Truly Anonymous Paper Submission and Review Scheme

Chun-I Fan, Ming-Te Chen, Yu-Kuang Liang, Long-Sian Chen

Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan.

E-mail: cifan@faculty.nsysu.edu.tw

Abstract. Due to the flush development of academic research, a great deal of research papers have been published in conference proceedings and journals. However, these articles need to be inspected by some professionals in specific fields. It is the most important that the entire process of reviewing must be kept fair. However, the privacy of reviewers is not preserved because that the reviewers must sign their comments on the reviewed papers for some conference proceedings or journals. The leakage of the reviewers' identities will affect the fairness of paper reviewing. In addition, it is also necessary for the authors to show their names to the editors of conference proceedings or journals such that the inspection results may be unfair. Unfortunately, the solutions proposed in the literature cannot cope with the problems on fairness well. Therefore, in order to eliminate the above drawbacks, we formally analyze the paper review procedure to solicit the possible reasons that bring about these unfair results. Furthermore, we will present a generic idea which is independent of the underlying cryptographic components to achieve the fairness property and other key requirements in a paper review system. Finally, the security of the proposed scheme is also formally proved.

Keywords. Anonymous paper submission, Anonymous paper review, Blind signatures, Universal designated-verifier signatures (UDVS), Anonymous channels, Information security, Cryptography

1 Introduction

There are lots of papers in different kinds of research topics to be published in conference proceedings and journals every year. Authors attempt to submit their papers to the conferences and journals whose topics match the contents of their papers. A traditional physical "paper review system" contains three types of participants, i.e., authors, an editor of a conference proceedings or journal, and a group of reviewers and it operates according to the following procedures and assumptions:

1. The editor of a conference proceedings or journal announces a publication schedule and information such as the topics of the conference or the journal, the deadline for paper submission, the date for notification of acceptance, the format of a submitted paper, and so on.

[*] A partial result of this research was presented in the International Conference on Availability, Reliability and Security (ARES), March 16-19, 2009, Fukuoka, Japan.

2. The authors of a paper submit their paper with their names to the conference or journal.

3. When the editor receives the paper, she/he starts to check whether the paper matches the topics or not. After the due date of submission, the paper will be processed into the next stage if it passed the above verification. Otherwise, the authors will be notified that their paper is not matched.

4. A group of reviewers will help the editor to review papers. Nowadays this group is composed of the researchers and professors in the same or similar research fields. The editor allocates some reviewers and invites them to review the paper. But the editor may hide the names of the authors while allocating the paper.

5. The selected reviewers reply the invitation of the editor and receive the paper if they are willing to review it. Therefore, a lot of research people in the same research society inspect their papers one another, which is called peer reviewing.

6. The reviewers send their comments and results of the inspection back to the editor.

7. Finally, the editor collects all comments of the reviewers, makes judgement, and then notifies the authors whether the paper is accepted or not.

Thus, we can divide the paper review procedure into three phases: (1) paper submission, (2) paper allocation and review, and (3) result decision.

There are some drawbacks that we have found in the traditional paper review system as follows: (For simplicity, we assume that there is only one author of a paper.)

1. Incomplete Fairness: According to the steps described above, the editor can know who the author of a paper is. The final result may be influenced by the personal attributes of the author such as the author's institution or name. For example, in the paper allocation phase, the editor may assign a paper which was written by his friends or a famed researcher to the reviewers who review the paper loosely. Thus, the paper may be accepted by the editor more easily.

2. Insufficient Privacy Protection:

   2.1 Assume that a paper was not accepted by the editor. The rejected paper may be re-submitted to another conference or journal by the author. However, the editor has known who the author of the paper is and she/he may reveal it to someone else. Hence, the reviewers of the next conference or journal may have the name of the author of the paper before reviewing it.

   2.2 The editor knows the relationship between the reviewers and their comments on a paper. She/He is able to convince the author that someone has reviewed the paper.

It would be unnecessary to let the editor know everything. If the editor knows information about authors, then she/he may submit their paper to the reviewers that are good friends of the editor and asks them to give positive/negative comments at the editor's will. For example, as shown in [13], they proposed the *Anonymous Reviewing* idea and it can be applied on the paper review system to avoid the editor obtaining the authors' information.

A common known truth is that some researchers, especially those who are at the beginning of their career may disincline to write negative review comments as it could hamper future promotions. Thus, it may cause the situation that the comments of the reviewers are not fair to the authors. In order to solve this problem, we believe that it is required to keep the reviewers anonymous such that the editor does not have obtain any evidence to convince anyone else of the fact of reviewing.

In [4], it shows that the reviewers' recommendations are frequently biased. Hence, we also come up with *Anonymous Submission* which makes it possible for the authors to anonymously submit their papers to the editor. In this manuscript, we will present an anonymous paper submission and review scheme with both anonymity of the authors and the reviewers, respectively.

2 Related Works

Vincent Naessens, Liesje Demuynck, and Bart De Decker presented a fair anonymous submission and review system in [13]. Anonymous credentials were used as basic primitives. They claimed that anonymous credentials allow for anonymous yet accountable transactions between users and organizations. In [13], the authors presented a simplified version of the Idemix anonymous credential system in [5], [10]. The Idemix anonymous credential system uses a *pseudonym* to protect a user's anonymity and the user must generate a zero-knowledge proof to convince the service providers that she/he is the real one.

The scheme of [13] presented a framework about an anonymous paper review system and [13] also showed that anonymous reviewing and anonymous submission can improve the fairness of paper review. However, it allows each attendant to use a pseudonym to keep anonymous in the protocol. It would not be a good idea to achieve the anonymity property since the author of each paper needs to register a pseudonym with an organization. The registration may break the anonymity of the author if the organization is not trusted. In addition, the editor in the protocol may store allocation record of each paper in the paper review phase such that she/he can convince others of the reviewers' identities of the paper. The scheme contains lots of functions such that its structure is relatively complicated and it may be impractical for implementation.

Recently, Esma Aïmeur, Gilles Brassard, Sébastien Gambs, and David Schönfeld presented a privacy-preserving peer review system in [3]. They designed a distributed conference review system based on group signatures, which can preserve the privacy of all participants involved in the peer review process. It needs two trusted servers (group managers) for the authors and the reviewers, respectively, to preserve the privacy of the two parties. Moreover, it introduces a trusted website for handling the peer review process. The privacy of all participants can be protected under the strong assumptions, however, it will also be impractical for real implementation.

3 Preliminaries

3.1 Partially Blind Signatures

Our anonymous paper review scheme adopts the functions of a partially blind signature scheme. In this subsection we will define a generic partially blind signature scheme. In the scenario of issuing a partially blind signature, the signer and a user are assumed to agree on a piece of common information, denoted as *info*. In some applications, *info* may be decided by the signer, while in some other applications it may just be sent from the user to the signer. Here we discuss the first case that *info* is decided by the singer only. Normally, a generic partially blind signature protocol [1], [2], [7], [8], [9] contains four phases: blinding, signing, unblinding, and verifying, which are described below.

1. Blinding: A user blinds a message and sends the blinded message to the signer to request a signature on it.

2. Signing: After receiving the blinded message, the signer signs the blinded message and the common information *info* by using its signing function and sends it back. The signing result is called the partially blind signature since the message is unknown to the signer but the common information *info* is clear to the signer.

3. Unblinding: The user unblinds the partially blind signature and then gets a signature of the signer on the combination of the original message and the common information *info*.

4. Verifying: Finally, the user or others can verify the signature by using a verification formula with the parameters containing the signature, the message, and the common information *info*.

Now we introduce the functions that used in a generic partially blind signature scheme. Let M be the underlying set of messages, R be a finite set of random strings, W be a finite set of strings with the predefined format which is negotiated by the signer and all users in advance. There are five elements (B, S, H, U, V) in a generic partially blind signature scheme. They are defined as follows:

1. H: $M \to M$ is a public one-way hash function.

2. S: $M \times W \to M^k$ is the signing function which is kept secret by the signer where k is a positive integer. Given a message $m \in M$ and a common information $w \in W$, it is computationally infeasible to form $S(H(m), w)$ or modify m and w embedded in $S(H(m), w)$ without signing function S, where $S(H(m), w)$ is called the signer's signature on message m and the common information w.

3. V: $M^k \times M \times W \to \{\mathsf{True}, \mathsf{False}\}$ is the public verification formula. $V(t, H(m), w) = \mathsf{True}$ if and only if t is the signature of the signer on m with the common information w. Therefore, $V(S(H(m), w), H(m), w)$ is always true for each $m \in M$ and $w \in W$.

4. B: $M \times R \to M$ is the blinding function. Select a random string $r \in R$, which is prepared to be a blinding factor and kept secret by some user. The user takes r to form the blinded message $B(H(m), r)$. None can decide $H(m)$ from the blinded message without the blinding factor r.

