Protocol for Non-NEAT Front End (NNF)

Securities Lending and Borrowing Market Trading System

Version 2.7

Nov 2024

National Stock Exchange of India Ltd

Exchange Plaza, Plot No. C/1, G Block,

Bandra-Kurla Complex, Bandra (E)

Mumbai - 400 051.

Notice

SLBM Trading System Revision History		
Version	Page No	Description
2.6		1) Added chapter 8 Encryption Decryption of Interactive Messages 2) TLS/SSL communication implementation for sending and receiving the GR_REQUEST and GR_RESPONSE. 3) Symmetric Encryption/Decryption implementation for all the messages between member application and allocated Gateway server. 4) Annexure for Encryption/Decryption.
		Introduction of Error Code for User having No Trading Rights
		Removal of Trading Access Point (TAP) chapter
2.7	138	Addition/update of error message for client debarment

Preface

Purpose

This document describes the protocol to be used for Non-NEAT Front end (NNF) to communicate with the Securities Lending and Borrowing Market Trading System and thus serves as a development guide for the NNF users.

Target Audience

This document is written for system designers and programmers of user organizations and third party software developers who are responsible for the development of software to interact with NSE's SLBM Trading System.

Organization of This Document

This document is organized as follows:

Chapters	Description
Chapter 1	Provides a brief introduction to Non-NEAT Front end (NNF). It also details the NNF Terminal requirements, and TCP/IP Communication Network.
Chapter 2	Describes the general guidelines for the designers and programmers who develop NNF. It details the data types used and also covers the Message Header that is prefaced with all the structures.
Chapter 3	Describes how a trader logs on to the trading system. It also discusses the download of the updated information on the securities, participants and the status of the markets, and describes the log on request and the system responses.
Chapter 4	Describes entering fresh orders, modifying an existing order, and canceling outstanding orders.
Chapter 5	Covers the messages that are received on the interactive connection. These messages are received by users not in response to any request.
Chapter 6	Describes the end of the trading day activities. It covers the transmission of Security Bhav Copy.
Chapter 7	Describes the various Broadcast messages and the Compression and Decompression algorithm of Broadcast data.
Chapter 8	Encryption Decryption of Interactive Messages.
Chapter 9	Describes how member systems can directly connect to NSE for trading, while using existing formats of business messages from NNF API documents.

Chapters	Description
Chapter 10	Describes how exception at trading end should be handled.
Appendix	Lists the error, transaction and reason codes and also covers the various market statuses, market types and book types. Also covers security_slb.txt, participant_slb.txt and security.txt structures. Added Annexure For Encryption/Decryption

Abbreviations and Acronyms Used

The abbreviations and acronyms used in this document are:

AGM	Annual General Meeting
AON	All Or None
ATO	At The Opening
AU	Auction
BCID	Broadcast Circuit ID
BM	Branch Manager
CM	Corporate Manager
DL	Dealer
DQ	Disclosed Quantity
EGM	Extraordinary General Meeting
GTC	Good Till Cancellation
GTD	Good Till Date
IOC	Immediate Or Cancel
LTP	Last Traded Price
MBO	Market By Order
MBP	Market By Price
MF	Minimum Fill
NEAT	National Exchange for Automated Trading
NNF	Non Neat Front End
NSE	National Stock Exchange

NT	Negotiated Trade
OL	Odd Lot
OVS	Order Validation System
RC	ReCall order type
RL	Regular Lot
RP	RePay order type
SL	Stop Loss
SLBM	Securities Lending and Borrowing Market
ST	Special Terms
STPC	Self Trade Prevention Check
TM	Trading Member
TP	Trigger Price
TWS	Trader Workstation
VCID	Virtual Circuit ID
VSAT	Very Small Aperture Terminal
VV.RR.SS	Version. Release. Sub-release
WHS	Warehouse

CONTENTS

Chapter 1 Introduction

The National Stock Exchange of India Ltd (NSEIL) provides a fully automated screen based trading system, enabling trading members spread across the length and breadth of India to trade directly from their offices through an extensive telecommunication network. The system is known as 'National Exchange for Automated Trading' (NEAT) system. It adopts the principles of an order driven market, based on price-time priority. The trading members can use NEAT Front end or Non-NEAT Front end (NNF) to establish a network connection with the host system of National Stock Exchange (NSE) for trading. NNF is a front end which is developed and maintained by vendors other than NSE. NSE provides the NNF users with the general guideline document of the front end whereas they are supported by their respective vendors and NSE is not responsible for the performance of the NNF.

Communication Network

Connectivity to the Exchange is through WAN protocol on TCP/IP. There are two types of connectivity:

Interactive connectivity – On TCP/IP for user specific messages to and from the Exchange.Trading Front can connect to the exchange through a middleware application called Trading Access Point (TAP), The TAP protocol is defined in the annexure below.

Market data – The market data is sent as broadcast based on UDP connection Broadcast circuit ID (BCID) follows a unidirectional path which is from the host end to the TWS. All the broadcast data are transmitted through this broadcast circuit from the host end for all the traders. Since this is a one way connection, the data flow is always from the exchange (host end) to the trader terminal.

Chapter 2 General Guidelines

Introduction

This chapter provides general guidelines for the designers and programmers who develop NNF. It also provides information on data types and their size which can help in understanding various structures.

Message Structure Details

The message structure consists of two parts namely message header and message data. The message header consists of the fields of the header which is prefaced with all the structures.

The message data consists of the actual data that is sent across to the host or received from the host.

Transaction code, an important field of the message header, is a unique numeric identifier which is sent to or received from the trading system. This is used to identify the transaction between the TWS and the host end.

Guidelines for Designers

1. The order of the log-on messages should strictly be maintained as given in the following section (Chapter 3) of the document. Otherwise, the user cannot log on to the trading system.
2. All time fields are number of seconds from midnight January 1 1980.
3. No host-end inquiries are permitted for NNF users.
4. All price fields must be multiplied by 100 before sending to the host end and divided by 100 while receiving from the host end as the host system processes prices in paise.

Guidelines for Programmers

1. If your system uses little-endian order, the data types such as UINT, SHORT, LONG and DOUBLE contained in a packet, which occupy more than one byte should be twiddled (byte reversed). Twiddling involves reversing a given number of bytes such that the byte in ‘n’ position comes to the first position; the byte in (n-1) position comes to the

second position and so on. For example, if the value to be sent is 1A2B (hexadecimal), reverse the bytes to 2B1A. The same applies while receiving messages. So if the value received is 02BC, the actual value is BC02. So twiddle such data types before sending and after receiving to ensure that correct data is sent and received.

Note:

Twiddling is required because of the variety in endian order—big and little. A big-endian representation has a multibyte integer written with its most significant byte on the left. A little-endian representation, on the other hand, places the most significant byte on the right. The trading system host end uses big-endian order.

2. All alphabetical data, excluding password, must be converted to upper case before sending to the host. No NULL terminated strings should be sent to the host end. Instead, fill it with blanks before sending. The strings received from the host end are padded with blanks and are not NULL terminated.
3. All the structures should be defined in the following manner:
   - Items of type char or unsigned char, or arrays containing items of these types, are byte aligned.
   - Structures are word aligned.
   - All other types of structure members are word aligned.
4. All numeric data must be set to zero (0) before sending to the host, unless a value is assigned to it.
5. All reserved fields mentioned, should be mapped to CHAR buffer and initialized to NULL.

Note:

- The values of all the constants and transaction codes given in the document are listed in Appendix.
- The suffix IN in the transaction codes implies that the request is sent from the TWS to the host end whereas OUT implies that the message is sent from the host end to TWS

Data Types Used

Data Type	Size of Bytes	Signed / Unsigned
CHAR	1	Signed
UINT	2	Unsigned
SHORT	2	Signed
LONG	4	Signed
DOUBLE	8	Signed and Floating Point
BIT	1 bit	NA

Message Header

Each structure is prefaced with a MESSAGE_HEADER which is an interactive header. Some data in the header are fixed whereas some data are variable and set differently for each transaction code. The structure of the Message Header is as follows:

Table 1 MESSAGE HEADER

Structure Name	MESSAGE HEADER		
Packet Length	40 bytes		
Field Name	Data Type	Size in Byte	Offset
Reserved	CHAR	4	0
LogTime	LONG	4	4
AlphaChar [2]	CHAR	2	8
TransactionCode	SHORT	2	10
ErrorCode	SHORT	2	12
TimeStamp	LONG LONG	8	14
TimeStamp1 [8]	CHAR	8	22
TimeStamp2 [8]	CHAR	8	30
MessageLength	SHORT	2	38

The fields of Message Header are described below.

Field Name	Brief Description
LogTime	This field should be set to zero while sending messages.

AlphaChar	This field should be set to the first two characters of Symbol if the structure contains Symbol and Series; otherwise it should be set to **blank**.
TransactionCode	Transaction message number. This describes the type of message received or sent.
ErrorCode	This field should be set to zero while sending messages to the host. In the messages coming from the host, this field describes the type of error. Refer to *List of Error Codes* in Appendix.
TimeStamp	This field should be set to numeric zero while sending to the host. This is used in host end.
TimeStamp1	This field should be set to numeric zero while sending. This is the time the message arrives at the trading system host. In TimeStamp1, time is sent in jiffies from host end. This 8 byte data needs to be typecasted as first four byte into double variable and typecast the other four byte into another double variable. These values need to be used while requesting message area download in the same order.
TimeStamp2	This field should be set to numeric zero while sending to the host. For messages coming from the host, this field contains the machine number from which the packet is coming. In TimeStamp2, machine number is sent from host end.
MessageLength	This field should be set to the length of the entire message, including the length of message header while sending to host.

Broadcast Process Header

The broadcast messages like market open, market close, market in pre-open are prefaced with BCAST_HEADER. Some fields in the header are fixed. The remaining fields are variable and set differently for each transaction code. The structure of the BCAST_HEADER is as follows:

Table 2 BROADCAST_HEADER

Structure Name	BCAST_HEADER

Packet Length	40 bytes		
Field Name	Data Type	Size in Byte	Offset
Reserved	CHAR	4	0
LogTime	LONG	4	4
AlphaChar [2]	CHAR	2	8
TransCode	SHORT	2	10
ErrorCode	SHORT	2	12
BCSeqNo	LONG	4	14
Reserved	CHAR	4	18
TimeStamp2 [8]	CHAR	8	22
Filler2 [8]	CHAR	8	30
MessageLength	SHORT	2	38

Field Name	Brief Description
LogTime	This field should be set to zero whie sending to host end.For messages sent from host end this field contains the time when the message was generated by the trading system host.
AlphaChar	This field is set to the first two characters of Symbol if the structure contains Symbol and Series; otherwise it is set to blank.
TransactionCode	This field contains the transaction message number. This describes the type of message received or sent.
ErrorCode	This field contains the error number which describes the type of error. Refer to *List of Error Codes* in Appendix.
BCSeqNo	This field contains BCAST Sequence number for Ericcson switch.
TimeStamp2	This field contains the time when message is sent from the host.
Filler2	This field contains the machine number.
MessageLength	This field is set to the length of the entire message, including the length of the message header.

Note: BCAST_HEADER is prefaced with a system header which is of eight bytes

SEC INFO

SEC_INFO

Structure Name	SEC_INFO		
Packet Length	12 bytes		
Field Name	Data Type	Size in Byte	Offset
Symbol	CHAR	10	0
Series	CHAR	2	10

Field Name	Brief Description
Symbol	This field should contain the symbol of a security.
Series	This field should contain the series of a security.

Error Message

When the Error Code in the Message Header is not zero, ERROR RESPONSE is sent. The Error Message will describe the error received. The structure is as follows:

Table 3 ERROR_RESPONSE

Structure Name	ERROR RESPONSE		
Packet Length	180 bytes		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table1)	STRUCT	40	0
SEC_INFO (Refer sec_info)	STRUCT	12	40
Error Message	CHAR	128	52

Field Name	Brief Description
Symbol	This field should contain the symbol of a security.
Series	This field should contain the series of a security.
ErrorMessage	Stores the error message. Refer to *List of Error Codes* in Appendix.

Communication Network Connections for NNF Users

There are two types of virtual circuit connections used to communicate with the host end. One is the Interactive Virtual Circuit ID (VCID) and the other is the Broadcast Circuit ID (BCID).

Interactive VCID follows a bidirectional path between the NNF and NEAT to host end. All the interactive / request messages and its respective response follow through this channel. Even the unsolicited message such as trade message flows from exchange (host end) to the trader terminal through this channel.

Standard implementation of TCP/IP protocol exists on the exchange's infrastructure as a result of which default features like IP fragmentation, no QoS etc. continue to be enabled and available for use by members. Default IP fragmentation a valid feature in the TCP/IP protocol works at message level and usage of same by one member connection will not block or impact the messages of other member connections.

BCID follows a unidirectional path which is from the host end to the NFF / NEAT. All the broadcast data are transmitted through this broadcast circuit from the host end for all the traders. Since this is a one way connection, the data flow is always from the exchange (host end) to the trader terminal.

Member Guide to the Gateway Router Functionality

Currently Exchange publishes a list of gateway servers (NET) in the respective segments to which members can connect. Members have the choice of connecting to any of the gateway servers.

However, the members have represented that they are required to try to login on multiple gateway server sequentially before they are able to successfully login on the Exchange for trading activity. Thus, valuable time is lost by the member for trying to access the Exchange. The same is more severe during re-login / disconnections faced by the members.

In order to address these queries the Gateway Router Functionality has been proposed to be implemented.

1. It is now proposed that members will first connect to a gateway router server in the

respective segment details of which will be published by the Exchange.

2. The gateway router server will decide which gateway server is available for the member and will accordingly provide the details of the allocated gateway server to the member through the response message.
3. After getting the response message the member will need to connect to the allocated gateway server.

Thus, the process of allocating gateway servers becomes Exchange determined and highly simplified for the member.

The gateway router will decide the gateway server for the member for each trading day in the following manner:

1. The gateway router will maintain the used capacity of each gateway server. The gateway router will allocate least used gateway server (according to capacity). The capacity is based on the no. of messages allotted for each Box Id.
2. If all gateway servers have similar used capacity then a gateway server will be randomly allocated by the gateway router server.
3. Once a member has been provided session key with gateway server details by gateway router server, the member is expected to connect and login to the allocated gateway server at any time during rest of the trading day.
4. If the member gets logged off from the allocated gateway server, then the member has to request the gateway router server for getting new session key and gateway server details.
5. A member will be directed to the same gateway server by the gateway router server, once it has been allocated for the trading day.
6. Though the user will get directed to the same gateway, the user must ask the gateway router for getting the gateway details and session key as the old session key will be unique for that particular session and is cleaned up from the gateway once the user gets logged off.

7. Also, if the gateway has a failure during the day, the user will be allocated a new gateway server. This will be done transparently for the user by the gateway router server.

At the end of each trading day the gateway router server will clean up the used capacity, and will have the same capacity (full capacity) available for all gateway servers for the next day.

Chapter 3 Logon Process

Introduction

This section describes how a trader logs on to the trading system. It covers the log-on request and the system responses. This section also describes the download of the updated information on the securities, participants and the status of the markets. It covers the structures and field descriptions of System Information Download, Local Database Download and Message Download.

The process by which a trader logs on to the trading system is called Logon Process. The trader, after issuing a sign-on request, waits for the system response. The response could be a successful logon or an error message.

Order of Events to Be Followed During Logon and Logoff

The following sequence explains the order in which transaction codes are sent and received during log-on process.

Sequence No	Transaction Code	Sent By	Received By
1	*SIGN_ON_REQUEST_IN (2300)*	*TWS*	*Host End*
2	*SIGN_ON_REQUEST_OUT (2301)*	*Host End*	*TWS*
3	*SYSTEM_INFORMATION_IN (1600)*	*TWS*	*Host End*
4	*SYSTEM_INFORMATION_OUT (1601)*	*Host End*	*TWS*
5	*UPDATE_LOCALDB_IN (7300)*	*TWS*	*Host End*
6	*UPDATE_LOCALDB_HEADER (7307)*	*Host End*	*TWS*
7	*UPDATE_LOCALDB_DATA (7304)*	*Host End*	*TWS*
8	*UPDATE_LOCALDB_TRAILER (7308)*	*Host End*	*TWS*
9	*DOWNLOAD_REQUEST (7000)*	*TWS*	*Host End*
10	*HEADER_RECORD (7011)*	*Host End*	*TWS*
11	*MESSAGE_RECORD (7021)*	*Host End*	*TWS*
12	*TRAILER_RECORD (7031)*	*Host End*	*TWS*

The following sequence explains the order in which the transaction codes are sent and received during log-off process.

Sequence No	Transaction Code	Sent By	Received By
1	SIGN_OFF_REQUEST_IN (2320)	TWS	Host End
2	SIGN_OFF_REQUEST_OUT (2321)	Host End	TWS

Logon Request

When the user wants to establish an interactive circuit with the host, he sends this request. The structure for this request is as follows:

Table 5 SIGNON_IN

Structure Name	SIGNON IN		
Packet Length	212 bytes		
Transaction Code	SIGN_ON_REQUEST_IN (2300)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER(Refer Table 1)	STRUCT	40	0
UserId	LONG	4	40
Password	CHAR	8	44
NewPassword	CHAR	8	52
TraderName	CHAR	26	60
LastPasswordChangeDateTime	LONG	4	86
BrokerId	CHAR	5	90
Reserved	CHAR	1	95
BranchId	SHORT	2	96
VersionNumber	LONG	4	98
Reserved	CHAR	56	102
UserType	SHORT	2	158
SequenceNumber	DOUBLE	8	160
WorkstationNumber	CHAR	14	168
BrokerStatus	CHAR	1	182
ShowIndex	CHAR	1	183
BrokerEligibilityPerMarket (Refer **Table 5.1** for Small Endian machines and **Table 5.2** for Big Endian machines)	STRUCT	2	184

Structure Name	SIGNON IN		
Packet Length	212 bytes		
Transaction Code	SIGN_ON_REQUEST_IN (2300)		
Field Name	Data Type	Size in Byte	Offset
BrokerName	CHAR	26	186

For Small Endian Machines:

Table 5.1 BrokerEligibilityPerMarket

Structure Name	BrokerEligibilityPerMarket		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Reserved	BIT	4	0
Auction market	BIT	1	0
Spot market	BIT	1	0
Oddlot market	BIT	1	0
Normal market	BIT	1	0
Reserved	BIT	8	1

For Big Endian Machines:

Table 5.2 BrokerEligibilityPerMarket

Structure Name	BrokerEligibilityPerMarket		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Normal market	BIT	1	0
Oddlot market	BIT	1	0
Spot market	BIT	1	0
Auction market	BIT	1	0
Reserved	BIT	4	0
Reserved	BIT	8	1

Field Name	Description
TransactionCode	The transaction code is SIGN_ON_REQUEST_IN (2300).
UserId	This field should contain User ID of user/broker. This field accepts numbers only.

Field Name	Description
Password	This field should contain the password entered by the user. A combination of alphabet, numbers and special characters are allowed in the password. The user should enter the password for a successful Logon. When the user logs on for the first time the default password provided by NSE must be entered and the password should be changed by entering a new password.
NewPassword	This field should contain the new password entered by the user. This field should be entered only when the user wishes to change the password or the password has expired. Otherwise this field should be blank. The New Password should be entered along with the old password in the Password field. While logging on the system for the first time, the default password provided by NSE must be changed. the new password entered will undergo following new validations : • The length of password should be of exact 8 characters. • The password should contain atleast 1 upper case letter, 1 lower case letter, 1 numeral and 1 special characters from the list (@ # $ % & * / \). • New password must be different from previous 5 passwords. • User Id shall be locked after 3 invalid login attempts. • User shall not be allowed to set the default password as new password.
TraderName	This field when received from the host contains the user's name. This field should be sent to host as blanks.
LastPassword ChangeDateTime	This field should be set to numerical zero while log on.
BrokerId	This field should contain the trading member ID.
BranchId	This field should contain the Branch ID to which the broker belongs.
VersionNumber	This field should contain the version number of the trading system. It must be in the following format: VERSION.RELEASE.SUB_RELEASE (For example,

Field Name	Description
	01.00.01)
UserType	This field indicates the type of user. It can take one of the following values when it is sent from the host: '0' denotes Dealer '4' denotes Corporate Manager '5' denotes Branch Manager This field should be set to '0' while sending to the host.
SequenceNumber	This field should be set to numerical zero while sending the request to host.
WorkstationNumber	The network ID of the workstation should be provided. This is a seven digit number. The first five digits are fixed by the Exchange and represent the various ports / switch locations. The last two digits denote the user's PC - ID. It must be any number other than '00'.
BrokerStatus	This field should be set to blank.
BrokerEligibilityPer Market	This field should be set to numerical zero.
BrokerName	This field should be set to blank

Logon Response

The response will either be **Confirmation** or **Logon Error**.

Logon Confirmation Response

A successful logon results in the Logon Confirmation Response. The following structure is sent back:

Table 6 SIGNON OUT

Structure Name	SIGNON OUT		
Packet Length	212 bytes		
Transaction Code	SIGN_ON_REQUEST_OUT (2301)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
UserId	LONG	4	40
Password	CHAR	8	44
NewPassword	CHAR	8	52
TraderName	CHAR	26	60
LastPasswordChangeDate	LONG	4	86
BrokerId	CHAR	5	90
Reserved	CHAR	1	95
BranchId	SHORT	2	96
VersionNumber	LONG	4	98
EndTime	LONG	4	102
Reserved	CHAR	52	106
UserType	SHORT	2	158
SequenceNumber	DOUBLE	8	160
Reserved	CHAR	14	168
BrokerStatus	CHAR	1	182
Reserved	CHAR	1	183
BrokerEligibilityPerMarket (Refer Table 6.1 for Small Endian Machines and 6.2 for Big Endian Machines)	STRUCT	2	184
BrokerName	CHAR	25	186
char_filler3	CHAR	1	211

Table 6.1 BrokerEligibilityPerMarket (For Small Endian Machines)

Structure Name	BrokerEligibilityPerMarket		
Packet Length	2 bytes		
Field Name	Data Type	Size in Byte	Offset
Reserved	BIT	4	0
Auction market	BIT	1	0
Spot market	BIT	1	0
Oddlot market	BIT	1	0
Normal market	BIT	1	0
Reserved	BIT	8	1

Table 6.2 BrokerEligibilityPerMarket (For Big Endian Machines)

Structure Name	BrokerEligibilityPerMarket		
Packet Length	2 bytes		
Field Name	Data Type	Size in Byte	Offset
Normal market	BIT	1	0
Oddlot market	BIT	1	0
Spot market	BIT	1	0
Auction market	BIT	1	0
Reserved	BIT	4	0
Reserved	BIT	8	1

Field Name	Brief Description
TransactionCode	The transaction code is SIGN_ON_REQUEST_OUT (2301).
LogTime	The current time at the trading system is sent back as number of seconds since midnight of January 1, 1980. The time at the Trader workstation must be synchronized with this.
UserId	This field contains the ID of the user.
TraderName	This field contains the user name.
LastPassword ChangeDate	This filed contains the last date time when the password was changed.
BrokerId	This field contains the Trading Member ID.
BranchId	This field contains the branch ID of the particular user.
Version No	This field contains the version number of the trading system
EndTime	This field contains the time the markets last closed and is sent as the number of seconds since midnight of January 1, 1980. If this time is different from the time sent in an earlier log on, all orders, trades and messages for this trader must be deleted from the Local Database.
UserType	This field contains the type of user who is logging on: • '0' – Dealer • '4' – Corporate Manager

Field Name	Brief Description
	• '5' – Branch Manager
SequenceNumber	This field contains the time when the markets closed the previous trading day.
BrokerStatus	This field contains the current status of the broker: • 'S' for Suspended • 'A' for Active • 'D' for Deactivated • 'C' for Closeout
BrokerEligibility PerMarket	This structure specifies the markets that are allowed for the trading member. The trading member is eligible to enter orders in the markets that are set to 1.
BrokerName	This field contains the broker's name (trading member name).

Logon Error

In case of any error, the structure returned is:

ERROR RESPONSE (Refer to *Error Message* in Chapter 2)

Field Name	Description
TransactionCode	The transaction code is SIGN_ON_REQUEST_OUT (2301).
ErrorCode	This contains the error number. If the version number is not the same as at the host end, the version number at the host can be extracted from Error_Message in ERROR_RESPONSE (8 bytes from location 95 in the string). The format of it will be VV.RR.SS. The version number at the front end should be set to VVRRSS. Refer to *List of Error Codes* in Appendix.

