Secret Handshakeにおける管理者に対する新たな匿名性の提案

川合 豊† 國廣 昇†

† 東京大学大学院 新領域創成科学研究科
277-8561 千葉県柏市柏の葉 5-1-5

kawai@it.k.u-tokyo.ac.jp, kunihiro@k.u-tokyo.ac.jp

あらまし Secret Handshake(SH) とは参加者がお互いが同じグループに所属しているかを自らのグループを明かすことなく検証可能な匿名認証方式である．SH の匿名性では，実行者のグループの匿名性と，実行者自身の匿名性，の 2 種類が考えられる．これらの匿名性はグループ管理者に対しても考えられなければいけいない状況が存在し，SH の使用用途によって必要な匿名性は異なる．しかし SH を実社会で用いる場合，既存研究で考えられている管理者に対する匿名性が強すぎる場合が数多く存在する．そこで本稿では，管理者に対する匿名性を厳密に分類し，より多くの場面で SH を用いるために，既存研究では達成されていない「実行者は特定できないが，自らのグループのメンバかを判断可能」な方式を提案する．

New Anonymity against Group Authority for Secret Handshake

Yutaka Kawai† Noboru Kunihiron†

†Graduate School of Frontier Sciences, The University of Tokyo
5-1-5 Kashiwanoha, Kashiwa-shi, Chiba 277-8561 Japan

kawai@it.k.u-tokyo.ac.jp, kunihiro@k.u-tokyo.ac.jp

Abstract Secret handshake allows two members of the same group to authenticate each other secretly, in the sense that each party reveals his affiliation (group) information to the other, only if the other party belongs to the same group, secretly from anyone parties including the group authority (GA). In secret handshake system, two types of anonymity are discussed: the anonymity for a member who executed secret handshake, and the anonymity for a group which a member is belonged to. These anonimities should be discussed against a group authority, also the anonymity which should be satisfied is different with an application. In this paper, we propose new anonymity that GA cannot identify the member who executed secret handshake, but GA can distinguish that a member executed secret handshake belongs to own group. In addition, we propose the concrete secret handshake scheme which satisfied above anonymity.

1 はじめに

個人情報保護法が施行されるなど，匿名性やプライバシーへの関心がますます高まっている昨今，それらを守る様々な技術が研究されている．その技術の一つとして Secret Handshake(SH) がある．Secret Handshake とは，Balfanz ら [1] によって提案された方式で，2 人の参加者が同じグループに属しているときのみ accept を出力し，同じグループに属していることを認証しあうことができ，それ以外の場合は reject を出力して，かつ参加者の属しているグ

ループに関する情報が一切漏れないような認証方式である．所属グループ情報が必要最低限しか漏れないことより，Secret Handshake は部外者にグループを明らかにすることなく，グループに所属している者同士が互いを認証しようとする状況で有用である．

Secret Handshake において最も基本的な匿名性は,「実行者が所属するグループ」の匿名性である．近年，それに加え実行者自身の匿名性として,「2 つの Handshake プロトコルの通信履歴を見ることで，同一人物が実行したことがわからない」という安全性である Unlinkability が提案されている．文献 [1] や [2] などの方式では，実行者がグループに所属する際に割り当てられた ID を Handshake プロトコルにおいて相手にそのまま送信するため，Unlinkability が満たされていなかった．それに対し，文献 [3] などでは Unlinkability を満たす方式を提案している．

さらに，文献 [4] では，匿名性を管理者に対しても考えている．具体的には,「たとえグループ管理者であっても，Handshake プロトコルの実行者の一人と協力しなければ，実行者を特定できない」という Co-Traceability と,「たとえグループ管理者であっても Handshake プロトコルの実行者が自分のグループのメンバであるかを判別できない」という Strong Detector Resistance を提案しこれらを満たす方式を提案している．

しかし，Handshake プロトコルの使用用途によってはこれらの匿名性が強すぎる場面も存在する．例えば，Secret Handshake をインターネット上のコミュニティサービスに使用する場合を考える．メンバは各コミュニティに参加し，他のユーザが自分と同じ興味を持っているかを Secret Handshake を用いて判断する．この場合のグループ管理者はコミュニティの管理者となる．この場合，コミュニティメンバの個人 ID は管理者に対して秘匿されるべきであるが，メンバ同士がどの程度やり取りを行っているかまで秘匿する必要はなく，逆に管理者の能力を制限しすぎであるといえる．

