Application Programming Interface - WFH

(Technical Specifications for interfacing with ODIN™)

Version 1.1

63 moons technologies limited

Table of Contents

1 Change History

Date	Message	Change Carried out	Doc Version
29/01/2018	Initial API	Initial API	1.0
16/02/2022	Logon request update	Introduced Broadcast access token for validation of connections initiated by B2C clients	1.1

2 Introduction

2.1 Preface

This document describes the interface message to be used to with the ODIN™ Trading engine using a web based or a client application.

2.2 Restriction on use and Disclosure of information and Data

No part of this document may be copied, reproduced, stored in a retrieval system, or transmitted, in any form or by any means whether, electronic, mechanical, or otherwise without the prior written permission of 63 moons technologies limited(formerly Financial Technologies India Limited.

Although 63MT has made every effort to provide accurate information at the date of publication, it does not give any representations or warranties as to the accuracy, reliability or completeness of the information in this document. Accordingly, 63MT, it subsidiaries and their employees, officers and contractors shall not, to the extent permitted by law, be liable for any direct or indirect loss arising in any way (including by way of negligence) from or in connection with anything provided in or omitted from this document or from any action taken, or inaction, in reliance on this document.

63MT reserves the right to amend details in this document at any time and without notice.

3 Session Management

The client application will be responsible for establishing session with the ODIN trading engine and manage the life cycle of the same. At the time of login, a token will be given by the trading engine that will form the basis of the session identification.

Each session will have a defined lifetime (in secs) and unless a heart beat (request) is sent during this time frame, the session will expire and all operations on the expired token will return an error.

Following are the stages in the session life cycle.

3.1 Session Establishment

Session is established via a *Logon* request. The log-on request will be authenticated by the trading engine and on success the caller will receive a *successful logon response* message along with a session TokenId.

3.2 Message Exchange

Message exchange can commence after session has been established with the trading engine i.e. after successful Logon. All are required to carry the valid TokenId received during the logon response.

3.3 Session Expiry

Session can be terminated by explicitly sending a LogOff Request message. The trading engine will then respond with a Logoff Response message and the session will expire. The session will also expire in case no messages are sent to the trading engine for a specified duration.

3.4 List of Exchanges and Market segment IDs

The Following is list of Exchanges and their respective Market segment IDs supported by the API:

Exchange	**Market Segment ID**
NSE Cash	1
NSE Derivatives	2
BSE Cash	3
BSE Derivatives	4
MCX Futures	5
MCX Spot	6

NCDEX Futures	7
NCDEX Spot	8
NSEL SPOTCOM	9
NSEL COM	10
MSE Curr Futures	11
MSE Curr SPOT	12
NSECDS Futures	13
NSECDS SPOT	14
MSE Cash	15
MSE Derivatives	16
NSE-OFS	33
BSE-OFS	25

4 Summary of Feed Handler Messages

The method of communicating with Feed Handler is via industry standard WebSocket. Any platform that supports a connection to WebSocket can access Feed Handler.

Feed Handler and FH have been used interchangeably across the document and should otherwise mean the same.

4.1 Establishing a WebSocket Connection to the gateway

To establish the WebSocket connection to the gateway complete the following steps:

- Ensure the IP address of the Broadcast Server is available
- Ensure the IP address of a Broadcast Server is visible to the Third Party Application i.e. first establish that the connectivity is in place by pinging the gateway.
- Connect to the socket using the port number specified.

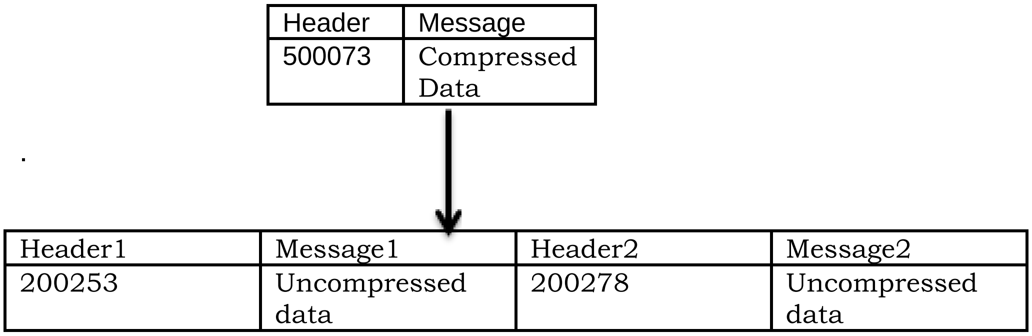
The gateway IP address and port number will be specified on installation time.
All Broadcast messages & unsolicited messages will be sent on this WebSocket connection. All subscription requests for broadcast need to be made on WebSocket connection.
Network fragmentation need to be handled at Front End Level

