, first computes $ck_{\mathcal{P}} = \mathsf{KeyDer}(\mathcal{P}, K_i)$ and decrypts $m = D_{ck_{\mathcal{P}}}(C)$.

In this scheme, it is always assumed that $ck_{\mathcal{P}}$ is well defined (i.e., $\mathcal{P}$ is in the conference access structure $\Gamma_{\geq r}$ of the NICKDS scheme). The proposed security model for a NICKDS based broadcast encryption is similar to Naor et al. [28]: (i) an adversary adaptively corrupts users and eavesdrop ciphertexts in the channel; (ii) the adversary is

[1] Note that the model can be developed as $(\Gamma_{\leq r}, n)$-NICKDS or (Γ_r, n)-NICKDS. We choose to consider $(\Gamma_{\geq r}, n)$-NIKCDS since in many broadcast encryptions the number of unauthorized users are relatively small and so the authorized user set $\mathcal{P}$ has a large size (e.g., close to n).

given a ciphertext indistinguishability test, with the addition of allowing the adversary to access encryption and decryption oracles after being given the challenge ciphertext. The broadcast encryption system is secure if the adversary's success probability is negligible. Formally,

DEFINITION 7. *Let a broadcast encryption system be specified by a* $(\Gamma_{\geq r}, n)$-NICKDS $(\mathsf{KeyDist}, \mathsf{KeyDer})$ *and a symmetric cryptosystem* (E, D). *Assume* $(PK, K_1, \cdots, K_n) \leftarrow \mathsf{KeyDist}_r^n(1^\kappa)$. *$PK$ is public and K_i is the secret key for user $U_i, i = 1, \cdots, n$. Let $\mathcal{A}$ be an adversary with an adaptive access to the following oracles.*

- `Corrupt`(i). *$\mathcal{A}$ can corrupt any user of his choice. If a user U_i is corrupted, his secret key K_i will be available to $\mathcal{A}$.*
- `Encrypt`$(m, \mathcal{P})$. *$\mathcal{A}$ can issue an encryption query with a message/receivers pair $(m, \mathcal{P})$. Here it is required that $\mathcal{P}$ be a feasible conference set in NICKDS. Upon this query, compute a conference key $ck_{\mathcal{P}}$ and the ciphertext $C = E_{ck_{\mathcal{P}}}(m)$. Then C is feedback to $\mathcal{A}$.*
- `Decrypt`$(\mathcal{P}, C)$. *$\mathcal{A}$ can issue a decryption query with a receivers/ciphertext pair $(\mathcal{P}, C)$. Again we assume $\mathcal{P}$ must be a feasible conference set in NICKDS. Upon this query, compute $ck_{\mathcal{P}}$ and decrypt $m' = D_{ck_{\mathcal{P}}}(C) \in \mathcal{M} \cup \{\perp\}$, where $\perp$ implies that C is an invalid ciphertext. Finally, m' is returned to $\mathcal{A}$.*
- `Test`$(m_0, m_1, \mathcal{P}^*)$. *$\mathcal{A}$ issues a test query with a message pair (m_0, m_1) and a receiver set $\mathcal{P}$. In this case a bit b is chosen, $b \leftarrow \{0, 1\}$, the conference key $ck_{\mathcal{P}^*}$ is computed and the ciphertext $C^* = E_{ck_{\mathcal{P}^*}}(m_b)$ will be produced and returned to $\mathcal{A}$. The adversary must compute a guess bit b' for b. This query is allowed once. Intuitively, the system is secure if probability of $b' = b$ is $\frac{1}{2}$.*

After the `Test` *query, $\mathcal{A}$ can continue with the other three types of queries, assuming that (1) no user $U \in \mathcal{P}^*$ is corrupted, and (2) $(\mathcal{P}^*, C^*)$ is not issued to the decryption oracle. Finally, $\mathcal{A}$ outputs a guess bit b' for b. He is said to be successful if $b' = b$.*

The broadcast encryption scheme is (w, t, T, ϵ)-secure if for an adversary $\mathcal{A}$ with running time bounded by T, who issues at most w `Corrupt` *queries and t* `Encrypt/Decrypt` *queries, it holds that* $\Pr[b' = b] \leq \frac{1}{2} + \epsilon$.

