TEXAS STOCK EXCHANGE

# RAKE TCP

## Contents



### 0.1 Revision History

| Date | Version | Notes |
|---|---|---|
| June 25, 2025 | 0.1 | Initial version |
| September 09, 2025 | 0.6.draft | Alpha release. Refined message layout, identifiers and enumerations in line with early internal feedback. |

# 1 Overview

RAKE TCP is a payload agnostic message framing and network session protocol in the tradition of the OSI network layer model. RAKE is message based and guarantees delivery of messages from the Exchange to the Member. RAKE uses TCP/IP sockets for communication and includes a simple authentication mechanism.

Each sequenced message from the Exchange is guaranteed to be delivered in order and exactly once. Un-sequenced messages from the Member have no guarantees for delivery, order, or de-duplication. Every un-sequenced message from the Member will result in one or more corresponding sequenced messages from the Exchange. Members are responsible for tracking responses from the Exchange for their un-sequenced message deliveries. The higher-level protocols such as SEED OrderEntry perform uniqueness guarantees.

Most RAKE server higher-level applications are "Single Message In Flight", meaning the Rake server will queue subsequent messages from the Member until the first message has been processed.

## 1.1 StreamIDs, Session Recovery, and Fail-over

Members should track the sequence number of the messages they have received from the RAKE server. In the event of a network interruption they can logon to the RAKE server and request data to be replayed from a specific sequence number. Both primary and backup RAKE servers maintain the same order of messages..

RAKE sequence numbers start at 1. Logging on with a sequence of 0 skips replay and sends new messages.

Internal to the exchange, each matching engine is assigned its own StreamID, and set of symbols to match. RAKE informs the Member which StreamID sourced each message. This is informational only.

## 1.2 Heartbeats

Both the Exchange and the Member must send periodic messages to ensure the TCP/IP connection is operating correctly. If network traffic between Exchange and Member is frequent enough that is sufficient. If not, then both sides must send Heartbeat messages. The frequency of these should be no less than once a second. If no traffic or heartbeat message is received after 3 seconds, the network connection is assumed to be broken, and the network socket is closed.

No heartbeats should be sent until after a successful LogonResponse from the Exchange.

After a TCP connection, if the Exchange does not receive a `LogonRequest` within 3 seconds, the socket is closed.

# 2 Types

- This document uses the terms `Byte, Short, Int, Long` to refer to numeric values that are 8,16,32,64 bit long respectively. All are two's complement signed little-endian encoded.

| Type | Byte Length |
|---|---|
| `Byte` | 1 |
| `Short` | 2 |
| `Int` | 4 |
| `Long` | 8 |

- Strings are all `ASCII`, fixed-length. Strings shorter than the fixed-length should be left justified, by right padded with spaces (`ASCII` 0x20) to the full string length
- Enums are encoded as a single `Byte` unless they appear in a bitfield, then they use the number of bits specified.

## 2.1 LogonResponseCode

Success or failure code from a Logon Request. Additional LogonResponseCode values may be added without notice in the future and should not break Member applications.

| Enum Constant | value | Notes |
|---|---|---|
| SUCCESS | 0 | |
| INCORRECT_SENDER_COMP | 1 | |
| INCORRECT_SESSION | 2 | |
| INVALID_NEXT_SEQUENCE | 3 | The sequence number requested is larger than the largest known sequence, or is less than 0 |
| INVALID_CONFIGURATION | 4 | The RAKE server is configured incorrectly for the login request specified. Contact Trading Operations |
| INCORRECT_TOKEN | 5 | |

# 3 Messages from Member to Exchange

The following messages are expected to be sent from the Member to the exchange. The exchange will never send any of these messages to you.

