A New Secure Publication Subscription Framework with Multiple Arbitrators

Shugo Yoshimura
Graduate School of Information Sci.
and Electrical Eng., Kyushu Univ.
Fukuoka, Japan
yoshimura.shugo.822@s.kyushu-u.ac.jp

Kouki Inoue
Graduate School of Information Sci.
and Electrical Eng., Kyushu Univ.
Fukuoka, Japan
inoue.kouki.882@s.kyushu-u.ac.jp

Dirceu Cavendish
Graduate School of Eng.
Kyushu Institute of Tech.
Iizuka, Japan
cavendish@ndrc.kyutech.ac.jp

Hiroshi Koide
Research Institute of Info. Tech.,
Kyushu Univ.
Fukuoka, Japan
koide@cc.kyushu-u.ac.jp

Abstract— **In this study, to make it easy for everyone to distinguish the right information from the wrong information, we suggest a new framework (Secure Publication Subscription Framework) that defines the reliability of publishers and provides it to subscribers. Nowadays, services like blogs and social media make available large amounts of information easily. On the other hand, there is a lot of unreliable information on the Internet. It is difficult to distinguish between true and false information. This problem is known as fake news and has become a serious problem. To solve this problem, we suggest a new framework for publishers and subscribers. The framework allows subscribers to easily confirm the authenticity of information by registering publishers and subscribers, and tracking publishers' reputation via a reputation score, guaranteeing the quality of the information that subscribers view. In this study, we show a proof of concept of a simple Secure Publication Subscription Framework and confirm that it is possible to implement a framework with the proposed functionality. We also confirm that the reputation score can be used as an indicator of the reliability of the information by using 1000 randomly generated articles within the framework. In addition, We also proposed three models of how to incorporate multiple Arbitrators to be considered when realizing this framework.**

Keywords-dissemination; publication; social networking; authenticity of information; reputation score.

I. Introduction

In our previous research [1], we proposed a Secure Publication Subscription Framework that allows subscribers to easily confirm the authenticity of the information and provides the publisher's reputation score. It consists of three parts, Publisher, Arbitrator, and Subscriber. The Subscriber can request the information challenge to the Arbitrator, and the Arbitrator verifies data truthfulness. A reputation score describes the Publishers' truthfulness and is increased or decreased according to the authenticity of the Publishers' information. We conducted experiments to confirm that the reputation score can be an indicator of the reliability of the Publishers. In this paper, we also include a model with multiple Arbitrators, considering the construction of a practical system. We propose three models for setting multiple Arbitrators. The merits, demerits, and conditions under which they should be used are discussed for each mechanism, reinforcing the realism of this framework.

In recent years, Internet technologies have made great progress, with the population of Internet users increasing rapidly. Thanks to services like blogs and social media, anyone can get a large amount of information easily. Nowadays, we can be aware of what is happening around the world, no matter where we are.

On the other hand, there is a lot of unreliable information on the Internet. It is difficult to distinguish between true and false information. This problem is known as fake news and has become a serious problem. Fake news is fabricated information that mimics news media content in form but not in organizational process or intent [2]. It is not just a prank, but a serious problem. As an example, during the 2016 United status presidential election, fake news was highly used and had a big impact on Twitter [3] [4].

To solve this problem, we suggest a new framework for publishers and subscribers. This framework allows subscribers to easily confirm the authenticity of information by registering publishers and subscribers, guaranteeing the publisher of the information that subscribers view, checking the information challenge from subscribers, and providing the publisher's reputation score that increases or decreases as a result of the authenticity of the information.

This framework consists of three parts, Publisher, Subscriber and Arbitrator. The main role of the Publisher is publishing articles or news. The Subscriber registers with the Publisher and subscribes for publications. The Arbitrator provides the Publisher's reputation and verifies the information challenge from the Subscriber.

The paper is organized as follows. Related work is in-

This study is supported by JSPS KAKENHI Grant Number 21K11888 and Hitachi Systems, Ltd.

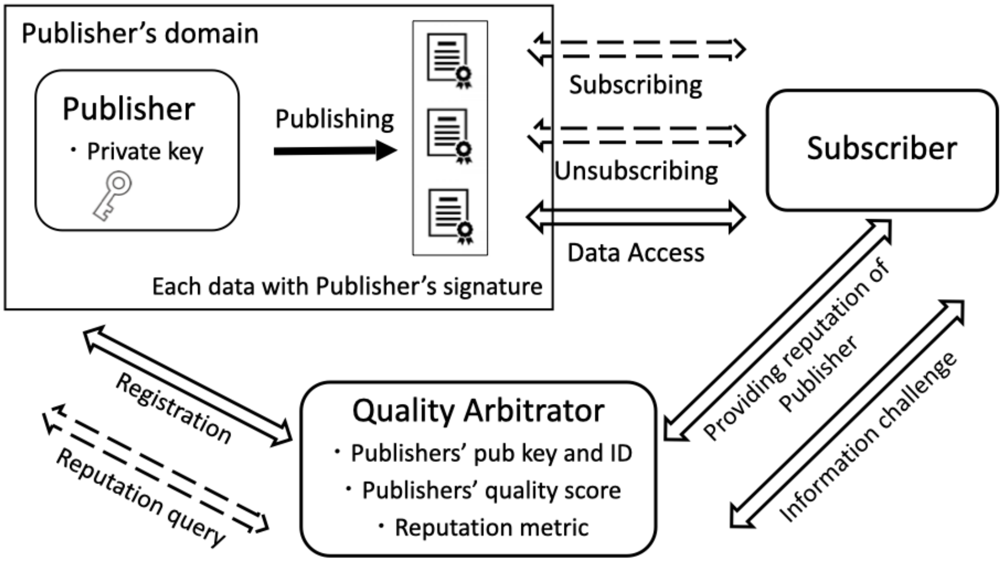

Figure 1. Secure Publication/Subscriber Architecture

cluded in Section II. Section III describes our proposed secure publication/subscription reference model. Section IV describes a proof of concept implementation of the reference model. Section V describes two experiments used to track the performance of the proposed publication/subscription model. Section VI presents the performance results and discussions. Section VII proposed three models for how to incorporate multiple arbitrators to be considered when implementing this framework, and discusses the advantages and disadvantages of each model. Section VIII summarizes our studies and addresses directions we are pursuing as follow up to this work.