5. U: $M^k \times R \to M^k$ is the unblinding function. For each $m \in M$, $r \in R$, and $w \in W$, U can be used to shuck the blinding factor to get the signature on the clear message m and w, i.e., $U(S(B(H(m), r), w), r) = S(H(m), w)$. It is also impossible to decide $S(H(m), w)$ from $S(B(H(m), r), w)$ without r.

3.2 Universal Designated-Verifier Signatures (UDVS)

In addition to a generic partially blind signature scheme, we use another technique called Universal Designated-Verifier Signatures (UDVS) [16], [17]. A UDVS scheme is a digital signature scheme with an additional functionality which allows any holder of a signature to assign the signature to any desired *designated-verifier* such that the designated-verifier can verify that the message was signed actually by the signer but the verifier cannot use this signature to convince anyone else of this fact. This is because that the verifier's secret key allows her/him to forge the same signature without the signer's cooperation. Hence, UDVS protects the privacy of signature holders against signature dissemination of verifiers. A UDVS scheme is made up by eight algorithms and all of these algorithms may be randomized. The functions of a UDVS scheme and the security notions are defined as follows.

1. Common Parameter Generation GC: On inputting a security parameter k, it outputs a string cp that consists of common scheme parameters.

2. Signer Key Generation GKS: On inputting a common parameter string cp, it outputs a key pair (SK_{R_i}, PK_{R_i}) for a signer R_i, where $i = 1, ..., n$.

3. Verifier Key Generation GKV: On inputting a common parameter string cp, it outputs a key pair (SK_{V_j}, PK_{V_j}) for a verifier V_j, where $j = 1, ..., n$.

4. Signing S: On inputting a secret key SK_{R_i} and a message m, it outputs a publicly-verifiable (PV) signature σ of the signer R_i.

5. Public Verification V: On inputting a signer's public key PK_{R_i} and a string pair (m, σ) consisting of the message and corresponding signature, it outputs a verification result $d \in \{\mathsf{True}, \mathsf{False}\}$.

6. Designation DV: On inputting a signer's public key PK_{R_i}, a verifier's public key PK_{V_j}, and a message/PV-signature pair (m, σ), it outputs a designated-verifier (DV for short) signature $\hat{\sigma}$.

7. Designated Verification VDV: On inputting a signer's public key PK_{R_i}, a verifier's secret key SK_{V_j}, and a message/DV-signature pair $(m, \hat{\sigma})$, it outputs a verification result $d \in \{\mathsf{True}, \mathsf{False}\}$.

8. Verifier Key-Registration P_{KR}: A Verifier (VER) wishes to register a verifier's public key with a Key Registration Authority (KRA). On inputting a common string cp, VER and KRA send messages alternately to each other. Then KRA outputs a $(PK_{V_j}, Auth)$ pair where PK_{V_j} is the verifier's public key and $Auth$ is an authorization decision of the key-registration authority.

There are two major properties in UDVS, where one is unforgeability and the other is non-transferability privacy.

1. Unforgebility: A UDVS scheme consists of two types of unforgeability properties. The first one is PV-Unforgeability where the definition of the property is the same as the typical unforgeability notion under CMA (Chosen-Message Attack) for the standard signature scheme which consists of GC, GKS, S and V. The second one is DV-Unforgeability which makes it difficult for an attacker to forge a DV-signature σ' on a new message m' that can pass the VDV-verification with a given designated-verifier's public key PK_{V_j}.

2. Non-Transferability Privacy: The goal of this property for a UDVS scheme is to protect the actual signer's privacy. It prevents a designated-verifier from using the DV-signature on a message m to convince someone that the signature on message m is signed by the actual signer.

3.3 The Requirements

We may encounter some problems when designing an anonymous paper submission and review system in the following.

1. When the authors and the reviewers in the same group are anonymous to the editor, an author may be a reviewer of her/his own paper. It will be unfair in the reviewing process.
2. An author may ask a reviewer to give positive comments on her/his paper.
3. When an author submits her/his paper anonymously, an attacker may impersonate her/him to be the author.
4. The editor may reveal the identities of reviewers to the authors.
5. Reviewer's comments may be forged by an attacker. The attacker can modify the comments about a paper arbitrarily if she/he can forge a comment signed by a reviewer.

In order to construct a secure anonymous paper submission and review system, we collect the following security requirements.

1. Anonymity: The anonymity property is quite important in the paper submission and review system. It is strongly related to the fairness property and can be divided into several parts as follows:
   - Author→Editor : The author needs to blind her/his name when she/he submits her/his paper to the editor. The editor does not know who the author of the paper is such that she/he will allocate it to reviewers more fairly.
   - Author→Reviewer: The author also should cover her/his name in her/his paper. If her/his identity was known by the reviewers, the reviewers' comments may be influenced.
   - Reviewer→Author: While a reviewer's identity is not disclosed, she/he can inspect the paper more fairly. She/He will not be asked to give positive or negative comments by coercers, bribers, or the authors.
2. Uniqueness: None can claim that she/he is the author of a paper except that she/he is the actual one.
3. Comment Unforgeability: The comments can only be written by the reviewers, i.e., the comments cannot be forged.
4. Honesty: When a user submits her/his paper, she/he cannot be a reviewer of her/his own paper.

4 The Proposed Anonymous Paper Submission and Review Scheme

We make use of the generic partially blind signature scheme, an anonymous secure channel [6], [12], and a universal designated-verifier signature scheme [16], [17] to design the anonymous paper submission and review protocol. There are four parties in the protocol: a time-stamp server, authors, an editor, and a group of reviewers where the authors get times-tampped signatures from the time-stamp server and submit their papers to the editor and the reviewers to examine the quality of the papers. The editor decides whether the paper is accepted according to the responses and comments of the reviewers.

In order to make the protocol more simple, we assume that there is only one author for each paper. Our protocol is also suitable for the situation that there are several authors of a paper. In the following, we give the notation's definition and the description of our protocol.

4.1 Notations

- m: the paper that an author attempts to submit, where it contains no identification information of the author
- ID_i: the identity of author i
- M: the message space
- PK_{TS}: the public key of the timestamp server
- SK_{TS}: the secret key of the timestamp server
- $H(\cdot)$: a one-way hash function
- $ST_{SK}(\cdot)$: the signing function of the timestamp server based on a generic partially blind signature with the key SK
- $VT_{PK}(\cdot)$: the verifying function of the timestamp server based on the generic partially blind signature with the key PK.
- $B_T(\cdot)$: the blinding function of the timestamp server
- $U_T(\cdot)$: the unblinding function of the timestamp server
- PK_E: the public key of the editor

- SK_E: the private key of the editor
- $E_{PK}(\cdot)$: an encrypting function with the key PK
- $D_{SK}(\cdot)$: a decrypting function with the key SK
- $S_{SK}(\cdot)$: a signing function with the key SK
- $V_{PK}(\cdot)$: a signature verifying function with the key PK
- PK_{R_i}: the public key of the i-th reviewer
- SK_{R_i}: the secret key of the i-th reviewer
- $VDV_{SK}(\cdot)$: the designated-verifier-signature (called DV-signature) verifying function with the private key SK in a UDVS scheme
- $DV_{PK}(\cdot)$: the designating function with the public key PK of the designated-verifier in the UDVS scheme
- A_m: the abstract of a paper m without containing any identification information of the author
- C_i: the decision of reviewer i for inspecting a paper, where $C_i \in \{\textsf{Yes}, \textsf{No}\}$
- $Time$: the string of time created by the time-stamp server
- $Comment_j$: the comment that reviewer j sends to the editor
- Candidate pool: the reviewers whose decision for inspecting a paper is Yes

In the following, our protocol consists of four phases: preparing, submitting papers, dispatching papers, inspecting, and declaring the result which are described in the followings.

4.2 Preparing Phase

In the preparing phase, there are five steps shown as follows.

1. An author ID_i chooses a random string r as a blinding factor.
2. Let m be an author's paper. She/He uses the blinding factor r to compute the blinded message $\alpha = B_T(H(m\|ID_i), r)$ and sends α to the time-stamp server.

3. Then the time-stamp server sets the string of current time $Time$ according to its clock. It signs α and $Time$ with the private key SK_{TS} by computing $Z = ST_{SK_{TS}}(\alpha, Time)$.

4. The time-stamp server forwards Z and $Time$ to the author.

5. The author uses her/his blinding factor r and U_T to unblind Z and then obtains $S = U_T(Z, r)$. The 4-tuple $(S, m, ID_i, Time)$ will satisfy $VT_{PK_{TS}}(S, H(m||ID_i), Time) =$ True. Thus, the author obtains a paper credential $Sig = (S, m, ID_i, Time)$

4.3 Submitting Papers

This phase is shown in the following.

1. The author encrypts her/his own paper m which does not contain any identity of the author and sends $E_{PK_E}(m)$ to the editor via an anonymous channel.

2. After receiving the encrypted paper $E_{PK_E}(m)$, the editor decrypts it with her/his private key SK_E to get m. By the way, the editor does not know who the real author of m is.

4.4 Dispatching Papers

When the editor received the paper m, she/he has to select some reviewers to inspect it. But it is an important issue that how the editor chooses them. We hope to prevent the reviewers from being bullied by the author. The followings are our dispatching steps.