In our model the adversary can access `Decryption` oracle after the `Test` query and so a secure system in this model has CCA2 like security.

7.0.1 Security of Subset Cover Scheme of Naor et al

The subset cover scheme scheme can be outlined as follows. Let $\mathcal{U}$ be the universe of users. Consider a collection of subsets of $\mathcal{U}$: $A_1, \cdots, A_\mu$. For each A_j, associate a key k_j to it. The secret key K_i of a user's U_i defined $K_i = \{k_j \mid U_j \in A_j, j = 1, \cdots, \mu\}$[2]. To send a message m to a user set $\mathcal{P}$, BC first finds $A_{i_1}, \cdots, A_{i_v}$ such that $\cup A_{i_j} = \mathcal{P}$. Then he defines the broadcast ciphertext as $C = \langle E_{k_{i_1}}(sk), \cdots, E_{k_{i_v}}(sk), F_{sk}(m)\rangle$, where E, F are encryption schemes and sk is a random session key. A user $U_z \in A_{i_j}$ can decrypt C by first decrypting $E_{k_{i_j}}(sk)$ to obtain sk and thus m.

A CCA2 attacker can break security of this scheme as follows. He first requests a challenge test $(m_0, m_1, A_1 \cup A_2)$. After receiving $C^* = \langle E_{k_1}(sk), E_{k_2}(sk), F_{sk}(m_b)\rangle$, the adversary asks for decryption of $\langle E_{k_1}(sk), F_{sk}(m_b)\rangle$ with receiver A_1. This ciphertext is valid. He thus obtains m_b and thus the bit b. This attack is successful because the broadcast ciphertext is *malleable*.

In the following, we show that if a NICKDS satisfies the security requirements given in Section 3, then the NICKDS-based BES scheme will be immune against this type of attack and will be secure under Definition 7. To our knowledge, this is the first formal proof that a *symmetric cryptography* based broadcast encryption has a CCA2 like security. (Naor et al.'s scheme has only CCA1 security and Dodis et al. [17] has a CCA2 security in the *public key setting*.) The proof is to use the reduction to security NICKDS and appears in the full paper.

THEOREM 8. *Let* $(\mathsf{KeyDist}, \mathsf{KeyDer})$ *be a $(w, t, \infty, 0)$-secure* $(\Gamma_{\geq r}, n)$-NICKDS *and (E, D) be an (ϵ, T_1)-CCA2-secure symmetric encryption scheme. Then the* NICKDS*-based broadcast encryption is (w, t, T, ϵ)-secure, where $T = T_1 - Q - t_i$ and Q is the time to initialize NICKDS and t_i is the time to sample a random conference set, $\epsilon = \epsilon' \sum_{\nu=0}^{n-r} \binom{n}{\nu}$.*

Remark. Note that in this theorem, NICKDS is assumed to be $(w, t, \infty, 0)$-secure. This is mainly to simplify the proofs. One can prove a similar result if the NICKDS is (w, t, T_2, ϵ_2)-secure.

8. CONCLUSION AND OPEN QUESTION

Non-interactive key agreement has been only analysed against corruption-only adversaries. In this paper, we proposed a new security model for non-interactive conference key distribution which is a natural adaptation of the widely used security model for interactive setting. We considered the security of some of the well-known NICKDSs in the new model and showed the conditions under which the system is secure.

We gave two composition theorems for NICKDSs, one with a MAC and one with an encryption system. We showed that these compositions give two interesting cryptographic primitives: a secret key based ring authentication system and a broadcast encryption system that provides CCA2 security. These two primitives extend known models of ring authentication and broadcast encryption by adding new and important properties, and constructions with provable security.