II. Related work

Previous research on publication/subscription systems have covered various areas, such as security, confidentiality and scalability.

Nakamura and Enokido [5] focused on a peer to peer publication/subscription model where multiple topics are supported. In that work, they propose a subscription initialization protocol to ensure that peers not authorized to have access to topics do not have access to them. They do not address the quality of the information exchanged within topics. In contrast, our framework addresses information quality on a generic publication/subscription architecture, not necessarily requiring a peer to peer model.

Salem [6] addresses the problem of authenticating users of a pub/sub system containing a message broker in a privacy-preserving way. The proposal supports mutual authentication in a scalable way, and may be adopted by pub/sub systems with a broker. In contrast, our work does not focus on anonymity of publishers/subscribers, although our pub/sub model could be adapted to include a broker, if necessary.

In Srivatsa [7], a secure event dissemination protocol is proposed where encryption and authorization keys are used on top of an IP network that does not provide confidentiality nor integrity of data. In contrast, although our pub/sub model supports integrity verification of data, our focus is on the control of the quality of data published.

Bovet and Makse [4] describe an information ranking mechanism to fight unreliable (spam) data in a pub/sub system model with a broker reference architecture. They propose to rank information as a way to avoid blacklisting. However, their ranking system is still based on participants' voting. Although the purpose of the research is similar to ours, our solution to control quality of disseminated data is based on an arbitrator that is supposed to be able to verify data quality on specific domains, rather than relying on voting.

III. Secure Publication/Subscription

This section describes the operation of the Secure Publication Subscription Framework in detail.

Figure 1 describes our proposed secure publication/subscription system architecture. Multiple publishers provide signed data contents to consumers, or subscribers. Data content quality is tracked by an independent quality arbitrator. The quality arbitrator provides publishers' reputation to subscribers. Also, the arbitrator may receive data truthfulness challenges from subscribers.

A. Sec Pub/Sub Components

Figure 2 illustrates how Publishers provide signed data contents. Publishers also produce a digest of the data content using standard asymmetric cryptography, using their private key to ensure data integrity.

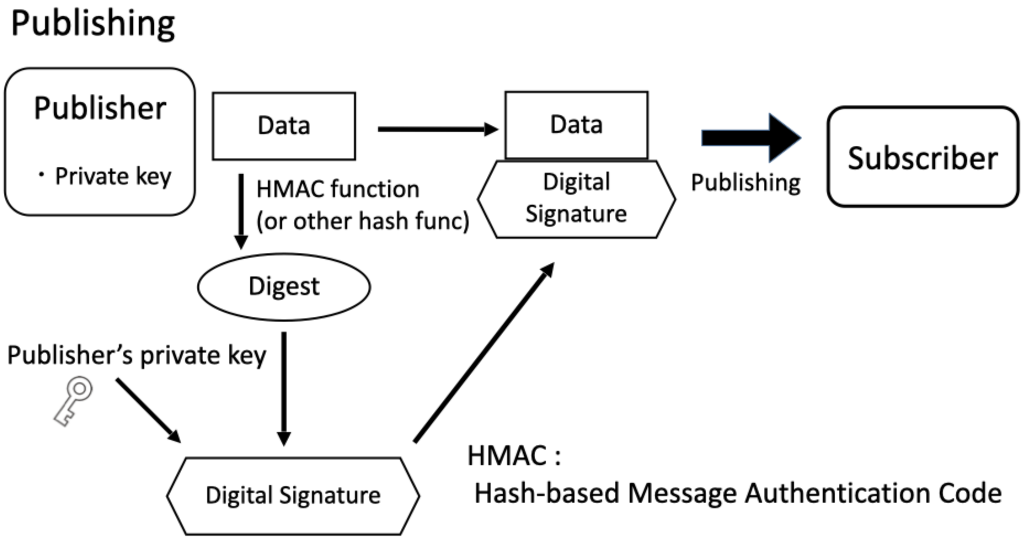

Figure 2. Signed publishing

Figure 3 illustrates publisher/subscriber interfaces. The subscriber requests subscription services from a publisher and receives the publisher public key used to verify data authenticity. Once the subscription service has been agreed upon, an information retrieval interface is used to request signed data from the publisher.

Figure 4 illustrates the subscriber's data processing of published data. Data processing includes data integrity verification and confirmation authorship. The subscriber verifies the digital signature and the digest of the data, using the publisher public key. In this process, the subscriber verifies the integrity of the received data and confirms the data's authorship.

Figure 5 illustrates publisher reputation tracking feature of the secure pub/sub framework. Each publisher registers first with the quality arbitrator, upon which its public key is passed to the arbitrator. The arbitrator then tests the publisher's possession of the corresponding private key as part of the registration. Each successfully registered publisher is associated with a reputation score metric, which can be queried by both the publisher itself as well as subscribers.

Figure 6 illustrates the subscriber/quality arbitrator interfaces. Subscribers can request publisher's reputation score from the arbitrator. In addition, subscribers can challenge publisher's trustfulness for each data received. The quality arbitrator, upon receiving the challenge, verifies data truthfulness, and adjusts the publisher reputation score according with data verification status.

