Asterisk and SS7 Performance Tests

Jan Rudinský

CESNET z.s.p.o., Zikova 4, Praha, Czech Republic

(Technical Report is under oponent review)

rudinsky@cesnet.cz

6. 11. 2007

At Cesnet z.s.p.o. we have installed and configured Asterisk PBX with Signalling System #7 (SS7) support. This document describes tests to verify a functionality of two SS7 implementations for Asterisk PBX. The tests include interoperability with PSTN switch, signalling procedures verification and load performance tests using SS7 protocol analyzer/simulator..

Keywords: Asterisk, SS7, chan_ss7, libss7, performance testing

1. Introduction

PSTN, the world's circuit-switched network, has employed Signalling System #7 as its protocol suite for international and national interconnection during past decades. VoIP networks however have developed different signalling protocols suitable for IP environment. Gateways interconnecting VoIP and PSTN networks are usually proprietary and expensive solutions. Today an open source software is able to perform this function. As an example we have decided to test open source PBX Asterisk and two open source implementations of SS7, the SS7 channel driver and SS7 library.

Asterisk is an open source software private branch exchange (PBX). It has a large functionality representing traditional PBX system with voice mail, interactive voice response system (IVR), automatic call distributor and signalling or media gateway functions.

Asterisk can interface both traditional TDM based systems (PSTN or PLMN networks) as well as packet based systems (VoIP networks). In our project we have focused on the PSTN interconnection. We tested the link between Asterisk PBX and other telecommunication equipment using SS7.

2. Asterisk and SS7 installation and configuration

We have downloaded Asterisk source code from developers web page and installed the application on Linux operating system.

SS7 implementations for Asterisk are available in several variants. We have focused on open the source software. Chosen representatives were SS7 channel driver (chan_ss7) and SS7 library (libss7).

Asterisk SS7 channel driver (chan_ss7) is an open source software developed by Danish company SIFIRA A/S. It has been released under GPL license (General Public License) and it is not officially certified for SS7 interoperability. Functionality of the solution includes implementations of MTP2 layer, bare essentials of MTP3 and a large subset of ISUP functions.

SS7 library is the latest implementation of native SS7 support for Asterisk. It was released by Asterisk developers in summer 2006. Libss7 functionality covers the ITU variant of SS7 implementing MTP2, MTP3, and ISUP protocols.

A complete process of installation and configuration of Asterisk PBX and installation and configuration of SS7 variants for Asterisk (chan_ss7 and libss7) is described in Cesnet Technical Report “Asterisk and SS7” [1].

3. Asterisk and SS7 tests

3.A. Interconnection tests

First we run interconnection tests to verify basic call control functionality. As a result of cooperation between Cesnet z.s.p.o. and CTU-Ericsson-Vodafone R&D Centre (RDC) we have interconnected PSTN switch with Asterisk PBX using SS7. The PSTN exchange was Ericsson AXE platform.

Physical layer of the connection was represented by one trunk (E1 interface) terminated at Digium PCI card on Asterisk server. Trunk configuration composed of 32 time slots with following structure:

- time slot 0 - synchronization
- time slot 16 - signalling channel (with CRC-4)
- remaining 30 time slots - voice channels.

Upper layers of SS7 including MTP 2, MTP 3 and ISUP were implemented by chan_ss7 and libss7 solutions. This functionality was automatically loaded and SS7 link initialized when Asterisk started.

After Asterisk startup and SS7 link initialization we exchanged calls in both directions over the E1 interface. Incoming calls from PSTN (PLMN) were either routed to VoIP terminals or terminated in IVR. Calls from VoIP clients via Asterisk were routed to PLMN exchange and terminated on mobile phones. All test calls were successful.

3.B. Conformance tests

Next step was to perform conformance tests according to ITU recommendations Q.784 and Q.785.

In cooperation with Sitronics TS we have interconnected Tektronix K1297 protocol tester and Asterisk SS7 using E1 interface. We have configured the conformance test suite on the protocol tester and started the tests. We found out that the protocol tester insisted on voice channel initialization (group reset) done by the remote side before every single test (though SS7 link was inservice already). Unfortunatelly none of the SS7 implementations (chan_ss7, libss7) supported this feature.

Therefore we couldn't complete the conformance tests.

3.C. Performance tests

Final and most extensive part of SS7 implementation testing were performance tests. We ran performance tests of Asterisk with SS7 support to test the ability of load processing and unexpected situations handling. Tests were carried out in cooperation with Sunrise Telecom (STT) and HKE who kindly provided us with STT Multi-service Analyzer (MSA).