4.2 Compression

Each message between a third party client and Broadcast servers has to be compressed using Zlib Library. Zlib is industry standard Compression library available for JAVA and MS .net Platform. Header Packet of Five character needs to prefix with all Compress Message.

How to decode the Message:

- The first bytes of the broadcast packet indicate the whether message is compressed or not. If first byte contain 5(♣) then message is compressed message and if it contain 2(☻) then it is uncompressed message.
- Next 5 byte indicate the length of message.
- In case of multiple message, Each message contain header and non-compressed message.



Header	Message
500073	Compressed Data

.

Header1	Message1	Header2	Message2
200253	Uncompressed data	200278	Uncompressed data

4.3 Interface Protocol

Each connection between a Third Party API user i.e. client and the broadcast Server is defined as a session. An API Interface session is made up of three parts:

- session establishment
- broadcast Suscription
- session termination

4.3.1 Session Establishment

Session establishment is via a 101 *Logon* request from a Third Party API user. Successful establishment of the session will be indicated when the API user receives a 102 *successful logon response* message on WebSocket.

All Subscribe Feed data will be sent from FH to the user's systems only once a valid session has been established.

.

4.3.2 Session termination

Session termination will occur when an API user disconnect WebSocket.

4.4 Summary of messages

The Following is a summary of messages supported by the native API.

Message Type	Message Code
Login Request Message.	101
Login Response message.	102

4.4.1 Broadcast Message

Message Type	Message Code
Best Five Subscription Request	127
Best Five Information Response	128
Multiple Touchline Subscription Request	206
Multiple Touchline Subscription Response	209
Index Info Request	1002
Index Info Response	222
Index OHLC Request(indices details request)	224
Index OHLC Response(indices detail response)	225
Top Gainer and Loser Request	157
Top Gainer and Loser Response	158
Most Active Securities Request	159

Most Active Securities Response	160

4.5 Message Structure

All Messages are composed of the standard header followed by the body.
Native API has been modeled on the lines of FIX i.e. the convention followed is tag=value pair for any information. The messaging format is ASCII. All the fields are pipe ("|") separated.

4.5.1 Message Header

Standard message Header

Tag	**FieldName**	**Req'd**	**Data type (Max)**	**Comment**
NA	MessageStartValue	Y	Byte	This field will be start of the message. This field doesn't contain any tag no. Default value = 5 Length = 1 This field is not compressed and has to be prefix at start of compressed FT message.
63	Version	Y	String	Identify the version number Currently, this is to be hard-coded as FIX3.0
64	Message Code	Y	short	This field contains message type specified in Summary of message code For e.g. for Logon Request = 101 Logon Response = 102
65	Message Length	Y	short	Length of the whole message excluding MessageStartValue character
66	Message Time	O	String	Date and Time of the message in following format, time is in

				24 hour format e.g. hhmmss (24 hour format)

Header and Body will consist of delimited String which looks like the following:

♣63=FIX3.0|64=256|65=73|66=2003-01-01155434|1=11|70=10004|4=1000|97=2|19=

Message Header **Body**

Remarks:

All Messages will have 1 Byte at the start, which contains byte (1 Byte) for MessageStartValue.

All messages will contain Message Header at the start.

Since data will be formed using Tag and Value Pair, Message Header Length is not fixed.
Message Header will include the following (in sequence):

Version – 63=FIX3.0
Message Code – 64=255
Message Length – 65=99
Message Time – 66 in 24 Hour Format like: 233022

Maximum of 3 Delimiters are used:

“=” Assignment Delimiter for Tag=Value Pair
“|” Field Delimiter
“$” Join Delimiter, this is used when specifying array of values

4.5.2 Message Trailer

There is no message trailer

4.5.3 Broadcast Messages

4.5.3.1 Logon Request (Client to WFH)

This message is sent by the client for establishing a session with WFH for broadcast subscription. The WFH will authenticate the user by validating
The logon message and respond with a Logon Response message.