1. First, the editor signs the abstract A_m of the paper m with the private key to generate $Sig_{A_m} = S_{SK_E}(H(A_m))$.

2. Then she/he encrypts A_m and Sig_{A_m} with the public key of each reviewer i by computing $EN_{A_m,R_i} = E_{PK_{R_i}}(A_m, Sig_{A_m})$.

3. She/He sends EN_{A_m,R_i} to each reviewer i and asks her/him to return the decision about inspecting this paper.

4. Reviewer i decrypts EN_{A_m,R_i} and reads the abstract A_m of the paper. She/He can also check the correctness of Sig_{A_m} via V_{PK_E}.

5. Reviewer i sets the decision C_i which may be Yes or No. Note that the author should set her/his decision as Yes if she/he also is a reviewer. If the author does not do so, it will be detected when the paper is accepted.

6. Reviewer i signs C_i and A_m to generate a PV-signature $\beta_i = S_{SK_{R_i}}(H(A_m||C_i))$.

7. Finally, reviewer i designates the editor as the designated-verifier by computing $\hat{\beta}_i = DV_{PK_E}(PK_{R_i}, \beta_i, A_m||C_i)$. She/He sets $\delta_i = (\hat{\beta}_i, (A_m, C_i))$ subsequently and sends it to the editor.

8. The editor verifies δ_i by using the DV-signature verifying function VDV with the PK_{R_i} and SK_E and then checks the decision of the reviewer. If the decision of the reviewer is Yes, the editor will add the reviewer to a candidate pool.

9. After all reviewers finishing step 3 to step 7, the editor chooses some reviewers in the candidate pool and go to next phase.

4.5 Inspecting and Declaring the Result

In this phase, the editor will decide whether the paper can be accepted or not according to the comments of the selected reviewers. The identities of the reviewers inspecting the paper cannot be known by anyone else. We make use of a UDVS scheme to achieve this goal.

1. The editor sends the ciphertext $EN_{m,R_j} = E_{PK_{R_j}}(m, Sig_m)$ to each selected reviewer j, where Sig_m is the signature of m signed by the editor.

2. Each selected reviewer j decrypts EN_{m,R_j} to get m and checks whether Sig_m is valid or not by V_{PK_E}. She/He writes down her/his comment $Comment_j$ and signs on it with m, i.e., each selected reviewer j computes $\gamma_j = S_{SK_{R_j}}(H(m||Comment_j))$.

3. Each selected reviewer j generates her/his DV-signatures $\hat{\gamma_j} = DV_{PK_E}(PK_{R_j}, \gamma_j, m|| Comment_j)$ and assigns the editor to be the designated-verifier. Then she/he sends back $\varepsilon_j = (\hat{\gamma_j}, (m, Comment_j))$ to the editor.

4. The editor verifies each ε_j via VDV with the PK_{R_j} and SK_E.

5. After all selected reviewers has sent back their own ε_j's, the editor can decide whether the paper m can be accepted according to the comments of the reviewers or not.

6. The result of inspecting paper m will be published by the editor. The author of the paper m must show her/his paper credential Sig which has been obtained in the first phase to convince the editor that she/he is the actual author of the paper m when the paper m is accepted.

5 Security

To demonstrate the security of our proposed anonymous paper submission and review scheme, we first show the security model and definitions and than give formal security proofs of our proposed scheme.

5.1 Security Model and Definitions

In this section, we formalize the security analysis of truly anonymous paper submission and review scheme (TAPSRS for short). First we define the partially blindness and the unforgeable properties of a general partially blind signature ($\Gamma = (B, S, H, U, V)$) as defined above in subsection 3.1. In the following, we define the game of "Partial Blindness" (PB for short).

Definition 1. The game for the Partially Blindness
Let $\mathcal{S}^*$ be the attacker engaging with two honesty users $\mathcal{U}_0$ and $\mathcal{U}_1$ in the following game.

1. **Setup.**

   (a) The simulator $\mathcal{C}$ runs the key generation algorithm to generate the singer's key pair (pk, sk).

(b) Then the simulator $\mathcal{C}$ gives the pk to the attacker $\mathcal{S}^*$ and $\mathcal{S}^*$ outputs two challenge plaintexts and common information $(m_0, m_1, info_{u_0}, info_{u_1})$.

(c) We setup the input tapes of $\mathcal{U}_0$ and $\mathcal{U}_1$ as follows:

i. Select a coin flip $b \in \{0, 1\}$ and take m_b, m_{1-b} on their private input tapes of $\mathcal{U}_0$ and $\mathcal{U}_1$.

ii. Then put the $info_b$ and $info_{1-b}$ on the public input tape of $\mathcal{U}_0$ and $\mathcal{U}_1$ with pk.

2. **Signing query.**

(a) $\mathcal{S}^*$ engages the signature protocol with two users $\mathcal{U}_0$ and $\mathcal{U}_1$.

(b) If $\mathcal{U}_0$ and $\mathcal{U}_1$ outputs $(info_0, m_0, sig_b)$, $(info_1, m_1, sig_{1-b})$, on their private output tapes and $info_0 = info_1$ holds, then give those outputs to $\mathcal{S}^*$.

3. **Output.** $\mathcal{S}^*$ outputs $b' \in \{0, 1\}$.

We define that the advantage of adversary $\mathcal{S}^*$ that wins in the game is $Adv_{\Gamma}^{PB}(\mathcal{S}^*) = |Pr[b = b'] - \frac{1}{2}| \geq \epsilon$.

Definition 2. Partial Blindness
A general signature scheme is partial blind (PB for short) if no polynomial adversary $\mathcal{S}^*$ with time t has the advantage $Adv_{\Gamma}^{PB}(\mathcal{S}^*) \geq \epsilon$ after performing the game of **Definition 1**.

Definition 3. The game for Unforgeability
We define the game of "Unforgeability" (Unf for short) of a partial blind signature scheme. Let $\mathcal{U}^*$ be the attacker and an honesty signer $\mathcal{S}$ and they engage the following game.

1. **Setup.** (pk, sk) was generated by the key generation algorithm and pk was given to the attacker $\mathcal{U}^*$ and sk is given to the signer $\mathcal{S}$. By the way, $\mathcal{U}^*$ can make the following training.

2. **Hash query.** The attacker $\mathcal{U}^*$ can make the hash query with the message m. When receiving this query, the simulator returns the hash value of m to $\mathcal{U}^*$.

3. **Signing Query.**

(a) During the run of the signing protocol with the signer $\mathcal{S}, \mathcal{U}^*$ can obtain the common information $info$ from the singer $\mathcal{S}$. Then $\mathcal{U}^*$ can make the signing query to $\mathcal{S}$.

(b) For each $info$, we define l_{info} to be the number of execution times of the signing protocol, where $\mathcal{S}$ outputs the valid signature with given $info$ (For $info$ that has never appeared on the input tape of $\mathcal{S}$, we define it as $l_{info} = 0$.).

4. **Output.** $\mathcal{U}^*$ wins the game if $\mathcal{U}^*$ outputs the common information $info$ and $l_{info} + 1$ signatures $(m_1, \sigma_1), \ldots, (m_{l_{info}+1}, \sigma_{l_{info}+1})$.

Let E_1 be the above event that $\mathcal{U}^*$ outputs $l_{info} + 1$ signatures after performing the l_{info} times query in the above game. We define the advantage of the adversary $\mathcal{U}^*$ that wins the game is $Adv_{\Gamma}^{Unf}(\mathcal{U}^*) = Pr[E_1] \geq \epsilon$.

Definition 4. Unforgeability
A general signature scheme is unforgeable if no polynomial adversary $\mathcal{U}^*$ with time t has the advantage $Adv_{\Gamma}^{Unf}(\mathcal{U}^*) \geq \epsilon$ after performing the game of **Definition 3**.

Definition 5. A Secure Partially Blind Signature
A signature protocol $\Gamma = (B, S, H, U, V)$ is a secure partially blind signature if the following properties are satisfied:

(1) Partially Blindness: The advantage of $\mathcal{S}^*$ that wins the game of **Definition 1** is negligible.

(2) Unforgeability: The advantage of $\mathcal{U}^*$ that wins the game of **Definition 3** is also negligible.

Then we can claim that Γ is a secure partially blind signature scheme.

In the followings, we define two properties of universal designated verifier signature scheme as mentioned in the Section 3.2.

Definition 6. The game for DV-Unfogeability
We define the DV-Unforgeability of a general universal designated signature scheme $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$ as mentioned above in Section 3.1. We consider the following game. Let $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$ be a UDVS scheme and let $\mathcal{A}$ be the forger that she/he attacks the unforgeability of Φ. The DV-unforgeability is defined as follows:

1. Attacker Input: Let signer and verifier's public key (pk_1, pk_3), where $(pk_1, sk_1) \longleftarrow GKS(cp)$, $(pk_3, sk_3) \longleftarrow GKS(cp)$ and $cp = GC(k)$.

