

Binary for Linux - SCCP

Release Notes for Version 3.01

1 Overview

This release adds a number of configuration options to enhance compatibility with different SCCP networks. New optional functionality includes protocol class validation and forced use of XUDT messages.

The release also correct faults in the generation of the hop counter parameter in XUDTS messages and the generation of UDTs/XUDTS messages in Network Congestion situations.

The release is backwards compatible with the previous release.

2 New functionality

2.1 Called Party address point code handling for GTT relay

The default operation of the module is to add the point code from the MTP routing label when the calling party address does not contain a point code. The existing extended option SCPXF_NO_CALLING_PC allows this point code insertion to be prevented for received messages which terminate locally. This functionality has been extended and point code insertion for messages which are relayed after Global Title Translation can now also be prevented using the same extended option.

2.2 National bit handling for GTT

The handling of the national/spare bit in the SCCP called party address has been extended to offer two modes of operation. The default handling will take the value national/spare bit from the configured primary or secondary address. If the extended option SCPXF_IGNORE_GTT_NAT_IND (0x0800) is set then the value of the national/spare bit will be copied from the original pre-translated address. The handling of addresses which do not require translation is unchanged.

2.3 Protocol Class validation

The module can now optionally validate message types against SCCP protocol classes. Set the extended option SCPXF_VALIDATE_PCLASS (0x0400) in the SCP_MSG_CONFIG (0x7740) message to enable this functionality. Invalid messages will be rejected with a message SCP_MSG_MAINT_IND (0x761) and a status value of SCPEV_STX_ERR (0xb).

2.4 SCCP management support for SST

The mode can now optionally support the handling of received SST message for SCCP Management itself (sub-system 1). If the extended option SCPXF_SCMG_SST_SUPPORT (0x1000) in the SCP_MSG_CONFIG (0x7740) message is set then the module will respond with a SSA in response to a SST for the local point code and SSN 1.

2.5 RSS availability on MTP-RESUME

The default behavior for the module is to initiate a sub-system test (SST) procedure 30 seconds after an MTP-Resume has been received for a remote signaling point. The module now allows the user to turn off the SST procedure for a specific remote sub-system by setting an option in the **ssr_flags** field of the SCP_MSG_CNF_SSR (0x7741) message. This new option SSRF_RSS_AVAIL (0x0004) is only appropriate for remote sub-systems and has no affect on remote signaling points or local sub-systems.

2.6 Forced use of XUDT

The module has been enhanced to allow the use of XUDT messages to be forced used instead of UDT. If XUDT messages are required then the option SCPXF_FORCE_XUDT (0x0200) should be set in the extended options field of the SCP_MSG_CONFIG (0x7740) message.

The XUDT message will include the hop counter value specified in the hop_counter parameter in the SCP_MSG_CONFIG (0x7740) message. If this parameter is not specified then a default value of 15 (0xf) is used instead.

2.7 UDTS/XUDTS Calling Party address handling

This release adds the option to select the calling party address in a UDTS/XUDTS to be either derived from the called party address of the original UDT/XUDT or the address of the local system. By default the module uses the local pc and an unknown SSN for the calling party address of the UDTS/XUDTS. If the option SCPXF_SWAP_UDT_ADDR (0x2000) is set then the called and calling party addresses in the UDT/XUDT are swapped for the UDTS/XUDTS. If required the option should be set in the extended options field of the SCP_MSG_CONFIG (0x7740) message. The option has no affect on the addresses which are not in UDTS/XUDTS messages.

3 Other Changes

3.1 UDTs/XUDTs generation on SCCP link congestion

Previously if a message could not be routed due to network congestion an SCCP Maintenance event with reason SCPEV_RTF_NET_CONG (4) was generated but no UDTs/XUDTs was set back to the sending node. This release now sends back an UDTs or XUDTs as appropriate.