STT-MSA can operate either as a protocol analyzer or as a signalling node simulator. STT-MSA protocol analyzer was used to monitor signalling message exchange and signalling procedures verification in the first part of the tests. SST-MSA simulator was later used as a call generator and call data record (CDR) collector to perform load performance tests.

3.C.1 Signalling procedures monitoring

We have analyzed signalling message exchange during SS7 link initialization. Figure 1 represents MTP and ISUP message exchange observed when SS7 link was brought into service. The signalling node with point code “806” represents Asterisk (with chan_ss7 or libss7) and “2001” is the point code of protocol tester.

Fig. 1. SS7 link establishment (chan_ss7 and libss7)

The signalling procedures include messages such as SLTM and SLTA messages that test the signalling link. STLM carries a code sequence that has to be identical to the sequence received by STLA message from the adjacent node. TRA (Traffic Restart Allowed) indicates the end of routing information exchange on MTP layer and CBD (changeback declaration) confirms particular signalling link as a current signalling channel.

Main difference between these signalling procedures is that only libss7 exchange GRS/GRA group reset messages to initialize voice channels, while chan_ss7 does not.

Verification of other signalling procedures such as call setup and tear down are desribed in following paragraphs.

3.C.2 Chan_ss7 performance test

We have used the STT-MSA capability of a call generator. The tester has offered around ten thousands of call requests during a period of about twelve minutes. The signalling procedure of a typical successful call is on figure 2.

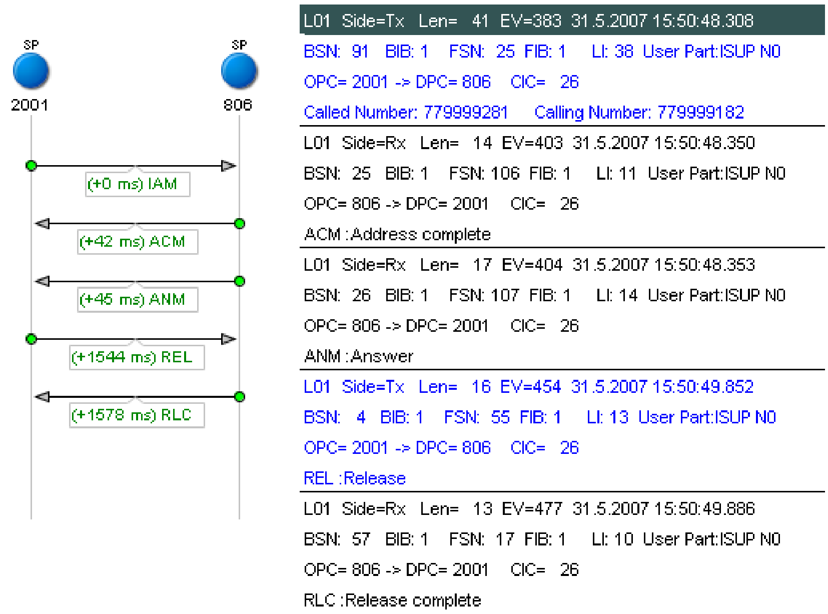

Fig. 2. Performance tests - typical test call

Call is initiated by ISUP Initial Address Message (IAM) which is acknowledged by Address Complete Message (ACM) sent by remote side. Call answer is signalled by Answer Message (ANM) sent back to initiator. After random period of time the call is released by exchange of Release (REL) and Release Complete (RLC) messages. All behaviour corresponds with ITU recommendations.

Following graphs display the total amount of requested calls in testing period (on the left) and the average number of concurrent calls being actively exchanged between PSTN and Asterisk (on the right).

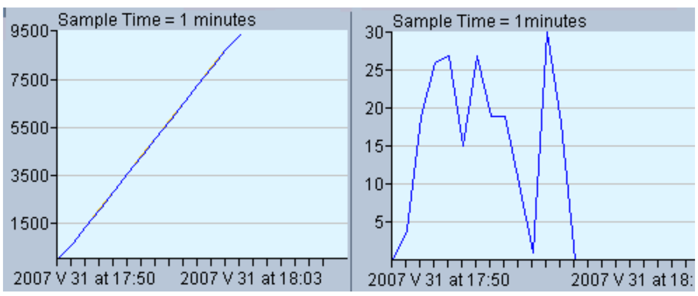

Fig. 3. Requested calls in time, average number of served calls

All calls have been successfully established during the test. We encountered only one difference from standard procedure were Asterisk sent a REL message few miliseconds before tester REL. Although this call was successfully closed by RLC message.

3.C.3 Libss7 performance test

The duration of libss7 performance test was also approximately twelwe minutes and the number of requested calls reached ten thousands.