System Information Download

The current status of the markets and the values of global variables are downloaded to the trader in response to *system information* request.

System Information Request

This request can be sent only if the user has logged on successfully. The format of the request is as follows:

Table 7 SYSTEM_INFO_REQ

Structure Name	SYSTEM_INFO_REQ		
Packet Length	40 bytes		
Transaction Code	SYSTEM_INFORMATION_IN (1600)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0

Field Name	Description
TransactionCode	The transaction code is SYSTEM_INFORMATION_IN (1600).

System Information Response

The following structure is returned as a response to the system information request:

Table 8 SYSTEM_INFORMATION_DATA

Structure Name	SYSTEM_INFORMATION_DATA		
Packet Length	88 bytes		
Transaction Code	SYSTEM_INFORMATION_OUT (1601)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
Normal	SHORT	2	40
Oddlot	SHORT	2	42
Spot	SHORT	2	44
Auction	SHORT	2	46
MarketIndex	LONG	4	48
DefaultSettlementPeriod (Normal)	SHORT	2	52
DefaultSettlementPeriod (Spot)	SHORT	2	54
DefaultSettlementPeriod (Auction)	SHORT	2	56
CompetitorPeriod	SHORT	2	58
SolicitorPeriod	SHORT	2	60

Structure Name	SYSTEM_INFORMATION_DATA		
Packet Length	88 bytes		
Transaction Code	SYSTEM_INFORMATION_OUT (1601)		
Field Name	Data Type	Size in Byte	Offset
WarningPercent	SHORT	2	62
VolumeFreezePercent	SHORT	2	64
Reserved	CHAR	2	66
TerminalIdleTime	SHORT	2	68
BoardLotQuantity	LONG	4	72
TickSize	LONG	4	76
MaximumGtcDays	SHORT	2	78
SECURITY ELIGIBLE INDICATORS(Refer Table 8.1 for Small Endian machines and Table 8.2 for Big Endian machines)	STRUCT	2	80
DisclosedQuantityPercentAllowed	SHORT	2	82
Reserved	CHAR	4	84

Table 8.1 SECURITY ELIGIBLE INDICATORS (For Small Endian Machines)

Structure Name	SECURITY ELIGIBLE INDICATORS		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Reserved	BIT	5	0
Books Merged	BIT	1	0
Minimum Fill	BIT	1	0
AON	BIT	1	0
Reserved	CHAR	1	1

Table 8.2 SECURITY ELIGIBLE INDICATORS (For Big Endian Machines)

Structure Name	SECURITY ELIGIBLE INDICATORS		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
AON	BIT	1	0
Minimum Fill	BIT	1	0
Books Merged	BIT	1	0
Reserved	BIT	5	0
Reserved	CHAR	1	1

Field Name	Brief Description
TransactionCode	The transaction code is SYSTEM_INFORMATION_OUT (1601).
AlphaChar	This field contains the number of streams present in the host from which message will be downloaded. **Note:** This field is present in the Message Header. This is of two bytes. Stream number will be populated in the first byte of Alphachar
MarketStatus	This field contains a value assigned for market status are: • '0' if it is Preopen • '1' if it is Open • '2' if it is Closed • '3' if it is Suspended In the pre-open state of the market, order entry is not allowed. The trading starts when the market is open. No orders can be entered for a security when the market is closed.
MarketIndex	This field contains the current market index.
SettlementPeriod	This field contains the default settlement period in various markets. Default Settlement (Normal), Default Settlement (Spot) and Default Settlement (Auction).
CompetitorPeriod	This field contains the default competitor period for auction.
SolicitorPeriod	This field contains the default solicitor period for auction.
WarningPercent	This field contains the warning percentage. If a broker exceeds his turnover by this value in percent, a warning message is broadcast to all traders.
VolumeFreezePercent	This field contains the volume freeze percentage. If a broker exceeds his turnover by this value in percent, the broker is deactivated and a message is broadcasted to all traders.
TerminalIdleTime	This field contains the idle time of the TWS terminal.
BoardLotQuantity	This field contains the board lot quantity. The regular lot order quantity must be a multiple of this quantity.
TickSize	This field contains the Tick size. The order price and the trigger price, if applicable, must be a multiple of this tick size.

Field Name	Brief Description
MaximumGTCDays	This field contains the maximum GTC days, that is, the maximum number of days after which a Good Till Canceled order will be canceled.
SecurityEligibilityIndicator	If the Minimum Fill flag is set, then orders will have the Minimum Fill attribute set. If the All Or None (AON) flag is set, then orders will have the AON attribute set.
DisclosedQuantity PercentAllowed	This field contains the disclosed quantity allowed percentage. The disclosed quantity, if set, will not be greater than this percent of the total quantity.

Update Local Database Download

The list of updated securities and participants is downloaded in response to *update local database* request. Any carried over GTC or GTD orders are also downloaded with this request.

Update Local Database Request

This message is sent to request the host end to update the local database at the front end. The structure sent is as follows:

Table 9 UPDATE_LOCALDB_IN

Structure Name	UPDATE_LOCALDB_IN		
Packet Length	58 bytes		
Transaction Code	UPDATE_LOCALDB_IN (7300)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
LastUpdateSecurityTime	LONG	4	40
LastUpdateParticipantTime	LONG	4	44
RequestForOpenOrders	CHAR	1	48
Reserved	CHAR	1	49
NormalMarketStatus	SHORT	2	50
OddLotMarketStatus	SHORT	2	52
SpotMarketStatus	SHORT	2	54
AuctionMarketStatus	SHORT	2	56

Field Name	Brief Description
TransactionCode	The transaction code is UPDATE_LOCALDB_IN (7300).
LastUpdateSecurityTime	This field should contain the time when the security information was last updated. This field is for each security for which information is downloaded. Further download requests can use the latest time to get updated information on the securities. Setting this time to zero results in complete download.
LastUpdateParticipantTime	This field should contain the time when the participant information was updated. This field is set for each participant for whom information is downloaded. Further download requests can use the latest time to get updated information on the participants. Setting this time to zero results in complete download.
RequestForOpenOrders	This field should be set to 'G' if GTC and GTD orders are to be downloaded. In other cases, it should be set to 'N'.
NormalMarketStatus	This field should contain the latest Normal Market status available at TWS.
OddLotMarketStatus	This field should contain the latest Odd Lot Market status available at TWS.
SpotMarketStatus	This field should contain the latest Spot Market status available at TWS.
AuctionMarketStatus	This field should contain the latest Auction Market status available at TWS.

Update Local Database Response

The response will be either the database download, or a partial system information download. The latter will occur if the trader does not have the latest market status.

Partial System Information Response

This is returned if the market status sent in the previous message is not the same at the host end or the markets are opening. In this case the market status at the host end is sent back in the MARKET STATUS as 'wait till markets are open'. The following structure is returned:

SYSTEM INFORMATION DATA (Refer to *System Information Response* in Chapter 3)

Field Name	Brief Description
TransactionCode	The transaction code is PARTIAL_SYSTEM_INFORMATION (7321).
MarketStatus	This contains the latest market status.

Update Local Database Download

The download comprises of a header, data and the trailer. Each updated security status, participant (if selected) and GTC/GTD order will be sent as a separate message.

Update Local Database Header

This is sent only to indicate that a sign-on download is going to commence. There is no additional data sent. The header is sent in the following format:

Table 10 UPDATE_LDB_HEADER

Structure Name	UPDATE_LDB_HEADER		
Packet Length	42 bytes		
Transaction Code	UPDATE_LOCALDB_HEADER (7307)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
Reserved	CHAR	2	40

Field Name	Description
TransactionCode	The transaction code is UPDATE_LOCALDB_HEADER (7307).

Update Local Database Data

The actual data is sent wrapped in another header. The outer header indicates that this message is part of the Update Local Database Data. The inner header indicates the type of data received.

The structure is as follows:

Table 11 UPDATE_LDB_DATA

Structure Name	UPDATE_LDB_DATA		
Packet Length	80 to 512 bytes		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
Data	CHAR	472	40

Field Name	Brief Description
TransactionCode	The transaction code is UPDATE_LOCALDB_DATA (7304).
InnerTransactionCode	The transaction codes sent are • BCAST_SECURITY_MSTR_CHG. It is determined by NSE-Control whether to send this or not. (Refer to *Change in Security Master* in Chapter 7) • BCAST_SECURITY_STATUS_CHG. This transaction code is sent when the status of the stock is different from the expected status at the host end (Refer to *Change of Security Status* in Chapter 7) • BCAST_PART_MSTR_CHG. If there is any change in the participant master after the time specified by the Last Update Participant Time, it is downloaded.(Refer to *Change Participant Status* in Chapter 7)

Update Local Database Trailer

This indicates that the download is complete. This is sent in the following format:

Table 12 UPDATE_LDB_HEADER

Structure Name	UPDATE_LDB_HEADER		
Packet Length	42 bytes		
Transaction Code	UPDATE_LOCALDB_TRAILER. (7308)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
Reserved	CHAR	2	40

Field Name	Description
TransactionCode	The transaction code is UPDATE_LOCALDB_TRAILER (7308).

Message Download

This request is used to download the messages intended for the trader from the trading system. When the trader makes a request for message download, all the transactions of the trader and other important broadcasts are downloaded. The response consists of Header and Trailer to indicate the beginning and end of download and is similar to Update Local Database Download.

Message Download Request

This message is sent for requesting message download. The structure sent to the trading system is:

Table 13 MESSAGE DOWNLOAD

Structure Name	MESSAGE DOWNLOAD		
Packet Length	48 bytes		
Transaction Code	DOWNLOAD_REQUEST (7000)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER(Refer Table 1)	STRUCT	40	0
SequenceNumber	DOUBLE	8	40

Field Name	Brief Description
TransactionCode	The transaction code is DOWNLOAD_REQUEST (7000).
Alpha_Char (Header)	This contains the stream number of the host to which it has to send the DOWNLOAD_REQUEST.
SequenceNumber	This contains the time last message was received by the workstation. This can be obtained from the Time Stamp1 of the MESSAGE_HEADER. To retrieve the messages from the beginning of the trading day, this field should be set to '0' or the Sequence Number received in the logon response message.

Message Download Response

The download comprises of a header, data and the trailer. Each trader specific and broadcast message will be sent as a separate message.

Message Download Header

This is only to indicate that a message download is going to commence. There is no additional data sent. The header is sent in the following format:

MESSAGE HEADER (Refer to *Message Header* in Chapter 2)

Field Name	Brief Description
TransactionCode	The transaction code is HEADER_RECORD (7011).

Message Download Data

The messages are similar to Update Local Database Data. The actual data is sent wrapped in another structure. The outer header indicates that this message is part of the Message Download Data. The inner header indicates the type of data received. The structure is shown below.

Table 14 MESSAGE HEADER

Structure Name	MESSAGE_HEADER		
Packet Length	80 to 512 bytes		
Transaction Code	MESSAGE_RECORD (7021)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER InnerHeader (Refer Table 1)	STRUCT	40	0
Data	CHAR	472	40

Field Name	Description
TransactionCode	The transaction code is MESSAGE_RECORD (7021).
InnerData	Set of transaction codes are received. They include **Trader Specific Messages** • Logon / Logoff response Refer to *Logon Process*, Chapter 3.

Field Name	Description
	• Interactive message sent to the user from the NSE-Control. Refer to *Unsolicited Messages*, Chapter 5. • Order entry, Modification, Cancellation responses Refer to *Order and Trade Management*, Chapter 4 • Trade Modification, Cancellation responses *Order and Trade Management*, Chapter 4. • Trade Confirmation, Stop Loss Trigger Refer to *Unsolicited Messages*, Chapter 5. **Broadcast Messages** • Market Open, Market Close, Market Pre-Open ended, Preopen Shutdown Message, Broadcast Message String, Turnover exceeded, Broker Reactivated, Broadcast message sent from NSE-Control. Refer to Broadcast, Chapter 7 Contingency Broadcast Message Refer to Exception Handling, Chapter 10

Message Download Trailer

This indicates that message download is complete. The format is as follows:

MESSAGE HEADER (Refer to *Message Header* in Chapter 2)

Field Name	Description
TransactionCode	The transaction code is TRAILER_RECORD (7031).

Logoff Request

The process by which a trader quits or signs off from the trading system is called Logoff Process.

The structure sent is:

MESSAGE HEADER (Refer to *Message Header* in Chapter 2).

Field Name	Description
TransactionCode	The transaction code is SIGN_OFF_REQUEST_IN (2320).

Logoff Confirmation Response

When the user logs on again, the user receives a packet giving the details of when he/she logged off. The structure sent is:

MESSAGE HEADER (Refer to *Message Header* in Chapter 2)

Note: MS_SIGNOFF message is sent in the Message Header itself. The length of the packet is 40 bytes.

Field Name	Description
TransactionCode	The transaction code is SIGN_OFF_REQUEST_OUT (2321).
LogTime	This field contains the current time at the trading system is sent back as number of seconds since midnight of January 1, 1980. The time at the workstation must be synchronized with this.

Chapter 4 Order and Trade Management

Introduction

This section describes about entering new orders, modifying existing orders, and canceling outstanding orders. The trader can begin entering the orders once he has logged on to the trading system and the market is in open state. Please note that for SLBM Buy indicates Borrow and Sell indicates Lend.

Order Entry

Order entry allows the trader to place orders in the market. The system accepts the orders from the users and tries to match the orders with the orders in the books immediately. If the order does not match, the order is placed in the appropriate book with the price and time stamp.

The trader can also place recall and repay orders. The recall and repay orders will be validated by the OVS system and if the order is valid, the order will get confirmed. This order gets traded in the system like a normal order.

The system also allows the trader to enter the trades negotiated outside. Both the parties involved in the trade have to enter the trade as negotiated trade entries. Negotiated trade will only be allowed for Regular Lot orders. The negotiated trade orders can only be “Day” orders.

Order Types

Regular Lot

Regular Lot Orders are orders in the normal market that have none of the following terms attached: All Or None, Minimum Fill and Trigger Price.

Special Terms

Special Terms Orders are orders in the normal market which have special attribute attached to it. They must have Minimum Fill (MF) or All Or None (AON).**However Special Term orders are not allowed in SLBM.**

Negotiated Trade Orders

Negotiated trade orders are regular lot orders with the Counter Party ID.

Stop Loss Orders

Stop Loss Orders are orders in normal market with Trigger Price specified.

Recall and Repay Orders

Recall orders are regular lot buy orders. Repay orders are regular lot sell orders.

Odd Lot Orders

Odd lot orders are orders in the Odd Lot Market with the order quantity being less than the Regular lot quantity.**However, oddlot orders are not allowed in SLBM**.

Spot Orders

Spot Orders are orders in spot market where the settlement period is different from the normal market and is fixed by the exchange. **However, spot orders are not allowed in SLBM.**

Auction Orders

Auction Orders are simple day orders and can only have the 'Day' term set to 1. **Auction orders are not allowed in SLBM.**

Order Terms

Following terms and conditions can be used during order entry and order modification.

Disclosed Quantity (DQ)

This term allows the dealer to disclose only a portion of the order quantity to the market. After the initial disclosed quantity is matched, subsequent disclosed quantity is shown to the market. All the disclosures will be shown to the market with the same order number.

Trigger Price (TP)

The Stop Loss book type allows the broker to release an order into the system after the market price crosses a threshold price referred to as the trigger price. This facility is available for orders in Normal market only. For a stop loss buy order, the trigger price should not be greater

than the limit price. For a stop loss sell order, the trigger price should not be less than the limit price. All the stop loss orders will be kept in a separate book till they are triggered.

Immediate or Cancel (IOC)

This term forces the order to match immediately, else be cancelled. If the order trades partially, the remaining part is cancelled.

Day

This is the default term for an order. At the end of the trading day, all outstanding Day orders are cancelled by the system.

Good till Date (GTD)

This term allows the dealer to keep an order in the system for a certain number of days. The number of days must be greater than 1 and less than or equal to the maximum number of days allowed for GTC orders. Each day is a calendar day. **However, GTD orders are not allowed in SLBM.**

Hence the Good till date field will be used for the purpose of reverse leg settlement date of every order related transaction viz, order entry, order modification, order cancellation.The order placed by the trader, should contain the reverse leg settlement date of that corresponding contract in this GTD field. And the same will be validated in the HE correspondingly.

Good till Cancelled (GTC)

This term allows the broker to keep an order in the system until it is canceled. However, the order is canceled by the system automatically if it remains outstanding for more than the maximum number of days allowed for GTC orders.

Minimum Fill (MF)

This term allows the broker to ensure that the quantity traded is at least the Minimum Fill amount specified. The minimum fill must be in multiples of the market lot and less than the order quantity. MF quantity must be less than or equal to Disclosed Quantity when the order has both MF and Disclosed Quantity attributes.

All or None (AON)

This term allows the broker to ensure that the entire order is traded and if not, nothing is traded at all. This can result in multiple trades or a single trade.

Rules of Order Entry

Order entry is not allowed in the following conditions:

- Markets are closed.
- Security is suspended.
- Security has matured.
- Security is expelled.
- Security admission date is greater than current date.
- Security is not eligible in the particular market.
- Security does not exist in the system.
- Broker is suspended.
- Broker does not exist in trading system.
- Broker is deactivated.
- User's branch order limit has exceeded.
- User is disabled.
- User is an inquiry user.
- User does not exist in trading system.
- Participant is suspended.
- Participant does not exist in trading system.
- Order price is beyond day's minimum maximum range.
- Trigger price is worse than limit price.
- Quantity is more than issued capital.
- Quantity is not equal to multiples of regular lot.
- Disclosed Quantity is more than the given percentage (determined by exchange) of order Quantity.
- Disclosed Quantity is more than order Quantity.

- Disclosed Quantity is not equal to multiples of regular lot.
- MF Quantity is more than order Quantity.
- MF Quantity is not a multiple of regular lot.
- Limit Price is not a multiple of Tick Size.
- Trigger Price is not a multiple of Tick Size.
- GTC/GTD days more than specified days.
- NT orders with GTC/GTD/IOC attribute.
- Spot orders with GTC/GTD.
- Auction orders with GTC/GTD/IOC.
- IOC and Disclosed Quantity combination.
- NT order with non existing Counter Party ID.
- NT order with suspended Counter Party ID.
- NT order with deactivated Counter Party ID.
- NT order with market price.
- Recall order type with sell
- Repay order type with buy
- Revserleg settlement date mismatch
- Recall and repay orders when security wise or SLBM market wise is disabled

Order Entry Request

The format of the order entry request is as follows:

Table 15 ORDER_ENTRY_ REQUEST

Structure Name	ORDER_ENTRY_REQUEST		
Packet Length	214 bytes		
Transaction Code	BOARD_LOT_IN (2000)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
ParticipantType	CHAR	1	40
Reserved	CHAR	1	41

Structure Name	ORDER_ENTRY_REQUEST		
Packet Length	214 bytes		
Transaction Code	BOARD_LOT_IN (2000)		
Field Name	Data Type	Size in Byte	Offset
CompetitorPeriod	SHORT	2	42
SolicitorPeriod	SHORT	2	44
ModCxlBy	CHAR	1	46
Filler9	CHAR	1	47
ReasonCode	SHORT	2	48
Reserved	CHAR	4	50
SEC_INFO (Refer sec_info)	STRUCT	12	54
AuctionNumber	SHORT	2	66
OpBrokerId	CHAR	5	68
Suspended	CHAR	1	73
OrderNumber	DOUBLE	8	74
AccountNumber	CHAR	10	82
BookType	SHORT	2	92
BuySell	SHORT	2	94
DisclosedVol	LONG	4	96
DisclosedVolRemaining	LONG	4	100
TotalVolRemaining	LONG	4	104
Volume	LONG	4	108
VolumeFilledToday	LONG	4	112
Price	LONG	4	116
TriggerPrice	LONG	4	120
ReverseLegSettlDate	LONG	4	124
EntryDateTime	LONG	4	128
MinFillAon	LONG	4	132
LastModified	LONG	4	136
ST_ORDER_FLAGS (Refer Table 15.1 for small endian machines and Table 15.2 for big endian machines)	STRUCT	2	140
BranchId	SHORT	2	142
TraderId	LONG	4	144
BrokerId	CHAR	5	148
OERemarks	CHAR	21	153
TransactionId	LONG	4	174
Settlor	CHAR	12	178
ProClient	SHORT	2	190

Structure Name	ORDER_ENTRY_REQUEST		
Packet Length	214 bytes		
Transaction Code	BOARD_LOT_IN (2000)		
Field Name	Data Type	Size in Byte	Offset
SettlementPeriod	SHORT	2	192
NNFField	DOUBLE	8	194
ExecTimeStamp	DOUBLE	8	202
Reserved	CHAR	2	210
OrderType	CHAR	2	212

For Small Endian Machines:

Table 15.1 ST_ORDER_FLAGS

Structure Name	ST_ORDER_FLAGS		
Packet Length	2 bytes		
Field Name	Data Type	Size in Bit	Offset
MF	BIT	1	0
AON	BIT	1	0
FOK	BIT	1	0
GTC	BIT	1	0
Day	BIT	1	0
OnStop	BIT	1	0
Mkt	BIT	1	0
ATO	BIT	1	0
Reserved	BIT	1	1
STPC	BIT	1	1
Reserved	BIT	1	1
Filler1	BIT	1	1
Frozen	BIT	1	1
Modified	BIT	1	1
Traded	BIT	1	1
MatchedInd	BIT	1	1

For Big Endian Machines:

Table 15.2 ST_ORDER_FLAGS

Structure Name	ST_ORDER_FLAGS		
Packet Length	2 bytes		
Field Name	Data Type	Size in Bit	Offset
ATO	BIT	1	0
Mkt	BIT	1	0
OnStop	BIT	1	0
Day	BIT	1	0
GTC	BIT	1	0
FOK	BIT	1	0
AON	BIT	1	0
MF	BIT	1	0
MatchedInd	BIT	1	1
Traded	BIT	1	1
Modified	BIT	1	1
Frozen	BIT	1	1
Filler1	BIT	1	1
Reserved	BIT	1	1
STPC	BIT	1	1
Reserved	BIT	1	1

The description and values of the fields are given below.

Field Name	Brief Description
TransactionCode	The transaction code is BOARD_LOT_IN (2000).
ParticipantType	Since only exchange can initiate the auction, this field should not be set to 'I' for initiator. This should be set to 'C' for competitor order and 'S' for solicitor order.
CompetitorPeriod	This field should be set to zero.
SolicitorPeriod	This field should be set to zero.
ModCxlBy	This field denotes which person has modified or cancelled a particular order. It should contain one of the following values: • 'T' for Trader • 'B' for Branch Manager • 'M' for Corporate Manager • 'C' for Control

Field Name	Brief Description
ReasonCode	This field contains the reason code for a particular order request rejection or order being frozen. This has the details regarding the error along with the error code. This field should be set to zero while sending the request to the host. Refer to *Reason Codes* in Appendix.
SEC_INFO (Refer sec_info)	This structure should contain the Symbol and Series of the security.
AuctionNumber	Auction number is available when initiation of auction is broadcast (Auction Status Change Broadcast). For an auction order, valid auction number should be given. For other books, this field should be set to zero.
OpBrokerId	This field should contain the counter party broker code for the negotiated trade order. This field is valid only for negotiated trade orders. For other books this field should contain blank.
Suspended	This field specifies whether the security is suspended or not. It should be set to blank while sending order entry request.
AccountNumber	If the order is entered on behalf of a trader, the trader account number should be specified in this field. For broker's own order, this field should be set to the broker code.
BookType	This field should contain the type of order. Refer to *Book Types* in Appendix.
BuySell	This field should specify whether the order is a buy or sell. It should take one of the following values. • '1' for Buy order • '2' for Sell order
DisclosedVol	This field should specify the quantity that has to be disclosed to the market. It is not applicable if the order has either the All Or None or the Immediate Or Cancel attribute set. It should not be greater than the volume of the order and not less than the Minimum Fill quantity if the Minimum Fill attribute is set. In either case, it cannot be less than the Minimum Disclosed Quantity allowed. It should be a multiple of the Regular lot.
DisclosedVolRemaining	This field contains the disclosed volume remaining from the original disclosed volume after trade(s). This should be set to zero while sending to the host.