Tag	FieldName	Req’d	Data type (Max)	Comment
	MessageHeader	Y		Message header to be filled with message code = 101

400	User type	Y	Integer	17 = Broadcast for B2C clients
67	User Id	Y	String (10)	Max Len 10
401	Auth Type	Y	Integer	1 = Password 2 = Broadcast access token
68	Password/ Broadcast access token	Y	String	Password/ Broadcast access token Password condition: Password should be between 6 and 8 characters Password should be alphanumeric Special characters not allowed. e.g.] [~ @ # $ % ^ & * ( ) - \ + { } \ / = ? `

♣63=FT1.0|64=101|65=75|66=2004-05-04 233022|400=16|67=user1|401=1|68=123456|51=16

Remarks:

The format of the row data field in logon request message (sent by Client) is as follows:

- Message Header with logon request message code and length of the row data field
- User Id, max length is 10 char
- Password and New Password should be between 6 and 8 characters

4.5.3.2 Logon Response (WFH to Client):

For every logon request, WFH will send the Logon response message to Client.

Tag	**FieldName**	**Req'd**	**Data type (Max)**	**Comment**
	MessageHeader	Y		Message header to be filled with message code = 102
70	Logon Status	Y	Short	Value to identify the logon status. 10000 – Login Successful
97	Days to Expire	O	Short	No. of days for expire. This will be populated within 15 days of expiry of the account. In this case, the logon status will be 10004.
1	Market Segment Id	Y	Short	Login allowed for the user.

Broadcast mechanism is developed over a publish/subscribe model. In order to receive market information, user of the API will need to subscribe to the security as well as the information.

4.5.3.3 Multiple Touchline Request (Client to WFH)

The Market Watch is send by User to subscribe/unsubscribe to the touchline broadcast (best buy qty, buy value, sell qty, sell value.etc) message for any specified security.
Request can be send as Multiple Pair of Segment Id and Token. The Response for Multiple touchline Request is Market Snapshot Information.

Tag	FieldName	Req'd	Data type (Max)	Comment
	MessageHeader	Y		Message header to be filled with message code = 206
1	Segment Id	Y	Short	
7	Security Code	Y	Long	
230	Operation Type	Y	Short	1 – Subscribe 2 – Unsubscribe

Eg.
63=FT3.0|64=206|65=80|1=1$7=22|1=1$7=1594|1=3$7=505537|1=3$7=500105|1=1$7=553|1=3$7=526881|1=3$7=500180|1=1$7=11630|4=1000|230=1

4.5.3.4 Multiple Touchline Response (WFH to Client)

The Market Watch Response is send by WFH whenever the update is received from exchange for the specified security. This message will give best buy qty, buy price, sell qty, sell value etc.

Tag	FieldName	Req'd	Data type (Max)	Comment
	MessageHeader	Y		Message header to be filled with message code = 209
1	Segment Id	Y	Short	
7	Security Code	Y	Long	
2	Buy Qty	O	Long	
5	Sell Qty	O	Long	
3	Buy Price	O	Long	
6	Sell Price	O	Long	
8	Last Trade Price	O	Long	
9	Last Trade Quantity	O	Long	
73	Last Trade Time	O	String(21)	
74	Last Update Time	Y	String(21)	
88	Open Interest	N	Long	
54	NetChangeFromPrevClose	O	String	In Rs
76	Close Price	O	Long	
75	Open Price	O	Long	
77	High Price	O	Long	
78	Low Price	O	Long	
79	Volume	O	Long	
80	Average Traded Price	O	Long	Average Traded Price currently zero
81	Total Buy Qty	O	Long	currently zero
82	Total Sell Qty	O	Long	currently zero

93	Life Time High	O	Long	currently zero
94	Life Time Low	O	Long	currently zero
38	Market Type	O	String	Applicable only for NCDEX spot

E.g

```
63=FT3.0|64=209|65=272|66=04092012 071138|1=1|73=2012-09-04
152908|7=22|79=121758|8=130035|54=-0.63|9=395|80=130117|74=2012-09-04
152908|2=5|3=130035|5=59|6=130100|81=99436|82=82671|76=130860|75=130770|77=1
30770|78=129615|88=|93=0|94=0|393=-8.25|80=130117|380=0 - 0|399=100
```

4.5.3.5 Best Five Subscription Request (Client to WFH)

The Best Five Request is send by User to subscribe/Unsubscribe the full broadcast (best five buy qty, buy value, sell qty, sell value) message for specified security, which includes the market depth as provided by the exchange.

