# SINAP/SIP Quick-Start Guide

Part Number: R8072-00

January 2005

# Notice



Manual Name: *SINAP/SIP Quick-Start Guide*

Part Number: R8072
Revision Number: 00
SINAP/SS7 Release Number: 1.0
Publication Date: January 2005

Stratus Technologies, Inc.
111 Powdermill Road
Maynard, Massachusetts 01754-3409

# Contents

# Figures

# Tables

# Preface

## Purpose

The *SINAP/SIP Quick-Start Guide* (R8072) describes how to install and configure Stratus Intelligent Network Applications Platform SIP (SINAP/SIP) software on an ftServer T Series system that runs the ft Linux® operating system. It describes how to test the installation and configure SINAP/SIP. It also describes how SINAP/SIP differs from the SIP stack and SIP User Agent (UA) toolkit from Hughes Software Systems® (HSS®) on which SINAP/SIP is based.

## Audience

The *SINAP/SIP Quick-Start Guide* (R8072) is intended for developers of SIP applications. Such applications establish, modify, and terminate multimedia sessions (conferences), such as Internet telephony calls. To perform these tasks, you must have a good working knowledge of the SIP protocol and the UNIX operating system and utilities.

## Revision Information

This is the first edition of this manual.

## Manual Organization

This manual is divided into the following chapters and appendixes. The manual also contains an index.

- Chapter 1, “Introduction”
- Chapter 2, “SINAP/SIP Software”
- Chapter 3, “SINAP/SIP Quick-Start Process”
- Appendix A, “SINAP/SIP User Agent and Stack Compliance”

## Notation Conventions

This manual uses the following notation conventions.

- `Monospace` represents text that would appear on your display screen (such as commands, functions, code fragments, and names of files and directories). For example:

    Do not forget to specify the following: `#include <iostream.h>`.

- `Monospace italic` represents terms to be replaced by literal values. In the following example, the user must replace the monospace-italic term with a literal value. For example:

  The `nmtr` program has the following syntax (where *`filename`* is the name of the file to be converted).

- **`Monospace bold`** represents user input in examples and figures that contain both user input and system output (which appears in monospace). For example:

  ```
  [root@ftlinux8 sinap_sip_master]# ls -l /usr/lib/libsip*
  ```

- *Italics* introduces or defines new terms. For example:

  The *Terminal Handler* accepts commands in Man-Machine Language (MML).

- **Boldface** emphasizes words in text. For example:

  You **must** create a link set before you provision its member links.

- When you need to press a key on the keyboard to perform an action, the keys are indicated in SMALL CAPS, for example:

  Press CTRL P to go to the next page or `ENTER` to exit the screen.

  NOTE

  There is an implied pressing of `ENTER` at the end of each command and menu response that you enter.

- The dollar sign (`$`) and the number sign (`#`) are standard default prompt signs that have a specific meaning at the UNIX prompt. Although a prompt is sometimes shown at the beginning of a command line as it would appear on the screen, you do not type it.
  - `$` indicates that you are logged in to a user account and are subject to certain access limitations.
  - `#` indicates that you are logged in to the system administrator account and have *superuser* access. Users of this account are referred to as `root`. The `#` prompt sign used in an example indicates the command can only be issued by `root`.

- When the full path name of a command appears in an example (for example, `/etc/fsck`), you must enter the command exactly as it appears.
- When the text that would appear on a single line on the screen is too long to fit in the space available on a line in the document, a slash (\) is used as a line-continuation character.

## Syntax Notation

This manual uses the format conventions for documenting commands as shown in the following example.

**rpm [-i | -install] [install-options]** ***package_file***

The following chart explains the notation used in command formats.

| Notation | Meaning |
|---|---|
| [ ] | Square brackets enclose command argument choices that are optional. For example:<br><br>**rpm [-i \| -install] [install-options]** ***package_file*** |
| \| | The vertical bar separates mutually exclusive arguments from which you choose one. For example:<br><br>**command [*arg1* \| *arg2*]**<br><br>You may use either `arg1` or `arg2` when you issue the command. |
| ... | Ellipsis indicates that you can have many arguments on a single command line. For example:<br><br>**command [*arg1, arg2, arg3,...*]** |
| # | In sample commands, the pound sign (#) is sometimes used for the shell command prompt. Your system prompt might differ. Although a prompt is sometimes shown at the beginning of a command line as it would appear on your screen, you do not type it. |
| **Note:** Dots and brackets are not literal characters; you should **not** type them. Any list or set of arguments can contain more than two elements. | |

## Related Manuals

Refer to the following Hughes System Software SIP manuals, available on the same CD-ROM as this guide, for related documentation:

- *SIP Stack API Reference Manual*
- *SIP Stack User Manual*
- *SIP User Agent Toolkit API Reference Manual*
- *SIP User Agent Toolkit User Manual*

Refer to the following ftServer T Series guides, available on the StrataDOC for the ft Linux platform Web site (http://stratadoc4ftlinux.stratus.com), for the following related documentation:

- The *Stratus ftServer T 30 Systems: Operation and Maintenance Guide* (R004L)
- The *Stratus ft Linux System Administrator's Guide* (R003L)

### A Note on the Contents of Stratus Manuals

Stratus manuals document subroutines and commands of the user interface. Any other commands and subroutines contained in the operating system are intended solely for use by Stratus personnel and are subject to change without warning.

### Accessing Documentation

The SINAP/SIP product documentation is provided on CD-ROM. You can request a documentation CD-ROM by calling the CAC (see "Commenting on the Documentation").

When requesting a documentation CD-ROM, please specify the product and platform documentation you desire, as there are several documentation CD-ROMs available.

### Commenting on the Documentation

To provide corrections and suggestions for improving this documentation, send email to Comments@stratus.com. If it is possible, please include the title and part number from the Notice page and the page numbers.

This information will assist Stratus Publications in making any needed changes to the documentation. Your assistance is most appreciated.

## Contacting the CAC

If you need assistance, contact your local systems engineer, or telephone the Stratus Customer Assistance Center (CAC) that services your area. If you cannot reach the center that services your area, contact the CAC in the United States.

The following table lists the CAC telephone numbers, all of which are available 24 x 7. For the most current list of CAC telephone numbers, see the following Web site: `http://www.stratus.com/support/cac`.

| **Customer Assistance Center (CAC)** | **Telephone Numbers** |
|---|---|
| North America, Central America, and South America | 800-221-6588 (toll-free within USA or Canada) |
| | 800-828-8513 (toll-free within USA or Canada) |
| | +1-978-461-7200 (Maynard, MA; for local and international direct) |
| | +1-602-852-3200 (Phoenix, AZ; for local and international direct) |
| Australia | 1800-025-046 (toll-free within Australia) |
| Belgium[1] | +32 2-512-63-70 (Dutch language) |
| | +32 2-512-77-06 (French language) |
| France | +33 (0) 1-41-20-37-08 |
| Germany | +49 (0) 6196-472518 |
| Hong Kong | 800-900-938 (toll-free within Hong Kong) |
| Italy | +39 02-467440-216 |
| Japan | 0120-725530 |
| Mexico | +52 55-5553-4792 |
| The Netherlands[†] | +31 (0) 346-582-112 |
| New Zealand | 0800-443-051 (toll-free within New Zealand) |
| People’s Republic of China | +86 139-010-39512 (Beijing) |
| | +86 21-63877700 (Shanghai) |
| Singapore | 1800-2727482 (toll-free within Singapore) |
| South Africa | +27 11-2675-709 |

| Customer Assistance Center (CAC) | Telephone Numbers |
|---|---|
| Spain | +34 91-383-4294 |
| United Kingdom | +44 (0) 1784-246056 |

1 After hours in Belgium, Denmark, Luxembourg, The Netherlands, Norway, and Sweden, you can call 00800-000-99999, a toll-free number. Your call will be directed to Phoenix Support Coordination.

NOTES

1. The plus sign (+) indicates that an international access code is required. The access code for international calls varies from country to country. In the United States, it is 011.
2. When you call from within the same country as the CAC office, be sure to include any necessary long distance or STD call prefix. If you use an international telephone number within the same country, you must replace the country code with the necessary prefix. For example, within the United States, callers dial 1-800-221-6588.
3. The telephone numbers in the preceding list are for CACs operated by Stratus. If you receive service from a distributor of Stratus products, contact your distributor for instructions about obtaining assistance.

# Chapter 1
# Introduction

Over the years, the Session Initiation Protocol (SIP) has evolved from a flexible but limited protocol, to a protocol in use across the Internet and at the core of the next generation of commercial telephony networks with their hybrid packet and circuit-switched networks. Much work has been done to enhance SIP to support Quality of Service (QoS) and other regulatory requirements.

SIP is an application-layer control protocol that can establish, modify, and terminate multimedia sessions (conferences), such as Internet telephony calls. SIP can also invite participants to already existing sessions, such as to multicast conferences. Media can be added to and removed from an existing session. SIP transparently supports name-mapping and redirection services to support personal mobility, enabling users to maintain a single, externally visible identifier, regardless of the network location of the device.

