Within-Group Signatures

Anson Hu
MIT
ansonhu@mit.edu

Leo Wang
MIT
shihyu@mit.edu

Sidharth Menon
Harvard
skmenon@college.harvard.edu

1 INTRODUCTION

In this paper, we explore the idea of within-group signatures. A within-group signature allows any member of a well-defined group of people to (anonmyously) sign a message, but the resulting signature cannot be distinguished from forgery by those outside the group. In other words, this combines (roughly) two ideas from existing literature. First, the notion of ring signatures allows any member of a group to produce a signature on behalf of that group [6]. Second, the notion of designated parties for signature verification allows only a pre-fixed set of verifiers to actually perform verification on a signature to prove it came from a group [1].

As a combination of these two ideas, within-group signatures would be useful for a secret group (e.g. the Illuminati) to send authenticated messages to each other publicly without revealing their identity. Furthermore, a within-group signature formulation would allow for this group to retain plausible deniability about any message's authenticity in a public forum. This idea has deep real-world motivations. For example, a whistle-blower in a political organization or company could wish to expose a scandal, but not wish to inform the general public yet. On the other hand, the whistle-blower may still wish to remain anonymous to protect their personal interests. In some cases, privately dealing with issues is preferable to dealing with the court of public opinion, and a within-group signature formulation provides the framework for this.

In our project, we will first define the within-group signature and create formal notions of security for it. Then, we will provide an interactive and non-interactive scheme for the within-group signatures.

2 RELATED WORK

From our research, there was no exploration of the idea of within-group signatures in existing literature. There were, however, a few sources of related work that we did analyze as a theoretical basis and inspiration for our formulation:

(1) Ring Signatures: Rivest, Shamir, and Kalai introduce the idea of *Ring Signatures*[6]. Essentially, any one in a given group can sign on behalf of that group. To formalize this, consider F parties in a group defined by public-key, secret-key pairs: $(P_1, S_1), \dots (P_F, S_F)$. Formally, a signature σ for a given message m generated by party i is given by $\mathsf{Sign}(m, S_i, (P_1, \dots, P_F))$. Verification of σ as authentic for m can be computed by *anyone* by $\mathsf{Verify}(\sigma, m, (P_1, \dots, P_F))$. The key virtues of this approach are twofold. First, there is no "manager" who signs on behalf of the group or has greater power than the other members. Second, the identity of the signer is concealed to (computationally bound) adversaries, which provides anonymity.

(2) Designated Verifier Signatures: A paper written by Jakobsson, Sako & Impagliazzo establishes the idea of designated verifier proofs.[1] In this formulation, only a predetermined, designated verifier can actually verify a given cryptographic signature (and the signature cannot be verified by other parties). This allows for *authenatication* but also *private* communication.

(3) Group-Oriented Strong Designated Verifier Signature Scheme: Lin, Wu, Ting, and Lee combine 1. & 2. with a designated verifier scheme for ring signatures.[3] This is similiar in principle for to our project, with two key differences: there is no within group anonymity, which means that a verifier knows which group member produced a given signature, and the signing scheme is many-to-one; a group of signers all sign the message for one verifier. One advantage of this paper to our (current) approach, however, is that the signatures are constant size – this is something that we will look to improve in the future.

3 SECURITY DEFINITIONS

3.1 Definitions

To formalize the notion of a *within-group signature*, we defined the security goals below. We assume an a group of people are represented by a well-defined public/private key-pair. Note the delineation between "group members" (those with access to a group member's private key) and an "outsider" (those without access to a member's private key).

(1) **Group Signatures**: One member should be able to sign on behalf of the group
(2) **Within-Group Anonymity**: Those inside the group should not be able to tell (beyond random chance) which group member created the signature
(3) **Outside-Group Obfuscation**: Those outside the group (so-called "outsiders" without access to a group member's private key) cannot determine if the signature

came from a group member or an adversary beyond random chance.

Note that goals 1 and 2 are about the capabilities of group members *with* a group member's private key. These actors can efficiently verify the signature came from a group member, but cannot know beyond random chance from *which* group member. Goal 3 is about the limitations of those *without* a group member's private key – they cannot verify whether a signature actually came from the group members or an adversary beyond random chance.

