A Secure and Efficient Conference Key Distribution System

(Extended Abstract)

Mike Burmester[1]* and Yvo Desmedt[2]**

[1] Department of Mathematics, RH – University of London, Egham, Surrey TW20 OEX, U.K., e-mail uhah205@vax.rhbnc.ac.uk
[2] Department of EE & CS, University of Wisconsin – Milwaukee, P.O. Box 784, Milwaukee WI 53201, U.S.A., e-mail desmedt@cs.uwm.edu

Abstract. We present practical conference key distribution systems based on public keys, which authenticate the users and which are 'proven' secure provided the Diffie-Hellman problem is intractable. A certain number of interactions is needed but the overall cost is low. There is a complexity tradeoff. Depending on the network used, we either have a constant (in the number of conference participants) number of rounds (exchanges) or a constant communication and computation overhead. Our technique for authentication can be extended and used as the basis for an authentication scheme which is 'proven' secure against any type of attack, provided the Discrete Logarithm problem is intractable.

1 Introduction

To communicate securely over insecure channels it is essential that secret keys are distributed securely. Even if the encryption algorithm used is computationally infeasible to break, the entire system is vulnerable if the keys are not securely distributed. Key distribution is central to cryptography and has attracted a lot of attention (e.g., [17, 24, 6, 5, 30, 26, 31]). Research has focused on security and on efficiency. Many practical systems have been proposed [30, 26, 31, 36, 18, 38]. The most familiar system is the Diffie-Hellman key distribution system [17]. This enables two users to compute a common key from a secret key and publicly exchanged information. If more than two users want to compute a common key, then a conference key distribution system is used. Designing such systems can be particularly challenging because of the complexity of the interactions between the many users. Many conference key distribution systems have been presented recently [24, 25, 31, 36, 19, 8]. These however are either impractical, or only heuristic arguments are used to address their security. Our goal in this paper is to present a practical and proven secure conference key distribution system.

Ingemarsson, Tang and Wong proposed several conference key distribution systems in which the common key is a symmetric function [24]. These have many

* Research partly carried out while visiting the University of Wisconsin – Milwaukee.
** Research partly carried out while visiting Royal Holloway, University of London. Supported in part by NSF Grant NCR-9106327 and NSF Grant INT-9123464.

attractive features, particularly the second order system which has a low communication and computation overhead. However they demonstrated that this particular system is insecure because the information exchanged by the users makes it possible for a passive eavesdropper to compute the key. Our main system is similar, but we use *cyclic* functions. This prevents the attack by passive eavesdroppers whilst retaining the efficiency of the former scheme. For authentication we use a public key (interactive) authentication scheme which is proven secure assuming the Discrete Logarithm problem is intractable. Combining the two systems we get a conference key distribution scheme which is provably secure against *any* known type of attack, including those by malicious active adversaries working together, provided the Diffie-Hellman problem is intractable.

Our authentication scheme is of interest in itself, because of its efficiency and proven security. We note that all proven secure signature schemes presented so far [22, 28, 33, 1, 2] are impractical. We therefore extend our scheme so that it is proven secure against any type of attack, including adaptive chosen text attacks by real-time middle-persons, under the same cryptographic assumption. The resulting scheme is roughly as fast as RSA [32], but in addition is proven secure.

The organization of this paper is as follows. In Section 2 we give definitions and present our model for conference key distribution systems and for authentication schemes. In Section 3 we present various protocols for conference key distribution systems which are secure against attacks by passive eavesdroppers provided the Diffie-Hellman problem is hard. In Section 4 we adapt the protocols to get authentication. In Section 5 we present an authentication scheme which can be used to get a conference key distribution scheme which is secure against any type of attack. In Section 6 we extend the security of our authentication scheme, and we conclude in Section 7.

Because of page limitations there are no proofs. These will be given in the full version of the paper.

2 Definitions

In one of our scenarios we consider networks[3] in which the users U_i can *broadcast* 'messages' (strings) to each other. We allow for the possibility that an eavesdropper[4] E (a malicious adversary) may read the broadcast messages or substitute some of them. We distinguish two types of networks: those for which E is passive and those for which E is active. Let N be the security parameter.

[3] A network is a collection of n interactive probabilistic Turing machines U_i with e_i^1 write-only tapes, e_i^2 read-only tapes, a history tape, a knowledge tape and worktapes.

