Multi-Partied Quantum Digital Signature Scheme Without Assumptions On Quantum Channel Security

Mustafa ŞAHİN1, İhsan YILMAZ2
1,2 Çanakkale Onsekiz Mart University, Computer Eng. Dept., Canakkale, Turkey
E-mail: msahin@comu.edu.tr, iyilmaz@comu.edu.tr

Abstract. In this study, we generalized quantum digital signature scheme for three-partied[5] to the multi-partied. Also we analyzed this scheme for repudiation, forgery and colluding adversaries.

1. Introduction
Digital signatures are like signatures on paper. The main purpose is to identify the signatory and to determine any falsification made on the document. Available methods of digital signature are based on difficult mathematical problems [8]. However, to solve this problem several quantum algorithms are proposed [1]. Gottesman and Chuang[7] has proposed concept of quantum digital signature. The method of Gottesman and Chuang[7] becomes unsuccessful as in the states of swap-test $|\psi\rangle_B|\phi\rangle_C + |\phi\rangle_B|\psi\rangle_C$. Complete-symmetry test in Buhrman et.al[2] is used to solve this problem. Andersson et al.[9] showed that coherent states and linear optical elements can be used instead of swap-test experimentally. Dunjko et al.[10] showed that QDS can be generated without quantum memory. They used optic and photon detector. Walden et al.[11] showed that available usable/sell-able quantum key distribution devices can be used for the solution of the some problems. Juan Miguel Arrazola et al.[3] generalized the schema of Dunjko and Wallden[10] to the multiuser schema. Hua-Lei Yin et al.[4] show that it is not necessary authenticated quantum channel. Amiri et al.[5, 6] developed quantum signature without needing all security assumptions on quantum channel.

In this study, we generalized quantum digital signature scheme for three-partied[5] to the multi-partied. We don’t have any assumptions about trust of quantum channels between participants. And all participants creates keys and send part of it to the message sender so that sender can not create any entangled keys and sends them to the participants to repudiate them. And, also we analyzed this scheme for repudiation, forgery and colluding adversaries.

2. Key Generation Stage
In this study, to create correlated but not the same keys between participants, we have used quantum key distribution(QKD)[13, 14] via phase randomized weak-coherent states and decoy states[15]. We don’t do post-processing stage of QKD so that these keys have some little mismatches. Asymmetric coding of BB84[17] was used. $\{X, Z\}$ basis are chosen with different probabilities $\{p_X, p_Z\}$. $p_X \geq 1/2, p_Z = 1 - p_X$ and also intensities $\{\mu_1, \mu_2, \mu_3\}$ of laser pulses can be chosen with different probabilities $\{p_{\mu_1}, p_{\mu_2}, p_{\mu_3}\}$. Where $\mu_1 > \mu_2 + \mu_3$ and $\mu_2, \mu_3 \geq 0$. The
intensities \{\mu_2, \mu_3\} are used for decoy states. Secret keys will be created from the measurement results which Alice and Bob choose X basis and same intensities.

Following the same procedure in[5] Bob is sender and Alice is receiver to prevent against Alice who can create entangled states. Also she can send the same key to the other participants. After measurement, Bob selects \(L + n_{cor}\) bits from \(X_\mu\) and randomly divide \(L\) bits into two part with length \(L/2\), \((K^B_{keep},K^B_{forward})\). \(K^B_{forward}\) bits will be used to forward to the other participants. \(n_{cor}\) bits, \(Cor^B_X\) from X basis measurements will be shared via classical channel to calculate correlation between Alice’s and Bob’s keys \(\{K^A_m, K^B_m\}\). \(Cor^B_X\) won’t be used into protocol. \(Z\) will be used to estimate Eve’s information about Bob’s key. To achieve this we will use the smooth min-entropy. We would like to bound Eve’s success rate about signature creation which includes less errors from certain number of errors. So we will calculate the smooth min entropy of Eve about Bob’s key \(K^B_{keep}\).