2. Attacker Resources: Run-time plus program-length at most t, Oracle access to signer's singing oracle $S(sk_1, .)$ (q_s queries), and, if scheme Φ makes use of n random oracles $RO_1, ..., RO_n$ allow q_{RO_i} queries to the ith oracle RO_i for $i = 1, ..., n$. We write attacker's Resources Parameters(RPs) as $RP = (t, q_s, q_{RO_1}, ..., q_{RO_n})$.

3. Attacker Goal: Output a forgery message/DV-signature pair(m^*, $\hat{\sigma}^*$) such that

   (a) The forgery is valid, i.e. $VDV(pk_1, pk_3, m^*, \hat{\sigma}^*) = Acc$.

   (b) Message m^* is 'new', i.e. has not been queried by the attacker to S.

We say that Φ scheme is unforgeable in the sense of DV-unforgeability if, for any efficiently adversary $\mathcal{A}$, the probability $Adv_{A,\phi}^{DV-unf}$ that $\mathcal{A}$ succeeds in achieving above goal is at most ϵ, i.e. $Adv_{\phi}^{DV-unf}(\mathcal{A}) \leq \epsilon$.

Definition 7. The game for PR-Privacy
We define the PR-Privacy of a general universal designated signature scheme $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$ as mentioned above in Section 3.1. We consider the following game. Let $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$ be a UDVS scheme and let (A_1, A_2) denote an attackers against the privacy of Φ. Let $\widehat{A_1}$ denote a forgery strategy. The privacy notion PR is defined as follows:

1. Attacker Input: Signer public key pk_1, where $(pk_1, sk_1) = GKS(cp)$, and $cp = GC(k)$. Note that $\widehat{A_1}$ also accepts the program for A_1 as input.
2. Resources for $(A_1, \widehat{A_1})$: Run time $(t_1, \widehat{t_1})$ and access to signing oracle $S(sk_1, .)$ (up to $(q_s, \widehat{q_s})$ queried messages different from m^*), access to key-reg. protocol with the **KRA** (up to $(q_k, \widehat{q_k})$ interactions), access to A_2 oracle (up to $(q_c, \widehat{q_c})$ messages). In the stage 2, A_1 also has access to designation oracle **CDV**($pk_1, ., m^*, \sigma^*$) (up to q_d queried keys successfully registered with **KRA**), where $\sigma^* = S(sk_1, m^*)$ is a signer's signature on the challenge message m^* output by A_1 at end of stage 1. Note that $\widehat{A_1}$ can not make any designation queries.
3. Resources for A_2: Run-time t_2.
4. Attacker Goal: Let $P(A_1, A_2)$ and P($\widehat{A_1}, A_2$) denote the probabilities that A_2 outputs **yes** when interacting with A_1(game **yes**) and $\widehat{A_1}$ (game **no**), respectively. The goal of (A_1, A_2) is to achieve a non-negligible convincing measure $Adv^{PR-Privacy}_{\widehat{A_1},\Phi}(A_1, A_2) \overset{def}{=} |P(A_1, A_2) - P(\widehat{A_1}, A_2)|$.

Definition 8. A secure universal designated verifier signature scheme
A signature protocol $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$ is a secure universal designated verifier signature scheme if the following properties are satisfied:

(1) DV-unforgeability: The advantage of $\mathcal{A}$ that wins the game of **Definition 8** is negligible.

(2) PR-Privacy: The advantage of $\widehat{A_1}$ that wins the game of **Definition 9** is also negligible.

Then we can claim that Φ is a secure universal designated verifier signature scheme.
In the followings, we define three properties of truly anonymous paper submission and review system (TAPSRS for short) as mentioned in the Section 4.

Definition 9. The game for Unique
Let $\mathcal{B}$ be the attacker and she/he plays with the simulator $\mathcal{S}$ in the following game.

1. **Setup.** (pk, sk) was generated by the key generation algorithm of our proposed scheme and pk was given to the attacker $\mathcal{B}$ and sk is given to the signer $\mathcal{S}$. Attacker $\mathcal{B}$ can make the following training.
2. **Hash query.** The attacker $\mathcal{B}$ can make the hash query with the message m. When receiving this query, the simulator returns the hash value of m to $\mathcal{B}$.
3. **Signing query.**
   (a) During the run of the signing protocol with signer $\mathcal{B}$, $\mathcal{B}$ can obtain the time-stamp information $time_i$ from the time-stamped server with the help of the singer $\mathcal{S}$, where she/he was given the signing function S in the $\Gamma = (B, H, U, V, S)$ and $i = 1, ..., q_t$. Then $\mathcal{B}$ can engage the signing protocol with $\mathcal{S}$.
   (b) For each time-stamp $time_i$, let l_{time_i} be the number of execution times of the signing protocol, where $\mathcal{S}$ outputs the valid signature with given $time_i$ (For each $time_i$ that has never appeared on the input tape of $\mathcal{S}$ and this $time_i$ is the earliest one of $\mathcal{S}$, we define it as the $l_{time_i} = 0$). Here, we assume that $l_{time_i} = 0$.

4. **Output.** We claim that $\mathcal{B}$ wins the game if $\mathcal{B}$ outputs $time^* < time_i$ and $l_{time^*} + 1$ signatures $(m_1, \sigma_1), \ldots, (m_{l_{time^*}+1}, \sigma_{l_{time^*}+1})$.

Let E_2 be the event that the adversary $\mathcal{B}$ outputs the $l_{time^*} + 1$ signatures and $time^* < time_i$ after performing l_{time^*} times query in the game, where $i = 1, \ldots, q_t$. We define the advantage of the adversary $\mathcal{B}$ that wins the game is $Adv^{Uni}_{TAPSRS}(\mathcal{B}) = Pr[E_2] \geq \epsilon$. From the above game, we can discover that only the real author can show the exactly time $time_i$ proof and related signature on her/his submitted papers and the attacker can not forge a early time $time^*$ signature to claim that she/he is the real author with the non-negligible advantage ϵ in the polynomial time t, where $time^* \leq time_i$ and $i = 1, \ldots, q_t$.

Definition 10. Unique
Our proposed scheme (TAPSRS) is unique (Uni for short) on each author's submitted papers if no polynomial adversary $\mathcal{B}$ with time t has the advantage $Adv^{Uni}_{TAPSRS}(\mathcal{B}) \geq \epsilon$ after performing the game of **Definition 9**.

Definition 11. The game for Comment Unforgeability
Let $\mathcal{C}$ be the attacker and she/he plays with the simulator $\mathcal{S}$ in the following game of the Inspecting and Declaring phase of our scheme.

1. **Setup.** (pk_E, sk_E) and (pk_{R_i}, sk_{R_i}) were generated by the key generation algorithm of our proposed scheme for editor E and each reviewer i, where $1 \leq i \leq n$. (pk_E, pk_{R_i}) were given to the attacker $\mathcal{C}$ and sk_{R_i} and sk_E are given to the PV signature oracle and VDV oracle, respectively. Attacker $\mathcal{C}$ can make the following training.

2. **Hash query.** The attacker $\mathcal{C}$ can make the hash query with the abstract of the paper A_m. When receiving this query, the simulator returns the hash value of A_m to $\mathcal{C}$.

3. **PV signature query.** When the attacker $\mathcal{C}$ makes the PV-signature query on the message m, the simulator checks if it exists in the PV-signature list L_{pv}. If not, the simulator computes the PV signature Sig_m and stores (Sig_m, m) into the list L_{PV}. Then it returns Sig_m back to $\mathcal{C}$.

4. **DV signature query.** When $\mathcal{C}$ makes the DV-signature query on the j-th message and PV-signature pair $(m||Comment_i, \gamma_i)$ with the reviewer i's public key $pk_{R_i} \in \{pk_{R_1}, \ldots, pk_{R_n}\}$, the simulator checks if there exists a DV-signature ε_i in the DV-signature list L_{dv}. If not, the simulator generates the reviewer i's DV-signature ε_i. Then it keeps $(\varepsilon_i, m||Comment_i, \gamma_i)$ into the list L_{dv} and returns ε_i back to $\mathcal{C}$.

5. **VDV verification query.** When $\mathcal{C}$ makes the DV-signature verification query on the j-th DV-signature $(\varepsilon_i, m||Comment_i, Sig_m)$, $\mathcal{A}$ forwards it to VDV oracle and returns the verification result $d \in \{Acc, Rej\}$ to $\mathcal{C}$.

6. **Secret key query.** When $\mathcal{C}$ queries the secret key of the public key pk_{R_i}, where $pk_{R_i} \in \{pk_{R_1}, \ldots, pk_{R_n}\}$, the simulator returns the secret key sk_{R_i} back to $\mathcal{C}$.

We say the $\mathcal{C}$ wins the above game if $\mathcal{C}$ outputs a forged signature $(m^*, Comment^*, \varepsilon^*)$ with the public key pk^* and $Comment^*$ after making above all queries such that:

1. $VDV(pk^*, sk_E, \varepsilon^*, m^*, Comment^*) = Acc$.