3.2 XUDTs Hop counter value

Previously the value of the hop counter in an XUDTs was one less than the original XUDT message which caused the XUDTs to be generated. This was incorrect and could lead to inappropriate hop counter values.

XUDTs messages will now be initialized to the hop counter value specified in the hop_counter parameter in the SCP_MSG_CONFIG (0x7740) message. If this parameter is not specified then a default value of 15 (0xf) is used instead.

Dialogic
19-Jun-06

Binary for Linux - SCCP

Release Notes for Version 3.02

1 Overview

This release introduces enhanced functionality to allow systems making use of the SMB broadcast functionality to offer faster update of RSS availability on restart of an SCCP module.

The handling of GTT addresses has been modified and may not be compatible if the GTT parameters were not set correctly. The module may be forced to preserve compatibility via an extended option. In other respects this release is compatible with the previous release.

2 New functionality

2.1 RSS Availability Synchronization

Where SCCP is configured to use an SMB broadcast mechanism this release includes functionality to enable Remote Sub-Systems (RSS) to become available to a restarting SCCP module faster.

In the case where an SCCP module is started or restarted on reception of an MTP Resume for a remote point code which matches that of remote subsystems a synchronization procedure is triggered. This is optionally enabled using the SMB flag value 0x1000 in the main SCCP configuration message. The procedure causes a message SCP_MSG_SMB_SSR (0x774d) to be sent to the SMB broadcast module id. This message should be forwarded by the broadcast module to all active SCCP modules.

When the SCP_MSG_SMB_SSR message is received by an SCCP module if it believes the RSS is currently active and this was a result of a previous SSA message then an SSA will be broadcast to the SMB module id and hence on to all other SCCP modules in the system. This will cause the RSS to be marked as available without the restarting SCCP module to need to wait for the sub-system test procedure to be completed.

Note: If the RSS is not known to be active, no message is sent as a reply. If the RSS is believed to be active but this has not been verified by the reception of an SSA then again no message reply is sent. These measures ensure that at least one valid and successful sub-system test procedure has been followed before any SSA is sent.

Note: If the SSRF_RSS_AVAIL SSR configuration flag is set then the synchronization is not performed as the RSS is marked as available immediately.

The message format for SCP_MSG_SMB_SSR is included below but does not need to be sent by the user.

2.1.1 SCCP SSR State Broadcast Request

Synopsis:

Message used to request an SCCP module to broadcast a state indication for the specified SSR.

Message Format:

MESSAGE HEADER		
FIELD NAME		MEANING
type		SCP_MSG_SMB_SSR_REQ (0x774d)
id		0
src		SCP_TASK_ID
dst		SCCP SMB TASK ID
rsp_req		0
hclass		0
status		0
err_info		0
len		9
PARAMETER AREA		
OFFSET	SIZE	NAME
0	1	version
1	1	ssr_type
2	2	nc
4	4	spc
8	1	ssn

Parameters:

version

Must be set to 0.

ssr_type

As defined for **SCP_MSG_CNF_SSR** message.

nc

Network Context

spc

The signalling point code of the resource.

ssn

The sub-system number of the resource.

2.2 Modified GTT handling for GTI, ES and NoA

During Global Title Translation previous releases did not consider the GTI, ES and NoA parameters in the GT to be significant for the translation selection. This release now compares these parameters, where present, in order to select the correct translation. If this behavior is not desired then the SCCP extended option SCPXF_DIS_GTI can be used to disable this.

Dialogic

04-Jan-07

Binary for Linux - SCCP

Release Notes for Version 3.03

1 Overview

This release now correctly handles the SCCP Modify Translation Request message (0x7747) message and no longer rejects valid messages.

This release is compatible with the previous release.

2 Changes

2.1 SCCP Modify Translation Request message

The SCCP Modify Translation Request message (SCP_GTT_MOD, 0x7747) is now handled correctly. In previous releases, valid messages may be rejected with status code 0x06. This is now corrected and valid SCP_GTT_MOD messages should no longer be rejected.

