International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2019): 7.583

Secure RFQ Negotiations: Enhancing Privacy and Efficiency in OTC Markets

Ananth Majumdar

thisisananth[at]gmail.com

Abstract: ***Privacy is crucial for market participants as it protects competitive strategies, maintains market stability, and ensures security against fraud, while complying with legal regulations. The current over-the-counter (OTC) trading mechanisms, primarily reliant on voice and chat communications, are cumbersome and prone to errors. Existing automated RFQ (Request for Quote) negotiation platforms raise concerns about data privacy, as platform providers can potentially access and misuse order information. This paper presents a privacy-preserving RFQ negotiation platform designed to facilitate secure and efficient transactions between multiple buyside and sellside participants. The platform ensures that the central system only facilitates transactions without accessing RFQ details, thus preserving privacy. Additionally, it offers scalability and configurability, allowing participants to selectively include counterparties and automate the negotiation process, thereby enhancing operational efficiency and compliance with regulatory requirements through comprehensive audit logs.***

Keywords: privacy, market participants, OTC trading, automated RFQ platforms, data security

1. Introduction

Over-the-counter (OTC) trading has long been characterized by direct, one-on-one negotiations between buyside traders and sellside salespersons, typically conducted through voice or chat mechanisms. While this traditional approach allows for personalized interactions, it suffers from significant inefficiencies and error-prone processes. Traders must manually request and compare prices from various dealers, leading to potential trading errors due to copying, pasting, or miscommunications during voice calls. This cumbersome process not only hampers operational efficiency but also increases the risk of costly mistakes.

Despite the advent of automated RFQ negotiation platforms aimed at streamlining these processes, concerns regarding data privacy remain a major barrier to their widespread adoption. Market participants fear that platform providers can access sensitive order details, potentially leveraging this information for undue advantage. Such privacy concerns undermine trust and deter the use of otherwise beneficial automated systems.

In response to these challenges, this paper introduces a novel RFQ negotiation platform that prioritizes privacy and security. Unlike existing solutions, our platform ensures that the central system only knows the participants involved but remains blind to the specifics of the RFQs, such as asset details or order sizes. This privacy-preserving design is complemented by features that enhance scalability and configurability. Buyside participants can selectively include specific sellsides in RFQs, and new participants can seamlessly join the platform with minimal configuration.

Moreover, our system automates the entire RFQ negotiation process, providing participants with a unified blotter that tracks all ongoing negotiations and finalized trades. This automation not only eliminates manual errors but also ensures that trades are accurately booked with the winning dealer. To comply with regulatory requirements, the platform maintains full audit logs accessible to all RFQ participants, ensuring transparency and accountability.

Through this innovative approach, we aim to provide a secure, efficient, and trustworthy solution for RFQ negotiations, addressing the key concerns of market participants and paving the way for more robust and reliable OTC trading processes.

2. Background

2.1 OTC Market

OTC or over the counter market is one that is off exchanges. A variety of financial products trade on OTC market's main ones among them include fixed income products like bonds, interest rate swaps and equity derivatives like options. OTC trading can happen for stocks traded on exchanges and also those which are not traded on exchanges.

Financial instruments can be traded OTC for various reasons. If the investor is doing a large trade and if it is publicized pre-trade, other traders and investors can take advantage of it by moving the market away from the original investor. Buysides and dealers have favorable trade relations and they might make the trade OTC so that they can get better deals that way.

Some instruments are not available on exchanges and the only way to trade them is over the counter. Similarly, bonds don't have a liquid market and hence might not be available to trade on an exchange and investors have to rely on otc markets for these trades also.

OTC trades happen mainly through voice (telephone/turrets) and through chat (Bloomberg IB). Some trading platforms or facilities are also being created for trading of OTC instruments.