[4] An eavesdropper is an interactive probabilistic Turing machine with $\sum_{i=1}^{n} e_i^1$ read-only tapes T_{ij} and $\sum_{i=1}^{n} e_i^2$ write-only tapes W_{ij}. The eavesdropper reads from T_{ij} and writes on W_{ij}. If what is written is different from what is read then the eavesdropper is active. Otherwise the eavesdropper is passive. This, together with our definition of a network, allows for a scenario in which a broadcasted message can be substituted for each individual receiver. Eavesdroppers are polynomially bounded.

Definition 1. Suppose that $n = O(N^c)$, $c > 0$ constant, interactive Turing machines $U_1, \ldots, U_n$ take part in a protocol to generate a key. We say that the protocol is a *conference key distribution system* if, when all the $U_1, \ldots, U_n$ are as specified, then each U_i computes the same key $K = K_i$. A conference key distribution system *guarantees privacy* if it is computationally infeasible for a passive eavesdropper to compute the key K.

Definition 2. Suppose that $n = O(N^c)$ interactive Turing machines $U_1, \ldots, U_n$ use a conference key distribution system, and that each U_i has received (from an oracle) a secret key s_i (written on its knowledge tape) which corresponds to its public key k_i, which is published. Let $n' > 0$ of these be honest[5], $n'' = n - n' \geq 0$ be impersonators[5], and assume that there is a secure network between the impersonators and the (passive or active) eavesdropper. A conference key distribution system is (computationally) *secure*, if it is computationally infeasible for any set of n'', $0 \leq n'' < n$, impersonators U_j' in collaboration with the eavesdropper to compute the same key K_i as computed by any of the honest machines U_i.

Remark. If the set of impersonators is empty then we require that the (active) eavesdropper cannot compute K_i.

Definition 3. (Informal) Consider a network with eavesdropper E. A protocol (U_1, U_2) in which U_1 sends a message m is an *authentication system* if,

- *Compliance*: When U_1, U_2 are honest and E is passive then U_2 accepts and outputs m with overwhelming probability,
- *Secure against impersonation*: U_2 rejects with overwhelming probability a dishonest U_1',
- *Secure against substitution*: If E is active and U_2 outputs $m' \neq m$ then U_2 rejects with overwhelming probability.

Definition 4. *The Diffie-Hellman* [17] *problem*: given p, α, β, γ, find $\beta^{\log_\alpha \gamma} \bmod p$ if it exists.

Breaking this problem has remained an open problem for more than 15 years. Even if the factorization of the order of α is known [29, 15, 27, 23] the problem is assumed to be hard (cf. [7, 9, 10]). It is well known that if the Discrete Logarithm problem is easy then so is the Diffie-Hellman problem, but the converse may not be true.

3 Private Conference Key Distribution

In this section we are only concerned with privacy. Authenticity is addressed in Section 5. We consider various conference key distribution systems which are

[5] An honest machine U_i has a secret key s_i written on its knowledge tape. An impersonator U_j' is any polynomially bounded interactive probabilistic Turing machine which replaces U_j but does not have the secret key of U_j (or an equivalent). In our model the eavesdropper is not an impersonator: it can only impersonate U_i with the help of an impersonator (if there is one). We will strengthen the definition in the final paper to allow for insiders' attacks.

based on the Diffie-Hellman [17] key exchange. These are designed to exploit the particular configuration of the network used. Our main protocol is in Section 3.3 and Section 3.4. The other protocols are given for comparison.

We use a discrete logarithm setting. A center chooses a prime $p = \Theta(2^{N^c})$, $c \geq 1$ constant, and an element $\alpha \in Z_p$ of order $q = \Theta(2^N)$. If the order has to be verified then the factorization of q is given. The center then publishes p, α and q. Let n be polynomially bounded in the length of p.

3.1 A Star Based System

In this system a chair U_1 exchanges a Diffie-Hellman key K_i with each user U_i, and then chooses a random conference key K which it sends to each U_i encrypted under K_i. That is,

Protocol 1. Let $U_1, \ldots, U_n$ be a dynamic set of users[6] who want to generate a common conference key. U_1 is the chair.