そこで本稿では，管理者に対する匿名性という観点から既存研究を分類し，既存研究では達成されていない匿名性を満たす方式を提案する．

2 Secret Handshake の定義

2.1 SH のエンティティ

SH ではグループ G に対して以下のエンティティを考える．

ユーザ: システムに所属しているすべての人物．

メンバ: ユーザの中でグループ G に所属している人物．以下，A がグループ G のメンバであることを $A \in G$ と表す．

グループ管理者 Group Authority（GA）: グループ G の管理者．ユーザとやり取りすることでグループ G のメンバにすることができる．必要に応じて，GA をメンバをグループに加える能力を持つ発行管理人 Issue Authority (IA) と，Handshake の実行者を特定する追跡管理人 Trace Authority (TA) に分ける．

2.2 SH のアルゴリズムモデル

SH は 6 つのアルゴリズムから構成される[1]．

SH.Setup: 全てのグループに共通のパラメータ $param$ を生成するアルゴリズム．

SH.CreateGroup: GA が実行し，$param$ を入力とし，グループ公開鍵 gpk と GA の秘密鍵 gsk を出力するアルゴリズム．

SH.AddMember: グループ G に所属していないユーザ U と IA との間で行われるプロトコル．入力を $param, gsk, gpk$ とし，IA は U にメンバ証明書 $cert_U$ とメンバ秘密鍵 sk_U を出力して，U をメンバとする．

SH.Handshake: ユーザ U と V との間で行われるプロトコル．U と V が同じグループに所属していれば $accept$ を，そうでなければ $reject$ を出力する．

SH.Trace: TA が実行するアルゴリズム．SH.Handshake の通信履歴と TA の秘密鍵を用いて SH.Handshake の実行者を出力する．

[1] いくつかの SH には，SH.Trace や SH.Co-Trace が存在しない方式もある．

SH.Co-Trace: TA と SH.Handshake の一方の実行者が協力して実行するアルゴリズム．SH.Handshake の通信履歴と TA の秘密鍵，そして一方の実行者がプロトコル中で用いたパラメータを用いてもう一方の実行者を出力する．

2.3 安全性要件

Secret Handshake は，以下の **Correctness, Impersonator Resistance, Detector Resistance** の安全性を満たさなければならない．

Correctness: 同じグループに所属しているメンバ U と V が SH.Handshake を実行したならば，両者とも必ず *accept* を出力する．

Impersonator Resistance (IR)**:** $V \in G$ が，攻撃者 ($\notin G$) と SH.Handshake を実行すると，圧倒的な確率で *reject* を出力する．

Detector Resistance (DR)**:** 攻撃者 ($\notin G$) が $V \in G$ と SH.Handshake を実行した時，攻撃者は V が G に所属しているかを確率 1/2 以上では判別できない．

また近年，より高い匿名性を考慮し，以下のような安全性も提案されている．

Unlinkability (Unlink)**:** 攻撃者 ($\notin G$) が，グループ G のメンバと 2 回 SH.HandShake を実行した際，2 つの通信履歴を関連づけることができない．

Traceability (Trace)**:** SH.Trace がある SH において，TA 単独で Handshake 実行者を特定できる．

Co-Traceability (Co-Trace)**:** SH.Co-Trace がある SH において，TA 単独では SH.Handshake 実行者のどちらも特定できないが，TA は実行者の一方と協力することで必ずもう一方の実行者を特定できる．

Strong Detector Resistance (SDR)**:** 攻撃者 ($\notin G$) が TA の能力を持っていたとしても，$V \in G$ と SH.Handshake を実行したとしても，攻撃者は V が G に所属しているかを確率 1/2 以上では判別できない．

3 グループ管理者に対する匿名性

本章では，GA に対する匿名性を考察する．また，新たな GA に対する匿名性を達成するために「TA が単体で Handshake 実行者が自分のグループであるかを判別する」という新たなアルゴリズム SH.GroupTrace を導入する．