SIP supports the following aspects of establishing and terminating multimedia communications:

- User location: Determination of the end system to be used for communication
- User availability: Determination of the willingness of the called party to engage in communications
- User capabilities: Determination of the media and media parameters to be used
- Session setup: *Ringing* a party and establishing the session parameters for both the called and calling party
- Session management: Transferring and terminating sessions, modifying session parameters, and invoking services

  SIP provides a suite of security services, which includes denial-of-service prevention, authentication of both user-to-user and proxy-to-user, integrity protection, and encryption and privacy services.

# What is SINAP/SIP?

Stratus Intelligent Network Applications Platform SIP (SINAP/SIP) software uses the SIP stack and SIP User Agent (UA) toolkit sources from Hughes Software Systems (HSS). SINAP/SIP includes adaptations of this HSS SIP stack and UA toolkit sources to the ftServer T Series platform and provides users with easy-to-use APIs.

The SIP stack and UA toolkit are compliant with the latest RFC 3261 and RFC 2327 for SDP, as described in the Hughes SIP User Agent and Stack Compliance Statement, reprinted in Appendix A. SINAP/SIP runs on Stratus ftServer T Series systems, supporting 32-bit user applications on the 32-bit Stratus ft Linux operating system. The T Series system and ft Linux operating system provide additional robustness and a fail-safe SIP environment.

The SINAP/SIP software product fully resides in user space and provides easy-to-use APIs. These APIs encapsulate most of the basic SIP call- and session-setup primitives. You can use these APIs to code your service or applications to run on top of the SINAP/SIP stack. The SINAP/SIP stack and UA toolkit provide a set of dynamically linked libraries (DLLs). The source files and binary files of the SIP stack and of UA toolkit sample test programs are provided in the SINAP/SIP package as examples for your reference.

The SINAP/SIP product differs from the HSS SIP Stack and UA toolkit in the following respects:

- Delivery and packaging, described in “Delivery and Packaging” on page 2-1
- Feature support list, described in “Features” on page 2-1
- Licensing, described in “Licensing” on page 2-5
- Installation procedure, described in “Installing SINAP/SIP Software” on page 3-5
- Directory structure, described in “Installation Directories and Files” on page 3-8

# Overview of the Quick-Start Process

This SINAP/SIP Quick-Start Guide guides you through the installation, configuration, and testing of the SINAP/SIP product on an ftServer T Series system, which runs the ft Linux operating system. Use this guide with the other SIP manuals listed in the Related Manuals” section of the Preface.

# Acronyms

The following defines the acronyms used in this guide.

**Table 1-1. Acronyms**

| | |
|---|---|
| API | Application Programming Interface |
| HSS | Hughes Software Systems |
| hssUA | HSS User Agent sample test program |
| QoS | Quality of Service |
| RPM | RedHat Package Management |
| SCTP | Stream Control Transmission Protocol |
| SDP | Session Description Protocol |
| SINAP | Stratus Intelligent Network Applications Platform product family |
| SIP | Session Initiation Protocol |
| SSL | Secure Socket Layer |
| TCP | Transmission Control Protocol |
| TLS | Transport Layer Security |
| UA | User Agent |
| UAC | User Agent Client |
| UAS | User Agent Server |
| UDP | User Datagram Protocol |
| URI | Uniform Resource Identifier |

# Chapter 2
# SINAP/SIP Software

This chapter contains the following sections:



## Delivery and Packaging

SINAP/SIP is packaged as an RPM and is delivered in binary format. The RPM contains shared libraries and sample application programs (source files, binary files, and make files). Note that no source RPMs for the SIP stack or UA toolkit are provided with this distribution.

So that you do not need to rebuild your application whenever a new library patch is provided to fix bugs in the API routines, SINAP/SIP provides shared libraries. Two sets of the shared libraries are provided with the SINAP/SIP distribution: development and deployment. See "Selecting a SINAP/SIP Library Version" on page 3-18 for information on the proper use of these libraries.

## Features

SINAP/SIP is a collection of shared library objects, sample test programs, and utilities, such as `collect_sip_license_info` and `select_build_type`, which implement the following:

- The SIP stack, as defined by relevant RFCs
- The UA toolkit, which provides APIs to make and terminate calls. Call states are maintained by the UA toolkit along with callback mechanisms through which applications can invoke special treatments for various events or state transitions.

## SIP Configuration Options

The HSS SIP sources can be compiled with different options and compiler flags. Each option enables different features. Please refer to Chapter 7 of the *HSS SIP User Agent Toolkit User Manual* for details about options and compiler flags.

Table 2-1 describes the options and compiler flags used to build SINAP/SIP and the test programs, the configuration options set by the HSS SIP UA toolkit configuration script, and the compiler flags and `makefile` variables that were set to compile the SINAP/SIP stack and libraries.

**Table 2-1. SIP Configuration Options**

| Item # | Compile time Configuration Options | HSS SIP Default Settings | Stratus SINAP/SIP Settings | Compiler Flags or `Makefile` Variables[1] |
|---|---|---|---|---|
| | Can this package be compiled under Linux? | y[2] | y | `SDF_LINUX`<br>`SIP_LINUX` |
| | Do You want to make UATK (u) or MicroUA (m)? | u[3] | u | `SIP_BY_REFERENCE`<br>`SDF_TXN`<br>`SIP_TXN_LAYER`<br>`SIP_RPR`<br>`SIP_WARNING`<br>`SIP_TIMESTAMP`<br>`SIP_ISUP`<br>`SIP_MIME_PARSING`<br>`SIP_AUTHENTICATE`<br>`SIP_REPLACES`<br>`SIP_BADMESSAGE_PARSING` |
| 1 | Compiler Selection Options<br>1. gcc<br>2. g++<br>Select (1,2) | 1 | 1 | `CC=gcc COMPILER=gcc` |
| 2 | Do you want me to include all warnings flags for your compiler? | y | y | `GCC_WFLAGS e.g. -Wall -Wmissing-prototypes` |
| 3 | Generate shared objects? | n | y | |
| 4 | Enable Instant Messaging support? | y | y | `SDF_IM`<br>`SIP_IMPP` |
| 5 | Do you want to enable Call Transfer Service? | y | y | `SDF_REFER`<br>`SDF_SERVICE` |

**Table 2-1. SIP Configuration Options *(Continued)***

| Item # | Compile time Configuration Options | HSS SIP Default Settings | Stratus SINAP/SIP Settings | Compiler Flags or `Makefile` Variables[1] |
|---|---|---|---|---|
| 6 | Enable TEL support? | y | y | `SDF_TEL`<br>`SIP_TEL` |
| 7 | Enable session-timer support? | y | y | `SDF_SESSION_TIMER`<br>`SIP_SESSIONTIMER` |
| 8 | Enable Thread safety? | y | y | `SDF_THREAD_SAFE`<br>`SIP_THREAD_SAFE` |
| 9 | Enable Trace[4]in:<br>1. core stack<br>2. UA<br>3. both<br>4. neither<br>Select (1,2,3,4) | 4 | 4 (for deployment system) or<br>3 (for application development and debug stage) | These flags are undefined if 4 is selected; else defined if 3.<br>`SDF_TRACE`<br>`SIP_TRACE` |
| 10 | Enable Statistics[4] in:<br>1. core stack<br>2. UA<br>3. both<br>4. neither Select (1,2,3,4) | 4 | 4 (for deployment system) or<br>3 (for application development and debug stage) | These flags are undefined if 4 is selected; else defined if 3.<br>`SDF_STATISTICS`<br>`SIP_STATISTICS` |
| 11 | Enable Function Entry/Exit Debug in UA Toolkit? | n | n | These flags are undefined.<br>`SDF_DEBUG`<br>`SIP_FNDEBUG` |
| 12 | Enable Error in:<br>1. core stack<br>2. UA<br>3. both<br>4. neither<br>Select (1,2,3,4) | 4 | 3 | These flags are undefined if 4 is selected; else defined if 3.<br>`SDF_ERROR`<br>`SIP_ERROR` |
| 13 | Enable Call Flow display in UA Toolkit? | n | n | `SDF_CALLFLOW` undefined |
| 14 | Enable PreAllocated Buffers? | n | y[5] | This flag is undefined if “n”; else defined if “y”.<br>`SDF_USE_PREALLOCBUFFERS` |
| 15 | Enable parameter validation checks in UA Toolkit APIs? | y | y | `DF_PARAMVALIDATION` |

**Table 2-1. SIP Configuration Options *(Continued)***

| Item # | Compile time Configuration Options | HSS SIP Default Settings | Stratus SINAP/SIP Settings | Compiler Flags or `Makefile` Variables[1] |
|---|---|---|---|---|
| 16 | Do you want to Enable HA functionality? | n | y | `SDF_HA` |
| 17 | Do you want to support IPV6 addresses? | n | n[6] | `SDF_IPV6` undefined |
| 18 | Do you want to support QOS? | n | y | `SDF_QOS` |
| 19 | Do you want to support Media Group? | n | y | `SDF_MEDIAGROUP` |
| 20 | Do you want to Enable Session Change? | n | y | `SDF_SESSIONCHANGE` |
| 21 | Do you want to Enable TLS support? | n | y | `SDF_TLS`[7] |
| 22 | Do you want to build hssUA (h) SUPEr (s) simUA? (U) By Default hssUA is built | h | h[8] | If s (SUPEr) is selected, `SDF_SUPER` is defined. |

1 Defined at the SIP UA toolkit or stack layers, depending on configuration settings.
2 Set to the Linux platform automatically by the UA toolkit configuration script, which checks whether the `${STACK_DIR}/makefiles/packages/PLATFORM_LINUX` directory exists.
3 Set to u (UATK) automatically by UA toolkit configure script.
4 See “Selecting a SINAP/SIP Library Version” on page 3-18 for a description of the different settings for these options in the development and deployment libraries.
5 As specified in the *HSS SIP User Agent Toolkit User Manual* section 7.2.15, choose **y** to optimize UA toolkit performance. See “Buffer Allocation” on page 2-5 for more information.
6 Although the HSS SIP stack supports running over either IPv4 or IPv6 protocols, IPv6 is not supported in this release of SINAP/SIP because IPv6 is not supported in ft Linux Release 2.2.
7 As described in the *HSS SIP User Agent Toolkit User Manual* 25.6 (TLS - FAQ), OpenSSL libraries and header files need to be installed before you can build HSS SIP source code with the TLS feature enabled; otherwise, compilation errors will occur. See “Installing OpenSSL” on page 3-2.
8 Only the hssUA SIP UA toolkit sample test program is included in the SINAP/SIP package. Note that core SIP stack test programs, such as `siptest` and `sendmessage`, are also built along with the SIP UA toolkit hssUA test program, which are packaged in SINAP/SIP Release 1.0.