Step 1 Each U_i, $i = 1, \ldots, n$, selects[7] $r_i \in_R Z_q$ and computes $z_i = \alpha^{r_i} \bmod p$. Then U_1 sends z_1 to all the U_i and the U_i send z_i to U_1, $i = 2, \ldots, n$.

Step 2 U_1 checks[8] that $\mathrm{ord}(\alpha) = q$. Then U_1 computes $K_i = z_i^{r_1} \bmod p$ for $i = 2, \ldots n$, and selects[9] a conference session key $K \in_R \langle\alpha\rangle$. U_1 sends[10] $Y_i \equiv K \cdot K_i \pmod p$ to each U_i, $i = 2, \ldots n$.

Step 3 Each U_i, $i = 2, \ldots, n$, checks[8] that $\mathrm{ord}(\alpha) = q$, computes $K_i \equiv z_1^{r_i} \bmod p$, and decrypts Y_i to get the session key K.

3.2 A Tree Based System

This is similar to the star based system, except that a tree configuration network is used. The users $U_1, U_2, \ldots$ are labelled in such a way that the sons of U_a are U_{2a} and U_{2a+1}. U_1 is the root.

Protocol 2. Let $U_1, \ldots, U_n$ be a dynamic set of users who want to generate a common conference key. U_1 is the chair.

Step 1 Each U_a in the conference selects $r_a \in_R Z_q$ and computes $z_a = \alpha^{r_a} \bmod p$. Then U_a sends z_a to $U_{\lfloor a/2 \rfloor}$, if $a > 1$, and to U_{2a} if $2a \leq n$, and to U_{2a+1} if $2a + 1 \leq n$.

Step 2 Each U_a in the conference checks[8] that $\mathrm{ord}(\alpha) = q$. Then if $a > 1$ it computes $K_a = z_{\lfloor a/2 \rfloor}^{r_a} \bmod p$ and $K_{2a+i} = z_{2a+i}^{r_a} \bmod p$, $i = 0, 1$, if $2a + i \leq n$. U_1 selects a conference session key $K \in_R \langle\alpha\rangle$ and then sends[10] $Y_{2+i} = K \cdot K_{2+i} \bmod p$ to U_{2+i}, $i = 0, 1$. Set $\ell = 0$.

---

[6] Any set of n users, which may dynamically change.

[7] We use the notation $a \in_R A$ to indicate that a is selected from the set A uniformly and independently.

[8] This check is only done once. If the center is trusted (oracle) it is even not required.

[9] $\langle\alpha\rangle$ is the multiplicative group generated by α in Z_p^*.

[10] Other encryption schemes may be used.

Step 3+ℓ If U_a is at level ℓ of the tree (if $\lfloor \log_2 a \rfloor = \ell$), then U_a decrypts Y_a to get K, and then sends $Y_{2a+i} = K \cdot K_{2a+i} \bmod p$ to U_{2a+i}, $i = 0, 1$, if $2a + i \leq n$. Set $\ell := \ell + 1$.

Remark. The users in conference must trust each other against jamming. If U_a replaces the key K by K' then all his descendents will use K', and not K.

3.3 A Broadcast System

Protocol 3. Let $U_1, \ldots, U_n$ be a dynamic set of users who want to generate a common conference key. The indices are taken in a cycle: so U_{n+1} is U_1, and U_0 is U_n.

Step 1 Each U_i, $i = 1, \ldots, n$, selects $r_i \in_R Z_q$, and then computes and broadcasts $z_i = \alpha^{r_i} \bmod p$.

Step 2 Each U_i, $i = 1, \ldots, n$, checks[8] that $\mathrm{ord}(\alpha) = q$. Then it computes and broadcasts

$$X_i \equiv (z_{i+1}/z_{i-1})^{r_i} \pmod{p}.$$

Step 3 Each U_i, $i = 1, \ldots, n$, computes the conference key,

$$K_i \equiv (z_{i-1})^{nr_i} \cdot X_i^{n-1} \cdot X_{i+1}^{n-2} \cdots X_{i-2} \pmod{p}.$$

Remark. Honest users compute the same key,

$$K \equiv \alpha^{r_1 r_2 + r_2 r_3 + \cdots + r_n r_1} \pmod{p}.$$

Indeed, set $A_{i-1} \equiv (z_{i-1})^{r_i} \equiv \alpha^{r_{i-1} r_i} \pmod{p}$, $A_i \equiv (z_{i-1})^{r_i} \cdot X_i \equiv \alpha^{r_i r_{i+1}} \pmod{p}$, $A_{i+1} \equiv (z_{i-1})^{r_i} \cdot X_i \cdot X_{i+1} \equiv \alpha^{r_{i+1} r_{i+2}} \pmod{p}$, etc., and we have $K_i = A_{i-1} \cdot A_i \cdot A_{i+1} \cdots A_{i-2}$. So the key is a second order cyclic function of the r_i (but not symmetric as in [24]).