An interesting aspect of our work is applicability of our results to information theoretic and computational setting both. We give a general method of converting a NICKDS with information theoretic security to one with computational security. However, security of the computationally secure scheme is in random oracle model.

Important open questions in this area include deriving information theoretic bounds on users' key size in the information theoretically secure NICKDs and, constructing NICKDS that are secure in standard model.

[2]Note here $\{k_j\}$ are *not* necessarily independent and so K_i can have a compact representation (but it is not our concern here).

9. REFERENCES

[1] M. Bellare, R. Canetti, and H. Krawczyk, a modular approach to the design and analysis of authentication and key exchange protocols, *STOC'98*, 419-428, 1998.

[2] M. Bellare, R. Canetti and H. Krawczyk, Keying Hash Functions for Message Authentication. *CRYPTO'96*, pp. 1-15.

[3] M. Bellare, D. Pointcheval and P. Rogaway, authenticated key exchange secure against dictionary attacks, *Advances in Cryptology-EUROCRYPT 2000*, B. Preneel (Ed.), LNCS 1807, Springer-Verlag, pp. 139-155, 2000.

[4] M. Bellare and P. Rogaway, entity authentication and key distribution, *Advances in Cryptology-CRYPTO 1993*, D. R. Stinson (Ed.), Springer-Verlag, LNCS 773, pp. 232-249, 1994.

[5] A. Bender, J. Katz and R. Morselli, Ring Signatures: Stronger Definitions, and Constructions Without Random Oracles, *TCC 2006*, S. Halevi and T. Rabin (Eds.), LNCS 3876, Springer-Verlag, pp. 60-79, 2006.

[6] R. Blom, An optimal Class of Symmetric Key Generation Systems, *Advances in Cryptology-EUROCRYPT'84*, LNCS 209, Springer-Verlag, pp. 335-338, 1984.

[7] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro and Moti Yung, Perfectly Secure Key Distribution for Dynamic Conferences, *Inf. Comput.* 146(1): 1-23 (1998)

[8] D. Boneh and A. Silverberg, Applications of Multilinear Forms to Cryptography, *Contemporary Mathematics*, Vol. 324, American Mathematical Society, pp. 71–90, 2003.

[9] E. Bresson, J. Stern, M. Szydlo, Threshold Ring Signatures and Applications to Ad-hoc Groups, *Advances in Cryptology-CRYPTO 2002*, M. Yung (Ed.), LNCS 2442, Springer-Verlag, pp. 465-480, 2002.

[10] E. Bresson, O. Chevassut, D. Pointcheval and J. Quisquater, Provably Authenticated Group Diffie-Hellman Key Exchange, *ACM CCS'01*.

[11] R. Canetti and H. Krawczyk, analysis of key-exchange protocols and their use for building secure channels, *Advances in Cryptology-EUROCRYPT 2001*, B. Pfitzmann (Ed.), LNCS 2045, Springer-Verlag, pp. 453-474, 2001.

[12] R. Canetti and H. Krawczyk, universally composable notions of key exchange and secure channels, *Advances in Cryptology-EUROCRYPT 2002*, L. R. Knudsen (Ed.), LNCS 2332, Springer-Verlag, pp. 337-351, 2002.

[13] D. Chaum, E. van Heyst, Group Signatures, *advances in Cryptology-EUROCRYPT 1991*, D. W. Davies (Ed.), LNCS 547, Springer-Verlag, pp. 257-265, 1991.

[14] S. M. Chow, V. K.-W. Wei, J. K. Liu and T. H. Yuen, Ring signatures without random oracles, *AsiaCCS 2006*, F. Lin et al (Eds.), pp. 297-302, Taipei, Taiwan, 2006.

[15] Y. Desmedt, V. Viswanathan, Unconditionally Secure Dynamic Conference Key Distribution, *ISIT'98*, pp. 383, Cambridge, MA, USA, August 16-31, 1998.