Dialogic
18-Apr-07

Binary for Linux - SCCP

Release Notes for Version 3.04

1 Overview

This release introduces Global Title Load Share Tables for the load sharing of messages matching a specific global title translation rule across several destination point codes. It also increases the number of sub-system resources supported and the number of concerned entities that can be associated with each sub-system resource.

This release is fully backwards compatible with the previous release.

2 New functionality

2.1 Global Title Load Share Tables

Global Title Load Share Tables (GLST) support the distribution of messages for a configured Global Title translation rule across a number of remote point codes. A load share table holds a number of Point Codes across which messages will be distributed. The Point Code selection is made using the Signalling Link Selector (sls) value for messages received from the network, or Sequence Control (seq_ctrl) parameter for messages received from the User Part. This supports in sequence delivery of messages for a particular TCAP dialogue.

Three new messages have been defined:

- Add Global Title Load Share Table Request
- Modify Global Title Load Share Table Request
- End Global Title Load Share Table Request

The configuration requires one additional new message type, Global Title Load Share Table Configuration Request (0x774e) and the addition of two optional parameters to the existing Add Translation Request message (0x7745).

In sequence delivery and loadsharing

For TCAP messages both the sls and seq_ctrl parameter are derived from the TCAP transaction ID such that all messages with the same transaction ID (and thus requiring in sequence delivery) have the same sls and seq_ctrl value.

In the absence of network failure events, any message routed using GLSTs with the same MTP sls or SCCP seq_ctrl value will be sent to the same destination point code. The destination point code to be used for a particular message is determined by indexing a GLST table with

either the sls value for messages received from the network, or the seq_ctrl parameter for messages received from the user.

In the event of a network failure event causing the point code in a GLST associated with a particular sls or seq_ctrl value to become unavailable, the next entry in the GLST table will be used. If the next entry is also unavailable then the next will be considered until all destination point codes have been considered. If no destination pc is found from the GLST table then the message will be routed to the point code found from the GTT translation itself. It should be noted that a GTT rule must still result in a point code being found prior to further routing with a GLST. If the route to the point code found from the GTT is not available then the backup GTT will be used.

Called Party Point Code

The Point Code is normally not left in the SCCP called party address when the message is sent to the network as it is present in the MTP3 Routing Label. If the point code is to be left in the message then the point code from the Load Share Table will be used.

Support has also been added to modify existing GLSTs and add or remove GLSTs from existing GTT rules.

2.1.1 Add Global Title Load Share Table Request

Synopsis:

Message used to add a Global Title Load Share Table for reference by a Global Title.

Message Format:

MESSAGE HEADER		
FIELD NAME		MEANING
type		SCP_MSG_GLST_ADD (0x774e)
id		glst_id
src		Sending module_id
dst		SCP_TASK_ID
rsp_req		used to request a confirmation
hclass		0
status		0
err_info		0
len		132
PARAMETER AREA		
OFFSET	SIZE	NAME
0	2	nc – network context
2	2	Num_pc
4	128	dpc[num_pc] – Array of up to 32 destination point codes.

Description:

Load Share Tables may be added to the system at any time providing that the maximum number supported by the system is not exceeded.

glst_id

logical id of the glst instance.

nc

The network context ID of the point codes.

num_pc

The number of destination point codes in the following array.

dpc[]

Array of up to 32 four byte point codes across which messages will be load shared. If less than 32 point codes have been submitted then the point codes that have been submitted will be repeatedly added to the GLST table in turn until the table is full.

2.1.2 Modify Global Title Load Share Table Request

Synopsis:

Message used to modify a previously configured Global Title Load Share Table.

Message Format:

MESSAGE HEADER		
FIELD NAME		MEANING
type		SCP_MSG_GLST_MOD (0x774f)
id		glst_id
src		Sending module_id
dst		SCP_TASK_ID
rsp_req		used to request a confirmation
hclass		0
status		0
err_info		0
len		132
PARAMETER AREA		
OFFSET	SIZE	NAME
0	2	nc – network context
2	2	Num_pc
4	128	dpc[num_pc] – Array of destination point codes.