## 3.1 LogonRequest

This must be the first message a Member sends to the Exchange after establishing a TCP/IP connection. If this message is sent at any other time, the RAKE server will close the TCP socket.

| Field Name | Offset | Size | Type | Notes |
|---|---|---|---|---|
| length | 0 | 2 | `Short` | size of the message exclusive of length field |
| messageType | 2 | 1 | `Byte` | '5'/0x35/53 |
| session | 3 | 8 | `Long` | Set to the current trading session if recovering from an outage, or 0 if connecting for the first time. |
| senderComp | 11 | 8 | `Str(8)` | Assigned by the Exchange |
| token | 19 | 8 | `Str(8)` | Assigned by the Exchange |
| nextSequenceNumber | 27 | 8 | `Long` | 1-based numbering. Use 0 to start at the end (skip currently stored messages). Use 1 to replay all the messages. Send the next sequence number you expect to receive from the Exchange (one greater than your last received sequence) to recover from a disconnection. |

## 3.2 MemberHeartbeat

Send during low traffic. At least once a second.

| Field Name | Offset | Size | Type | Notes |
|---|---|---|---|---|
| length | 0 | 2 | `Short` | size of the message exclusive of length field |
| messageType | 2 | 1 | `Byte` | '7'/0x37/55 |

## 3.3 UnsequencedMessage

Send message from Member to higher-level protocol at Exchange.

| Field Name | Offset | Size | Type | Notes |
|---|---|---|---|---|
| length | 0 | 2 | `Short` | size of the message exclusive of length field |
| messageType | 2 | 1 | `Byte` | '6'/0x36/54 |
| payload | type | 3 | variable | The remainder of bytes in the message. |

# 4 Messages from Exchange to Member

The following messages will be sent from the exchange to the Member. It is an error to send any of these messages to the exchange.

## 4.1 Debug

A human-readable debug message. As this message usually precedes the server closing the TCP connection, delivery of this message is not guaranteed. The payload of this message contains the ASCII message. It is not null-terminated.

| Field Name | Offset | Size | Type | Notes |
|---|---|---|---|---|
| length | 0 | 2 | Short | size of the message exclusive of length field |
| messageType | 2 | 1 | Byte | '0'/0x30/48 |
| payload | type | 3 | variable | The remainder of bytes in the message. |

## 4.2 EndOfSession

The Exchange trading session has ended. No more un-sequenced messages will be accepted from the Member. No sequenced messages will be sent by the RAKE server after this message.

| Field Name | Offset | Size | Type | Notes |
|---|---|---|---|---|
| length | 0 | 2 | Short | size of the message exclusive of length field |
| messageType | 2 | 1 | Byte | '4'/0x34/52 |

## 4.3 LogonResponse

Response to a logon request. See responseCode field to determine failure or success.

| Field Name | Offset | Size | Type | Notes |
|---|---|---|---|---|
| length | 0 | 2 | Short | size of the message exclusive of length field |
| messageType | 2 | 1 | Byte | '1'/0x31/49 |
| session | 3 | 8 | Long | The currently active trading session. This changes daily and will match the trading session fields on other protocols |
| nextSequenceNumber | 11 | 8 | Long | The sequence number of the next sequenced message the Member will receive |
| highestKnownSequenceNumber | 19 | 8 | Long | The highest sequence number the server knows at the time of logon. Members may use this as a proxy for when they have "caught up" when recovering a RAKE session. But there is no guarantee that the highest sequence has changed while a Member is rewinding data. Use with caution. |
| responseCode | 27 | 1 | Enum | Enum LogonResponseCode. Success or failure code from a Logon Request. Additional LogonResponseCode values may be added without notice in the future and should not break Member applications. |
| numberStreamIDs | 28 | 1 | Byte | The total number of StreamIDs that will be used in this trading session |
| instance | 29 | 4 | Int | Informational only. A unique ID for this RAKE server instance. Will be different between primary and backup. If it changes upon successive connections to the same RAKE server intra-day, the server has been restarted |

## 4.4 SequencedMessage

A sequenced message from the Exchange. Members should increment their message sequence count upon receipt of this message type.

| Field Name | Offset | Size | Type | Notes |
|---|---|---|---|---|
| length | 0 | 2 | `Short` | size of the message exclusive of length field |
| messageType | 2 | 1 | `Byte` | '2'/0x32/50 |
| streamId | 3 | 1 | `Byte` | The internal Exchange StreamID that created this message. see Session Recovery |
| payload | type | 4 | variable | The remainder of bytes in the message. |

## 4.5 ServerHeartbeat

Heartbeats will be sent once a second by the server if no other traffic has been sent

| Field Name | Offset | Size | Type | Notes |
|---|---|---|---|---|
| length | 0 | 2 | `Short` | size of the message exclusive of length field |
| messageType | 2 | 1 | `Byte` | '3'/0x33/51 |