For $n = 2$ we get $X_1 = X_2 = 1$ and $K \equiv \alpha^{r_1 r_2 + r_2 r_1} \equiv \alpha^{2 r_1 r_2} \pmod{p}$, which is essentially the same as for the Diffie-Hellman [17] system (clearly there is no need to broadcast X_1, X_2 in this case).

3.4 A Cyclic System

This is similar to the broadcast system except that a bi-directional cyclic network is used. So $U_1, \ldots, U_n$ are linked in a cycle, with U_i connected to U_{i+1}.

Protocol 4. Let $U_1, \ldots, U_n$ be a dynamic set of users who want to generate a common conference key.

Step 1 Each U_i, $i = 1, \ldots, n$, selects $r_i \in_R Z_q$, and then computes and sends $z_i = \alpha^{r_i} \bmod p$ to U_{i-1} and U_{i+1}. Then U_i checks[8] that $\mathrm{ord}(\alpha) = q$.

Step 2 Each U_i $i = 1, \ldots, n$, computes $X_i \equiv (z_{i+1}/z_{i-1})^{r_i} \pmod{p}$. Let $i = 1$. Let $b_0 = c_0 = 1$.

Step 3+$i-1$ U_i sends to U_{i+1} (b_i, c_i) where $b_i = X_1 \cdot X_2 \cdots X_i \pmod{p}$, and $c_i = X_1^{i-1} \cdot X_2^{i-2} \cdots X_{i-1} \pmod{p}$. Observe that $c_i := b_{i-1} \cdot c_{i-1} \pmod{p}$. Let $i := i+1$. Let $\ell = 1$.

Step $n+1+\ell$ U_ℓ sends to $U_{\ell+1}$: $X_1 \cdot X_2 \cdots X_n \pmod{p}$, and $d_\ell = X_{\ell+1}^{n-1} \cdot X_{\ell+2}^{n-2} \cdots X_{\ell-1} \pmod{p}$. Observe that $d_\ell := (X_1 \cdot X_2 \cdots X_n) \cdot d_{\ell-1} \cdot X_\ell^{-n} \pmod{p}$.

Step $2n+2$ Each U_i, $i = 1, \ldots, n$, computes the conference key,

$$\begin{aligned} K_i &\equiv (z_{i-1})^{nr_i} \cdot X_i^{n-1} \cdot X_{i+1}^{n-2} \cdots X_{i-2} \\ &\equiv \alpha^{r_1r_2+r_2r_3+\cdots+r_nr_1} \pmod{p}. \end{aligned}$$

3.5 Security

Theorem 5. *If n is polynomially bounded in the length of p and if the Diffie-Hellman problem is intractable, then Protocols 1,2,3, and 4 are conference key distribution systems which guarantee privacy.*

3.6 A Comparison of the Communication and Computational Complexity of the Proposed Systems

In the following table we summarize the communication and computational costs of the proposed systems (compared to the Diffie-Hellman scheme).

PRIVACY (without authenticity)					
Complexity	Star		Tree	Broadcast	Cyclic
	chair	others			
Communication*	$\star\, 2(n-1)^\dagger$	$\star\, 1$	$\star\, 5^\ddagger$	$\star\, 2$	$\star\, 6$
Round	2	2	$1 + \lceil \log n \rceil$	2	$2n+1$
Computation*	$\star\, n^\S$	$\star\, 2$	$\star\, 4$	$\star\, 2 + n\lceil \log n \rceil / \lceil \log p \rceil$	$\star$ constant
Delay	In the final paper				

* Per user.

† This means $2(n-1) \star \log p$ bits.

‡ Users corresponding to leafs have lower communication and computation costs.

§ This means $n \star 2 \log p$ multiplications.

