yesi novaria kunang <yesinovariakunang@binadarma.ac.id>

Reviewer Invitation for EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions

1 message

Journal of Computer Security <em@editorialmanager.com> Tue, Jun 9, 2020 at 4:39 AM
Reply-To: Journal of Computer Security <editorial@iospress.nl>
To: Yesi Novaria Kunang <yesinovariakunang@binadarma.ac.id>

Dear Mrs. Kunang,

You have been invited to review a revision of a manuscript for Journal of Computer Security.

I would be grateful if you would re-review a paper entitled "EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions" for this journal. Your original comments can be found at the end of this e-mail. They can also be found online once you agree to re-review this paper.

If you would like to re-review this paper, please click this link: https://www.editorialmanager.com/j-cs/l.asp?i=28248&l=OHB3H325 *

If you do not wish to re-review this paper, please click this link: https://www.editorialmanager.com/j-cs/l.asp?i=28249&l=RF8EDRUP *

If the above links do not work, please go to https://www.editorialmanager.com/j-cs/. Your User Name is yesikunang. If you do not know your confidential password, you may reset it by clicking this link: https://www.editorialmanager.com/j-cs/l.asp?i=28250&l=UWU6DVTV

The manuscript reference is JCS-191416R1.

If possible, I would appreciate receiving your review by Jun 23, 2020 (IF JOURNAL IS IN 'INVITATION MODE'). If possible, I would appreciate receiving your review in 15 days (IF JOURNAL IS IN 'AGREED MODE'). You may submit your comments online at the above URL. There you will find spaces for confidential comments to the editor, comments for the author and a report form to be completed.

With kind regards

Javier Lopez
Associate Editor

*If clicking the link above does not open an Editorial Manager window, your email program may have inserted some spaces and/or line markers into the link. Please open a browser window manually and copy and paste the entire link from the email into the url address box. The link starts with the letters "http" and ends with the letters "rev=X" (where X represents a number such as 0,1,2, etc.) Note that the end of the link may be shown on a different line in this email, and may be shown in a different color than the beginning of the link. The entire link must be copied and pasted into the browser in order for the correct Editorial Manager window to be displayed. After copying the link into the url address box, you must also remove any spaces and line markers (e.g. > or >>) by using the delete or backspace keys on your keyboard.

In compliance with data protection regulations, you may request that we remove your personal registration details at any time. (Use the following URL: https://www.editorialmanager.com/j-cs/login.asp?a=r). Please contact the publication office if you have any questions.

yesi novaria kunang <yesinovariakunang@binadarma.ac.id>

Thank you for agreeing to review

1 message

Journal of Computer Security <em@editorialmanager.com> Sat, Jun 13, 2020 at 2:11 PM
Reply-To: Journal of Computer Security <editorial@iospress.nl>
To: Yesi Novaria Kunang <yesinovariakunang@binadarma.ac.id>

Dear Mrs. Kunang,

Thank you for agreeing to review manuscript JCS-191416R1 for Journal of Computer Security.

I would be grateful if you would review a paper entitled "EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions" for this journal.

To download the paper now, please click this link: https://www.editorialmanager.com/j-cs/l.asp?i=28305&l=CTUZG8UJ *

If possible, I would appreciate receiving your review by Jun 23, 2020.

You may submit your comments online at https://www.editorialmanager.com/j-cs/. Your User Name is yesikunang. If you do not know your confidential password, you may reset it by clicking this link: https://www.editorialmanager.com/j-cs/l.asp?i=28306&l=O622IEGY

You may also submit your comments using this link: https://www.editorialmanager.com/j-cs/l.asp?i=28307&l=PQL3R585

There you will find spaces for confidential comments to the editor, comments for the author and a report form to be completed.

With kind regards

Javier Lopez
Associate Editor
Journal of Computer Security

*If clicking the link above does not open an Editorial Manager window, your email program may have inserted some spaces and/or line markers into the link. Please open a browser window manually and copy and paste the entire link from the email into the url address box. The link starts with the letters "http" and ends with the letters "rev=X" (where X represents a number such as 0,1,2, etc.) Note that the end of the link may be shown on a different line in this email, and may be shown in a different color than the beginning of the link. The entire link must be copied and pasted into the browser in order for the correct Editorial Manager window to be displayed. After copying the link into the url address box, you must also remove any spaces and line markers (e.g. > or >>) by using the delete or backspace keys on your keyboard.

Review_Due.ics
1K

Journal of Computer Security

EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions

--Manuscript Draft--

Manuscript Number:	JCS-191416R1
Full Title:	EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions
Article Type:	Research Article
Keywords:	QR code; e-coupon; e-commerce; commitment algorithms; visual cryptography
Abstract:	In recent years, with the rapid development and popularization of e-commerce, the applications of e-coupons have become a market trend. As a typical bar code technique, QR codes can be well adopted in e-coupon-based payment services. However, there are many security threats to QR codes, including the QR code tempering, forgery, privacy information leakage and so on. To address these security problems for real situations, in this paper, we introduce a novel fragment coding-based approach for QR codes using the idea of visual cryptography. Then, we propose a QR code scheme with high security by combining the fragment coding with the commitment technique. Finally, an enhanced QR code-based secure e-coupon transaction framework is presented, which has a triple-verification feature and supports both online and offline scenarios. The following properties are provided: high information confidentiality, difficult to tamper with and forge, and the ability to resist against collusion attacks. Furthermore, the performance evaluation of computing and communication overhead is given to show the efficiency of the proposed framework.
Response to Reviewers:	Thanks a lot for all the professional and constructive suggestions from the reviewers. We are grateful for their time and efforts in reviewing our manuscript titled “EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions”. The detailed comments and suggestions are helpful for us to further improve the paper. We have carefully studied the comments and revised our manuscript accordingly. The major revisions in this paper are as follows: 1. We have introduced new research development of e-coupons, as well as several secure e-coupon systems in Section 2 Related Work. 2. We have further introduced the cryptographic techniques to make it clearer for readers in Section 3.1 Cryptographic Techniques. 3. We have illustrated some typical situations to show how the proposed framework resists the threats in practice and achieves the security goals in Section 5.7 Defense Against the Typical Attacks. 4. We have compared our work with other related secure e-coupon protocols in security properties, computation overhead and communication overhead. Details can be found in Section 6.4 Comparison with Related Protocols. 5. We have further discussed the possible generalization, remaining problems and poten- tial solutions of the proposed framework in Section 7 Further Discussion. 6. We have further explained Table 2, 3, and Fig. 5, 6, 9 and 10. We have also revised Table 3 and made two new tables, i.e., Table 6 and 7. 7. We have made the code we used in the work public. The code can be found at: https://github.com/RuiLiu-Uvic/EQRC. 8. We have rearranged the structure of the paper to make it more reasonable and logical. 9. We have revised the sentences to improve the quality of the paper. Unclear notations and points have been further explained. Spelling and grammatical errors have been corrected. We provide detailed point-by-point replies in the attacked response letter.

Response to Reviewers' Comments

Thanks a lot for all the professional and constructive suggestions from the reviewers. We are grateful for their time and efforts in reviewing our manuscript titled "EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions". The detailed comments and suggestions are helpful for us to further improve the paper. We have carefully studied the comments and revised our manuscript accordingly. The major revisions in this paper are as follows:

1. We have introduced new research development of e-coupons, as well as several secure e-coupon systems in Section 2 Related Work.

2. We have further introduced the cryptographic techniques to make it clearer for readers in Section 3.1 Cryptographic Techniques.

3. We have illustrated some typical situations to show how the proposed framework resists the threats in practice and achieves the security goals in Section 5.7 Defense Against the Typical Attacks.

4. We have compared our work with other related secure e-coupon protocols in security properties, computation overhead and communication overhead. Details can be found in Section 6.4 Comparison with Related Protocols.

5. We have further discussed the possible generalization, remaining problems and potential solutions of the proposed framework in Section 7 Further Discussion.

6. We have further explained Table 2, 3, and Fig. 5, 6, 9 and 10. We have also revised Table 3 and made two new tables, i.e., Table 6 and 7.

7. We have made the code we used in the work public. The code can be found at: https://github.com/RuiLiu-Uvic/EQRC.

8. We have rearranged the structure of the paper to make it more reasonable and logical.

9. We have revised the sentences to improve the quality of the paper. Unclear notations and points have been further explained. Spelling and grammatical errors have been corrected.

Note that the major revisions in the manuscript are highlighted in yellow.
Below, we provide detailed point-by-point replies. Related sentences revised in the manuscript are also quoted in deep blue.

Reviewer 1

We appreciate the time and efforts that the reviewer has dedicated to providing the valuable feedback.

Reviewer Point P 1.1 — *Need further investigation to evaluate the security threats performance of the proposed Secure QR codes. The performance evaluations are too limited. The author only evaluate the efficiency side of the running time overhead and delay.*

Reply: Thanks for your valuable suggestions. To explain the security performance on threats clearly, we have done more work and added a subsection, Section 5.7. In this subsection, some specific situations are illustrated. We consider an adversary and show how the proposed framework can defend against the typical attacks, including the eavesdropping, data leakage, e-coupon forgery and modification, collusion attack, message forgery and modification and replay attack.
However, there are many uncertain factors and difficulties to simulate attacks directly in the current work. The reasons are as follows. There are many kinds of attacks mentioned in this paper, such as data leakage attacks, e-coupon forgery and modification attacks, and message forgery and modification attacks. These attacks are only patterns rather than specific techniques. To be specific, each attack has many different possible approaches to implement, which are composed of a series of steps. For example, to simulate the eavesdropping attack, we need to first point out which technique an adversary will use to intercept the message. If we suppose the adversary uses data sniffing, then different kinds of sniffing methods should be considered, including application-level sniffing, LAN sniffing, TCP session stealing and so on. Different techniques and methods produce different results. Other attacks mentioned in this paper also have such problems.
Overall, to simulate all the threats, we need to implement all the attack techniques with all possible situations first. Due to the epidemic situation, it is difficult for us to carry out comprehensive experiments in a short time.
We have also done our best to investigate a large amount of related literatures. We could not find a proper and reasonable way to simulate these threats. Most of authors analyze the security theoretically and evaluate the overhead with simulations. Thus we also used the similar methods to evaluate these attacks. Thanks a lot for the reviewer's suggestions.

Corresponding contents: (Section 5.7)"We suppose that $\hat{\mathcal{A}}$ is an adversary. Considering the similarity of the attack principles and solutions in practice, we only choose some typical situations for deep analysis. For example, we only discuss the eavesdropping on messages sent from C to S because that from M to S is similar. The mechanism to defend against the typical attacks are described as follows.

1. *Eavesdropping and data leakage*: 1) Suppose $\hat{\mathcal{A}}$ can intercept the message $msg = ((\overline{SQR^1})_{pk_S} \parallel \sigma)$ sent from C to S. $\hat{\mathcal{A}}$ wants to get $\overline{SQR^1}$ or the identity of C from

msg. However, without sk_s, it is infeasible for $\hat{\mathcal{A}}$ to break the ciphertext $(\overline{SQR^1})_{pk_S}$, which is guaranteed by the hardness of the **Decisional Diffie-Hellman (DDH) Assumption**, as described in Section 5.2. The true identity f and the true credential (A, B, C) of C cannot be revealed with the anonymous signature σ. Thus, the data confidentiality and identity anonymity are provided to defend against the eavesdropping attack. 2) Furthermore, suppose $\hat{\mathcal{A}}$ can get the e-coupon $\overline{SQR^1}$ and wants to guess $\overline{SQR^2}$ and then recover $\overline{QR}$. Because of the pseudorandomness property of the QRFC approach, the probability of $\hat{\mathcal{A}}$ to succeed is only 0.5^n where n is the number of blocks in $\overline{QR}$. Details can be found in Section 5.5. 3) Even if $\hat{\mathcal{A}}$ gets $\overline{SQR^1}$ and $\overline{SQR^2}$ in the same time, he can open the commitment in $\overline{QR}$ with a negligible probability without the security parameter r, which is guaranteed by the Pedersen commitment as described in Section 5.3. Thus, the confidentiality of the sensitive information $data_0$ can be provided.

2. *E-coupon forgery and modification*: 1) Suppose $\hat{\mathcal{A}}$ gets the e-coupon $\overline{SQR^1}$ and tampers or forges it. However, he cannot generate a valid $h(QR^1)$ stored in $\overline{CQR}$ without *key*, which is guaranteed by the soundness of the **Merkle-Damgåd Structure**. As the first verification of EQRC, the details are introduced in Section 5.4. 2) Suppose $\hat{\mathcal{A}}$ gets the e-coupon $\overline{SQR^2}$ and tampers or forges it. The server can also figure it out by stacking $\overline{SQR^1}$ and $\overline{SQR^2}$. To be specific, anomaly in $\overline{CQR}$ appears according to the security properties of the QRFC approach when $\overline{SQR^1}$ passes the first verification, which means that $\overline{SQR^2}$ is modified or fake. Detailed analysis is shown in Section 5.6. Overall, the authenticity and integrity of e-coupons are provided in EQRC.

3. *Collusion attack*: Suppose $\hat{\mathcal{A}}$ is registered as a legal merchant. Suppose another adversary, $\hat{\hat{\mathcal{A}}}$, is a legal customer and has collusion with $\hat{\mathcal{A}}$. The advantage of them is that they have pairs of e-coupons $(\overline{SQR^1}, \overline{SQR^2})$ and know the rule to recover $\overline{QR}$. However, they can obtain $data_0$ from $\overline{QR}$ with a negligible probability which is guaranteed by the **Perfect Hiding Property** provided by the Pedersen commitment. Thus, it is infeasible to deploy the collusion attack on EQRC.

4. *Message forgery, modification and replay*: 1) Suppose $\hat{\mathcal{A}}$ can intercept the message *msg* sent from C to S and wants to modify or forge it. However, it is infeasible because *msg* is protected by encryption, whose security is guaranteed by the hardness of the **Decisional Diffie-Hellman (DDH) Assumption**, as described in Section 5.2. Any modification on *msg* can be detected by the signature σ. In addition, σ is constructed with the secret key sk_C and the credential (R, S, T). Thus, without sk_C and (A, B, C), $\hat{\mathcal{A}}$ cannot forge a valid anonymous signature due to the hardness of the **Blind Bilinear LRSW Assumption**. Details can be found in Section 5.1. With the mechanisms above, the authenticity and integrity of messages, and the identity anonymity are guaranteed in EQRC. 2) Suppose $\hat{\mathcal{A}}$ can intercept the message *msg* sent from C to S and wants to resend it to S. However, it is infeasible because the signature σ

is constructed with a timestamp n_t as shown in Algorithm 2. That is, the replay attack can be detected by checking n_t and the time threshold, which is guaranteed by the hardness of the **Blind Bilinear LRSW Assumption**. Therefore, EQRC can effectively resist the attack of message replay.

"

Reviewer Point P 1.2 — *In the description of Figure 5, it would be helpful to give detail explanation why at the computing phase of the QR code version has a significant effect on running time overhead.*

Reply: Thanks for the helpful suggestion. We have revised the corresponding paragraphs to explain Fig. 5 clearer. Compared with the preprocessing stage, the overhead for the computing stage is trivial. The reason can be described intuitively: the handling of pictures is generally slower than computation on given arrays or integers in the experiments. More details are given on page 19.

Corresponding contents: (Section 6.2) "The preprocessing stage includes the processes such as reading the QR codes and building Bitmap objects with C#. Thus, we can see that the version of QR codes has little effect on the preprocessing time but the size of QR codes affects a lot. In contrast, the computing is on blocks instead of pixels, as described in Section 4.2.2. Thus, as shown in Fig. 5(b), the version of QR codes has effect on the computing time but the size does not. Compared with the preprocessing stage (around 40 to 380 ms in the experiments), the overhead for the computing stage (less than 6 ms in the experiments) is trivial. The reason can be described intuitively: the handling of pictures is slower than computation on given arrays or integers in the experiments. Overall, the encoding time is mainly determined by the preprocessing stage, which is the same for all schemes regardless security. "

Reviewer Point P 1.3 — *Further Discussion in Figure 6, it would be helpful to explain why the decoding process more efficient.*

Reply: Thanks for this point. Further discussion has been added to the corresponding paragraph. In short, pseudo-random numbers should be generated for each block in the encoding process but only a modulo-2 addition in the decoding process. Thus, the decoding process is more efficient, which is a beneficial property for e-coupon transactions.

Corresponding contents: (Section 6.2) "On the other hand, the decoding process for QRFC also contains two parts: the preprocessing stage and the computing stage. The overhead of the preprocessing stage is shown in Fig. 6(a). The overhead of the computing stage which includes the processes of decoding computation and original QR codes recovery is shown in Fig. 6(b). It is clear that the computing stage (less than 0.25 ms in the experiments) costs less time than the preprocessing stage (about 13 to 35 ms in the experiments). Comparing the decoding process with the encoding process, we can find that the decoding

process is more efficient, costing less time than the encoding process. The result is reasonable. As described in Section 4.2.2, pseudo-random numbers are generated for each block to split a QR code in the encoding process. However, only a modulo-2 addition operation is necessary for each block in the decoding process due to the attribute of QRFC. The quicker decoding is a beneficial property for e-coupon transactions, because users concern more about the successful transaction time than the generation time of e-coupons."

Reviewer Point P 1.4 — *The statements in figures 9 and 10 describe the end to end delay for EQRC-I or EQRC- II or both of them?*

Reply: We are sorry for the misunderstanding we made to readers. Thanks a lot for pointing this out. EQRC-I and EQRC-II are not simulated separately for Fig. 9 and Fig. 10. Because we only consider the end-to-end process from users, i.e., merchants or customers, to a server. The only difference for EQRC-I and EQRC-II in an end-to-end communication is the possible length of messages. Thanks for the question. We have added sentences in Section 6.3 to make it clearer for readers.