B. Reputation Algorithm

The reputation score of a publisher is defined as

$$score = \frac{the\ number\ of\ correct\ data}{the\ number\ of\ all\ published\ data}.$$

However, as the quality arbitrator may not estimate correctly every and all data published, we introduce a noise model for data verification, as shown in Figure 7. In the model, p is the probability that a true piece of data be recognized as false, whereas q represents the probability of a false piece of information be admitted as true. In the experimental section, we exemplify the arbitrator score reputation tracking on two publisher scenarios: i- trusted publisher (all data is truthful); ii- untrusted publisher; Publisher produces up to 1000 data pieces (the data can be right or wrong).

IV. Implementation

In this section, we describe an overview of the implementation of Publisher, Arbitrator, Subscriber. We implemented the Publisher and the Arbitrator with Node.js and Express that is a JavaScript Web framework, and we implemented the Subscriber with Python3. The Publisher and the Arbitrator operate like a Web server, independently, and the Subscriber accesses them according to the scenarios. The versions used in the implementation are summarized in Table I.

TABLE I
Implementation

Application	Version
Node.js	12
MySQL	5.7
Python	3.9.12

A. Publisher

The Publisher is implemented with Node.js and Express, and it operates as a Web server. Figure 8 describes the im-

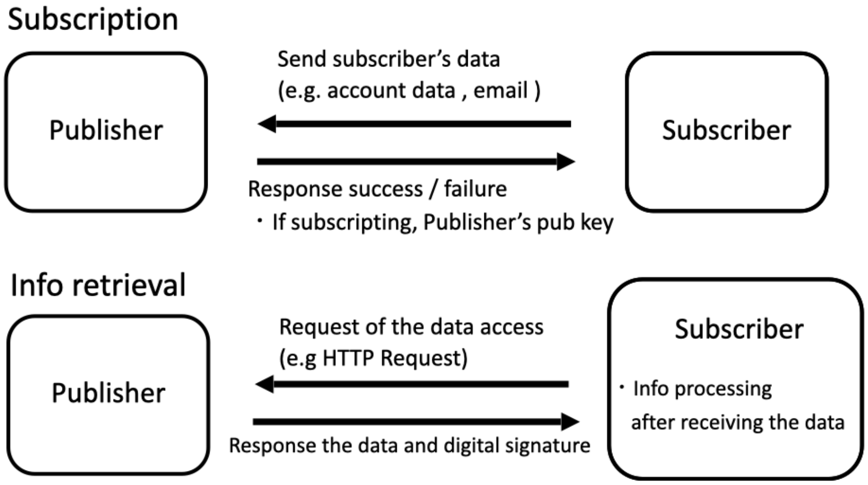
Subscription
Publisher
Send subscriber's data
(e.g. account data , email)
Subscriber
Response success / failure
・If subscripting, Publisher's pub key
Info retrieval
Publisher
Request of the data access
(e.g HTTP Request)
Subscriber
・Info processing
after receiving the data
Response the data and digital signature

Figure 3. Subscription and Information Retrieval

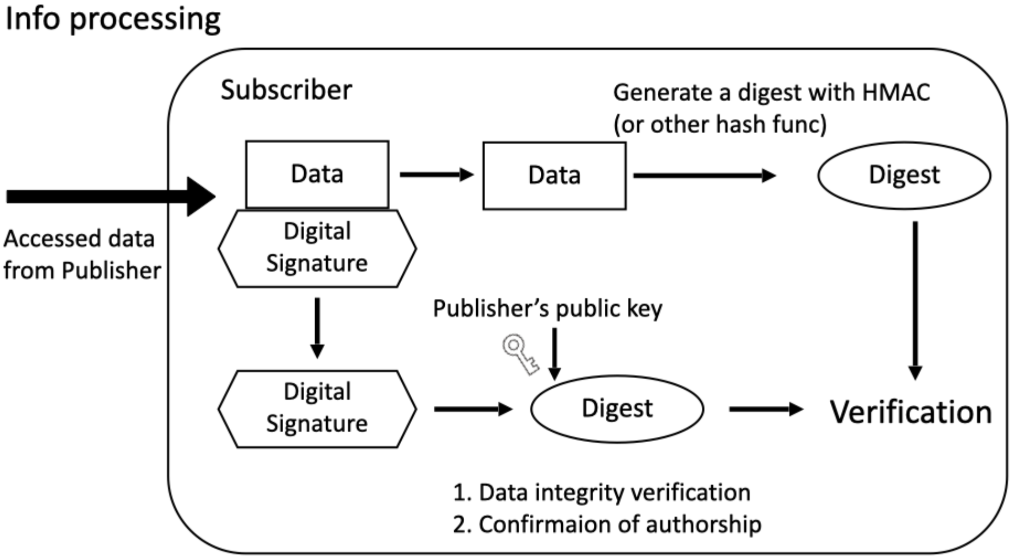

Figure 4. Data Integrity Verification

plementation. The Publisher has subscriber registration, login, some data pages and digital signatures. In addition, it has a MySQL database that saves the Subscriber's name and hashed password. If it receives an HTTP Request from the Subscriber, it replies with an HTTP Response and sends the data.

B. Arbitrator

The Arbitrator is also implemented with Node.js and Express, and operates as a Web server. Figure 9 describes the implementation of the Arbitrator. The Arbitrator receives the Publisher's registration, reputation query, as well as information challenge and request for publisher's public key. Additionally, the Arbitrator supports a MySQL database, which saves the Publisher's name, password, public key and Publisher reputation score. Firstly, the Publisher registers its name, password and public key. In our experiment scenarios, the Publisher's information is saved in initial state, so this step is omitted. If the Subscriber requests the Publisher's public key, the Arbitrator responds to it. If the Subscriber requests the Publisher's reputation score, the Arbitrator sends the Publisher's score. If the Arbitrator receives an information challenge from the Subscriber, it verifies data truthfulness, updates the score of the Publisher.