## Buffer Allocation

When the `SDF_USE_PREALLOCBUFFERS` flag is enabled in your SIP UA toolkit application, specify the preallocated buffer size and number of buffers parameters, `dInitParams.dPreAllocBufSize` and `dInitParams.dNumPreAllocMsgBuffers`, before calling the UA toolkit initialization routine, `sdf_ivk_uaInitToolkit()`.

The documentation provided by HSS does not describe how to tune these parameters. However, HSS has provided the following recommendation:

- The number of direct preallocated message buffers (`dInitParams.dNumPreAllocMsgBuffers`) is equal to the result of:

  32 * *numbufs*.

  32 is used because when an INVITE message is sent, the INVITE message uses one buffer for up to 32 seconds to allow for retransmission time out.

- The value of *numbufs* is equal to the number of SIP messages (calls) that can transmit in one sec.

  Usually, one call has three SIP messages, so:

  *numbufs* = 3 * <*number-of-calls-per-second (CPS)*>

So the number of direct preallocated message buffers (`dInitParams.dNumPreAllocMsgBuffers`) is equal to the result of 32 * (3 * CPS). For example, for 300 CPS, the result should be greater than or equal to 32*(3*300).

The preallocated buffer size depends on the maximum SIP message size expected. For example, the hssUA SIP UA toolkit sample program sets the `dInitParams.dPreAllocBufSize` to 3072 bytes.

## Licensing

SINAP/SIP and other SINAP products use FLEXlm® license management software to manage their licenses. After installing SINAP/SIP, obtain a valid license from the Stratus Customer Assistance Center before starting to configure and run SINAP/SIP. See “Obtaining and Maintaining Licenses” on page 3-14 for more information.

The SINAP/SIP product’s default license file location and file name are the same as those on the SINAP product (`/etc/sinap_license/sinap.lic`). Although initially the SIP product does not require SINAP to run, using the same file makes it easy to integrate the license with other SINAP family products later on. Maintaining all features’ licenses in one file also makes it easier for both the vendor and the end user.

### License Feature Line

The SINAP/SIP product has an entry in the (`/etc/sinap_license/sinap.lic`) license file. The following are examples of the SIP entry, which licenses only the SIP stack:

```
INCREMENT SIP dncp 1.0 permanent uncounted \
VENDOR_STRING=SIPLEVEL=STACK HOSTID=31422 SIGN=0
```

This following entry licenses both the SIP UA toolkit and the SIP stack:

```
INCREMENT SIP dncp 1.0 permanent uncounted \
VENDOR_STRING=SIPLEVEL=UA HOSTID=31422 SIGN=0
```

These entries license SIP on the host with host ID 31422. The `VENDOR_STRING` field `SIPLEVEL` is required; otherwise an error is reported.

# Chapter 3
# SINAP/SIP Quick-Start Process

This chapter describes how to install, configure, and test the SINAP/SIP product on an ftServer T Series system. In each section of this chapter, a short procedure or inspection is described that you can use to confirm if each major step in the process was successful. Some troubleshooting hints are provided in case of difficulty.

This chapter contains the following sections, which are steps in the installation, configuration and testing process:



NOTE

If the operating system has been installed on the system at the factory, several of these steps might have already been performed.

## Preparing the Hardware

To install and set up an ftServer T Series system, follow the instructions detailed in the *Stratus ftServer T 30 Systems: Installation Guide* (R002L) and the *Stratus ft Linux System Administrator’s Guide* (R003L). Before you install any additional PCI adapters, see “Verifying the Operating System” on page 3-2.

# Verifying the Operating System

On an ftServer T Series system, the operating system is factory installed, so it is unlikely you will need to install it yourself. However, should the need arise, see the *Stratus ft Linux System Administrator's Guide* (R003L). Verify that the operating system is properly installed by checking the version of the operating system. To do this, enter

```
uname -srp
```

The following output indicates a successful installation:

```
Linux 2.4.18 i686
```

# Installing Other Packages

Before you install the SINAP/SIP software, install OpenSSL and Ethereal. See the following:

- "Installing OpenSSL" on page 3-2
- "Installing Ethereal" on page 3-5

## Installing OpenSSL

As described in the *HSS SIP UA Toolkit User Manual* section 7.2.22, the SIP UA toolkit requires the installation of OpenSSL libraries and header files with the TLS feature enabled. If your SIP UA toolkit application will use the TLS feature, then, prior to the compilation of the UA toolkit application, install the OpenSSL libraries and header files as described in the *HSS SIP User Agent Toolkit User Manual*, section 25.6, question # 2. Since the SIP UA toolkit code references OpenSSL libraries and header files at the default `/usr/local/ssl` directory, install the OpenSSL libraries at this location.

If you do not install OpenSSL, a SIP UA toolkit application that uses the TLS feature will fail to compile.

Note that the *HSS SIP UA Toolkit User Manual* only has instructions for installing OpenSSL libraries and randomness source on Solaris®, Windows®, and VxWorks® operating systems. HSS advises that the OpenSSL installation instructions for the Solaris operating system are applicable to a Linux platform.

Also, section 25.6 states that the "UATK has been tested with OpenSSL version 0.9.7b and the same can be installed and used with UATK." However, there are newer versions available from www.openssl.org to address security advisories and bugs uncovered after the release of 0.9.7b version.

To avoid any compilation errors, install the latest OpenSSL package. Stratus has tested SINAP/SIP with the OpenSSL 0.9.7d release.

Download the RPM file from www.openssl.org and install it, following the instructions provided in the README file of the package and shown in the following.

```
[root@ftlinux9 src]# pwd
/usr/src

[root@ftlinux9 src]# ls
debug openssl-0.9.7d.tar.gz redhat

[root@ftlinux9 src]# ls -l
total 14028
drwxr-xr-x    2 root     root         4096 Jan 24  2003 debug
-rw-r--r--    1 root     root      2798433 Jan 14 11:50
openssl-0.9.7d.tar.gz
drwxr-xr-x    7 root     root         4096 Jan 12 12:18 redhat

[root@ftlinux9 src]# gzip -d openssl-0.9.7d.tar.gz

[root@ftlinux9 src]# tar xvf openssl-0.9.7d.tar
.
.
.
Configured for linux-elf.

[root@ftlinux9 openssl-0.9.7d]# make
making all in crypto...
.
.
.
make[1]: Leaving directory `/usr/src/openssl-0.9.7d/tools'

[root@ftlinux9 openssl-0.9.7d]# make install
.
.
.
chmod 644 /usr/local/ssl/lib/pkgconfig/openssl.pc
[root@ftlinux9 openssl-0.9.7d]#

[root@ftlinux9 openssl-0.9.7d]# ls -l /usr/src
total 14304
drwxr-xr-x    2 root     root         4096 Jan 24  2003 debug
-rw-r--r--    1 root     root        14566 Jan 14 11:54
log.config-openssl-0.9.7d
-rw-r--r--    1 root     root        53725 Jan 14 12:05
log.make-install-openssl-0.9.7d
-rw-r--r--    1 root     root       197640 Jan 14 11:57
log.make-openssl-0.9.7d
```

```
drwxr-xr-x   20 root     root         4096 Jan 14 11:57 openssl-0.9.7d
-rw-r--r--    1 root    root    14330368 Jan 14 11:45 openssl-0.9.7d.tar
drwxr-xr-x    7 root     root         4096 Jan 12 12:18 redhat

[root@ftlinux9 openssl-0.9.7d]# ls -l /usr/local
total 40
drwxr-xr-x    2 root     root         4096 Jan 24  2003 bin
drwxr-xr-x    2 root     root         4096 Jan 24  2003 etc
drwxr-xr-x    2 root     root         4096 Jan 24  2003 games
drwxr-xr-x    2 root     root         4096 Jan 24  2003 include
drwxr-xr-x    2 root     root         4096 Jan 24  2003 lib
drwxr-xr-x    2 root     root         4096 Jan 24  2003 libexec
drwxr-xr-x    2 root     root         4096 Jan 24  2003 sbin
drwxr-xr-x    4 root     root         4096 Jan 12 08:39 share
drwxr-xr-x    2 root     root         4096 Jan 24  2003 src
drwxr-xr-x    9 root     root         4096 Jan 14 12:05 ssl

[root@ftlinux9 openssl-0.9.7d]# ls -l /usr/local/ssl
total 36
drwxr-xr-x    2 root     root         4096 Jan 14 12:05 bin
drwxr-xr-x    2 root     root         4096 Jan 14 12:05 certs
drwxr-xr-x    3 root     root         4096 Jan 14 12:05 include
drwxr-xr-x    3 root     root         4096 Jan 14 12:05 lib
drwxr-xr-x    6 root     root         4096 Jan 14 12:04 man
drwxr-xr-x    2 root     root         4096 Jan 14 12:05 misc
-rw-r--r--    1 root     root         7782 Jan 14 12:05 openssl.cnf
drwxr-xr-x    2 root     root         4096 Jan 14 12:05 private

[root@ftlinux9 openssl-0.9.7d]#
```

## Installing Ethereal

To capture, filter, and debug SIP messages using the Ethereal utility, install the following RPMs in the order shown after you install the ft Linux operating system on an ftServer T Series system.

