Binary for Solaris - TUP

Release Notes for Version 2.00

1 Overview

This release adds support for licensing of TUP for a particular machine. It is backwards compatible with the previous release (V1.00).

2 New functionality

2.1 Support for end user licensing

Prior to this release the use of Host Protocol Software required a signed licence agreement between the supplier and the end user.

This release offers the ability to licence the software for a specific target machine using a licence file obtained electronically without the need to enter into a customer specific licence agreement.

To determine whether a licence file is required, run the binary using the –v command line option. If a licence file is required please refer to the *Host Protocol Licensing User Guide*. Note that a separate licence file is required for each target machine.

Intel Corporation
25-Feb-03

Binary for Solaris - TUP

Release Notes for Version 2.02

1 Overview

This is the first full release since V2.00. It introduces the ability to run the binary for up to 10 hours at a time without a licence whilst developing an application. It also provides support for an increased number of circuits (up to 64K) and clears various problems as described below.

The software is backwards compatible with the previous release.

2 New Functionality

2.1 Trial mode

An additional command line parameter '-t' has been added. If this flag is set then the binary will run in trial mode and the presence of a valid license certificate will not be enforced. The binary will, however, terminate after ten hours.

The licensing command line parameters (-Lp, and -Lt) should not be used when the trial mode is configured.

2.2 Increase to 64K circuits

TUP now supports up to 65535 circuits and 2732 circuit groups. Supporting 64K circuits may require some changes to the application: see 2.3 and 2.4.

2.3 Change in use of id field

The most significant bit of the id field in the TUP-Transmit Request (CAL_MSG_TX_REQ) is no longer ignored when more than 32768 circuits are configured in TUP_MSG_CONFIG.

Early versions of the TUP module (supporting less than 32K circuits) required the most significant bit to be set for outgoing calls so that an id value of 0x8001, for example, would be used to indicate an outgoing call on circuit id 1. In more recent versions, this bit has been ignored by the TUP module, so that either 0x8001 or 0x0001 could be used to indicate an outgoing call on circuit id 1. In this release, when more than 32768 circuits are configured, the most significant bit is considered to be part of the circuit id so that an id value of 0x8001 indicates a call (in either direction) on circuit id 32769. When 32768 circuits or less are configured, the MSB of the call_ref is ignored by the TUP module.

2.4 New module option TUPF_16CID in TUP_MSG_CONFIG

A new per module option, TUPF_16CID (bit 12), has been added to the <options> field in the Configure Request message (0x7700).

If the TUPF_16CID option is not set and up to a maximum of 32768 circuits are configured (in TUP_MSG_CONFIG), the most significant bit of the 16-bit id field in the TUP-Receive Indication (CAL_MSG_TX_REQ) will be set by the TUP module to indicate an outgoing call (compatible with release 2.00). If a maximum of 65535 circuits are configured, the most significant bit is part of an actual circuit id.

If the option is set, the most significant bit will be part of an actual circuit id whether the maximum number of circuits configured is 32768 or 65535.

3 Faults cleared

3.1 Mapping of nature of address subfield

When the nature of address subfield is set to “subscriber number” in the Called party number, Calling party number or Original called number, this is now correctly set to “subscriber number” in the outgoing message. Previously it was mapped to “national number” in the outgoing message.

3.2 SSUTR2: Wrong option used for T113

T113 (timer waiting for first backwards message) is now started when the TUPGOP_RESP_TIM is set, rather than the TUPGOP_SAM_TIM option is set.

3.3 Hardware blocking status cancelled

If a call is in the process of being set up and the circuit is hardware blocked, all subsequent call processing messages are now ignored until the circuit is unblocked. Previously, receipt of an unexpected call set up message (for example an ACM for the call that was in progress) would have resulted in the circuit being reset and the hardware blocked condition being cancelled.

3.4 Incorrect circuit status reported

The status of circuits in a circuit group may be queried by sending a Read Circuit Group Status request to the TUP module. The confirmation message that is returned now only contains status for circuits that belong to the circuit group specified in the ID field in the request message. Previously, status could also be returned for circuits

belonging to other groups if the <cic_mask> field in the request message had extra bits set.

Intel Corporation
17-May-04

Binary for Solaris - TUP

Release Notes for Version 3.00

1 Overview

This is the first release since V2.02 and is functionally equivalent to that release. This release is developed for use with the SS7 Development Package for Solaris V3.00 or later. It cannot be used with earlier development packages.

Customers who wish to make use of the Large Message support offered in V3.00 of the development package should upgrade to this release of software (Large Message support is required for SCCP Segmentation). Other customers need not upgrade.

2 Changes

2.1 Use of Solaris shared library

This release makes use of shared library version of the GCT library included in V3.00 of the development package. The module has not been changed to support Large Messages itself but it does permit the module to be used in an environment where Large Messages are being used.

Intel Corporation
13-Apr-06