Description:

Load Share Tables may be modified at any time.

glst_id

logical id of the glst instance.

nc

The network context ID of the point codes.

num_pc

The number of destination point codes in the following array.

dpc[]

Array of up to 32 four byte point codes across which messages will be load shared. If less than 32 point codes have been submitted then the point codes that have been submitted will be repeatedly added to the GLST table in turn until the table is full.

2.1.3 End Global Title Load Share Table Request

Synopsis:

Message used to end a previously configured Global Title Load Share Table.

Message Format:

MESSAGE HEADER	
FIELD NAME	MEANING
type	SCP_MSG_GLST_REM (0x7750)
id	glst_id
src	Sending module_id
dst	SCP_TASK_ID
rsp_req	used to request a confirmation
hclass	0
status	0
err_info	0
len	0

Description:

Load Share Tables may only be ended if no GTT rules are currently referencing it.

glst_id

logical id of the glst instance.

2.1.4 New GTT Configuration Parameters

The SCP_MSG_GTT_ADD and SCP_MSG_GTT_MOD commands now support the following parameters to identify GLST tables for use.

Parameter name	SCPPN_PRIMARY_GLST (value 0x20)
Parameter length	Fixed to 2
Parameter data	Logical ID of the GT Load Share Table for use with the Primary translation.

Parameter name	SCPPN_BACKUP_GLST (value 0x21)
Parameter length	Fixed to 2
Parameter data	Logical ID of the GT Load Share Table for use with the Backup translation.

Dialogic
08-Aug-07
Updated 05-Oct-07

Binary for Linux - SCCP

Release Notes for Version 3.05

1 Overview

This release increases the number of sub-system resources supported and the number of concerned entities that can be associated with each sub-system resource. It also corrects the handling of the removal of concerned resources.

This release is fully backwards compatible with the previous release.

2 Changes

2.1 Increased sub-system resources

This release increases the number of sub-system resources (SSR) which can be configured to 512. These resources are shared between local sub-systems, remote sub-systems and remote point codes.

2.2 Increased concerned entity support

This release increased the number of sub-system resource entities that can be concerned about a particular sub-system resources (SSR). It is now possible for up-to 32 other SSR to be concerned about a specified SSR.

2.3 SCP_MSG_END_CONC message

The SCP_MSG_END_CONC (End Concerned Resource Request, 0x7744) message now only removes the resources specified in this message.

Previously, the SCP_MSG_END_CONC message may occasionally remove another concerned resource in addition to those specified in the message. This behaviour only occurred after a number of concerned resources had been added and the resource to be removed was not the last added.

Under such conditions, this prevented the resource erroneously removed from receiving notification messages. The SCP_MSG_END_CONC message would be rejected with an error status if an attempt was made at a later stage to remove the affected resource.

Binary for Linux - SCCP

Release Notes for Version 3.06

1 Overview

This release adds enhancements to the handling of Concerned Sub-system Resources and corrects a related fault that prevented remote signalling points from being informed of local sub-system status changes.

This release is fully backwards compatible with the previous release.

2 New Functionality

2.1 Improved message handling of Concerned Local Sub-systems on receipt of MTP-PAUSE

When an MTP-Pause is received from a Remote Signalling Point (RSP), all local sub-systems that have been set up as concerned with the RSP will be informed of the status change. The previous release processed all concerned relationships up to a maximum of 8 RSPs before introducing a gap until the next system timer tick. Since the maximum number of concerned resources per RSP is 32, this potentially allows up to 256 messages to be produced per 100 msec timer tick. In most system configurations a much lower number of peak messages per tick would be produced.