1. `gtk+-1.2.10-25.i386.rpm`
2. `libpcap-0.7.2-1.i386.rpm`
3. `ethereal-0.10.0a-0.90.1.i386.rpm`
4. `ethereal-gnome-0.10.0a-0.90.1.i386.rpm`

> NOTE
>
> Newer versions of Ethereal exist. However, when tested, they failed to install on an ft Linux system due to dependency checks. The 0.10.0a-0.90.1 version of Ethereal RPMs are the latest that can be installed on an ft Linux system without the need to specify the `--nodeps` option for the `rpm -i` command.

To install Ethereal, enter the commands shown as follows:

```
[root@ftlinux1 i386]# rpm -i gtk+-1.2.10-25.i386.rpm
warning: gtk+-1.2.10-25.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e

[root@ftlinux1 i386]# rpm -i libpcap-0.7.2-1.i386.rpm
warning: libpcap-0.7.2-1.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e

[root@ftlinux1 i386]# rpm -i ethereal-0.10.0a-0.90.1.i386.rpm
warning: ethereal-0.10.0a-0.90.1.i386.rpm: V3 DSA signature: NOKEY, key ID
db42a60e

[root@ftlinux1 i386]# rpm -i ethereal-gnome-0.10.0a-0.90.1.i386.rpm
warning: ethereal-gnome-0.10.0a-0.90.1.i386.rpm: V3 DSA signature: NOKEY,
key ID db42a60e
```

# Installing SINAP/SIP Software

The following sections describe how to install the SINAP/SIP software:

- "Preparing to Install SINAP/SIP Software" on page 3-6
- "Installing the SINAP/SIP Package" on page 3-6
- "Collecting System Information" on page 3-14

## Preparing to Install SINAP/SIP Software

Before you install the SINAP/SIP software on a new system, create a group named **sinap** in the `/etc/group` file.

1. Log in as root.
2. Create the sinap group with the following command:

   **`groupadd sinap`**

## Installing the SINAP/SIP Package

This section provides instructions for installing the SINAP/SIP software from a CD-ROM.

1. Log in as root.
2. To make sure that the `/mnt/cdrom` directory exists, type the following command:

   **`ls /mnt`**

   If no `/mnt/cdrom` directory exists, type the following command to create the directory:

   **`mkdir /mnt/cdrom`**

3. Insert the SINAP/SIP CD-ROM into the CD-ROM drive.
4. Type the following command:

   **`mount /dev/cdrom/mnt/cdrom`**

5. Type the following command:

   **`cd /mnt/cdrom`**

6. Type the following command:

   **`./sinap_sip_install`**

Figure 3-1 shows a sample SINAP/SIP installation on the ft Linux operating system.

```
# ls /mnt
cdrom cdrom1
# mount /dev/cdrom /mnt/cdrom
mount: block device /dev/cdrom is write-protected, mounting
read-only
# cd /mnt/cdrom
# ./sinap_sip_install

PLEASE READ THE FOLLOWING STATEMENT CAREFULLY.

        STRATUS TECHNOLOGIES
END-USER LICENSE AGREEMENT FOR STRATUS SINAP SOFTWARE
.
.
<license text>
.
.
Press ENTER to continue:
.
.
Do you accept the above statement? (y/n) y

Starting SINAP/SIP installation...
Preparing...
########################################### [100%]
1:sinap_sip
########################################### [100%]
```

**Figure 3-1. Sample SINAP/SIP Installation on an ftServer T Series System**

## Verifying the Installation

You can verify that the SINAP/SIP software was correctly installed by issuing the following command:

```
# rpm -q sinap_sip
```

If a line similar to the following is displayed, the installation was successful:

```
sinap_sip-1.0.0.0_03BE110304-1
```

This line displays the result for an installation of the SINAP/SIP Release 1.0.0.0 software.

# Installation Directories and Files

The following is a list of directories and files created after an installation of SINAP/SIP software:

1. `/home/sinap_sip_master` (SINAP/SIP installation master directory)

`/home/sinap_sip_master/README` (SINAP/SIP README file)
`/home/sinap_sip_master/select_build_type` (Utility to set up development or deployment version of SIP libraries and `makefile`'s)

2. `/home/sinap_sip_master/etc/sinap_license` (SINAP/SIP license management tools)

```
/home/sinap_sip_master/etc/sinap_license/collect_sip_license_info
/home/sinap_sip_master/etc/sinap_license/dncp
/home/sinap_sip_master/etc/sinap_license/lmdown
/home/sinap_sip_master/etc/sinap_license/lmgrd
/home/sinap_sip_master/etc/sinap_license/lmhostid
/home/sinap_sip_master/etc/sinap_license/lmstat
/home/sinap_sip_master/etc/sinap_license/lmstrip
```

3. `/home/sinap_sip_master/ua` (Top level SIP installation directory)

`/home/sinap_sip_master/ua/MakeVars` (Compilation flags)

4. `/home/sinap_sip_master/ua/sip_stack` (SIP stack directory)

`/home/sinap_sip_master/ua/sip_stack/MakeVars.deploy` (Deployment version)
`/home/sinap_sip_master/ua/sip_stack/MakeVars.devel` (Development version)
`/home/sinap_sip_master/ua/sip_stack/lib` (SIP Stack libraries)
`/home/sinap_sip_master/ua/sip_stack/lib/deploy` (Deployment version)
`/home/sinap_sip_master/ua/sip_stack/lib/deploy/libsipapi.so`
`/home/sinap_sip_master/ua/sip_stack/lib/deploy/libsipcore.so`

`/home/sinap_sip_master/ua/sip_stack/lib/deploy/libsipserialize.so`
`/home/sinap_sip_master/ua/sip_stack/lib/devel` (Development version)
`/home/sinap_sip_master/ua/sip_stack/lib/devel/libsipapi.so`
`/home/sinap_sip_master/ua/sip_stack/lib/devel/libsipcore.so`
`/home/sinap_sip_master/ua/sip_stack/lib/devel/libsipserialize.so`

`/home/sinap_sip_master/ua/sip_stack/obj` (Object files created from compiling SIP Stack sample test programs)
`/home/sinap_sip_master/ua/sip_stack/obj/parser`
`/home/sinap_sip_master/ua/sip_stack/obj/parser/sdpdecode.o`
`/home/sinap_sip_master/ua/sip_stack/obj/parser/sipdecode.o`
`/home/sinap_sip_master/ua/sip_stack/obj/parser/sipformmessage.o`
`/home/sinap_sip_master/ua/sip_stack/obj/parser/sipparserclone.o`
`/home/sinap_sip_master/ua/sip_stack/obj/parser/sipsendmessage.o`
`/home/sinap_sip_master/ua/sip_stack/obj/parser/telerror.o`
`/home/sinap_sip_master/ua/sip_stack/obj/parser/txndecode.o`
`/home/sinap_sip_master/ua/sip_stack/obj/parser/txndecodeintrnl.o`
`/home/sinap_sip_master/ua/sip_stack/obj/parser/txnmidway.o`

`/home/sinap_sip_master/ua/sip_stack/stack_headers` (SIP Stack header files)
`/home/sinap_sip_master/ua/sip_stack/stack_headers/bcpt.h`
`/home/sinap_sip_master/ua/sip_stack/stack_headers/txnmidway.h`

`/home/sinap_sip_master/ua/sip_stack/test` (SIP Stack sample test programs)
`/home/sinap_sip_master/ua/sip_stack/test/Makefile`
`/home/sinap_sip_master/ua/sip_stack/test/bin`
`/home/sinap_sip_master/ua/sip_stack/test/bin/sendmessage`
`/home/sinap_sip_master/ua/sip_stack/test/bin/siptest`

`/home/sinap_sip_master/ua/sip_stack/test/parser` (SIP Stack sample program source files)
`/home/sinap_sip_master/ua/sip_stack/test/parser/README.TXT`
`/home/sinap_sip_master/ua/sip_stack/test/parser/encodetime.c`
`/home/sinap_sip_master/ua/sip_stack/test/parser/parsetime.c`
`/home/sinap_sip_master/ua/sip_stack/test/parser/sendmessage.c`
`/home/sinap_sip_master/ua/sip_stack/test/parser/siptest.c`
`/home/sinap_sip_master/ua/sip_stack/test/parser/siptxntest.h`
`/home/sinap_sip_master/ua/sip_stack/test/parser/siptxntimer.c`