Field Name	Brief Description
TotalVolRemaining	This field specifies the total quantity remaining from the original quantity after trade(s). For order entry, this field should be set to Volume. Thereafter, for every response the trading system will return this value.
Volume	This field should specify the quantity of the order placed. The quantity should always be in multiples of Regular Lot except for Odd Lot orders, and be less than the issued capital. The order will go for a freeze if the quantity is greater than the freeze quantity determined by NSE-Control.
VolumeFilledToday	This field contains the total quantity traded in a day.
Price	This field should contain the price at which the order is placed. To enter a Market order, the price should be zero. The price must be a multiple of the tick size. For Stop Loss orders, the limit price must be greater than the trigger price in case of a Buy order and less if it is a Sell order. Market attribute is not allowed for Negotiated orders. This is to be multiplied by 100 before sending to the trading system host.
TriggerPrice	This field is applicable only for a Stop Loss order and should be a multiple of the tick size. This field should contain the price at which the order is to be triggered and brought to the market. For a Stop Loss buy order, the trigger price will be less than or equal to the limit price but greater than the last traded price. For a Stop Loss sell order, the trigger price will be greater than or equal to the limit price but less than the last traded price. This is to be multiplied by 100 before sending to trading system.
ReverseLegSettlDate	This field should contain the Reverse Leg Settlment date of that particular contract.
EntryDateTime	This field should be set to zero while sending the order entry request.
MinimumFillAon	This field should contain the minimum fill quantity when the minimum fill attribute is set for an order. It should not be greater than either the volume of the order or the disclosed quantity and must be a multiple of the regular lot.
LastModified	If the order has been modified, this field contains the time when the order was last modified. It is the time in seconds from midnight of January 1 1980, This field should be set to zero for the order entry request

Field Name	Brief Description
	(it is same as Entry Date Time.)
Order_Flags	This structure specifies the attributes of an order. They are: • MF if set to 1, represents Minimum Fill attribute. • AON if set to 1, represents All Or None attribute. • IOC if set to 1, represents Immediate Or Cancel attribute. • GTC if set to 1, represents Good Till Cancel. • Day if set to 1, represents Day attribute. This is the default attribute. • SL if set to 1, represents Stop Loss attribute. • Market if set to 1, represents a Market order. • ATO if set to 1, represents a market order in PREOPEN. • STPC if set to 0, represents order resulting in self trade to be cancelled as per default action by the exchange. If set to 1, represents active order resulting in self trade to be cancelled o Order modification will be rejected if this bit is modified. In case of triggered stop loss order, bit selected during order entry will be considered. • Frozen if set to 1, represents the order has gone for a freeze. • Modified if set to 1, represents the order is modified. • Traded if set to 1, represents the order is traded partially or fully. • MatchedInd if set to 1, represents the NT order found a matching order. For a market order, the price should be 0. **If an attribute is not to be set, it should be set to 0.**
BranchId	This field should contain the ID of the branch of the particular broker.
TraderId	This field should contain the ID of the user. This field accepts only numbers.
BrokerId	This field should contain the trading member ID.

Field Name	Brief Description
OERemarks	This field may contain any remarks that the dealer can enter about the order in this field.
TransactionId	This field is used for transaction Id (Echo Back) Field. The value received from the FE will be sent back. There is no validation in the HE for this field.
Settlor	This field contains the ID of the participants who are responsible for settling the trades through the custodians. By default, all orders are treated as broker's own orders and this field defaults to the Broker Code.
ProClient	This field should contain one of the following values based on the order entering is on behalf of the broker or a trader. '1' – represents the client's order. '2' – represents a broker's order. '4' – represents warehousing order.
SettlementPeriod	This field should contain the number of days in a settlement cycle.
NNFField	This field should contain information of the NNF users First 6 digits will contain the pin code of the dealer. Next 3 digits will contain the Branch Number Next 3 digits will contain the terminal number. These should match with the details provided by the member to The Exchange. For internet trading the entire above mentioned 12-digits are to be populated as "111111111111" The 13th digit will bear a code to identify program "0" for automated trading and "1" for non-automated trading. The 14th digit will bear a code to identify the vendor. For members developing inhouse software it has to be populated with"0" The 15th digit is reserved for the use of Exchange and is to be populated as "0".
ExecTimeStamp	This field is used to store the time of writing to the order book. This should be set to zero while sending to the host.
Order Type	This field should conatin the order attribute as follows: Normal Order : Blank Recall Order (Buy Order) : RC Repay Order (sell Order) : RP

Order Entry Response

The response can be Order Confirmation, Order Freeze, Order Error or one of the general error responses. Order freeze response is generated when the order placed by the trader has resulted in freeze and is waiting for the approval of the exchange. The order error response is given when the entered order is rejected by the trading system. The reason for the rejection is given by Error Code.

Order Confirmation Response

Successful order entry results in Order Confirmation Response. The confirmed order is returned to the user. When the entered order goes for a freeze and that freeze is approved, this same transaction code is sent. This can be an unsolicited message as well. The message sent is as follows:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_CONFIRMATION (2073).
Suspended	This field contains 'C' if the broker is in Closeout.
OrderNumber	This field contains the order number assigned to the order. The first two digits will contain the stream number (This will be different from the stream number for Journal Download Request-Response). The next fourteen digits will contain fourteen digit sequence number.
Price	This field contains the price of the order. If a Market order was entered when market was in Open state, the 'Market' flag in Order Terms is set and is priced at the prevailing price at the trading system. If the market order is entered when the market was in preopen, the trading system sets the 'ATO' bit in Order Terms and prices at '-1'. If it was a priced order the order gets confirmed at that price.
Order_Flags	(Refer to *Order Entry Request* in Chapter 4)
EntryDateTime	This field contains the time at which order confirmed.

Market Price Confirmation Response

Market Price response is generated only when the order placed by the trader is a market order and the market order entered is not fully traded at exchange. This response is not expected for the limit orders. The response packet is sent only when there is any untraded quantity left in the order.

The message sent is:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is PRICE_CONFIRMATION (2012).
Price	This field contains the price of the order. . If a Market order was entered when market was in Open state, the 'Market' flag in Order Terms is set and price is set at the prevailing price at the trading system. If the market order is entered when the market was in preopen, this transcode is not received.
Order_Flags	(Refer to *Order Entry Request* in Chapter 4)

Order Freeze Response

Order freeze response is generated when the order placed by the trader or the order after modification is awaiting approval from the exchange. This response is not expected for Auction Orders. Exchange approval of the order results in a Freeze Approval Response and rejection results in Freeze Reject Response. These responses are sent as unsolicited messages. The format sent is as follows:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is FREEZE_TO_CONTROL (2170).
Order_Flags	(Refer to *Order Entry Request* in Chapter 4)

Order Error Response

The order error response is sent when the entered order is rejected by the trading system. The reason for the rejection is given by the reason code and the reason string. The message sent is:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_ERROR (2231).
ErrorCode	This field contains the error number. Refer to *List of Error Codes* in Appendix.
Suspended	This field contains 'C' if the broker is in Closeout.

Order Modification

Order Modification enables the trader to modify unmatched orders. All order types except Auction can be modified.

Rules of Order Modification

The following modifications are not allowed:

- Buy to Sell or vice versa.
- Modifying Symbol and Series.
- Modifying Participant field.
- Modifying Pro/Cli field.
- Modifying Frozen orders.
- BM modifying CM's orders.
- DL modifying BM's orders.
- DL modifying CM's orders.
- Modifying non existing order.
- Inquiry user trying to modify.
- Modifying an order in such a way that it results in a branch order value to be exceeded.

- Modifying Auction orders.
- Modifying NT order once it results in an alert.
- Modifying deactivated broker's orders.
- Changing of original data.
- Modifying NT, AU, SP, OL book type fields.
- Modifying RC and RP attribute order type

Note: RL/ST/SL book types can be toggled between themselves only. They cannot be modified to NT or AU or SP or OL.

Order Modification Request

The trader can modify the quantity, price and attributes of an order by specifying the order number of the order to be modified. The message sent is:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_MOD_IN (2040).
OrderNumber	This should contain order number which is the identity of the order to be modified.
Last ModifiedTime	This should contain time of last activity done on that order. Last activity could be order entry, order modification or last trade time of that order. It is in number of seconds from midnight of January 1, 1980.

Note: The other fields of order modification request are same as the fields of order entry request.

Order Modification Response

This response is sent back when an order modification is requested. This does not imply that the order modification is confirmed. The response can be Order Modification Confirmation, Order Freeze, Order Modification Error or one of the general error responses. The order modification error response is given when the modified order is rejected by the trading system.

The reason for the rejection is given by the reason code and the reason string. The message sent is of the following format:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_MOD_OUT (2041).

Order Modification Confirmation Response

Successful modification of the order results in Order Modification Confirmation. When the order modification is confirmed, the order-modified time is filled and sent back. On modification, the order can result in a freeze. If the freeze is approved, order modification will be received as an "Unsolicited Message". The structure sent is as follows:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_MOD_CONFIRMATION (2074).
Last ModifiedTime	This field contains the time when the order was last modified. It is in number of seconds from midnight of January 1, 1980,
EntryDateTime	This field contains the time at which last modified by user. It is in number of seconds from midnight of January 1, 1980,

Order Modification Error Response

The reason for rejection is given by the Error Code in the header. The message sent is as follows:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_MOD_REJECT (2042).

Effect of Modifying the Terms of an Order on Price-Time Priority

Field	Can Change	Comment
Buy/Sell	No	
Order Type		Refer to Rules of Order Modification
Symbol	No	
Series	No	
ReverseLegSettlDate	NO	
Price	Yes	Changing the order price will always result in the order losing its time priority.
Quantity	Yes	The quantity of an order can be reduced any number of times without the order losing its time priority. However, increasing the quantity of an order will always result in the order losing its time priority.
PRO/CLI	No	
Account No.	No	
Day	Yes	Changing to or from a Day order retains time priority
GTC	Yes	Changing to or from a GTC order retains time priority
GTD	Yes	Changing to or from a GTD order retains time priority
Days in GTD	Yes	
DQ	Yes	Same as in quantity. In addition, the changed DQ should not be more than the remaining order quantity. Changing to or from DQ results in loss of time priority.
MF & AON	Yes	Changing MF to AON order or vice-versa will result in the order losing its time priority.
MF	Yes	Same as in Quantity.
SL	Yes	A SL order can be changed to a normal limit order or a Special Terms order by removing the SL attribute. The SL limit and trigger price can also be changed. In each of these cases the order loses its time priority.

Field	Can Change	Comment
Participant	No	
Remarks	Yes	Changing this does not change time priority.
Note: When the order quantity of an ATO or 'Market' order is modified, the order loses priority irrespective of increase or decrease in the quantity.		

Order Cancellation

The trader can cancel any unmatched/partially matched order by specifying the order number.

Rules of Order Cancellation

- CM can cancel BM's and DL's order, but BM and DL cannot cancel CM's order.
- BM can cancel DL's order, but DL cannot cancel BM's order.
- Deactivated broker cannot cancel his/her order.
- NT orders cannot be cancelled once it results in an alert.

Order Cancellation Request

The format of the message is as follows:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_CANCEL_IN (2070).
OrderNumber	This field should contain the order number which is the identity of the order to be cancelled.
Last ModifiedTime	This should contain time of last activity done on that order. Last activity could be order entry, order modification or last trade time of that order. It is in number of seconds from midnight of January 1, 1980,

Order Cancellation Response

The response can be Order Cancellation Confirmation, Order Cancellation Error or one of the general error responses.

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_CANCEL_OUT (2071).

Order Cancellation Confirmation Response

Successful cancellation of order results in Order Cancellation Confirmation Response. This will be an "Unsolicited Message" if NSE-Control cancels the order. The message sent is as follows:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_CANCEL_CONFIRMATION (2075).
Suspended	This field contains 'C' if the broker is in Closeout.

Order Cancellation Error Response

The order cancellation error is sent when the cancellation request is rejected by the trading system. The reason for rejection is given by the Error Code in the header. The message sent is as follows:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is ORDER_CANCEL_REJECT (2072).

Trade Modification

Currently, trade modification is not allowed.

Trade Cancellation

To cancel a trade, both the parties of the trade must request for trade cancellation. As soon as the request reaches the trading system, a requested message is sent. If any error is encountered in the entered data, Trade Error message is sent. Otherwise it goes to the NSE-

Control as an alert. The counter party to the trade is notified of the trade cancellation request (Refer to *Unsolicited Messages* in Chapter 5). When both the parties of the trade ask for trade cancellation, it may be approved or rejected by the Exchange (Refer to *Unsolicited Messages* in Chapter 5).

Trade Cancellation Request

The format of the message is as follows:

Table 16 TRADE_INQUIRY_DATA

Structure Name	TRADE INQUIRY DATA		
Packet Length	114 bytes		
Transaction Code	TRADE INQUIRY DATA (5440)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER(Refer Table 1)	STRUCT	40	0
SEC_INFO (Refer sec_info)	STRUCT	12	40
ReverseLegSettlDate	LONG	4	52
lFillNumber	LONG	4	56
lFillQty	LONG	4	60
lFillPrice	LONG	4	64
iMktType	SHORT	2	68
lNewVolume	LONG	4	70
BuyParticipantId	CHAR	12	74
SellParticipantId	CHAR	12	86
BuyBrokerId	CHAR	5	98
SellBrokerId	CHAR	5	103
TraderId	LONG	4	108
RequestedBy	SHORT	2	112

Field Name	Brief Description
TransactionCode	The transaction code is TRADE_CANCEL_IN (5440).
FillNumber	This field should contain the trade number of the trade to be cancelled.

Trade Cancellation Requested Response

This is an acknowledgement signifying that the request has reached the trading system.

TRADE INQUIRY DATA (Refer to *Trade Cancellation Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is TRADE_CANCEL_OUT (5441).

Trade Cancellation Error

After the requested response, if any error is detected in the data, the following structure is sent:

TRADE INQUIRY DATA (Refer to *Trade Cancellation Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is TRADE_ERROR (2223)
ErrorCode	Refer to *List of Error Codes* in Appendix.

Chapter 5 Unsolicited Messages

Introduction

This section details the unsolicited messages that are received on the interactive connection. These messages are not received by the users in response to any request.

Cancellation of Orders in Batch

The Day orders which are not traded are deleted in the batch at the end of the day. GTC\GTD orders which are valid till date, if not traded, are also removed from the book. A response for the same is sent to the user. The structure sent is:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is BATCH_ORDER_CANCEL (9002).

Stop Loss Order Triggering

When any stop loss order entered is triggered, the user who entered the order is sent the following message:

TRADE CONFIRM (Refer to *Trade Confirmation* discussed later in this chapter)

Field Name	Brief Description
TransactionCode	The transaction code is ON_STOP_NOTIFICATION (2212).

Freeze Approve Response

This message is sent when a previous order, which resulted in freeze, is approved by the Exchange. The format of the message is as follows:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction codes are: • If the entered order went for a freeze, and then got freeze approval, ORDER_CONFIRMATION (2073). • If the modified order went for a freeze, and then got freeze approval, ORDER_MOD_CONFIRMATION (2074).
LastModifiedDateTime	This field contains the time when the order was last modified.

Freeze Reject Response

This message is sent when a previous order, which resulted in freeze, is rejected by the Exchange. The format of the message is as follows:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction codes are: • If the entered order went for a freeze then for freeze reject ORDER_ERROR_OUT (2231). • If the modified order went for a freeze then for freeze reject ORDER_MOD_REJECT_OUT (2042).

Trade Confirmation:

Trade confirmation is an unsolicited message which is generated when any order of the trader is traded. The order may trade completely or partially. The following structure is sent:

Table 17 MS_TRADE_CONFIRM

Structure Name	MS_TRADE_CONFIRM		
Packet Length	154 bytes		
Transaction Code	TRADE_CONFIRMATION (2222)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER(Refer Table 1)	STRUCT	40	0
ResponseOrderNumber	DOUBLE	8	40
OrderType	CHAR	2	48
BrokerId	CHAR	5	50
Reserved	CHAR	1	55
TraderNum	LONG	4	56
AccountNum	CHAR	10	60
BuySell	SHORT	2	70
OriginalVol	LONG	4	72
DisclosedVol	LONG	4	76
RemainingVol	LONG	4	80
DisclosedVolRemaining	LONG	4	84
Price	LONG	4	88
ST_ORDER_FLAGS (Refer Table 15.1 for small endian machines and Table 15.2 for big endian machines)	STRUCT	2	92
Gtd	LONG	4	94
FillNumber	LONG	4	98
FillQty	LONG	4	102
FillPrice	LONG	4	106
VolFilledToday	LONG	4	110
ActivityType	CHAR	2	114
ActivityTime	LONG	4	116
OpOrderNumber	DOUBLE	8	120
OpBrokerId	CHAR	5	128
SEC_INFO (Refer sec_info)	STRUCT	12	133
Reserved	CHAR	1	145
BookType	SHORT	2	146
NewVolume	LONG	4	148
ProClient	SHORT	2	152

Field Name	Brief Description
TransactionCode	The transaction code is TRADE_CONFIRMATION (2222).
ResponseOrderNumber	This field contains the order number of the trader's order taking part in the trade.
Order type	This field contains the order type of the order under trade. Will be blank for normal and 'RC' and 'RP' for repay orders.
BrokerId	This field contains the Trading Member ID.
TraderNum	This field contains the trader's or user ID.
AccountNum	This field contains the Account Number or Client code.
BuySell	This field contains one of the following values based on Buy or Sell. • '1' for Buy • '2' for Sell.
OriginalVol	This field contains the Original traded volume.
DisclosedVol	This filed contains the quantity to be disclosed to the market. It is not applicable if the order has either the All Or None or the Immediate Or Cancel attribute set. It should not be greater than the volume of the order and not less than the Minimum Fill quantity if the Minimum Fill attribute is set. In either case, it cannot be less than the Minimum Disclosed Quantity allowed. It should be a multiple of the Regular lot.
RemainingVol	This field contains the volume remaining after trade(s).
DisclosedVolRemaining	This field contains the disclosed volume remaining after trade(s).
Price	This field contains the order price.
OrderFlags	Refer to *Order Entry Request* in Chapter 4.
Gtd	This field contains the reverse leg settlement date of the contract.
FillNumber	This field contains the trade number.
FillQty	This field contains the traded volume.
FillPrice	This field contains the price at which order is traded.

Field Name	Brief Description
VolFilledToday	This field contains the quantity traded today.
ActivityType	This field contains the activity type. • 'B' for Buy • 'S' for Sell
ActivityTime	This field contains the time when the activity took place.
OpOrderNumber	This field contains the order number of the counter order taking part in the trade.
OpBrokerId	This field contains the Trading Member ID of the counter party taking part in the trade.
SEC_INFO (Refer sec_info)	This field contains the Symbol and Series of the security.
BookType	This field contains the book type - RL/ ST/ SL/ NT/ OL/ SP/ AU. Refer to *Book Types* in Appendix.
NewVolume	This field is always set to zero for trade confirmation.
ProCli	This field is same as Pro/Client /WHS indicator having one of the following values: • '1' - client's order • '2' - broker's order • '4' - warehousing order

Trade Cancellation Requested Notification

This message is sent when the counter party of the trade requests a trade cancellation. The structure sent is:

MS_TRADER_INT_MSG (Refer to *Interactive/Broadcast MessagesSent from Control* discussed later in this chapter)

Field Name	Brief Description
TransactionCode	The transaction code is CTRL_MSG_TO_TRADER (5295).

Trade Cancellation Confirmation Response

When NSE-Control approves the trade cancellation request the structure sent is:

TRADE CONFIRM (Refer to *Trade Confirmation* discussed earlier in this chapter)

Field Name	Brief Description
TransactionCode	The transaction code is TRADE_CANCEL_CONFIRM (2282).

Trade Cancellation Rejection

When NSE-Control rejects the trade cancellation alert the structure sent is:

TRADE CONFIRM (Refer to *Trade Confirmation* discussed earlier in this chapter)

Field Name	Brief Description
TransactionCode	The transaction code is TRADE_CANCEL_REJECT (2286).

Negotiated Order Entered By Counter Party

Whenever a negotiated order is entered and the counter party ID is same as the user's broker ID then all the users under this broker are notified. The structure is:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
TransactionCode	The transaction code is NEG_ORDER_BY_CPID (2009).

Negotiated Trade Approval Response

When a negotiated trade is approved by NSE-Control the following structure is sent:

TRADE_CONFIRM (Refer to *Trade Confirmation* discussed earlier in this chapter)

Field Name	Brief Description
TransactionCode	The transaction code is TRADE_CONFIRMATION (2222).

Negotiated Trade Reject Response

After the alert is generated for a negotiated trade, the NSE-Control can reject the trade, send both the orders to the regular lot book or send one to the regular lot and cancel the other. The structure sent is:

ORDER ENTRY REQUEST (Refer to *Order Entry Request* in Chapter 4)

Field Name	Brief Description
Transaction Code	• The party whose order is sent to regular lot receives the following transaction code: NEG_ORDER_TO_BL (2008). • The party whose order is cancelled gets the following transaction code: CANCEL_NEG_ORDER (2076).
EntryDateTime	This field contains the order time at wich CWS user approves for 2008 transcode

Interactive/Broadcast Messages Sent from Control

A message can be sent to the trader(s) from the NSE-Control Work Station. If it is sent to all the traders, it comes as a broadcast in the structure BROADCAST_MESSAGE. (Refer to *General Message Broadcast* in Chapter 7)

When the message is sent to a particular user, it comes as an interactive message in the following structure:

Table 18 MS_TRADER_INT_MSG

Structure Name	MS_TRADER_INT_MSG		
Packet Length	289 bytes		
Transaction Code	CTRL_MSG_TO_TRADER (5295)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER(Refer Table 1)	STRUCT	40	0
TraderId	LONG	4	40
ActionCode	CHAR	3	44
Reserved	CHAR	1	47
MsgLength	SHORT	2	48

Structure Name	MS_TRADER_INT_MSG		
Packet Length	289 bytes		
Transaction Code	CTRL_MSG_TO_TRADER (5295)		
Field Name	Data Type	Size in Byte	Offset
Msg	CHAR	239	50

Field Name	Brief Description
TransactionCode	The transaction codes are: • CTRL_MSG_TO_TRADER (5295) for interactive messages
ActionCode	This field contains the action code to indicate the action taken. For example, 'SYS' - System 'AUI' - Auction Initiation 'AUC' - Auction Complete 'LIS' – Listing

Table 19 MS_TRADER_INT_MSG

Structure Name	MS_TRADER_INT_MSG		
Packet Length	297 bytes		
Transaction Code	BCAST_JRNL_VCT_MSG (6501)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER(Refer Table 1)	STRUCT	40	0
BranchNumber	SHORT	2	40
BrokerNumber	CHAR	5	42
ActionCode	CHAR	3	47
Reserved	CHAR	4	50
BROADCAST DESTINATION (Refer Table 19.1and Table 19.2)	STRUCT	2	54
MsgLength	SHORT	2	56
Msg	CHAR	239	58

For Small Endian Machines:

Table 19.1 BROADCAST DESTINATION

Structure Name	BROADCAST DESTINATION		
Packet Length	2 bytes		
Field Name	Data Type	Size in Byte	Offset
Reserved	BIT	7	0
TraderWs	BIT	1	0
Reserved	CHAR	1	1

For Big Endian Machines:

Table 19.2 BROADCAST DESTINATION

Structure Name	BROADCAST DESTINATION		
Packet Length	2 bytes		
Field Name	Data Type	Size in Byte	Offset
TraderWs	BIT	1	0
Reserved	BIT	7	0
Reserved	CHAR	1	1

Field Name	Brief Description
TransactionCode	BCAST_JRNL_VCT_MSG (6501) for broadcasting messages.
ActionCode	This field contains the action code to indicate the action taken. For example, 'SYS' - System 'AUI' - Auction Initiation 'AUC' - Auction Complete 'LIS' – Listing

Chapter 6 Bhav Copy

Introduction

This section describes the end of the trading day activities. It covers the transmission of Security Bhav Copy. This takes place after the markets close for the day.

Security Bhav Copy

Message Stating the Transmission of Security Bhav Copy Will Start Now

This is the first message which is broadcasted saying that the bhav copy will be started now. The structure sent is:

BROADCAST MESSAGE (Refer to *General Message Broadcast* in Chapter 7)

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_JRNL_VCT_MSG (6501). Message: Security Bhav Copy is being broadcast now.