Asterisk with libss7 had problems with serving the offered calls at initial phase of test. Signalling message exchange differed from the standard. Asterisk often sent REL message, although it is the protocol tester that initiated the call and that would normally clear the call. And though the release cause in Asterisk messages stated “normal call clearing” more probable cause can be lack of resources for processing the call.

A serious problem has occurred in about one third of the test. Asterisk has stopped serving the calls. Repeated tests confirmed this behavior. The final situation was that Asterisk system process still ran on the server and administrator could log into the system console. However no commands on Asterisk command line interface could be executed and system did not respond. It was necessary to restart the application.

Total amount of requested calls in time (on the left) and the average number of concurrent calls actively served by Asterisk (on the right) is displayed on following graphs. The problems in the intial phase and later the processing outage is clearly visible on Figure 4.

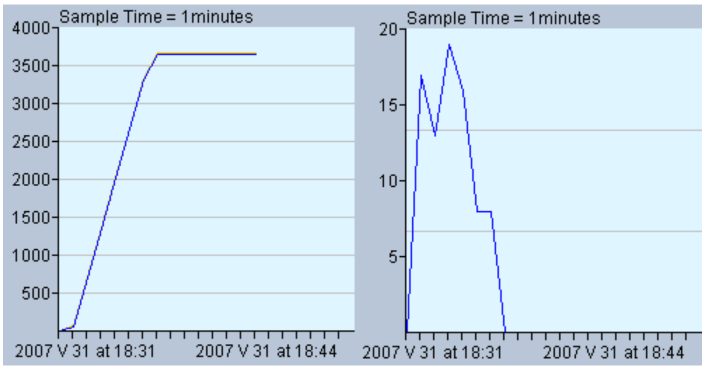

Fig. 4. Requested calls in time, average number of served calls

3.C.4 Non-existent CIC tests

To proof unexpected situations handling we have ran non-existent circuit identification code (CIC) tests. We simulated signalling messages that had been assigned to non-existent voice channels (using non-existent CIC) using the protocol tester. The messages were sent to Asterisk under test.

In case of libss7 the Asterisk has reported incorrect message on the command line interface and received message was ignored. Figure 5 shows CDR for IAM message assigned to CIC 32 (out-of-range) and REL message sent after response timeout.

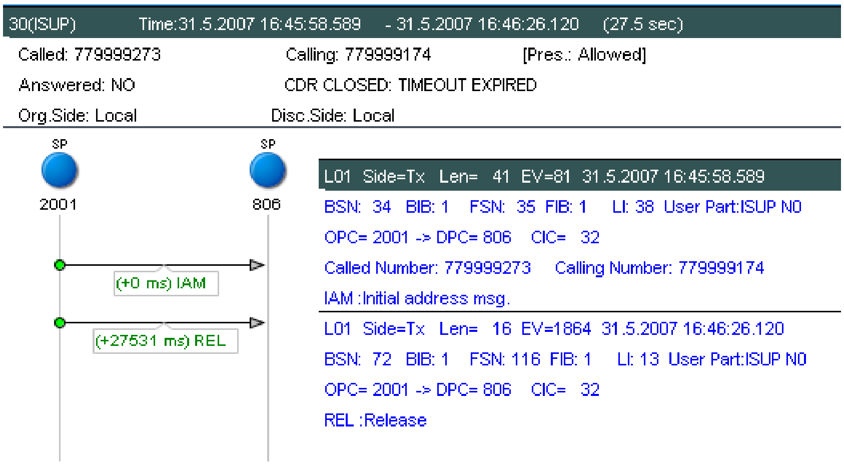

Fig. 6. Unconfigured CIC test (libss7)

Chan_ss7 has also detected incorrect message content in the test. However REL message was sent to a non-existent signalling point with point code “0” and the signalling link is subsequently tested by SLTM/SLTA messages. Chan_ss7 is not prepared to handle such situations. Finally the requested call is cancelled by REL message sent by protocol tester. The message exchange is displayed on figure 6.