Remark. Clearly anybody can masquerade as U_i in the protocols described in this section. So the users are not authenticated. In the following section we present an authentication scheme which, when combined with the systems above, offers both privacy and authentication.

4 Authenticated Conference Key Distribution

In this section we use a general authentication protocol, e.g. signatures [22].

Remark. One has to be careful when using authentication to achieve authenticated key distribution [18]. We discuss this problem and time dependency problems in more details in the final paper.

We are mainly interested in indirect authentication [18].

4.1 Star Based Authentication

Protocol 5.
Each U_i, $i = 2, \ldots, n$, in Protocol 1 authenticates z_i to U_1, and U_1 authenticates z_1 to all U_i. Then U_1 sends Y_i to each U_i, $i = 2, \ldots, n$. If some U_i fails to authenticate z_i then if $i > 1$, U_1 does not send Y_i, else ($i = 1$) one stops.

4.2 Tree Based Authentication

Protocol 6.
Each U_a in Protocol 2 authenticates z_a to its parent $U_{\lfloor a/2 \rfloor}$ and to its sons U_{2a} and U_{2a+1} (if these nodes exist). If the authentication of some z_a fails no further communication with U_a takes place. There is no need to authenticate Y_{2a+i} if we are only interested in indirect authentication.

A variation of this scheme is obtained by having each U_a authenticate z_a to its parent in the first round, sequentially (from leafs to root, no parallelism between levels). Then, in the next round, the reverse procedure is used. This idea can be adapted to authenticate the broadcast and cyclic systems (Protocols 3 and 4). Details will be given in the final paper.

4.3 Broadcast Authentication

Protocol 7. Each U_i in Protocol 3 authenticates z_i to U_{i+1}, $i = 1, \ldots, n$. If the authentication of z_i fails then U_{i+1} halts. Then this process is repeated sequentially. That is, U_1 first authenticates z_1 to U_2. Then each U_i, $i = 2, \ldots, n$ waits until z_{i-1} is authenticated, and if this is successful, it authenticates the empty string to U_{i+1}. This second round serves to guarantee that all the z_i are authenticated, as will be explained in the full paper.

4.4 Cyclic Authentication

Protocol 8. This is essentially the same as the previous protocol, the only difference being that a cyclic network is used.

4.5 Security

Theorem 6. *If n is polynomially bounded in the length of p and if the Diffie-Hellman problem is intractable, and if a secure authentication scheme is used then Protocols 5,6,7, and 8 are conference key distribution systems which are secure against impersonation and substitution attacks.*

4.6 A Comparison of the Communication and Computational Complexity of the Authenticated Systems

This is similar to Section 3.6. A table with details is given in the full paper.

5 An Authentication Scheme

5.1 The Basic Scheme

As in Section 3, a center chooses p, α and q, but now q is a prime. Then each user P selects $a, b \in_R Z_q$, computes $\beta = \alpha^a \bmod p$, $\gamma = \alpha^b \bmod p$, and registers $k = (\beta, \gamma)$ as its public key.[11]

Protocol 9. *Common input:* $(p, \alpha, q, \beta, \gamma)$.
P has a, b written on the knowledge tape, where $\beta = \alpha^a \bmod p$, $\gamma = \alpha^b \bmod p$. P is given $z \in Z_q$.

P authenticates z to V: P sends z to V and then proves to V that it knows the discrete logarithm of $\beta^z \gamma \bmod p$ $(= az + b \bmod q)$, by using any interactive zero-knowledge proof of knowledge (*e.g.*, [14, 13, 3, 16]).
V verifies this and checks[8] that $\alpha \not\equiv 1 (\bmod p)$, $\alpha^q \equiv \beta^q \equiv \gamma^q \equiv 1 (\bmod p)$ and that q is a prime. If this fails V halts.

Theorem 7. *Protocol 9 is an authentication scheme secure against a generic chosen-message attack ($z \in Z_q$ is chosen independently of γ) if the order of α is prime, provided the Discrete Logarithm problem is intractable.*

Remark. Although zero-knowledge proofs do not guarantee inherently secure identification [4], in the context of authentication only real-time attacks in which the message is not authentic (*e.g.*, substituted) make sense. To prevent real-time substitution attacks in which the adversary combines proofs of knowledge of different messages, only one proof at a time must be ran. We shall discuss such real-time attacks and ways to avoid them in the full version of the paper.