Header of Report on Market Statistics

A header precedes the actual bhav copy that is sent to the trader. The message structure sent is:

Table 20 MS_RP_HDR

Structure Name	MS_RP_HDR		
Packet Length	108 bytes		
Transaction Code	MARKET_STATS_REPORT_DATA (1833)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
MsgType	CHAR	1	40
Reserved	CHAR	1	41
ReportDate	LONG	4	42
UserType	SHORT	2	46
BrokerId	CHAR	5	48
BrokerName	CHAR	25	53

Structure Name	MS_RP_HDR		
Packet Length	108 bytes		
Transaction Code	MARKET_STATS_REPORT_DATA (1833)		
Field Name	Data Type	Size in Byte	Offset
TraderNumber	LONG	4	78
TraderName	CHAR	26	82

Field Name	Brief Description
TransactionCode	The transaction code is MARKET_STATS_REPORT_DATA (1833).
MsgType	This field is set to 'H' denoting Header
ReportDate	This field is set to the report date.
UserType	This field contains the type of user. This is set to '-1'.
BrokerId	This field contains Trading Member ID. This is set to blanks.
BrokerName	This field contains the name of the broker. This is set to blanks.
TraderNumber	This field contains the trader/user ID. This is set to zero.
TraderName	This field contains the name of the trader. This is set to blanks.

Report on Market Statistics

This is the actual data that is sent for the report. The structure is as follows:

Table 21 REPORT MARKET STATISTICS

Structure Name	REPORT MARKET STATISTICS		
Packet Length	478 bytes		
Transaction Code	MARKET_STATS_REPORT_DATA (1833)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER(Refer Table 1)	STRUCT	40	0
MessageType	CHAR	1	40
Reserved	CHAR	1	41
NumberOfRecords	SHORT	2	42
MARKET STATISTICS DATA (Refer Table 21.1)	STRUCT	434	44

Table 21.1 MARKET STATISTICS DATA

Structure Name	MARKET STATISTICS DATA		
Packet Length	62 bytes		
Field Name	Data Type	Size in Byte	Offset
SEC_INFO (Refer sec_info)	STRUCT	12	0
ReverseLegSettlDate	LONG	4	12
MarketType	SHORT	2	16
OpenPrice	LONG	4	18
HighPrice	LONG	4	22
LowPrice	LONG	4	26
ClosingPrice	LONG	4	30
TotalQuantityTraded	LONG	4	34
TotalValueTraded	DOUBLE	8	38
PreviousClosePrice	LONG	4	46
LifeTimeHigh	LONG	4	50
LIfeTimeLow	LONG	4	54
CorporateActionIndicator	CHAR	4	58

Field Name	Brief Description
TransactionCode	The transaction code is MARKET_STATS_REPORT_DATA (1833).
MessageType	This field is set to 'R' denoting Report Data.
NumberOfRecords	This field contains the number of markets for which Market Statistics is being sent. In a packet at most 7 records can be packed.
Symbol	This field contains the Symbol of the security.
Series	This field contains the series of a security.
ReverseLegSettlDate	This field contains the reverse leg settlement date of the contract.
MarketType	This field contains one of the following values indicating the market type as: • '1' – Normal • '2' – Odd lot • '3' – Spot • '4' – Auction

Field Name	Brief Description
OpenPrice	This field contains the open price of a security.
HighPrice	This field contains the highest trade price.
LowPrice	This field contains the lowest trade price.
ClosingPrice	This field contains the closing price of a security.
TotalQuantityTraded	This field contains the total quantity of the security that is traded today.
TotalValueTraded	This field contains the total value of the securities traded.
PreviousClosePrice	This field contains the previous day's closing price of the security.
LifeTimeHighPrice	This field contains the highest trade price in a security.
LifeTimeLowPrice	This field contains the lowest trade price in a security.
CorporateActionIndicator	This field contains the Corporate Action. The EGM, AGM, Interest, Bonus, Rights and Dividend flags are set depending on the corporate action.

Packet Indicating Data for Depository Securities Begins

This message indicates that hereafter the bhav copy for depository securities will be broadcast. The structure sent is:

REPORT MARKET STATISTICS (Refer to *Report on Report on Market Statistics* discussed earlier in this chapter)

Field Name	Brief Description
TransactionCode	The transaction code is MARKET_STATS_REPORT_DATA (1833).
MessageType	This field is set to 'D' denoting Data.

Data for Depository Securities

This is same as the data packet for non-Depository securities. The structure sent is:

REPORT MARKET STATISTICS (Refer to *Report on Market Statistics* discussed earlier in this chapter)

Field Name	Brief Description
TransactionCode	The transaction code is MARKET_STATS_REPORT_DATA (1833).

Trailer Record

This indicates that the transmission of bhav copy ends here. The structure is:

Table 22 REPORT TRAILER

Structure Name	REPORT TRAILER		
Packet Length	46 bytes		
Transaction Code	MARKET_STATS_REPORT_DATA (1833)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER(Refer Table 1)	STRUCT	40	0
MessageType	CHAR	1	40
NumberOfRecords	LONG	4	41
Reserved	CHAR	1	45

Field Name	Brief Description
TransactionCode	The transaction code is MARKET_STATS_REPORT_DATA (1833).
MessageType	This filed is set as 'T' for trailer record.
NumberOfRecords	This field contains the number of data packets sent in the bhav copy.

Note: The Bhavcopy which is broadcasted from CM will be dropped and will not be received by SLBM users in NEAT.

Chapter 7 Broadcast

Introduction

This section describes the Compression and Decompression algorithm of Broadcast data and the various Broadcast messages with their structures.

The following information of Capital Market will also be broadcasted to the SLBM users in NEAT system:

1. MBP information of Capital Market for Normal and Retdebt market.
2. MBO information of CM for Oddlot and Auction Market.
3. Ticker and Index Broadcast of Capital Market.
4. Market Watch updation of Capital Market securities.
5. All Broadcast messages from Capital Market.

Structures are same for both CM/SLBM.

Compression of the Broadcast Data

The broadcast traffic from the exchange which gives the on-line quotes to the trading terminals has been continually increasing, especially during market open and market close. To accommodate the increased broadcast traffic, the exchange has come up with a compression algorithm to compress some of the specific broadcast transaction codes, which are as follows:

Transaction Code	Represents
7200	MBO/MBP.
7201	Mkt Watch
7202	Ticker
7208	Only MBP

LZO compression algorithm is used to compress the above specified broadcast transaction codes. The details of the LZO compression algorithm are described below.

The LZO stands for Lempel Ziv Oberhaumer. This algorithm is freely available on the internet (URL: http://www.oberhumer.com/opensource/lzo). It is made available by free software foundation. The algorithm is tested on various operating systems like UNIX and red hat Linux.

Decompression Routine

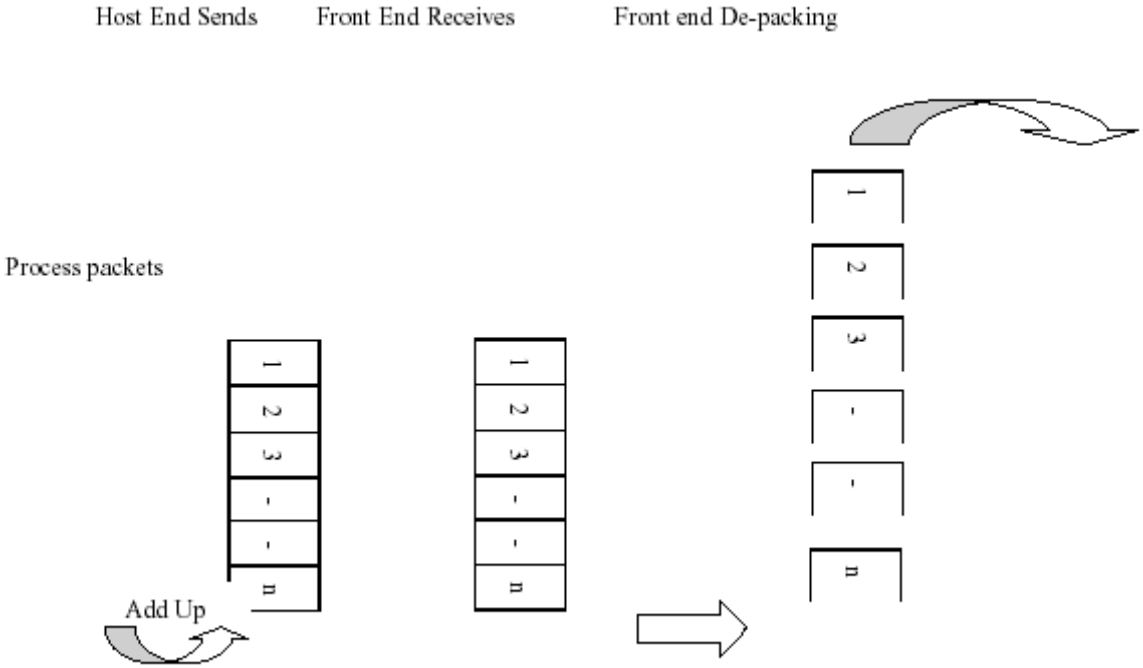
NSE will provide the object file containing the decompression routine.

Sequential Packing

To improve the effective data transfer, the idea of sequential packing along with the lzo compression algorithm has been incorporated. At the host end, sequential packing algorithm packs the incoming data packets, which is then transmitted over the network. The data packets are packed in FIFO order.

For example,
If 'n' packets are packed in a buffer, they are arranged in the following order:
1st packet will be stored at the first place in the buffer, 2nd Packet will be stored at the second place, and so on.

At the front end while de packing the buffer, the packets are to be segregated in the same order, that is, isolate each packet and process each packet as per the sequence viz- first packet first and last packet at the end. The packets within a buffer may be an admixture of compressed and uncompressed data packets.



Calling Convention

The decompression routine is a C-callable routine with the following prototype:

l

```
Void Sigdec2 (char *ip,
              unsigned short  *ipL,
              char *op,
              unsigned short *opL,
              unsigned short *errorcode);
```

Parameters

Ip: it is the pointer to the input buffer.

IpL: It is the pointer to a short containing the length of input.

Op: it is the pointer of the output buffer.

OpL: It is the pointer to a short containing the length of output.

ErrorCode: it is the pointer to a short containing the error code.

Packet Format

Incoming packet at the front end can be interpreted by mapping onto the following structure.

```
Struct {
```

```
    CHAR   cNetId [2]
    SHORT iNoPackets
    CHAR   cPackData [512]
}   BcastPackData
```

where,

cNetId[2]	Identifies the machine (CM broadcast or SLBM Broadcast)
iNoPackets	The number of packets that are sequentially packed
cPackData	Buffer containing all the packets.

The buffer when mapped to, by the above structure, the number of packets in the buffer can be known. The next task is to segregate the packets and process the individual packets.

The packets received through the broadcast traffic have to be interpreted as follows

```
COMPRESSION_BROADCAST_DATA
{
        SHORT       CompressionLen
        CHAR        BroadcastData [ ]
}
```

Note:

- The first two bytes of the broadcast packet indicate the length of the data after compression.
- If the compression length is zero, the data received is not compressed.
- If the length is non-zero, the data following the length should be decompressed by using the decompression routine.
- Inside the broadcast data, the first 8 bytes before the message header should be ignored. The message header starts from the 9th byte.

Implementation at Front End

The lzo directory (lzo1.07) contains all the lzo source, header and library files.

These files are to be included while building an application.

lzo1z_decompress is used for decompression. This is a function of the lzo library.

An API has to be developed to encompass the above LZO decompression function.

The syntax of the call should be:

lzo_decomp (char* inp_buff, unsigned int* inp_len, char* buffer_decomp,
unsigned int *output_len, unsigned short *errorCode)

Where, lzo_decomp is a function of the API (to be developed by referring to the examples specified in the lzo 1.07 directory) that calls the lzo function for decompression "lzo1z_decompress"

Inp_buff	Specifies the input buffer (Compressed Buffer)
Inp_len	Specifies the length of input buffer (Compressed Length)
Buffer_decomp	Specifies the Buffer after decompression
output_len	Specifies the length after decompression (Out put length)
errorCode	Specifies the error code

The syntax of the lzo decompress function is as follows:

lzo1z_decompress (out, decomp_inlen, in, & decomp_outlen, NULL)

Where

out	Specifies input compressed buffer
decomp_inlen	Specifies the input length of the buffer (Length of Compressed buffer)
in	Specifies the out put (decompressed) buffer
decomp_outlen	Specifies the output length of the decompressed buffer

Note:

Inside the broadcast data, the first byte indicates the market type. Ignore the rest of the 7 bytes before message header. If the first byte has the value of '**4**', it is **Capital market** and if it is '**7**' then it is of **SLBM.**

The message header starts from 9th byte.

General Message Broadcast

Any general message is broadcasted in the following structure. The structure sent is:

Table 23 BROADCAST_MESSAGE

Structure Name	BROADCAST MESSAGE		
Packet Length	297 bytes		
Transaction Code	BCAST_JRNL_VCT_MSG (6501)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER	Table 1	40	0
BranchNumber	SHORT	2	40
BrokerNumber	CHAR	5	42
ActionCode	CHAR	3	47
Reserved	CHAR	4	50
BROADCAST DESTINATION (Refer Table No. 23.1 for small endian & Table No. 23.2 for big endian)	STRUCT	2	54
BroadcastMessageLength	SHORT	2	56
BroadcastMessage	CHAR	239	58

Table 23.1 BROADCAST_DESTINATION (For Small Endian Machines)

Structure Name	BROADCAST_DESTINATION		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Reserved	BIT	7	0
TraderWs	BIT	1	0
Reserved	CHAR	1	1

Table 23.2 BROADCAST_DESTINATION (For Big Endian Machines)

Structure Name	BROADCAST_DESTINATION		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
TraderWs	BIT	1	0
Reserved	BIT	7	0
Reserved	CHAR	1	1

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_JRNL_VCT_MSG (6501).
BranchNumber	This field contains the branch number of the trader or broker.
BrokerNumber	This field contains the Trading Member ID of the broker.
ActionCode	This field Indicates the action taken. For example, 'SYS' - System 'AUI' - Auction Initiation 'AUC' - Auction Complete 'LIS' - Listing
BroadcastDestination	This field contains the destination of the message, that is, Trader Workstation or Control Workstation.
BroadcastMessageLength	This field contains the length of the broadcast message.
BroadcastMessage	This field contains the broadcast message.

Change in System Status / Parameters

This message is sent when any global operating parameters are changed or status of markets is changed. The structure of the message is:

SYSTEM INFORMATION DATA (Refer to *System Information Response* in Chapter 3)

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_SYSTEM_INFORMATION_OUT (7206)

Change in Security Master

This is sent whenever the parameter of any security is changed. The structure is given below.

Table 24 SECURITY UPDATE INFORMATION

Structure Name	SECURITY UPDATE INFORMATION		
Packet Length	240 bytes		
Transaction Code	BCAST_STOCK_MSTR_CHG (7305)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
Token	LONG	4	40
SEC_INFO (Refer sec_info)	STRUCT	12	44
InstrumentType	SHORT	2	56
PermittedToTrade	SHORT	2	58
IssuedCapital	DOUBLE	8	60
WarningPercent	SHORT	2	68
FreezePercent	SHORT	2	70
CreditRating	CHAR	17	72
SECURITY ELIGIBILITY PER MARKET [4] (refer table 24.1 for small endian & 24.2 for big endian)	STRUCT	12	89
IssueRate	SHORT	2	101
IssueStartDate	LONG	4	103
InterestPaymentDate	LONG	4	107
IssueMaturityDate	LONG	4	111
BoardLotQuantity	LONG	4	115
TickSize	LONG	4	119
Name	CHAR	25	123
Reserved	CHAR	1	148
ListingDate	LONG	4	149
ExpulsionDate	LONG	4	153
ReAdmissionDate	LONG	4	157
RecordDate	LONG	4	161
ExpiryDate	LONG	4	165
NoDeliveryStartDate	LONG	4	169
NoDeliveryEndDate	LONG	4	173
ELIGIBLITY INDICATORS (refer table 24.3 for small endian & 24.4 for big	STRUCT	2	177

Structure Name	SECURITY UPDATE INFORMATION		
Packet Length	240 bytes		
Transaction Code	BCAST_STOCK_MSTR_CHG (7305)		
Field Name	Data Type	Size in Byte	Offset
endian)			
BookClosureStartDate	LONG	4	179
BookClosureEndDate	LONG	4	183
PURPOSE structures(refer table 24.5 for small endian & 24.6 for big endian)	STRUCT	2	187
LocalUpdateDateTime	LONG	4	189
DeleteFlag	CHAR	1	193
Remark	CHAR	25	194
FaceValue	LONG	4	219
ISINNumber	CHAR	12	223

Table 24.1 SECUIRITY ELIGIBILITY PER MARKET (For Small Endian Machines)

Structure Name	SECUIRITY ELIGIBILITY PER MARKET		
Packet Length	3 bytes		
Field Name	Data Type	Size	Offset
Reserved	BIT	7	0
Eligibility	BIT	1	0
Status	SHORT	2	1

Table 24.2 SECUIRITY ELIGIBILITY PER MARKET (For Big Endian Machines)

Structure Name	SECUIRITY ELIGIBILITY PER MARKET		
Packet Length	3 bytes		
Field Name	Data Type	Size in Byte	Offset
Eligibility	BIT	1	0
Reserved	BIT	7	0
Status	SHORT	2	1

Table 24.3 ELIGIBLITY INDICATORS (For Small Endian Machines)

Structure Name	ELIGIBLITY INDICATORS		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
AllowRepay	BIT	1	0
AllowRecall	BIT	1	0
Reserved	BIT	1	0
Reserved	BIT	1	0
Reserved	BIT	1	0
MinimumFill	BIT	1	0
AON	BIT	1	0
ParticipateInMarketIndex	BIT	1	0
Reserved	CHAR	1	1

Table 24.4 ELIGIBLITY INDICATORS (For Big Endian Machines)

Structure Name	ELIGIBLITY INDICATORS		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
ParticipateInMarketIndex	BIT	1	0
AON	BIT	1	0
MinimumFill	BIT	1	0
Reserved	BIT	1	0
Reserved	BIT	1	0
Reserved	BIT	1	0
AllowRepay	BIT	1	0
AllowRecall	BIT	1	0
Reserved	CHAR	1	1

Table 24.5 PURPOSE (For Small Endian Machines)

Structure Name	PURPOSE		
Packet Length	2 bytes		
Field Name	Data Type	Size in Byte	Offset
Reserved	BIT	2	0
EGM	BIT	1	0
AGM	BIT	1	0
Interest	BIT	1	0

Structure Name	PURPOSE		
Packet Length	2 bytes		
Field Name	Data Type	Size in Byte	Offset
Bonus	BIT	1	0
Rights	BIT	1	0
Dividend	BIT	1	0
Reserved	CHAR	1	1

Table 24.6 PURPOSE (For Big Endian Machines)

Structure Name	PURPOSE		
Packet Length	2 bytes		
Field Name	Data Type	Size in Byte	Offset
Dividend	BIT	1	0
Rights	BIT	1	0
Bonus	BIT	1	0
Interest	BIT	1	0
AGM	BIT	1	0
EGM	BIT	1	0
Reserved	BIT	2	0
Reserved	CHAR	1	1

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_SECURITY_MSTR_CHG (7305).
Token	This field contains the token number of the security being updated. This is unique for a particular symbol-series combination.
SecurityInformation	This field contains the Symbol and Series (FL) of the security.
InstrumentType	This field contains the instrument type of the security. It can be one of the following: ▪ '0' – Equities ▪ '1' – Preference Shares ▪ '2' – Debentures ▪ '3' – Warrants ▪ '4' – Miscellaneous

Field Name	Brief Description
PermittedToTrade	This field contains one of the following values: • '0' – Listed but not permitted to trade • '1' – Permitted to trade
IssuedCapital	This field contains issue size of the security.
WarningPercent	This field contains the warning percent of outstanding volume.
FreezePercent	This field contains the volume freeze percent.
CreditRating	This field contains the credit rating of the security.
Eligibility	The flag is set to '1' if the security is allowed to trade in a particular market.
Status	This field contains one of the following values: • '1' - Preopen (Only for Normal Market) • '2' - Open • '3' - Suspended • '4' - Preopen extended
IssueRate	This field contains the price of the issue.
IssueStartDate	This field contains the date of issue of the security.
InterestPaymentDate	This field contains the interest payment date of the issue.
ReverseLegSettlDate	This field contains the reverse leg settlement date of the contract.
BoardLotQuantity	This field contains the Regular lot size.
TickSize	This field contains the Tick size/ Min spread size.
Name	This field contains the security name.
ListingDate	This field contains the date of listing.
ExpulsionDate	This field contains the date of expulsion.
ReAdmissionDate	This field contains the date of readmission.
RecordDate	This field contains the date of record changed.
ExpiryDate	This field contains the last date of trading before any corporate action.
NoDeliveryStartDate	This field contains the date from when physical delivery of share certificates is stopped for book closure.

Field Name	Brief Description
NoDeliveryEndDate	This field contains the date from when physical delivery of share certificates starts after book closure.
MinimumFill	This flag is set if Minimum Fill attribute is allowed in orders of this security.
AON	This flag is set if AON attribute is allowed in orders of this security.
ParticipateInMarketIndex	This flag is set if this security participates in the market index.
AllowRecall	This flag indicates whether the recall order is eligible for the contract
AllowRepay	This flag indicates whether the repay order is eligible for the contract
BookClosureStartDate	This field contains the date when the record books in the company for shareholder names starts.
BookClosureEndDate	This field contains the date when the record books in the company for shareholder names ends.
Purpose	This field contains the EGM / AGM / Interest / Bonus / Rights / Dividend flags set depending on the corporate action.
LocalUpdateDateTime	This field contains the local database update date and time.
DeleteFlag	This field contains the status of the security, that is, whether the security is deleted or not.
Remark	This field contains remarks.
FaceValue	This field contains face value of the security.
ISIN Number	This field contains ISIN number of the security.

Change Participant Status

This message is sent whenever there is any participant change. The structure sent is:

Table 25 Change Participant Status

Structure Name	PARTICIPANT UPDATE INFO		
Packet Length	83 bytes		
Transaction Code	BCAST_PART_MSTR_CHG (7306)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER(refer Table 1)	STRUCT	40	0

Structure Name	PARTICIPANT UPDATE INFO		
Packet Length	83 bytes		
Transaction Code	BCAST_PART_MSTR_CHG (7306)		
Field Name	Data Type	Size in Byte	Offset
ParticipantId	CHAR	12	40
ParticipantName	CHAR	25	52
ParticipantStatus	CHAR	1	77
ParticipantUpdateDateTime	LONG	4	78
DeleteFlag	CHAR	1	82

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_PART_MSTR_CHG (7306).
ParticipantId	This field contains the Participant ID.
ParticipantName	This field contains the name of the participant that is changed.
ParticipantStatus	This field contains the status of the participant which is changed: • 'S' for Suspended • 'A' for Active
ParticipantUpdateDateTime	This field contains the time when the participant information was changed. It is in number of seconds from January 1, 1980.
DeleteFlag	This field indicates whether the participant is deleted: • 'Y' means deleted • 'N' means not deleted

Change of Security Status

This message is sent whenever the status of any security changes. The structure sent is:

Table 26 Change of Security Status

Structure Name	SECURITY STATUS UPDATE INFORMATION		
Packet Length	462 bytes		
Transaction Code	BCAST_SECURITY_STATUS_CHG (7320) OR BCAST_SECURITY_STATUS_CHG_PREOPEN (7210)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer	STRUCT	40	0

Structure Name	SECURITY STATUS UPDATE INFORMATION		
Packet Length	462 bytes		
Transaction Code	BCAST_SECURITY_STATUS_CHG (7320) OR BCAST_SECURITY_STATUS_CHG_PREOPEN (7210)		
Field Name	Data Type	Size in Byte	Offset
Table 1)			
NumberOfRecords	SHORT	2	40
TOKEN AND ELIGIBILITY [35] (Refer table 26.1)	STRUCT	420	42

Table 26.1 TOKEN AND ELIGIBILITY

Structure Name	TOKEN AND ELIGIBILITY		
Packet Length	12 bytes		
Field Name	Data Type	Size in Byte	Offset
Token	LONG	4	0
SECURITY STATUS PER MARKET[4] (Refer table 26.2)	STRUCT	8	2

Table 26.2 SECURITY STATUS PER MARKET

Structure Name	SECURITY STATUS PER MARKET		
Packet Length	2 bytes		
Field Name	Data Type	Size in Byte	Offset
Status	Short	2	0

Field Name	Brief Description
TransactionCode	The transaction code is: • When the status of the security changes BCAST_SECURITY_STATUS_CHG (7320). • BCAST_SECURITY_STATUS_CHG_PREOPEN (7210).
NumberOfRecords	This field contains the number of records of the structure TOKEN AND ELIGIBILITY.
Token	This field contains the token number of the security which has been changed.
Status	This field contains the new status of the security. This can take one of the following values:

Field Name	Brief Description
	• '1' - Preopen • '2' - Open • '3' - Suspended • '4' - Preopen extended

Turnover Limit Exceeded or Broker Reactivated

When a broker's turnover limit exceeds, the broker is deactivated and a message is broadcast to all workstations. The same structure is also sent when any broker is reactivated. The structure is:

Table 27 Turnover Limit Exceeded or Broker Reactivated

Structure Name	BROADCAST LIMIT EXCEEDED		
Packet Length	77 bytes		
Transaction Code	BCAST_TURNOVER_EXCEEDED (9010) OR BROADCAST_BROKER_ACTIVATED (9011)		
Field Name	Data Type	Size in Byte	Offset
BCAST_HEADER (Refer Table 2)	STRUCT	40	0
BrokerCode	CHAR	5	40
CounterBroker Code	CHAR	5	45
WarningType	SHORT	2	50
SEC_INFO (Refer sec_info)	STRUCT	12	52
TradeNumber	LONG	4	64
TradePrice	LONG	4	68
TradeVolume	LONG	4	72
Final	CHAR	1	76

Field Name	Brief Description
TransactionCode	The transaction code is: • BCAST_TURNOVER_EXCEEDED (9010), if the broker turnover is about to exceed or has already exceeded. • BROADCAST_BROKER_ACTIVATED (9011), if the broker is reactivated after being deactivated.