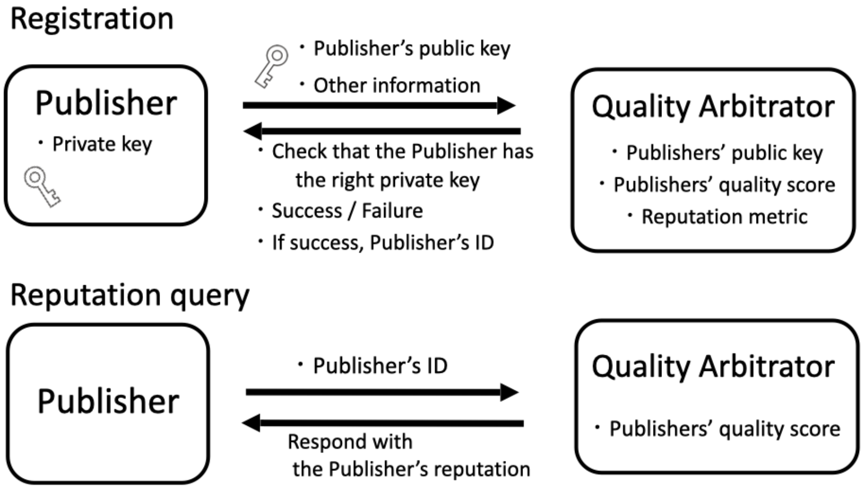

Figure 5. Publisher registration and Reputation Tracking

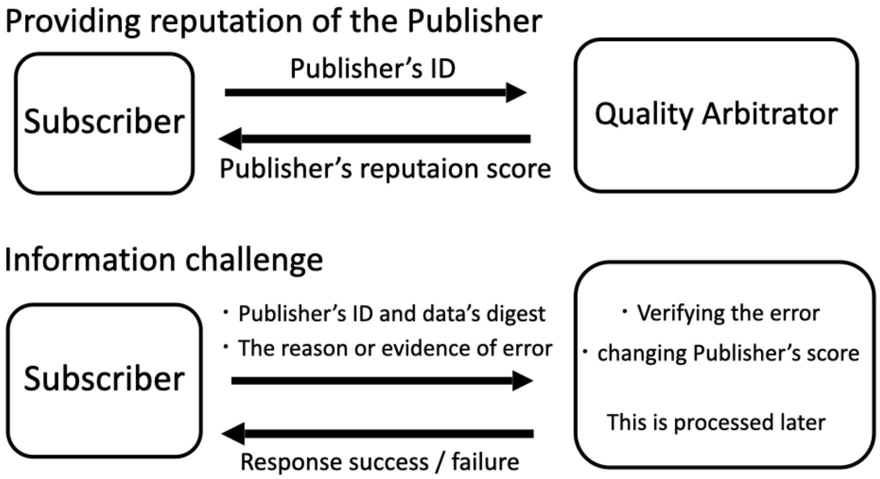
Providing reputation of the Publisher
Subscriber
Publisher's ID
Publisher's reputaion score
Quality Arbitrator
Information challenge
Subscriber
・Publisher's ID and data's digest
・The reason or evidence of error
Response success / failure
・Verifying the error
・changing Publisher's score
This is processed later

Figure 6. Reputation service interface

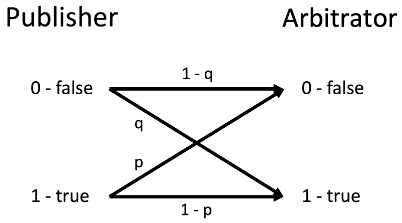

Figure 7. Noisy Channel Model

C. Subscriber

The Subscriber is implemented with Python3. It accesses the Publisher and the Arbitrator according to the different scenarios. During information processing, it verifies the integrity of received data and confirms data authorship (Figure 10).

V. Experiment

This section demonstrates the evolution of the reputation estimator and reputation score for the Secure Publication Subscription Framework using 1000 randomly generated true and false data.

The resulting graph shows 3 lines:

- Actual reputation score: the reputation score actually obtained after going through the Secure Publication Subscription Framework,
- Expected reputation score: the expected value of the reputation score obtained from the actual truth of the data, p and q,

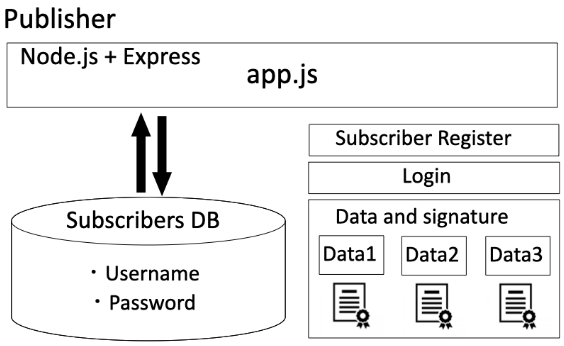

Figure 8. Publisher

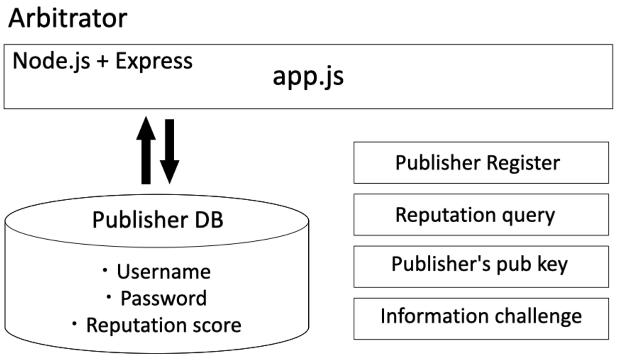

Figure 9. Arbitrator

- True reputation: proportion of data that is actually true.