5.2 Application to Key Distribution

Theorem 8. *Let p_1, α_1 and q_1 be as in Section 3, and p_2, α_2 and q_2 be as in Section 5 with q_2 a prime and $p_1 \leq q_2$. If each U_i authenticates z_i as in Protocol 9 with parameters p_2, α_2, q_2 and public key $k_i = (\beta_2, \gamma_2)$, as required in each of the protocols of Section 4, then the conference key distribution systems are secure against impersonation and substitution attacks, provided the Diffie-Hellman problem is intractable.*

Corollary 9. *Protocol 9 can be replaced by any proven secure authentication scheme, provided its security assumption is added to the conditions of Theorem 8.*

[11] There is no need for p, q to be standard.

6 A Proven Secure Authentication Scheme

The authentication Protocol 9 has not been proven secure against a chosen attack. Indeed in Theorem 7 the proof of security against a substitution attack relies on the independence of the message from γ, of the public key. We now will modify Protocol 9 to obtain security against all known attacks, including adaptive chosen text attacks.

Let $(p_2, \alpha_2, q_2), (p_3, \alpha_3, q_3)$ be as in Section 5 with $p_2 \leq q_3$, and $k = (\beta_2, \beta_3, \gamma_3)$ be the public key of user U, with $\beta_2 = \alpha_2^{a_2} \bmod p_2$, $\beta_3 = \alpha_3^{a_3} \bmod p_3$, $\gamma_3 = \alpha_3^{b_3} \bmod p_3$, $a_2 \in_R Z_{q_2}$, $a_3, b_3 \in_R Z_{q_3}$. The following protocol is used to authenticate any number $z \in Z_{q_2}$.

Protocol 10. *Common input:* $(p_2, \alpha_2, q_2, p_3, \alpha_3, q_3\,;\, \beta_2, \beta_3, \gamma_3)$.
P has written on its knowledge tape a_2, a_3, b_3, where $\beta_2 = \alpha_2^{a_2} \bmod p_2$, $\beta_3 = \alpha_3^{a_3} \bmod p_3$, $\gamma_3 = \alpha_3^{b_3} \bmod p_3$. P is given $z \in Z_{q_2}$.

P authenticates z to V: P sends to V: z and $\gamma_2 = \alpha_2^{b_2} \bmod p_2$, where $b_2 \in_R Z_{q_2}$, and then proves to V, simultaneously, that it knows the discrete logarithm base α_2 of $\beta_2^z \cdot \gamma_2 \bmod p_2$ $(= a_2 z + b_2 \bmod q_2)$, and the discrete logarithm base α_3 of $\beta_3^{\gamma_2} \cdot \gamma_3 \bmod p_3$ $(= a_3\gamma_2 + b_3 \bmod q_3)$, by using a zero-knowledge proof of knowledge (*e.g.*, [14, 13, 3, 16]).
V verifies this, checks that $\gamma_2^{q_2} \equiv 1 \,(\bmod p_2)$, and then checks[8] that $\alpha \not\equiv 1 (\bmod p)$, $\alpha_2^{q_2} \equiv \beta_2^{q_2} \equiv 1 \,(\bmod p_2)$, $\alpha_3^{q_3} \equiv \beta_3^{q_3} \equiv \gamma_3^{q_3} \equiv 1 \,(\bmod p_3)$ and that q_2, q_3 are primes and $p_2 \leq q_3$. If this fails V halts.

Theorem 10. *Protocol 9 is a secure authentication scheme if the Discrete Logarithm problem is intractable.*

7 Conclusion

We have presented a variety of conference key distribution systems which are proven secure against a passive adversary if the Diffie-Hellman problem (a 15 year open problem) is hard. The session key of our main system is a cyclic function (of the indices of the users) of degree two, which is the main reason for its practicality. Ingemarsson Tang and Wong considered conference systems for which the key was a *symmetric* function of degree two, but these were insecure. Shamir's signature scheme [35], cryptanalyzed by Coppersmith and Stern, also uses symmetric functions. Our results suggest that cyclic functions still have some use in cryptography. Although it is hard for an adversary to compute the session key, it is not clear which bits of this key are hard. Since this problem is also open for the Diffie-Hellman key exchange, it is beyond the scope of this paper.