Field Name	Brief Description
BrokerCode	This field contains the Broker code who is about to exceed or has already exceeded his turnover limit. If the transaction code is BROADCAST_BROKER_ACTIVATED, then this broker is reactivated.
CounterBrokerCode	This field is not in use.
WarningType	This field is applicable only if the transaction code is BCAST_TURNOVER_EXCEEDED. The value is '1' if the turnover limit is about to exceed, '2' if turnover limit is exceeded. In the latter case the broker is deactivated.
Symbol	This field is applicable only if the transaction code is BCAST_TURNOVER_EXCEEDED. This contains the symbol of the security in which the broker has last traded.
Series	This field is applicable only if the transaction code is BCAST_TURNOVER_EXCEEDED. This contains the series of the security.
TradeNumber	This field is applicable only if the transaction code is BCAST_TURNOVER_EXCEEDED. This is the trade number in which the broker has last traded
TradePrice	This field is applicable only if the transaction code is BCAST_TURNOVER_EXCEEDED. This contains the price of the trade.
TradeVolume	This field is applicable only if the transaction code is BCAST_TURNOVER_EXCEEDED. This contains the trade quantity of the trade.
Final	This field is applicable only if the transaction code is BCAST_TURNOVER_EXCEEDED. This indicates whether it is the final auction trade.

Auction Activity Message

This structure is sent whenever there is any auction related activity **in CM.** This includes any change in Auction MBO. The structure is:

Table 28 Auction Activity Message

Structure Name	MS_AUCTION_INQ_DATA		
Packet Length	78 bytes		
Transaction Code	BCAST_AUCTION_INQUIRY_OUT (6582).		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER	Table 1	40	0
ST_AUCTION_INQ_INFO (Refer Table 28.1)	STRUCT	38	40

Table 28.1 Auction Activity Message

Structure Name	ST_AUCTION_INQ_INFO		
Packet Length	38 bytes		
Field Name	Data Type	Size in Byte	Offset
Token	LONG	4	0
AuctionNumber	SHORT	2	4
AuctionStatus	SHORT	2	6
InitiatorType	SHORT	2	8
TotalBuyQty	LONG	4	10
BestBuyPrice	LONG	4	14
TotalSellQty	LONG	4	18
BestSellPrice	LONG	4	22
AuctionPrice	LONG	4	26
AuctionQty	LONG	4	30
SettlementPeriod	SHORT	2	34

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_AUCTION_INQUIRY_OUT (6582).
Token	This field contains the token number of the security in which the auction is started.
AuctionNumber	This field contains the number of the auction.
AuctionStatus	Refer to Appendix.
InitiatorType	This field specifies whether auction is initiated by trader or control. This field is set to control since only Exchange initiated auctions are permitted now.
TotalBuyQty	This field contains the total Buy Quantity for the auction.

Field Name	Brief Description
BestBuyPrice	This field contains the best Buy price. This is the highest price for a Buy auction.
TotalSellQty	This field contains the total Sell quantity for the auction.
BestSellPrice	This field contains the best Sell price. This is the lowest price for a Sell auction.
AuctionPrice	This field contains the price at which auction trade has taken place.
AuctionQty	This field contains the quantity of securities that have been auctioned.
SettlementPeriod	This field contains the period by which settlement between the parties should take place. This value is defaulted by the Exchange and cannot be modified by the user.

Change of Auction Status

When the status of an auction changes **in CM** (from pending to active or, competitor period or solicitor period is ended or started) a message is broadcast to all workstations with the following structure and transaction codes:

Table 29 Change of Auction Status

Structure Name	AUCTION STATUS CHANGE		
Packet Length	301 bytes		
Transaction Code	BC_AUCTION_STATUS_CHANGE (6581)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER (Refer Table 1)	STRUCT	40	0
SEC_INFO (Refer sec_info)	STRUCT	12	40
AuctionNumber	SHORT	2	52
AuctionStatus	CHAR	1	54
ActionCode	CHAR	3	55
BROADCAST_DESTINATION (Refer Table 29.1 for small endian & 29.2 for big endian)	STRUCT	2	58
BroadcastMessageLength	SHORT	2	60
BroadcastMessage	CHAR	239	62

Table 29.1 BROADCAST_DESTINATION (For Small Endian Machines)

Structure Name	BROADCAST_DESTINATION		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Reserved	BIT	7	0
TraderWs	BIT	1	0
Reserved	CHAR	1	1

Table 29.2 BROADCAST_DESTINATION (For Big Endian Machines)

Structure Name	BROADCAST DESTINATION		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
TraderWs	BIT	1	0
Reserved	BIT	7	0
Reserved	CHAR	1	1

Field Name	Brief Description
TransactionCode	The transaction code is BC_AUCTION_STATUS_CHANGE (6581).
Symbol	This field contains the symbol of the security.
Series	This field contains the series of the security.
AuctionNumber	This field contains the auction number.
AuctionStatus	This field contains the status of the auction. Refer to Auction Status in Appendix.
ActionCode	This field contains the action code to indicate the action taken. For example, 'AUI' - Auction Initiation 'AUC' - Auction Complete
BroadcastDestination	This field contains the destination of the message. For example, it is for the Trader Workstation.

Field Name	Brief Description
BroadcastMessageLength	This field contains the length of the broadcast message.
BroadcastMessage	This field contains the contents of the broadcast message.

Change of Market Status

Whenever the status of the market changes, the following structure is sent:

Table 30 Change of Market Status

Structure Name	BCAST_VCT_MESSAGES		
Packet Length	297 bytes		
Transaction Code	BC_OPEN_MESSAGE (6511) OR BC_CLOSE_MESSAGE (6521) OR BC_PREOPEN_SHUTDOWN_MSG (6531) OR BC_NORMAL_MKT_PREOPEN_ENDED (6571)		
Field Name	Data Type	Size in Byte	Offset
BCAST_HEADER (Refer Table 2)	STRUCT	40	0
SEC_INFO (Refer sec_info)	STRUCT	12	40
MarketType	SHORT	2	52
BROADCAST_DESTINATION (refer table 30.1 for small endian & 30.2 for big endian)	STRUCT	2	54
BroadcastMessageLength	SHORT	2	56
BroadcastMessage	CHAR	239	58

Table 30.1 BROADCAST_DESTINATION (For Small Endian Machines)

Structure Name	BROADCAST_DESTINATION		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Reserved	BIT	7	0
TraderWs	BIT	1	0
Reserved	CHAR	1	1

Table 30.2 BROADCAST_DESTINATION (For Big Endian Machines)

Structure Name	BROADCAST DESTINATION		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
TraderWs	BIT	1	0
Reserved	BIT	7	0
Reserved	CHAR	1	1

Field Name	Brief Description
TransactionCode	The transaction codes are as follows: • BC_OPEN_MESSAGE (6511). This is sent when the market is opened. • BC_CLOSE_MESSAGE (6521). This is sent when the market is closed. • BC_PREOPEN_SHUTDOWN_MSG (6531). This is sent when the market is preopened. • BC_NORMAL_MKT_PREOPEN_ENDED (6571). This is sent when the preopen period ends.
SecurityInformation	This field contains the symbol and series of a security.
MarketType	This field indicates the type of market. It contains one of the following values: • '1' - Normal • '2' - Odd Lot • '3' - Spot • '4' – Auction
BroadcastDestination	This field is set to '1' if it signifies that the message is for the Trader Workstation.
BroadcastMessageLength	This field contains the length of the broadcast message.
BroadcastMessage	This field contains the contents of the broadcast message.

Ticker and Market Index

Ticker and market index information is sent in the following structure:

Ticker structure for CM/SLBM:

Table 31 Ticker and Market Index

Structure Name	TICKER TRADE DATA		
Packet Length	492 bytes		
Transaction Code	BCAST_TICKER_AND_MKT_INDEX (7202)		
Field Name	Data Type	Size in Byte	Offset
BCAST_HEADER (Refer Table 2)	STRUCT	40	0
NumberOfRecords	SHORT	2	40
TICKER INDEX INFORMATION [25] (Refer to TABLE No. 31.1)	STRUCT	450	42

Table 31.1 TICKER INDEX INFORMATION

Structure Name	TICKER INDEX INFORMATION		
Packet Length	18 bytes		
Field Name	Data Type	Size in Byte	Offset
Token	LONG	4	0
MarketType	SHORT	2	2
FillPrice	LONG	4	4
FillVolume	LONG	4	8
MarketIndexValue	LONG	4	12

Field Name	Brief Description
TransactionCode	The transaction code sent is BCAST_TICKER_AND_MKT_INDEX (7202).
NumberOfRecords	This field indicates the number of times (Maximum 28) the structure TICKER INDEX INFORMATION is repeated.
Token	This field contains the token number—a unique number given to a particular symbol-series combination.
MarketType	This field contains the market type.
FillPrice	This field contains the price at which the order has been traded.
FillVolume	This field contains the quantity of security traded.
MarketIndexValue	This field contains the value of the market index.

Market by Order / Market by Price Update

The information regarding the best buy orders and the best sell orders is given in the following format:

Table 32 Market by Order / Market by PriceUpdate

Structure Name	BROADCAST MBO MBP		
Packet Length	434 bytes		
Transaction Code	BCAST_MBO_MBP_UPDATE (7200)		
Field Name	Data Type	Size in Byte	Offset
BCAST_HEADER (Refer Table 2)	STRUCT	40	0
INTERACTIVE MBO DATA (Refer Table 32.1)	STRUCT	236	40
MBPBuffer [size of (MBP INFORMATION) * 10] (Refer MBP_INFORMATION in Table 32.7)	CHAR	120	276
BbTotalBuyFlag	SHORT	2	396
BbTotalSellFlag	SHORT	2	398
TotalBuyQuantity	DOUBLE	8	400
TotalSellQuantity	DOUBLE	8	408
MBO MBP INDICATOR (Refer Table 32.2 for Small Endian & Table 32.3 for Big Endian)	STRUCT	2	416
ClosingPrice	LONG	4	418
OpenPrice	LONG	4	422
HighPrice	LONG	4	426
LowPrice	LONG	4	430

Table 32.1 INTERACTIVE MBO DATA

Structure Name	INTERACTIVE MBO DATA		
Packet Length	236 bytes		
Field Name	Data Type	Size in Byte	Offset
Token	LONG	4	0
BookType	SHORT	2	4
TradingStatus	SHORT	2	6
VolumeTradedToday	LONG	4	8

Structure Name	INTERACTIVE MBO DATA		
Packet Length	236 bytes		
Field Name	Data Type	Size in Byte	Offset
LastTradedPrice	LONG	4	12
NetChangeIndicator	CHAR	1	16
Reserved	CHAR	1	17
NetPriceChangeFromClosingPrice	LONG	4	18
LastTradeQuantity	LONG	4	22
LastTradeTime	LONG	4	26
AverageTradePrice	LONG	4	30
AuctionNumber	SHORT	2	34
AuctionStatus	SHORT	2	36
InitiatorType	SHORT	2	38
InitiatorPrice	LONG	4	40
InitiatorQuantity	LONG	4	44
AuctionPrice	LONG	4	48
AuctionQuantity	LONG	4	52
MBOBuffer[sizeof (MBO INFORMATION) *10]	CHAR	180	56

Table 32.2 MBO MBP INDICATOR (For Small Endian Machines)

Structure Name	MBO MBP INDICATOR		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Reserved	BIT	4	0
Sell	BIT	1	0
Buy	BIT	1	0
LastTradeLess	BIT	1	0
LastTradeMore	BIT	1	0
Reserved	CHAR	1	1

Table 32.3 MBO MBP INDICATOR (For Big Endian Machines)

Structure Name	MBO MBP INDICATOR		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
LastTradeMore	BIT	1	0
LastTradeLess	BIT	1	0

Structure Name	MBO MBP INDICATOR		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Buy	BIT	1	0
Sell	BIT	1	0
Reserved	BIT	4	0
Reserved	CHAR	1	1

Table 32.4 MBO INFORMATION

Structure Name	MBO INFORMATION		
Packet Length	18 bytes		
Field Name	Data Type	Size in Byte	Offset
TraderId	LONG	4	0
Qty	LONG	4	4
Price	LONG	4	8
ST MBO MBP TERMS (Refer Table 32.5 & Table 32.6)	STRUCT	2	12
MinFillQty	LONG	4	14

Table 32.5 ST MBO MBP TERMS (For Small Endian Machines)

Structure Name	ST MBO MBP TERMS		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Reserved1	BIT	6	0
Aon	BIT	1	0
Mf	BIT	1	0
Reserved2	BIT	8	1

Table 32.6 ST MBO MBP TERMS (For Big Endian Machines)

Structure Name	ST MBO MBP TERMS		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Mf	BIT	1	0
Aon	BIT	1	0
Reserved1	BIT	6	0
Reserved2	BIT	8	1

Table 32.7 MBP INFORMATION

Structure Name	MBP INFORMATION		
Packet Length	12 bytes		
Field Name	Data Type	Size in Byte	Offset
Quantity	LONG	4	0
Price	LONG	4	4
NumberOfOrders	SHORT	2	8
BbBuySellFlag	SHORT	2	10

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_MBO_MBP_UPDATE (7200).
Token	This field contains the token number—a unique number given to a particular symbol-series combination.
BookType	This field contains the book type—RL / ST / SL / NT / OL/ SP / AU
TradingStatus	This field contains the trading status of the security: • '1' - Preopen • '2' - Open • '3' - Suspended • '4' - Preopen Extended
VolumeTradedToday	This field contains the total quantity of a security traded on the current day.
LastTradedPrice	This field contains the price at which the latest trade in a security has taken place.
NetChangeIndicator	This field is a flag which indicates any change of the order price from the LTP. • '+' for increase '-' for decrease
NetPriceChange	This field contains the net change between the order price and the LTP.
LastTradeQuantity	This field contains the quantity at which the last trade took place in a security.
LastTradeTime	This field contains the time when the last trade took place in a security.

Field Name	Brief Description
AverageTradePrice	This field contains the average price of all the trades in a security.
AuctionNumber	This field contains the auction number. The maximum value this can take is 9999. In other cases, it is set to zero.
AuctionStatus	Refer to Appendix.
InitiatorType	This field contains the initiator type—control or trader. Presently initiator type is control, since only the Exchange can initiate an auction. Otherwise it is blank.
InitiatorPrice	This field contains the price of the security of the initiator‘s auction order. Otherwise it is set to zero.
InitiatorQuantity	This field contains the quantity of the security of the initiator‘s auction order. Otherwise it is set to zero.
AuctionPrice	This field contains the price at which auction in a security takes place. Otherwise it is set to zero.
AuctionQuantity	This field contains the quantity at which auction in a security takes place. Otherwise it is set to zero.
RecordBuffer (MBO INFORMATION)	This field contains five best Buy orders and five best Sell orders from the order book. First five contains Buy orders and next five contains Sell orders.
RecordBuffer (MBP INFORMATION)	This field contains five best Buy prices and five best Sell prices from the order book .First five are for Buy and next five for Sell.
BbTotalBuyFlag	This field contains value ‗1‘ if there is a buy back order in the buy side else its value is zero. This is useful if the buy back order is not amongst the top five.
BbTotalSellFlag	Currently, its value is set to zero.
TotalBuyQuantity	This field contains the total quantity of buy orders in a Security.
TotalSellQuantity	This field contains the total quantity of sell orders in a security.
Indicator	This structure contains flags which can be set to indicate Buy, Sell and latest trade less than or greater than the immediately previous LTP.
ClosingPrice	This field contains the closing price of a security.
OpenPrice	This field contains the open price of a security.

Field Name	Brief Description
HighPrice	This field contains the highest trade price.
LowPrice	This field contains the lowest trade price.
MBOInformation	This field contains the quantity and price for a maximum of five best prices.
MBPInformation	This field contains the quantity, price and number of orders for a maximum of five best prices.

Only Market by Price Update

The information regarding the best buy orders and the best sell orders is given in the following format:

Table 33 BROADCAST ONLY MBP

Structure Name	BROADCAST ONLY MBP		
Packet Length	468 bytes		
Transaction Code	BCAST_ONLY_MBP (7208)		
Field Name	Data Type	Size in Byte	Offset
BCAST_HEADER (Refer Table 2)	STRUCT	40	0
NoOfRecords	SHORT	2	40
INTERACTIVE ONLY MBP DATA [2] (Refer Table 33.1)	STRUCT	426	42

Table 33.1 INTERACTIVE ONLY MBP DATA

Structure Name	INTERACTIVE ONLY MBP DATA		
Packet Length	213 bytes		
Field Name	Data Type	Size in Byte	Offset
Token	LONG	4	0
BookType	SHORT	2	4
TradingStatus	SHORT	2	6
VolumeTradedToday	LONG	4	8
LastTradedPrice	LONG	4	12
NetChangeIndicator	CHAR	1	16
NetPriceChangeFromClosingPrice	LONG	4	17
LastTradeQuantity	LONG	4	21

Structure Name	INTERACTIVE ONLY MBP DATA		
Packet Length	213 bytes		
Field Name	Data Type	Size in Byte	Offset
LastTradeTime	LONG	4	25
AverageTradePrice	LONG	4	29
AuctionNumber	SHORT	2	33
AuctionStatus	SHORT	2	35
InitiatorType	SHORT	2	37
InitiatorPrice	LONG	4	39
InitiatorQuantity	LONG	4	43
AuctionPrice	LONG	4	47
AuctionQuantity	LONG	4	51
RecordBuffer [size of (MBP INFORMATION) * 10]	CHAR	120	55
BbTotalBuyFlag	SHORT	2	175
BbTotalSellFlag	SHORT	2	177
TotalBuyQuantity	DOUBLE	8	179
TotalSellQuantity	DOUBLE	8	187
MBP INDICATOR(Refer Table 33.2 for Small Endian & Refer Table 33.3 Big Endian)	STRUCT	2	195
ClosingPrice	LONG	4	197
OpenPrice	LONG	4	201
HighPrice	LONG	4	205
LowPrice	LONG	4	209

Table 33.2 MBP INDICATOR (For Small Endian Machines)

Structure Name	MBP INDICATOR		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Reserved [4]	BIT	4	0
Sell	BIT	1	0
Buy	BIT	1	0
LastTradeLess	BIT	1	0
LastTradeMore	BIT	1	0
Reserved	CHAR	1	1

Table 33.3 MBP INDICATOR (For Big Endian Machines)

Structure Name	MBP INDICATOR		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
LastTradeMore	BIT	1	0
LastTradeLess	BIT	1	0
Buy	BIT	1	0
Sell	BIT	1	0
Reserved	BIT	4	0
Reserved	CHAR	1	1

Table 33.4 MBP INFORMATION

Structure Name	MBP INFORMATION		
Packet Length	12 bytes		
Field Name	Data Type	Size in Byte	Offset
Quantity	LONG	4	0
Price	LONG	4	4
NumberOfOrders	SHORT	2	8
BbBuySellFlag	SHORT	2	10

Field Name	Brief Description
TransactionCode	The transaction code set for the purpose is BCAST_MBO_MBP_UPDATE (7200).
NoOfRecords	This field contains the number of securities sent.
Token	This field contains the token number—a unique number given to a particular symbol-series combination.
BookType	This field contains the book type—RL / ST / SL / NT / OL/ SP / AU
TradingStatus	This field specifies trading status of the security. It contains one of the following values. • '1' - Preopen • '2' - Open • '3' - Suspended • '4' - Preopen Extended

Field Name	Brief Description
VolumeTradedToday	This field contains the total quantity of a security traded on the current day.
LastTradedPrice	This field contains the price at which the latest trade in a security has taken place.
NetChangeIndicator	This field is a flag which indicates any change of the order price from the LTP. • '+' for increase • '-' for decrease.
NetPriceChange	This field contains the net change between the order price and the LTP.
LastTradeQuantity	This field contains the quantity at which the last trade took place in a security.
LastTradeTime	This field contains the time when the last trade took place in a security.
AverageTradePrice	This field contains the average price of all the trades in a security.
AuctionNumber	This field contains the auction number. The maximum value this can take is 9999. Otherwise it is set to zero.
AuctionStatus	Refer to *Auction Status* in Appendix.
InitiatorType	This field contains the initiator type—control or trader. Presently initiator type is control, since only the Exchange can initiate an auction. Otherwise it is set to blank.
InitiatorPrice	This field contains the price of the security of the initiator's auction order. Otherwise it is set to zero.
InitiatorQuantity	This field contains the quantity of the security of the initiator's auction order. Otherwise it is set to zero.
AuctionPrice	This field contains the price at which auction in a security takes place. Otherwise it is set to zero.
AuctionQuantity	This field contains the quantity at which auction in a security takes place. Otherwise it is zero.
Record Buffer (MBP INFORMATION)	This field contains five best Buy prices and five best Sell prices from the order book. First five are for buy and next five for sell.
BbTotalbuyFlag	This field contains the value of '1' which indicates that there is a buy back order in the buy side else its value is set to zero. This is useful if the buy back order is not amongst the top five.

Field Name	Brief Description
BbTotalsellFlag	Currently, this field is set to zero.
TotalBuyQuantity	This field contains the total quantity of buy orders in a security.
TotalSellQuantity	This field contains the total quantity of sell orders in a security.
Indicator	This field contains flags which can be set to indicate Buy, Sell and Latest trade less than or greater than the immediately previous LTP.
ClosingPrice	This field contains the closing price of a security.
OpenPrice	This field contains the open price of a security.
HighPrice	This field contains the highest trade price.
LowPrice	This field contains the lowest trade price.
MBPInformation	This field contains the quantity, price and number of orders for a maximum of five best prices.