Bob has following key set \(\{Cor^B_X, Z^B, K^B_{keep}, K^B_{forward}\}\). Lengths of the keys are \(|Cor^B_X| = n_{cor}, |Z^B| = \eta, |K^B_{keep}| = |K^B_{forward}| = L/2\). Alice also has these keys with the same manner.

As mentioned, \(Cor^B_X\) will be used to calculate correlation between \(K^A\) and \(K^B\). \(Z\) can be used to gain information about eavesdropper existence. In the protocol \(K^B_{forward}\) will be sent to other participants to protect himself against repudiation from Alice.

Since Eve can get information about the measurement of \(\{Z^A, Z^B\}\) on quantum channel, and about \(Cor^B_X, Cor^A_X\), and also about \(basis^B, basis^A\). We can calculate information of Eve about \(K^B_{keep}\) by evaluating correlations of \(\{Z^A, Z^B\}\). Also we assume that Eve is a participant in the protocol, so Eve has \((K^B_{forward, Cor^B_X, Cor^A_X, basis^B, basis^A})\). Eve’s all information system can be denoted as \(H_E\).

The number of mismatches with receiver’s key is important for quantum signature protocols. We will show how we can bound the forger’s mismatch count for our protocol. Thus if we provide bigger mismatch count for forger’s (message,signature) pair then any receiver won’t accept the message with small probability. Our main goal is to achieve that \(d(K^A_m, K^B_m) < d(K^E_m, K^B_m)\). \(K^E_m\) is the Eve’s guess about Bob’s key by using all collected information about keys. Where \(d(.,.)\) represents Hamming distance between the two states. Min and Smooth entropy [19] are used to calculate information of states.

Lim et al.[12] and Ma et al. [16] gives bounds for Eve’s information about Bob’s key in the context of QKD. We will use same ways to bound Eve’s mismatch count about Bob’s key. But Eve has extra information \(K^B_{forward}\).

By extracting error correction and privacy amplification steps in [12], we can write following [5]:

\[ H^*_{min}(X^B_{keep}|E) \geq S_{X,0} + S_{X,1} - S_{X,1} h(\phi_{X,1}). \]

(1)

Where \(S_{X,0}, S_{X,1}\) corresponds respectively number of vacuum events and number of single photon events in X basis measurements. We discard double photon events and never use in the protocol due to photon-splitting attacks. Where \(\phi_{X,1} = c_{X,1}/S_{X,1}\) is the phase error rate in X measurements where \(c_{X,1}\) is the number of phase errors. And \(h(\cdot)\) is binary entropy[16].

As shown and proofed in [5]:

\[ < p_r > \leq \sum_{k=0}^{r} \binom{L/2}{k} 2^{-H^*_{min}(X^B_{keep}|E)p_{X_{keep}E}} + \epsilon \]

(2)

This shows Eve’s probability of making at most \(r\) mistakes by using best strategies to guess \(X_{keep}\) with all information he/she has. In this equation \(p_{X_{keep}E}\) is \(\epsilon - close\) to near optimal \(p_{X_{keep}E}^*\). As \(L/2\) grows binom coefficient can be written as \(\binom{L/2}{k} \approx 2(L/2)h(\frac{x}{L/2})\). By using this approximation and Eq.(1), Eq.(2) can be written as:
\[ < p_r > \leq 2^{-(S_{X,0} + S_{X,1} - S_{X,1} h(\phi_{X,1})) + (L/2)h(\frac{r}{L/2}) + \epsilon} \]  

(3)

We can get \( L/2 \) parenthesis as follows:

\[ < p_r > \leq 2^{-(L/2)(\frac{S_{X,0} + S_{X,1} - S_{X,1} h(\phi_{X,1}) - h(\frac{r}{L/2})) + \epsilon} \]  

(4)

If we have neglected \( \epsilon \) in Eq. (4), we get the following equation.

\[ \frac{S_{X,0}}{L/2} + \frac{S_{X,1}}{L/2} - \frac{S_{X,1}}{L/2} h(\phi_{X,1}) - h(\frac{r}{L/2}) > 0 \]  