3.2 Formalization

Note that the notion of ring signatures as presented in literature provides formal notions for security goals 1 and 2 (group signatures and within-group anonymity, respectively) [6]. We seek to provide for security goal 3 in ring signature schemes as well, and provide obfuscation of valid signatures to outsiders or adversaries. Because we could not find analogues in existing literature, we define a game to formalize the notion of Outside-Group Obfuscation.

Drawing from game formulations present in the existing literature, communication in these games proceed in synchronized rounds and messages are always received during their given rounds. Both challenger and adversary have access to a public broadcast channel which can be used to send and receive messages [2, 4, 5].

3.2.1 Outside-Group Obfuscation. A ring signature scheme Σ provides Outside-Group Obfuscation if any probabilistic polynomial time adversary $\mathcal{A}$ cannot win the game below (hereafter defined as the Outside-Group Obfuscation Game, or "OGO" Game) with more than probability $1/2$ (diagrammed in Figure 1).

(1) Initialization: The game begins with an execution of $\mathrm{Gen}(pp, n, \lambda)$ to create the players $P_1, \ldots, P_n$ and corresponding verification keys $\mathbf{pk} = (pk_1, \ldots, pk_n)$ and signing keys $\mathbf{sk} = (sk_1, \ldots, sk_n)$ according to some public parameters pp, number of group members n, and security parameter λ.

(2) Query Phase: On polynomially many occasions, the adversary $\mathcal{A}$ can issue the query

$$\mathrm{GenerateSig}(m, i, \{pk_1, \ldots, pk_n\})$$

asking for a signature on behalf of group member P_i (where $1 \leq i \leq n$) on an arbitrary chosen message m. The challenger responds by computing $\sigma = \mathsf{Sign}(m, sk_i, \mathbf{pk})$ and returning σ to the adversary.

(3) Challenge: The adversary sends over a previously-unseen message m (to avoid trivially repeating a previous signature). The challenger $\mathcal{CH}$ samples $b \longleftarrow \{0, 1\}$ uniformly at random. If $b = 0$, then the challenger samples $i \longleftarrow \{1, \ldots, n\}$ uniformly at random and outputs

$$\sigma_b = \mathsf{Sign}(m, sk_i, \{pk_1, \ldots, pk_n\})$$

Otherwise, the challenger outputs

$$\sigma_b = \mathsf{Forge}(m, \{pk_1, \ldots, pk_n\})$$

for some algorithm Forge which does not take any sk as input.

(4) The adversary outputs $b' \in \{0, 1\}$. The adversary wins the game if $b' = b$.

In other words, let $\mathcal{A}(\sigma_b) = b'$. If there exists an algorithm Forge such that a poly-time adversary $\mathcal{A}$ wins the OGO game for a scheme Σ with probability :

$$\Pr[b \longleftarrow \{0, 1\}; \mathcal{A}(\sigma_b) = b] \leq \frac{1}{2}$$

then Σ provides Outside-Group Obfuscation.

In our formulation, a ring signature scheme Σ which satisfies the OGO Game is a *within-group* signature scheme. Note that we will only explore the game above for the final, non-interactive solution below, since it has superior security to the interactive version.

4 INTERACTIVE SOLUTION

In this section, we provide a solution that consists of interactions between group members. Our idea is to create a secret key s that only the group members can know, and sign the message with this secret key. Specifically, we utilize multi-party Diffie-Hellman scheme to achieve this. Assume that n people $p_1, p_2, \cdots p_n$ have secret key $k_1, k_2, \cdots, k_n$, respectively. In the elliptic curve version of multi-party Diffie-Hellman scheme, they first agree on a base point G and do the following.

(1) Every person holds an elliptic curve point that is initially the base point G.

(2) Repeat the following n times: Each person multiply the point they have by their secret key, and pass that new point to the next person (p_1 to p_2, p_2 to p_3 and so on).

In the end, everyone has the value of $(k_1 k_2 \cdots k_n)G$, which can be used to deterministically extract the secret value s.

With introduction of the multi-party Diffie-Hellman scheme, we introduce our interactive solution to our main problem.

(1) Perform the multi-party Diffie-Hellman scheme so that every member has a secret key s.

(2) The signer publishes the value of $\mathsf{HMAC}(m, s)$ for message m.

The scheme's correctness is clear by the fact that only the group members know the value of s, and therefore can verify $\mathsf{HMAC}(m, s)$. However, there are some details that are needed for analysis.