Market Watch Update

The market watch information gives the best buy order and its quantity, best sell order and its quantity and the last trade price. The structure sent for the purpose is:

Table 34 BROADCAST INQUIRY RESPONSE

Structure Name	BROADCAST INQUIRY RESPONSE		
Packet Length	452 bytes		
Transaction Code	BCAST_MW_ROUND_ROBIN (7201)		
Field Name	Data Type	Size in Byte	Offset
BCAST_HEADER (Refer table 2)	STRUCT	40	0
NoOfRecords	SHORT	2	40
MARKETWATCHBROADCAST [5] (Refer table 34.1)	STRUCT	410	42

Table 34.1 MARKETWATCHBROADCAST

Structure Name	MARKETWATCHBROADCAST		
Packet Length	82 bytes		
Field Name	Data Type	Size in Byte	Offset
Token	SHORT	4	0
MARKET WISE INFORMATION [3] (Refer Table 34.2)	STRUCT	78	4

Table 34.2 MARKET WISE INFORMATION

Structure Name	MARKET WISE INFORMATION		
Packet Length	26 bytes		
Field Name	Data Type	Size in Byte	Offset
MBO MBP INDICATOR (Refer table no. 34.3 for small endian & table no. 34.4 for big endian)	STRUCT	2	0
BuyVolume	LONG	4	2
BuyPrice	LONG	4	6
SellVolume	LONG	4	10
SellPrice	LONG	4	14
LastTradePrice	LONG	4	18
LastTradeTime	LONG	4	22

Table 34.3 MBO MBP INDICATOR (For Small Endian Machines)

Structure Name	MBO MBP INDICATOR		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
Reserved	BIT	4	0
Sell	BIT	1	0
Buy	BIT	1	0
LastTradeLess	BIT	1	0
LastTradeMore	BIT	1	0
Reserved	CHAR	1	1

Table 34.4 MBO MBP INDICATOR (For Big Endian Machines)

Structure Name	MBO MBP INDICATOR		
Packet Length	2 bytes		
Field Name	Data Type	Size	Offset
LastTradeMore	BIT	1	0
LastTradeLess	BIT	1	0
Buy	BIT	1	0
Sell	BIT	1	0
Reserved	BIT	4	0
Reserved	CHAR	1	1

Field Name	Brief Description
TransactionCode	The transaction code sent is BCAST_MW_ROUND_ROBIN (7201).
NumberOfRecords	This field contains the number of times the structure MARKET WATCH BROADCAST is repeated.
Token	This field contains the token number—a unique number given to a particular symbol-series combination.
Indicator	This structure contains the flags which can be set to indicate Buy, Sell and Last trade less than or greater than previous LTP.
BuyVolume	This field contains the quantity of the best Buy order.
BuyPrice	This field contains the price of the best Buy order.
SellVolume	This field contains the quantity of the best Sell order.
SellPrice	This field contains the price of the best Sell order.
LastTradePrice	This field contains the latest trade price of a security.
LastTradeTime	This field contains the latest trade time of a security.

Security Open Message

When the market opens the open price of the security is sent in the following structure:

Table 35 MS SEC OPEN MSGS

Structure Name	MS_SEC_OPEN_MSGS		
Packet Length	60 bytes		
Transaction Code	SECURITY_OPEN_PRICE (6013)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER	STRUCT	40	0
SEC_INFO (Refer sec_info)	SHORT	12	40
Token	LONG	4	52
OpeningPrice	LONG	4	56

Field Name	Brief Description
TransactionCode	The transaction code sent is SECURITY_OPEN_PRICE (6013).
SEC_INFO (Refer sec_info)	This structure contains the symbol and series for a particular security.
Token	This field contains a unique number that is given to a particular symbol-series combination.
OpeningPrice	This field contains open price of the security.

Broadcast Circuit Check

If there has been no data on the broadcast circuit for a stipulated time period, then a pulse is sent. This time is nine seconds now but it can be changed by NSE–Control. This is only to intimate that the circuit is still there but there is no data to send. The structure sent is:

MESSAGE HEADER (Refer to *Message Header* in Chapter 2)

Field Name	Brief Description
TransactionCode	The transaction code sent is BC_CIRCUIT_CHECK (6541).

Multiple Index Broadcast

The multiple index broadcast structure **for CM** is as follows. **There is no index related broadcast from SLBM Trading System.**

Table 36 BROADCAST INDICES

Structure Name	BROADCAST INDICES		
Transaction Code	BCAST_INDICES (7207)		
Packet Length	468 Bytes		
Field Name	Data Type	Size in Byte	Offset
BCAST_HEADER (Refer Table 2)	STRUCT	40	0
NumberOfRecords	SHORT	2	40
Indices[6] (Refer Table 36.1)	STRUCT	426	42

Table 36.1 Indices

Structure Name	INDICES		
Packet Length	71 Bytes		
Field Name	Data Type	Size in Byte	Offset
IndexName	CHAR	21	0
IndexValue	LONG	4	21
HighIndexValue	LONG	4	25
LowIndexValue	LONG	4	29
OpeningIndex	LONG	4	33
ClosingIndex	LONG	4	37
PercentChange	LONG	4	41
YearlyHigh	LONG	4	45
YearlyLow	LONG	4	49
NoOfUpmoves	LONG	4	53
NoOfDownmoves	LONG	4	57
MarketCapitalisation	DOUBLE	8	61
NetChangeIndicator	CHAR	1	69
FILLER	CHAR	1	70

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_INDICES (7207).
NoOfRecords	This field contains the number of indices currently supported by the system. Depending upon this number, there will be records filled up in subsequent Indices structure.
Indices	This field is an array of structure. The attributes of this structure are given below in this table itself.
IndexName	This field contains Name of the index. For example, Defty, Nifty
IndexValue	This field contains the online market index value at that instance of broadcast.
HighIndexValue	This field contains the day's highest index value at the time of broadcast.
LowIndexValue	This field contains day's lowest index value at the time of broadcast.
OpeningIndex	This field contains the opening index value at the time of market open.
ClosingIndex	If market is open, this field it is set to previous day's closing index.

Field Name	Brief Description
	After completion of day's batch processing, this field value shows today's close.
PercentChange	This field contains the percent change in current index with respect to yesterday's closing index.
YearlyHigh	This field contains the highest index in the year.
YearlyLow	This field contains the lowest index in the year.
NoOfupmoves	This field contains the number of time index has moved up with respect to previous index.
NoOfdownmoves	This field contains the number of time index has moved down with respect to previous index.
MarketCapitalization	This field contains the Market Capitalization of securities participating in the index.
NetChange Indicator	This field contains one of the following values. • '+' - if the current index is greater than previous index. • '-' - if the current index is less than previous index. • ' ' - if the current index is equal to previous index.

Broadcast industry index

This Packet contains the index values of Nine Indices with name. The structure is as follows:

Table 37 BROADCAST INDUSTRY INDICES

Structure Name	BROADCAST INDUSTRY INDICES		
Transaction Code	BCAST_IND_INDICES (7203)		
Packet Length	467 Bytes		
Field Name	Data Type	Size in Byte	Offset
BCAST_HEADER (Refer Table 2)	STRUCT	40	0
NumberOfRecords	SHORT	2	40
Indices[17] (Refer Table 37.1)	STRUCT	425	42

Table 37.1 INDICES

Structure Name	INDICES		
Packet Length	25 Bytes		
Field Name	Data Type	Size in Byte	Offset
IndexName[21]	CHAR	21	0
IndexValue	LONG	4	21

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_IND_INDICES (7203).
NoOfRecords	This field contains the number of indices currently supported by the system. Depending upon this number, there will be records filled up in subsequent Indices structure.
Indices	This field is an array of structure. The attributes of this structure are given below in this table itself.
IndexName	This field contains Name of the index. For example, Defty, CNX IT
IndexValue	This field contains the online market index value at that instance of broadcast.

Broadcast buy back Information

This packet will contain the buy back Information which are running on that day.This will be broadcasted for every one hour from Market open till market closes on that day.

BUYBACK is not there in SLBM system.

The structure is as follows:

Table 38 BROADCAST BUY_BACK

Structure Name	BROADCAST BUY_BACK		
Transaction Code	BCAST_BUY_BACK (7211)		
Packet Length	426 Bytes		
Field Name	Data Type	Size in Byte	Offset
BCAST_HEADER (Refer Table 2)	STRUCT	40	0
NumberOfRecords	SHORT	2	40
BuyBackData [6] (Refer Table 38.1)	STRUCT	384	42

Table 38.1 BUYBACKDATA

Structure Name	BUYBACKDATA		
Packet Length	64 Bytes		
Field Name	Data Type	Size in Byte	Offset
Token	LONG	4	0

Structure Name	BUYBACKDATA		
Packet Length	64 Bytes		
Field Name	Data Type	Size in Byte	Offset
Symbol	CHAR	10	4
Series	CHAR	2	14
PdayCumVol	DOUBLE	8	16
PdayHighPrice	LONG	4	24
PdayLowPrice	LONG	4	28
PdayWtAvg	LONG	4	32
CdayCumVol	DOUBLE	8	36
CdayHighPrice	LONG	4	44
CdayLowPrice	LONG	4	48
CdayWtAvg	LONG	4	52
StartDate	LONG	4	56
EndDate	LONG	4	60

Field Name	Brief Description
TransactionCode	The transaction code is BCAST_BUY_BACK (7211)..
NoOfRecords	This field contains the number of indices currently supported by the system. Depending upon this number, there will be records filled up in subsequent Indices structure.
BuyBackData	This field is an array of structure. The attributes of this structure are given below in this table itself.
Token	This field contains a unique number that is given to a particular symbol-series combination.
Symbol	This field contains the symbol of the security.
Series	This field contains the series of the security.
PDayCumVolume	This field contains previous day cummulative Volume
PDayHighPrice	This field contains Previous day's High Price
PDayLowPrice	This field contains Previous day's Low Price
PDayWeightAvg	This field contains Previous day's Weighted Average Price
CDayCummulativeVolume	This field contains current day's cummulative Volume
CDayHighPrice	This field contains current day's High Price
CDayLowPrice	This field contains current day's Low Price
CDayWeightAvg	This field contains current day's Weighted Average Price
StartDate	This field contains Start Date of Buy back period
EndDate	This field contains End Date of Buy back period

Chapter 8 Encryption and Decryption of Interactive Messages

Background

NSE provides a pan-India trading platform to its trading members. Members connect to this platform using client-server architecture. Connections are made using TCP/IP protocol and messages are exchanged using NSE's own messaging format (also known as NNF format). Messages exchanged are binary in nature. Currently these messages are not encrypted, exchange now proposes to encrypt them. This section of document provides an overview of the implementation approach that exchange has finalized, for doing the same.

Overview

Interactive messages which are exchanged between member applications and the exchange today use the NNF protocol published by exchange. As for every trading platform, similarly in this case as well availability, reliability and speed are the key considerations in the protocol. In order to enhance the security posture, it is now proposed to encrypt these messages on an end-to-end basis. While encryption of messages within member environment towards their clients will need to be done by respective members. For the communication that happens between member applications and exchange, a few changes into NNF protocol are being proposed. Changes have been envisaged considering the following attributes.

(i) Secure communication

(ii) Availability

(iii) Reliability

(iv) Speed

Minimal changes in member applications

Proposed Methodology

Exchange proposes a combination of TLS 1.3 security protocol and AES-256 bits-based symmetric encryption approach. Following is an overview.

1st Step: Member applications will connect initially to Exchange Gateway Router server using TCP with TLS 1.3 security protocol and will receive unique session key from the Exchange through the secured connection.

2nd Step: Member applications will then connect to allocated Exchange Gateway server through TCP, and each and every message will be encrypted/decrypted using the same session key (symmetric cryptography AES 256 bits GCM mode) at both member end and Exchange end.

Below are the details of the methodology

(i) Exchange will generate self-signed CA certificates on periodic basis. CA certificate will remain common for all members and shall be distributed as and when generated via extranet.

(ii) On a daily basis when member applications need to connect to trading platform they will need to do the following

a. Member applications will connect to Exchange Gateway Router server on TCP using TLS 1.3 security protocol. As part of TLS 1.3 security protocol, it is recommended that member applications verify Gateway Router server authenticity using the CA certificate provided by the Exchange.

b. GR request and GR response messages will be sent and received by member applications using TLS 1.3 security protocol.

c. A unique 32-byte session key will be provided to member applications as part of GR response message.

(iii) Post successful communication with Gateway router server, member applications will establish a new TCP connection with the allocated gateway server of Exchange. The first message after connecting through TCP will be a non-encrypted special registration message (SECURE_BOX_REGISTRATION_REQUEST) to indicate that member application is using encryption. All the messages, after the first message, that are exchanged on this connection from both sides (member applications and

Exchange) will be encrypted and decrypted using the 32-byte session key that was provided from Exchange at the time of Gateway Router handshake. GCM mode of symmetric cryptography AES 256 bits will be used by member applications and Exchange.

(iv) In case of new login or disconnection and then re login, the above-mentioned steps will be repeated

We envisage minimal changes in member applications. Sample function calls which could be considered for encryption-decryption for the above proposed approaches are provided in annexure for Encryption/Decryption.

Disconnection on MD5 Checksum failure

(i) If member is connected on encrypted channel and MD5 checksum fails then a box sign off message with error code (19031) will be sent to member before disconnection.

(ii) If member is connected on non-encrypted channel and MD5 checksum fails then there will be no change in the behavior. The packet will be dropped by Trading system and continue reading the next packet.

Chapter 9 Direct Interface to Exchange Trading System

This chapter describes how member systems can directly connect to NSE for trading, while using existing formats of business messages from NNF API documents.

To directly connect to NSE for trading, member systems will have carry out the changes specified herein.

Message Formats

Change to packet format

Length (2 bytes)	Sequence number (4 bytes)	Checksum(MD5) for Message data (16 bytes)	Message Data (Variable length)

- Max length will be the predefined value of 1024 bytes.
  Length = size of length field (2 bytes) +
  size of sequence number field (4 bytes) +
  size of the checksum field (16 bytes) +
  size of Message data (variable number of bytes as per the transcode)
- For members connecting on encrypted mode, the sequence number received in the request message for Order related interactive messages will be echoed back in the sequence number field of corresponding response messages. It is recommended to send an incremental sequence number.
- For members connecting on non-encrypted mode, there is no change in sequence number. Sequence number will be sent as 0 in all the packets.
- Message data will be of variable length
- The checksum algorithm used will be MD5. Checksum is applied only on the Message data field and not on the entire packet.
- For more details on MD5 refer: RFC 1321 (rfc1321) - The MD5 Message-Digest Algorithm ()

Change to structure for 'MESSAGE_HEADER'

MESSAGE_HEADER

Structure Name	MESSAGE_HEADER		
Packet Length	40 bytes		
Field Name	Data Type	Size in Byte	Offset
Transaction Code	SHORT	2	0
LogTime	LONG	4	2
AlphaChar	CHAR	2	6
User Id	LONG	4	8
ErrorCode	SHORT	2	12
Timestamp	LONG LONG	8	14
TimeStamp1	CHAR	8	22
TimeStamp2	CHAR	8	30
MessageLength	SHORT	2	38

Note: Member systems must populate relevant User ID field in the header.

Connecting to NSE for Trading

Sequence to be followed by the member for login

1. Member to connect (TCP/IP, SSL connection)to the IP and port provided by the exchange and send the GR_REQUEST using OpenSSL (Version 1.1.1) library calls with TLS versions 1.3 (TLS1_3_VERSION). Refer annexure for Encryption/Decryption .
2. Exchange will send the GR_RESPONSE to the member containing the IP address, Port and the Session key and cryptographic key & cryptographic IV (Initialization Vector) on SSL connection. If there is any error then ErrorCode field in MESSAGE_HEADER will be populated with relevant error code in the GR_RESPONSE.
3. Member applications will then make a new TCP connection with the allocated Gateway server (IP and port provided in the GR_RESPONSE) and send SECURE_BOX_REGISTRATION_REQUEST. BoxID (received in GR_RESPONSE) is to be populated in SECURE_BOX_REGISTRATION_REQUEST
4. Exchange will send the SECURE_BOX_REGISTRATION_RESPONSE. If there is any error, then ErrorCode field in MESSAGE_HEADER will be populated with relevant error code in the SECURE_BOX_REGISTRATION_RESPONSE and the Box connection will be terminated.

5. If there is no error in SECURE_BOX_REGISTRATION_RESPONSE, member should do encryption and decryption initialization to create encryption and decryption contexts (Please refer annexure). This initialization should be done only once. Once initialized, all further messages between member application and allocated Gateway server will be encrypted and decrypted using same encryption and decryption contexts respectively. Further member should send the BOX_SIGN_ON_REQUEST_IN. BoxID, BrokerID and Session key (received in GR_RESPONSE) is to be populated in BOX_SIGN_ON_REQUEST_IN. MD5 Algorithm to be performed on plain messages. That means, while sending the messages to Trading system, MD5 is to be performed first and then encryption. Encrypted message length + 22 (sizeof(Header)) will have to be written in first 2 bytes of header, Sequence Number in next 4 bytes and MD5 value (of plain message) will be written in last 16 bytes of Header and the header will have to be prepended to the encrypted message. This message will be sent out to Trading System. While receiving the messages from Trading System, decryption should be done first and then MD5 is to be applied on decrypted buffer. Decryption should be done on message excluding first 22 bytes of header.
6. Exchange will send the BOX_SIGN_ON_REQUEST_OUT. If there is any error then ErrorCode field in MESSAGE_HEADER will be populated with relevant error code in the BOX_SIGN_ON_REQUEST_OUT and the Box connection will be terminated.

   Note: Multiple BOX_SIGN_ON_REQUEST_IN requests on a successfully established box connection will lead to the existing box connection termination.
7. Once a connection for a particular Box ID is established, all users linked with this Box ID can login using the SIGNON_IN structure. Refer Chapter 3 for login request and response using SIGNON_IN structure.
8. For further flow refer to existing protocol defined in Chapter 3 of Protocol Document

Gateway Router Request

MS_GR_REQUEST

Structure Name	MS_GR_REQUEST		
Packet Length	48 bytes		
Transaction Code	GR_REQUEST (2400)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER(Refer Msg_hdr of ch 9)	STRUCT	40	0
Box ID	SHORT	2	40
BrokerID	CHAR	5	42
Filler	CHAR	1	47

Field Name	Brief Description
Transaction Code	This field is the part of Message Header.The transaction code is 2400.
Box ID	Exchange provided Box ID to be used for this connection
BrokerID	This field should contain the trading member ID

Gateway Router Response

MS_GR_RESPONSE

Structure Name	MS_GR_RESPONSE		
Packet Length	76 bytes		
Transaction Code	GR_RESPONSE(2401)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER(Refer Msg_hdr of ch 9)	STRUCT	40	0
Box ID	SHORT	2	40
BrokerID	CHAR	5	42
Filler	CHAR	1	47
IP Address	CHAR	16	48
Port	LONG	4	64
Session Key	CHAR	8	68
Cryptographic Key	CHAR	32	76
Cryptographic IV(Initialization Vector)	CHAR	16	108

Field Name	Brief Description
Transaction Code	This field is the part of Message Header.The transaction code is 2401
Error Code	This field is the part of Message Header.Error Code will be set if the query

Field Name	Brief Description
	is unsuccessful. Refer to *List of Error Codes* in Appendix.
Box ID	Exchange provided Box ID used for this connection
BrokerID	This field should contain the trading member ID
IP Address	IP address assigned by exchange
Port	Port Number given by exchange
Session Key	Session key to be used for authentication
Cryptographic Key	Cryptographic key for both the encryption and decryption of all messages between member application and allocated Gateway Server.
Cryptographic IV (Initialization Vector)	Cryptographic IV (Initialization Vector) for both the encryption and decryption of all messages between member application and allocayed Gateway Server.

Secure Box Registration Request

SECURE_BOX_REGISTRATION_REQUEST

Structure Name	MS_ SECURE_BOX_REGISTRATION_REQUEST_IN		
Packet Length	42 bytes		
Transaction Code	SECURE_BOX_REGISTRATION_REQUEST_IN (23008)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER (Refer Message Header structure)	STRUCT	40	0
BoxId	SHORT	2	40

Field Name	Brief Description
Transcode	This field is the part of Message Header. The transaction code is 23008
BoxId	Exchange provided Box ID to be used for this connection

Secure Box Registration Response

Structure Name	MS_ SECURE_BOX_REGISTRATION_RESPONSE_OUT		
Packet Length	40 bytes		
Transaction Code	SECURE_BOX_REGISTRATION_REQUEST_IN (23009)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER (Refer Message Header structure)	STRUCT	40	0

Field Name	Brief Description
Transcode	This field is the part of Message Header. The transaction code is 23009
ErrorCode	This field is the part of Message Header. Error Code will be set if the query is unsuccessful. *Refer to List of Error Codes in Appendix*

Box Sign on Request

MS_BOX_SIGN_ON_REQUEST_IN

Structure Name	MS_BOX_SIGN_ON_REQUEST_IN		
Packet Length	60 bytes		
Transaction Code	BOX_SIGN_ON_REQUEST_IN(23000)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER(Refer Msg_hdr of ch 9)	STRUCT	40	0
BoxId	SHORT	2	40
BrokerID	CHAR	5	42
Reserved	CHAR	5	47
SessionKey	CHAR	8	52

Field Name	Brief Description
Transcode	This field is the part of Message Header.The transaction code is 23000
BoxId	Exchange provided Box ID to be used for this connection
BrokerID	This field should contain the trading member ID
SessionKey	Session key received in GR_RESPONSE(2401)

Box Sign on Response

MS_BOX_SIGN_ON_REQUEST_OUT

Structure Name	MS_BOX_SIGN_ON_REQUEST_OUT		
Packet Length	52 bytes		
Transaction Code	BOX_SIGN_ON_REQUEST_OUT(23001)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER(Refer Msg_hdr of ch 9)	STRUCT	40	0
BoxId	SHORT	2	40
Reserved	CHAR	10	42

Field Name	Brief Description
Transaction Code	This field is the part of Message Header.The transaction code is 23001
Error Code	This field is the part of Message Header.Error Code will be set if the query is unsuccessful. Refer to *List of Error Codes* in Appendix.
BoxId	Exchange provided Box ID used for this connection

User Log on Request

All the fields of Logon message (transaction code 2300) to be populated as prescribed in Chapter 3 of protocol document except for MESSAGE HEADER field which should be refered from this chapter.

If authentication information is correct, member systems will receive a successful SIGN_ON response.

How to Logoff?

To logoff from the exchange trading system, there is no change and use the existing protocol defined in Chapter 3 of protocol document.

Heartbeat exchange

Member systems must exchange heartbeat signals at BOX connection level with exchange trading system during periods of inactivity. Trading Host will consider the member system as inactive after missing two heartbeats in succession, and disconnect the socket connection. Heartbeats will carry following data in MessageData segment of the message. Heartbeat is to be sent only if there is inactivity for 30 seconds. The format is MESSAGE_HEADER with following detail.

HEARTBEAT

Structure Name	HEARTBEAT		
Packet Length	40 bytes		
Transaction Code	23506		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER(Refer Msg_hdr of ch 9)	STRUCT	40	0

Field Name	Description
TransactionCode	The transaction code is (23506).

Recovering from disconnections

If member system detects a loss of TCP connection with the exchange trading system, please perform the same operations for starting a fresh login given above.

Performing Trading activities

Once authenticated connection is successfully established, member systems can send any business message to exchange as described in NNF protocol documents. Care should be taken to use MESSAGE_HEADER described in this chapter wherever applicable in front of business messages.

Connection Termination

When connection is terminated by exchange, BOX_SIGN_OFF (20322) message with appropriate error code will be sent.

Box Sign Off

MS_BOX_SIGN_OFF

Structure Name	MS_BOX_SIGN_OFF		
Packet Length	42 bytes		
Transction code	BOX_SIGN_OFF (20322)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER(Refer Msg_hdr of ch 9)	STRUCT	40	0
BoxId	SHORT	2	40

Field Name	Brief Description
Transaction Code	This field is the part of Message Header.The transaction code is 20322.
Error Code	This field is the part of Message Header.Error Code will be set if the query is unsuccessful. Refer to *List of Error Codes* in Appendix.
BoxId	Exchange provided Box ID used for this connection

Chapter 10 Exception Handling

Introduction

NSE's trading system constitutes of multiple matching engines (streams). Each stream hosts a range of contracts on which trading is allowed. In case of an exception single/multiple streams will get impacted. It is necessary that relevant information is disseminated in such events so that necessary action can be taken at member's end to bring their systems into a consistent state.

Exception handling:

1. At the start of the outage message will be sent on broadcast channel with StreamNumber and status as 1 (start of outage) and members may get disconnected from the exchange (Member can also receive this message through journal download).
2. On receiving message in step 1, members should clear outstanding orders at their end for the respective streams. Exchange would also cancel all the outstanding orders and no cancellation messages will be sent for these orders.
3. Once exchange has restored the stream, message will be sent on broadcast channel with StreamNumber and status as 0 (end of outage) (Member can also receive this message through journal download).
4. On receiving the message in step 3, Members can reconnect to the exchange in case they have got disconnected in step 1.