(5)

Which determines Eve’s probability of making fewer than \( r \) errors. To bound this probability we can define minimum error rate \( E_{R_{Eve}} \) of Eve as follows:

\[ h(E_{R_{Eve}}) = \left( \frac{S_{X,0}}{L/2} + \frac{S_{X,1}}{L/2} - \frac{S_{X,1}}{L/2} h(\phi_{X,1}) \right) \]  

(6)

If we achieve error rate between Alice-Bob lesser than error rate of Eve about Bob’s key \( X^{B_{keep}} \) then the protocol will be secure. The condition of \( E_{R_{Eve}} > E_{R_{AB}} \) must be held. Alice and Bob can use \( k \) length of \( Cor_X^B \) and \( Cor_X^A \) keys to calculate the estimate of error rate about keys \( k^A \) and \( B^B \) which are used in the protocol. By using \( Cor_X^B \) and \( Cor_X^A \) samples with Serfling[20] inequality we can bound \( k^A, k^B \) error rate as follows:

\[ E_{R_{AB}} \leq \frac{E_{R_{AB}}}{ER_{AB}} + \varepsilon_{ER_{AB}} \]  

(7)

Where \( \frac{E_{R_{AB}}}{ER_{AB}} \) is observed from \( \{ Cor_X^B, Cor_X^A \} \) keys and \( \varepsilon_{ER_{AB}} \) is calculated from Serfling inequality as follows [20]:

\[ \varepsilon_{ER_{AB}} \leq \sqrt{\frac{\ln(P_k(\epsilon))}{2k} \left( 1 - \frac{k - 1}{L/2} \right)} \]  

(8)

If we select acceptance threshold value \( t_{e1} > E_{R_{AB}} \) in the protocol then we can bound \( E_{R_{Eve}} \) with small probability like in Eq. (8).

3. Multi-Partied Quantum Signature Protocol with Majority Vote Dispute Resolution

Arrazola et.al.[18] defined properties of the multi-party unconditionally secure signature(USS) protocols to achieve their security goals as Non-Forgery, Non-repudiation and Transferability. Quantum Signature Protocol has been formed with the following sets and functions[18]:

(i) \( P = \{ P^0, P^1, P^2...P^{N-1} \} \) are \( N \) different participants. We assumed that \( P^0 \) is sender and \( P^i, i = 1..N - 1 \) are the other possible receivers and dispute resolvers.

(ii) \( X \) is possible messages set and \( \Sigma \) is the possible signature set of the messages \( X \).

(iii) \( Sign(x) \) is a signature function that maps \( X \rightarrow \Sigma \). That \( Sign(x) \) function gets \( x \) value and returns \( \sigma \in \Sigma \).

(iv) \( l = \{ -1, 0, 1, 2, 3...l_{max} \} \) is the set of verification levels that \( l \) verification level means that how many times a signed message can be transferred to the other participants. \( l_{max} \leq N - 1 \). \( l = 0 \) guarantees that this message comes from signer but does not guarantees it can be forwarded other participants. \( l = -1 \) level will be used in Majority-Vote dispute resolution so \( l = -1 \) level is used to prevent repudiation from sender.
(v) $\text{Ver}(X, \Sigma, P, l) \rightarrow \{\text{True}, \text{False}\}$ gets message, signature, participant and verification level and returns $\{\text{True}, \text{False}\}$ which means this message and signature will be accepted by $P_i$ participant at level $l$.