2. m^* has never asked the PV-signature oracle before.

3. $(m^*||Comment^*)$ has never asked the DV-signature oracle before.
4. pk^* has never submitted as one of the **Secret key query** before.

We define that the advantage of $\mathcal{C}$ wins the above game is

$$Adv_{TAPSRS}^{Comment-Unf}(\mathcal{C}) \geq \epsilon.$$

Definition 12. Comment Unforgeability
Our proposed scheme (TAPSRS) is Comment Unforgeable ($Comment - Unf$ for short) if no polynomial adversary $\mathcal{C}$ with time t has the advantage $Adv_{TAPSRS}^{Comment-Unf}(\mathcal{C}) \geq \epsilon$ after performing the game of **Definition 11**.

Definition 13. The game for Honesty

We define the Honesty of our TAPSRS scheme as mentioned above in Section 3.1. We consider the following game. Let $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$ be a secure UDVS scheme and let $\mathcal{F}$ denote a forgery strategy. On the other hand, she/he plays with the simulator $\mathcal{S}$ in the following game of the dispatching phase of our TAPSRS scheme.

1. **Setup.** (pk_E, sk_E) and (pk_{R_i}, sk_{R_i}) were generated by the key generation algorithm of our proposed scheme for editor E and each reviewer i, where $1 \leq i \leq n$. $(pk_E, pk_{R_i}, sk_{R_i})$ were given to the attacker $\mathcal{F}$ and sk_E are given to the VDV signature oracle and PV-signature oracle. Attacker $\mathcal{F}$ can make the following training queries.
2. **Hash query.** The attacker $\mathcal{F}$ can make the hash query with the abstract A_m of the paper m. When receiving this query, the simulator returns the hash value of A_m to $\mathcal{F}$.
3. **PV signature query.** When the attacker $\mathcal{F}$ makes the editor's PV-signature query on the message A_m and the decision C_i, where C_i = Yes, the simulator checks if it exists in the PV-signature list L_{pv}. If not, the simulator computes the signature $\beta_i = Sig_{A_m} = S_{sk_E}(H(A_m||C_i))$ and stores (β_i, A_m, C_i) into the list L_{PV}. Then it returns β_i back to $\mathcal{F}$.
4. **DV signature query.** When $\mathcal{F}$ makes the DV-signature query on the user i's message/PV-signature pair $(A_m||C_i, \beta_i)$ with the reviewer i's public key $pk_{R_i} \in \{pk_{R_1}, \ldots, pk_{R_n}\}$ and the editor's public key pk_E, the simulator checks if there exists a DV-signature δ_i in the DV-signature list L_{dv}. If not, the simulator generates the reviewer i's DV-signature δ_i. Then it keeps $(\delta_i, A_m||C_i, \beta_i)$ into the list L_{dv} and returns δ_i back to $\mathcal{F}$.
5. **VDV verification query.** When $\mathcal{F}$ makes the DV-signature verification query on the the editor's DV-signature $(\delta_i, A_m||C_i, \beta_i)$, $\mathcal{S}$ forwards it to VDV oracle and returns the verification result $d \in \{Acc, Rej\}$ to $\mathcal{F}$.

We say the $\mathcal{F}$ wins the above game if $\mathcal{F}$ outputs a forged signature $(A_m^*, \beta_i^*, \delta_i^*, C_i^*)$ with the public key $pk^* \in \{pk_{R_1}, \ldots, pk_{R_n}\}$ after making above all queries such that:

1. $V(pk^*, \beta_i^*, A_m^*, C_i^*) = Acc$ and C_i^*=Yes in the dispatching phase.
2. $VDV(pk^*, sk_E, \delta_i^*, A_m^*, C_i^*) = Acc$ but C_i^* =No after reviewing phase.

We define that the advantage of $\mathcal{F}$ wins the above game is

$$Adv_{TAPSRS}^{Honesty}(\mathcal{F}) \geq \epsilon.$$

From the above game, we can discover that if the signer chooses $C_i = \textsf{Yes}$ in the dispatching paper phase, then she/he can be discovered when her/his paper is accepted and she/he must proof her/his C_i and (β_i, δ_i, A_m) to the editor. In other word, the attacker can not have the non-negligible advantage ϵ to forge a DV-signature δ^* which its $C_i = \textsf{Yes}$ in the dispatching phase but C_i becomes $\textsf{No}$ after the paper reviewing.

Definition 14. Honesty

Our proposed scheme (TAPSRS) is Honest if no polynomial adversary $\mathcal{F}$ with time t has the advantage $Adv_{TAPSRS}^{Honesty}(\mathcal{F}) \geq \epsilon$ after performing the game of **Definition 13**.

Definition 15. A secure truly anonymous paper submission and review scheme

A truly anonymous paper submission and review scheme is secure if the following properties are satisfied:

(1) Unique: The advantage of $\mathcal{B}$ that wins the game of **Definition 9** is negligible.

(2) Comment Unforgeability: The advantage of $\mathcal{C}$ that wins the game of **Definition 11** is also negligible.

(3) Honesty: The advantage of $\mathcal{F}$ that wins the game of **Definition 13** is also negligible.

(4) Anonymity: This property is included in the **PR-Privacy** from the universal designated verifier signature scheme in **Definition 8**. The third party can not distinguish the signature which was generated from the actual signer or the designated verifier.

Then we can claim that our truly anonymous paper submission and review scheme is secure. In the followings, we give the proofs of these properties of truly anonymous paper submission and review system (TAPSRS for short) as mentioned above.

5.2 Security Proofs

Theorem 1. If there exists an attacker $\mathcal{B} - (\epsilon, t, q_t, q_s, q_h, q_l)$ who can break the property unique in **Definition 10** of our proposed scheme (TAPSRS for short), then there exists a challenge $\mathcal{C} - (\epsilon^*, t^*)$ who can break the property unforgeability in **Definition 4** of the secure partially blind signarue scheme $\Gamma = (B, H, U, V, S)$, where

$$\epsilon \geq \frac{\epsilon^*}{(\frac{1}{q_h}(\frac{1}{q_t+q_l})^{q_s} + (\frac{1}{q_t+q_l})^{q_s})}$$

$$t^* \leq t - (q_s(q_t + q_l) + q_h)$$

with at most q_s times signing queries, q_t times time-stamp queries, q_l common information queries, and q_h times hash queries.

Proof. Suppose that there exists an attacker $\mathcal{B}$ that she/he wins the the game of **Definition 3** with advantage at least ϵ. We can take $\mathcal{B}$ as the black box and construct an adversary $\mathcal{C}$ against the underlying partial blind signature scheme ($\Gamma = (B, H, U, V, S)$).

- **Setup.** In the simulation of the game of **Definition 10**, $\mathcal{C}$ prepares all parameters including the signing oracle S response to attacker $\mathcal{B}$. After setting up all parameters, $\mathcal{C}$ simulates the game of **Definition 10** with the attacker $\mathcal{B}$.
- **Training.** During the simulation, the attacker $\mathcal{B}$ can ask the hash query and the signing query. The simulator will forward these queries to the signing function S and H in the scheme Γ, respectively. We assume that the attacker $\mathcal{B}$ can ask at most q_t times time-stamp query, q_h times hash query, and q_s signing query, respectively. $\mathcal{C}$ performs the corresponding result in the following.
  - Hash query: If $\mathcal{B}$ asks the hash query with m_i to $\mathcal{C}$, $\mathcal{C}$ computes the hash value α_i and adds (m_i, α_i) into the hash list, where $i \in (1, \ldots, q_h)$. Then $\mathcal{C}$ returns α_i back to $\mathcal{B}$.
  - Sign query: If $\mathcal{B}$ asks the signing query with α_i to $\mathcal{C}$, $\mathcal{C}$ fetches the $time_i$ from the time-stamp server and forwards (α_i, $time_j$) to signing oracle S, where $j \in (1, \ldots, q_t)$. Then the signing oracle chooses a time $time_j$ and a common information $info_k$, sets $info_k$ into the $time_j$, and computes the signature $Z = S_{SK_{TS}}(\alpha_i, time_j)$, where $k \in (1, \ldots, q_l)$. Then it returns $(Z, time_j)$ back to attacker $\mathcal{B}$ and adds $(\alpha_i, time_j, info_k, Z, l_{time_j})$ into the signing list.

  After l_{time^*} times queries, if $\mathcal{B}$ forges $l_{time^*}+1$ signatures S^* on m^* with $time^*$ successfully, $\mathcal{C}$ can use $\mathcal{B}$'s ability to break the unforgeability of the scheme Γ. We consider the following cases that $\mathcal{B}$ produces the forged signature S^* on (m^*, $time^*$) with time list $\{time_1, \ldots, time_{q_t}\}$. Then we define two events in the following case.

  1. E_3 be the event that $\mathcal{C}$ does not hold in the signing query of the simulation.
  2. E_4 be the event that $\mathcal{C}$ does not hold in the signing query and $\mathcal{B}$ forged $l_{time^*} + 1$ signatures successfully.
- Case 1: If $time^* < time_j$ and $m^* \in \{m_1, \ldots, m_{q_h}\}$, where for all $j = 1, \ldots, q_t$, it means that the attacker $\mathcal{B}$ forges a new signature S^* on the message m^* with a earliest time before the author one.