This software release checks whether message gapping is required after the resources for each signalling point has been processed. A gap is introduced after the processing of the concerned relationships for a single sub-system resource when 8 or more messages have been sent since the previous gap. This will ensure that if many indications are required that under 40 messages will be generated per timer tick.

2.2 Add message gapping when informing Concerned Local Sub-systems on receipt of MTP-RESUME

When an MTP3 Resume is received from a remote signalling point, it is possible for large numbers of continuous messages to be generated by the module. Local sub-systems that have been set up as concerned with the signalling point's resources will be informed of the status change. The immediate output of a large number of messages could occur if remote sub-systems at that signalling point have been configured to be deemed immediately available by setting the SSRF_RSS_AVAIL flag, bit 2, in the SCP_MSG_END_SSR (0x7742) message.

This release extends the message gapping introduced in Section 2.1 to the scenario described above for the reception of MTP-Resume.

3 Faults cleared

3.1 Failure to send to more than 8 Concerned Remote Signalling Points

The previous release would not inform more than 8 remote signalling points set up as concerned for a local sub-system. The module will allow up to 32 concerned resources to be defined for a sub-system. However, when change of state messages are required to be sent to remote signalling points, only the first 8 are sent the message.

This is corrected such that up to 32 remote signalling points will be notified of status changes to the local sub-system, as appropriate.

Dialogic
19-Oct-07

Binary for Linux - SCCP

Release Notes for Version 3.07

1 Overview

This release makes the SCCP module compliant with the Q.714 specification with regard to unrecognised parameters in messages received from the network layer. Unrecognised parameters are now relayed transparently. The release changes the Global Title Load Share Table (GLST) operation to allow the user to select one of two forms of routing based resource availability. The two modes are appropriate either for routing via a number of intermediate signalling points or for routing direct to a number of end points. The release clears two faults: one that could have affected the gapping of status change messages sent to local sub-systems and one related to the reception of segmented messages that caused messages not to be released back to the message pool.

This release has changed two message definitions (see Sections 2.1.1 and 2.1.2) however it retains backward compatibility with the previous release. Customers using the Global Title Load Share Tables and customers using message segmentation should upgrade to this release. Other users need not upgrade.

2 New Functionality

2.1 Add availability testing option to Global Title Load Share Table

This release modifies the operation of the Global Title Load Share Table feature. Two modes of operation are now supported:

- Load sharing via a number of intermediate signalling points
- Load sharing direct to a number of end points

The GLST operation modes are selected using the SCP_GLSTF_PC_AND_SS_AVAIL option bit of a new 'options' parameter defined for the GLST Add and Modify messages.

The option bit selects the following modes of operation :

- SCP_GLSTF_PC_AND_SS_AVAIL = 0 : this setting should be used for load sharing via a number of intermediate signalling points.
In this mode, if the Primary GTT translation is available, the Primary GLST table is checked. If the Primary GTT translation is unavailable, the Backup GTT translation used and the Backup GLST table is checked.

When checking the GLST tables, the table entry is used if the module knows that the entry's point code is available. This mode of operation is the same as used for the previous SCCP release.

- `SCP_GLSTF_PC_AND_SS_AVAIL = 1` : this setting should be used for load sharing direct to a number of end points. In this mode, after the Primary GTT translation is performed the Primary GLST table is checked. If the Primary GLST table entry cannot be used (see below), the Backup GTT translation is performed and the Backup GLST table is checked. When checking the GLST tables, the table entry is used if the module knows that both the point code is available and the sub-system 'has not been reported as being unavailable'. This means that if the point code is available, the entry will be used if: the sub-system has been reported as being available; or the sub-system has not been configured; or the translation being tested has no sub-system number.

It is recommended that the `SCP_GLSTF_PC_AND_SS_AVAIL` option bit is set to the same value for both Primary and Backup GLST tables. Where the tables to have different settings, the Primary GLST option bit would select both when the GLST tables are checked (as described above) and the availability test used for the Primary GLST table entries. The Backup GLST option bit would only define the availability test used for that table's entries.