Message structure

Message structure is as follows:

Structure Name	MS_BCAST_CONT_MESSAGE		
Packet Length	244 bytes		
Transction code	BCAST_CONTI_MSG (5294)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE_HEADER (Refer Table 1)	STRUCT	40	0
StreamNumber	SHORT	2	40
Status	SHORT	2	42

Structure Name	MS_BCAST_CONT_MESSAGE		
Packet Length	244 bytes		
Transction code	BCAST_CONTI_MSG (5294)		
Field Name	Data Type	Size in Byte	Offset
Reserved	CHAR	200	44

The following table provides details of the various fields present in above Message structure.

Field Name	Brief Description
Transaction Code	This field is the part of Message Header.The transaction code is 5294.
StreamNumber	0 – All streams are impacted or impacted stream number (eg 1, 2, 3, 4...)
Status	1 – Start of outage 0 – End of outage
Reserved	Reserved for future use

DR 45 Initiative

NSE trading system provides high availability of its services by having identical setup available at NSE DR Site.

Please find below list of point to be considered in case of switchover to DR site

1. Members will have to reconnect to trading system, as they will be disconnected once the primary site is unavailable
2. Member should continue to use existing connectivity parameter for connecting to NSE trading system at DR site
3. Member on reconnecting at DR site will receive start of outage message as a part of journal download.

   The message sent in the following format

   (MS_BCAST_CONT_MESSAGE) (refer to Exception handling)
4. Exchange shall not carry forward outstanding orders from primary site to DR site and no cancellation messages will be sent for these orders. Accordingly members are advised to clear outstanding orders at their end.

5. Exchange shall publish streamwise trade number of the last trade (Exchange trade number) available at DR site. Member may note that streamwise trades upto the last trade number shall only be considered.
6. Exchange shall broadcast streamwise last trade number.
   The message sent in the following format
   (MS_TRADER_INT_MSG) (refer to Interactive/broadcast messages sent from control)
7. Member shall be able to perform trade modification or trade cancellation on trades which are available at DR site.
8. In case member is connected after switchover, they will receive end of outage message. The message sent in the following format
   (MS_BCAST_CONT_MESSAGE) (refer to Exception handling)
   In case member is not connected, they will receive this message as a part of journal download post reconnecting to NSE trading system at DR site.
   The message sent in the following format
   (MS_BCAST_CONT_MESSAGE) (refer to Exception handling)
9. Journal download information before switchover shall not be available ,
10. Used limit value in User Order Value Limit (UOVL) and Branch Order Value Limit (BOVL) will be reset to zero after switchover to DR site.

Appendix

List of Error Codes

Error Code ID	Error Code Value	Description of Error Numbers
ERR_MARKET_NOT_OPEN	16000	The trading system is not available for trading.
ERR_INVALID_BRANCH ERR_BAD_TRANSACTION_CODE	16001 16003	Erroneous transaction code received.
ERR_USER_ALREADY_SIGNED_ON	16004	User already signed on.
ERR_INVALID_SIGNON	16006	Invalid Box/User sign-on, Please try again.
ERR_SIGNON_NOT_POSSIBLE	16007	Signing on to the trading system is restricted. Please try later on.
ERR_INVALID_SYMBOL	16012	Invalid symbol/series.
NOT_YOUR_FILL	16015	Invalid trade cancel request.
ERR_SECURITY_NOT_AVAILABLE	16035	Security is unavailable for trading at this time. Please try later.
ERR_INVALID_BROKER_OR_BRANCH	16041	Trading Member does not exist in the system.
ERR_USER_NOT_FOUND	16042	Dealer does not exist in the system.
ERR_PASSWORD_HAS_EXPIRED	16053	Your password has expired, must be changed.
ERR_INVALID_BRANCH	16054	Branch does not exist in the system.
ERR_PROGRAM_ERROR	16056	Program error.
ORDER_NOT_FOUND	16060	Modified/Cancelled order not found
ERR_INVALID_BUYER_USER_ID	16098	Invalid trader ID for buyer.

Error Code ID	Error Code Value	Description of Error Numbers
ERR_INVALID_SELLER_USER_ID	16099	Invalid trader ID for seller.
ERR_INVALID_SYSTEM_VERSION	16100	Your system version has not been updated.
ERR_SYSTEM_ERROR	16104	System could not complete your transaction - ADMIN notified.
ERR_CANT_COMPLETE_YOUR_REQUEST	16123	System not able to complete your request. Please try again.
ERR_USER_IS_DISABLED	16134	This Dealer is disabled. Please call the Exchange.
ERR_INVALID_USER_ID	16148	Invalid Dealer ID entered.
ERR_INVALID_TRADER_ID	16154	Invalid Trader ID entered.
ERR_ATO_IN_OPEN	16169	Order priced ATO cannot be entered when a security is open.
ORD_NOT_ALLOWED_IN_PREOPEN	16197	Order Entry or Modification not allowed in preopen.
INVALID_PRICE	16247	Invalid price in the price field.
ERR_TRADE_MOD_DIFF_VOL	16251	Trade modification with different quantities is received.
CXLD_TRADE_MOD_REQUEST	16252	Cancelled the trade modify request.
ERR_NOT_FOUND	16273	Does not exist.
ERR_MARKETS_CLOSED	16278	The markets have not been opened for trading.
ERR_SECURITY_NOT_ADMITTED	16279	The security has not yet been admitted for trading.
ERR_SECURITY_MATURED	16280	The security has matured.
ERR_SECURITY_EXPELLED	16281	The security has been expelled.

Error Code ID	Error Code Value	Description of Error Numbers
ERR_QUANTITY_EXCEEDS_ISSUED_CAPITAL	16282	The order quantity is greater than the issued capital.
ERR_PRICE_NOT_MULT_TICK_SIZE	16283	The order price is not multiple of the tick size.
ERR_PRICE_EXCEEDS_DAY_MIN_MAX	16284	The order price is out of the day's price range.
ERR_BROKER_NOT_ACTIVE	16285	The broker is not active.
OE_AUCTION_PENDING	16303	Request denied. Pending auctions.
ERR_ QUANTITY_FREEZE_CANCELLED	16307	The order is canceled due to quantity freeze.
ERR_PRICE_FREEZE_CANCELLED	16308	The order is canceled due to price freeze.
AON_VOLUME_NOT_ENOUGH	16310	AON volume not enough
ERR_SOLICITOR_PERIOD_OVER	16311	The Solicitor period for the Auction is over.
ERR_COMPETITIOR_PERIOD_OVER	16312	The Competitor period for the Auction is over.
OE_AUC_PERIOD_GREATER	16313	The Auction period will cross Market Close time.
OE_AUC_NOT_CAN	16314	The Auction cannot be cancelled.
ERR_LIMIT_WORSE_TRIGGER	16315	The limit price is worse than the trigger price.
ERR_TRG_PRICE_NOT_MULT_TICK_SIZE	16316	The trigger price is not a multiple of tick size.
ERR_NO_AON_IN_LIMITS	16317	AON attribute not allowed.
ERR_NO_MF_IN_LIMITS	16318	MF attribute not allowed.
ERR_NO_AON_IN_SECURITY	16319	AON attribute not allowed at security level.
ERR_NO_MF_IN_SECURITY	16320	MF attribute not allowed at

Error Code ID	Error Code Value	Description of Error Numbers
		security level.
ERR_MF_EXCEEDS_DQ	16321	MF quantity is greater than Disclosed quantity.
ERR_MF_NOT_MULT_BOARD_LOT	16322	MF quantity is not a multiple of regular lot.
ERR_MF_EXCEEDS_ORIGINAL_ QUANTITY	16323	MF quantity is greater than Original quantity.
ERR_DQ_EXCEEDS_ORIGINAL_ QUANTITY	16324	Disclosed quantity is greater than Original quantity.
ERR_DQ_NOT_MULT_BOARD_LOT	16325	Disclosed quantity is not a multiple of regular lot.
ERR_GTD_EXCEEDS_LIMIT	16326	GTD is greater than that specified at System.
OE_QUANTITY_GERATER_RL	16327	Quantity is greater than Regular lot size.
ERR_QUANTITY_NOT_MULT_BOARD_LOT	16328	Quantity is not a multiple of regular lot.
ERR_BROKER_NOT_PERMITTED_IN_MKT	16329	Trading Member not permitted in the market.
ERR_SECURITY_IS_SUSPENDED	16330	Security is suspended.
CXL_REMAIN_ACTIVE_ORDER	16332	Remaining passive order has to be cancelled.
ERR_BRANCH_LIMIT_EXCEEDED	16333	Branch Order Value Limit is exceeded.
OE_ORD_CAN_CHANGED	16343	The order to be cancelled has changed.
OE_ORD_CANNOT_CANCEL	16344	The order cannot be cancelled.
OE_INIT_ORD_CANCEL	16345	Initiator order cannot be cancelled.
OE_ORD_CANNOT_MODIFY	16346	Order cannot be modified.

Error Code ID	Error Code Value	Description of Error Numbers
ERR_TRADING_NOT_ALLOWED	16348	Trading is not allowed in this market.
OE_NT_REJECTED	16357	Order entered for negotiated trade is cancelled.
CHG_ST_EXISTS	16363	New status requested should not be same as existing one.
OE_SECURITY_IN_PREOPEN	16369	The security status is preopen.
ERR_USER_TYPE_INQUIRY	16372	Order entry not allowed for user as it is of inquiry type.
ERR_SOLICITION_NOT_ALLOWED	16379	The broker is not allowed to enter soliciting orders.
ERR_AUCTION_FINISHED	16383	Trading in this auction is finished.
ERR_NO_TRADING_IN_SECURITY	16387	Security is not allowed to trade in this market.
ERR_TURNOVER_LIMIT_NOT_SET	16392	Turnover limit not provided. Please contact Exchange.
CANNOT_CANCEL_NEGOTIATED_TRADES	16395	The Negotiated trades cannot be cancelled
ERR_DQ_EXCEEDS_LIMIT	16400	DQ has exceeded limit set in control.
ERR_WRONG_LOGIN_ADDRESS	16403	You are trying to sign on from a different location. Sign on is not allowed.
ERR_ADMIN_SUSP_CANCELLED	16404	Order is cancelled due to freeze admin suspension.
RECALL REJECT DUE TO INSUFFICIENT POSITION	16407	Recall order rejected due to insufficient positions.
REPAY REJECT DUE TO INSUFFICIENT POSITION	16408	Repay order rejected due to insufficient positions.

Error Code ID	Error Code Value	Description of Error Numbers
ERR_INVALID_PRO_CLIENT	16411	Pro-client can be either Pro or Client only.
ERR_INVALID_NEW_VOLUME	16412	New volume should be less than the traded volume.
ERR_INVALID_BUY_SELL	16413	Requested by can be BUY or SELL or BOTH.
ERR_INVALID_INST	16414	Invalid combination of book type and instructions (order_type).
ERR_INVALID_ORDER_PARAM	16415	Invalid combination of MF / AON / Disclosed Volume.
ERR_INVALID_CP_ID	16416	Invalid counter broker Id.
ERR_NNF_REQ_EXCEEDED	16417	Number of NNF requests exceeded.
NNF_REQUESTS_EXCEEDED	16417	Number of NNF requests within a given time has exceeded.
ERR_INVALID_ORDER	16418	Order entered has invalid data.
ERR _CXLED_TRADE_CXL_REQ	16419	Cancelled trade cancel request.
ERR_INVALID_ALPHA_CHAR	16420	Alpha char must be the same as first two chars of symbol.
ERR_TRADER_CANT_INIT_AUCTION	16421	Only control can initiate auctions, not trader.
ERR_INVALID_BOOK_TYPE	16422	Book type should be between 1(RL) and 7(AU).
ERR_INVALID_TRIGGER_PRICE	16423	Invalid trigger price entered.
ERR_INVALID_MSG_LENGTH	16424	Message length is invalid.
ERR_INVALID_PARTICIPANT	16425	Participant does not exist.

Error Code ID	Error Code Value	Description of Error Numbers
ERR_PARTICIPANT_AND_VOLUME_ CHANGED	16426	Participant and volume cannot be changed simultaneously.
INVALID_AUCTION_INQUIRY	16430	Invalid auction inquiry request.
INVALID_ACCOUNT	16431	Invalid Account in the Account field
ORDER_VALUE_LIMIT_EXCEEDED	16436	The order value limit has exceeded
DQ_NOT_ALLOWED_IN_PREOPEN	16439	DQ Orders are not allowed in preopen.
SERIES_NOT_ALLOWED_IN_PREOPEN	16440	Order Entry is not allowed in preopen for the series.
ST_NOT_ALLOWED_IN_PREOPEN	16441	ST Orders are not allowed in preopen.
ORDER_VALUE_EXCEEDS_ORDER_VALUE_LIMIT	16442	The current placed order's value is more than users order value limit
ACCOUNT_MANDATORY	16450	Account number is mandatory in Account field
OE_BL_MKT_ORDERS_IN_CLOSING	16473	Only board lot market orders are allowed
ORDER_CANCELED_DUE_TO_SECURITY_ SUSPENSION	16482	The order has been cancelled as security has been suspended
ORDER_CANCELED_DUE_TO_PARTICIPANT_ SUSPENSION	16483	The order has been cancelled as participant has been suspended
BUY_ORDER_VALUE_LIMIT_EXCEEDED	16530	Users buy order value limit has exceeded.
SELL_ORDER_VALUE_LIMIT_ EXCEEDED	16531	The order value limit for the sell quantity has exceeded its limit

Error Code ID	Error Code Value	Description of Error Numbers
NO_BUY_BACK_RUNNING	16534	No buyback running for that sercurity.
PARTIAL_ORDER_REJECTED	16535	Order partially rejected. Remaining order quantity specified rejected due to system error.
QUICK_CXL_REJECTED	16536	Quick Cancel request rejected due to system error. Retry Quick Cancel Request
ERR_FRZ_REJECT_FOR_CLOSEOUT	16567	This error code will be returned when a Recall/Repay order goes into freeze (i.e. order price is out of operating range).
ERR_CLOSEOUT_NOT_ALLOWED	16568	This error code is returned when a Close out order entry is not allowed.
ERR_CLOSEOUT_ORDER_REJECT	16569	This error code is returned when a Close out order is rejected by the system.
ERR_CLOSEOUT_TRDMOD_REJECT	16571	This error code will be returned when a user under a broker in 'Close out' state tries to modifyl Trade.
INVALID_MSG_LENGTH	16573	Message length is invalid.
RECALL_ORDER_REJECTED	17015	This error code will be returned when recall orders are not allowed for the security
REPAY ORDER REJECTED	17016	This error code will be returned when repay orders are not allowed for the security
RECALL AND REPAY ORDERS NOT ALLOWED	17017	Recall and repay orders are not allowed in SLBM.

Error Code ID	Error Code Value	Description of Error Numbers
RECALL ORDER REJECTED BY THE SYSTEM	17019	Recall order rejected by system.
REPAY ORDER REJECTED BY THE SYSTEM	17020	Repay order rejected by system.
ROLLOVER ORDER REJECTED DUE TO INSUFFICIENT POSITION.	17022	Rollover order rejected due to insufficient Position.
ROLLOVER ORDER REJECTED BY SYSTEM.	17023	Rollover order rejected by system.
The Account is Debarred by Exchange.	17024	The account is disabled from trading as per directions of SEBI/Statutory Authority.
ERR_ACCNT_DISABLE_TRADING_FOR_PIT	17025	Account is disabled for trading in the scrip during the Trading Window closure period (SEBI PIT Reg). Please contact the company for more details.
ERR_ORD_COULD_RESULT_IN_SELF_TRADE	17080	The order could have resulted in self trade
ERROR_INVALID_PACKET	17101	The packet has invalid transaction code
ERR_HEARTBEAT_NOT_RECEIVED	17102	Heart Beat not received
ERR_INVALID_BOX_ID	17104	Invalid Box Id
ERR_SEQ_NUM_MISMATCH	17105	Sequence number mismatch found
ERROR_NET_BOX_RATE_EXCEEDED	17106	Box Rate has been exceeded by the Member
ERR_MAX_USR_LOGIN_EXCEEDED	17142	Maximum user login allowed per box has been excceded
ERR_USER_HAVING_NULL_RIGHTS	17184	Order rejected as user has NO trading rights
ERR_ENCRYPTION_FLAG_MISMATCH	19030	Encryption Flag Mismatch
ERR_MD5_CHECKSUM_FAILURE	19031	MD5 Checksum Failed

Reason Codes

The reason codes and the corresponding values are given below.

Reason Code	Value
Security	5
Broker	6
Branch	7
User	8
Participant	9
Counter Party	10
Order Number	11
Auction Number	15
Price Freeze	17
Quantity Freeze	18

List of Transaction Codes

Transaction Code	Code	Structure	Size	I/B*
SYSTEM_INFORMATION_IN	1600	MESSAGE_HEADER	40	I
SYSTEM_INFORMATION_OUT	1601	SYSTEM_INFORMATION_DATA	90	I
MARKET_STATS_REPORT_DATA	1833	REPORT_MARKET_STATISTICS REPORT_TRAILER REPORT_HEADER	478 47 106	B
BOARD_LOT_IN	2000	ORDER_ENTRY_REQUEST	214	I
BOARD_LOT_OUT	2001	ORDER_ENTRY_REQUEST	214	I
NEG_ORDER_TO_BL	2008	ORDER_ENTRY_REQUEST	214	I
NEG_ORDER_BY_CPID	2009	ORDER_ENTRY_REQUEST	214	B
ORDER_MOD_IN	2040	ORDER_ENTRY_REQUEST	214	I
ORDER_MOD_OUT	2041	ORDER_ENTRY_REQUEST	214	I
ORDER_MOD_REJECT	2042	ORDER_ENTRY_REQUEST	214	I

Transaction Code	Code	Structure	Size	I/B*
ORDER_CANCEL_IN	2070	ORDER_ENTRY_REQUEST	214	I
ORDER_CANCEL_OUT	2071	ORDER_ENTRY_REQUEST	214	I
ORDER_CANCEL_REJECT	2072	ORDER_ENTRY_REQUEST	214	I
ORDER_CONFIRMATION	2073	ORDER_ENTRY_REQUEST	214	I
PRICE_CONFIRMATION	2012	ORDER_ENTRY_REQUEST	214	I
ORDER_MOD_CONFIRMATION	2074	ORDER_ENTRY_REQUEST	214	I
ORDER_CANCEL_CONFIRMATION	2075	ORDER_ENTRY_REQUEST	214	I
CANCEL_NEG_ORDER	2076	ORDER_ENTRY_REQUEST	214	I
FREEZE_TO_CONTROL	2170	ORDER_ENTRY_REQUEST	214	I
ON_STOP_NOTIFICATION	2212	TRADE_CONFIRM	152	I
TRADE_CONFIRMATION	2222	TRADE_CONFIRM	152	I
TRADE_ERROR	2223	TRADE_INQUIRY_DATA	110	I
ORDER_ERROR	2231	ORDER_ENTRY_REQUEST	214	I
TRADE_CANCEL_CONFIRM	2282	TRADE_CONFIRM	152	I
TRADE_CANCEL_REJECT	2286	TRADE_CONFIRM	152	I
TRADE_MODIFY_CONFIRM	2287	TRADE_CONFIRM	152	I
TRADE_MODIFY_REJECT	2288	TRADE_CONFIRM	152	I
SIGN_ON_REQUEST_IN	2300	SIGNON_IN	212	I
SIGN_ON_REQUEST_OUT	2301	SIGNON OUT ERROR_RESPONSE	212 180	I
SIGN_OFF_REQUEST_OUT	2321	SIGNOFF OUT	40	I
GR_REQUEST	2400	MS_GR_REQUEST	48	I
GR_RESPONSE	2401	MS_GR_RESPONSE	124	I
BCAST_CONT_MSG	5294	MS_BCAST_CONT_MESSAGE	244	B
CTRL_MSG_TO_TRADER	5295	MS_TRADER_INT_MSG	289	B
TRADE_CANCEL_IN	5440	TRADE_INQUIRY_DATA	110	I
TRADE_CANCEL_OUT	5441	TRADE_INQUIRY_DATA	110	I
TRADE_MOD_IN	5445	TRADE_INQUIRY_DATA	110	I

Transaction Code	Code	Structure	Size	I/B*
TRADE_MOD_OUT	5446	TRADE_INQUIRY_DATA	110	I
SECURITY_OPEN_PRICE	6013	SECURITY_OPEN_MESSAGES	60	B
BCAST_JRNL_VCT_MSG	6501	BROADCAST_MESSAGES	297	B
BC_OPEN_MESSAGE	6511	BCAST_VCT_MESSAGES	297	B
BC_CLOSE_MESSAGE	6521	BCAST_VCT_MESSAGES	297	B
BC_PREOPEN_SHUTDOWN_MSG	6531	BCAST_VCT_MESSAGES	297	B
BC_CIRCUIT_CHECK	6541	MESSAGE_HEADER	40	B
BC_NORMAL_MKT_PREOPEN_ENDED	6571	BCAST_VCT_MESSAGES	297	B
BC_AUCTION_STATUS_CHANGE	6581	AUCTION_STATUS_CHANGE	301	B
BCAST_AUCTION_INQUIRY_OUT	6582	AUCTION_BROADCAST_DATA	78	B
DOWNLOAD_REQUEST	7000	MESSAGE_DOWNLOAD	48	I
HEADER_RECORD	7011	MESSAGE_HEADER	40	I
MESSAGE_RECORD	7021	MESSAGE_HEADER	40	I
TRAILER_RECORD	7031	MESSAGE_HEADER	40	I
BCAST_MW_ROUND_ROBIN	7201	BCAST_INQUIRY_REPSONSE	452	B
BCAST_TICKER_AND_MKT_INDEX	7202	TICKER_TRADE_DATA	492	B
BCAST_SYSTEM_INFORMATION_OUT	7206	SYSTEM_INFORMATION_DATA	90	B
BCAST_SECURITY_STATUS_CHG_PREOPEN	7210	SECURITY_STATUS_UPDATE_INFORMATION	462	B
UPDATE_LOCALDB_IN	7300	UPDATE_LOCAL_DATABASE	58	I
UPDATE_LOCALDB_DATA	7304	Packet of size >80 and <=512	512	I
BCAST_SECURITY_MSTR_CHG	7305	SECURITY_UPDATE_INFORMATION	240	B
BCAST_PART_MSTR_CHG	7306	PARTICIPANT_UPDATE_INFO	83	B
UPDATE_LOCALDB_HEADER	7307	MESSAGE_HEADER	42	I
UPDATE_LOCALDB_TRAILER	7308	MESSAGE_HEADER	42	I
BCAST_SECURITY_STATUS_CHG	7320	SECURITY_STATUS_UPDATE_INFORMATION	462	B

Transaction Code	Code	Structure	Size	I/B*
PARTIAL_SYSTEM_INFORMATION	7321	SYSTEM_INFORMATION_DATA	90	I
BATCH_ORDER_CANCEL	9002	ORDER_ENTRY_REQUEST	214	I
BCAST_TURNOVER_EXCEEDED	9010	BROADCAST_LIMIT_EXCEEDED	77	B
BROADCAST_BROKER_REACTIVATED	9011	BROADCAST_LIMIT_EXCEEDED	77	B
BOX_SIGN_ON_REQUEST_IN	23000	MS_BOX_SIGN_ON_REQUEST_IN	60	I
BOX_SIGN_ON_REQUEST_OUT	23001	MS_BOX_SIGN_ON_REQUEST_OUT	52	I
BOX_SIGN_OFF	20322	MS_ BOX_SIGN_OFF	42	I
ERROR_RESPONSE_OUT	2302	ERROR_RESPONSE	180	I
HEARTBEAT	23506	HEARTBEAT	40	I
SECURE_BOX_REGISTRATION_REQUEST_IN	23008	MS_SECURE_BOX_REGISTRATION_REQUEST_IN	42	I
SECURE_BOX_REGISTRATION_RESPONSE_OUT	23009	MS_SECURE_BOX_REGISTRATION_RESPONSE_OUT	40	I

* Interactive/Broadcast

Quick Reference for Order Entry Parameters

The order flags are given below.