We illustrate the secure publication/subscription model with the following scenarios:

A. Scenario 1

1) Subscribers register and login in with the Publisher
2) Subscribers subscribe to data from the Publisher
3) Subscribers retrieve the data
4) Subscribers send a query about the Publisher's reputation to the Arbitrator

In Scenario 1, the credibility of the Publisher's data is 100%, hence the Publisher's true reputation is 1. However, the expected reputation score is

$$1 - p$$

because there is a possibility that the Arbitrator will judge it to be false. In this experiment, the values of the p and q are set to 0.3 to check the reputation scores. To show that the accuracy of the reputation score does not drop even if the accuracy of the true/false discrimination is not so high, p and q were set to fairly low values. We think that there is still room for further study on this value.

Figure 11 shows the graph of the results for Scenario 1.

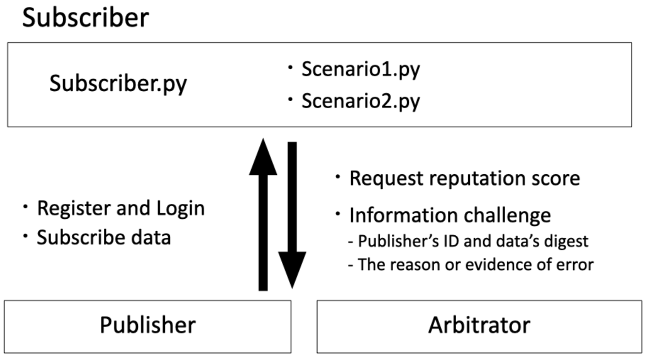

Figure 10. Subscriber

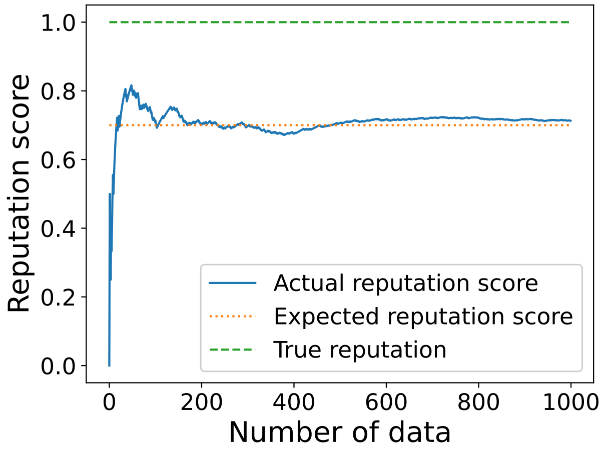

Figure 11. scenario 1

B. Scenario 2

In scenario 2, Publisher's data is not always true.

1) Subscribers register and login in with the Publisher
2) Subscribers subscribe to data from the Publisher
3) Subscribers retrieve the data
4) Subscribers issue an information challenge
5) The Arbitrator decides the data as false, and updates the Publisher's reputation
6) Subscribers query the reputation of the Publisher from the Arbitrator

Let a be the probability that the publisher's data is false. Then, the expected value of the true reputation is

$$1 - a,$$

while the expected reputation score is

$$a * q + (1 - a) * (1 - p).$$

In Scenario 2, step 1, 2, 3 are the same as in Scenario 1. However, the Subscriber carries out an information challenge in steps 4 and 5. The probability of judging the data to be

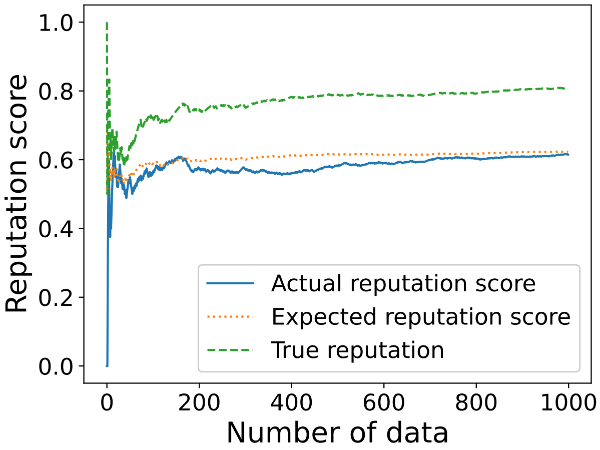

Figure 12. scenario2 data accuracy = 0.8

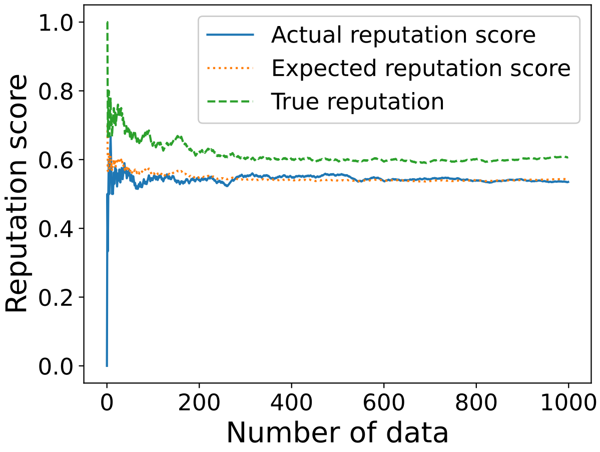

Figure 13. scenario2 data accuracy = 0.6

correct was varied between 0.8 and 0.6, and p and q were 0.3 to check the reputation scores for each case.