  1. $l_{time^*} = 0$:

     1-1. In this situation, the probability of event E_3 is that

     $$Pr[E_3] = \frac{1}{q_h}(\frac{1}{q_t + q_l})^{q_s}.$$

     1-2. Then we discuss the probability of event E_4. In this situation, E_4 be the event that $\mathcal{C}$ does not hold in the signing query and $\mathcal{B}$ forged $l_{time^*} + 1$ signatures successfully. During the simulation, we can see that the probability of E_4 is the probability of $\mathcal{B}$ forged $l_{time^*} + 1$ signatures successfully. That is

     $$Pr[E_4|E_3] \geq \epsilon.$$

- Case 2: If $time^* < time_j$, where for all $j = 1, \ldots, q_t$ and the attacker $\mathcal{B}$ forges the signature S^* on a new message $m^* \notin \{m_1, \ldots, m_{q_h}\}$ before the author one. We take this as the framing situation that is the attacker $\mathcal{B}$ which uses the author's identity to submit the low quality paper to conference or journal in order to decrease the credit of the author.

  1. $l_{time^*} = 0$:

  2-1. When $\mathcal{B}$ forges a signature S^* on the message m^* and $m^* \notin \{m_1, \ldots, m_{q_t}\}$, the simulator can not find the entry from the (m_i, α_i) and $(\alpha_i, time_i, info_j, Z, l_{time_i})$ of the hash list and the above signing list, respectively. If $l_{time^*} = 0$ and the simulator can not find the matched entry of the signing list, it means that the $info_j$ is a new common information.
  In this situation, the probability of E_3 is

$$Pr[E_3] \geq (\frac{1}{q_t + q_l})^{q_s}.$$

  2-2. Then we discuss the probability of event E_4. In this situation, E_4 be the event that $\mathcal{C}$ does not hold in the signing query and $\mathcal{B}$ forged $l_{time^*} + 1$ signatures successfully. During the silmulation, we can see that the probability of E_4 is the probability of $\mathcal{B}$ forged $l_{time^*} + 1$ signatures successfully. That is

$$Pr[E_4|E_3] \geq \epsilon.$$

Finally, we can conclude that the probability of $\mathcal{C}$ who breaks the general partial blind signature is

$$\begin{aligned} Adv_{\Gamma}^{Unf}(\mathcal{C}) &= Pr[E_2] = Pr[E_4 \wedge E_3] = Pr[E_3]Pr[E_4|E_3] \\ &\geq \epsilon(\frac{1}{q_h}(\frac{1}{q_t + q_l})^{q_s} + (\frac{1}{q_t + q_l})^{q_s}) \geq \epsilon^*. \end{aligned}$$

□

Theorem 2. If there exists an attacker $\mathcal{C} - (\epsilon, t, q_h, q_{pv}, q_{dv})$ who can break the property comment unforgeability in **Definition 12** of our TAPSRS scheme, then there exists a challenge $\mathcal{A} - (\epsilon^*, t^*)$ who can break the property DV-unforgeability in **Definition 8** of the secure universal designated verifier signarue scheme $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$, where

$$\epsilon \geq \frac{\epsilon^*}{((1 - \frac{1}{2^k})^{q_{pv}^2} \cdot (1 - \frac{1}{2^k})^{q_{dv}})}$$

$$t^* \leq t - (q_{pv} + q_{dv} + q_h)$$

with at most q_h times hash queries, q_{pv} times **PV-Signature** queries, and q_{dv} times **DV-signature** queries in the polynomial time t^*.

Proof. Suppose that there exists an attacker $\mathcal{C}$ that he/she wins the the game of **Definition 11** with advantage at least ϵ. We can take $\mathcal{C}$ as the black box and construct an adversary $\mathcal{A}$ against the underlying universal designated verifier signature scheme (i.e., $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$). Then $\mathcal{A}$ starts to simulate the environment and $\mathcal{C}$ can make the following queries.

- **Key generation.** Before the environment simulation, $\mathcal{A}$ chooses the editor and reviewer's public key (pk_E, pk_E), where $(pk_E, sk_E) \longleftarrow GKS(cp)$, $(pk_i, sk_i) \longleftarrow GKS(cp)$, $cp = GC(k)$ and $i \in \{1, \ldots, n\}$. After generating these key pairs, $\mathcal{A}$ gives (pk_E, pk_i) to the $\mathcal{C}$, where $i \in \{1, \ldots, n\}$ and sets sk_i and sk_E to the PV-signature oracle $S(sk_i, .)$ and VDV-signature verification oracle $VDV(., sk_E, .)$, respectively.

- **hash query.** When $\mathcal{C}$ makes the hash query on the message m, $\mathcal{A}$ transfers this query to the random oracle RO_i, where $i \in \{1, \ldots, n\}$. Then $\mathcal{A}$ returns the hash value ϑ back to $\mathcal{A}$ and keeps (ϑ, m) into the hash list L_h.

- **PV-signature query.** When $\mathcal{C}$ queries the PV-signature on the message m, $\mathcal{A}$ forwards it to the PV-signature oracle $S(sk_E, \cdot)$. Then $\mathcal{A}$ returns the signature value Sig_m back to $\mathcal{A}$ and stores (Sig_m, m) to the PV-signature list L_{pv}.

- **DV-signature query.** When $\mathcal{C}$ makes the DV-signature query on the $(m||Comment_j, \gamma_j)$ with the public key pk_{R_j}, where $j \in \{1, \ldots, n\}$. In this time, $\mathcal{A}$ sets $m' = (m||Comment_j)$ and forwards (m', pk_{R_j}) to the CDV-signature oracle in the scheme Φ. After obtaining the result ε_j, $\mathcal{A}$ returns $(\varepsilon_j, m, Comment_j)$ to $\mathcal{C}$ and stores $(\varepsilon_j, m, Comment_j, pk_{R_j})$ into the list L_{dv}.

- **VDV-signature verification query.** When $\mathcal{C}$ queries the VDV-signature verification query on $(\varepsilon_j, m, Comment_j, pk_{R_j})$, $\mathcal{A}$ forwards $(\varepsilon_j, m, Comment_j, pk_{R_j})$ to the VDV-signature verification oracle. Then the $\mathcal{A}$ returns the result $d \in \{Acc, Rej\}$ to $\mathcal{C}$.

After querying all the above queries, if $\mathcal{C}$ wins the game defined in **Definition 11** that it outputs a forged DV-signature $(m^*, Comment^*, \varepsilon_j^*)$ on the public key pk_{R_j}, where $j \in \{1, \ldots, n\}$. Then $\mathcal{A}$ can use $\mathcal{C}$'s ability to break the property defined in **Definition 8**.

Then we define two events in the following case.

1. E_5 be the event that $\mathcal{A}$ does not hold in the PV-signing query of the simulation.

2. E_6 be the event that $\mathcal{A}$ does not hold in the DV-signing query of the simulation.

3. E_7 be the event that $\mathcal{A}$ does not hold in the VDV-signature verification query and $\mathcal{C}$ forged a DV-signature $(m^*, Comment_j, \varepsilon_j^*)$ on the public key pk_{R_j}, where $j \in \{1, \ldots, n\}$ successfully.

- Case 1: In the event E_5, we can discover that $\mathcal{A}$ does hold when $\mathcal{C}$ queries the signature on message m^*. Then we can conclude that the probability of event E_5 is

$$Pr[E_5] \geq (1 - \frac{1}{2^k})^{q_{pv}}$$

with at most q_{pv} times PV-signature queries.

- Case 2: In the event E_6, we can discover that $\mathcal{A}$ does hold when $\mathcal{C}$ queries the DV-signature on the message $m^* = m'$ and $Comment = Comment^*$ and the PV-signature γ_j has been queried **PV-signature query** before. Let E_{6-1} be the event that $\mathcal{C}$ queries the DV-signature on the message $m' = m^*$ and $Comment = Comment^*$ and E_{6-2} be the event that PV-signature γ_j has never been queried **PV-signature query** before.