Order Terms:

MF : input - to be set when the min fill quantity is given

AON : input

IOC : input

GTC : input

Day : input

SL : input

Market : output

ATO : output

STPC : input

Frozen : output

Modified : input

Traded : output

MatchedInd : output

Status	Market	Book Type	Order Terms and Other Characteristic Fields
Preopen	Normal Market	RL**	Non-zero value of Good Till Date/DAY/GTC mandatory, mutually exclusive, input ATO output, set if Market order, value of order price returned is '-1'.
Open	Normal Market	RL**	Non-zero value of Good Till Date/DAY/ GTC/ IOC mandatory, mutually exclusive, input MKT output, set if it is Market order.
Open	Normal Market	SL**	SL mandatory, input Non-zero value of Good Till Date/DAY/ GTC/ IOC mandatory, mutually exclusive, input MF/ AON mutually exclusive, input MKT output, set if Market order Trigger Price is mandatory.
Open	Normal Market	ST**	Non-zero value of Good Till Date /DAY/ GTC/ IOC mandatory, mutually exclusive, input MF/ AON mandatory, mutually exclusive, input MKT output, set if it is Market order.
Open	Normal Market	NT**	DAY mandatory, input Counter Party ID is mandatory.
Open	Spot Market	SP**	DAY/ IOC mandatory, mutually exclusive, input.
Open	Odd Lot Market	OL**	Non-zero value of Good Till Date/DAY/ GTC/ IOC mandatory, mutually exclusive, input. Volume is less than Board Lot quantity.
Open	Auction Market	AU**	DAY mandatory, input. Auction Number and Participant Type are mandatory.

Status	Market	Book Type	Order Terms and Other Characteristic Fields
Close			Order entry is not allowed.

** Other input flags in the order terms are not allowed, hence should not be set.

Market Type

The market types are:

Status	Market Status ID
Normal Market	1
Odd Lot Market	2
Spot Market	3
Auction Market	4

Market Status

The market can be in one of the following statuses:

Status	Market Status ID
PreOpen (only for Normal Market)	0
Open	1
Closed	2
Preopen ended	3

Book Types

There are seven books. These books fall in four markets.

Book Type	Book ID	Market Type
Regular Lot Order	1	Normal Market
Special Terms Order	2	Normal Market
Stop Loss Order	3	Normal Market

Negotiated Order	4	Normal Market
Odd Lot Order	5	Odd Lot Market
Spot Order	6	Spot Market
Auction Order	7	Auction Market

Auction Status

Status	ID	Description
AUCTION_PENDING_APPROVAL	1	If the auction is initiated by the trader an alert is generated at the CWS. The auction status is in pending for approval.
AUCTION_PENDING	2	If any auction in the particular security is already going on, the status of the auction entered next is pending.
OPEN_COMPETITIOR_PERIOD	3	When the auction gets initiated, this is the status.
OPEN_SOLICITOR_PERIOD	4	Auction enters solicitor period.
AUCTION_MATCHING	5	After solicitor period ends, the auction enters matching state. The matching of auction orders takes place.
AUCTION_FINISHED	6	Status after matching of orders is done and auction trades are generated.
AUCTION_CXLED	7	Auction is cancelled by NSE-Control.

Security Status

Status	Status ID
Preopen	1
Open	2
Suspended	3
Preopen Extended	4

Activity Types

The activity types that are sent in reports are given below.

Activity Type	Description	Code

Activity Type	Description	Code
ORIGINAL_ORDER	This is the order that was entered. GTC/GTD orders still in the book also come with this activity type.	1
ACTIVITY_TRADE	The trade was done.	2
ACTIVITY_ORDER_CANCEL	The order was cancelled.	3
ACTIVITY_ORDER_MODIFY	The order was modified.	4
ACTIVITY_TRADE_MOD	The trade was modified.	5
ACTIVITY_TRADE_CXL_1	Trade cancellation was requested.	6
ACTIVITY_TRADE_CXL_2	Action has been taken on this request.	7
ACTIVITY_BATCH_ORDER_CANCEL	At the end of the day, all untraded Day orders are cancelled. GTC/GTD orders due for cancellation are also cancelled.	8

Pipe Delimited File Structures

The upload files have a header record at the beginning of the file followed by the detail records. All the fields in both the header and detail records are separated by pipe ('|').The fields are not of fixed width. Any two fields are separated by a '|' symbol.

Security File Structure

Header

Table 39 SECURITY_FILE_HEADER

Structure Name	SECURITY_FILE_HEADER		
Packet Length	19 bytes		
Field Name	Data Type	Size in Byte	Offset
NEATCM	CHAR	6	0
Reserved	CHAR	1	6
VersionNumber	CHAR	7	7

Structure Name	SECURITY_FILE_HEADER		
Packet Length	19 bytes		
Field Name	Data Type	Size in Byte	Offset
Reserved	CHAR	1	14
DATE	LONG	4	15

Stock Structure

Table 40 STOCK_STRUCTURE

Structure Name	STOCK_STRUCTURE		
Packet Length	266 bytes		
Field Name	Data Type	Size in Byte	Offset
Token	SHORT	2	0
Reserved	CHAR	1	2
Symbol	CHAR	10	3
Reserved	CHAR	1	13
Series	CHAR	2	14
Reserved	CHAR	1	16
InstrumentType	SHORT	2	17
Reserved	CHAR	1	19
IssuedCapital	DOUBLE	8	20
Reserved	CHAR	1	28
PermittedToTrade	SHORT	2	29
Reserved	CHAR	1	31
CreditRating	CHAR	17	32
Reserved	CHAR	1	49
ST_SEC_ELIGIBILITY_ PER_ MARKET [6] (Refer Table 40.1)	STRUCT	30	50
BoardLotQuantity	LONG	4	80
Reserved	CHAR	1	84
TickSize	LONG	4	85
Reserved	CHAR	1	89
Name	CHAR	25	90
Reserved	CHAR	1	115
IssueRate	SHORT	2	116
Reserved	CHAR	1	118
IssueStartDate	LONG	4	119

Structure Name	STOCK_STRUCTURE		
Packet Length	266 bytes		
Field Name	Data Type	Size in Byte	Offset
Reserved	CHAR	1	123
IssueIPDate	LONG	4	124
Reserved	CHAR	1	128
MaturityDate	LONG	4	129
Reserved	CHAR	1	133
FreezePercent	SHORT	2	134
Reserved	CHAR	1	136
ListingDate	LONG	4	137
Reserved	CHAR	1	141
ExpulsionDate	LONG	4	142
Reserved	CHAR	1	146
ReAdmissionDate	LONG	4	147
Reserved	CHAR	1	151
ExDate	LONG	4	152
Reserved	CHAR	1	156
RecordDate	LONG	4	157
Reserved	CHAR	1	161
NoDeliveryDateStart	LONG	4	162
Reserved	CHAR	1	166
NoDeliveryDateEnd	LONG	4	167
Reserved	CHAR	1	171
ParticipantInMktIndex	CHAR	1	172
Reserved	CHAR	1	173
AON	CHAR	1	174
Reserved	CHAR	1	175
MF	CHAR	1	176
Reserved	CHAR	1	177
WarningPercent	SHORT	2	178
Reserved	CHAR	1	180
BookClosureStartDate	LONG	4	181
Reserved	CHAR	1	185
BookClosureEndDate	LONG	4	186
Reserved	CHAR	1	190
Dividend	CHAR	1	191
Reserved	CHAR	1	192
Rights	CHAR	1	193
Reserved	CHAR	1	194

Structure Name	STOCK_STRUCTURE		
Packet Length	266 bytes		
Field Name	Data Type	Size in Byte	Offset
Bonus	CHAR	1	195
Reserved	CHAR	1	196
Interest	CHAR	1	197
Reserved	CHAR	1	198
AGM	CHAR	1	199
Reserved	CHAR	1	200
EGM	CHAR	1	201
Reserved	CHAR	1	202
Spread	LONG	4	203
Reserved	CHAR	1	207
MinQty	LONG	4	208
Reserved	CHAR	1	212
SSEC	SHORT	2	213
Reserved	CHAR	1	215
Remarks	CHAR	25	216
Reserved	CHAR	1	241
LocalDBUpdateDateTime	LONG	4	242
Reserved	CHAR	1	246
DeleteFlag	CHAR	1	247
Reserved	CHAR	1	248
FaceValue	LONG	4	249
Reserved	CHAR	1	253
ISIN Number	CHAR	12	254

Table 40.1 ST_SEC_ELIGIBILITY_PER_MARKET

Structure Name	ST_SEC_ELIGIBILITY_PER_MAKRET		
Packet Length	5 bytes		
Field Name	**Data Type**	**Size in Byte**	**Offset**
Security Status	SHORT	2	0
Reserved	CHAR	1	2
Eligibility	CHAR	1	3
Reserved	CHAR	1	4

Field Name	**Brief Descriptions**
Token	Token number of the security being updated. This is unique for a

Field Name	Brief Descriptions
	particular symbol-series combination.
Symbol	This field should contain the symbol of a security.
Series	This field should contain the series of a security
InstrumentType	This field contains the instrument type of the security. It can be one of the following: ▪ '0' – Equities ▪ '1' – Preference Shares ▪ '2' – Debentures ▪ '3' – Warrants ▪ '4' – Miscellaneous
IssuedCapital	Issue size of the security.
PermittedToTrade	• '0' - Listed but not permitted to trade • '1' - Permitted to trade
CreditRating	Credit rating of the security.
SecurityStatus (1 to 6)	• '1' - Preopen (Only for Normal market) • '2' - Open • '3' - Suspended • '4' - Preopen extended • '5' - Stock Open With Market • '6' – Price Discovery This will contain the Call Auction Markets security status at 5th position The values can be : '1' : Preopen '3' : Suspended '6' : Price Discovery. 6th position reserved for future
Eligibility	• '0' – for Stocks not eligible in current market • '1' – for stocks eligible in current Market 5th Position represents eligibility for Call Auction Market. 6th Position reserved for future
BoardLotQuantity	Regular lot size.

Field Name	Brief Descriptions
TickSize	Tick size/ Min spread size.
Name	Security name.
IssueRate	Price of the issue.
IssueStartDate	Date of issue of the security.
IssueIPDate	Interest Payment Date
IssueMaturityDate	Maturity Date.
FreezePercent	Freeze percent.
ListingDate	Date of listing.
ExpulsionDate	Date of expulsion.
ReAdmissionDate	Date of readmission.
ExDate	Last date of trading before any corporate action.
RecordDate	Date of record changed.
NoDeliveryStartDate	Date from when physical delivery of share certificates is stopped for book closure.
NoDeliveryEndDate	No delivery end date.
NoDeliveryEndDate	Minimum price at which order can be placed without causing a price freeze.
ParticipateInMktIndex	'1' – Security is present in NIFTY Index. '0' – Security is not present in NIFTY Index.
AON	'1'- AON is allowed. '0'- AON is not allowed
MF	'1'- MF is allowed. '0'- MF is not allowed
WarningPercent	Warning percent
BookClosureStartDate	Date at which the record books in the company for shareholder names starts.
BookClosureEndDate	Date at which the record books in the company for shareholder names ends.
Dividend	'1' – Dividend '0' – No Dividend
Rights	'1' – Rights '0' - No Rights
Bonus	'1' – Rights '0' - No Rights

Field Name	Brief Descriptions
Interest	'1' – Interest '0' - No Interest
AGM	'1' – AGM '0' - No AGM
EGM	'1' – EGM '0' – No EGM
Spread	Added for exchange usage
MinQty	Added for exchange usage
SSEC	'1' – Non Call Aution1 Market security. '5' – Call Aution1 market security. This is used as identifier for Call Auction 1 market securities.
Remark	Remarks
LocalLDBUpdateDateTime	This is the local database update date-time.
DeleteFlag	This indicates the status of the security, whether the security is deleted or not. • 'N' : Active • 'Y' : Deleted
FaceValue	This field contains face value of the security
ISIN Number	This field contains the ISIN Number of the security.

Security_slb File Structure

Header

Table 41 SECURITY_SLB_FILE_HEADER

Structure Name	CONTRACT_FILE_HEADER		
Packet Length	19 bytes		
Field Name	Data Type	Size in Byte	Offset
NEATSLB	CHAR	6	0
Reserved	CHAR	1	6
VersionNumber	CHAR	7	7
Reserved	CHAR	1	14
DATE	LONG	4	15

Stock Structure

Table 42 STOCK_STRUCTURE

Structure Name	STOCK_STRUCTURE		
Packet Length	249bytes		
Field Name	Data Type	Size in Byte	Offset
Token	LONG	4	0
Reserved	CHAR	1	4
Symbol	CHAR	10	5
Reserved	CHAR	1	15
Series	CHAR	2	16
Reserved	CHAR	1	18
InstrumentType	SHORT	2	19
Reserved	CHAR	1	21
IssuedCapital	DOUBLE	8	22
Reserved	CHAR	1	30
PermittedToTrade	SHORT	2	31
Reserved	CHAR	1	33
CreditRating	CHAR	17	34
Reserved	CHAR	1	51
ST_SEC_ELIGIBILITY_ PER_ MARKET [4] (Refer Table 42.1)	STRUCT	20	52
BoardLotQuantity	LONG	4	72
Reserved	CHAR	1	76
TickSize	LONG	4	77
Reserved	CHAR	1	81
Name	CHAR	25	82
Reserved	CHAR	1	107
IssueRate	SHORT	2	108
Reserved	CHAR	1	110
IssueStartDate	LONG	4	111
Reserved	CHAR	1	115
IssueIPDate	LONG	4	116
Reserved	CHAR	1	120
ReverseLegSettlDate	LONG	4	121
Reserved	CHAR	1	125
FreezePercent	SHORT	2	126
Reserved	CHAR	1	128
ListingDate	LONG	4	129
Reserved	CHAR	1	133

Structure Name	STOCK_STRUCTURE		
Packet Length	249bytes		
Field Name	Data Type	Size in Byte	Offset
ExpulsionDate	LONG	4	134
Reserved	CHAR	1	138
ReAdmissionDate	LONG	4	139
Reserved	CHAR	1	143
ExDate	LONG	4	144
Reserved	CHAR	1	148
RecordDate	LONG	4	149
Reserved	CHAR	1	153
NoDeliveryDateStart	LONG	4	154
Reserved	CHAR	1	158
NoDeliveryDateEnd	LONG	4	159
Reserved	CHAR	1	163
ParticipantInMktIndex	CHAR	1	164
Reserved	CHAR	1	165
AON	CHAR	1	166
Reserved	CHAR	1	167
MF	CHAR	1	168
Reserved	CHAR	1	169
WarningPercent	SHORT	2	170
Reserved	CHAR	1	172
BookClosureStartDate	LONG	4	173
Reserved	CHAR	1	177
BookClosureEndDate	LONG	4	178
Reserved	CHAR	1	182
Dividend	CHAR	1	183
Reserved	CHAR	1	184
Rights	CHAR	1	185
Reserved	CHAR	1	186
Bonus	CHAR	1	187
Reserved	CHAR	1	188
Interest	CHAR	1	189
Reserved	CHAR	1	190
AGM	CHAR	1	191
Reserved	CHAR	1	192
EGM	CHAR	1	193
Reserved	CHAR	1	194
Remarks	CHAR	25	195

Structure Name	STOCK_STRUCTURE		
Packet Length	249bytes		
Field Name	Data Type	Size in Byte	Offset
Reserved	CHAR	1	220
LocalDBUpdateDateTime	LONG	4	221
Reserved	CHAR	1	225
DeleteFlag	CHAR	1	226
Reserved	CHAR	1	227
FaceValue	LONG	4	228
Reserved	CHAR	1	232
ISIN Number	CHAR	12	233
Reserved	CHAR	1	245
AllowRecall	CHAR	1	246
Reserved	CHAR	1	247
AllowRepay	CHAR	1	248

Table 42.1 ST_SEC_ELIGIBILITY_PER_MARKET

Structure Name	ST_SEC_ELIGIBILITY_PER_MAKRET		
Packet Length	5 bytes		
Field Name	Data Type	Size in Byte	Offset
Security Status	SHORT	2	0
Reserved	CHAR	1	2
Eligibility	CHAR	1	3
Reserved	CHAR	1	4

Field Name	Brief Descriptions
Token	Token number of the security being updated. This is unique for a particular symbol-series combination.
Symbol	This field should contain the symbol of a security.
Series	This field should contain the series of a security

Field Name	Brief Descriptions
InstrumentType	This field contains the instrument type of the security. It can be one of the following: ▪ '0' – Equities ▪ '1' – Preference Shares ▪ '2' – Debentures ▪ '3' – Warrants ▪ '4' – Miscellaneous
IssuedCapital	Issue size of the security.
PermittedToTrade	• '0' - Listed but not permitted to trade • '1' - Permitted to trade
CreditRating	Credit rating of the security.
SecurityStatus	• '1' - Preopen (Only for Normal market) • '2' - Open • '3' - Suspended • '4' - Preopen extended • '5' - Stock Open With Market
Eligibility	Contracts Eligibility in the market
BoardLotQuantity	Regular lot size.
TickSize	Tick size/ Min spread size.
Name	Security name.
IssueRate	Price of the issue.
IssueStartDate	Date of issue of the security.
IssueIPDate	Interest Payment Date
ReverseLegSettlDate	This will contain the reverset leg settlement date of the security
FreezePercent	Freeze percent.
ListingDate	Date of listing.
ExpulsionDate	Date of expulsion.
ReAdmissionDate	Date of readmission.
ExDate	Last date of trading before any corporate action.
RecordDate	Date of record changed.
NoDeliveryStartDate	Date from when physical delivery of share certificates is stopped

Field Name	Brief Descriptions
	for book closure.
NoDeliveryEndDate	No delivery end date.
ParticipateInMktIndex	'1' – Security is present in NIFTY Index. '0' – Security is not present in NIFTY Index.
AON	'1'- AON is allowed. '0'- AON is not allowed
MF	'1'- MF is allowed. '0'- MF is not allowed
WarningPercent	Warning percent
BookClosureStartDate	Date at which the record books in the company for shareholder names starts.
BookClosureEndDate	Date at which the record books in the company for shareholder names ends.
Dividend	'1' – Dividend '0' – No Dividend
Rights	'1' – Rights '0' - No Rights
Bonus	'1' – Rights '0' - No Rights
Interest	'1' – Interest '0' - No Interest
AGM	'1' – AGM '0' - No AGM
EGM	'1' – EGM '0' – No EGM
Remark	Remarks
LocalLDBUpdateDateTime	This is the local database update date-time.
DeleteFlag	This indicates the status of the security, whether the security is deleted or not. • 'N' : Active • 'Y' : Deleted
FaceValue	This field contains face value of the security
ISIN Number	This field contains the ISIN Number of the security.

Field Name	Brief Descriptions
AllowRecall	This indicates whether the security is allowed to recall. 0 –Not allowed 1 – Allowed
AllowRepay	This indicates whether the security is allowed to repay 0 – Not allowed 1 – Allowed

Participant_slb Structure

Header

Table 43 PARTICIPANT_FILE_HEADER

Structure Name	PARTICIPANT_FILE_HEADER		
Packet Length	19 bytes		
Field Name	Data Type	Size in Byte	Offset
NEATSLB	CHAR	6	0
Reserved	CHAR	1	6
VersionNumber	CHAR	7	7
Reserved	CHAR	1	14
DATE	LONG	4	15

Structure

Table 44 PARTICIPANT_STRUCTURE

Structure Name	PARTICIPANT_STRUCTURE		
Packet Length	47 bytes		
Field Name	Data Type	Size in Byte	Offset
ParticipantId	CHAR	12	0
Reserved	CHAR	1	12
ParticipantName	CHAR	25	13
Reserved	CHAR	1	38
ParticipantStatus	CHAR	1	39
Reserved	CHAR	1	40
DeleteFlag	CHAR	1	41
Reserved	CHAR	1	42
LastUpdateTime	LONG	4	43

Field Name	Brief Descriptions
ParticipantId	ID of the Participant
ParticipantName	Name of the participant
ParticipantStatus	If this field is 'S' then the Participant is Suspended. If this is 'A' then the Participant is Active.
DeleteFlag	If this field is 'Y' then the participant is deleted from the system, else he is present in the system.
LastUpdateTime	The last time this record was modified.

Annexure For Encryption/Decryption

Sr. No.	The following are sample function calls of OpenSSL library in Linux (for reference)
1	***Note –*** • Openssl Library version used is OpenSSL 1.1.1. • TLS protocol version has been set to 1.3 (TLS1_3_VERSION). Following are the system library calls for TLS1.3- **SSL/TLS library initialization →** 1. **SSL_library_init**() - Initialize SSL library by registering algorithms. 2. **OpenSSL_add_all_algorithms**() - Adds all algorithms to the table (digests and ciphers) 3. **SSL_load_error_strings**() - Registers the error strings for all libcrypto and libssl error strings. 4. **SSL_CTX_new**(**TLS_client_method**()) - Create a new SSL_CTX object as framework for TLS/SSL enabled functions. 5. **SSL_CTX_set_min_proto_version**(SSL_CTX *ctx, int version) - Set the minimum protocol versions to TLS1_3_VERSION. 6. **SSL_CTX_set_max_proto_version**(SSL_CTX *ctx, int version) - Set the maximum protocol versions to TLS1_3_VERSION. **Establishing the SSL/TLS connection→** 1. **socket**(PF_INET, SOCK_STREAM, 0) - Create TCP socket. 2. **connect**(int sockfd, const struct sockaddr *addr, socklen_t addrlen) - Initiate the TCP/IP connection with server. 3. **SSL_new**(SSL_CTX *ctx) - Create new SSL connection state. 4. **SSL_set_fd**(SSL *ssl, int fd) - Attach the socket descriptor. 5. **SSL_connect**(SSL *ssl) - Perform the SSL connection.

	Validating the Gateway Router server certificate → 1. **SSL_get_peer_certificate**(const SSL *ssl) - Get the server's certificate. 2. **X509_STORE_new**() - This function returns a new X509_STORE. 3. **X509_STORE_CTX_new**() - This function returns a newly initialised X509_STORE_CTX. 4. **X509_STORE_load_locations**(X509_STORE *ctx, const char *file, const char *dir) - Configure files and directories used by a certificate store. The path of CA certificate (gr_ca_cert1.pem) will be used in this function. The CA certificate (gr_ca_cert1.pem) will be provided by the Exchange for validation of Gateway Router certificate. 5. **X509_STORE_CTX_init**(X509_STORE_CTX *ctx, X509_STORE *trust_store, X509 *target, STACK_OF(X509) *untrusted) - This function returns a newly initialised X509_STORE_CTX structure. 6. **X509_verify_cert**(X509_STORE_CTX *ctx) - This function builds and verify X509 certificate chain. **Send and Receive messages on SSL/TLS connection →** 1. **SSL_write**(SSL *ssl, const void *buf, int num) - Send message on SSL. 2. **SSL_read**(SSL *ssl, void *buf, int num) - Receive message from SSL.

2	**For symmetric encryption/decryption methodology –**
	
	<u>Encryption:</u>
	
	Initialization→
	void encrypt_EVP_aes_256_cbc_init(EVP_CIPHER_CTX **ctx, unsigned char *key, unsigned char *iv)
	{
	if(!(*ctx = **EVP_CIPHER_CTX_new()**))
	handleErrors();
	
	if(1 != **EVP_EncryptInit_ex**(*ctx**, EVP_aes_256_gcm()**, NULL, key, iv))
	handleErrors();
	}
	
	Encryption→
	void encrypt(EVP_CIPHER_CTX *ctx, unsigned char *plaintext, int plaintext_len, unsigned char *ciphertext, int *ciphertext_len)
	{
	int len;

```
    if(1 != EVP_EncryptUpdate(ctx, ciphertext, &len, plaintext, plaintext_len))
      handleErrors();
    *ciphertext_len = len;
  }
```

Decryption:

Initialization→

```
  void decrypt_EVP_aes_256_cbc_init(EVP_CIPHER_CTX **ctx, unsigned char *key,
  unsigned char *iv)
  {
    if(!(*ctx = EVP_CIPHER_CTX_new()))
      handleErrors();

    if(1 != EVP_DecryptInit_ex(*ctx, EVP_aes_256_gcm(), NULL, key, iv))
      handleErrors();
  }
```

Decryption→

```
  int decrypt(EVP_CIPHER_CTX *ctx, unsigned char *ciphertext, int ciphertext_len,
  unsigned char *plaintext, int *plaintext_len)
  {
    int len;

    if(1 != EVP_DecryptUpdate(ctx, plaintext, &len, ciphertext, ciphertext_len))
      handleErrors();
    *plaintext_len = len;
  }
```

Note –

- The ones highlighted in bold are OpenSSL library functions.
- plaintext is the actual message buffer.
- ciphertext is the encrypted message buffer.