3.1 グループ管理者に対する匿名性の分類

Secret Handshake において，GA に対する匿名性の観点は，「(1):SH.Handshake の実行者自身を GA 単独で特定可能」，「(2):GA と SH.Handshake の実行者の一方と協力することでもう一方の実行者を特定可能」と「(3)：GA は SH.Handshake の実行者が自分のグループであるかを判別可能」の 3 つの観点が存在する．自明に，(1) が可能であれば (2) は可能であり，メンバの ID を登録するような SH においては (1) であれば (3) が可能である．よって，GA に対する匿名性は以下の (A)〜(E) の 5 つに分類される．

(A): GA は，単独であっても実行者の一方と協力しても SH の実行者を特定することはできず，また自分のグループメンバであったかもわからない．

(B): GA は，実行者が自分のグループに所属しているかを判定できる．また，実行者自身を特定することは例え一方の実行者と協力してもできない．

(C): GA だけでは実行者を特定することができない．また，SH.Handshake の一方の実行者と協力することで，もう一方の実行者を特定することができる．また GA は実行者が自分のグループに所属しているか判別できない．

(D): GA は SH.Handshake の一方の実行者と協力することでもう一方の実行者を特定することができる．また GA は単独で実行者が自分のグループかどうかを判別できる．

(E): GA は単独で実行者を特定することができる．また，同時に実行者が自分のグループに所属していいるかを判定できる．

表 1: GA に対する匿名性

	SH.Trace	SH.Co-Trace	SH.GroupTrace	方式
(A)				[5]
(B)			√	提案方式 2
(C)		√		[4]
(D)		√	√	提案方式 1
(E)	√		√	[1],[2]

既存技術では GA 単体では実行者を特定できないが，実行者が自分のグループに所属していることを判別可能な方式は達成されていない (B,D)．そこで本稿では，GA が SH.Handshake の実行者が自分のグループに所属しているかを判別するアルゴリズム SH.GroupTrace を提案し，(B),(D) を満たすような具体的な方式を 4 章で提案する．

安全な SH.GroupTrace を持つ SH は,「GA 以外の第三者では Handshake プロトコルの実行者 $U \in G$ がグループ G に所属してるかが判別不可能」である必要がある．これは DR と同値な安全性である．ただし，DR を満たしているからと言って，SH.GroupTrace のアルゴリズムが必ず存在するわけではない．

（A)〜（E) のそれぞれの方式に含まれるアルゴリズムは表 1 のようになる．

3.2 SH.GroupTrace

本章では，新たに SH.GroupTrace を導入する．

SH.GroupTrace**:** GA が行うアルゴリズム．入力を $param, gsk, gpk$ 及び SH.Handshake の通信履歴とし，実行者が GA のグループに所属していれば 1 をそうでなければ 0 を出力するアルゴリズム．

文献 [1] や [2] では，SH.Handshake において実行者は互いの ID を交換する．また，GA はメンバ個人の ID を所持しているため実行者の ID と比較することで自分のグループに所属しているかを判別できる．しかし，これらの方式には Unlinkability がない．本稿では Unlinkability を達成しつつ SH.GroupTrace を実現できるような方式を提案する．

4 提案方式

本稿では 2 つの方式を提案する（表 1)．提案方式 1 では，TA は単体で Handshake プロトコルの実行者が自分のグループに所属しているかは判別可能だが，実行者を特定するためには一方の実行者の協力が必要な方式である．それに対して提案方式 2 は，TA が実行者を特定することは一方の実行者の協力があっても不可能であるが，単体で実行者が自分のグループに所属しているかの判別は可能な方式となっている．

4.1 準備

定義 1 (Bilinear map) $(\mathbb{G}_1, \mathbb{G}_2)$ を以下の条件を満たす群とする．

- $\mathbb{G}_1$,$\mathbb{G}_2$ は位数 p の乗法的巡回群とし，それぞれの原始元を G_1, G_2 とする．
- ψ を $\psi(G_2) = G_1$ を満たすような $\mathbb{G}_2$ から $\mathbb{G}_1$ への効率的な写像とする．
- e を $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ の双線形写像とする．

本稿ではセキュリティパラメータ k に対して上記のような $(p, \mathbb{G}_1, \mathbb{G}_2, G_1, G_2, \psi, e)$ を出力するアルゴリズム Set を考える．