2.2 Request for Quote

RFQ (Request for Quote) functionality, which is widely

used in ETF, Foreign Exchange and Fixed Income marketplaces, offers a trader the opportunity to ask for a quote from potential counterparties, based on the security, "side", and quantity. An RFQ can be "advertised" to potential counterparties within a trading facility at the discretion of the trader either on a one-to-one, one-to-many, or "all-to-all" basis, depending upon the situation and the trader's preferences. This means traders can manage exposure of their intentions in a way that is appropriate for the order. It also provides clients with an opportunity to find a match at a price other than a simple midpoint of the displayed quote and can serve as an efficient means of price discovery for very large orders, where no quote is available. [1]

2.3 Voice trading

Voice trading for over-the-counter (OTC) securities refers to the process of buying and selling financial instruments through direct communication, typically via phone calls or other verbal means, rather than through electronic trading platforms. This method is often used in the OTC market. Voice trading is particularly useful for complex and bespoke transactions that may not fit the standard mold of electronic trading platforms. This includes large block trades, illiquid securities, or customized financial instruments. Voice trading allows for more flexibility in terms of negotiation and structuring of trades. Traders can discuss and adjust the terms in real-time to reach a mutually agreeable deal.This has higher potential for errors and miscommunication given the unstructured nature of the communication. Voice trading remains an integral part of the OTC market, particularly for transactions that require a personal touch and customized terms.

2.4 Trading over chat

Similar to voice trading, trades for over-the-counter instruments are also conducted via instant messaging platforms like Bloomberg IB, Refinitiv messenger and Symphony. Trades conducted through chat platforms like Bloomberg IB often involve transactions that benefit from direct negotiation, customization, and discretion. This includes large block trades, illiquid securities, or customized financial instruments. Chat allows for real-time, detailed negotiations between traders, which is crucial for non-standardized trades. This flexibility is often necessary for agreeing on terms that electronic platforms cannot easily accommodate. It also helps to get liquidity due to the personal relationship nature of the communications in situations with low liquidity. Text-based communication can lead to misunderstandings or misinterpretations of trade terms and instructions. Miscommunication can result in incorrect trades, disputes between parties, and potential financial losses. Chat-based trading often requires manual entry and confirmation of trade details increasing the risk of human error and operational risks.

2.5 Trading venues

Trading venues for Request for Quotes (RFQs) are specialized platforms that facilitate the buying and selling of securities by allowing market participants to request and provide quotes for specific financial instruments. These platforms connect a broad network of market participants, enhancing liquidity and price discovery while offering tools and features to streamline the trading process and ensure compliance. The popular ones include Bloomberg terminal, TradeWeb and Instinet.

These platforms are designed with robust security features, including encryption and user authentication, to ensure secure communication. They also comply with regulatory requirements for data retention and surveillance.

But given the encryption keys are held by the platforms, the data is decrypted at various times and there is still danger of private data leaks or even worse the platform using the data for other purposes [2]. This paper details a system for trading RFQs over a secure communication platform where the data is encrypted with the firm's own encryption keys and the trading venue doesn't have visibility into the RFQ details at all.

3. Secure Messaging Platform

There are a variety of messaging platforms all of which vary in their terms of security and privacy guarantees. Most of the platforms in use today use secure HTTP to transmit messages and save the messages encrypted in a database providing both transport security and security at rest. Coming to privacy, the platforms differ in terms of the flexibility they provide in the encryption keys used for encrypting the messages. There are three flavors of these privacy guarantees.

3.1 Types of privacy architectures

First, a majority of the cloud platforms generate the keys by themselves and encrypt the messages, which makes them secure but leaves open the chance that the administrators of the platform could in theory get access to the messages. Given the sensitive nature of messages exchanged on these platforms, firms are wary of giving the keys to their data to the cloud vendors. This also guarantees the least amount of privacy given the cloud vendor has access to the keys and in theory can read the messages. The second type of messaging platforms allow their clients to bring their own keys (BYOK) and put them in the cloud. Since the keys are generated by the firms who use the messaging platform, these are more secure in theory since the cloud platform promises to not access the keys. But in practice some BYOK companies also allow their customers to upload the keys in the cloud infrastructure for easier maintenance. In this case, it is not much different from the first type as the firms are still losing control of the keys.