All participants have imperfect quantum channels and authenticated classical channels between them to make post-processing. We can outline the protocol for single bit value $m = \{0, 1\}$. To generalize to the multi-bit message, the protocol can be repeated. The protocol can be outlined as follows:

(i) **Key Generation And Distribution Stage:** $P_i, i = 1..N - 1$ participants uses individually Key Generation Stage to create and share $L$ length key pairs $K_m(P^1), \bar{K}_m(P^1 \rightarrow P^0), K_m(P^2), \bar{K}_m(P^2 \rightarrow P^0), \ldots, K_m(P^{N-1}), \bar{K}_m(P^{N-1} \rightarrow P^0)$ with $P_0$. $K_m(P^i), i = 1..N - 1$ are receiver’s keys and $\bar{K}_m(P^i \rightarrow P^0), i = 1..N - 1$ are sender’s keys which are received from quantum channel with noise and has little changes. All participants send their key pair’s part to $P_0$. $P_0$ collects all keys $\bar{K}_m(P^i \rightarrow P^0), i = 1..N - 1$ which are sent to her from other participants to transfer message to them later. $P_0$ use signature $\text{Sig}_m = (\bar{K}_m(P^i \rightarrow P^0), i = 1..N - 1)$ to transfer message $m$ to the other participants. Note that only $P_0$ knows all keys which were sent by other participants that will protect the protocol against forgery.

(ii) **Key Sharing Stage:** $P_i, i = 1..N - 1$ participants randomly choose $\lfloor |L|/2 \rfloor$ of bits from their keys $K_m(P^i), i = 1..N - 1$ and then send them with the positions to the other participants except $P_0$ via authenticated classical channel. $P_i, i = 1..N - 1$ save the positions and values of the forwarded bits. $P_i, i = 1..N - 1$ will firstly and importantly use the unsent bits and bits which come from other parties to check the validity of the signature that will be sent from $P_0$. All participants except $P_i$ shares half of the her/his keys to protect him/her selfs against repudiation from $P_0$. Since $P_0$ does not know which positions of the keys are sent to the other participants, $P_0$ can not deceive $P_i, i = 1..N - 1$ about signature. Secondly, they use forwarded part of the keys to protect him/her selfs against collective adversaries. We can show their new keys as $S^0_{m^i}, S^1_{m^i}, \ldots, S^{N-1}_{m^i}$ that formed from unsent bits and other forwarded keys from other participants. The length of the new keys is $(N - 1)L/2$. $P_i, i = 1..N - 1$ knows all bits of $K_m^{P_i}$ and half of the $\{K_m^{P_j}, j \neq i, j = 1..N - 1\}$. If say $P_i$ is dishonest, he knows only half bits of the $\{K_m^{P_j}, j = 2..N - 1\}$, so $P_i$ can not forge the other participants.

(iii) **Messaging Stage:** In this stage, $P_0$ wants to send one bit message $m = \{0, 1\}$ to $P_i$ or any other participant. Then receiver checks the validity of the signature and then forwards message and signature to the other participants.

(a) $P_0$ sends $(m, \text{Sig}_m)$ one bit message to $P_i$ (or any $P^j$).

(b) $P^j$ checks the $\text{Sig}_m$, with his unsent and forwarded parts of $K_m^{P_j}$, and keys which are sent by other participants. Then $P^j$ saves separately unmatched bit counts. $P^i$ accepts the message if the following mismatch count rules are hold.

1. The mismatch count of unsent part of the key $K_m^{P_i}$, and the parts of the $S^0_{m^i}$ which are received from other participants is less than $t_{v1}L/2$. Where $t_{v1} < 1/2$ is a acceptance threshold that is determined by the security level and settings of the protocol. Our protocols verification function can be defined as follows:

$$\text{Ver}(m, \text{Sig}_m, P^i, l = -1) = \begin{cases} \text{True} & |\text{Ver}(m, \text{Sig}_m, P^j, l = 0) = \text{True}| > \frac{N-1}{2} \\ \text{False} & \text{otherwise} \end{cases}$$

$$\text{Ver}(m, \text{Sig}_m, P^i, l = 0) = \begin{cases} \text{True} & \text{MC}(\text{Sig}_m^{P^j}) < t_{v1}\frac{L}{2} \\ \text{False} & \text{MC}(\text{Sig}_m^{P^j}) > = t_{v1}\frac{L}{2} \end{cases}$$

(9)
Where $MC(.)$ is mismatch count in the parts of the signature. If we select $t_{v_j}$ as in Eq.(12) then $Ver(m, Sig_m, P^i, l \geq 1)$ will be hold.