5. `/home/sinap_sip_master/ua/toolkit` (SIP UA Toolkit directory)

`/home/sinap_sip_master/ua/toolkit/MakeVars.deploy` (Deployment version)
`/home/sinap_sip_master/ua/toolkit/MakeVars.devel` (Development version)
`/home/sinap_sip_master/ua/toolkit/hssua` (SIP UATK sample test program directory)
`/home/sinap_sip_master/ua/toolkit/hssua/Makefile`
`/home/sinap_sip_master/ua/toolkit/hssua/authPasswd.txt`
`/home/sinap_sip_master/ua/toolkit/hssua/bin`
`/home/sinap_sip_master/ua/toolkit/hssua/bin/hssUA` (SIP UATK sample test program)

/home/sinap_sip_master/ua/toolkit/hssua/cert (TLS certificates directory)
/home/sinap_sip_master/ua/toolkit/hssua/cert/cmds
/home/sinap_sip_master/ua/toolkit/hssua/cert/cmds/chain_dsa_pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/cmds/chain_rsa_pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/cmds/selfsign_dsa_pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/cmds/selfsign_rsa_pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/cnf
/home/sinap_sip_master/ua/toolkit/hssua/cert/cnf/client.cnf
/home/sinap_sip_master/ua/toolkit/hssua/cert/cnf/root.cnf
/home/sinap_sip_master/ua/toolkit/hssua/cert/cnf/server.cnf
/home/sinap_sip_master/ua/toolkit/hssua/cert/cnf/serverCA.cnf
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/99570c64.0
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/9fe7d60a.0
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/clientcert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/clientkey.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/clientreq.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/dsaparam.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/ea558e7e.0
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/f345392a.0
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/rootcert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/rootcert.srl
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/rootkey.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/rootreq.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/serverCAcert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/serverCAcert.srl
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/serverCAkey.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/serverCAreq.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/servercert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/serverkey.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_dsa/serverreq.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/29cd3d6f.0
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/43ee835e.0
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/99570c64.0
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/99d8c992.0
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/clientcert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/clientkey.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/clientreq.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/dhparam.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/root.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/root.srl
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/rootcert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/rootcert.srl
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/rootkey.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/rootreq.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/serverCA.pem

/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/serverCA.srl
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/serverCAcert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/serverCAkey.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/serverCAreq.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/servercert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/serverkey.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/cert_rsa/serverreq.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/selfsign_dsa
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/selfsign_dsa/servercert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/selfsign_dsa/serverkey.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/selfsign_rsa
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/selfsign_rsa/servercert.pem
/home/sinap_sip_master/ua/toolkit/hssua/cert/pem/selfsign_rsa/serverkey.pem

/home/sinap_sip_master/ua/toolkit/hssua/h (SIP UATK header files for hssUA)
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_common.h
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_console.h
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_eventproc.h
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_ha.h
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_menu.h
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_network.h
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_struct.h
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_timer.h
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_uac.h
/home/sinap_sip_master/ua/toolkit/hssua/h/hssua_util.h
/home/sinap_sip_master/ua/toolkit/hssua/h/tcputil.h
/home/sinap_sip_master/ua/toolkit/hssua/h/timer.h

/home/sinap_sip_master/ua/toolkit/hssua/obj (SIP UATK object files from compiling hssUA)
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_commoncallbacks.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_console.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_eventproc.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_main.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_network.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_networkcallbacks.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_timer.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_uac.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_uaccallbacks.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_uascallbacks.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/hssua_util.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/stack_callbacks.o
/home/sinap_sip_master/ua/toolkit/hssua/obj/tcputil.o

/home/sinap_sip_master/ua/toolkit/hssua/profile-0.xml.ua1 (hssUA profile 0 of user agent 1)
/home/sinap_sip_master/ua/toolkit/hssua/profile-0.xml.ua2 (hssUA profile 0 of user agent 2)

/home/sinap_sip_master/ua/toolkit/hssua/profile-1.xml.ua1 (hssUA profile 1 of user agent 1)
/home/sinap_sip_master/ua/toolkit/hssua/profile-1.xml.ua2 (hssUA profile 1 of user agent 2)

/home/sinap_sip_master/ua/toolkit/hssua/src (SIP UATK hssUA sample program's source files)
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_commoncallbacks.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_console.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_eventproc.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_haactive.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_hastandby.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_main.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_network.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_networkcallbacks.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_timer.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_uac.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_uaccallbacks.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_uascallbacks.c
/home/sinap_sip_master/ua/toolkit/hssua/src/hssua_util.c
/home/sinap_sip_master/ua/toolkit/hssua/src/stack_callbacks.c
/home/sinap_sip_master/ua/toolkit/hssua/src/tcputil.c

/home/sinap_sip_master/ua/toolkit/lib (SIP UATK libraries)
/home/sinap_sip_master/ua/toolkit/lib/deploy (Deployment version)
/home/sinap_sip_master/ua/toolkit/lib/deploy/libua.so
/home/sinap_sip_master/ua/toolkit/lib/deploy/libuact.so
/home/sinap_sip_master/ua/toolkit/lib/deploy/libuaport.so
/home/sinap_sip_master/ua/toolkit/lib/deploy/libuaservice.so

/home/sinap_sip_master/ua/toolkit/lib/devel (Development version)
/home/sinap_sip_master/ua/toolkit/lib/devel/libua.so
/home/sinap_sip_master/ua/toolkit/lib/devel/libuact.so
/home/sinap_sip_master/ua/toolkit/lib/devel/libuaport.so
/home/sinap_sip_master/ua/toolkit/lib/devel/libuaservice.so

/home/sinap_sip_master/ua/toolkit/uatk_headers (SIP UATK header files)
/home/sinap_sip_master/ua/toolkit/uatk_headers/ALheader.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/basic.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/digcalc.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/global.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/md5.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_accessor.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_authorization.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_basictypes.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_callapis.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_callbacks.h

```
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_common.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_configs.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_ct.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_ctfree.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_ctinit.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_ctinternal.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_ctstruct.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_debug.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_free.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_haactive.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_haapi.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_haapiint.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_hacommon.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_hadeserstruct.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_haint.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_haser.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_haserint.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_haserstruct.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_hash.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_hastandby.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_init.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_internal.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_list.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_mempool.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_network.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_portlayer.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_reqresp.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_sdp.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_sdpaccessor.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_sdpfree.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_sdpinit.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_sdpinternal.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_sdpstruct.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_service.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_servicecallbacks.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_servicecallobjectassoc.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_servicefree.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_serviceinit.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_serviceinternal.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_servicestruct.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_sessTimer.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_statistics.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_struct.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_tables.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_trace.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/sdf_txn.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/uatkCppWrappers.h
```

`/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config.h.deploy`
(Deployment version)
`/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config.h.devel`
(Development version)
`/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_headers.h`

# Obtaining and Maintaining Licenses

This section describes how to acquire and maintain the SINAP/SIP product licenses.

## Collecting System Information

After the SINAP/SIP software is installed on the target machine, invoke the `collect_sip_license_info` tool to collect the system information required for license generation. The system information collected includes platform type, number of CPUs, and the machine ID (HostID).

1. Log in as root.
2. Collect licensing information. To do this, type the following:

   **`/etc/sinap_license/collect_sip_license_info >`** ***`license_info`***

   *`license_info`* is a file to store the information. (You can specify any name and location for the file.)

3. Send the *`license_info`* file to Stratus customer service to generate a license. Because this is an ASCII text file, you can send it through email or by FTP, or you can communicate its content through a FAX or verbally through the telephone. See “Contacting the CAC” on page x.

   The following is sample output for an ftServer T Series system:

```
[root@ftlinux9 sinap_license]# /etc/sinap_license/collect_sip_license_info
SINAP/SIP License Information Collecting Utility
Binary Version: Rel 1.0.0.0_03BE 11/03/04
================================================
Platform Type: Intel(R)Xeon(TM)CPU2.40GHz
Number of CPU(s): 4
Serial Number: N/A
FTSERVER: ftServer_3300
FTSERIAL: 322017
FTKERNEL: 2.2.0.1-GA1338
FTMACID: 0004FC011AF7 ================================================
```

4. After the Stratus CAC receives the license information, they generate a license and send the license information back to you. At this point, follow the procedure in the section“The License File” on page 3-15.

NOTE

A SINAP/SIP license is normally created as *node-locked*. It authorizes you to use SINAP/SIP software on a specific target machine only. If the configuration information changes, you might need to acquire a new license so that the SINAP/SIP software continues working. Therefore, check the *license_info* information before sending it to Stratus customer service.

## The License File

The `sinap.lic` license file is your key to use the SINAP/SIP software. SINAP/SIP software checks the content of this file to determine if the SIP feature is available to you. The `sinap.lic` file can include license authorization for multiple products in the SINAP family. For example, SINAP/SIP software and SINAP/SS7 software are separately licensed, but both license authorizations are stored in the `sinap.lic` file.