To achieve security against active adversaries we have extended our conference key distribution protocol. Users have a public key and authenticate their messages using an appropriate authentication scheme. The resulting system is

proven secure against an active attack under the same assumptions as before, while remaining practical.

The authentication used in our protocol is only proven secure against a generic chosen-message attack [22], *i.e.*, an attack in which the message to be authenticated is chosen independently of the public key (which is sufficient for the security of the conference key system). We have extended our authentication system so that it is also proven secure against an adaptive chosen text attack by a real time middle-person provided the Discrete Logarithm problem is intractable. This resulting scheme remains practical.

Acknowledgements

The authors wish to thank René Peralta, Adi Shamir, Oded Goldreich, and Moti Yung, for helpful discussions and suggestions, in particular in Section 3.1. Also Kevin McCurley, Tom Berson and Paul van Oorschot for suggestions and various improvements.

References

1. M. Bellare, S. Goldwasser: New paradigms for digital signatures and message authentication based on non-interactive zero-knowledge proofs. In: G. Brassard, (ed.): Advances in Cryptology – Crypto '89. Lecture Notes in Computer Science 435. Berlin: Springer 1990, pp. 194–211
2. M. Bellare, S. Micali: How to sign given any trapdoor function. Journal of the ACM 39, 214–233 (1992)
3. M. Bellare, S. Micali, R. Ostrovsky: Perfect zero-knowledge in constant rounds. In: Proceedings of the Twenty Second Annual ACM Symp. Theory of Computing. ACM Press 1990, pp. 482–493
4. S. Bengio, G. Brassard, Y.G. Desmedt, C. Goutier, J.-J. Quisquater: Secure implementations of identification systems. Journal of Cryptology 4, pp. 175–183 (1991)
5. C. H. Bennett, G. Brassard: Quantum cryptography, and its application to provable secure key expansion, public-key distribution, and coin tossing. In: International Symposium on Information Theory (abstracts), IEEE Computer Society Press 1983, p. 91
6. R. Blom: Key distribution and key management. In: Proc. Eurocrypt 83, Udine, Italy, March 1983.
7. M. Blum, S. Micali: How to generate cryptographically strong sequences of pseudo-random bits. Siam J. Comput. 13, 850–864 (1984)
8. C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, M. Yung: Perfectly-secure key distribution for dynamic conferences. In: E. Brickell (ed.): Advances in Cryptology – Crypto 92. Lecture Notes in Computer Science 740. Berlin: Springer 1993, pp. 471–487
9. J. Boyar, M.W. Krentel, S.A. Kurtz: A discrete logarithm implementation of zero-knowledge blobs. Technical Report 87-002, University of Chicago, March 1987.
10. G. Brassard, D. Chaum, C. Crépeau: Minimum disclosure proofs of knowledge. Journal of Computer and System Sciences 37, 156–189 (1988)
11. M. Burmester: On the risk of opening distributed keys. To appear in the Proceedings of Crypto '94. Berlin: Springer 1994.