The third type of messaging platforms allow their client to truly bring their own keys. These companies allow their clients to generate strong keys in tamper resistant hardware security modules and never allow the keys to leave their premises. This makes sure the keys are secure and provides the highest level of privacy as the cloud vendor or for that matter anyone else doesn't have access to the messages.

3.2 Messaging workflow

In the system with true customer owned keys, the messaging providing application (central server) doesn't have access to the raw messages. It only works with the encrypted messages and some metadata specifying the tokens about the routing of messages.

In this system, each firm that wishes to use the secure messaging infrastructure will have a component on their premises which will interact with the hardware security manager (HSM) to generate the keys and work with the messaging client to encrypt the messages for sending to the central server.

Key wrapping is a method used to securely encrypt keys, allowing them to be safely stored or transmitted. It involves using a Key Encryption Key (KEK) to encrypt a Data Encryption Key (DEK) or other keys. This process ensures that the DEK is protected, even if the storage medium is compromised. By using this key wrapping method, a firm's internal key manager can exchange keys with the central server, using this to authenticate the users of the firm. To increase the privacy of conversations, each conversation is encrypted with a separate conversation key. By using sophisticated key wrapping cryptographic methods outlined in [2], it is possible to exchange the wrapped keys around the server and the users' client validated by the firm's key manager ensuring the authenticity and privacy of messages. The keys are encrypted with a multilayer wrapping where the conversation key is wrapped with the user's account key which is then wrapped with the firm key so that privacy is protected at each layer.

It is also possible for users of different firms to also exchange messages securely. Since the messages in each conversation are encrypted with a separate encryption key, the central server can help exchange these keys using public key infrastructure. Firm 1 can sign the conversation key with firm 2's public key and send it to the server. The server can then send this signed key to the firm 2. Firm 2 can extract the conversation key with their private key gaining access to the key to decrypt the messages.

To further increase the privacy, the firm's key manager can rotate the keys per a fixed time like daily or weekly to reduce the surface area of exposed messages even if one conversation key leaks.

3.3 Exchange objects with custom schema securely

This idea was further extended to allow the exchange of objects with custom schema securely and privately. We can use the same messaging rails to also exchange custom object schema and store them in an object store. In this users define a schema with the fields in the object they want to share using the OpenAPI3.0 standard. Then they can use the firm's key manager to generate a key for encrypting this schema. The encrypted schema is then sent to the central server. This schema can then be shared with another firm using the process described above using public key infrastructure. Objects using this schema can be stored in a central object store. Once the encryption keys are stored and shared securely using key wrapping, these objects can be shared with participants in different firms.

3.4 Bot Accounts

In addition to user accounts, the system has automated accounts or bot accounts which when added to a conversation can use an API to query the messages in the conversation and then use those to create some automated actions. Privacy is not compromised even in this case the bot account needs to added each conversation separately.

This sets up the primitives that can be used to design an application to negotiate RFQs using this infrastructure that will be private and secure. This paper further describes the design of an application using the secure messaging platform and a secure object store with custom schemas to negotiate RFQs between different buyside and sellside firms in a private and secure manner.

4. Approach

As described above, we can use the secure messaging framework and the secure object store to negotiate RFQs securely. Below is the high-level architecture for it