定義 2 (Descrete Logarithm (DL) 仮定) $P \leftarrow_R \mathbb{G}_1$ と $x \leftarrow \mathbb{Z}_p$ において，xP が与えられた際，x を求める問題を DL 問題という．DL 問題が多項式時間で解くことが困難であることを DL 仮定という．

定義 3 (Decisional Linear Diffie-Hellman) (DLDH) 仮定 [7] $/ U, V, H, Q \leftarrow_R \mathbb{G}_1$ と $a, b \leftarrow \mathbb{Z}_p$ において，U, V, H, aU, bV, Q が与えられた際，$Q = (a+b)H$ かを判定する問題を $DLDH$ 問題という．$DLDH$ 問題を多項式時間で解くことが困難であることを $DLDH$ 仮定という．

GSMR$(param, gpk, sk, cert, m) \rightarrow \sigma$:

1. ユーザ U は $r \leftarrow_R \mathbb{Z}_p{}^*$ を選び $F = \mathcal{G}(gpk, r)$ を計算する．ユーザ V に F を送り，$\tilde{x}_V F$ と $\tilde{x}_V K$ を受け取る．
2. U は $(\alpha, \beta, \gamma, \delta) \leftarrow_R \mathbb{Z}_p^*$ を選び，
   $\hat{F} = \psi(F), \hat{K} = \psi(K)$
   $R_1 = \alpha T, R_2 = \beta S, R_3 = (\alpha + \beta)H + vK, R_4 = \gamma\hat{F} + x_U(\tilde{x_v}\hat{F}), R_5 = \gamma\hat{K}, R_6 = A_U + \delta H$ を計算する．
3. U は $(r_x, r_y, r_\alpha, r_\beta, r_\gamma, r_v) \leftarrow_R (\mathbb{Z}_p^*)^6$ を選び，以下の計算を行う．
   $R'_1 = r_\alpha T, R'_2 = r_\beta S, R'_3 = (r_\alpha + r_\beta)H + r_v K, R'_4 = r_\gamma\hat{F} + r_x(\tilde{x_V}\hat{F}), R'_5 = r_\gamma\hat{K}$,
   $R'_6 = e(R_6, G_2)^{r_y} e(H, W)^{-r_\gamma} e(H, G_2)^{r_x}$
4. U は
   $c' = \mathcal{H}_1(param, gpk, R_1, R_2, R_3, R_4, R_5, R_6, R'_1, R'_2, R'_3, R'_4, R'_5, R'_6), c = c' \oplus m$,
   $s_x := r_x + cx_U, s_y := r_y + cy_U$,
   $s_v := r_v + cv, s_\alpha := r_\alpha + c\alpha$,
   $s_\beta := r_\beta + c\beta, s_\gamma := r_\gamma + c\gamma$ を生成し，
   $\sigma_U = (r, R_1, R_2, R_3, R_4, R_5, s_x, s_y, s_\alpha, s_\beta, s_\gamma, c)$ を出力する．

MR:$(param, gpk, \sigma) \rightarrow m$

1. V は $param, gpk$ と U の GSMRσ を受け取り，以下の計算を行う．
   $R'_1 = s_\alpha T - cR_1, R'_2 = s_\beta S - cR_2$,
   $R'_3 = (s_\alpha + s_\beta)H + s_v K - cR_3$,
   $R'_4 = s_\gamma F + s_x(\tilde{x_V}\hat{F}) - cR_4, R'_5 = s_\gamma\hat{K} - cR_5$,
   $R'_6 = e(R_6, s_y G_2 + cW)e(H, W)^{-s_\gamma} e(H, G_2)^{s_x} e(G_1, G_2)^{-c}$
2. V は
   $c' = \mathcal{H}_1(param, gpk, R_1, R_2, R_3, R_4, R_5, R_6, R'_1, R'_2, R'_3, R'_4, R'_5, R'_6), m = c \oplus c'$
   を計算し m を出力する．