2. The mismatch count of sent part of the key $K_m^{P_1}$ is less than $t_{v_j}L/2$.

(c) $P^1$ forwards the data $(m, Sig_m)$ to the one of the other participant $P^j, j = 2..N - 1$, say $P^2$.

(d) $P^2$ checks the unmatched count in the same manner with $P^1$ but he/she uses different threshold value $t_{v_j}$. Where $0 < t_{v_1} < t_{v_2} < 1/2$. Every participant must use different acceptance threshold for the $(m, Sig_m)$ which is sent from $P^0$ or forwarded by other participant. The number of forward determines the threshold, so every forward process needs bigger threshold. This is necessary of all unconditionally secure signature protocols [18]. $P^2$ then forwards $(m, Sig_m)$ to the other participant.

(e) All participants can use the same manner except with bigger threshold values.

Majority-Vote Dispute Resolution: This method is invoked whenever there is a disagreement on a message’s signature is valid for all users. When this method returns False all participants accept that signature is invalid otherwise valid. If more than half of the users get $Ver(x, \sigma, i, -1) = True$ than signature is valid for all users otherwise invalid.

4. Security Analysis

We can define this protocol’s security analysis as follows:

(i) **Non-forgery**: Since all participants have half keys of the other participants, they can try to forge any other participant. We assume $P_i$ wants to forge $P_j, j \neq i, i, j = 1..N - 1$. To achieve forging $P_i$ must send $(m, Sig_m)$ to $P_j$ with fewer than $t_{v_j}L/2$ mismatch count. Due to $P_i$ knows only half key of the $P_j$, $K_j^{P_j}$ then $P_i$ cannot achieve $t_{v_j}L/2$ threshold condition for the other part of $P_j$, $K_j^{P_j}$.

Let $ER^{MAX} = \max \ ER_{P_i}^{P_j}, i, j = 0..N - 1$ is the maximum of the error rates observed between $\{Cor_{P_i}^{P_j} and Cor_{P_j}^{P_i}\}$ with k size. By calculating Eq.(6) for all key generation stages between $P^0$ and $P^j, j = 1..N - 1$ then we can get $ER_{Eve}^{MIN} = \min \ ER_{Eve}^{(P_i, P_j), j = 1..N - 1}$. If we select $t_{v_j}$ between $0 \leq ER^{MAX} < t_{v_j} < ER_{Eve}^{MIN} \leq 1/2$ then $P_i$ participant will accept the legitimate $(m, Sig_m)$ from $P_0$ and also will reject forged $(m, Sig_m)$ from the participant $P_i, i \neq j, i, j = 1..N - 1$. Because all verification functions must be different we can choose acceptance threshold values as follows:

$$h = \frac{|ER_{Eve}^{MIN} - ER^{MAX}|}{N - 1}$$

$$v_j = ER^{MAX} + h_j \tag{12}$$

Every threshold value is different from the other at least $h$. Then threshold values holds $ER^{MAX} < t_{v_1} < t_{v_2} < ... < t_{v_{N-1}} < ER_{Eve}^{MIN}$. By increasing $L$ this protocol will be much secure. If at least $(N-1)/2$ participants are dishonest in the protocol, non-forgery condition cannot be achieved.