Tag	**FieldName**	**Req'd**	**Data type (Max)**	**Comment**
	MessageHeader	Y		Message header to be filled with message code = 127
1	Segment Id	Y	Short	
7	Security Code	Y	Long	
230	Operation Type	Y	Short	1 – Subscribe 2 – Unsubscribe

4.5.3.6 Best Five Information (WFH to Client)

The Best Five Response is sent by WFH whenever the update is received from exchange for specified security. This message will give top five buy qty, buy price, sell qty, sell value etc.

Tag	**FieldName**	**Req'd**	**Data type (Max)**	**Comment**
	MessageHeader	Y		Message header to be filled with message code = 128
1	Segment Id	Y	Short	
7	Security Code	Y	Long	
74	Last Update Time	Y	Int	
73	Last Trade Time	O	String(21)	
8	Last Trade Price	O	Long	
9	Last Trade Quantity	O	Long	
81	Total Buy Qty	O	Long	
82	Total Sell Qty	O	Long	
76	Close Price	O	Long	
75	Open Price	O	Long	
77	High Price	O	Long	
78	Low Price	O	Long	
79	Volume	O	Long	

80	Average Traded Price	O	Long	Average Traded Price
54	NetChangeFromPrevClose	O	String	In Rs
This substructure will repeat 5 times				
11	B/S Flag	Y	Short	1 for Buy and 2 for Sell
14	Price	Y	Long	B/S Price
12	Quantity	Y	Long	B/S Quantity
37	No. of Orders	Y	Long	This signifies No. Of Buyers / Sellers
88	Open Interest	N	Long	
380	Daily Price Range	O	String	
93	Life Time High	O	String	
94	Life Time Low	O	String	

Explanation for above Message:
Last 3 fields will be coming 5 times for both Buy and Sell.

4.5.3.7 Trade Execution Range (WFH to Client)

The Trade Execution Range Response is send by WFH whenever the update is received from exchange for the specified security. These messages will be given in case subscription is available for such token either for touchline or Best Five.

Tag	FieldName	Req'd	Data type (Max)	Comment
	MessageHeader	Y		Message header to be filled with message code = 155
1	Segment Id	Y	Short	
74	Last Update Time	Y	String(21)	
7	Security Code	Y	Long	
499	Trade Execution Range	Y	String	Eg. 1500.25-1530.65

Example:
"63=FIX3.0|64=155|65=130|66=2015-01-07 094620|1=3|74=2015-01-07 094620|7=1278|499=73.9275-81.7100"
Kindly note that for Refresh Mode TAG 499 will Part of Touchline & Best Five Response.

4.5.3.8 Pause/Resume Request (Client to WFH)

This method can be use to pause or resume the broadcast subscription for user when portal / app is in minimine mode or broadcast is not needed temporarily. The Pause/Resume message is send by user to WFH. when App is minimized / Activated for any specified security. Request can be send as Multiple Pair of Segment Id and Token.

Tag	FieldName	Req'd	Data type (Max)	Comment
	MessageHeader	Y		Message header to be filled with message code = 106
230	Operation Type	Y	Short	1 – Pause 2 – Resume

Eg. 63=FT3.0|64=206|65=80|66=19:02:31|230=1 (for Pause)

63=FT3.0|64=206|65=80|66=19:02:31|230=2 (for Resume)

4.5.3.9 BroadcastSocketDisconnect (WFH to Client)

This method will be received by client when broadcast socket is disconnected / about to disconnect due to broadcast instance unavailability. On receiving this message, Client application shall call MDaaS URL received in logon response to get new broadcast IP & Port. This process is to be followed only in case of MDaaS flag is received as true during login response else client application can ignore this message.

Tag	FieldName	Req'd	Data type (Max)	Comment
	MessageHeader	Y		Message header to be filled with message code = 1111

Eg. 63=FIX3.0|64=1111