GLST tables can be configured to include the Local Point Code (LPC). When this is the case, messages are directed to the LPC according to the GLST load sharing rules with the exception that the addressed local sub-system (LSS) must be available for the LPC to be used, i.e. the LSS is tested for availability even if the `SCP_GLSTF_PC_AND_SS_AVAIL` option bit is set to 0. The LPC is always considered to be available.

The following message definitions re-define the GLST Add and Modify messages replacing the message definitions listed for SCCP Linux Binary Release 3.04. The SCCP module remains compatibility with the previous messages definitions. Customers are however advised to move to using the new message definitions.

2.1.1 Add Global Title Load Share Table Request

Synopsis:

Message used to add a Global Title Load Share Table for reference by a Global Title.

Message Format:

MESSAGE HEADER		
FIELD NAME		MEANING
type		SCP_MSG_GLST_ADD (0x774E)
id		GLST id.
src		Sending module_id
dst		SCP_TASK_ID
rsp_req		used to request a confirmation
hclass		0
status		0
err_info		0
len		136
PARAMETER AREA		
OFFSET	SIZE	NAME
0	4	options – option flags
4	2	reserved – set to 0
6	2	num_pc
8	128	dpc[num_pc] – Array of destination point codes.

Description:

Load Share Tables may be added to the system at any time providing that the maximum number supported by the system is not exceeded.

id

Logical id of the GLST instance.

options

Option flags to control GLST functionality for this GLST instance. The bit mapped options are shown in the following table -

Bit	Mnemonic	Description
0	SCP_GLSTF_PC_AND_SS_AVAIL	If set, the point code and the sub-system is tested for availability. If not set, only the point code is tested for availability.
1-31		All other bits are reserved for future use and must be set to zero.

num_pc

The number of destination point codes in the following array.

dpc[]

Array of up to 32 * 4 byte point codes across which messages will be load shared. If less than 32 point codes are submitted the user should pad the end of the array with zero to the correct length.

2.1.2 Modify Global Title Load Share Table Request

Synopsis:

Message used to modify a previously configured Global Title Load Share Table.

Message Format:

MESSAGE HEADER		
FIELD NAME		MEANING
type		SCP_MSG_GLST_MOD (0x774F)
id		GLST id.
src		Sending module_id
dst		SCP_TASK_ID
rsp_req		used to request a confirmation
hclass		0
status		0
err_info		0
len		136
PARAMETER AREA		
OFFSET	SIZE	NAME
0	4	options – option flags
4	2	reserved – set to 0
6	2	num_pc
8	128	dpc[num_pc] – Array of destination point codes.

Description:

Load Share Tables may be modified at any time.

id

Logical id of the GLST instance.

options

Option flags to control GLST functionality for this GLST instance. The bit mapped options are shown in the table for section 2.1.1.

num_pc

The number of destination point codes in the following array.

dpc[]

Array of up to 32 * 4 byte point codes across which messages will be load shared. If less than 32 point codes are submitted the user should pad the end of the array with zero to the correct length.

2.1.3 Change to use of the SCP_MSG_GTT_MOD message type

The SCP_MSG_GTT_MOD (0x7747) message will now accept any parameter in addition to the reference *Target address*. The user can supply the *Primary address*, *Backup address*, *GTT Mask*, *Primary GLST ID* or the *Backup GLST ID* or any combination of these. Previously the message had to supply at least the *Primary address* and/or the *Backup address*. For both the *Primary address* and the *Backup address*, if they are not supplied any existing configuration for that parameter is retained. For the other parameters (*GTT Mask*, *Primary GLST ID* or *Backup GLST ID*), if they are not supplied any existing configuration for that parameter is removed.

2.2 Unrecognised Parameters are now relayed transparently

The module will now transparently transport any unrecognised parameters when a received MTP message is relayed. Previous module versions discarded unrecognised parameters. This meant that the parameters were missing if the module subsequently relayed the message. The module behaviour now corresponds to the Q.714 specification (07/96) Section 1.1.4.2.