(ii) **Non-repudiation**: $P_0$ wants that $P_i, i = 1..N - 1$ accepts validity of $(m, Sig_m)$. When $P_i$ forwards $(m, Sig_m)$ to the other participants $P_j, j \neq i, i, j = 1..N - 1$, they rejects the same $(m, Sig_m)$. In order that $P_j$ participant reject $(m, Sig_m)$, he/she has bigger mismatch count than $t_{v_j}L/2$ mismatch count some part of $Sig_m$. If $P_j$ accepts the message which is sent by $P_0$, then $P_i$ has lesser mismatch count than $t_{v_j}L/2$ mismatch count for all parts of $Sig_m$. Note that $P_i$ has half parts of other $P_j$ participants’ exact keys $K_j^{P_j}$ due to protocol. To repudiate $P_0$ must inject more errors $P_j$’s key part but it is impossible due to key sharing part of protocol.
5. Conclusion
In this study we have tried to develop multi-participant quantum digital signature protocol by extending Amiri[5]. We generalized this protocol to the multi-participant case. We used majority vote dispute resolution and bounded-ordered acceptance thresholds for this protocol to make secure against repudiation and forgery. We also used decoy states against to the photon number splitting attack. Also majority vote dispute resolution method has advances according to using arbiter. Since the acceptance mechanism of the majority vote method is distributed to the all participants, this method is much more secure than arbiter.

References
[1] Shor. P. W., Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, SIAM Journal on Computing, 26(5), pp. 1484-1509, (1997).
[2] Buhrman H., Cleve R., Watrous J., and Wolf R. Quantum Fingerprinting, Physical Review Letters, vol:87, num:16, (2001).
[3] Arrazola J. M, Walden P., and Andersson E. Multiparty Quantum Signature Schemes, arXiv:quant-ph/1505.07509v1 (2015).
[4] Yin H. Y., Yao F., Chen Z. B. Practical Quantum Digital Signature, arXiv:quant-ph/1507.03333v3 (2016).
[5] Amiri R., Walden P., Kent A., and Andersson E. Secure Quantum Signatures Using Insecure Quantum Channels, arXiv:quant-ph/1507.02975v1 (2015).
[6] Amiri R., and Andersson E. Unconditionally Secure Quantum Signatures. Entropy, 17, 5635-5659, (2015).
[7] Gottesman, D. and Chuang, I. Quantum Digital Signatures:arXiv:quant-ph/0105032 (2001).
[8] Lamport L. Constructing digital signatures from a one-way function. Technical Report CSL-98-, SRI International (October 1979).
[9] Andersson, E., Curty, M., and Jex, I. (2006). Experimentally realizable quantum comparison of coherent states and its applications. Physical Review A, 74(2), 022304-1, 10.1103/PhysRevA.74.022304
[10] Dunjko, V., Wallden, P. and Andersson, E. Quantum digital signatures without Quantum Memory. Phys. Rev. Lett. 112, 040502 (2014).
[11] Wallden, P., Dunjko, V., Kent, A., Andersson, E. Quantum digital signatures with quantum-key-distribution components. Phys. Rev. A 91, 042304 (2015).
[12] C. C. W. Lim, M. Curty, N. Walenta, F. Xu, and H. Zbinden, Concise security bounds for practical decoy-state quantum key distribution, Phys. Rev. A 89, 022307 (2014).
[13] C. H. Bennett and G. Brassard, in Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing, Bangalore, India (IEEE, New York, 1984), pp. 175–179.
[14] V. Scarani, H. Bechmann-Pasquinucci, N. J. Cerf, M. Dusek, N. Lütkenhaus, and M. Peev, ibid. 81, 1301 (2009).
[15] W. Y. Hwang, Quantum Key Distribution with High Loss: Toward Global Secure Communication, Phys. Rev. Lett. 91, 057901 (2003).
[16] Ma X., Qi B., Zhao Y., and Lo H-K., Practical decoy state for quantum key distribution, Physical Review A, 72, 012326 (2005).
[17] H.-K. Lo, H. F. Chau, and M. Ardehali,Decoy-State Quantum Key Distribution. J. Cryptol. 18, 133 (2005).
[18] J. M. Arrazola, P. Wallden, and E. Andersson, Multiparty quantum signature schemes, Quantum Inf. Comput. 6, 0435 (2016).
[19] Renner R. Security of quantum key distribution PhD Thesis Swiss Federal Institute of Technology (2005), arXiv:quant-ph/0512258v2.
[20] R. J. Serfling, Probability inequalities for the sum in sampling without replacement, Ann. Stat. 2, 39 (1974).