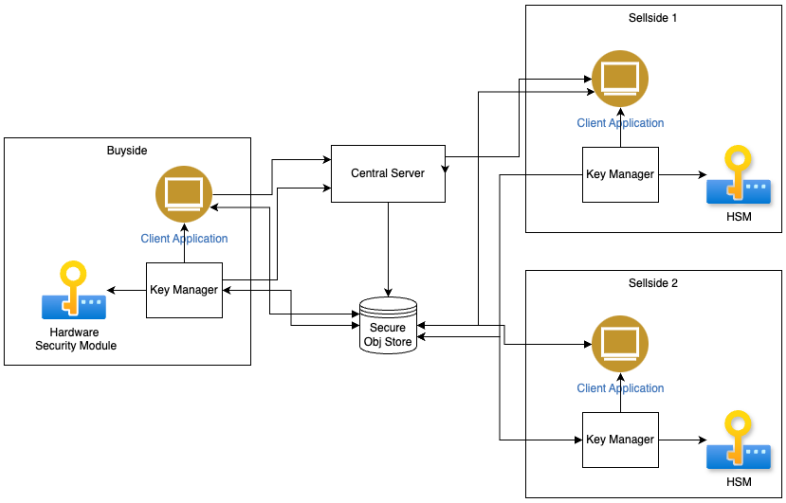

Figure 1: Architecture of secure messaging and secure object store

4.1 Schema design

For supporting RFQ negotiation using the secure object store, we need to create a model for the RFQ we want to send. It includes the instrument size and tenor/maturity and other options based on the asset for which the quote is requested. Considering an interest rate swap, it can contain the following fields. So, the schema for the Request object contains the following fields.

This data is sensitive and private, so this data will be encrypted.

In addition to the data that we need to send, we also need to identify who is the buy side who is interested in the quote and the dealers who are going to send quotes. In addition, we also need to know the action that this message is doing. The following actions are possible for the request - the buy side sending the initial quote, the dealers acknowledging the request, the dealer sending quotes, the buy side accepting a quote or the buy side countering the quote, the dealer responding with a new quote, the buy side agreeing to a quote and the dealer confirming the price ending the RFQ.

One more thing to note is that this information is needed by the central server to route the messages among the participants and keep the state of the RFQ. So, these need to be unencrypted.

So, our objects will have an encrypted data section and an unencrypted metadata section.

We need the following objects for our workflow

RFQRequest
Quote
QuoteAccept
QuoteConfirm
For each of these objects, we have a data section and a metadata section.
Metadata will be same for all the objects
Metadata
{
"sender": "b1",
"receivers":["d1","d2","d3"],
"action":"request" //could also be quote, quoteAccept, quoteConfirm
}
Below is the data schema for the different objects
RFQRequest
{
"sender": "b1",
"receivers":["d1","d2","d3"],
"notionalValue": 10000000,
"currency":"USD",
"refIndex":"LIBOR-3M",
"tenure":"10yr",
"id":"r1-b1"
}
Quote
{
"bid": 1.3,
"ask": 1.5,
"id": "d1-q1",
"rfqId":"r1-b1"
}
QuoteAccept
{
"quoteId":"d1-q1",
"rfqId":"r1-b1",
"side":"buy"
}

QuoteConfirm:

{
"confirmId":"r1-b1-d1-q1-c1",
"rfqId":"r1-b1",
"quoteId":"d1-q1"
}

4.2 State Management

As the RFQs are sent from the buy side to sellside it is important for the server to capture and maintain the state centrally. For state management, we used the Camunda state machine. We modelled the state using Business Process Model Notation and use the embedded Camunda within Spring with state stored in Postgres database. Having a central state machine allows the server to maintain a consistent state.

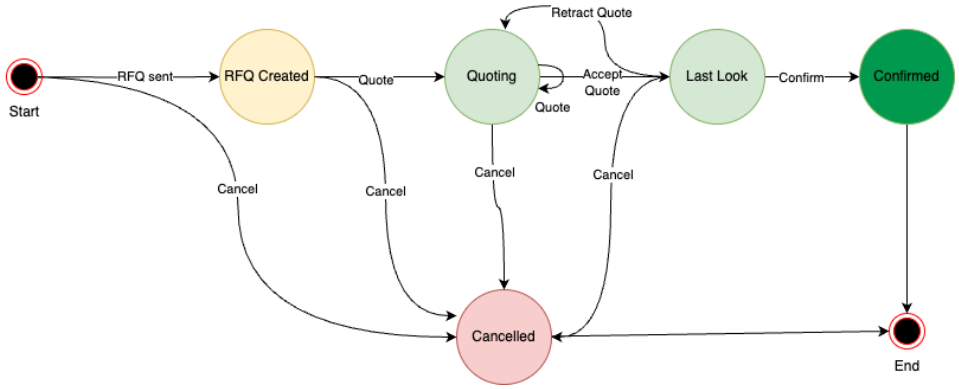

Figure 2: State Diagram for RFQ negotiation

Setup

4.2.1 Creating keys

Each buyside and sellside needs to create a public and private key and register their public keys with the central server. The central server creates the schema and shares it with all the sides in the conversation.