For the SINAP/SIP software to gain access to the license file, do the following:

1. Create a file called `sinap.lic` in the license directory and copy the license information into that file. The default license directory is `/etc/sinap_license`. (See the section “Maintaining Licenses” on page 3-16 for details about the license information to copy into the `sinap.lic` file.)

   **CAUTION**

   To ensure that FLEXlm can manage the licenses properly, never put multiple `*.lic` files that contain features lines for the same feature in the license directory.

2. If you are running other software on the target system that uses a FLEXlm license file, you can copy the `sinap.lic` file to a centralized directory that contains files provided by other vendors. In this case, set the `LM_LICENSE_FILE` environment variable to point SINAP to this new license file location. Please note that if you do this, you need to set the `LM_LICENSE_FILE` environment variable for root (used to configure SINAP) as well as the SINAP user account for each node (used to start SINAP). Here is an example of how to set the environment variable in the C shell:

   **`setenv LM_LICENSE_FILE`** ***`my_own_license_path`*** **`/sinap.lic`**

3. After the license file is put in place, configure and start the SIP sample test program.

## Maintaining Licenses

The license file is an ASCII text file. You can get most of the license information from the file and determine what kind of license type (permanent or time-limited) you have for a particular software feature. Both license types require you to keep the license file accessible by the SINAP/SIP software at all times. The license file should contain an entry for each licensed product, for example, a string for the SINAP/SIP product and a second entry for the SINAP/SS7 product.

- If the license type is permanent, no action is required from you to keep the license up-to-date. A permanent license is always node-locked, so the feature will not work if you change the host configuration, such as increase the number of CPUs or change the machine ID. The following is a sample permanent license string:

```
INCREMENT SIP dncp 1.0 permanent uncounted \
VENDOR_STRING=FTSERVER=ftServer_3300;FTSERIAL=310821;FTKERNEL=2.x;SIPLEVEL=UA \
HOSTID=<id_number> SIGN=<encrypted entry>
```

- If the license type is time-limited, the expiration date is written in each feature line of the license file. Daily warnings will be reported to the system log (`/var/log/messages` for the Stratus ft Linux operating system) 30 days before expiration, and the feature will eventually stop working when its expiration date has passed. It is your responsibility to renew the license or acquire a permanent license before it expires if you decide to continue using the feature. See "Online License Update" on page 3-17 for details.

The following is a sample time-limited license string:

```
INCREMENT SIP dncp 1.0 10-mar-2005 uncounted \
VENDOR_STRING=FTSERVER=ftServer_3300;FTSERIAL=310821;FTKERNEL=2.x;SIPLEVEL=UA \
HOSTID=<id_number> SIGN=<encrypted entry>
```

Table 3-1 describes the license file fields.

**Table 3-1. License File Fields**

| Fields | Description |
|---|---|
| `<licensed product>` | Specifies the licensed SIP product and release; in this example, SINAP/SIP Release 1.0. |
| `<license type>` | Specifies the type of license, either permanent or time-limited. |
| `FTSERVER=<model>` | Specifies the licensed ftServer platform type. |
| `FTSERIAL=<number>` | Specifies the serial ID for the system. |

**Table 3-1. License File Fields *(Continued)***

| Fields | Description |
|---|---|
| `FTKERNEL=<number>` | Specifies the ft Linux operating system version ID for the system. |
| `SIPLEVEL=<level>` | Specifies the SINAP/SIP product's interface level (STACK or UA)[1]. |
| `HOSTID=<number>` | Specifies the host ID for the system. |

1 If `SIPLEVEL=STACK` is specified, then only the SIP libraries of the stack level are licensed to be referenced, not the libraries of the UA Toolkit. If `SIPLEVEL=UA` is specified, then both the SIP Stack and UA toolkit libraries are licensed to be referenced by the SIP application program.

The SINAP/SIP license is always bound to a particular *major.minor* release. It does not enable you to run any prior or later release. You have to acquire new licenses if you need to use different SINAP/SIP releases. For example, if you have the SINAP/SIP 1.1 license, the license file will enable you to run SINAP/SIP 1.1 and every 1.1.x.y maintenance or patch release, but not any previous release, such as 1.0.x.y or later release like 1.2.x.y. If you replace an existing license file, the information in the new license file will not be validated until you restart the SIP application.

## Online License Update

Online license update is supported so that you can change the license type or specify a different value for the `SIPLEVEL` parameter without stopping the application that is using the SINAP/SIP software. For example, you can change the SINAP/SIP license from time-limited to permanent after the SINAP/SIP software trial period. A hook for online license update is provided so you can choose which signal is used to trigger the license update by calling the `sip_renew_license()` function included in the SINAP/SIP `libsipcore.so` library. You must write your own triggering program, or simply use the `kill` command to send the triggering signal to the application. Refer to the `hssUA` sample test program as an example, where `hssua_main.c` has set up the SIGUSR1 signal handler, `sigusr1Handler()`, which calls the `sip_renew_license()` function to run an online license update when it catches SIGUSR1.

This function first performs a verification of the new license file. If there is anything wrong with it, the update will abort and the original license remains in effect in the running SINAP/SIP application. This verification prevents the SINAP/SIP application from shutting down because of errors in the new license file. A cut-and-paste error can make a license file invalid, causing a service interruption. All license information on the system that is running SINAP/SIP is updated after calling this API.

## Licensing Problems

Encrypted signatures protect the content of the license file. Any attempt to modify its content will result in license authorization failure. The license is verified when the SIP application runs.

If the license cannot be authorized, error messages are displayed on the screen. All major errors are also printed to the system log (`/var/log/messages`).

# Running the SINAP/SIP `hssUA` Sample Test Program

After the SINAP/SIP software and license file are installed, the next step is to test the SINAP/SIP UA toolkit `hssUA` sample test program. The first step is to ensure that the correct libraries are chosen for the task. There are two version of the libraries provided with the package: development and deployment.

## Selecting a SINAP/SIP Library Version

Two sets of libraries are provided with the SINAP/SIP distribution.

- Development libraries

  Statistics and logging (trace) flags are enabled in the development libraries to provide the SDF_TRACE and SDF_STATISTICS routines for the SIP UA toolkit layer and SIP_TRACE and SIP_STATISTICS routines for the SIP stack layer.

  You can use these libraries while developing, testing, and troubleshooting your application.

- Deployment libraries

  Use the deployment libraries to prepare the service or application for deployment.

These two sets of libraries are installed in two different directories. By default, the deployment version of the libraries is symbolically linked to the corresponding ones at the `/usr/lib` directory as the default. A shell script, `select_build_type`, is provided that you can use to automatically switch from one mode to the other by replacing the links to either the deployment or the development version.

**CAUTION**

To avoid a mismatch in data structures, make sure you compile your application, including the sample test program, with the same set of compilation flags or configuration options as the library being used. See “SIP Configuration Options” on page 2-2.

**CAUTION**

Two copies of the `makefile` for the sample test program are provided, one that uses the deployment libraries and another that uses the development libraries. Make sure you use the correct `makefile`. If the symbolic links point to the deployment libraries, use the deployment `makefile`.

The `select_build_type` script under `/home/sinap_sip_master` is a shell script you can use to symbolically link to the selected the SIP software build type: development or deployment. The development version enables SIP statistics, trace, and logging features to provide additional information during the SIP application- or service-development stage. Use the deployment version of the libraries to create applications and services for the production environment to be more efficient in performance. By default, the deployment version of the SIP libraries and configuration files (for example, `MakeVars` and `uatk_config.h`) are symbolically linked. Refer to the `makefile` in `/home/sinap_sip_master/ua/toolkit/hssua/` for an example `makefile` that includes MakeVars.

**To use the development libraries**

1. Run `select_build_type` as follows:

```
[root@ftlinux8 sinap_sip_master]# /home/sinap_sip_master/select_build_type
```

2. At the `Enter build type` prompt, type **2** to use the development libraries, as shown in the following:

```
--------- Modify SINAP/SIP build type ----------

1. change build type to deployment
2. change build type to development
q Quit without changeing build type

Enter build type (1 or 2) or q: 2
```

3. List the contents of directories as shown in the following example, verifying the contents of your directories with the listings shown:

```
[root@ftlinux8 sinap_sip_master]# ls -l /usr/lib/libsip*
lrwxrwxrwx  1 root      root        58 Feb 1 05:18 /usr/lib/libsipapi.so
/home/sinap_sip_master/ua/sip_stack/lib/devel/libsipapi.so
lrwxrwxrwx  1 root      root        59 Feb 1 05:18 /usr/lib/libsipcore.so
/home/sinap_sip_master/ua/sip_stack/lib/devel/libsipcore.so
lrwxrwxrwx  1 root      root  64 Feb 1 05:18 /usr/lib/libsipserialize.so
/home/sinap_sip_master/ua/sip_stack/lib/devel/libsipserialize.so
```

```
[root@ftlinux8 sinap_sip_master]# ls -l\
/home/sinap_sip_master/ua/*/MakeVars
lrwxrwxrwx  1 root      root        50 Feb 1 05:18
/home/sinap_sip_master/ua/sip_stack/MakeVars
/home/sinap_sip_master/ua/sip_stack/MakeVars.devel
lrwxrwxrwx  1 root      root        48 Feb 1 05:18
/home/sinap_sip_master/ua/toolkit/MakeVars
/home/sinap_sip_master/ua/toolkit/MakeVars.devel

[root@ftlinux8 sinap_sip_master]# ls -l\
/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config.h
lrwxrwxrwx  1 root      root        66 Feb 1 05:18
/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config.h.devel

[root@ftlinux8 sinap_sip_master]# ls -l\
/home/sinap_sip_master/ua/sip_stack/stack_headers/stack_config.h
lrwxrwxrwx  1 root      root        70 Feb 1 05:18
/home/sinap_sip_master/ua/sip_stack/stack_headers/stack_config.h
/home/sinap_sip_master/ua/sip_stack/stack_headers/stack_config.h.devel
```