  1. If $m' = m^*$ and $Comment = Comment^*$, then $\mathcal{A}$ does not hold and the probability of E_6 is

     $$Pr[E_{6-1}] \geq (1 - \frac{1}{2^k})^{q_{dv}}.$$

  2. If γ_j has never queried queried **PV-signature query** before, then we can conclude that the probability of E_{6-2} is

     $$Pr[E_{6-2}] \geq (1 - \frac{1}{2^k})^{q_{pv}}.$$

  Hence, we can summarize that the probability of E_6 is

  $$Pr[E_6] \geq Pr[E_{6-1}] \cdot Pr[E_{6-2}] \geq (1 - \frac{1}{2^k})^{q_{dv}} \cdot (1 - \frac{1}{2^k})^{q_{pv}}.$$

  On the hand, We also compute the probability of $Pr[E_6|E_5]$ and we can discover that these two events are independent. So we can conclude that

  $$Pr[E_6|E_5] \geq Pr[E_6] \geq (1 - \frac{1}{2^k})^{q_{dv}} \cdot (1 - \frac{1}{2^k})^{q_{pv}}.$$

- Case 3: In this situation, if the attacker $\mathcal{C}$ can forge a DV-signature ε_j^* on the message (m^*, $Comment^*$), the probability of E_7 is

  $$Pr[E_7] \geq \epsilon.$$

  On the other hand, we also consider the probability of $Pr[E_7|E_6 \wedge E_5]$. In these three events, we can discover that the event E_7 and E_6 are both independent. The event E_6 is also independent of the event E_5. Then we can conclude that the probability is

  $$Pr[E_7|E_5 \wedge E_6] \geq Pr[E_7] \geq \epsilon.$$

Hence, we summarize that above events that $\mathcal{C}$ attacker outputs a forged DV-signature ε_j^* on message ($m^*, Comment^*$) and we can build $\mathcal{A}$ to break the **DV-Unforgeability** in the **Definition 8**. The probability of the attacker $\mathcal{A}$ is

$$\begin{aligned}
&Pr[E_5 \wedge E_6 \wedge E_7] \geq \epsilon^* \\
&= (Pr[E_5]) \cdot (Pr[E_6|E_5]) \cdot (Pr[E_7|E_5 \wedge E_6]) \geq \epsilon^* \\
&= ((1 - \tfrac{1}{2^k})^{q_{pv}}) \cdot ((1 - \tfrac{1}{2^k})^{q_{dv}} \cdot (1 - \tfrac{1}{2^k})^{q_{pv}}) \cdot (\epsilon) \geq \epsilon^* \\
&= (\epsilon \cdot (1 - \tfrac{1}{2^k})^{q_{pv}} \cdot (1 - \tfrac{1}{2^k})^{q_{pv}} \cdot (1 - \tfrac{1}{2^k})^{q_{dv}}) \geq \epsilon^* \\
&= \epsilon \geq \frac{\epsilon^*}{((1 - \frac{1}{2^k})^{{q_{pv}}^2} \cdot (1 - \frac{1}{2^k})^{q_{dv}})}.
\end{aligned}$$

□

Theorem 3. If there exists an attacker $\mathcal{F} - (\epsilon, t, q_t, q_s, q_h, q_l)$ who can break the property honesty in **Definition 13** of our proposed scheme (TAPSRS for short), then there exists a challenger $\mathcal{S} - (\epsilon^*, t^*)$ who can break the property DV-unforgeability in **Definition 6** of the secure universal designated verifier signarue scheme $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$, where

$$\epsilon \geq \frac{\epsilon^*}{(\frac{1}{2^n})^{q_{dv}}}$$

$$t^* \leq t - (q_{pv} + q_h + q_{dv})$$

with at most q_{pv} times PV-signature queries, q_h times hash queries, and q_{dv} times DV-signature queries.

Proof. Suppose that there exists an attacker $\mathcal{F}$ that he/she wins the the game of **Definition 13** with advantage at least ϵ. We can take $\mathcal{F}$ as the black box and construct an adversary $\mathcal{S}$ against the underlying universal designated verifier signature scheme (i.e., $\Phi = (GC, GKS, GKV, S, V, CDV, VDV, P_{KR})$). Then $\mathcal{S}$ starts to simulate the environment and $\mathcal{F}$ can make the following queries and games, respectively.

1. **Setup.** (pk_E, sk_E) and (pk_{R_i}, sk_{R_i}) were generated by the key generation algorithm of our proposed scheme for editor E and each reviewer i, where $1 \leq i \leq n$. $(pk_E, pk_{R_i}, sk_{R_i})$ were given to the attacker $\mathcal{F}$ and sk_E are given to the VDV signature oracle and PV-signature oracle. Attacker $\mathcal{F}$ can make the following training queries.

2. **Hash query.** The attacker $\mathcal{F}$ can make the hash query with the abstract A_m of the paper m. When receiving this query, the simulator returns the hash value of A_m to $\mathcal{F}$.

3. **PV signature query.** When the attacker $\mathcal{F}$ makes the editor's PV-signature query on the message A_m and the decision C_i, where $C_i = \mathsf{Yes}$, the simulator checks if it exists in the PV-signature list L_{pv}. If not, the simulator computes the signature $\beta_i = Sig_{A_m} = S_{sk_E}(H(A_m||C_i))$ and stores (β_i, A_m, C_i) into the list L_{PV}. Then it returns β_i back to $\mathcal{F}$.

4. **DV signature query.** When $\mathcal{F}$ makes the DV-signature query on the user i's message/PV-signature pair $(A_m||C_i, \beta_i)$ with the reviewer i's public key $pk_{R_i} \in \{pk_{R_1}, \ldots, pk_{R_n}\}$ and the editor's public key pk_E, the simulator checks if there exists a DV-signature δ_i in the DV-signature list L_{dv}. If not, the simulator generates the reviewer i's DV-signature δ_i. Then it keeps $(\delta_i, A_m||C_i, \beta_i)$ into the list L_{dv} and returns δ_i back to $\mathcal{F}$.

5. **VDV verification query.** when $\mathcal{F}$ makes the DV-signature verification query on the the editor's DV-signature $(\delta_i, A_m||C_i, \beta_i)$, $\mathcal{S}$ forwards it to VDV oracle and returns the verification result $d \in \{Acc, Rej\}$ to $\mathcal{F}$.

In the following, we consider the following events that $\mathcal{F}$ forges a signature $(\delta_i^*, A_m||C_i^*, \beta_i^*)$ such that

1. Let E_1 be the event that $V(\beta_i^*, A_m^*, C_i^*) = Acc$ and $C_i^* = \mathsf{Yes}$.

2. Let E_2 be the event that $VDV(pk^*, sk_E, \delta_i^*, A_m^*, C_i^*) = Acc$ but $C_i = \mathsf{No}$.

1. Case E_1: In this event, $\mathcal{S}$ can know that the attacker $\mathcal{F}$ can impersonates the user i and generates the PV-signature $(\beta_i^*, A_m^*, C_i^*)$, with $C_i^* = \mathsf{Yes}$ and the help of secret key Sk_{R_i}. Then we can conclude that the probability

$$Pr[E_1] = 1.$$

2. Case E_2: In the event, we consider that the attacker $\mathcal{F}$ must wins the game of the **Definition 15** with the non-negligible probability ϵ in the polynomial time t. When $\mathcal{F}$ outputs the forged DV-signature δ_i^* with (A_m^*, C_i^*), where $C_i = \mathsf{No}$, then the simulator $\mathcal{S}$ can use the ability of $\mathcal{F}$ to break the DV-unforgeability of the general UDVS scheme Φ in the **Definition 6**. Then We conclude the probability of the event E_2

$$Pr[E_2] \geq \epsilon \cdot (\frac{1}{2^n})^{q_{dv}}.$$

On the other hand, we consider that the probability of the event $E_2 \wedge E_1$ is

$$Pr[E_2|E_1] \geq Pr[E_2] \geq \epsilon \cdot (\frac{1}{2^n})^{q_{dv}}.$$

We conclude that the probability of the event $E_2 \wedge E_1$ is

$$\begin{aligned} Adv_{TAPSRS}^{Honesty}(\mathcal{F}) &\geq Pr[E_2 \wedge E_1] \\ &= Pr[E_2|E_1] \cdot Pr[E_1] \\ &= \epsilon \cdot (\frac{1}{2^n})^{q_{dv}} \cdot 1 \\ &\geq Adv_{UDVS}^{DV-unf}(\mathcal{S}) \\ &= \epsilon^*. \end{aligned}$$

□

6 Evaluation

6.1 Security Analysis

In this section, we will explain why our protocol satisfies all requirements shown in Section 3.3.

1. Anonymity:
The author's identity is blinded in the preparing phase. The author chooses a blinding factor to hide her/his name. Nobody can know who the actual author is before the paper is accepted.

    (a) Author → Editor: The author's network address is kept secret by using an anonymous channel when she/he submits her/his paper to the editor. The editor only receives an anonymous paper without any unnecessary information. Therefore, it is successful to keep the author anonymous to the editor before her/his paper is accepted.

(b) Author → Reviewer: The reviewers get the same message as that the editor received in the submitting paper phase, where it is an anonymous paper which does not contain the author's identity. The author is also anonymous to the reviewers.

(c) Reviewer → Author: When a reviewer sends her/his comments on the author's paper to the editor, she/he takes the UDVS scheme [16], [17] to offer her/his privacy protection. The reviewer is the signer of the comments and the signature designator. She/He designates the editor as the only verifier to check the designated-verifier signature which produced by the reviewer. But the editor can also take her/his secret key to generate the DV-signature which is the same as the one produced by the reviewer. When the author receives the comments published by the editor, she/he cannot know who the reviewer is. The editor cannot prove that the DV-signature $\hat{\gamma}_j$ was produced by reviewer j. Thus, the reviewer's identity is unknown to the author.

