

Secret Handshakes with Dynamic and Fuzzy Matching

Marina Blanton
Purdue University

Joint work with Giuseppe Ateniese and Jonathan Kirsch
The Johns Hopkins University

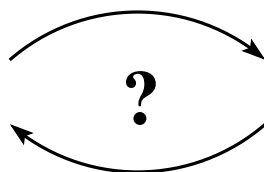
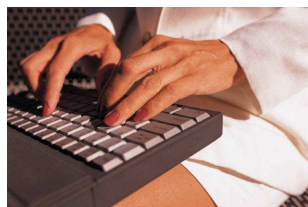
March 2007

What are Secret Handshakes?

- A **secret handshake scheme** allows two members of the **same group** to:
 - secretly authenticate to each other
 - agree on a shared key for further communication
- If the protocol succeeds, the players only learn that they are valid members of the same group
- If the protocol fails, no information is leaked
 - neither player knows why it failed
 - neither player learns anything about the group of the other
 - outside observers learn nothing about either party

Examples of Secret Handshakes

- **Most common example**
 - A **CIA agent Alice** wants to authenticate to **another CIA agent**
 - Alice doesn't want to reveal her credentials to anyone other than a **CIA agent**
 - Alice will **successfully complete the handshake** with **Bob** only if **Bob is also a CIA agent**



Examples of Secret Handshakes (cont.)

- **Secret handshakes are also useful for**
 - authentication between members of **secret societies**
 - discovery and use of **secret services** by the military
 - and others
- **More practically**
 - secret handshakes can be used for **video protection**
 - in **High-bandwidth Digital Content Protection (HDCP)** systems
 - networks for **family communication**
 - recently they were used to efficiently realize an **anonymous routing protocol** in ad-hoc networks

Secret Handshakes with Roles

- In secret handshakes **with roles**
 - Alice and Bob still must belong to the same group to succeed
 - Alice **specifies the role Bob must have** within the group
 - Bob **specifies the role Alice must have**
 - The protocol succeeds only if Alice and Bob have the specified credentials
- **Example**
 - Alice will authenticate as a **vehicle operator** to Bob only if Bob can authenticate as a **policeman**
- Roles and other extensions make the problem interesting and challenging to work on

Background

- **The problem was introduced by Balfanz et al. in 2003**
 - it has received significant attention since then
 - other anonymity tools cannot adequately solve this seemingly simple problem
- **Most existing solutions either**
 - use one-time credentials to maintain anonymity
 - or lack efficiency

How Secret Handshakes are Defined

- A **secret handshake scheme (SHS)** consists of the following algorithms:
 - **System Setup**
 - **Create Group**
 - **Add Member**
 - **Perform Handshake**
- If **traceability** is supported, there is also a **Trace User** algorithm
- If user **revocation** is possible, we also have **Revoke User**

How Secret Handshakes are Defined (cont.)

- **The core properties of a SHS we require:**
 - **Correctness**
 - **Impersonator resistance**
 - **Detector resistance**
 - **Unlinkability**
- **Other security properties were considered in the literature**

Our Approach

- We take the flexibility of secret handshakes with roles to a new level:
 - in addition to specifying the role of the other party, now members can also **specify the group of the other party**
 - we call this **dynamic matching**
 - this allows to authenticate, e.g., with members of sister societies
- In addition, we introduce **attribute-based secret handshakes**
 - now each **member has a number of attributes** (call it n)
 - during a handshake, Alice specifies the attributes Bob must have
 - this allows for **approximate (or fuzzy) matching** for some overlap threshold $d \leq n$
- We call both these new types of secret handshakes **unrestricted**

Overview of the Result

- Our solutions to both types of handshakes rely on **Identity Based Encryption (IBE)**
- Secret handshakes with fuzzy matching use **fuzzy IBE** and also **privacy-preserving set intersection protocols**
- We slightly modify the IBE schemes to achieve privacy
- We implement our secret handshake with dynamic matching and **integrate it with IPsec**
- We also provide a prototype implementation of handshakes with fuzzy matching

Secret Handshakes with Dynamic Matching

- **(Background)** In an IBE scheme:
 - an authority sets up the system
 - any string (e.g., an email address) can be used as a public encryption key
 - a user obtains a private decryption key corresponding to her identity and can recover messages encrypted under her identity
- We use the IBE scheme due to **Waters (2005)**
 - encryption of M under the key ID is formed as
$$E_{ID}(M) = (E_1, E_2, E_3) = (vM, v', v'')$$
 - a user with identity ID has decryption key is d_{ID}
 - to decrypt, compute v using $v', v'',$ and d_{ID} and recover M

Secret Handshakes with Dynamic Matching (cont.)