12. J.L. Carter, M.N. Wegman: Universal classes of hash functions. Journal of Computer and System Sciences 18, 143–154 (1979)
13. D. Chaum, J.-H. Evertse, J. van de Graaf: An improved protocol for demonstrating possession of discrete logarithms and some generalizations. In: D. Chaum, W.L. Price (eds.): Advances in Cryptology – Eurocrypt '87. Lecture Notes in Computer Science 304. Berlin: Springer 1988, pp. 127–141
14. D. Chaum, J.-H. Evertse, J. van de Graaf, R. Peralta: Demonstrating possession of a discrete logarithm without revealing it. In: A. Odlyzko (ed.): Advances in Cryptology – Crypto '86. Lecture Notes in Computer Science 263. Berlin: Springer 1987, pp. 200–212
15. D. Coppersmith, A. Odlyzko, R. Schroeppel: Discrete logarithms in $GF(p)$. Algorithmica, pp. 1–15 (1986)
16. Y. Desmedt, M. Burmester: An efficient zero-knowledge scheme for the discrete logarithm based on smooth numbers. In: H. Imai, R.L. Rivest, T. Matsumoto (eds.): Advances in Cryptology – Asiacrypt '91. Lecture Notes in Computer Science 739. Berlin: Springer 1992, pp. 360–367
17. W. Diffie, M. E. Hellman: New directions in cryptography. IEEE Trans. Inform. Theory IT–22, 644–654 (1976)
18. W. Diffie, P.C. van Oorschot, M.J. Wiener: Authentication and authenticated key exchanges. Designs, Codes and Cryptography 2, 107–125 (1992)
19. M. J. Fischer, R. N. Wright: Multiparty secret key exchange using a random deal of cards. In: J. Feigenbaum (ed.): Advances in Cryptology – Crypto '91, Lecture Notes in Computer Science 576. Berlin: Springer 1992, pp. 141–155
20. Z. Galil, S. Haber, M. Yung: A private interactive test of a Boolean predicate and minimum-knowledge public key cryptosystems. In: Annual Symp. on Foundations of Computer Science. IEEE Computer Society Press 1985, pp. 360–371
21. S. Goldwasser, S. Micali, C. Rackoff: The knowledge complexity of interactive proof systems. Siam J. Comput. 18, 186–208 (1989)
22. S. Goldwasser, S. Micali, R. Rivest: A digital signature scheme secure against adaptive chosen-message attacks. Siam J. Comput. 17, 281–308 (1988)
23. D. Gordon: Discrete logarithm in $GF(p)$ using the number field sieve. Submitted.
24. I. Ingemarsson, D.T. Tang, C.K. Wong: A conference key distribution system. IEEE Trans. Inform. Theory 28, 714–720 (1982)
25. K. Koyama, K. Ohta: Identity-based conference key distribution systems. In: C. Pomerance (ed.): Advances in Cryptology – Crypto '87. Lecture Notes in Computer Science 293. Berlin: Springer 1988, pp. 175–185
26. K.S. McCurley: A key distribution system equivalent to factoring. J. Cryptology 1, 95–105 (1988)
27. A. Menezes, S. Vanstone, T. Okamoto: Reducing elliptic curve logarithms to logarithms in a finite field. In: Proceedings of the Twenty Third Annual ACM Symp. Theory of Computing. ACM Press 1991, pp. 80–89
28. M. Naor, M. Yung: Universal one-way hash functions and their cryptographic applications. In: Proceedings of the Twenty First Annual ACM Symp. Theory of Computing. ACM Press 1989, pp. 33–43
29. A.M. Odlyzko: Discrete logs in a finite field and their cryptographic significance. In: N. Cot, T. Beth, I. Ingemarsson, (eds.): Advances in Cryptology – Eurocrypt 84. Lecture Notes in Computer Science 209. Berlin: Springer 1984, pp. 224–314

30. E. Okamoto: Key distribution systems based on identification information. In: C. Pomerance (ed.): Advances in Cryptology – Crypto '87. Lecture Notes in Computer Science 293. Berlin: Springer 1988, pp. 194–202
31. E. Okamoto, K. Tanaka: Key distribution system based on identification information. IEEE J. Selected Areas in Commun. 7, 481–485 (1989)
32. R.L. Rivest, A. Shamir, L. Adleman: A method for obtaining digital signatures and public key cryptosystems. Commun. ACM 21, 120–126 (1978)
33. J. Rompel: One-way functions are necessary and sufficient for secure signatures. In: Proceedings of the Twenty Second Annual ACM Symp. Theory of Computing. ACM Press 1990, pp. 387–394
34. A. W. Schrift, A. Shamir: The discrete log is very discreet. In: Proceedings of the Twenty Second Annual ACM Symp. Theory of Computing. ACM Press 1990, pp. 405–415
35. A. Shamir: Efficient signature schemes based on birational permutations. To appear in the Proceedings of Crypto '93. Berlin: Springer.
36. S. Tsujii, T. Itoh: An ID-based cryptosystem based on the discrete logarithm. IEEE J. Selected Areas in Commun. 7, 467–473 (1989)
37. M.N. Wegman, J.L. Carter: New hash functions and their use in authentication and set equality. J. Computer and System Sciences 22, 265–279 (1981)
38. Y. Yacobi, Z. Shmuely: On key distribution systems. In: G. Brassard (ed.): Advances in Cryptology – Crypto '89. Lecture Notes in Computer Science 435. Berlin: Springer 1990, pp. 344–355