Corresponding contents: (Section 6.3) "To evaluate the communication overhead, we implemented a simulation on the end-to-end delay from users to a server. Because the transactions in both EQRC-I and EQRC-II are composed of several communication rounds and every rounds are similar, the experiment focuses on one round is reasonable and representative."
"Note that we only consider the end-to-end process from users, i.e., merchants or customers, to a server. The only difference for EQRC-I and EQRC-II is the possible length of messages. That is why we choose 4-KB messages in this experiment, which is probable for both EQRC-I and EQRC-II as shown in Table 5."

Reviewer 2

We are grateful to the reviewer for the detailed and insightful comments.

Reviewer Point P 2.1 — *The problem addressed by the paper is interesting and it is of practical interest. The proposed solution seems reasonable and effective in practice. However, the paper proposes no new theoretical contributions or insights and focuses only on solving the e-coupon problem relying on well established mechanisms and tools from the literature. The proposed solution seems to come out only by a careful application of well known cryptographic techniques. It is not clear to me if the proposed solution could be generalized and used in other settings. Please, add a discussion explaining whether the proposed approach could be generalized and applied to solve other similar problems.*

Reply: This is a great point. Thanks for your valuable suggestions. We have discussed the corresponding contents in a new section, Section 7. The main contributions of this work are the enhanced QR code scheme and the secure e-coupon transaction framework EQRC.

Although this work is proposed to address the security problems for e-coupon transactions, it can also be extended and generalized. Some possible applications, including payment cards for relatives, certificates and health barcodes, are described in detail on page 25. In addition, we have discussed the remaining problems and open questions in our work.

Corresponding contents: (Section 7) "EQRC has the ability to satisfy the essential security requirements, including data confidentiality, authentication, integrity and anonymity. Although it is proposed for e-coupon transactions, EQRC can be generalized for other scenarios easily. Some possible applications are described as follows.

1. *Generalized e-coupons*: The proposed enhanced QR codes can be used as the generalized e-coupons, i.e., not only the common coupons with discounts, but also the cash coupons, gift cards, pre-paid membership cards, and rechargeable cards. The only task is to update the QR codes after redemption. For example, the server can update the amount of money in a rechargeable card after consumption or recharging by issuing a new pair of $(\overline{SQR^1}, \overline{SQR^2})$.

2. *Payment cards for relatives*: Considering users are family members, we can apply the proposed framework to payment tasks. For example, the parents and their child hold one of the enhanced QR code pair $(\overline{SQR^1}, \overline{SQR^2})$, respectively, as fingerprints for transactions. To complete the payment, i.e., to recover $\overline{QR}$, the server needs to collect both of the QR codes. Thus, the child must get permission from the parents when paying. The fingerprints can defend against attacks such as forgery and modification by the triple-verification. As an additional payment method, it protects relatives, especially children and elders, from fraud and economic losses.

3. *Certificates*: The enhanced QR codes can also be used as an aided verification of certificates, or even certificates directly. Compared with classic certificates, enhanced QR codes have more advantages. For example, a paper certificate only with a stamp is easy to forge. However, the necessary information such as the name of the awardee and the awards can be stored in the enhanced QR codes, which satisfy the requirements of integrity and authenticity. In addition, the holders of certificates may not trust the issuing institutions. With the enhanced QR codes, each of the holder and the institution only keep one of $(\overline{SQR^1}, \overline{SQR^2})$, respectively. Any tampering or forgery can be figured out and traced by the server.

4. *Health barcodes*: As a special case of certificates, health barcodes can be well used when citizens are exposed to infectious disease seriously, such as COVID-19. People with different health codes have different access to community activities. The design of enhanced QR codes and the triple verification guarantee that it is infeasible to tamper or forge the health barcodes. It provides a social safety guarantee in difficult time.

Although EQRC satisfies the security goals with good performance, there are still some remaining problems. Some of the problems are open questions in related research. The detailed discussions are as follows.

1. *The embedding position*: Based on the Reed-Solomon error correction code, $\overline{QR^1}$ and $\overline{QR^2}$ can be embedded into $\overline{CQR}$. However, the process should be considered more carefully because there are functional elements in a QR code. These functional elements ensure that a scanning device can correctly identify and decode QR codes. For example, the three position patterns placed at corners are used to define the location of the QR code. Thus, the embedding must avoid destroying the functional elements.
   Because the elements have fixed shapes, colors and positions in standard QR codes, the detection is not a problem. In fact, the elements are first detected after scanning, easily, quickly and accurately [46]. In addition, a recently developed QR code called Frame QR has a region where the arbitrary altering of figures and contents will not affect other regions. Combining Frame QR into the design of our EQRC can make the generation and merging of enhanced QR codes more standardized and concise.

2. *The size of enhanced QR codes*: Comparing with some classic e-coupon systems, which only use a bit string as e-coupons, EQRC has larger e-coupons in size. This is also a problem for other QR code-based innovative applications [9,11].
   Considering the network performance nowadays, it is not a significant limitation. A possible solution is to increase the capability of QR codes, which is a hot topic for researchers. With higher capability, a smaller QR code can be used for coding the same amount of data. One mature scheme is the colored QR codes. In EQRC, using colored QR codes to implement the fragment coding of QR codes, can provide more storage space. Thus, it will be an attractive research direction for us in the future.

3. *Copies on QR codes*: An illegitimate copy on e-coupons has the potential to infringe the rights of legitimate holders. It is an open question for QR codes for a long time. Because QR codes are usually used on smartphones, the risk of illegal copies by capturing the screen or taking pictures is high [47], We cannot ensure that users' phones are not accessible from attackers. Thus, it is hard to avoid copies.
   If the QR code is linked with the identity of the legal holder, the problem can be solved by checking the signature. From this point of view, EQRC can be extended to e-coupon services provided for particular users. Another possible solution is to check the freshness. For example, Alipay refreshes the payment QR codes every minute. However, both of the solutions above cannot solve the problem thoroughly.

4. *Privacy issues*: Although EQRC provides the identity anonymity and audit trail for users by the TAA scheme, there are many other privacy issues in practice. For example, 1) Once a user is determined dishonest, all the actions of the user in previous transactions are expected to be revealed. Some latest oblivious transfer schemes can be considered [19]. 2) User behaviors are usually studied for improving economic efficiency. However, how to protect the user's privacy while aggregating the statistical information is also a problem. One possible solution is the differential privacy, which withholds the information of individuals while revealing the patterns of groups.

5. *Portability*: As EQRC, most of the work on e-coupon systems is self-contained. How-

ever, a problem is whether the protocols can be successfully integrated with existing mature applications, such as Walmart, Amazon, Paypal, Aliexpress and Alipay. The difficulty of the work is that, interfaces of many mature shopping and payment applications are not public for researchers. In addition, to make the proposed framework user-friendly and practical, many details should be considered more carefully, including processing interruptions and communication interruptions. Strengthened cooperation between academic and industrial communities may provide more development space.

"

Reviewer Point P 2.2 — *At the moment Section 2 is quite heterogeneous: it contains related work, the system model and some background on the used cryptographic tools. Please, consider to split this section in such a way its scope is well defined. A possible approach would be add a new section for related work just before conclusion, and move the system model and the security goals just before Section 3.*

Reply: Thanks for the good suggestion. We have rearranged the sections. The related work has been moved to a new section, as Section 2. Because a new section, Section 7, has been added, and we also need to introduce the schemes to be compared with beforehand, we did not move the related work before the conclusion. The system model and the security goals have been moved just before Section 4.

Corresponding contents: (Section 2, Section 3, Section 7)

Reviewer Point P 2.3 — *there are some proposal for using cryptography with QR code, e.g., in Riccardo Focardi, Flaminia L. Luccio, Heider A.M. Wahsheh, Usable security for QR code, Journal of Information Security and Applications, Volume 48, 2019. Please, add a comparing between your proposal and those proposed in the literature.*

Reply: We agree that it is an important point for this paper. We appreciate the reviewer's professional suggestions. The suggested paper [2] did a good work on analyzing the security of QR codes. We have quoted it as a strong argument in Section 1. Considering the comparability, we have compared our work with other frameworks [12, 13], which focus on secure e-coupon transactions. Security properties, computation overhead and communication overhead are compared in two tables. Thus, we believe the comparison is meaningful and reasonable. For the reviewer's convenience, we put the two tables in the reply, Table 6 and 7, which are the same in the manuscript. The detailed explanation is given in Section 6.4.

Corresponding contents: (Section 6.4) "In Table 6, EQRC is compared with two recent research papers published in 2014 and 2015 [12,13]. All of the schemes focus on satisfying the security requirements for the e-coupon system, including authentication and integrity. The difference is as follows. The anonymity of users in EQRC is protected by the private key of CA, $isk(x, y)$, as described in Section 5.1, so that it is infeasible to link the true identity with a signature for C, M and S. However, in Framework-A [12], the anonymity is

Table 6: Comparisons of the Secure E-coupon Frameworks

Framework	Anonymity†	Verification‡	Privacy§	Design ¶	Message #	Rounds ††
Framework-A [12]	sk_S	sk_S	a session key	`Out`	e-coupons & `sig`	3
Framework-B [13]	✗	sk_S	✗	`Out`	e-coupons & `MAC`	2
EQRC-I	$isk(x, y)$	triple-verification	triple-protection	`Out` & `In`	`cip` & `sig`	4
EQRC-II	$isk(x, y)$	triple-verification	triple-protection	`Out` & `In`	`cip` & `sig`	3

† The anonymity of users, i.e., merchants and customers. ‡ The verification of e-coupons. § The privacy protection for e-coupons. ¶ The design for security. `Out` and `In` denote outside and inside e-coupons. # The message that sent to redeem e-coupons. `sig` is used to denote signatures, `MAC` for message authentication codes, and `cip` for ciphertext of e-coupons. For more details, we refer readers to the related papers. †† The number of communication rounds for e-coupons redemption and verification.

guaranteed by the server's private key sk_S, which means that the identity is exposed to the server. Framework-B [13] even does not hold such property. Besides, both Framework-A and Framework-B verify e-coupons via a PKI-based solution. To be specific, the authenticity and integrity of e-coupons are protected by the server's private key. In EQRC, to provide stronger security, the triple-verification is presented. More details can be found in Section 5.6.

The detailed computation overhead is compared in Table 7. The result shows that merchants and customers in EQRC only have the work necessary for communication, including signing and encrypting the message while most of the work is on the server. Although Framework-B also has fewer cryptography operations on the user side, it does not provide enough and strong security properties as EQRC. Overall, EQRC is secure and more friendly to users. The advantage is guaranteed by the design of the framework, as shown in Table 6. To be specific, our scheme applies both outside and inside the e-coupons, which means that we proposed not only how e-coupons are wrapped with cryptography techniques (such as the anonymous signature), but also the components of e-coupons for security (i.e., the generation of the enhanced QR codes). Thus, users have much less computation overhead than the cloud server. However, the techniques are adopted outside of e-coupons directly for security in Framework-A and Framework-B.

As for communication, in EQRC-I and EQRC-II, the numbers of communication rounds for e-coupon redemption and verification are slightly larger than that in Framework-A and Framework-B. It is reasonable because each of the customer and the merchant only holds one of the pair $(\overline{SQR}^1, \overline{SQR}^2)$ while both are necessary for verification. Though we likely have longer messages due to the QR-codes we use in the scheme, it is still acceptable and can be reduced via coding algorithms."

Reviewer Point P 2.4 — *An e-coupon is represented by a pair of QR code, you mention it throughout the introduction and in the system model section. However, you never say it explicitly. Please, add a sentence to make it clear.*

Reply: Thanks a lot for pointing it out. We have revised the corresponding sentences to make it clearer in the paper as suggested.

Corresponding contents: (Section 1) "The proposed enhanced QR codes can be used as e-coupons."

Table 7: Comparisons on Computation Overhead

Framework	Phase†	Entity	Operation‡							
			`Commit`	`Cod`	`Spl`	`Sta`	`Enc`	`Ch`	`Hash`	`Commu`
Framework-A [12]	Issuing	customer	0	0	0	0	0	0	0	*
		merchant	0	0	0	0	2	3	2	
		server	0	0	0	0	4	2	0	
	Downloading & using	customer	0	0	0	0	2	3	2	
		merchant	0	0	0	0	1	0	0	
		server	0	0	0	0	3	2	0	
	Total	customer	0	0	0	0	2	3	2	
		merchant	0	0	0	0	3	3	2	
		server	0	0	0	0	7	4	0	
Framework-B [13]	Issuing & distribution	customer	0	0	0	0	0	0	0	*
		merchant	0	0	0	0	0	0	0	
		server	0	0	0	0	1	0	1	
	Using & verification	customer	0	0	0	0	0	0	0	
		merchant	0	0	0	0	1	0	1	
		server	0	0	0	0	0	0	0	
	Total	customer	0	0	0	0	0	0	0	
		merchant	0	0	0	0	1	0	1	
		server	0	0	0	0	1	0	1	
EQRC-I & -II	Generation	customer	0	0	0	0	0	0	0	0
		merchant	0	0	0	0	0	0	0	0
		server	1	1	1	2	0	0	0	0
	Distribution	customer	0	0	0	0	0	0	0	0
		merchant	0	0	0	0	0	0	0	0
		server	0	0	0	0	0	0	0	2
	Using & verification	customer	0	0	0	0	0	0	0	1
		merchant	0	0	0	0	0	0	0	1
		server	1	2	0	1	0	0	1	2
	Total	customer	0	0	0	0	0	0	0	1
		merchant	0	0	0	0	0	0	0	1
		server	2	3	1	3	0	0	1	4

† We keep the original terms used in the related papers so that readers can find and understand them easily. Note that although the names in frameworks are not exactly the same, the processes are matched. To be specific, we are considering the phases from preparing e-coupons to using it successfully. Registration is not involved. ‡ The operations are commitment, QR codes encoding or decoding, splitting, stacking, encrypting, chaotic mapping, hash, and necessary operations (i.e., encrypting and signing messages) for communication from left to right. Note that, we consider `Commu` as a whole without splitting it, because it is the common operations on the messages sent.

* Not mentioned in the references.

(Section 4.2) "With this scheme, a standard QR code is processed to finally generate a pair of enhanced QR codes ($\overline{SQR^1}$, $\overline{SQR^2}$), which can be seen as two fragments of an e-coupon." (Section 4.2.3) "To emphasize that the pair of $\{\overline{SQR^1}, \overline{SQR^2}\}$ is used as e-coupons in EQRC."

Reviewer Point P 2.5 — *Section 2.4 presents the cryptographic tools used in the paper. At the moment I believe that this presentation is very concise. Please, consider to add further explanations.*

Reply: Thanks a lot for the professional comments. We have explained the tools further with new contents and improved the presentation to make it clearer. The contents now can be found in Section 3.1 due to the rearrangement of the sections. Considering the complexity of the techniques mentioned in this paper, we only added the necessary details for conciseness. We hope the revision has improved the quality of this section.

Corresponding contents: (Section 3.1)

Reviewer Point P 2.6 — *At page 5 line 21, what is G_p? Please, clarify.*

Reply: Thanks for the question. In the Pedersen commitment, p and q denote large primes. q divides $p-1$. $\mathbb{G}_q$ is the unique subgroup of $\mathbb{Z}_p^*$ of order q. The corresponding explanation has been added to page 4.

Corresponding contents: (Section 3.1) "The protocols of the Pedersen commitment are as follows, 1) p and q denote large primes. q divides $p-1$. $\mathbb{G}_q$ is the unique subgroup of $\mathbb{Z}_p^*$ of order q. g is a generator of $\mathbb{G}_q$. *open* is set as $r \in \mathbb{Z}_q$. 2) The commitment to a value $m \in \mathbb{Z}_q$ can be defined as $com = \texttt{Comm}(m, r) = g^m h^r$ where h is an element of $\mathbb{G}_q$ such that only the receiver knows $\log_g h$. 3) The receiver can check if $\texttt{Opnv}(com, m, r) = true$ by re-computation when m and r are both revealed. "

Reviewer Point P 2.7 — *at page 5 line 8, the typesetting of word like "Setup", "Join", etc. contains an unnecessary space. Please, consider to use the \mathit macro. This happens also in other parts of the paper, thus, consider to fix it.*

Reply: Thanks a lot for the useful tip. We have fixed all the problems throughout the paper.

Corresponding contents: (Throughout the whole paper. For example, in Section 3.1.)

Reviewer Point P 2.8 — *Please, consider to add further explanation to Table 2.*

Reply: Thanks for the suggestion. We have explained it clearer and given examples as well for easy understanding. Details are added on page 10.

Corresponding contents: (Section 4.2.2) "We choose $k = 4$ in our framework. The construction method is shown in Table 2. For example, if one block in $\overline{QR}$ is white, the

corresponding bit strings in $\overline{QR^1}$ and $\overline{QR^2}$ can be 1001 and 0110, respectively (or 0110 and 1001, respectively). To recover the white block, we only need to compute $1001 + 0110(\mathsf{mod}\ 2) = 1111$ (or $0110 + 1001(\mathsf{mod}\ 2) = 1111$). It is clear that there are two choices for each block to split. "

Reviewer Point P 2.9 — *Please, in Section 4 make explicit during the security analysis you carry out how the security goals are achieved.*

Reply: Thanks a lot for the professional suggestion. Besides analyzing the security properties according to the protocol flow of EQRC, we have given details about the defense against the attacks. We illustrate some specific situations to prove that the proposed framework can achieve the security goals mentioned in Section 3.3. A new subsection, Section 5.7, has been added.