**To use the deployment libraries:**

1. Run `select_build_type` as shown:

```
[root@ftlinux8 sinap_sip_master]# /home/sinap_sip_master/select_build_type
```

2. At the `Enter build type` prompt, type **1** to use the deployment libraries, as shown in the following:

```
--------- Modify SINAP/SIP build type ----------

1. change build type to deployment
2. change build type to development
q Quit without changeing build type

Enter build type (1 or 2) or q: 1
```

3. List the contents of directories as shown in the following example, verifying the contents of your directories with the listings shown:

```
[root@ftlinux8 sinap_sip_master]# ls -l /usr/lib/libsip*
lrwxrwxrwx  1 root      root        59 Feb 1 05:23 /usr/lib/libsipapi.so
/home/sinap_sip_master/ua/sip_stack/lib/deploy/libsipapi.so
lrwxrwxrwx  1 root      root        60 Feb 1 05:23 /usr/lib/libsipcore.so
/home/sinap_sip_master/ua/sip_stack/lib/deploy/libsipcore.so
lrwxrwxrwx  1 root      root        65 Feb 1 05:23
/usr/lib/libsipserialize.so
/home/sinap_sip_master/ua/sip_stack/lib/deploy/libsipserialize.so
```

```
[root@ftlinux8 sinap_sip_master]# ls -l\
/home/sinap_sip_master/ua/sip_stack/MakeVars*
lrwxrwxrwx 1 root root 51 Mar 2 04:08
/home/sinap_sip_master/ua/sip_stack/MakeVars
/home/sinap_sip_master/ua/sip_stack/MakeVars.deploy
-rw-rw-r-- 1 root sinap 5639 Nov 3 2004
/home/sinap_sip_master/ua/sip_stack/MakeVars.deploy
-rw-rw-r-- 1 root sinap 5668 Nov 3 2004
/home/sinap_sip_master/ua/sip_stack/MakeVars.devel

[root@ftlinux8 sinap_sip_master]# ls -l\
/home/sinap_sip_master/ua/toolkit/MakeVars*
lrwxrwxrwx 1 root root 49 Mar 2 04:08
/home/sinap_sip_master/ua/toolkit/MakeVars
/home/sinap_sip_master/ua/toolkit/MakeVars.deploy
-rw-rw-r-- 1 root sinap 4084 Nov 3 2004
/home/sinap_sip_master/ua/toolkit/MakeVars.deploy
-rw-rw-r-- 1 root sinap 4142 Nov 3 2004
/home/sinap_sip_master/ua/toolkit/MakeVars.devel

[root@ftlinux8 sinap_sip_master]# ls -l\
/home/sinap_sip_master/ua/sip_stack/stack_headers/stack_config*
lrwxrwxrwx 1 root root 71 Mar 2 04:08
/home/sinap_sip_master/ua/sip_stack/stack_headers/stack_config.h
/home/sinap_sip_master/ua/sip_stack/stack_headers/stack_config.h.deploy
-rw-r--r-- 1 root sinap 767 Nov 3 2004
/home/sinap_sip_master/ua/sip_stack/stack_headers/stack_config.h.deploy
-rw-r--r-- 1 root sinap 769 Nov 3 2004
/home/sinap_sip_master/ua/sip_stack/stack_headers/stack_config.h.devel

[root@ftlinux8 sinap_sip_master]# ls -l\
/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config*
lrwxrwxrwx 1 root root 67 Mar 2 04:08
/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config.h
/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config.h.deploy
-rw-r--r-- 1 root sinap 809 Nov 3 2004
/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config.h.deploy
-rw-r--r-- 1 root sinap 811 Nov 3 2004
/home/sinap_sip_master/ua/toolkit/uatk_headers/uatk_config.h.devel
```

## Running the hssUA Program in Loopback Mode

The hssUA program is a sample test application program provided with the SINAP/SIP package to demonstrate the usage of the UA Toolkit APIs. The UA toolkit uses user profiles to initiate and accept calls, so the application should maintain those user profiles. Refer to the *HSS SIP User Agent Toolkit User Manual* Chapter 9 and Appendix B, for more information on the user profile and the hssUA program.

These user profiles consist of details of the caller, such as the From address, the Contact address, and the media capabilities for the endpoint of the user. The hssUA UA uses two such profiles, which should be provided in the form of two XML files:

- profile-0.xml
- profile-1.xml

To be able to run two hssUA instances for loopback testing on the same system, two sets of profiles are provided that use localhost as the host name at the From and Contact header fields, each configured with a different port number, to be referenced by these.

The installation file names and locations of these profiles are as follows:

- /home/sinap_sip_master/ua/toolkit/hssua/profile-0.xml.ua1 (hssUA profile 0 of user agent 1)
- /home/sinap_sip_master/ua/toolkit/hssua/profile-0.xml.ua2 (hssUA profile 0 of user agent 2)
- /home/sinap_sip_master/ua/toolkit/hssua/profile-1.xml.ua1 (hssUA profile 1 of user agent 1)
- /home/sinap_sip_master/ua/toolkit/hssua/profile-1.xml.ua2 (hssUA profile 1 of user agent 2)

In these file names, the ua1 and ua2 suffixes indicate hssUA user agent instances 1 and 2, respectively.

By default, symbolic links are made from profile-0.xml.ua1 and profile-1.xml.ua1 to profile-0.xml and profile-1.xml respectively.

Start up an instance of hssUA so these two profiles can be parsed and referenced. Then, remove these two symbolic links and remake the links from profile-0.xml.ua2 and profile-2.xml.ua1 to profile-0.xml and profile-1.xml. Start up a second instance of hssUA, which will reference these profiles. In this way, two hssUA instances can be run for loopback testing on the same machine.

Note that URIs at profile-0*.xml are specified with the SIPS parameter to exercise the TLS feature, while URIs at profile-1*.xml are specified with the SIP parameter to exercise normal SIP messages using either UDP or TCP. Also, note that the profile selected by the `hssUA` program needs to be consistent with the destination's URI. That is, if the destination address is specified with a SIPS URI, then select `profile-0*.xml`; otherwise, the `hssUA` program fails at `sdf_ivk_uaMakeCall()`. Similarly, if the destination address is specified with a SIP URI, select `profile-1*.xml`.

Following are the steps and output from running two `hssUA` instances in loop-back mode with appropriate links set up to user profiles.

1. Run a first `hssUA` instance, which references to the default profile, `profile-[0,1].xml.ua1`.

   a. Make sure you are in the `toolkit/hssua` directory and verify that the profiles are linked to `profile-[0,1].xml.ua1`.

   ```
   [root@ftlinux1 hssua]# pwd
   /home/sinap_sip_master/ua/toolkit/hssua

   [sip@ftlinux1 hssua]$ ls -l *xml
   lrwxrwxrwx 1 sip       sinap       57 Oct 13 21:31 profile-0.xml ->
   /home/sinap_sip_master/ua/toolkit/hssua/profile-0.xml.ua1
   lrwxrwxrwx 1 sip       sinap       57 Oct 13 21:31 profile-1.xml ->
   /home/sinap_sip_master/ua/toolkit/hssua/profile-1.xml.ua1
   ```

   b. Start an instance of `hssUA`, providing responses to the prompts as shown:

   ```
   [sip@ftlinux1 hssua]$ bin/hssUA
   SIP license checked out...
   SIP UA license granted...
   Reading file: profile-0.xml
   Parsing file: "profile-0.xml" Done!!
   Reading file: profile-1.xml
   Parsing file: "profile-1.xml" Done!!
   TLS : Bound and Listening to port 9801
   TLS : Bound and Listening to port 9802
   UDP : Bound and Listening to port 9803
   TCP : Bound and Listening to port 9803
   UDP : Bound and Listening to port 9804
   TCP : Bound and Listening to port 9804

   Do you want to delay the responses(y/n):n

   Do you want to send 1xx responses(y/n):y

   Do you want to enable Session Change(y/N):n
   ```

```
==================================
WELCOME TO THE HSS SIP USER AGENT
==================================
===================================================
MAIN MENU
===================================================
Sip UA Toolkit product id is:1-000-5-0208-0904-14-0201-000
(i) Make a Call
(c) Cancel a Call
(b) Hang up
(l) List Active Calls
(v) Modify Media Params (Re-Invite)
(h) Hold Call
(t) UnHold Call
(a) Add Streams to an existing call
(g) Add Group to an existing call
(d) Delete Streams from an existing call
(r) Register UA
(x) Delete Registration UA
(o) Send Options
(e) Send REFER
(k) Transfer This Call
(f) Send Info
(u) Send Update Request
(j) Send Unknown Request
(m) Send Message Request
(n) Send Notify Request
(z) Send Subscribe Request
(y) Abort an ongoing transaction
(q) Quit User Agent
```