[16] W. Diffie and M. Hellman, new directions in cryptography, *IEEE Transactions on Information Theory*, Vol. 22, pp. 644-654, Nov. 1976.

[17] Y. Dodis and N. Fazio, public-key trace and revoke scheme secure against adaptive chosen ciphertext attack, *Public Key Cryptography 2003,* Y. Desmedt (Ed.), LNCS 2567, Springer-Verlag, pp. 100-115, 2003.

[18] Y. Dodis, A. Kiayias, Antonio Nicolosi and Victor Shoup, Anonymous Identification in Ad Hoc Groups, *Advances in Cryptology-EUROCRYPT 2004*, C. Cachin and J. Camenisch (Eds.), LNCS 3027, Springer-Verlag, pp. 609-626, 2004.

[19] A. Fiat and M. Naor, broadcast encryption, *Advances in Cryptology-CRYPTO 1993,* D. Stinson (Ed.), LNCS 773, Springer-Verlag, pp. 480-491, 1994.

[20] A. Joux, A One Round Protocol for Tripartite Diffie-Hellman, *ANTS 2000*, pp. 385-394, 2000.

[21] J. Katz and M. Yung, Scalable Protocols for Authenticated Group Key Exchange. *CRYPTO'03*.

[22] S. Kent and K. Seo, Security Architecture for the Internet Protocol, *Available at* http://www.rfc-editor.org/rfc/rfc4301.txt

[23] H. Krawczyk, the order of encryption and authentication for protecting communications (or: how secure is SSL?), *Advances in Cryptology-CRYPTO 2001*, J. Kilian (Ed.), LNCS 2139, Springer-Verlag, pp. 310-331, 2001.

[24] H. Kurnio, R. Safavi-Naini and H. Wang, A Group Key Distribution Scheme with Decenteralized User Join, *SCN'02*, S. Cimato et al. (Eds.), LNCS 2576, Springer-Verlag, pp. 146-163, 2003.

[25] T. Matsumoto and H. Imai, On the Key Predistribution System: A Practical Solution to the Key Distribution Problem, *Advances in Cryptology-CRYPTO'87*, LNCS 239, Springer-Verlag, pp. 185-193, 1987.

[26] C. J. Mitchell and F. C. Piper, Key Storage in Secure Network, *Discrete Applied Mathematics***21** (1988), 215-228.

[27] M. Naor, Deniable Ring Authentication, *Advances in Cryptology-CRYPTO'02*, M. Yung (Ed.), LNCS 2442, Springer-Verlag, pp. 481-498, 2002.

[28] D. Naor, M. Naor and J. Lotspiech, revocation and tracing schemes for stateless receivers, *Advances in Cryptology-CRYPTO 2001,* J. Kilian (Ed.), LNCS 2139, Springer-Verlag, pp. 41-62, 2001.

[29] R. Rivest, A. Shamir and L. Adleman, A Method for Obtaining Digital Signatures and Public-key Cryptosystems, *Communications of ACM*, Vol. 2, pp. 120-126, February 1978.

[30] R. L. Rivest, A. Shamir and Y. Tauman, How to Leak a Secret, *Advances in Cryptology-ASIACRYPT 2001*, pp. 552-565, 2001.

[31] R. Safavi-Naini, S. Wang and Y. Desmedt, Unconditionally secure ring authentication, *AsiaCCS 2007*, Feng Bao and Steven Miller (Eds.), pp. 173-181, Singapore, March 20-22, 2007.

[32] D. R. Stinson, On Some Methods for Unconditionally Secure Key Distribution and Broadcast Encryption, *Des. Codes Cryptography*, 12(3): 215-243 (1997).

[33] C. K. Wong, M. G. Gouda and S. S. Lam, secure group communication using key graphs, *ACM Sigcomm'98,* pp. 68-79, August 31 - September 4, 1998, Vancouver, B.C., Canada.