The experimental results are shown in Figures 12 and 13.

VI. Performance Analysis

In this section, we present the reputation tracking results of our secure pub/sub system. In scenario 1, the final three scores obtained from the 1000 data points are shown in Table II.

TABLE II
SCENARIO 1

Actual reputation score	0.713
Expected reputation score	0.700
True reputation	1.000

In scenario 2, the final three scores obtained from the 1000 data points are shown in Tables III and IV.

TABLE III
SCENARIO 2 DATA ACCURACY = 0.8

Actual reputation score	0.615
Expected reputation score	0.623
True reputation	0.808

TABLE IV
SCENARIO 2 DATA ACCURACY = 0.6

Actual reputation score	0.535
Expected reputation score	0.543
True reputation	0.607

From these experimental results, with a sufficient number of data points and a certain degree of accuracy in determining the truth of the data, we see that the actual reputation score converges to the expected reputation score.

Moreover, we use a noise model for data verification, and we define the expected reputation to be

$$a * q + (1 - a) * (1 - p).$$

So, if p and q are known, the Publisher's true reputation can be estimated from the actual score.

These results indicate that the reputation score is closely related to the probability of the correctness of the data (credibility) and that the actual reputation score can be calculated with considerable accuracy if p and q are known.

The result shows that the reputation score is a sufficiently reliable value for easily confirming the credibility of the Publisher.

VII. Incorporation of multiple Arbitrators

Although we were able to confirm that the reputation score is related to the credibility of the publisher in the proposed framework, there are still some problems to be solved in actual operation. One of the problems is that it is not realistic for a single arbitrator to handle all of the enormous amounts of info challenges. To solve this problem, multiple Arbitrators can be used instead of a single Arbitrator to perform fact-checking. However, there are various problems associated with this method, such as the sharing of secret keys and reputation scores.

In this section, we propose three mechanisms for setting up multiple Arbitrators. The merits, demerits, and conditions under which they should be used are discussed for each mechanism.

A. Basic method

In this model, each arbitrator maintains the same database that contains the data of all the Publishers, and it is necessary to rewrite the information in the database in case of registration of a Publisher, information challenge from a Subscriber, etc. while synchronizing with the other Arbitrators. The overall diagram is shown in Figure 14. The explanation is based on the case of two Arbitrators, but the same operation can be performed even if the number of Arbitrators is larger.

The operation of Publisher registration is as follows.

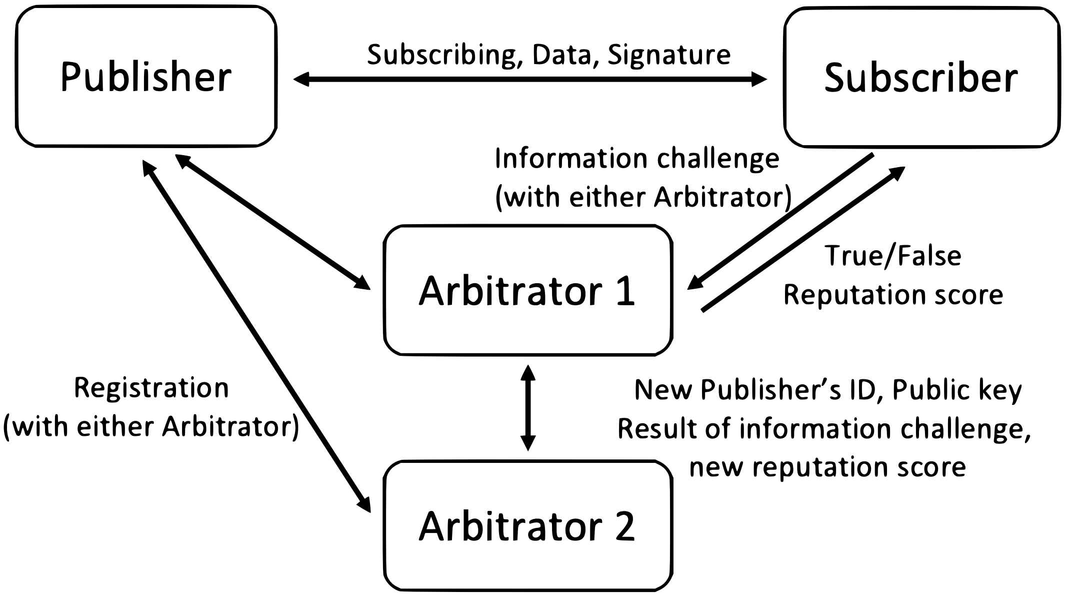

Figure 14. basic method

1) The Publisher selects one of the two Arbitrators and sends its public key and other information.
2) The selected Arbitrator verifies the key. If the key is invalid, it sends a message to the Publisher and terminates the operation.
3) If the key is OK, it shares the public key and other information with the other Arbitrator and updates the database.
4) The Publisher is notified that the registration has been completed.

This is how it works in the case of an information challenge.

1) The Subscriber selects one of the two Arbitrators to perform the information challenge.
2) The selected Arbitrator verifies the signature and performs a fact check.
3) The result of the fact check and the new reputation score is shared with the other Arbitrator, and the database is updated.
4) The results of the fact check and the new reputation score are sent to the Subscriber.

The advantage of this model is redundancy. If one Arbitrator becomes unavailable, another Arbitrator can be substituted and the entire system will not become unavailable. This model is suitable when availability at any time is important.

There are three possible disadvantages of this model.