- **Challenge**
 - in secret handshakes **IDs must be hidden**
 - the above construction is **not private**
 - given a ciphertext, everyone can test whether it was encrypted under a certain ID
- **We slightly modify the construction**
 - the change is at the lower level of pairing-based cryptography
 - the construction given above remains unchanged

Secret Handshakes with Dynamic Matching (cont.)

- **Our solution:**
 - **System Setup:** initialize the IBE scheme
 - **Create Group:** no action needed
 - **Add Member:** to add a member with role r to a group G
 - compute the representation of " $G||r$ " (call it ID)
 - issue the key d_{ID} to the user
 - **Handshake:** let users A and B engage in a handshake
 - A has d_A for $A = rep(G_A||r_A)$;
 B has d_B for $B = rep(G_B||r_B)$
 - A specifies credentials G'_B and r'_B that B must have;
 B specifies credentials G'_A and r'_A that A must have

Secret Handshakes with Dynamic Matching (cont.)

- **Handshake: (cont.)**
 - **A computes $B' = \text{rep}(G'_B || r'_B)$ and $E_{B'}(1) = (v_{B'}, v'_{B'}, v''_{B'})$**
 - **Similarly, B computes $A' = \text{rep}(G'_A || r'_A)$ and $E_{A'}(1) = (v_{A'}, v'_{A'}, v''_{A'})$**
 - **A sends $v'_{B'}$ and $v''_{B'}$ to B , B sends $v'_{A'}$ and $v''_{A'}$ to A**
 - **The shared key is $k = (k_1, k_2) = (v_{B'}, v_{A'})$**
- **A will be able to recover $k_2 = v_{A'}$ (using $v'_{A'}$, $v''_{A'}$, and d_A) iff $A' = A$**
- **Similar for B and k_1**
- **If A and B obtain different keys, they don't know what went wrong**

Secret Handshakes with Fuzzy Matching

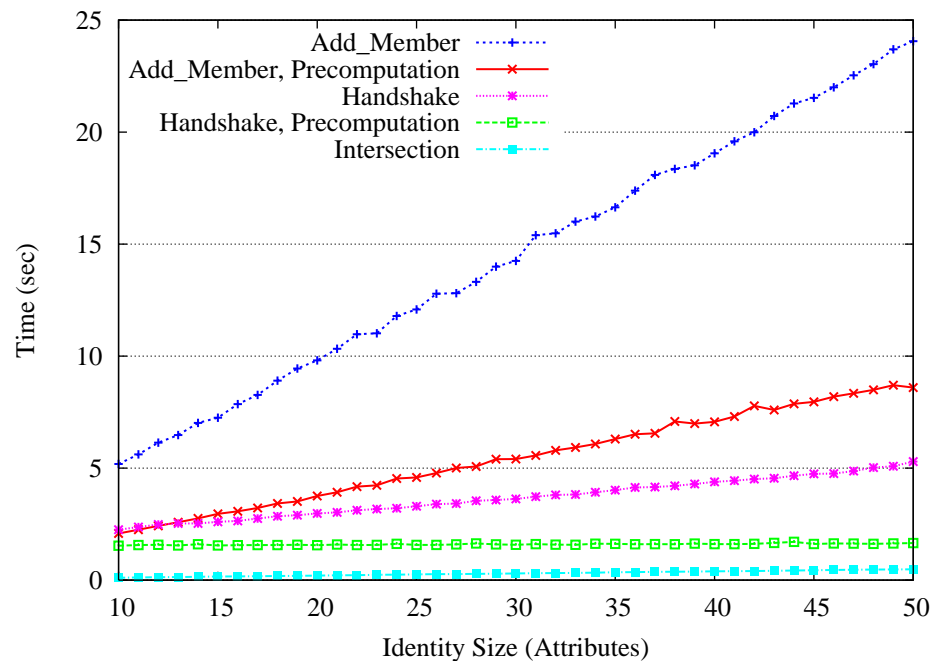
- **Fuzzy IBE is used**
 - identities consist of n attributes
 - a message encrypted under key ID can be decrypted by a user with ID' as long as ID and ID' have at least d attributes in common
- **To satisfy the requirements:**
 - we **change fuzzy IBE to achieve privacy**
 - we **use privacy-preserving set intersection protocol to compute $ID \cap ID'$**
- **The protocol is similar to the dynamic case**

Implementation

- The original secret handshake scheme was implemented in SSL/TLS
- We integrate handshakes at the IP level by extending IPsec key exchange functionality
- IPsec uses Internet Key Exchange protocol (IKE)
 - we replaced authenticated Diffie-Hellman key exchange in IKEv1 with our protocol
- Cryptographic library used: MIRACL
- Results
 - the average time for handshakes with dynamic matching is 0.78s
 - the original Diffie-Hellman key exchange takes 0.5s

Performance of Fuzzy Secret Handshakes

- **Our prototype implementation**
 - **MIRACL** was used for large number arithmetics
 - **communication latency wasn't counted**



fixed threshold $d = 10$