図 1: 提案方式 1 の文書復元型グループ署名の署名作成（GSMR) と文書復元（MR) アルゴリズム

4.2 提案方式 1

提案方式 1 は，TA が単体で Handshake プロトコルの実行者が自分のグループに所属しているかは判別可能だが，実行者を特定するためには一方の実行者の協力が必要な方式である．提案方式は [6, 4] と同様に文書復元型グループ署名 (GSMR) を用いて構成する．本稿では [8, 9] のグループ署名方式をもとに構成する．提案方式 1 で用いる GSMR を図 1 に，提案方式を図 2 に記述した．提案方式 1 は表 1 の (D) の匿名性を達成できている．

4.3 提案方式 2

提案方式 2 は，提案方式 1 を改良することで達成される．具体的には，提案方式 1 において SH.Co-Trace のために必要なパラメータである，$F, \mathcal{G}, r, \tilde{x}_V, B_U, R_4, R_5, R'_4, R'_5$ を無くしたものが提案方式 2 となる．SH.GroupTrace は提案方式 1 と同様の方法で可能である．書面の提案方式 2 の明記は省略する．提案方式 2 は表 1 の (B) の匿名性を達成できている．

4.4 提案方式の安全性

提案方式は以下のような定理を満たす．

定理 1 提案方式 *1* はランダムオラクルモデルと *DL* 仮定，*DLDH* 仮定において，*Impersonator Resistance*，*Detector Resistance*，*Unlinkability*，*Co-Traceability* を満たす．

定理 2 提案方式 *2* はランダムオラクルモデルと *DL* 仮定，*DLDH* 仮定において，*Impersonator Resistance*，*Detector Resistance*，*Unlinkability* を満たす．

上記の定理は，文献 [8] のグループ署名が DL 仮定において偽造不可能性を満たすこと，DLDH 仮定において追跡可能性と匿名性を満たすことから証明される．

参考文献

[1] D. Balfanz, G. Durfee, N. Shankar, D.K. Smetters, J. Staddon, and H. C. Wong. Secret handshakes from pairing-based key agreements. In *IEEE Symposium on Security and Privacy, 2003*, pp. 180–196. IEEE Computer Society, 2003.

SH.Setup: セキュリティパラメータ k を入力とし，$(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, G_1, G_2, G, \psi, e) \leftarrow \mathsf{Set}(k)$ を実行する．また別に DDH が困難な群 $\mathbb{G}$ を選ぶ．ハッシュ関数 $\mathcal{H}_1 : \{0,1\}^* \to \mathbb{Z}_p$, $\mathcal{H}_2 : \{0,1\}^* \to \{0,1\}^k$, $\mathcal{G} : \{0,1\}^* \to \mathbb{G}_1$ を選び出力する．

SH.CreateGruop: 発行管理者 IA は $v, w \leftarrow_R \mathbb{Z}_p$ と $H, K \leftarrow_R \mathbb{G}_1, G' \leftarrow_R \mathbb{G}$ を選び，$W = wG_2$ を計算する．ここで $isk = (v, w), ipk = (H, K, W, G')$ とする．次に，追跡管理者 TA は $(t, s) \leftarrow_R (\mathbb{Z}_p)^2$ を選び $H = sS = tT$ を満たす T, S を生成する．ここで $tsk = (t, s), tpk = (T, S)$ とする．

SH.AddMember: IA とユーザ間で以下のプロトコルを実行する．

1. ユーザ U は $(x', z) \leftarrow_R (\mathbb{Z}_p)^2$ を選び $H' = x'H + zG_1$ を IA に送る．
2. H' を受け取った IA は $(x'', z') \leftarrow_R (\mathbb{Z}_p)^2$ を選び U へ送る．
3. U は $x_U = x'x'' + z', H_U = x_U H$ および $B_U = x_U \psi(K)$ を計算し IA に B_U, H_U と以下の式を $(x_U, x''z)$ が満たすという知識の証明を非対話型ゼロ知識証明を用いて送る．
$$H_U = x_U H$$
$$x''H' + z'H - H_U = x''zG_1$$
また，$t_U \leftarrow_R \mathbb{Z}_p$ を選び，$t_U G'$ を IA に送る．
4. IA は (U, B_U) をメンバリストに加える．IA は $y_U \leftarrow_R \mathbb{Z}_p$ を選び $A_U = \frac{1}{w+y_U}(G_1 - H_U)$ とし (A_U, y_U) を U へ送る．また，IA は $r_U \leftarrow Z_p$ を選び，$(c_{U1}, c_{U2}) = (r_U G', vr_U(t_U G'))$ を U に送る．
5. U は $v = c_{U2}/t_u c_{U1}$ を計算する．