Corresponding contents: (Section 5.7)"We suppose that $\hat{\mathcal{A}}$ is an adversary. Considering the similarity of the attack principles and solutions in practice, we only choose some typical situations for deep analysis. For example, we only discuss the eavesdropping on messages sent from C to S because that from M to S is similar. The mechanism to defend against the typical attacks are described as follows.

1. *Eavesdropping and data leakage*: 1) Suppose $\hat{\mathcal{A}}$ can intercept the message $msg = ((\overline{SQR^1})_{pk_S} \parallel \sigma)$ sent from C to S. $\hat{\mathcal{A}}$ wants to get $\overline{SQR^1}$ or the identity of C from msg. However, without sk_s, it is infeasible for $\hat{\mathcal{A}}$ to break the ciphertext $(\overline{SQR^1})_{pk_S}$, which is guaranteed by the hardness of the **Decisional Diffie-Hellman (DDH) Assumption**, as described in Section 5.2. The true identity f and the true credential (A, B, C) of C cannot be revealed with the anonymous signature σ. Thus, the data confidentiality and identity anonymity are provided to defend against the eavesdropping attack. 2) Furthermore, suppose $\hat{\mathcal{A}}$ can get the e-coupon $\overline{SQR^1}$ and wants to guess $\overline{SQR^2}$ and then recover $\overline{QR}$. Because of the pseudorandomness property of the QRFC approach, the probability of $\hat{\mathcal{A}}$ to succeed is only 0.5^n where n is the number of blocks in $\overline{QR}$. Details can be found in Section 5.5. 3) Even if $\hat{\mathcal{A}}$ gets $\overline{SQR^1}$ and $\overline{SQR^2}$ in the same time, he can open the commitment in $\overline{QR}$ with a negligible probability without the security parameter r, which is guaranteed by the Pedersen commitment as described in Section 5.3. Thus, the confidentiality of the sensitive information $data_0$ can be provided.

2. *E-coupon forgery and modification*: 1) Suppose $\hat{\mathcal{A}}$ gets the e-coupon $\overline{SQR^1}$ and tampers or forges it. However, he cannot generate a valid $h(QR^1)$ stored in $\overline{CQR}$ without key, which is guaranteed by the soundness of the **Merkle-Damgåd Structure**. As the first verification of EQRC, the details are introduced in Section 5.4. 2) Suppose $\hat{\mathcal{A}}$ gets the e-coupon $\overline{SQR^2}$ and tampers or forges it. The server can also figure it out by stacking $\overline{SQR^1}$ and $\overline{SQR^2}$. To be specific, anomaly in $\overline{CQR}$ appears according to the security properties of the QRFC approach when $\overline{SQR^1}$ passes the first verification,

which means that $\overline{SQR^2}$ is modified or fake. Detailed analysis is shown in Section 5.6. Overall, the authenticity and integrity of e-coupons are provided in EQRC.

3. *Collusion attack*: Suppose $\hat{\mathcal{A}}$ is registered as a legal merchant. Suppose another adversary, $\hat{\hat{\mathcal{A}}}$, is a legal customer and has collusion with $\hat{\mathcal{A}}$. The advantage of them is that they have pairs of e-coupons $(\overline{SQR^1}, \overline{SQR^2})$ and know the rule to recover $\overline{QR}$. However, they can obtain $data_0$ from $\overline{QR}$ with a negligible probability which is guaranteed by the **Perfect Hiding Property** provided by the Pedersen commitment. Thus, it is infeasible to deploy the collusion attack on EQRC.

4. *Message forgery, modification and replay*: 1) Suppose $\hat{\mathcal{A}}$ can intercept the message msg sent from C to S and wants to modify or forge it. However, it is infeasible because msg is protected by encryption, whose security is guaranteed by the hardness of the **Decisional Diffie-Hellman (DDH) Assumption**, as described in Section 5.2. Any modification on msg can be detected by the signature σ. In addition, σ is constructed with the secret key sk_C and the credential (R, S, T). Thus, without sk_C and (A, B, C), $\hat{\mathcal{A}}$ cannot forge a valid anonymous signature due to the hardness of the **Blind Bilinear LRSW Assumption**. Details can be found in Section 5.1. With the mechanisms above, the authenticity and integrity of messages, and the identity anonymity are guaranteed in EQRC. 2) Suppose $\hat{\mathcal{A}}$ can intercept the message msg sent from C to S and wants to resend it to S. However, it is infeasible because the signature σ is constructed with a timestamp n_t as shown in Algorithm 2. That is, the replay attack can be detected by checking n_t and the time threshold, which is guaranteed by the hardness of the **Blind Bilinear LRSW Assumption**. Therefore, EQRC can effectively resist the attack of message replay.

"

Reviewer Point P 2.10 — *please, consider to make publicly available the prototype used for the experimental evaluation.*

Reply: Thanks for the suggestion. We have put all the code we used publicly on Github. Please see https://github.com/RuiLiu-Uvic/EQRC. We hope the resources can help other researchers in the future.

Corresponding contents: https://github.com/RuiLiu-Uvic/EQRC

Reviewer 3

We would like to express our sincere appreciation to the reviewer for the time and efforts in reviewing our paper.

Reviewer Point P 3.1 — *This paper tries to enhance QR code and its applications in various ways including enhanced QR coding, signing, verification and recovery etc. The*

conference version was accepted already. This journal version enhanced it by making the following improvements:

1. It proposes an offline secure e-coupon transaction framework. 2. They provide stronger security analyses on the digital signature, encryption, commitment, HMAC, QRFC approach, and triple verification. 3. To evaluate the performance further, they conduct a comprehensive simulation to analyze the communication overhead on OMNeT++ 5.5.

This paper is well organized. It includes enough background knowledge to understand the paper such as the current development of QR codes and its properties. They also show the security incidents toward the widely used QR codes.

The security analysis is performed from six aspects: Security of the Digital Signature, Security of the Encryption, Security of the Commitment, Security of the HMAC, Security of the QRFC Approach and Security of the Triple Verification. Based on my understanding, the analysis is sound and reasonable. A lot more evaluations are also added for both the old design and also newly added offline e-coupon transaction framework.

To me, this paper includes 1/3 new materials compared with their conference version, and also the newly added materials is highly associated with the original contents with similarity quality.

Reply: Thanks for the time and kind review. We have added some new contents to the manuscript. The major changes are as follows: 1) We have introduced new research development in Section 2; 2) We have further explained the techniques in Section 3.1; 3) The performance of defending against the typical attacks has been analyzed in Section 5.7; 4) We have compared our work with related protocols in Section 6.4; 5) We have further discussed the generalization, remaining problems and possible solutions in Section 7. In addition, we have revised the presentation and corrected the spelling errors. We hope the quality of the paper has been improved further.

References

[2] R. Focardi, F.L. Luccio and H.A. Wahsheh, Usable security for QR code, *Journal of Information Security and Applications* **48** (2019), 102369.

[12] C.-C. Chang and C.-Y. Sun, A secure and efficient authentication scheme for e-coupon systems, *Wireless Personal Communications* **77**(4) (2014), 2981–2996.

[13] C.-C. Chang, I.-C. Lin and Y.-L. Chi, Secure electronic coupons, in: *2015 10th Asia Joint Conference on Information Security*, IEEE, 2015, pp. 104–109.

Journal of Computer Security 0 (0) 1
IOS Press

EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions [1]

Rui Liu [a], Jun Song [b,*], Zhiming Huang [a] and Jianping Pan [a]
[a] *Department of Computer Science, University or Victoria, Victoria, Canada*
[b] *School of Computer Science, China University of Geosciences, Wuhan, Hubei, China*

Abstract. In recent years, with the rapid development and popularization of e-commerce, the applications of e-coupons have become a market trend. As a typical bar code technique, QR codes can be well adopted in e-coupon-based payment services. However, there are many security threats to QR codes, including the QR code tempering, forgery, privacy information leakage and so on. To address these security problems for real situations, in this paper, we introduce a novel fragment coding-based approach for QR codes using the idea of visual cryptography. Then, we propose a QR code scheme with high security by combining the fragment coding with the commitment technique. Finally, an enhanced QR code-based secure e-coupon transaction framework is presented, which has a triple-verification feature and supports both online and offline scenarios. The following properties are provided: high information confidentiality, difficult to tamper with and forge, and the ability to resist against collusion attacks. Furthermore, the performance evaluation of computing and communication overhead is given to show the efficiency of the proposed framework.

Keywords: QR code, e-coupon, e-commerce, commitment algorithms, visual cryptography.

1. Introduction

With the rapid development of electronic payment systems and digital marketing, e-coupons (Electronic Coupons) have become increasingly popular. Because e-coupons are easy to manage, quick to distribute, and eco-friendly, they are widely accepted as a replacement of paper coupons by many companies, such as Shoppers Drug Mart, McDonald's, Air Canada and so on. Besides, the various types of e-coupons, including but not limited to discount coupons, cash coupons, pre-paid cards and rechargeable cards, are appropriate for different uses in the market. The demands for e-coupon services include easy generation, fast readability, large storage capacity, error recovery and so on. QR (Quick Response) codes support the above properties well and thus are universally preferred.

Although QR codes have been widely used in many domains, such as mobile payment, document verification, commodity management, inventory checking, parcel tracking and so on, security incidents still occur frequently and the situation becomes increasingly serious. Economic losses and privacy leak which are caused by scanning malicious QR codes are reported many times. Focardi *et al.* [2] surveyed attacks on QR codes including phishing, barcode-in-barcode attacks, cross-site scripting attacks and so

[1] A preliminary version of this paper appeared in 2019 IEEE International Conference on Communications (ICC) [1].
[*] Corresponding author. E-mail: songjun@cug.edu.cn.

on. Cadger *et al.* [3] analyzed 12 different software packages which can decode QR codes, and found that none of them has the ability to detect a tampered QR code. Besides, scanning QR codes with sensitive personal information, such as tickets, payment codes and so on, poses a dramatic threat to the privacy of users. The risks above are mainly due to the open coding scheme of QR codes, plaintext format of the content in QR codes, and lack of verification mechanisms. Without proper measures and solutions, QR codes cannot be used in online transactions safely.

The studies on the security of QR codes are increasingly hot in recent years. Many works focus on the anti-phishing of QR codes with the techniques of link detection, digital signature and so on [4–6]. There are also some works using cryptography and steganography to provide the confidentiality of QR codes [7, 8]. However, due to the specific threats of e-coupon services, *e.g.*, the collusion attack between merchants and customers, tempering or forgery of e-coupons, and liability disputes among users, these research results cannot be applied well to the QR code-based e-coupon transaction services.

To address the above concerns, in this paper, we propose an enhanced QR code and triple-verification-based secure e-coupon transaction framework EQRC. It mainly aims at the common security risks such as the plaintext transmission, collusion attack, forgery and tampering in e-coupon services. Using encryption and scrambling, anonymous authentication, and commitment, EQRC has the ability to ensure data confidentiality and provide anti-tampering, anti-forgery, signature verification for both online and offline scenarios. The main work and contributions of this paper include the following four aspects:

(1) We presented a fragment coding-based approach for QR codes, which is based on the idea of visual cryptography. Due to the pseudo-randomness of fragments, it is hard to guess the true information of the original QR code from one of the split code pair. Thus, the safety of QR codes is enhanced efficiently as the attacker would be more difficult to access the original QR code.
(2) We presented an enhanced QR code scheme with a higher security, which is inspired by the fragment coding and commitment technique. This scheme not only has the ability to prevent the leakage of sensitive information in QR codes, but also can effectively reduce the security risks caused by QR code tampering or forgery. The proposed enhanced QR codes can be used as e-coupons.
(3) We presented a secure online e-coupon transaction framework EQRC, which relies on the enhanced QR codes with digital signature and commitment. EQRC provides three verifications, i.e., message digest computing, enhanced QR codes stacking and commitment opening. Due to these three verifications, our framework can provide integrity and authenticity for e-coupons.
(4) We extended EQRC from online to offline. In both of the scenarios, users can use e-coupons with the same security and privacy properties provided. The collusion attack between merchants and customers can be resisted effectively. Furthermore, in the offline scenario, liability disputes can be settled with an audit trail.

In addition, we analyzed and proved the security of EQRC based on cryptographic assumptions, mathematical properties, and potential attacks. The comprehensive evaluations of the computation overhead, encoding and decoding overhead, and communication overhead are provided. We also compared our work with other related secure e-coupon protocols. Results show that the proposed framework has a good performance in security and efficiency.

In the rest of this paper, the related work is shown in Section 2. Section 3 briefly introduces related cryptographic techniques, the system model, security goals and threats. Section 4 gives the detailed description of the schemes proposed in this paper. The security and performance evaluations are presented in Section 5 and 6, respectively. Section 7 discusses the possible generalization, remaining problems and potential solutions of the proposed framework. Section 8 concludes the paper.

2. Related Work

E-coupon systems have been studied for many years. To solve the problem of targeted marketing in e-coupon services, Yan *et al.* [9] extracted features from user's behavior records, and proposed a complex model based on random forest and extreme gradient boosting. Based on an oblivious transfer model and a blind signature scheme, Liu *et al.* [10] proposed a system that focuses on user privacy. In this work, a dishonest user will lose the redemption privacy when he redeems an e-coupon several times. He *et al.* [11] studied users' behaviors to predict coupon usage probabilities. The authors explored 34 factors that impact the probability and ranked them according to the effects. These works are from different perspectives. However, the more important thing is to ensure the security of e-coupons.

To satisfy the essential security requirements of e-coupons, several systems have been proposed. In 2014, Chang *et al.* [12] presented an e-coupon authentication scheme based on chaotic maps. The scheme satisfies anonymity, mutual authentication and privacy protection. In 2015, two schemes that allow servers to issue different types of e-coupons were proposed [13]. Both the two schemes in [13] target on providing authentication and integrity for e-coupon transactions and preventing the reuse of e-coupons. In the first scheme, e-coupons are not issued to specific customers. That is different in the second scheme which focuses on keeping loyal customers. In Section 6.4, the first scheme [13], the chaotic map-based scheme [12] and our framework are compared.

In our work, we focus on adopting QR codes in e-coupon services while solving the typical security problems, including the data leakage, e-coupon forgery and modification, authentication and privacy.

To protecting the content of QR codes, some novel schemes have been proposed these years. Cheng *et al.* [14] proposed an innovative secret sharing scheme for QR code applications. The proposed scheme uses the XVCS (XOR-based Visual Cryptography Scheme) theory and has more flexible access structures. Lin *et al.* [15] proposed a secret hiding mechanism based on QR code error correction with a high capacity. Tkachenko *et al.* [16] proposed a two-layer QR code-based scheme for sharing secret messages, which replaces the black blocks in the traditional QR code with a specific pattern. Although these strategies can effectively protect the contents of QR codes to some extent, they cannot be adopted well in e-coupon transactions where both malicious merchants and customers should be considered.

To solve the problem of forgery and tampering, some effective techniques have been presented. Zhang *et al.* [17] proposed a message authentication scheme with the help of roadside units for vehicular communications. The scheme has a better performance than previous work in message loss ratio and delay. Considering the large amount of information generated in vehicular networks at the same time, Lee and Lai [18] presented a secure batch verification scheme based on bilinear pairing. Hasan *et al.* [19] designed a secret information verification mechanism based on an authentication chain. Although this mechanism has the characteristic of traceability and anti-counterfeiting, it is not suitable for QR code services as keeping a chain for each QR code is space-consuming.

As for the anonymous digital signature technique which is often adopted for authentication, many related studies have been reported. Brickell *et al.* [20] proposed the direct anonymous attestation (DAA) in 2004. Based on zero-knowledge proof and the idea of group signature, users in DAA can obtain identity-anonymous certificates without revealing privacy information. Chen *et al.* [21, 22] proposed a pairing-based DAA protocol in the asymmetric setting in 2009 and proposed a threshold anonymous authentication (TAA) scheme for vehicular ad-hoc network (VANET) in 2011, which is adopted in our paper for e-coupon services.

3. Preliminaries

3.1. Cryptographic Techniques

In this section, we briefly introduce the cryptographic tools adopted in our framework.

(1) *Group Signature*: Group signature allows a user of a group to sign a message anonymously, i.e., a verifier can only tell whether the signer is a group member. With a group signature scheme, a signature σ to a message m can be generated by $\texttt{Sig}_{(sk,pk)}(m)$ where sk is the secret key of the signer and pk is the public key of the group. σ can be verified by $\texttt{Ver}_{pk}(\sigma, m)$. A third party named revocation manager can be involved in some schemes. It can trace the signature and determines the identity of the signer. Without the secret key of the revocation manager, given a message m and its signature σ, it is not feasible to find out the identity of the individual signer. Besides, only a member in the group can generate a valid signature. With these properties, group signature can be well used in anonymous authentication.
In this paper, we adopt an effective group signature-based authentication scheme TAA [22] and adapt it to e-coupon services. TAA was proposed for VANET (Vehicle Ad-hoc NETworks) communication. It achieves the reliability, privacy and auditability with direct anonymous attestation and one-time anonymous authentication. The main algorithms of TAA are as follows: 1) *Setup*. Inputting an integer t, the algorithm creates system parameters and long-term keys for trusted authorities. 2) *Join*. Every legitimate user can obtain credentials (A, B, C) and membership secrets f. 3) *Sign*. With the group public parameters, (A, B, C), f and a message msg, the algorithm outputs an anonymous signature σ. 4) *Verify*. σ can be verified with msg by the algorithm. There are also other algorithms, such as $ThresholdCheck$, *Link* and *Disavow*, which provide more properties in VANETs but are not adopted in our work.