4.2.2 Buyside and Sellside bot account

To make it easy to manage multiple RFQs and view and act upon the quotes from multiple dealers, it is important to automate and present a unified view of the running RFQs. For this we can create an automated account (bot account) for each of the participant firms in the RFQ and add them to all the bilateral RFQ conversation rooms so that the bots can aggregate the running RFQs and present a unified view to the end user.

4.2.3 Unified Views

To make it easy to manage multiple RFQs and view and act upon the quotes from multiple dealers, it is important to automate and present a unified view of the running RFQs. Instead of presenting the RFQs as simple text messages, we can create a custom iFrame to show the aggregated details of the running RFQs. One way to do this is to register a viewHandler with the messaging client such that if an object of that type is found, the message client can delegate the view to our custom application along with the object. Our view application can then consume this object and build a custom view for RFQ handling.

4.3 Custom blotters

The UIs can be further improved by creating custom blotter applications for both the buysides and sellsides.

4.3.1 Buyside Blotter

On the buyside, the blotter would show a list of current running RFQs and upon clicking it can show the RFQ details, the price levels from different dealers, the best bid/offer and a way for the buyside to accept a quote and complete the RFQ.

4.3.2 Sellside Blotter

On the sellside, the blotter would show a list of current running RFQs from various buy-sides and upon clicking it can show the RFQ details, the current quote that the sell side has provided and ability to modify the quote. It can also see if its quote was accepted and then confirm the RFQ.

4.4 Auditing

RFQs are subject to various regulations from authorities in various jurisdictions like MIFID II, FINRA, SEC etc. All of these require reporting of these RFQs and also data retention for any investigations. For that purpose, it is important that all RFQs are audited. We can take advantage of the private conversations to store the audit trail of all the actions pertaining to the RFQ. In each bilateral room related to each buyside and sellside all the actions pertaining to an RFQ from the buyside and sellside could be written as text messages. In order to not clutter the rooms and make it difficult for end users to see all the message, we could create separate audit rooms where all the messages can be written for record keeping purposes.

After the RFQ is complete, the automated bots can also send a final confirmation message with the final agreed terms of the RFQ.

5. Evaluation and Results

This solution was evaluated with a test run using 2 buysides and 5 sellsides. RFQs were sent across interest rate swaps and equity derivatives over a two-month period. During that time 50 RFQs were sent and with around 40 RFQs being completed and the rest being abandoned. All of the 2 buysides and 5 sellsides reported being satisfied with the flexibility and privacy of the solution. Further improvements like a separate sellside and buyside blotter. An independent security firm evaluated the solution and approved the security and privacy.

6. Conclusion

This paper gives an overview of the OTC markets and the Request for Quote process that the buyside firms use to gauge the market and to access liquidity in the markets. It then goes over the various trading options available in OTC markets and their privacy implications. The paper then reviewed how we can use sophisticated cryptography to create a secure and private messaging platform and how we can use the same technique to share objects with custom schemas securely. We then proceeded to describe the design and implementation of a secure RFQ negotiation platform using this secure messaging infrastructure. With this paper, we show that it is possible to have secure RFQ negotiations with sacrificing security and privacy.

References

[1] Instinet adds RFQ functions to Blockmatch https://www.instinet.com/sites/default/files/public/pdf_articles/20180313_RFQ.pdf
[2] Bloomberg reports accused of spying on Goldman Sachs Trading Terminals https://www.theguardian.com/media/2013/may/10/bloomberg-goldman-sachs-spying-terminals
[3] David MÄRaihi, David Gurle, Michael Harmon, Jon McLachlan, Ivan Rylach, Sergey Stelmakh, Secure End to End Communications, US 10432589 B1 2019