- The application address for information challenge by the Subscriber when the Publisher registers
  In the past, there was only one Arbitrator, so there was no need to worry about where to submit applications, but in this model, there are two Arbitrators, so the Publisher and Subscriber must choose one or the other, or submit to both.
- Sharing of publisher information and reputation score
  For example, if one of the Arbitrator performs an information challenge and the reputation score of the Publisher changes, the other Arbitrator will be notified that the information challenge was performed and that the Publisher's reputation score has changed. The results of the information challenge and the new reputation score need to be shared with the other Arbitrator. When updating the database is necessary, it must be handled in such a way that it does not cause errors in the synchronization process.
- Sharing of publisher information and public keys
  Arbitrator needs to verify whether an article is written by the correct Publisher at the time of information challenge. Therefore, all Arbitrators must maintain the IDs and public keys of all Publishers, which is inefficient.

B. Combination of specific Arbitrator and Publishers

This model is a method that eliminates the need to share reputation scores and keys with other Arbitrators by linking the Publisher to a specific Arbitrator. The overall diagram is shown in Figure 15.

In this model, the Publisher selects which Arbitrator he/she belongs to and applies for registration to that Arbitrator. In addition, when making an information challenge, the Subscriber must send it to the Arbitrator to which the Publisher of the article belongs. Therefore, it is necessary to indicate which Arbitrator the Publisher belongs to in the article. In this model, Arbitrator 1 and Arbitrator 2 have different databases. Each Arbitrator keeps information only on the Publishers who belong to the respective Arbitrator.

The advantage of this model is that the load on the Arbitrator is well distributed. This makes it suitable for large-

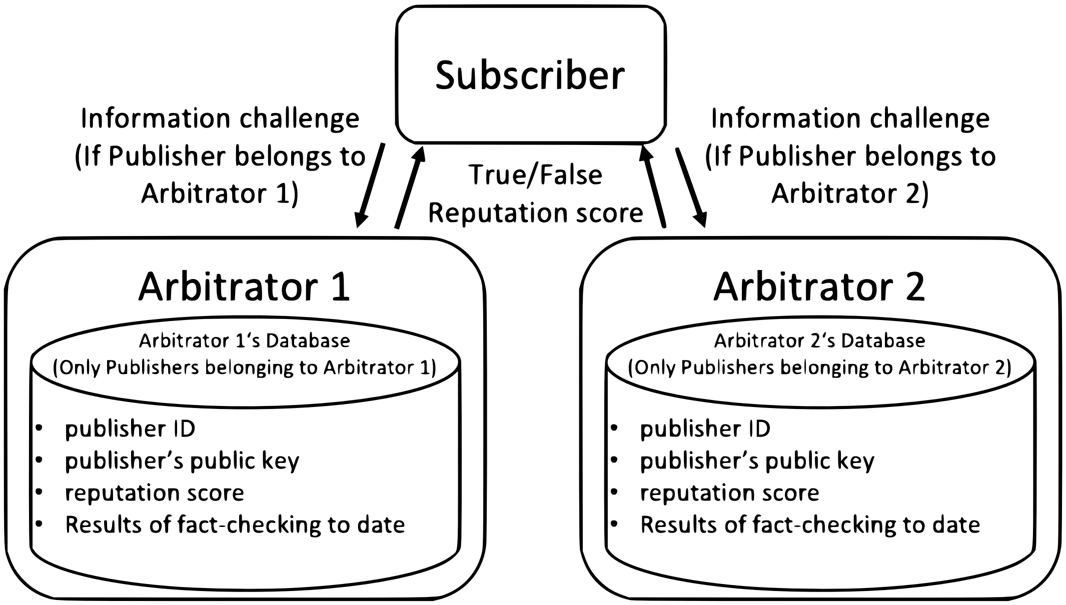
Subscriber
Information challenge (If Publisher belongs to Arbitrator 1)
True/False Reputation score
Information challenge (If Publisher belongs to Arbitrator 2)
Arbitrator 1
Arbitrator 1's Database (Only Publishers belonging to Arbitrator 1)
• publisher ID
• publisher's public key
• reputation score
• Results of fact-checking to date
Arbitrator 2
Arbitrator 2's Database (Only Publishers belonging to Arbitrator 2)
• publisher ID
• publisher's public key
• reputation score
• Results of fact-checking to date

Figure 15. Combination of specific Arbitrator and Publishers

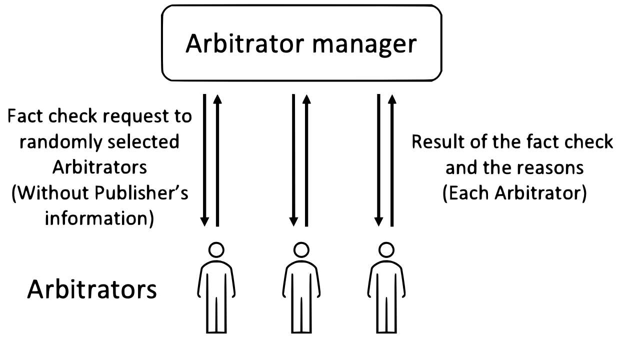
Arbitrator manager
Fact check request to randomly selected Arbitrators (Without Publisher's information)
Result of the fact check and the reasons (Each Arbitrator)
Arbitrators

Figure 16. Arbitrator manager model

scale systems. The accuracy of the information challenge can be improved because the Publisher can select the appropriate Arbitrator according to the field of expertise and language used.

The disadvantage of this model is that although there are multiple arbitrators, only one arbitrator performs fact-checking, which may bias the judgment of credibility. It also has no redundancy. Therefore, it is not suitable for cases where the accuracy of fact-checking or stable availability at any time is important.