(2) *Commitment*: Commitment is a basic cryptographic tool that is usually used in zero-knowledge proofs, contract signing, e-voting, secret sharing, secure coin flipping, secure computation and so on. It allows one to keep the sensitive information hidden to others while maintaining the ability to reveal it. A commitment protocol is a two-party scheme between a sender (also called committer) and a receiver. It usually comprises a generation phase, a commitment phase and an opening phase. In the generation phase, given generators g, h and a security parameter k, a key generation algorithm `Gnrt` outputs public parameters for the commitment scheme. Note that `Gnrt` is normally run by a trusted third party. In the commitment phase, the commitment com to a value m is computed by a committer with a parameter $open$ as $com = \texttt{Comm}(m, open)$. The opening phase is also named as reveal phase, in which m is revealed and checked with $\texttt{Opnv}(com, m, open)$.
The security properties of a commitment scheme include hiding and binding. To be specific, 1) Without $open$, it is infeasible for attackers to learn information about m from com with bounded (computational hiding) or unbounded (perfect or unconditional hiding) computing resources. 2) Finding another message with the same com to m is infeasible for a computationally -bounded (computational binding) or computationally-unbounded (perfect or unconditional binding) attacker.
The Pedersen commitment scheme is adopted in our work. It is based on the Discrete Logarithm Problem (DLP). The DLP is defined as follows: given a group, a generator g and an element h of it, to find $\log_g h$ in the group where log is the discrete logarithm. The hardness of DLP depends on the group. The protocols of the Pedersen commitment are as follows, 1) p and q denote large

primes. q divides $p-1$. $\mathbb{G}_q$ is the unique subgroup of $\mathbb{Z}_p^*$ of order q. g is a generator of $\mathbb{G}_q$. *open* is set as $r \in \mathbb{Z}_q$. 2) The commitment to a value $m \in \mathbb{Z}_q$ can be defined as $com = \texttt{Comm}(m,r) = g^m h^r$ where h is an element of $\mathbb{G}_q$ such that only the receiver knows $\log_g h$. 3) The receiver can check if $\texttt{Opnv}(com,m,r) = true$ by re-computation when m and r are both revealed. The Pedersen commitment scheme is a perfect-hiding scheme, binding under the discrete logarithm assumption.

(3) *Visual Cryptography*: Visual cryptography allows visual information such as pictures to be encrypted in such a way that the decryption can be performed by human vision instead of algorithms. One well-known secret sharing scheme based on visual cryptography [23] is achieved by breaking up an image *fig* into n shadow images called shares, say $\mathscr{A} = \{fig_0, fig_1, \ldots, fig_{n-1}\}$. Each participant in the scheme holds one share. There are two properties of the scheme: 1) The original image can be decrypted visually by overlaying all the elements in $\mathscr{A}$. 2) Any proper subset $\mathscr{B}$ in $\mathscr{A}$, i.e., $\mathscr{B} \in \mathscr{A}$ but $\mathscr{B} \neq \mathscr{A}$, cannot reveal information about *fig*.
One possible application is to encode the secret message into two shares $\{fig_0, fig_1\}$. fig_0 is printed and sent by mail as a ciphertext while fig_1 is kept as a private key. The scheme works like a private key cryptosystem but needs neither cryptography knowledge nor complex computation. The scheme is also similar to the one-time pad as each secret message is encrypted by different private keys, i.e., fig_1.
With the idea of visual cryptography, we proposed a fragment coding-based approach for QR codes. Each QR code can be easily split into two parts and recovered quickly, while it is hard for an adversary to get the original code by any one of the parts. The approach is described in detail in Section 4.2.

(4) *One-way Function and Randomness*: One-way function is a widely used cryptography tool, which is easy to check but hard to invert. To be specific, for any randomized algorithm $\mathcal{F}$ which attempts to compute a pseudo-inverse for a one-way function f, any positive integer c and sufficiently large n, we have [24]:

$$\Pr[f(\mathcal{F}(f(x))) = f(x)] < \frac{1}{n^c}. \tag{1}$$

A hash function is called one-way hash function because it satisfies the property above. It outputs a fixed-size value for an input data with arbitrary size. The output value is called hash values or hash codes. One particular application of one-way hash function is the keyed-hash message authentication code (HMAC). HMAC is usually used for message authentication and integrity.
In the design of the enhanced QR codes, we adopted SHA-2 (Secure Hash Algorithm 2) as the hash function used in the HMAC. SHA-2 is an iterated hash function, using the Merkle-Damgård structure. The Merkle-Damgård structure defines a hash function h based on an external one-way compression function $f : \{0,1\}^{m+n} \to \{0,1\}^m$ where $n \geqslant 2$. With HMAC-SHA-2, the modification and forgery attack can be resisted. Detailed analysis can be found in Section 5.4.
Additionally, one-way function can also apply to Pseudorandom Number Generators (PRNGs) in this work. A PRNG, also known as a deterministic random bit generator, is a deterministic polynomial-time algorithm mapping a uniformly chosen short string called seed to a longer string, i.e., $\mathcal{G} : \{0,1\}^n \to \{0,1\}^{l(n)}$ where l is a stretching function, n is the length of the seed and $l(n)$ is the length of the output string. The output $\mathcal{G}_n$ is computationally indistinguishable from a uniform distribution $\mathcal{R}_n$ on $\{0,1\}^{l(n)}$ where $n \in \mathbb{N}$. With a PRNG, the output of the fragment coding-based approach we proposed is pseudo-random. The confidentiality, authenticity and integrity are ensured with this property. The details are given in Section 5.5.

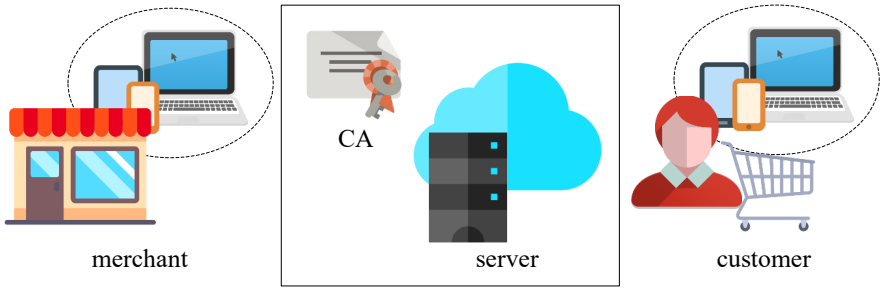

Fig. 1. System model

3.2. System Model

We consider a system model comprised of four entities, which are shown in Figure 1:

(1) Root certificate authority (CA): a root certificate authority generates and distributes keys for other entities. It issues certificates to merchants and customers. CA is trusted by other entities in the model.
(2) Merchant: a merchant provides services or goods to customers. It can support e-coupon transactions but may be a malicious entity.
(3) Customer: a customer uses e-coupons when buying services or goods from a merchant. It also may be a malicious entity and may be in collusion with a merchant.
(4) Server: a trusted server manages transactions with e-coupons. The main functions include e-coupon generation, e-coupon distribution, authentication, verification, e-coupon updating and so on. Besides, it handles disputes when necessary.

The scenarios of online transactions and offline transactions are considered in this work. In both scenarios, the server generates the e-coupons, i.e., enhanced QR code pairs, and distributes them to the merchant and customer, respectively.

(1) Online transactions allow the customer to finish transactions remotely on computers or mobile phones. Both the customer and the merchant need to send one enhanced QR code to the server, respectively. Note that the merchant only sends it when receiving the request from the server, which means there is no direct interaction between the merchant and the customer. The server will process the verification and payment when receiving the QR codes.
(2) In offline transactions, the customer does not directly connect to the server, but sends the encrypted and signed QR code to the merchant through near-field communications, e.g., bluetooth technologies. The merchant then sends his own QR code with the customer's one to the server. As there is a direct interaction between merchants and customers, disputes need to be resolved.

3.3. Security Goals and Threats

In this paper, we aim at achieving the following security goals,

(1) *Authentication*: merchants and customers must be authorized by a CA. In a transaction, the identities of both the merchant and the customer should be verified;
(2) *Data integrity*: no adversary can temper or damage e-coupons without being detected;

(3) *Data authenticity*: the server with EQRC should be able to detect the e-coupons forged by adversaries;
(4) *Data confidentiality*: the sensitive information of e-coupons is only visible to the server. Any other entities including merchants and customers cannot get the access;
(5) *Identity anonymity*: the true identity of a user should not be exposed during signing and verification so that the user privacy is preserved.

Besides, the integrity and authenticity of the messages sent between entities are also required.

The potential threats in e-coupon transaction services we focus on are *information leakage*, *message eavesdropping*, *message modification*, *message forgery*, *message replay attack* and *collusion attack* between a merchant and a customer. *Liability disputes* are also considered.

4. Framework Design

In this section, we first provide the details of the framework design of EQRC-I, which is used for online scenarios. Then we briefly talk about the framework of EQRC-II for offline scenarios. The protocol flow of EQRC-I is shown in Fig. 2. There are four main algorithms, namely Enhanced QR Coding (Alg. 1), Signing (Alg. 2), Verification (Alg. 3) and Recovery (Alg. 4).

The notations of our framework are listed in Table 1.

Table 1
Notations

Notation	Descriptions
S	A server
M	A merchant
C	A customer
$isk(x, y)$	The dualistic secret key of CA
(pk, sk)	A key pair: (public key, secret key)
key	A secret key of M, managed by S
$data_0$	The sensitive data of an e-coupon
$data$	The data of $data_0$ after preprocessing
msg	The message needs to be sent or received
com	A commitment result
$\overline{QR}$	The standard QR code of com
$(\overline{QR^1}, \overline{QR^2})$	A pair of patterns generated from $\overline{QR}$
$\overline{CQR}$	A carrier QR code
$(\overline{SQR^1}, \overline{SQR^2})$	A pair of enhanced QR codes
$\overline{ESQR}$	A QR code generated by $(\overline{SQR^1}, \overline{SQR^2})$

4.1. Protocol Setup

There is a trusted certificate authority CA in the framework we proposed. CA distributes key pairs (pk_S, sk_S), (pk_M, sk_M) and (pk_C, sk_C) for trusted servers S, merchants M and customers C, respectively. Besides, CA generates a secret key key for each M, which is secret to M and hosted in S.

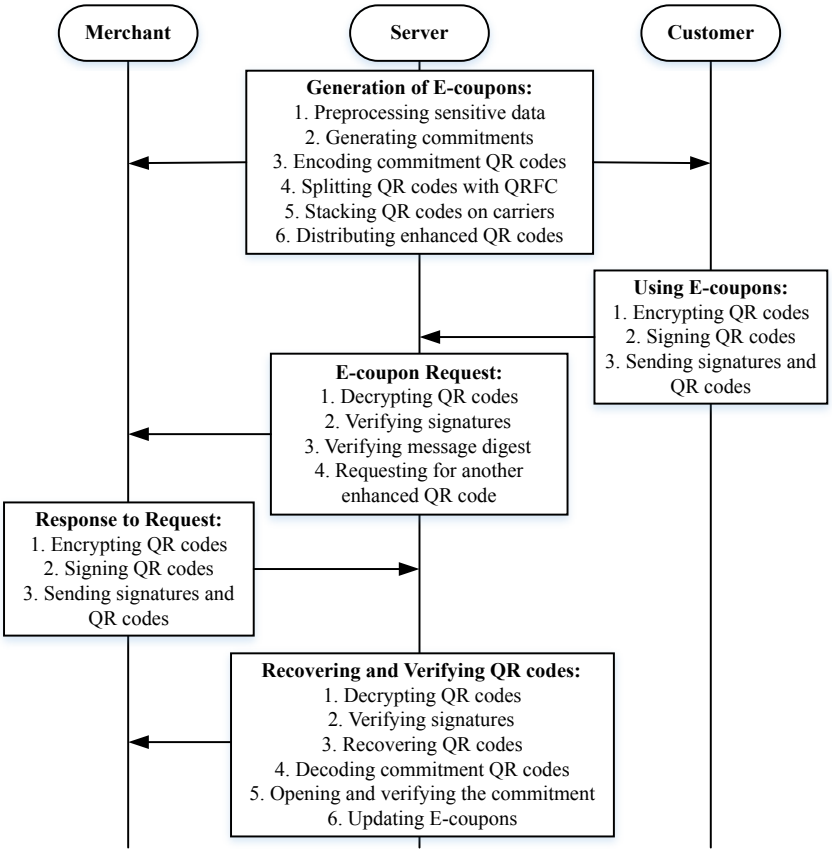

Fig. 2. Protocol flow of EQRC-I

In addition, for message signature, *CA* issues triplet certificates for users, i.e., merchants and customers, which provides the authentication of users. The protocol setup is similar to that in [22] and [25]. Three cyclic groups $\mathbb{G}_1 = \langle P_1 \rangle$, $\mathbb{G}_2 = \langle P_2 \rangle$ and $\mathbb{G}_T$ of sufficiently large prime order q are chosen. P_1 and P_2 are two random generators. A pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is chosen with following properties:

(1) *Bilinear*: $\hat{e}(aP_1, bP_2) = \hat{e}(P_1, P_2)^{ab}$ holds for any two integers $a, b \in \mathbb{Z}_q$;
(2) *Non-degenerate*: $\hat{e}(P_1, P_2) \neq 1_{\mathbb{G}_T}$ where $1_{\mathbb{G}_T}$ is the identity element of $\mathbb{G}_T$;
(3) *Computable*: there exists a polynomial time algorithm for computing $\hat{e}(P, Q)$ for $\forall P \in \mathbb{G}_1$ and $\forall Q \in \mathbb{G}_2$.

The triple certificates (A, B, C) are constructed with users' true identities f and the secret key of *CA*, i.e., the bigram $isk(x, y)$. To be specific, $A \leftarrow r \cdot P_1$, $B \leftarrow y \cdot A$ and $C \leftarrow (x \cdot A + fxy \cdot A)$ where r is a random integer in $\mathbb{Z}_q$. Then each certificate can be used to sign messages for a specific user. More details are in [22].

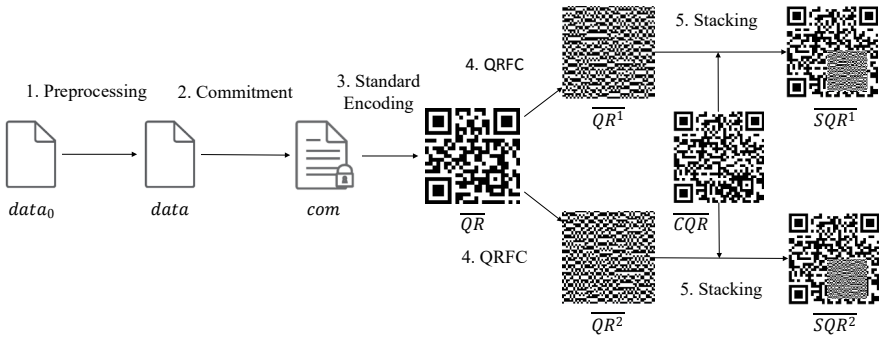

Fig. 3. Generation of enhanced QR codes

4.2. Generation of Enhanced QR Codes

The framework we proposed is based on an enhanced QR code scheme. With this scheme, a standard QR code is processed to finally generate a pair of enhanced QR codes $(\overline{SQR^1}, \overline{SQR^2})$, which can be seen as two fragments of an e-coupon. Enhanced QR codes ensure the confidentiality of e-coupons and are hard to tamper with or forge.

Algorithm 1 Enhanced QR Coding

procedure ENCODING(Sensitive Data *data*)
  $(com, decom) \leftarrow Com(crs, data)$
  $\overline{QR} \leftarrow Encode(com)$
  $(\overline{QR^1}, \overline{QR^2}) \leftarrow Split(\overline{QR})$
  $\overline{SQR^1} \leftarrow Stack(\overline{QR^1}, \overline{CQR})$
  $\overline{SQR^2} \leftarrow Stack(\overline{QR^2}, \overline{CQR})$
return $(\overline{SQR^1}, \overline{SQR^2})$
end procedure

The generation flow of enhanced QR codes is shown in Fig. 3. As Algorithm 1 discusses, there are four main functions in the generation process: 1) $Com(crs, data)$ denotes the commitment process on the sensitive data *data* with the parameter *crs*. It returns a commitment *com* and a parameter *decom* to open the commitment. 2) $Encode(com)$ returns a standard QR code with the specific content *com*. 3) $Split(\overline{QR})$ returns a pair of pseudo-random patterns with a fragment coding-based approach on a standard QR code $\overline{QR}$. 4) $Stack(\overline{QR^1}, \overline{CQR})$ is a graphic combination between an enhanced QR code $\overline{QR^1}$ and a carrier QR code $\overline{CQR}$. Some details are given as follows.

4.2.1. Commitment

Consider the capacity of QR codes and the size of commitments, the sensitive data $data_0$ in an e-coupon can be first preprocessed to *data* through a hash function, as the first step shown in Fig. 3. To ensure the confidentiality of $data_0$, as Step 2, we use the Pedersen commitment technique to generate the commitment $com = g^{data}h^r \mod p$ where (p, g, h, r) are security parameters. With such a perfect-hiding commitment scheme, even though *M* and *C* conduct the collusion attack, they can get nothing of $data_0$ from *com*.

4.2.2. QRFC Approach

With the idea of visual cryptography, we introduce QRFC, a fragment coding-based approach for QR codes. Through QRFC, we can split $\overline{QR}$, i.e., a standard QR code encoded with *com* by Step 3, to two fragments $(\overline{QR^1}, \overline{QR^2})$. The details are as follows. We use k bits 0 and 1 to denote the black and white blocks in an original QR code $\overline{QR}$, i.e., $\{0,1\}^k$, where $k \in \{k = 2k_0 | k_0 \in \mathbb{N}^* \wedge k_0 > 0\}$. Thus, one block has 2^k types of code words, i.e., $Str = \{r_0 r_1 \cdots r_{k-1} | r_i = 0 \vee 1, 0 \leqslant i \leqslant k-1\}$.