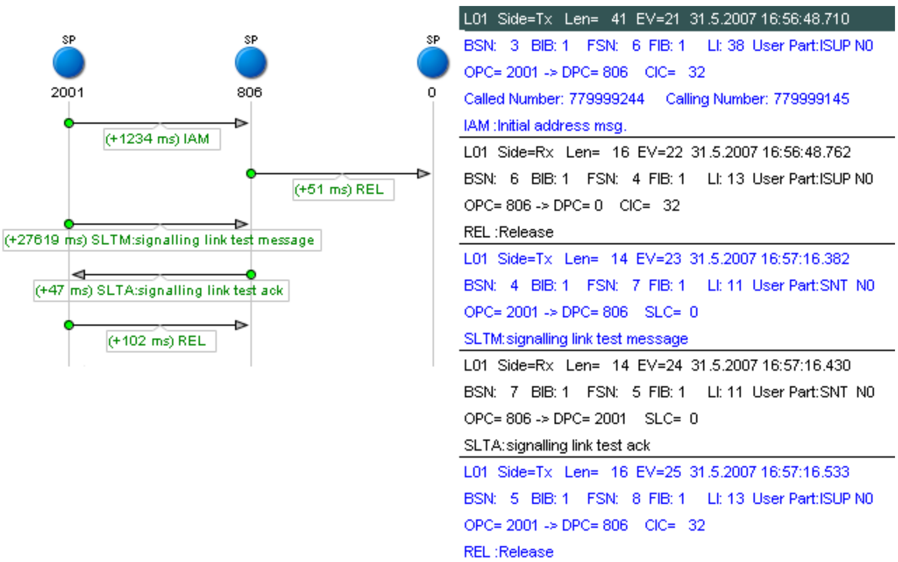

Fig. 6. Unconfigured CIC test (chan_ss7)

4. Conclusions

Asterisk already supports several variants of Signalling System #7 implementations. We have decided to test two open source solutions – SS7 channel driver (chan_ss7) and SS7 library (libss7).

Initialy we have tested the functionality of both solutions by interconnecting Asterisk PBX servers with with public PSTN exchange. We have exchanged calls from PSTN (PLMN) to IVR and to VoIP terminals. In opposite direction calls from VoIP clients via Asterisk were routed to PLMN exchange and terminated on mobile phones. All test calls were successful. Both SS7 channel driver and SS7 library processed most of MTP and ISUP messages and they proved to be capable of interoperability with PLMN Ericsson AXE exchange.

In the next step we have started ITU conformance tests to fully analyze SS7 solutions capabilities. We interconnected Tektronix K1297 protocol tester and Asterisk SS7. We found out that the protocol tester insists on voice channel initialization (group reset) to be done by the remote side. Unfortunatelly none of the SS7 implementations (chan_ss7, libss7) supported the feature. Therefore we couldn't complete the conformance tests.

Finally we ran performance tests of Asterisk with SS7 support to test the ability of load processing and unexpected situations handling. STT-MSA protocol analyzer was used to monitor the signalling message exchange. All calls have been successfully established during chan_ss7 test. We encountered only one difference from standard procedure were Asterisk sent a REL message few miliseconds before tester REL. Although this call was successfully closed by RLC message.

Asterisk with libss7 had problems with serving the offered load. Signalling message exchange differed from the standard, where Asterisk often sent REL message although it is the protocol tester would normally clear the call. And though the release cause in Asterisk messages stated “normal call clearing” more probable cause can be lack of resources for processing the call.

In the second part we used SST-MSA as call generator and call data record (CDR) collector. The performance tests proved SS7 channel driver stability and ability to process offered load. Whereas SS7 library failed to pass the performance test. A serious problem occurred in about one third of the test. Asterisk has stopped serving the calls. It was necessary to restart the application.

To proof unexpected situations handling we ran non-existent circuit identification code (CIC) tests. In case of libss7 Asterisk has reported incorrect message reception and the message was ignored. While tests for unconfigured circuit identification codes of SS7 channel driver proved its inability to handle these messages.

From the previous can be seen, that SS7 channel driver can be used as a operational SS7 solution for Asterisk PBX with some minor drawbacks in message handling. However SS7 library cannot be recommended for interconnection with SS7 nodes because of its inability to process offered load after certain period of time.

Abbreviations

PSTN – Public Switched Telephone Network

PLMN – Public Land Mobile Network

CDR – Call Detail Record

CIC – Circuit Identification Code

ISUP – Integrated Services User-part Protocol

MTP – Message Transfer Part of SS7 protocol stack

SS7 – Signalling System #7

ITU – International Telecommunications Union

References and Links

1. Rudinský J., Vozňák M., Růžička J., "Asterisk and SS7", Cesnet Technical Report, 12/2006
   http://www.cesnet.cz/doc/techzpravy/2006/asterisk-ss7/
2. Tomáš Wija, David Zukal and Miroslav Vozňák, “Asterisk a jeho použití”, CESNET Technical reports, December 2005,
   http://www.cesnet.cz/doc/techzpravy/2005/voip/asterisk.pdf
3. T. Russell, “Signalling System #7”, McGraw-Hill (2002), ISBN 0-07-138772-2
4. The Official Asterisk Community Page
   http://asterisk.org/
5. SIFIRA A/S home page
   http://www.sifira.dk/chan-ss7/
6. Libss7 Release Report
   http://www.venturevoip.com/news.php?rssid=1404