SH.Handshake: メンバ U と V のメンバ証明書を $cert_U = (A_U, y_U)$, $cert_V = (A_V, y_V)$, 秘密鍵を $sk_U = x_U$, $sk_V = x_V$ とする．

1. U は $r_U \leftarrow_R \mathbb{Z}_p^*$, $m_U := r_U G_1$ を選び，$\sigma_U \leftarrow \mathsf{GSMR}(param, gpk_U, sk_U, cert_U, m_U)$ を作成し V へ送る．
2. V は $r_V \leftarrow_R \mathbb{Z}_p^*$, $m_V := r_V G_1$ を選び，$\sigma_V \leftarrow \mathsf{GSMR}(param, gpk_V, sk_V cert_V, m_V)$ を作成し U へ送る．また V は $m'_U \leftarrow \mathrm{MR}(param, gpk_V, \sigma_U)$ を実行する．
3. U は $m'_V \leftarrow \mathrm{MR}(param, gpk_U, \sigma_V)$ を実行し，$resp_U := \mathcal{H}_2(r_U m'_V, m_U)$ を作成し V へ送る．
4. V は $resp_V := \mathcal{H}_2(r_V m'_U, m_V)$.
もし $resp_U = \mathcal{H}_2(r_V m'_U, m'_U)$ であれば，V は *accept* を出力し，$resp_V$ を U に送る．そうでなければ，V は *reject* を出力する．
5. U executes:
もし $resp_V = \mathcal{H}_2(r_U m'_V, m'_V)$ であれば，U は *accept* を出力する．そうでなければ，U は *reject* を出力する．

SH.GroupTrace:
TA は SH.Handshake プロトコルにおける σ から $R_3 - (tR_1 + sR_2)$ を計算し，vK と等しいかを判別する，もし等しければ 1 を，そうでなければ 0 を出力する．
SH.Co-Trace: U と V が SH.Handshake を実行し，TA と V が協力することで U を特定するとする．

1. TA は V から $sk_V, \tilde{x}_V$ を受け取り，$cert_U = (A_U, y_U), sk_U = x_U$ を用いて，もし $e(A, W)e(H, G_2)^x e(A, G_2)^y = e(G_1, G_2)$ であれば以下を実行する．そうでなければ $\perp$ を出力する．
2. TA は U の出力した σ とメンバリスト (i, B_i) に対して $e(R_4, K) = e(B_i \tilde{x}_V \psi(K), F)$ が成り立つかを調べる．成り立ったならば $i = U$ を出力する．

図 2: 提案方式 1

[2] C. Castelluccia, S. Jarecki, and G. Tsudik. Secret handshakes from ca-oblivious encryption. In *Proc. ASIACRYPT2004*, Vol. 3329 of LNCS, pp. 293–307. Springer, 2004.

[3] S. Xu and M. Yung. k-anonymous secret handshakes with reusable credentials. In *Proceedings of the 11th ACM conference on Computer and communications security*, pp. 158–167. ACM, 2004.

[4] Y.Kawai, K.Yoneyama, and K.Ohta. Secret handshake: Strong anonymity definition and construction. In *Proc. ISPEC 2009*, Vol. 5421 of LNCS, pp. 219–229. Springer, 2009.

[5] G. Ateniese, M. Blanton, and J. Kirsch. Secret handshakes with dynamic and fuzzy matching. In *Network and Distributed System Security Symposium*, 2007.

[6] 山下武志, 多田充. 新しい secret handshake の構成法. In *CSS2006*, 2006.

[7] D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In *Proc. CRYPTO 2004*, Vol. 3152 of LNCS, pp. 41–55. Springer, 2004.

[8] J. Furukawa and H. Imai. An efficient group signature scheme from bilinear maps. In *Proc. ACISP 2005*, Vol. 3574 of LNCS, pp. 455–467. Springer, 2005.

[9] T. Nakanishi and N. Funabiki. A short verifier-local revocation group signature scheme with backward unlinkability. In *Proc. IWSEC2006*, Vol. 4226 of LNCS, pp. 17–32. Springer, 2006.