Define $S^0 = \{r_0 r_1 \cdots r_{k-1} | r_0 = 0\}$, $S^1 = \{r_0 r_1 \cdots r_{k-1} | r_0 = 1\}$. If a block in $\overline{QR}$ is black, the corresponding code words, i.e., Str^1 in $\overline{QR^1}$ and Str^2 in $\overline{QR^2}$, should satisfy $Str^1 + Str^2 (\mathsf{mod}\ 2) = S^0$. If the block is white, $Str^1 + Str^2 (\mathsf{mod}\ 2) = S^1$.

With the rule above, all blocks in $\overline{QR}$ are re-encoded randomly to two sequences. Turn each 0 in sequences to a black pixel and 1 to a white pixel so that $\overline{QR^1}$ and $\overline{QR^2}$ are finally generated.

We choose $k = 4$ in our framework. The construction method is shown in Table 2. For example, if one block in $\overline{QR}$ is white, the corresponding bit strings in $\overline{QR^1}$ and $\overline{QR^2}$ can be 1001 and 0110, respectively (or 0110 and 1001, respectively). To recover the white block, we only need to compute $1001 + 0110 (\mathsf{mod}\ 2) = 1111$ (or $0110 + 1001 (\mathsf{mod}\ 2) = 1111$). It is clear that there are two choices for each block to split. The choice from *Choice* for any block is pseudo-random with the help of PRNGs, which does not affect the success of decoding.

Table 2
QRFC Approach ($k = 4$)

Block in $\overline{QR}$	Choice	$\overline{QR^1}$	$\overline{QR^2}$
black	1	1001	1001
	2	0110	0110
white	1	1001	0110
	2	0110	1001

4.2.3. Carrier QR Codes

As customers need an easy way to access some public information, such as the merchant's name, the expiry date of the coupon and the discount amount, we introduce a standard QR code, say carrier QR code $\overline{CQR}$. $\overline{CQR}$ also maintains a hash of $\overline{QR^1}$, $h(\overline{QR^1})$, which can be used to verify $\overline{QR^1}$. $h(\overline{QR^1})$ is generated through HMAC-SHA2 with the corresponding secret key *key*. The security provided by $h(\overline{QR^1})$ is analyzed in detail in Section 5.5.

Based on the *Reed-Solomon* error correction code used in QR codes, the enhanced QR codes $\overline{SQR^1}$ and $\overline{SQR^2}$ can be generated by Step 5, i.e., embedding $\overline{QR^1}$ and $\overline{QR^2}$ into $\overline{CQR}$ separately without damaging the necessary data in $\overline{CQR}$. Note that $\overline{SQR^1}$ and $\overline{SQR^2}$ use the same $\overline{CQR}$ and the same embedding position for future processing. More explanation is given in Section 4.4.

To emphasize that the pair of $\{\overline{SQR^1}, \overline{SQR^2}\}$ is used as e-coupons in EQRC. S distributes the fragment $\overline{SQR^1}$ to C and $\overline{SQR^2}$ to M after the generation.

4.3. Signing and Verification

To protect the messages sent between entities, we introduce encryption and anonymous authentication techniques. In this section, we mainly talk about the transaction process instead of the distribution process of S. Before being sent, a message must be encrypted and signed. $(\overline{SQR})_{pk_S}$ denotes the ciphertext

of $\overline{QR^1}$ or $\overline{QR^2}$ using *ElGamal* algorithm with pk_S. The anonymous authentication scheme we introduce is similar to [22] and [25].

Algorithm 2 performs the signing on message *msg* and generates a signature σ by an entity U. In this algorithm, (R,S,T) can be seen as an anonymous certificate. C and S provide the correlation proof of (R,S,T) and the true identity of the entity U. n_t is a timestamp to defend against the replay attack. $\hat{e}$ is a map function [22] from $\mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$. H is a hash function $\{0,1\}^* \to \mathbb{Z}_q$.

Algorithm 2 Signing

procedure SIGNING(Message *msg*)
  if $U = C$ **then** $msg \leftarrow \overline{SQR^1}$
  else if $U = M$ **then** $msg \leftarrow \overline{SQR^2}$
  end if
  $a \leftarrow \mathbb{Z}_q;\ z \leftarrow \mathbb{Z}_q; R \leftarrow a \cdot A;\ S \leftarrow a \cdot B;\ T \leftarrow a \cdot C;$
  $\tau \leftarrow \hat{e}(S,X)^z; c \leftarrow H(R||S||T||\tau||n_t||msg)$
  if $U = C$ **then** $s \leftarrow z + c \cdot sk_C \pmod q$
  else if $U = M$ **then** $s \leftarrow z + c \cdot sk_M \pmod q$
  end if
  $\sigma \leftarrow (R,S,T,c,s,n_t)$
return σ
end procedure

The verification algorithm carried out by S is shown in Algorithm 3. $Dec_{sk_S}(msg)$ is a decryption function of *msg* under sk_S. Because $S = a \cdot B = ay \cdot A = y \cdot R$, $\hat{e}(R,Y) = \hat{e}(A,P_2)^{ay} = \hat{e}(S,P_2)$ can be checked first. Then S re-computes τ from the elements it holds and checks the consistency of C.

Algorithm 3 Verification

procedure VERIFICATION(Message *msg*, Signature σ)
  $msg = (\overline{SQR})_{pk_S}$
  $demsg \leftarrow Dec_{sk_S}(msg)$
  if $\hat{e}(R,Y) \neq \hat{e}(S,P_2)$ **then**
    return Reject
  end if
  $\rho_a^\dagger \leftarrow \hat{e}(R,X); \rho_b^\dagger \leftarrow \hat{e}(S,X); \rho_c^\dagger \leftarrow \hat{e}(T,P_2)$
  $\tau^\dagger \leftarrow (\rho_b^\dagger)^s \cdot (\rho_c^\dagger / \rho_a^\dagger)^{-c}$
  if $c \neq H(R||S||T||\tau^\dagger||n_t||demsg)$ **then**
    return Reject
  end if
return Accept
end procedure

4.4. Recovery and Triple Verification

Once receiving a pair of enhanced QR codes $\overline{SQR^1}$ and $\overline{SQR^2}$, S is able to recover and verify e-coupons. Algorithm 4 describes the details of the recovery. It contains three main processes: 1) the decoding process of QRFC. Because the carrier QR codes of $\overline{SQR^1}$ and $\overline{SQR^2}$ are the same, by computing

$\overline{SQR^1} + \overline{SQR^2} (\mathsf{mod}\ 2)$, $\overline{CQR}$ turns to be all 0. Thus, $\overline{ESQR}$ can be easily extracted, which is actually $\overline{QR^1} + \overline{QR^2} (\mathsf{mod}\ 2)$. 2) $Decode(\overline{ESQR})$ is the standard decoding operation on $\overline{ESQR}$, which recovers the commitment value com'. 3) $Ver(crs, com', decom, data) = 1$ denotes the opening of commitment com', which is one of the three verifications we proposed.

Here are some details of the triple-verification which can defend against the tampering and forgery attacks on e-coupons. First, the message digest $h(\overline{QR^1})$ is checked under *key* before the recovery of e-coupons. It can verify if $\overline{QR^1}$ is modified, damaged or forged. Then the recovery is achieved and the graph of $\overline{ESQR}$ can be checked. In addition, through opening the commitment, com' is verified. With these three verifications, we cannot only ensure the authenticity and integrity of enhanced QR codes, but also figure out the attacker, i.e., *C* or *S*, if any.

Algorithm 4 Recovery

```
1: procedure RECOVERY(QR Codes (SQR^1, SQR^2))
2:     ESQR ← SQR^1 + SQR^2 (mod 2)
3:     com' ← Decode(ESQR)
4:     if Ver(crs, com', decom, data) = 1 then
5:         return (Accept, com')
6:     end if
7: return Reject
8: end procedure
```

4.5. EQRC-II for Offline Payment

We propose EQRC-II for a scenario where customers visit stores but are not connected to the Internet directly. In this scheme, *M* will transmit the enhanced QR code from *C* to *S*, which results in a new risk, i.e., *M* may tamper or forge the message from *C*. Thus we should pay attention to the audit trail of the messages.

The protocol flow of EQRC-II is shown in Fig. 4. The generation and distribution of QR code-based e-coupons in EQRC-II are the same with that in EQRC-I. Other necessary details are provided as follows.

4.5.1. E-coupons Delivery

Similar to EQRC-I, EQRC-II introduces encryption and authentication schemes. $\overline{SQR^1}$ held by *C* is encrypted first so that *M* cannot access $\overline{SQR^1}$. The message sent from *C* to *M* through near-field communications is $msg_1 = ((\overline{SQR^1})_{pk_S} \parallel \sigma_1)$ where $(\overline{SQR^1})_{pk_S}$ is an encryption result and σ_1 is a signature on $\overline{SQR^1}$ with Algorithm 2. Once receiving msg_1, *M* computes $m = ((\overline{SQR^1})_{pk_S} \parallel \sigma_1 \parallel (\overline{SRQ^2})_{pk_S})$ which combines msg_1 and the second enhanced QR code. The message sent from *M* to *S* is $msg_2 = (m \parallel \sigma_2)$ where σ_2 is a signature on *M*.

4.5.2. Audit Trail

With the triple-verification scheme, *S* can verify whether $\overline{SQR^1}$ and $\overline{SQR^2}$ are tampered or forged. However, if $\overline{SQR^1}$ is attacked, *S* cannot figure out who is the attacker. Thus, we propose an audit trail solution.

When the verification on SQR^1 failed, *S* submits an audit request to *CA* who manages the real identities of all users. To prove itself, *C* needs to connect the Internet and confirm its identity to *CA*. *CA* checks if

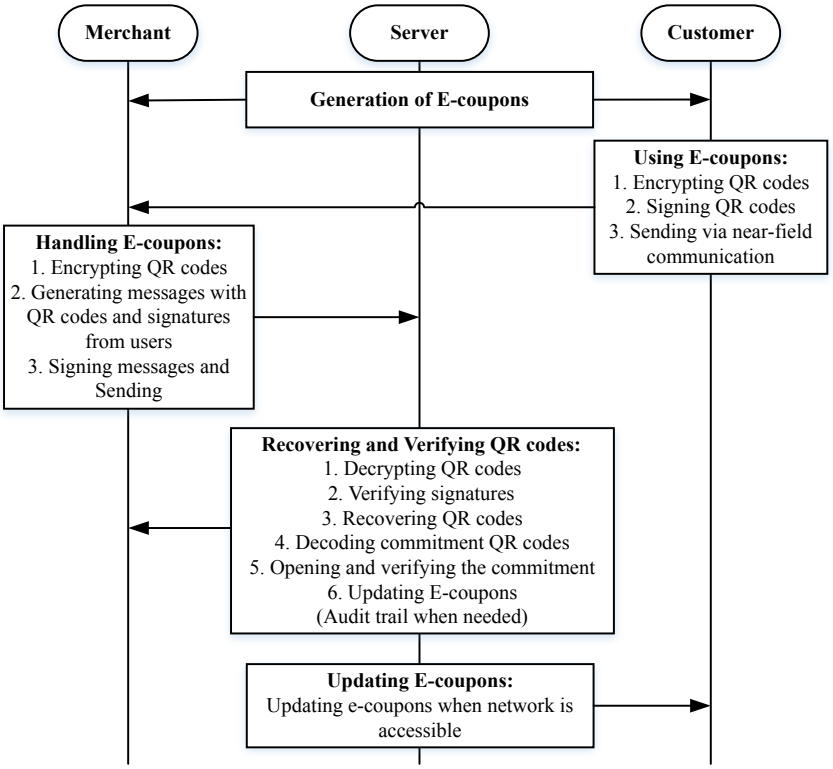

Fig. 4. Protocol flow of EQRC-II

$\sigma_1.T = x \cdot \sigma_1.R + x \cdot sk_C \cdot \sigma_1.S$. The proof is as follows. Note that some transformation is from the join algorithm in [22] which achieves the purpose of certificates, i.e., the credentials in [22].

$$\begin{aligned} \sigma_1.T &= a \cdot C \\ &= a \cdot x \cdot A + a \cdot rxy \cdot pk_C \\ &= a \cdot x \cdot A + a \cdot rxy \cdot sk_C \cdot P_1 \\ &\equiv x \cdot \sigma_1.R + x \cdot sk_C \cdot \sigma_1.S. \end{aligned} \tag{2}$$

If the equation is satisfied, C is the attacker or the data storage of C is compromised. If not, which means σ_1 is not the exact one generated by C, M should be the attacker or the data storage of M needs a strengthened protection.

5. Security Analysis

In this section, we first analyze the security of the proposed framework on six aspects, according to the protocol flow of EQRC. Then, we describe how the proposed framework resists attacks and achieves the corresponding security goals.

5.1. Security of the Digital Signature

The security of the digital signature is guaranteed by the hardness of the **Blind Bilinear LRSW Assumption** [25]. Suppose that a $Setup(1^k)$ algorithm generates the cyclic groups $\mathbb{G}_1 = \langle P_1 \rangle$, $\mathbb{G}_2 = \langle P_2 \rangle$ and $\mathbb{G}_T$ with a prime order q, where k is a parameter related to the security level. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ be a pairing. Let $X, Y \in \mathbb{G}_2, X = x \cdot P_2, Y = y \cdot P_2$. Let $O_{X,Y}(\cdot)$ denote an oracle that, with a randomly chosen $r \in \mathbb{Z}_q$ and an input $f \in \mathbb{Z}_q$, outputs a triplet (A, B, C) where $A = r \cdot P_1$, $B = y \cdot A$, and $C = (x \cdot A + fxy \cdot A)$. Then given the group setup $(\mathbb{G}_1, \mathbb{G}_2, P_1, P_2, q, \hat{e})$ and the public key(X, Y), it is impossible for a probabilistic polynomial-time (*p.p.t.*) adversary $\mathcal{A}$ to construct the triplet (A, B, C) without knowing the secret key(X, Y). To be specific, for all $\mathcal{A}$, given the security parameter k and the set Q that $\mathcal{A}$ queries with $O_{X,Y}(\cdot)$, $v(k)$ is a negligible function defined as follows:

$$\begin{aligned} &\Pr[(\mathbb{G}_1, \mathbb{G}_2, P_1, P_2, q, \hat{e}) \leftarrow Setup(1^k); x \leftarrow \mathbb{Z}_q; y \leftarrow \mathbb{Z}_q; X = x \cdot P_2, Y = y \cdot P_2; \\ &(f, A, B, C) \leftarrow \mathcal{A}^{O_{X,Y}}(\mathbb{G}_1, \mathbb{G}_2, P_1, P_2, q, \hat{e}) : f \notin Q \wedge f \in \mathbb{Z}_q \wedge f \neq 0 \wedge A \in \mathbb{G}_1 \wedge B \\ &\qquad = y \cdot A \wedge C = x \cdot A + fxy \cdot A] \leqslant v(k). \end{aligned} \tag{3}$$

This assumption guarantees that without the secret key(X, Y) and the random number $r \in \mathbb{Z}_q$, any *p.p.t.* adversary cannot forge a valid credential (A, B, C). Furthermore, for the case of the signing algorithm, a signature σ is constructed with the secret parameters a and z, the user's secret key sk_U (i.e., sk_C or sk_M), the timestamp n_t and the shuffled credential (R, S, T) where $R = a \cdot A$, $S = a \cdot B$, and $T = a \cdot C$. Any adversary $\mathcal{A}$ cannot produce a valid anonymous signature σ for any message msg without (A, B, C) and sk_U. Thus, a signed message cannot be forged or modified during the processes of transmission, which provides the non-repudiation, authenticity and integrity of messages. In addition, because there is no isomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$ in the asymmetric pairing setting, it is infeasible to link (R, S, T) to the original (A, B, C) without the secret parameter a, which provides users with anonymity.

For the case of the verification algorithm, the above assumption also guarantees that only holding the bilinear group parameters $(\mathbb{G}_1, \mathbb{G}_2, P_1, P_2, q, \hat{e})$, the public key$(X, Y)$, the message msg, and the shuffled credential (R, S, T), users can check whether the following verification equations hold: $\hat{e}(A, Y) = \hat{e}(B, P_2)$, $\hat{e}(A, X)\hat{e}(fB, X) = \hat{e}(C, P_2)$, $\hat{e}(R, P_2) = \hat{e}(A, P_2)^a$, $\hat{e}(S, P_2) = \hat{e}(A, Y)^a$, and $\hat{e}(T, P_2) = \hat{e}(A, X)^a \hat{e}(fB, X)^a$.

5.2. Security of the Encryption

The security of the ElGamal encryption in this work is guaranteed by the hardness of the **Decisional Diffie-Hellman (DDH) Assumption** [26]: Assume that $\mathbb{G}_1 = \langle P_1 \rangle$, $\mathbb{G}_2 = \langle P_2 \rangle$ and $\mathbb{G}_T$ are cyclic groups with a prime order q. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ be a pairing. Let $X, Y, Z \in \mathbb{G}_1$, $X = x \cdot P_1$, $Y = y \cdot P_1$, and $Z = z \cdot P_1$. Note that x, y and z are randomly and independently chosen from $\mathbb{Z}_p$. Then given the group parameters $(\mathbb{G}_1, \mathbb{G}_2, P_1, P_2, q, \hat{e})$, for any probabilistic-polynomial time $(p.p.t.)$ adversary $\mathcal{A}$, the advantage $\mathbf{Adv}_{\mathcal{A}}^{DDH}$ defined as follows is negligible:

$$\begin{aligned} \mathbf{Adv}_{\mathcal{A}}^{DDH} = |\Pr[x, y, z \leftarrow \mathbb{Z}_q; X = x \cdot P_1, Y = y \cdot P_1, Z = z \cdot P_1; \mathcal{A}(\mathbb{G}_1, \mathbb{G}_2, \\ P_1, P_2, X, Y, Z, q) = 1] - \Pr[x, y \leftarrow \mathbb{Z}_q; X = x \cdot P_1, Y = y \cdot P_1, Z = \mathbb{G}_1; \\ \mathcal{A}(\mathbb{G}_1, \mathbb{G}_2, P_1, P_2, X, Y, Z, q) = 1]|. \end{aligned} \tag{4}$$

In other words, the distributions $\langle x \cdot P_1, y \cdot P_1, xy \cdot P_1 \rangle$ and $\langle x \cdot P_1, y \cdot P_1, z \cdot P_1 \rangle$ are computationally indistinguishable, which is equivalent to the semantic secure in ElGamal encryption [26]. This assumption guarantees that, without knowing the private key, the probability of getting $z = xy$ from $Z = xy \cdot P_1$ for a *p.p.t.* adversary $\mathcal{A}$ is negligible. Thus, the confidentiality of messages in transmission is provided in our framework. As summarized in Table 3, the typical attacks on e-coupons and messages transmitted, including the data leakage, forgery and modification, are resisted.