The author's name cannot be known before the paper is accepted by the editor. An author can submit her/his paper to any conference or journal with privacy protection. Owing to the anonymity property, the editor will allocate the paper to reviewers more fairly in the dispatching paper phase. The editor does not have any information about the author such that she/he can just follow a reasonable process to dispatch papers. During the review process of each reviewer, she/he can provide her/his comments just depending on the professional knowledges without being influenced by the reputation of the author. Also, the reviewer is only responsible to the editor and she/he is anonymous to other people including the author. She/He is not afraid to write negative comments on the paper to offend the author. In the inspecting phase, the editor receives the DV-signature $\hat{\gamma}_j$ from the reviewer j. She/He cannot convince the author that $\hat{\gamma}_j$ was made by the reviewer j. We take advantages of UDVS such that the editor (the designated verifier) can produce the same signature $\hat{\gamma}_j$. Finally, the editor decides whether the paper is accepted or not only depending on the comments received from the reviewers. It will be more fair in this situation.

2. Uniqueness:
To modify $(m||ID_i)$ and $Time$ in Sig produced in the first phase is infeasible since the time-stamp server has signed on them. Thus, an attacker cannot forge a signature containing m and an earlier time $Time'$ to impersonate the actual author ID_i in l times query, where $Time' \leq Time_i$ for all $i = 1, ..., l$. A pilferer may steal the paper m after it is submitted and then get another time-stamp signature Sig' and submits m to another conference or journal, but she/he will be detected when she/he shows her/his Sig' and ID'_i. The $Time'$ in Sig' is always later than $Time$ in Sig. Hence, the paper can only be owned by a unique author or a unique group of authors. In the appendix, we provide a formal proof of this property.

3. Comment Unforgeability:
In the UDVS scheme [16], [17], the unforgeability has been concluded. The unforgeability of a UDVS scheme contains DV-signature unforgeability and PV-signature unforgeability. We make use of the two unforgeabilities to achieve comment unforgeability by adopting a secure general UDVS scheme which satisfies the two properties. In the appendix, we also offer a formal proof of this property.

4. Honesty:

In the dispatching paper phase, we show the abstract of the paper to all reviewers to ask them to return their decisions about inspecting the paper. The editor chooses some reviewers whose decisions are Yes. Here, we ask the author of the paper to set her/his decision as No. Therefore, if she/he sets her/his C_i = Yes and was selected by the editor, the editor can detect it in the final phase. When the author shows her/his identity to claim that the accepted paper was written by herself/himself, the editor can check whether the author is one of the selected reviewers. In other words, the authors can not forge a signature which its C_i = Yes in the dispatching phase, but its C_i = No of the DV-signature after reviewing phase. In the appendix, we give a formal proof of this property.

6.2 Comparison

The comparisons among our scheme, the traditional paper review system and the previous protocols [13], [3] are shown in Table 1.

Table 1: Property Comparisons

	P1	P2	P3	P4	P5	P6
Ours	✓	✓	✓	✓	✓	✓
T[1]	×[2]	×[3]	×	✓	×	✓
[13]	△[4]	×	△[4]	✓	✓	×
[3]	△[5]	△[5]	△[5]	✓	✓	✓

✓: Satisfied; ×: Not satisfied
△: Satisfied under some strong assumption
P1: Author Anonymity to Reviewer
P2: Reviewer Anonymity to Author
P3: Author Anonymity to Editor
P4: Comment Unforgeability
P5: Uniqueness
P6: Honesty

[1] T : The traditional paper review system
[2] The editor may reveal the author's identity
[3] The editor may reveal the reviewers' identities
[4] It needs a fully trusted third party to guarantee the property.
[5] It needs a semi-trusted third party and two additional trusted servers to guarantee the property.

6.3 Usability

There are some issues that pertain to the implementation of the proposed anonymous paper submission and review scheme, as follows.

- The problem of ciphertext length expansion on adoption of anonymous channels is discussed below. One should choose an anonymous channel scheme in which the length of ciphertext is irrelevant to the number of MIXes (control centers). Otherwise, the length of ciphertext will grow along with the number of MIXes and thus makes the scheme inefficient.

- The proposed scheme is independent of the underlying signature scheme. To implement the scheme, one should use an efficient signature scheme. Otherwise, an inefficient signature scheme will influence the whole system performance significantly.

7 Conclusions

In this manuscript, we have proposed an anonymous paper submission and review scheme which can make paper review more fair. We adopt a partially blind signature scheme and a universal designated-verifier signature scheme as the underlying primitives to construct the proposed anonymous paper submission and review scheme. The anonymity property in the proposed scheme can achieve the most important property, i.e., fairness. Therefore, the attendants in our scheme can more fairly perform their jobs without worrying about anything. All features of our scheme are summarized as follows:

1. The proposed scheme fully protects the privacy of authors and reviewers.
2. It can be realized and implemented easily.
3. The proposed idea is independent of the underlying partially blind signature scheme and UDVS scheme and we can take any secure partially blind signature and UDVS schemes to implement it.
4. It is flexible and extensible for any kind of paper review schemes.

Acknowledgment

This work was partially supported by the Ministry of Science and Technology of the Taiwan under grant MOST 103-2221-E-110-057, NSYSU-KMU Joint Research Project (NSYSUKMU 103-I001), and Aim for the Top University Plan of the National Sun Yat-sen University and Ministry of Education, Taiwan, R.O.C.

References

[1] M. Abe, E. Fujisaki, How to date blind signatures, *Advances in Cryptology, ASIACRYPT 1996, Lecture notes in computer science LNCS 1163*, pp.244-251, 1996.

[2] M. Abe, T. Okamoto, Provably secure partially blind signatures, *Advances in Cryptology CRYPTO 2000, Lecture notes in computer science LNCS 1880*, pp.271-286, 2000.

[3] E. Aïmeur, G. Brassard, S. Gambs, D. Schönfeld, P3ERS: Privacy-Preserving PEer Review System, *Trans. Data Privacy*, vol.5, pp.553-578, 2012.

[4] L. Bornmann, H. D. Daniel, Fairness and predictive validity of committee peer review, *FUTUR*, vol.19, pp.7-19, 2004.

[5] J. Camenisch, E. V. Herreweghen, Design and implementation of the Idemix Anonymous Credential System, *Research Report RZ 3419, IBM Research Division, ACM Computer and Communication Security*, 2002.

[6] D. Chaum, Untraceable electronic mail, return address, and digital pseudonyms, *Communications of the ACM*, vol.24(2), pp.84-88, 1981.

[7] C. I. Fan, Improved low-computation partially blind signatures, *Applied Mathematics and Computation*, vol.145, pp.853-867, 2003.

[8] C. I. Fan, C. L. Lei, Low-computation partially blind signatures for electronic cash, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol.E81A, pp.818-824, 1998.

[9] C. I. Fan, C. L. Lei, A user efficient fair blind signature scheme for untraceable electronic cash, *Journal of Information Science and Engineering*, vol.18, pp.47-58, 2002.

[10] E. V. Herreweghen, Unidentifiability and accountability in electronic transactions, *PhD Thesis*, KULeuven, 2004.

[11] M. Jakobsson, K. Sako, R. Impagliazzo, Designated verifier proofs and their applications, *Advances in cryptology-EUROCRYPT'96, Lecture notes in computer science LNCS 1070*, pp.143-154, 1996.

[12] G. R. Michael, F. S. Paul, M. G. David, Anonymous connections and onion routing, *IEEE Journal on Selected Areas in Communication*, vol.16(4), pp.482-493, 1998.

[13] V. Naessens, L. Demuynck, B. D. Decker, A fair anonymous submission and review system, *Communications and Multimedia Security, Lecture notes in computer science LNCS 4237*, pp.43-53, 2006.

[14] W. Ogata, K. Kurosawa, K. Sako, K. Takatani, Fault tolerant anonymous channel, *Information and communications security, Lecture notes in computer science LNCS 1334*, pp.440-444, 1997.

[15] C. Park, K. Itoh, K. Kurosawa, Efficient anonymous channel and all/nothing election scheme. *In Proc. Workshop on the theory and application of cryptographic techniques on advances in cryptology, ACM Portal*, pp.248-259, 1994.

[16] R. Steninfeld, L. Bull, H. Wang, J. Piperzyk, Universal designated-verifier signatures. *Advances in Cryptology ASIACRYPT 2003, Lecture notes in computer science LNCS 2894*, pp.523-542, 2003.

[17] R. Steninfeld, H. Wang, J. Piperzyk, Efficient extension of standard Schnorr RSA signatures into universal designated-verifier signatures, *Public Key Cryptography-PKC 2004, Lecture notes in computer science LNCS 2947*, pp.86-100, 2004.

[18] R. Zhang, J. Furukawa, H. Imai, Short signature and universal designated verifier signature without random oracle, *Applied cryptography and network security, Lecture notes in computer science LNCS 3531*, pp.483-498, 2005.