C. Arbitrator manager model

This model sets an ”Arbitrator manager” that accepts access from publishers and subscribers. Arbitrator manager takes requests such as registration from Publishers, information challenge from Subscribers, confirmation of reputation score, etc. Only the fact check required for the information challenge is requested and distributed to multiple Arbitrators. The public key of the Publisher, reputation score, ID, and other information are kept by the Arbitrator manager. The configuration is shown in Figure 16.

Information challenge in this system is performed as follows.

1) The arbitrator manager receives information challenge from the Subscriber.
2) The Arbitrator manager verifies the signature and verifies that it is the correct Publisher.
3) The arbitrator manager requests a fact check from randomly selected arbitrators (the number of arbitrators is arbitrary).
4) Each arbitrator performs fact-checking and returns the results and reasons to the arbitrator manager.
5) The Arbitrator manager compiles the results of all fact-checking, returns the results to the Subscriber, and updates the Reputation score.

In step 3, the number of arbitrators to request fact checks can be considered according to the situation. Using numerous arbitrators may improve credibility, but it also increases the time and cost. In addition, the method of selecting Arbitrators could be not only random but also selecting appropriate Arbitrators according to their expertise in the language or field of study.

In step 5, there are several possible ways to compile the results of all fact checks. One is to simply ask how many people perform fact checks and reflect the number of people who judged the results to be true in the reputation score, another is to adopt the result of a majority vote, and another is to use a majority vote, but if the number of true/false votes are close, the final decision is made by the Arbitrator manager.

The advantages of this model are that the Subscriber does not need to select an Arbitrator, but only needs to access the Arbitrator manager, that there is no problem of sharing and managing the Publisher's key and reputation score among multiple Arbitrators, and that the Arbitrator manager can make the final decision when there are multiple Arbitrators. In addition, the Publisher's key and reputation score can be shared and managed among multiple Arbitrators, and Arbitrators can be easily added or deleted. Therefore, this model is suitable for cases where high accuracy is important by having multiple Arbitrators perform fact-checking.

The disadvantage of this model is that it does not distribute the load of the Arbitrator manager itself and does not have redundancy. Therefore, it is not suitable for large-scale systems.

VIII. Conclusion and Future work

In this study, we proposed a new framework (Secure Publication Subscription Framework) that allows subscribers to check the accuracy of information based on the authenticity of the publisher's historical data by checking the reputation score. In this framework, subscribers can check the reputation score of the publisher and challenge data reliability if the information is suspected to be unreliable. We also conducted experiments on the publisher's reputation score, and found that the actual reputation score approximates the expected value calculated from the probability of correctly judging the reliability of information. In the actual operation of this framework, it will be necessary to incorporate multiple Arbitrators from the aspect of load distribution, etc. We have shown three applicable methods to support multiple Arbitrators, and discussed their technical feasibility. Each of them has different merits and can be applied to various situations.

The development of the Internet and social media has made it very convenient for anyone to easily disseminate information, but it has also caused a major problem: fake news. However, there is so much information that we see every day that it is practically difficult to check all of it to make sure it is not fake news. Moreover, some of the information is highly specialized and cannot be confirmed as true or false even if it is carefully read. Therefore, we believe that there is a demand for a framework that allows anyone to easily verify whether a Publisher is impersonating someone else, and to confirm the authenticity of that Publisher.

As future research, integration of AI(Artificial Intelligence) algorithms to automatically identify fake news with expert arbitrators is a promising path. Although the accuracy of discriminating fake news has been a challenge for AI technologies, our expert framework can aid by using AI algorithms to improve false positives/negatives. Combined with these technologies, we believe that a robust data reliability framework for publication/subscription platforms can emerge.

There are still some minor problems. For example, in the current reputation score algorithm, the score of publishers who publish a small number of articles is rated higher than the actual credibility of the articles. This problem can be improved by setting the score lower when the number of articles is below a certain level. We believe that improving the specification of these details will make this framework more realistic.

References

[1] S. Yoshimura, K. Inoue, D. Cavendish, and H. Koide, "Secure Publication Subscription Framework for Reliable Information Dissemination." *The 2022 IARIA Annual Congress on Frontiers in Science, Technology, Services, and Applications*, 2022.

[2] D. M. J. Lazer, M. A. Baum, Y. Benkler, A. J. Berinsky, K. M. Greenhill, F. Menczer, M. J. Metzger, B. Nyhan, G. Pennycook, D. Rothschild, M. Schudson, S. A. Sloman, C. R. Sunstein, E. A. Thorson, D. J. Watts, and J. L. Zittrain, "The science of fake news." *Science*, pp. 1094–1096, 2018.

[3] N. Grinberg, K. Joseph, L. Friedland, B. Swire-Thompson, and D. Lazer, "Fake news on Twitter during the 2016 US presidential election." *Science*, pp. 374–378, 2019.

[4] A. Bovet and H. A. Makse, "Influence of fake news in Twitter during the 2016 US presidential election." *Nature communications*, pp. 1–14, 2019.

[5] S. Nakamura, T. Enokido, and M. Takizawa, "Subscription Initialization (SI) Protocol to Prevent Illegal Information Flow in Peer-to-Peer Publish/Subscribe Systems," *19th International Conference on Network-Based Information Systems*, pp. 42–49, Sept. 2016.

[6] F. M. Salem, "A Secure Privacy-Preserving Mutual Authentication Scheme for Publish-Subscribe Fog Computing," *14th International Computer Engineering Conference*, pp. 213–218, Dec. 2018.

[7] M. Srivatsa and L. Liu, "Secure Event Dissemination in Publish-Subscribe Networks," *27th International Conference on Distributed Computing Systems (ICDCS '07)*, pp. 22–22, June 2007.