The SCCP User is not made aware of the reception of the unrecognised parameters. The parameters are only used if the message is relayed, they are then discarded.

The unrecognised parameters are always in the optional part of the SCCP message. Unrecognised parameters will therefore only be detected for SCCP message types which have an optional part, i.e. message types which have 'recognised' optional parameters.

The module will position the unrecognised parameters after all recognised optional parameters in the relayed messages (this may be different from the received parameter order, however Q.713 specification permits parameters to be sent in any order).

The module maintains an internal buffer of 100 characters per message for unrecognised parameters. The module will discard detected unrecognised parameters if there is not enough space to fully store them in the buffer. The received message is still processed and if required relayed. Only the unrecognised parameters stored in the buffer are used for a relayed message. If any unrecognised parameters

are discarded for a message a Software Event (SCP_MSG_ERROR_IND, 0x0762) is sent to the Maintenance module. A new software event is defined –

Mnemonic	Value	id	Description
SCPSWE_UNREC_PARM_DISCARDED	23	0	MTP unrecognised parameter was discarded due to not enough space in the internal buffer.

3 Faults cleared

3.1 Segmentation fails when messages mix

A fault in the SCCP module has been fixed in the reassembly of segmented messages received from the MTP3 module. The fault occurred when the first segment of a message is received followed by all segments of a second message. The remaining segments of the first message are then rejected by the module when they are received. When this happens a message is not released back to the message pool. If segmented messages are continually being received in this way, the message pool will be exhausted.

The fault is fixed in this release. The module no longer rejects segmented messages when they are mixed together and all messages allocated by the module are released correctly.

Customers not using segmentation and reassembly procedures would not be affected by this fault.

3.2 Update Count of Local Concerned Resources

A fault has been fixed that could have affected the gapping of status change messages sent to local sub-systems. These messages should be sent in bursts of eight with a gap between bursts. The fault could cause less messages to be sent in a burst. The fault was only possible when local concerned resources have been added then removed using the SCP_MSG_ADD_CONC and SCP_MSG_END_CONC messages.

Dialogic
19-Feb-08

Binary for Linux - SCCP

Release Notes for Version 3.08

1 Overview

This release corrects the handling of Connection Orientated (CO) SCCP messages. Customers not using Connection Orientated SCCP need not upgrade.

This release is backwards compatible with the previous release.

2 Changes

2.1 Allow Connection Orientated messages with no data field to be sent

Connection orientated (CO) SCCP messages with no data field present are no longer rejected and may now be sent to the network. This change only affects Connection Oriented SCCP users and is not significant to Connectionless SCCP users.

Dialogic
06-Nov-08

Binary for Linux - SCCP

Release Notes for Version 3.09

1 Overview

This release extends the existing Global Title Load Share Tables which supported Class 1 messages so that the functionality also applies with Class 0 messages. This release also adds the ability to send trace messages to a nominated Trace module and corrects a number of faults as detailed below.

This release is backwards compatible with the previous release.

2 New functionality

2.1 Global Title Load Share handling of Class 0 messages

Global Title Load Share Tables use the User SCPPN_SEQ_CTRL parameter (or received SLS value) to determine the Load Share Table entry (point code) to use. The SCPPN_SEQ_CTRL parameter is only present for Class 1 messages so the module has been enhanced so that for non-Class 1 messages, the Load Share Tables use the message's SLS value. The SLS value for these messages is an incrementing value, updated each time a (non-Class 1) message is sent.

Operation using Class 1 messages is not affected by this change.

Note: It is recommended that Class 1 messages are used for Global Title Load Share Tables in order to preserve message sequence throughout a network whenever a dialogue may have more than one message sent in the same direction.