5.3. Security of the Commitment

In general, there are three algorithms in a Pedersen commitment [27], i.e., the generation algorithm $\texttt{Gnrt}$, the commitment algorithm $\texttt{Comm}$ and the opening algorithm $\texttt{Opnv}$. Let k denote the security parameter. Suppose that the $\texttt{Gnrt}(1^k)$ algorithm generates a set of common parameters (p, q, g, h) as follows: p and q are two primes which are large enough and $q|p-1$. g and h are generators randomly chosen from a q-order subgroup $\mathbb{G}_q$ of $\mathbb{Z}_p^*$. $h = g^a \mod p$ where a is secret. On inputting a secret $m \in \mathbb{Z}_q$ and a random value $r \in \mathbb{Z}_q$, the $\texttt{Comm}$ algorithm computes a commitment $com = \texttt{Comm}(m, r) = g^m h^r \mod p$, where $com \in \mathbb{Z}_p^*$. Based on the DLP, $\log_g h$ is unknown to the committer. Correspondingly, given a commitment com, a message $m \in \mathbb{Z}_q$ and $r \in \mathbb{Z}_q$, the $\texttt{Opnv}(com, m, r)$ algorithm will output $TRUE$ if and only if com is a valid commitment to M with the given r.

Based on the design described above, for any given value r, the commitment is uniformly distributed with a randomly and uniformly chosen parameter [24], i.e., $|\Pr[\texttt{Comm}(m_1, r)] - \Pr[\texttt{Comm}(m_2, r)]| = 0$, where $m_1, m_2, r \in \mathbb{Z}_q$ and r follows the uniform distribution. In other words, given a commitment com, every value M is equally likely to be the value committed in com. This property is well-known as the **Perfect Hiding Property**.

In EQRC, the perfect hiding property provided by the Pedersen commitment guarantees that any adversary on the channel can only obtain data from $\overline{QR}$ with a negligible probability. To be specific, for every sensitive data $data$ in $\overline{QR}$, there exists a unique $data'$ such that $com = g^{data} h^{r_1} = g^{data'} h^{r_2}$. Thus, com perfectly hides all information about $data$, i.e., the adversary cannot get any advantage from com to guess $data$, even with unlimited computational power. Note that the commitment provides additional protection with the QRFC approach. Even both the two fragments $\overline{SQR^1}$ and $\overline{SQR^2}$ are intercepted and $\overline{QR}$ is recovered, $data$ is still safe. Thus, the commitment technology adopted in EQRC resists against not only the data leakage, forgery and modification attacks on e-coupons, but also the collusion attack between M and C, as in Table 3.

In addition, we introduce the **Non-ambiguity Property** to prove that it is infeasible for a *p.p.t.* adversary $\mathcal{A}$ to forge a different commitment in other ways [24]. To be specific, the advantage $\mathbf{Adv}_{\mathcal{A}}^{NAmb}$ of $\mathcal{A}$ defined as follows is negligible:

$$\begin{aligned} \mathbf{Adv}_{\mathcal{A}}^{NAmb} = \Pr[(g, h, q, p, p^*) \leftarrow \texttt{Gnrt}^*(1^k); r, r^*, m \leftarrow \mathbb{Z}_q; \\ (com, r, r^*) \leftarrow \texttt{Comm}^*(\mathbb{G}_q, g, h, p^*, m); \\ \texttt{Opnv}(p, com, r) \neq \perp, \texttt{Opnv}(p, com, r^*) \neq \perp, \\ \texttt{Opnv}(p, com, r) \neq \texttt{Opnv}(p, com, r^*)], \end{aligned} \tag{5}$$

where $\texttt{Gnrt}^*$ and $\texttt{Comm}^*$ are the generation and the commitment algorithms launched by $\mathcal{A}$.

5.4. Security of the HMAC

For one thing, the security of HMAC is provided by the common construction defined as follows: $HMAC(K,m) = H((K' \oplus opad)||H((K' \oplus ipad)||m))$, where M is a message, K is a secret key, K' is a block-sized key derived from K, *opad* and *ipad* are the block-sized outer and inner padding, respectively, and H is a hash function. In such a construction, the application of the outer function H masks the intermediate result of the internal $H((K' \oplus ipad)||m)$ [28, 29]. Additionally, the cryptographic strength of HMAC depends on the properties of the underlying hash function. In our framework, we adopt SHA-2, which uses the **Merkle-Damgåd Structure**. The soundness of the Merkle-Damgåd structure has been proved [30], that is, if the compression function $f : \{0,1\}^{m+n} \to \{0,1\}^m$ is collision resistant, then the constructed hash function h is collision resistant. In SHA-2, the compression function h_{DM} is designed from a block cipher $\{f_k\}$ using **Davies-Meyer Construction** $h_{DM}(k,x) = f_k(x) \oplus x$, where k and x are inputs of the model. Collision resistance of Davies-Meyer construction can be proved in the ideal-cipher model [31].

Based on the discussion above and the existing research [32], the security of SHA-2 has been widely analyzed and proved. Theoretically, taking SHA-256 as an example, the upper bound of finding a collision using the birthday attack is 2^{128} evaluations and that of a preimage attack using a brute force search is 2^{256}. Though there are some research efforts aiming at attacks [33–35], it is still widely accepted that SHA-2 family is a secure hash algorithm.

Overall, in QRFC, as only Server S knows the HMAC key *key*, both the origin authentication and data integrity for $\overline{QR^1}$ are provided based on the security of HMAC-SHA-2.

5.5. Security of the QRFC Approach

Because of the pseudorandomness property of the PRNG algorithm, the output sequences are computationally indistinguishable for any *p.p.t.* algorithm $\mathcal{A}$, i.e., for all sufficiently large n and positive polynomial $p(\cdot)$, we have

$$\left|\Pr[\mathcal{A}(\mathcal{G}(U_n)) = 1] - \Pr[\mathcal{A}(U_{l(n)}) = 1]\right| < \frac{1}{p(n)}. \tag{6}$$

$\mathcal{G}$ is a PRNG with an output length $l(n)$, U_n is the uniform distribution on $\{0,1\}^n$ and $U_{l(n)}$ on $\{0,1\}^{l(n)}$. The pseudorandomness property guarantees that when $\overline{QR}$ is re-encoded to $\overline{QR^1}$ and $\overline{QR^2}$ by the QRFC approach, the choice for any block is pseudorandom. Anyone with a single enhanced QR code can only guess $\overline{QR}$ with a 0.5^n probability of success, where n is the number of blocks in $\overline{QR}$. In our framework, each user, i.e., M or C can only hold one of the pair $(\overline{SQR^1}, \overline{SQR^2})$. Thus, the confidentiality and integrity of $\overline{QR}$ are provided as it is hard for M or C to recover $\overline{QR}$ by itself. In other words, the data leakage, forgery and modification attacks on e-coupons can be resisted, which is summarized in Table 3.

5.6. Security of the Triple Verification

The security of the first verification is guaranteed by the HMAC-SHA-2, as we analyze in Section 5.4. The secure hash $h(QR^1)$ stored in $\overline{CQR}$ is produced with *key*. Thus no one without *key* can recompute the hash while tampering or forging $\overline{SQR^1}$. Then, the security of the second verification is based on the

Table 3
Defense Against the Typical Attacks

Defense	Signing	Encryption	Commitment	QRFC	TriVerf	Timestamp	Security Properties
Data Leakage		✓	✓	✓			data confidentiality
E-coupon Forgery and Modification		✓	✓	✓	✓		data integrityand authenticity
Collusion Attack			✓		✓		data confidentiality,integrityand authenticity
Eavesdropping		✓					message confidentiality,identity anonymity
Message Forgery and Modification	✓	✓					message integrityand authenticity,identity anonymity
Message Replay						✓	usability

design of the enhanced QR code. Tampering on one of the enhanced QR codes may lead to anomaly in the process of recovery. See section 4.4, for example, the carrier QR code may not change to all black. The third verification is commitment opening, with which S can check *com* to guarantee the integrity and authenticity of the sensitive data in e-coupons. The detailed analysis is given in Section 5.3. The triple verification works together to ensure the security of our framework. Note that if an attack is detected by the first verification, it shows that the attack source is C, i.e., C is an attacker itself, or there are risks of the data storage or transmission in C. Otherwise, the attack source is M.

5.7. Defense Against the Typical Attacks

In this section, we illustrate some specific situations to prove that the proposed framework can resist the threats and achieve the security goals mentioned in Section 3.3. The security properties to defend against the typical attacks are summarized in Table 3. Note that *TriVerf* denotes the triple-verification. *Security Properties* are the corresponding security properties threatened by the attacks. We suppose that $\hat{\mathcal{A}}$ is an adversary. Considering the similarity of the attack principles and solutions in practice, we only choose some typical situations for deep analysis. For example, we only discuss the eavesdropping on messages sent from C to S because that from M to S is similar. The mechanism to defend against the typical attacks are described as follows.

(1) *Eavesdropping and data leakage*: 1) Suppose $\hat{\mathcal{A}}$ can intercept the message $msg = ((\overline{SQR^1})_{pk_S} \parallel \sigma)$ sent from C to S. $\hat{\mathcal{A}}$ wants to get $\overline{SQR^1}$ or the identity of C from msg. However, without sk_s, it is infeasible for $\hat{\mathcal{A}}$ to break the ciphertext $(\overline{SQR^1})_{pk_S}$, which is guaranteed by the hardness of the **Decisional Diffie-Hellman (DDH) Assumption**, as described in Section 5.2. The true identity f and the true credential (A, B, C) of C cannot be revealed with the anonymous signature σ. Thus, the data confidentiality and identity anonymity are provided to defend against the eavesdropping attack. 2) Furthermore, suppose $\hat{\mathcal{A}}$ can get the e-coupon $\overline{SQR^1}$ and wants to guess $\overline{SQR^2}$ and then recover $\overline{QR}$. Because of the pseudorandomness property of the QRFC approach, the probability of $\hat{\mathcal{A}}$ to succeed is only 0.5^n where n is the number of blocks in $\overline{QR}$. Details can be found in Section 5.5. 3) Even if $\hat{\mathcal{A}}$ gets $\overline{SQR^1}$ and $\overline{SQR^2}$ in the same time, he can open the commitment in $\overline{QR}$ with a negligible probability without the security parameter r, which is guaranteed by the Pedersen commitment as described in Section 5.3. Thus, the confidentiality of the sensitive information $data_0$ can be provided.

(2) *E-coupon forgery and modification*: 1) Suppose $\hat{\mathcal{A}}$ gets the e-coupon $\overline{SQR^1}$ and tampers or forges it. However, he cannot generate a valid $h(QR^1)$ stored in $\overline{CQR}$ without *key*, which is guaranteed by the soundness of the **Merkle-Damgård Structure**. As the first verification of EQRC, the details

are introduced in Section 5.4. 2) Suppose $\hat{\mathcal{A}}$ gets the e-coupon $\overline{SQR^2}$ and tampers or forges it. The server can also figure it out by stacking $\overline{SQR^1}$ and $\overline{SQR^2}$. To be specific, anomaly in $\overline{CQR}$ appears according to the security properties of the QRFC approach when $\overline{SQR^1}$ passes the first verification, which means that $\overline{SQR^2}$ is modified or fake. Detailed analysis is shown in Section 5.6. Overall, the authenticity and integrity of e-coupons are provided in EQRC.

(3) *Collusion attack*: Suppose $\hat{\mathcal{A}}$ is registered as a legal merchant. Suppose another adversary, $\hat{\hat{\mathcal{A}}}$, is a legal customer and has collusion with $\hat{\mathcal{A}}$. The advantage of them is that they have pairs of e-coupons $(\overline{SQR^1}, \overline{SQR^2})$ and know the rule to recover $\overline{QR}$. However, they can obtain $data_0$ from $\overline{QR}$ with a negligible probability which is guaranteed by the **Perfect Hiding Property** provided by the Pedersen commitment. Thus, it is infeasible to deploy the collusion attack on EQRC.

(4) *Message forgery, modification and replay*: 1) Suppose $\hat{\mathcal{A}}$ can intercept the message msg sent from C to S and wants to modify or forge it. However, it is infeasible because msg is protected by encryption, whose security is guaranteed by the hardness of the **Decisional Diffie-Hellman (DDH) Assumption**, as described in Section 5.2. Any modification on msg can be detected by the signature σ. In addition, σ is constructed with the secret key sk_C and the credential (R, S, T). Thus, without sk_C and (A, B, C), $\hat{\mathcal{A}}$ cannot forge a valid anonymous signature due to the hardness of the **Blind Bilinear LRSW Assumption**. Details can be found in Section 5.1. With the mechanisms above, the authenticity and integrity of messages, and the identity anonymity are guaranteed in EQRC. 2) Suppose $\hat{\mathcal{A}}$ can intercept the message msg sent from C to S and wants to resend it to S. However, it is infeasible because the signature σ is constructed with a timestamp n_t as shown in Algorithm 2. That is, the replay attack can be detected by checking n_t and the time threshold, which is guaranteed by the hardness of the **Blind Bilinear LRSW Assumption**. Therefore, EQRC can effectively resist the attack of message replay.

6. Performance Evaluation

In this section, we analyze the efficiency of the signature scheme, the overhead of enhanced QR codes, and the communication overhead. All experiments are conducted on Windows 8, with 2.8 GHz Intel CPU, 12 GB memory and 500 GB disk.

6.1. Computation Overhead of the Signature Scheme

We consider the scalar multiplications in $\mathbb{G}_1$, the exponentiations in $\mathbb{G}_t$ and pairing operations, which are time costly operations. The hash operations in $\mathbb{Z}_q$ can be neglected with little overhead. The exponentiations in $\mathbb{G}_t$ can be converted into the scalar multiplications in $\mathbb{G}_1$ to reduce the computation complexity [21]. Thus the computation overhead for signature is determined by $4 \cdot \mathbb{G}_1 + 1 \cdot P$ and that for verification is $3 \cdot \mathbb{G}_1 + 5 \cdot P$, where $n \cdot \mathbb{G}_1$ represents the n scalar multiplications on $\mathbb{G}_1$, and $m \cdot P$ is m pairing operations on $\mathbb{G}_t$.

The experiment in [36] shows that we need to set $|q| = 160$ bits and $|\mathbb{G}_1| = 161$ bits to meet the 80-bit security level. Then one scalar multiplication in $\mathbb{G}_1$ costs 0.6 ms and one exponentiation in $\mathbb{G}_t$ costs 4.5 ms [22]. In our work, the computation overhead for signing and verification is 6.9 ms and 24.3 ms, respectively. Compared with the experiment in [36], the signing in our work is better than that in TAA V1 [22], TAA V2 [22] and GSIS [37]. Because we do not consider the pre-computing of pairing operations, the verification efficiency is lower than that in GSIS, i.e., 13.8 ms.

Table 4
The Setting of QR Code Samples

	a1	a2	a3	b1	b2	b3	c1	c2	c3
Block	Version 1: 21 × 21			Version 3: 25 × 25			Version 7: 33 × 33		
Graph Length (Pixel)	105	210	294	100	200	300	99	198	297
Size (KB)	1.70	5.80	11.5	1.62	5.52	11.7	1.60	5.47	11.6

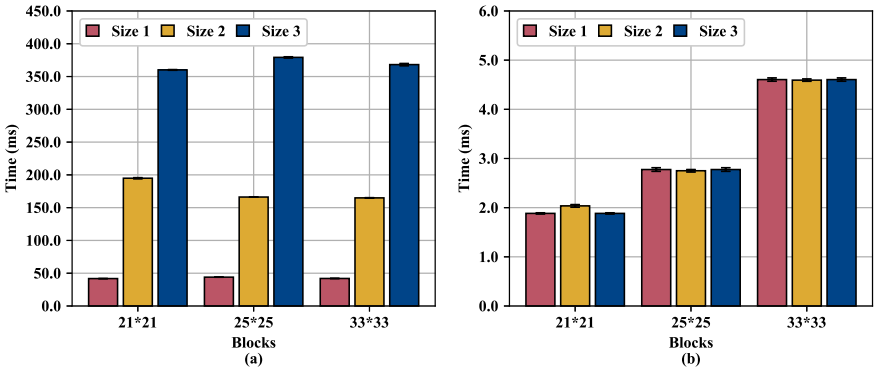

Fig. 5. Running-time overhead for encoding in (a) the preprocessing stage and in (b) the computing stage

6.2. *Encoding/Decoding Overhead for the Enhanced QR Codes*

In our framework, the generation, recovery and verifications of the enhanced QR codes are realized by S with cloud computing, which significantly reduces the computation burden of users.