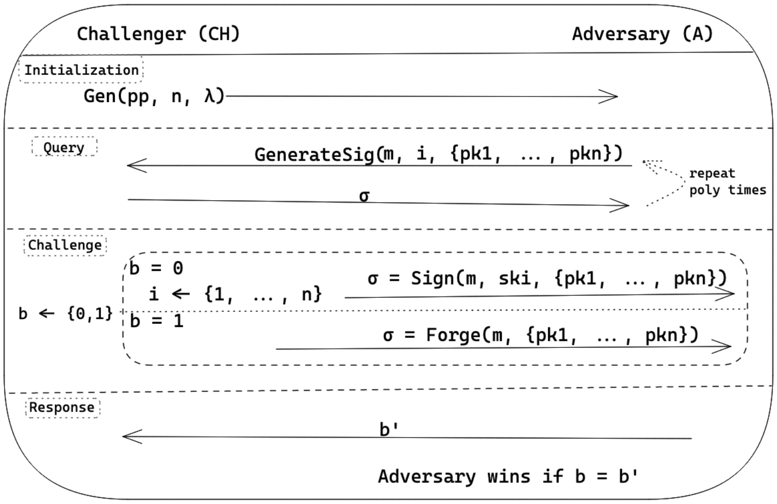

Figure 1: Visualization of Outside Group Obfuscation ("OGO") Game

First, a malicious group member can tell an outsider the value of s, allowing the outsider to sign and verify in the future. This means that the above scheme should be one-time, and the Diffie-Hellman exchange has to be done before each signature. The scheme can still work by adding a method for the signer to anonymously request a multi-party Diffie-Hellman interactions, but it creates more room for the developer to make mistakes such as reusing secret value s.

Second, it is well known that Diffie-Hellman key exchange is subject to man-in-the-middle attack, and multi-party version is not an exception. An outsider who has control over the network can change the value being passed, and paralyze the key exchange scheme, or even manipulate the values such that every member ends up with different points that are all known to the attacker.

The above analysis shows that while the scheme can work, it is not useful in practice because all interactions can potentially be tampered. This motivates us to create a non-interactive scheme.

5 NON-INTERACTIVE SOLUTION

The intuition behind the solution is to create a dummy person (represented by a key pair), add that person to the ring, and sign the message with a ring signature. The dummy person's public key must be unknown to outsiders and its private key must be unknown for everyone. This guarantees that outsiders cannot distinguish the dummy person from a random key pair, nor can they forge a signature by using the dummy person's key pair.

The above goal is easily achieved by the Diffie-Hellman key exchange scheme (we will use the elliptic curve version). Let Alice and Bob have key pairs (a, aG) and (b, bG), respectively. After the key exchange, both Alice and Bob know the value of $(ab)G$ but don't know the value of ab. Consider the key pair $(ab, (ab)G)$. It satisfies the goal of "no one knows the private key, and no outsiders can distinguish the public key from a random key pair". This motivates the following signature scheme:

(1) Each group member has an elliptic-curve key pair. Group member i has private key r_i and public key r_iG. Let our signer's key pair be r_{signer} and $r_{signer}G$.
(2) The signer chooses a temporary random key pair (r, rG).
(3) For group member i with key pair (r_i, r_iG), the signer signs the message by creating a ring signature with all public keys and $(rr_i)G$.

$$\sigma_i = \text{ring\_sign}(m, r_{signer}, \{r_1G, r_2G, \cdots, r_nG, (rr_i)G\})$$

(4) The above step is done with each member, and the signer publishes rG and all σ_i.

It is clear that the signer's anonymity holds by the definition of the ring signature, and the verification scheme is naturally derived by the signature scheme. Specifically, for group member i,

(1) Multiply rG (published by the signer) by their private key to get $(rr_i)G$.
(2) Verify the ring signature using all public keys and $(rr_i)G$.

Knowing that the group members are able to verify the signature, we only need to show why an outsider cannot distinguish the signature from a forged signature. The information that an outsider know is the public keys and rG. Assuming DDH is difficult, the outsider cannot distinguish

$(rr_i)G$ from tG for some random t. Therefore, the signature

$$\text{ring\_sign}(m, r_{signer}, \{r_1G, r_2G, \cdots, r_nG, (rr_i)G\})$$

is indistinguishable from

$$\text{ring\_sign}(m, t, \{r_1G, r_2G, \cdots, r_nG, tG\})$$

where t is truly random number. Since the latter can be forged by any person, the outsider cannot distinguish between a forged signature and a true signature. (Note that an insider will be able to detect the forgery by the $(rr_i)G$ check.)