2.2 User configurable trace module id

A Trace Module ID can now be defined in the SCCP Module Configuration message (SCP_MSG_CONFIG). The new '**trace_id**' field uses a previously reserved field in the message. If trace_id is defined as non-zero, the all trace messages are sent to that module id. If trace_id is set to zero, trace messages are sent to the Management Module ID (mngt_id) as for previous releases.

The format of the SCCP Module Configuration Message is now defined as –

MESSAGE HEADER		
FIELD NAME		MEANING
type		SCP_MSG_CONFIG (0x7740)
id		0
src		Sending module_id
dst		SCP_TASK_ID
rsp_req		used to request a confirmation
hclass		0
status		0
err_info		0
len		60
PARAMETER AREA		
OFFSET	SIZE	NAME
0	1	cnf_ver - must be set to one
1	1	sio - Service Information Octet value
2	2	options
4	1	module_id - SCCP module id
5	1	mtp_id - MTP module id
6	1	mngt_id - Management module id
7	1	maint_id - Maintenance module id
8	4	point_code - Local point code
12	2	max_sif
14	1	sccp_instance
15	1	smb_id - Broadcast module id
16	2	smb_flags - Broadcast flags
18	2	num_uc - Required number of User connections
20	2	uc_onset - Onset of congestion point for user connections
22	2	uc_abmt
24	2	num_ic
26	2	ic_onset
28	2	ic_abmt
30	2	num_data
32	2	data_onset
34	2	data_abmt
36	2	num_edata
38	2	edata_onset
40	2	edata_abmt
42	2	ext_options
44	2	base_id

MESSAGE HEADER		
FIELD NAME		MEANING
46	2	top_id
48	2	min_id
50	2	max_id
52	1	isup_id
53	2	nc
55	1	hop_counter
56	2	error_offset
58	1	trace_id – Trace module id
59	1	reserved – set to zero

trace_id

Module identifier defining the destination for all trace messages.

3 Faults cleared

3.1 SCP_MSG_R_SSR_STATS bss_msg count for LSS only

The 'bss_msg' count reported by the SCP_MSG_R_SSR_STATS message has been changed so that only Local Sub-Systems (LSS) will report a count. This is consistent with the Q.752 standard (section 3.4.5). This count is number of messages for a LSS that were directed to a backup sub-system due to the LSS being unavailable.

3.2 SCCP Sub-System Resource initialisation state

The SSRF_RSSR_INIT option is added to the SCP_MSG_CNF_SSR message to define the initial availability state of the Sub-System Resource (SSR) assumed by the SCCP module. The option applies only to Remote Signalling Points (RSP) and Remote Subsystems (RSS), it is ignored by Local Sub-Systems (LSS). The initial availability states for the SSRs are defined in the table below:

Sub-System	SSRF_RSSR_INIT option	
	0	1
RSP	Allowed	Prohibited
RSS	Allowed	Takes state from the RSP
LSS	Prohibited	Prohibited

This option has been added to prevent the initial availability state of a SSR being out of step with the MTP3 state. This is especially applicable when a SSR configuration is added after the initial SCCP configuration.

3.3 max_sif configuration up to 315 octets

The max_sif parameter for the SCP_MSG_CONFIG message can now be set to a value up to 315 octets. This allows systems that use SCCP with non-MTP3 protocols, e.g. M3UA, to send messages that are greater than the 272 octet limit imposed by MTP3. SCCP can generate valid non-segmented messages that are greater than 272 octets, setting a max_sif limit of up-to 315 octets allows these messages to be passed. The SCP_MSG_CONFIG message will be rejected if a max_sif value greater than 315 is set. The user should ensure that the remote end is capable of receiving the messages generated if max_sif is set to a value greater than 272.

Users using SCCP with MTP3 should still ensure that max_sif is configured to 272.

3.4 SCPF_PLIST option no longer supported

The SCPF_PLIST option for the SCP_MSG_CONFIG message is no longer supported. The ITU formatting associated with this option is no longer required and has been removed. SCCP Programmer's Manual states that this option must be set to 1.