We evaluate the time consumption for QRFC approach, which is developed based on C#. Three different versions of QR codes are tested, which are Version 1 (Block 21×21), Version 3 (Block 25×25) and Version 7 (Block 33×33). The samples of QR codes are shown in Table 4, where the QR codes with similar pixel resolutions (e.g., for mobile phones or posters) are given the same labels.

The running-time overhead for the encoding process in QRFC is calculated from two stages: the preprocessing stage and the computing stage. The overhead for the preprocessing stage is shown in Fig. 5(a) and that for the computing stage is shown in Fig. 5(b). The preprocessing stage includes the processes such as reading the QR codes and building Bitmap objects with C#. Thus, we can see that the version of QR codes has little effect on the preprocessing time but the size of QR codes affects a lot. In contrast, the computing is on blocks instead of pixels, as described in Section 4.2.2. Thus, as shown in Fig. 5(b), the version of QR codes has effect on the computing time but the size does not. Compared with the preprocessing stage (around 40 to 380 ms in the experiments), the overhead for the computing stage (less than 6 ms in the experiments) is trivial. The reason can be described intuitively: the handling of pictures is slower than computation on given arrays or integers in the experiments. Overall, the encoding time is mainly determined by the preprocessing stage, which is the same for all schemes regardless security.

On the other hand, the decoding process for QRFC also contains two parts: the preprocessing stage and the computing stage. The overhead of the preprocessing stage is shown in Fig. 6(a). The overhead of the computing stage which includes the processes of decoding computation and original QR codes recovery is shown in Fig. 6(b). It is clear that the computing stage (less than 0.25 ms in the experiments) costs

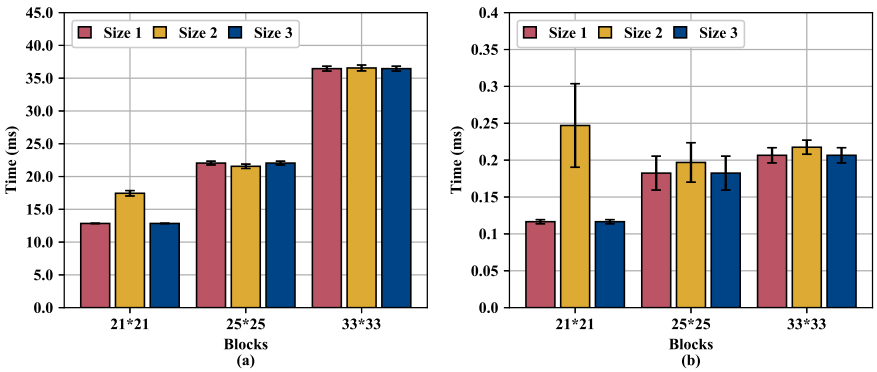

Fig. 6. Running-time overhead for decoding in (a) the preprocessing stage and in (b) the computing stage

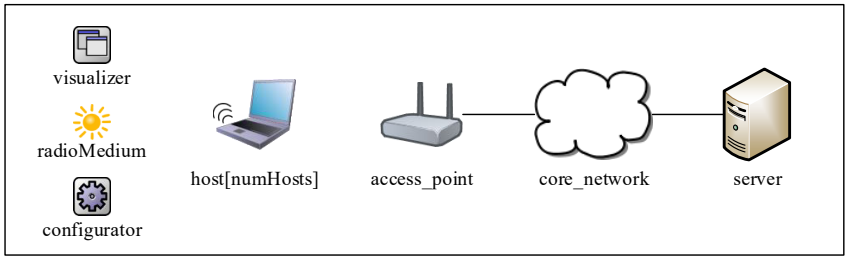

Fig. 7. Network design in Omnet++

less time than the preprocessing stage (about 13 to 35 ms in the experiments). Comparing the decoding process with the encoding process, we can find that the decoding process is more efficient, costing less time than the encoding process. The result is reasonable. As described in Section 4.2.2, pseudo-random numbers are generated for each block to split a QR code in the encoding process. However, only a modulo-2 addition operation is necessary for each block in the decoding process due to the attribute of QRFC. The quicker decoding is a beneficial property for e-coupon transactions, because users concern more about the successful transaction time than the generation time of e-coupons.

6.3. Communication Overhead

To evaluate the communication overhead, we implemented a simulation on the end-to-end delay from users to a server. Because the transactions in both EQRC-I and EQRC-II are composed of several communication rounds and every rounds are similar, the experiment focuses on one round is reasonable and representative. The experiment is conducted by a well-known simulation tool, OMNeT++ 5.5 [38]. An open-source OMNeT++ model suite, INET [39], is adopted.

The network is designed as shown in Fig. 7. The host denotes users in our system, which can be clients or merchants. The hosts are connected to an access point, AP, via a wireless network and thus can send messages to the server through a wired core network. Considering the networking technologies nowadays, we choose IEEE802.11ac with a bit rate of 346.7 Mbps as the wireless network to ensure that most of the new mobile devices can support [40, 41]. The bit rate of the wired network is set as 1000

Table 5
Typical Message Length

Version†	ECC‡ ($\overline{QR}$)	ECC ($\overline{CQR}$)	Size§	Length¶ (EQRC-I)	Length¶ (EQRC-II)
2	Q	H	1.04	2.08	4.16
3	H	H	1.4	2.8	5.6
3	H	Q	1.68	3.36	6.72
2	Q	L	4.46	8.92	17.84
3	H	L	6	12	24

† The versions chosen for $\overline{QR}$. Each version has a different module configuration or number of modules [43]. ‡ Four levels are available for different QR code error correction capabilities: Level L (7%), Level M (15%), Level Q (25%) and Level H (30%) [44]. § The size for one enhanced QR code (KB). ¶ Approximate length of each message (KB).

Mbps with a packet drop rate of 1%. The maximum propagation delay is calculated as the ratio between the half of the longest distance from east to west Canada (2757 km) [42] and the light speed for optical cables (180 000 km/s), which is 15.3167 ms. Note that the server is deployed on the cloud and thus can handle the requests via cloud computing, which is not the main consideration of this simulation.

In this simulation, we mainly focus on the messages sent from users to servers which are the signatures and the ciphertext of $\overline{SQR^1}$ or $\overline{SQR^2}$. The size of a signature is $|\sigma| = 3|\mathbb{G}_1| + 3|q|$. A 160-bit long prime number q and a 161-bit long group $\mathbb{G}_1$ are selected in order to meet the security level of the standard 1024-bit RSA. Therefore the size of a signature is close to 1 kb. The ciphertext in the ElGamal scheme doubles the size of the plaintext, i.e., $\overline{SQR^1}$ or $\overline{SQR^2}$.

The size of enhanced QR codes varies in different settings. For example, as the size of *com* and *data* is no larger than 160 bits, a version-3 QR code is able to encode them with an H-level ECC (Error Correction Capability). Thus, there are 29×29 blocks in $\overline{QR}$. To reduce the QR codes size, we use one pixel in each block. Then the 841 bits need $841 \times 4 = 3364$ bits to represent with the QRFC approach, i.e., each of the $\overline{QR^1}$ and $\overline{QR^2}$ is 3364 bits. If we embed the split QR codes into a carrier QR code with an L-level ECC, which can tolerate only 7% errors, the size of each enhanced QR codes generated is $3364/0.07 = 48058$ bits. Note that the split QR codes embedded are treated as errors by a regular decoder as described in Section 4.2.3. Thus, if the carrier QR codes are smaller than the bound, it cannot be read by a regular decoder with error correction. Overall, a single communication with the settings above should have about 12 KB in EQRC-I, and 24 KB in EQRC-II where a merchant sends two enhanced QR codes. Some typical message lengths are listed in Table 5.

To measure the effect of the message length on the end-to-end delay, we set the message length as 2, 4, 8 and 32 KB for EQRC-I and doubled the size for EQRC-II. In the experiment, 100 messages are sent following a Poisson process in 100 s, which simulates a common scenario with request intensity of 1 (per second). Note that, too large QR codes are seldom used in real life, but we also measure the delay with a rare message length of 32 KB in EQRC-I and 64 KB in EQRC-II, when the latest QR code version (40) is adopted as $\overline{CQR}$. The results are shown in Fig. 8. We can see that, the average end-to-end delay is mainly bounded by 0.1 s. Even with the rare size of enhanced QR codes, the delay is also acceptable.

To measure the effect of the scale of users on the end-to-end delay, we set the number of hosts in the simulation as 100, 500, 1000 and 1500. Each host is expected to send only one 4-KB message with an inter-arrival time following the exponential distribution [45]. The service time is set as 100 s to represent normal scenarios such as using coupons in a shopping mall. In other words, the request intensity is set as 1, 5, 10 and 15 per second. Note that we only consider the end-to-end process from users, i.e.,

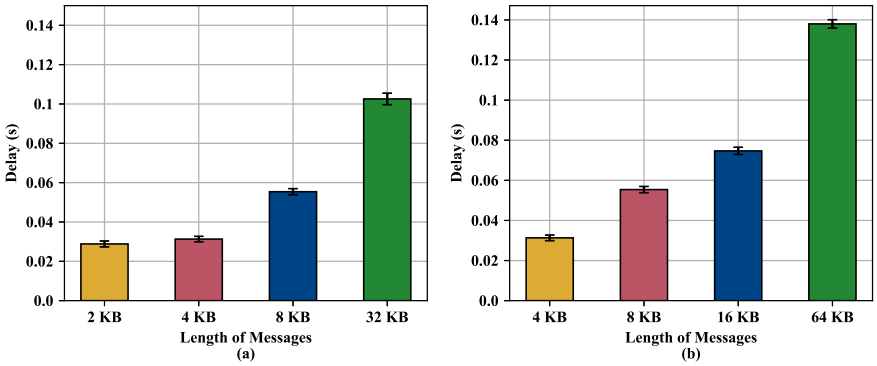

Fig. 8. End-to-end delay in (a) EQRC-I and in (b) EQRC-II

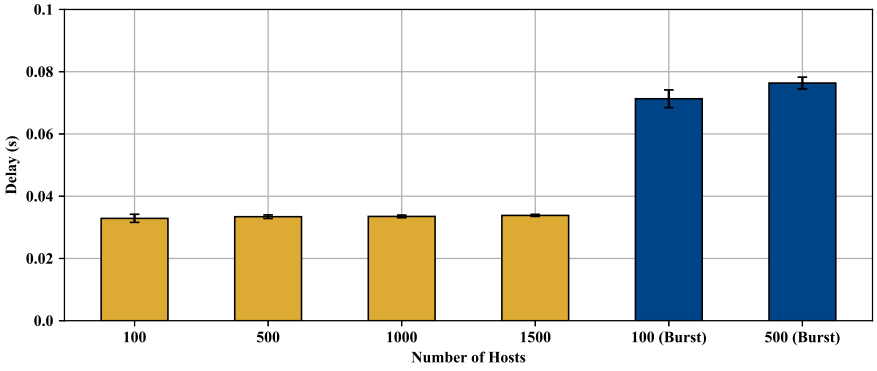

Fig. 9. End-to-end delay in normal scenarios (yellow) and in a burst (blue)

merchants or customers, to a server. The only difference for EQRC-I and EQRC-II is the possible length of messages. That is why we choose 4-KB messages in this experiment, which is probable for both EQRC-I and EQRC-II as shown in Table 5.

The intensity set is expected to have little influence on the end-to-end delay, otherwise, the system capacity is reached easily even in a normal scenario. Besides, another experiment is conducted where all messages are sent at the same time to test the performance further. The variable, number of requests, is set as 100 and 500. In such a burst, the delay is expected to be larger but in an acceptable range. The results in Fig. 9 are the same with the reasonable expectation. It indicates that the delay is not affected much by the scale of requests in normal scenarios while is larger in a burst.

Fig. 10 shows the transmission time of each message and the corresponding delay in bursts. It is clear that when there are 100 hosts trying to send QR codes at the same time, most of the requests are sent out directly and some succeed in around 1 second because of the collisions at the AP. Only a few wait for 3 seconds. When there are 500 hosts, the upper bound of waiting time is around 9 seconds. Overall, with the network configuration described in this section, the EQRC performance is good and acceptable even with a burst.

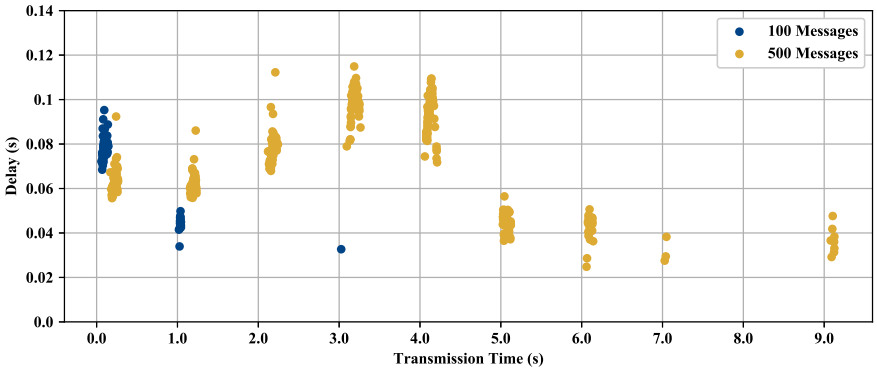

Fig. 10. End-to-end delay and the transmission time in a burst

Table 6
Comparisons of the Secure E-coupon Frameworks

Framework	Anonymity†	Verification‡	Privacy§	Design ¶	Message #	Rounds ††
Framework-A [12]	sk_S	sk_S	a session key	`Out`	e-coupons & `sig`	3
Framework-B [13]	✗	sk_S	✗	`Out`	e-coupons & `MAC`	2
EQRC-I	$isk(x,y)$	triple-verification	triple-protection	`Out` & `In`	`cip` & `sig`	4
EQRC-II	$isk(x,y)$	triple-verification	triple-protection	`Out` & `In`	`cip` & `sig`	3

† The anonymity of users, i.e., merchants and customers. ‡ The verification of e-coupons. § The privacy protection for e-coupons. ¶ The design for security. `Out` and `In` denote outside and inside e-coupons. # The message that sent to redeem e-coupons. `sig` is used to denote signatures, `MAC` for message authentication codes, and `cip` for ciphertext of e-coupons. For more details, we refer readers to the related papers. †† The number of communication rounds for e-coupons redemption and verification.

6.4. Comparison with Related Protocols

In Table 6, EQRC is compared with two recent research papers published in 2014 and 2015 [12, 13]. All of the schemes focus on satisfying the security requirements for the e-coupon system, including authentication and integrity. The difference is as follows. The anonymity of users in EQRC is protected by the private key of CA, $isk(x,y)$, as described in Section 5.1, so that it is infeasible to link the true identity with a signature for C, M and S. However, in Framework-A [12], the anonymity is guaranteed by the server's private key sk_S, which means that the identity is exposed to the server. Framework-B [13] even does not hold such property. Besides, both Framework-A and Framework-B verify e-coupons via a PKI-based solution. To be specific, the authenticity and integrity of e-coupons are protected by the server's private key. In EQRC, to provide stronger security, the triple-verification is presented. More details can be found in Section 5.6.

The detailed computation overhead is compared in Table 7. The result shows that merchants and customers in EQRC only have the work necessary for communication, including signing and encrypting the message while most of the work is on the server. Although Framework-B also has fewer cryptography operations on the user side, it does not provide enough and strong security properties as EQRC. Overall, EQRC is secure and more friendly to users. The advantage is guaranteed by the design of the framework, as shown in Table 6. To be specific, our scheme applies both outside and inside the e-coupons, which

Table 7
Comparisons on Computation Overhead

Framework	Phase†	Entity	Operation‡							
			Commit	Cod	Spl	Sta	Enc	Ch	Hash	Commu
Framework-A [12]	Issuing	customer	0	0	0	0	0	0	0	*
		merchant	0	0	0	0	2	3	2	
		server	0	0	0	0	4	2	0	
	Downloading & using	customer	0	0	0	0	2	3	2	
		merchant	0	0	0	0	1	0	0	
		server	0	0	0	0	3	2	0	
	Total	customer	0	0	0	0	2	3	2	
		merchant	0	0	0	0	3	3	2	
		server	0	0	0	0	7	4	0	
Framework-B [13]	Issuing & distribution	customer	0	0	0	0	0	0	0	*
		merchant	0	0	0	0	0	0	0	
		server	0	0	0	0	1	0	1	
	Using & verification	customer	0	0	0	0	0	0	0	
		merchant	0	0	0	0	1	0	1	
		server	0	0	0	0	0	0	0	
	Total	customer	0	0	0	0	0	0	0	
		merchant	0	0	0	0	1	0	1	
		server	0	0	0	0	1	0	1	
EQRC-I & -II	Generation	customer	0	0	0	0	0	0	0	0
		merchant	0	0	0	0	0	0	0	0
		server	1	1	1	2	0	0	0	0
	Distribution	customer	0	0	0	0	0	0	0	0
		merchant	0	0	0	0	0	0	0	0
		server	0	0	0	0	0	0	0	2
	Using & verification	customer	0	0	0	0	0	0	0	1
		merchant	0	0	0	0	0	0	0	1
		server	1	2	0	1	0	0	1	2
	Total	customer	0	0	0	0	0	0	0	1
		merchant	0	0	0	0	0	0	0	1
		server	2	3	1	3	0	0	1	4

† We keep the original terms used in the related papers so that readers can find and understand them easily. Note that although the names in frameworks are not exactly the same, the processes are matched. To be specific, we are considering the phases from preparing e-coupons to using it successfully. Registration is not involved. ‡ The operations are commitment, QR codes encoding or decoding, splitting, stacking, encrypting, chaotic mapping, hash, and necessary operations (i.e., encrypting and signing messages) for communication from left to right. Note that, we consider `Commu` as a whole without splitting it, because it is the common operations on the messages sent.