The scheme described above satisfies security goals 1 and 2, as the signer can sign on behalf of the group, and their anonymity within the group is protected by the security properties of ring signatures (and by the security of the Diffie-Helman shared secret, by assumption) [6]. Note that this scheme also satisfies the OGO game as follows:

In the challenge phase, an adversary knows the public keys and rG (along with previous signatures from queries). As stated before, assuming DDH is difficult, an adversary cannot distinguish $\text{ring\_sign}(m, r_i, \{r_1G, r_2G, \cdots, r_nG, (rr_i)G\})$ from $\text{ring\_sign}(m, t, \{r_1G, r_2G, \cdots, r_nG, tG\})$ for random t. This implies that the distribution of

$$\sigma_0 = \text{ring\_sign}(m, r_i, \{r_1G, r_2G, \cdots, r_nG, (rr_i)G\})$$

$$\sigma_1 = \text{Forge} = \text{ring\_sign}(m, t, \{r_1G, r_2G, \cdots, r_nG, tG\})$$

(for randomly sampled i, t) are computationally identical for a polynomially bounded adversary. Thus the adversary has no better strategy than to guess and cannot win the OGO game with more than probability $1/2$. Therefore, the above scheme satisfies the within-group signature requirements.

6 FUTURE WORK

The two aforementioned solutions are probably not optimal. The interactive solution has a short signature size, but it is vulnerable to man-in-the-middle and other attacks. On the other hand, the non-interactive solution is immune to those attacks, but has a signature size of $O(n^2)$. Here, we give some directions of possible future work:

(1) Create a within-group signature scheme that supports adding and removing group members.
(2) Reduce the signature size. Notably, any ring signature scheme which supports a key scheme with a form of secret sharing satisfying the above constraints will be capable of being inserted into this scheme. This can allow for more efficient signature generation and storage.
(3) Create t-of-n within-group signatures. This may seem trivial to do by using multi-party Diffie-Hellman and replacing the ring signatures in our scheme with t-of-n signatures, but this is not correct, since outsiders will still be able to tell $t - 1$ members of the ring signed. This is an area of future work.
(4) Create linkable/traceable ring signatures.

REFERENCES

[1] Sako Jackobsson and Impagliazzo. 1996. Designated Verifier Proofs and Their Applications. https://link.springer.com/chapter/10.1007/3-540-68339-9_13. 1070 (1996), 143–154.

[2] Benoît Libert, Marc Joye, Moti Yung, and Fabrice Mouhartem. [n. d.]. Fully Distributed Non-Interactive Adaptively-Secure Threshold Signature Scheme with Short Shares: Efficiency Considerations and Implementation. ([n. d.]), 15.

[3] Han-Yu Lin, Hong-Ru Wu, Pei-Yih Ting, and Po-Ting Lee. 2019. A Group-Oriented Strong Designated Verifier Signature Scheme with Constant-Size Signatures. In *2019 2nd International Conference on Communication Engineering and Technology (ICCET)*. 6–10. https://doi.org/10.1109/ICCET.2019.8726877

[4] Andrew Morgan and Rafael Pass. 2018. On the Security Loss of Unique Signatures. In *Theory of Cryptography*, Amos Beimel and Stefan Dziembowski (Eds.). Vol. 11239. Springer International Publishing, Cham, 507–536. https://doi.org/10.1007/978-3-030-03807-6_19 Series Title: Lecture Notes in Computer Science.

[5] Alexander Munch-Hansen, Claudio Orlandi, and Sophia Yakoubov. 2021. Stronger Notions and a More Efficient Construction of Threshold Ring Signatures. In *Progress in Cryptology – LATINCRYPT 2021*, Patrick Longa and Carla Ràfols (Eds.). Vol. 12912. Springer International Publishing, Cham, 363–381. https://doi.org/10.1007/978-3-030-88238-9_18 Series Title: Lecture Notes in Computer Science.

[6] Shamir Rivest and Kalai. 2001. How to Leak a Secret. https://link.springer.com/chapter/10.1007/3-540-45682-1_32. *ASIACRYPT: Advances in Crytography* 2248 (2001), 552–565.