* Not mentioned in the references.

means that we proposed not only how e-coupons are wrapped with cryptography techniques (such as the anonymous signature), but also the components of e-coupons for security (i.e., the generation of the enhanced QR codes). Thus, users have much less computation overhead than the cloud server. However, the techniques are adopted outside of e-coupons directly for security in Framework-A and Framework-B.

As for communication, in EQRC-I and EQRC-II, the numbers of communication rounds for e-coupon redemption and verification are slightly larger than that in Framework-A and Framework-B. It is reasonable because each of the customer and the merchant only holds one of the pair $(\overline{SQR^1}, \overline{SQR^2})$ while both are necessary for verification. Though we likely have longer messages due to the QR-codes we use in the scheme, it is still acceptable and can be reduced via coding algorithms.

7. Further Discussion

EQRC has the ability to satisfy the essential security requirements, including data confidentiality, authentication, integrity and anonymity. Although it is proposed for e-coupon transactions, EQRC can be generalized for other scenarios easily. Some possible applications are described as follows.

(1) *Generalized e-coupons*: The proposed enhanced QR codes can be used as the generalized e-coupons, i.e., not only the common coupons with discounts, but also the cash coupons, gift cards, pre-paid membership cards, and rechargeable cards. The only task is to update the QR codes after redemption. For example, the server can update the amount of money in a rechargeable card after consumption or recharging by issuing a new pair of $(\overline{SQR^1}, \overline{SQR^2})$.

(2) *Payment cards for relatives*: Considering users are family members, we can apply the proposed framework to payment tasks. For example, the parents and their child hold one of the enhanced QR code pair $(\overline{SQR^1}, \overline{SQR^2})$, respectively, as fingerprints for transactions. To complete the payment, i.e., to recover $\overline{QR}$, the server needs to collect both of the QR codes. Thus, the child must get permission from the parents when paying. The fingerprints can defend against attacks such as forgery and modification by the triple-verification. As an additional payment method, it protects relatives, especially children and elders, from fraud and economic losses.

(3) *Certificates*: The enhanced QR codes can also be used as an aided verification of certificates, or even certificates directly. Compared with classic certificates, enhanced QR codes have more advantages. For example, a paper certificate only with a stamp is easy to forge. However, the necessary information such as the name of the awardee and the awards can be stored in the enhanced QR codes, which satisfy the requirements of integrity and authenticity. In addition, the holders of certificates may not trust the issuing institutions. With the enhanced QR codes, each of the holder and the institution only keep one of $(\overline{SQR^1}, \overline{SQR^2})$, respectively. Any tampering or forgery can be figured out and traced by the server.

(4) *Health barcodes*: As a special case of certificates, health barcodes can be well used when citizens are exposed to infectious disease seriously, such as COVID-19. People with different health codes have different access to community activities. The design of enhanced QR codes and the triple verification guarantee that it is infeasible to tamper or forge the health barcodes. It provides a social safety guarantee in difficult time.

Although EQRC satisfies the security goals with good performance, there are still some remaining problems. Some of the problems are open questions in related research. The detailed discussions are as follows.

(1) *The embedding position*: Based on the Reed-Solomon error correction code, $\overline{QR^1}$ and $\overline{QR^2}$ can be embedded into $\overline{CQR}$. However, the process should be considered more carefully because there are functional elements in a QR code. These functional elements ensure that a scanning device can

correctly identify and decode QR codes. For example, the three position patterns placed at corners are used to define the location of the QR code. Thus, the embedding must avoid destroying the functional elements.

Because the elements have fixed shapes, colors and positions in standard QR codes, the detection is not a problem. In fact, the elements are first detected after scanning, easily, quickly and accurately [46]. In addition, a recently developed QR code called Frame QR has a region where the arbitrary altering of figures and contents will not affect other regions. Combining Frame QR into the design of our EQRC can make the generation and merging of enhanced QR codes more standardized and concise.

(2) *The size of enhanced QR codes*: Comparing with some classic e-coupon systems, which only use a bit string as e-coupons, EQRC has larger e-coupons in size. This is also a problem for other QR code-based innovative applications [14, 16].

Considering the network performance nowadays, it is not a significant limitation. A possible solution is to increase the capability of QR codes, which is a hot topic for researchers. With higher capability, a smaller QR code can be used for coding the same amount of data. One mature scheme is the colored QR codes. In EQRC, using colored QR codes to implement the fragment coding of QR codes, can provide more storage space. Thus, it will be an attractive research direction for us in the future.

(3) *Copies on QR codes*: An illegitimate copy on e-coupons has the potential to infringe the rights of legitimate holders. It is an open question for QR codes for a long time. Because QR codes are usually used on smartphones, the risk of illegal copies by capturing the screen or taking pictures is high [47]. We cannot ensure that users' phones are not accessible from attackers. Thus, it is hard to avoid copies.

If the QR code is linked with the identity of the legal holder, the problem can be solved by checking the signature. From this point of view, EQRC can be extended to e-coupon services provided for particular users. Another possible solution is to check the freshness. For example, Alipay refreshes the payment QR codes every minute. However, both of the solutions above cannot solve the problem thoroughly.

(4) *Privacy issues*: Although EQRC provides the identity anonymity and audit trail for users by the TAA scheme, there are many other privacy issues in practice. For example, 1) Once a user is determined dishonest, all the actions of the user in previous transactions are expected to be revealed. Some latest oblivious transfer schemes can be considered [10]. 2) User behaviors are usually studied for improving economic efficiency. However, how to protect the user's privacy while aggregating the statistical information is also a problem. One possible solution is the differential privacy, which withholds the information of individuals while revealing the patterns of groups.

(5) *Portability*: As EQRC, most of the work on e-coupon systems is self-contained. However, a problem is whether the protocols can be successfully integrated with existing mature applications, such as Walmart, Amazon, Paypal, Aliexpress and Alipay. The difficulty of the work is that, interfaces of many mature shopping and payment applications are not public for researchers. In addition, to make the proposed framework user-friendly and practical, many details should be considered more carefully, including processing interruptions and communication interruptions. Strengthened cooperation between academic and industrial communities may provide more development space.

8. Conclusion

In this paper, we first proposed a fragment-based approach to enhance the confidentiality of QR codes. Second, we designed an enhanced QR code scheme with the combination of the QRFC approach and the commitment technique. The enhanced QR code scheme can prevent the leakage of the sensitive information in QR codes and forgery or tampering of QR codes. Third, we gave a secure e-coupon transaction framework called EQRC based on the techniques above. EQRC provides a triple-verification mechanism, reducing the security threats during the e-coupon delivery and transaction. Both online and offline scenarios are supported by EQRC, which provides a comprehensive protection for the real situation. The strong analyses and evaluation have shown that the proposed framework has a high security and low computing and communication overhead.

References

[1] R. Liu, J. Song, Z. Huang and J. Pan, EQRC: An enhanced QR code-based secure e-coupon transaction framework, in: *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, 2019, pp. 1–6.

[2] R. Focardi, F.L. Luccio and H.A. Wahsheh, Usable security for QR code, *Journal of Information Security and Applications* **48** (2019), 102369.

[3] F. Cadger, K. Curran, J. Santos and S. Moffett, A survey of geographical routing in wireless ad-hoc networks, *IEEE Communications Surveys & Tutorials* **15**(2) (2013), 621–653.

[4] A. Kharraz, E. Kirda, W. Robertson, D. Balzarotti and A. Francillon, Optical delusions: a study of malicious QR codes in the wild, in: *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, IEEE, 2014, pp. 192–203.

[5] K. Krombholz, P. Frühwirt, T. Rieder, I. Kapsalis, J. Ullrich and E. Weippl, QR code security—How secure and usable apps can protect users against malicious QR codes, in: *2015 10th International Conference on Availability, Reliability and Security*, IEEE, 2015, pp. 230–237.

[6] V. Mavroeidis and M. Nicho, Quick response code secure: a cryptographically secure anti-phishing tool for QR code attacks, in: *International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security*, Springer, 2017, pp. 313–324.

[7] X. Zhang, H. Li, Y. Yang, G. Sun and G. Chen, LIPPS: Logistics information privacy protection system based on encrypted QR code, in: *Trustcom/BigDataSE/ISPA*, IEEE, 2016, pp. 996–1000.

[8] S. Sharma and V. Sejwar, Impementation of QR code based secure system for information sharing using Matlab, in: *2016 8th International Conference on Computational Intelligence and Communication Networks (CICN)*, IEEE, 2016, pp. 294–297.

[9] Z.-F. Yan, Y.-L. Shen, W.-J. Liu, J.-M. Long and Q. Wei, An e-commerce coupon target population positioning model based on random forest and extreme gradient boosting, in: *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, IEEE, 2018, pp. 1–5.

[10] W. Liu, Y. Mu, G. Yang and Y. Yu, Efficient E-coupon systems with strong user privacy, *Telecommunication Systems* **64**(4) (2017), 695–708.

[11] J. He and W. Jiang, Understanding users' coupon usage behaviors in e-commerce environments, in: *2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC)*, IEEE, 2017, pp. 1047–1053.

[12] C.-C. Chang and C.-Y. Sun, A secure and efficient authentication scheme for e-coupon systems, *Wireless Personal Communications* **77**(4) (2014), 2981–2996.

[13] C.-C. Chang, I.-C. Lin and Y.-L. Chi, Secure electronic coupons, in: *2015 10th Asia Joint Conference on Information Security*, IEEE, 2015, pp. 104–109.

[14] Y. Cheng, Z. Fu and B. Yu, Improved visual secret sharing scheme for QR code applications, *IEEE Transactions on Information Forensics and Security* **13**(9) (2018), 2393–2403.

[15] P. Lin and Y. Chen, High payload secret hiding technology for QR codes, *EURASIP Journal on Image and Video Processing* **2017**(1) (2017), 14–21.

[16] I. Tkachenko, W. Puech, C. Destruel, O. Strauss, J. Gaudin and C. Guichard, Two-level QR code for private message sharing and document authentication, *IEEE Transactions on Information Forensics and Security* **11**(3) (2016), 571–583.

[17] C. Zhang, X. Lin, R. Lu, P. Ho and X. Shen, An efficient message authentication scheme for vehicular communications, *IEEE Transactions on Vehicular Technology* **57**(6) (2008), 3357–3368.

[18] C. Lee and Y. Lai, Toward a secure batch verification with group testing for VANET, *Wireless Networks* **19**(6) (2013), 1441–1449.
[19] R. Hasan, R. Sion and M. Winslett, The case of the fake picasso: preventing history forgery with secure provenance., in: *Proccedings of the 7th Conference on File and Storage Technologies*, Vol. 9, USENIX Association, 2009, pp. 1–14.
[20] E. Brickell, J. Camenisch and L. Chen, Direct anonymous attestation, in: *Proceedings of the 11th ACM Conference on Computer and Communications Security*, ACM, 2004, pp. 132–145.
[21] L. Chen, P. Morrissey and N.P. Smart, DAA: Fixing the pairing based protocols, *Cryptology ePrint Archive* (2009).
[22] L. Chen, S. Ng and G. Wang, Threshold anonymous announcement in VANETs, *IEEE Journal on Selected Areas in Communications* **29**(3) (2011), 605–615.
[23] M. Naor and A. Shamir, Visual cryptography, in: *Workshop on the Theory and Application of of Cryptographic Techniques*, Springer, 1994, pp. 1–12.
[24] O. Goldreich, *Foundations of Cryptography*, Cambridge University Press, 2007.
[25] L. Zhang, J. Song and J. Pan, A privacy-preserving and secure framework for opportunistic routing in DTNs, *IEEE Transactions on Vehicular Technology* **65**(9) (2015), 7684–7697.
[26] A.J. Menezes, J. Katz, P.C. Van Oorschot and S.A. Vanstone, *Handbook of Applied Cryptography*, CRC press, 1996.
[27] Wikstr and M. Douglas, A commitment-consistent proof of a shuffle, in: *Information Security & Privacy, Australasian Conference, Australia,*, 2009.
[28] S. Kelly and S. Frankel, Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec, Technical Report, 2007.
[29] S. Turner and L. Chen, Updated security considerations for the MD5 message-digest and the HMAC-MD5 algorithms, Technical Report, 2011.
[30] P. Gauravaram, W. Millan, E. Dawson and K. Viswanathan, Constructing secure hash functions by enhancing Merkle-Damgård construction, in: *Australasian Conference on Information Security and Privacy*, Springer, 2006, pp. 407–420.
[31] J. Black, P. Rogaway and T. Shrimpton, Black-box analysis of the block-cipher-based hash-function constructions from PGV, in: *Annual International Cryptology Conference*, Springer, 2002, pp. 320–335.
[32] H. Gilbert and H. Handschuh, Security analysis of SHA-256 and sisters, in: *International Workshop on Selected Areas in Cryptography*, Springer, 2003, pp. 175–193.
[33] F. Mendel, T. Nad and M. Schläffer, Improving local collisions: new attacks on reduced SHA-256, in: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, 2013, pp. 262–278.
[34] M. Eichlseder, F. Mendel and M. Schläffer, Branching heuristics in differential collision search with applications to SHA-512, in: *International Workshop on Fast Software Encryption*, Springer, 2014, pp. 473–488.
[35] C. Dobraunig, M. Eichlseder and F. Mendel, Analysis of SHA-512/224 and SHA-512/256, in: *International Conference on the Theory and Application of Cryptology and Information Security*, Springer, 2015, pp. 612–630.
[36] M. Scott, *Efficient implementation of cryptographic pairings*, 2007. http://www.pairing-conference.org/2007/invited/Scottslide.pdf.
[37] X. Lin, X. Sun, P.-H. Ho and X. Shen, GSIS: A secure and privacy-preserving protocol for vehicular communications, *IEEE Transactions on Vehicular Technology* **56**(6) (2007), 3442–3456.
[38] *OMNeT++*. https://omnetpp.org.
[39] *INET Framework*. https://inet.omnetpp.org.
[40] R. Tarabuţă, D. Balan, A. Potorac and A. Graur, Performance investigation over 802.11 ac communication environment, in: *2016 22nd International Conference on Applied Electromagnetics and Communications (ICECOM)*, IEEE, 2016, pp. 1–5.
[41] E. Lopez Aguilera, E. Garcia Villegas and J. Casademont, Evaluation of IEEE 802.11 coexistence in WLAN deployments, *Wireless Networks* **25**(1) (2019), 87–104.
[42] *Geography - Statistics Canada*, 2016. https://www150.statcan.gc.ca/n1/pub/11-402-x/2012000/chap/geo/geo-eng.htm.
[43] D. Wave, Information technology automatic identification and data capture techniques QR code bar code symbology specification, *International Organization for Standardization, ISO/IEC* **18004** (2015).
[44] V. Hajduk, M. Broda, O. Kováč and D. Levickỳ, Image steganography with using QR code and cryptography, in: *2016 26th International Conference Radioelektronika (RADIOELEKTRONIKA)*, IEEE, 2016, pp. 350–353.
[45] X. Nan, Y. He and L. Guan, Optimal resource allocation for multimedia cloud based on queuing model, in: *2011 IEEE 13th International Workshop on Multimedia Signal Processing*, IEEE, 2011, pp. 1–6.
[46] S. Tiwari, An introduction to QR code technology, in: *2016 International Conference on Information Technology (ICIT)*, IEEE, 2016, pp. 39–44.
[47] S. Sung, J. Lee, J. Kim, J. Mun and D. Won, Security analysis of mobile authentication using QR-codes, in: *Computer Science & Information Technology-Computer Science Conference Proceedings*, 2015.

Close

JCS-191416R1
"EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions"
Revision 1

Yesi Novaria Kunang, M.Kom. (Reviewer 1)

Reviewer Recommendation Term:	Accept as is
Overall Reviewer Manuscript Rating:	87

Custom Review Question(s):	**Response**
Are you willing to review the revision of this manuscript?	Yes
Originality, novelty and significance of results:	Good
Technical Quality of Work:	Good
Comprehensibility and Presentation of Paper:	Excellent
What is the overall impression:	Good

Manuscript Rating Question(s):	**Scale**	**Rating**
Is the article significant to the field?	[1-5]	2
Is the article appropriate for the journal?	[1-5]	3
Is the work an original contribution?	[1-5]	3
Are the conclusions adequately supported by the data?	[1-5]	3
Is the research interesting and important?	[1-5]	3
Is the level of English adequate?	[1-5]	4

Comments to Editor:

There is no a financial or other conflict of interest between my work and that of the authors.

Please give a frank account of the strengths and weaknesses of the article:

Strengths:
The revision manuscript reported by Liu et al. enhance QR code and its applications in various ways that can be adopted in e-coupon payment services. Most of the study is well designed, detailed, and scientific value is given enough. For this reason, the reviewer recommends the manuscript can be accepted without any improvement

Weakness:
Some weaknesses, such as uncertain factors and difficulties to simulate attacks directly in their current work it is very understandable and can still be developed in the future research by the author or by other researchers

Comments to Author:

Thank you for the response and revision that has been done.

I'm sure the authors have done more work and added some significant value.
Some limitations, such as limitations to simulate all attacks, are understandable. In the future, development can be carried out to conduct comprehensive experiments.

Close

View Reviewer Comments for Manuscript
JCS-191416R1
"EQRC: A Secure QR Code-based E-coupon Framework Supporting Online and Offline Transactions"

Click the Reviewer recommendation term to view the Reviewer comments.

	Revision 1	Original Submission
Yesi Novaria Kunang, M.Kom. (Reviewer 1)	Accept as is	Accepted pending minor revisions
(Reviewer 2)	Accept as is	Accepted pending minor revisions
(Reviewer 3)	(None)	Accepted pending minor revisions
Author Decision Letter	Accept as is	Revise and resubmit pending major revisions
Author		Response to Reviewers