2. In another window, run a second `hssUA` instance that references `profile-[0,1].xml.ua2`. Make sure you are in the `toolkit/hssua` directory, list the profiles, and start another instance of `hssUA`, providing responses to the prompts as shown in the following:

```
[sip@ftlinux1 hssua]$ pwd
/home/sinap_sip_master/ua/toolkit/hssua

[sip@ftlinux1 hssua]$ rm -f *.xml

[sip@ftlinux1 hssua]$ ln -s \
/home/sinap_sip_master/ua/toolkit/hssua/profile-0.xml.ua2 \
profile-0.xml

[sip@ftlinux1 hssua]$ ln -s \
/home/sinap_sip_master/ua/toolkit/hssua/profile-1.xml.ua2 \
profile-1.xml
```

```
[sip@ftlinux1 hssua]$ ps -ef | grep hssUA
sip     17109 15525 0 21:32 pts/4     00:00:00 bin/hssUA
sip     20361 31974 0 21:35 pts/1     00:00:00 grep hssUA

[sip@ftlinux1 hssua]$ bin/hssUA
SIP license checked out...
SIP UA license granted...
Reading file: profile-0.xml
Parsing file: "profile-0.xml" Done!!
Reading file: profile-1.xml
Parsing file: "profile-1.xml" Done!!
TLS : Bound and Listening to port 9821
TLS : Bound and Listening to port 9822
UDP : Bound and Listening to port 9823
TCP : Bound and Listening to port 9823"
UDP : Bound and Listening to port 9824
TCP : Bound and Listening to port 9824

Do you want to delay the responses(y/n):n

Do you want to send 1xx responses(y/n):y

Do you want to enable Session Change(y/N):n
==================================
WELCOME TO THE HSS SIP USER AGENT
==================================
```

3. Initiate a call from the second `hssUA` instance. At the main menu prompt, type **`i`**. Then, at the `Display Name` prompt, type **`profile0-ua1`**, at the `Port` prompt, type **`9801`**, and answer the remaining prompts as shown in the following:

```
===================================================
MAIN MENU
===================================================
Sip UA Toolkit product id is:1-000-5-0208-0904-14-0201-000
(i) Make a Call
.
.
.
Choice: i
Initing CallObject Num 0
Please enter the To header information

Display Name: profile0-ua1
Address("[protocol:]address"): sips:profile0-ua1@localhost
```

```
Port: 9801

Do you want to add URI param? [n] n

Protocol(TCP='t'/TLS='s'/UDP='u'):s

Please enter [Y/y] letter to have modifiedReqURI [n] y
List of profiles :
Profile 0 : <sips:profile0-ua2@localhost>
Profile 1 : <sip:profile1-ua2@localhost>

Choose a profile: 0

Do you want to send Session-Expires header [y/n]: n

Do you want to send MinSE header[y/n]: n

Send Replaces Header in INVITE ? (y/n): n

Send SDP in INVITE ? (y/n) : y

---- CALLED sdf_cbk_uaSendMsgOnNetwork ----
------------------------------------------------------------
Trace generated at : Wed Oct 13 21:36:46 2004
------------------------------------------------------------
INVITE sips:profile0-ua1@localhost:9801 SIP/2.0
Via: SIP/2.0/TLS 127.0.0.1:9821;branch=z9hG4bK1473349716-20401
Max-Forwards: 70
Allow: INVITE,BYE,CANCEL,ACK,INFO,PRACK,COMET,OPTIONS,SUBSCRIBE,\
NOTIFY,MESSAGE,REFER,REGISTER,UPDATE
Supported: timer,replaces,billing,prescence,*
From: profile0-ua2 <sips:profile0-ua2@localhost:9821>\
;tag=hssUA_1473349865-20401
To: profile0-ua1 <sips:profile0-ua1@localhost:9801>
Call-ID: 1473349325-20401
CSeq: 1 INVITE
Contact: profile0-cont2 <sips:profile0-cont2@localhost:9822>
Content-Type: application/sdp
Content-Length: 344

v=0
o=hughes 256 2 IN IP4 139.85.229.21
s=SipSession with HssUA
i=SIP Session in HssUA
u=http://www.yahoo.com
e=airoy@hss.hns.com
p=+1-809-256
c=IN IP4 539.85.259.22
```

```
b=nscom:84
t=0 0
k=clear:hnscom
a=rtpmap2
m=audio 3000/2 RTP/AVP 3
a=rtpmap:3 GSM/8000/1
a=sendrecv
m=audio 2000/2 RTP/AVP 7
a=rtpmap:7 LPC/8000/1
a=sendrecv
--------------------------------------------------------------
```

4. In the window where you started the first `hssUA` instance, type **`y`** after the `Accept/Reject/Redirect the call` prompt to accept the call.
5. In the window where you started the second `hssUA` instance, the `hssUA` receives `100 Trying`, `180 Ringing`, and `200 OK` messages from the first `hssUA` instance. In the window where you started the first `hssUA`, you can see that an `ACK` message is sent in response to the `200 OK` message:

```
Got a message on existing TLS connection
------------------------------------------------------------
Trace generated at : Wed Oct 13 21:36:46 2004
------------------------------------------------------------
SIP/2.0 100 Trying
Via: SIP/2.0/TLS 127.0.0.1:9821;branch=z9hG4bK1473349716-20401
From:profile0-ua2 <sips:profile0-ua2@localhost:9821>\
;tag=hssUA_1473349865-20401
To: profile0-ua1 <sips:profile0-ua1@localhost:9801>\
;tag=hssUA_1475500507-17109
Call-ID: 1473349325-20401
CSeq: 1 INVITE
Contact: profile0-cont1 <sips:profile0-cont1@localhost:9802>
Content-Length: 0
--------------------------------------------------------------

====== Comparison function says: same call leg =====
 1xx Received for method : INVITE
**********IN PROGRESS,PLEASE HOLD ON **********

************OTHER EVENT**********

Got a message on existing TLS connection
------------------------------------------------------------
Trace generated at : Wed Oct 13 21:36:46 2004
------------------------------------------------------------
SIP/2.0 180 Ringing
```

```
Via: SIP/2.0/TLS 127.0.0.1:9821;branch=z9hG4bK1473349716-20401
From: profile0-ua2 <sips:profile0-ua2@localhost:9821>\
;tag=hssUA_1473349865-20401

To: profile0-ua1 <sips:profile0-ua1@localhost:9801>\
;tag=hssUA_1475500507-17109
Call-ID: 1473349325-20401
CSeq: 1 INVITE
Contact: profile0-cont1 <sips:profile0-cont1@localhost:9802>
Content-Length: 0
--------------------------------------------------------------

====== Comparison function says: same call leg =====
 1xx Received for method : INVITE
**********IN PROGRESS,PLEASE HOLD ON **********

************OTHER EVENT**********

Got a message on existing TLS connection
------------------------------------------------------------
Trace generated at : Wed Oct 13 21:36:52 2004
------------------------------------------------------------
SIP/2.0 200 OK
Via: SIP/2.0/TLS 127.0.0.1:9821;branch=z9hG4bK1473349716-20401
From: profile0-ua2 <sips:profile0-ua2@localhost:9821>\
;tag=hssUA_1473349865-20401
To: profile0-ua1 <sips:profile0-ua1@localhost:9801>\
;tag=hssUA_1475500507-17109
Call-ID: 1473349325-20401

CSeq: 1 INVITE
Contact: profile0-cont1 <sips:profile0-cont1@localhost:9802>
Allow: INVITE,BYE,CANCEL,ACK,INFO,PRACK,COMET,OPTIONS,SUBSCRIBE,
\NOTIFY, MESSAGE, REFER,REGISTER,UPDATE
Supported: billing,prescence,*
Content-Type: application/sdp
Content-Length: 344

v=0
o=hughes 256 2 IN IP4 139.85.229.21
s=SipSession with HssUA
i=SIP Session in HssUA
u=http://www.yahoo.com
e=airoy@hss.hns.com
p=+1-809-256
c=IN IP4 539.85.259.22
b=nscom:84
t=0 0
```

```
k=clear:hnscom
a=rtpmap2
m=audio 3000/2 RTP/AVP 3
a=rtpmap:3 GSM/8000/1
a=sendrecv
m=audio 2000/2 RTP/AVP 7
a=rtpmap:7 LPC/8000/1
a=sendrecv
--------------------------------------------------------------

====== Comparison function says: same call leg =====
********** YOUR CALL HAS BEEN ACCEPTED **********
---- CALLED sdf_cbk_uaSendMsgOnNetwork ----
------------------------------------------------------------
Trace generated at : Wed Oct 13 21:36:52 2004
------------------------------------------------------------
ACK sips:profile0-cont1@localhost:9802 SIP/2.0
Via: SIP/2.0/TLS 127.0.0.1:9821;branch=z9hG4bK1480739172-20401
Max-Forwards: 70
From: profile0-ua2 <sips:profile0-ua2@localhost:9821>\
;tag=hssUA_1473349865-20401
To: profile0-ua1 <sips:profile0-ua1@localhost:9801>\
;tag=hssUA_1475500507-17109
Call-ID: 1473349325-20401
CSeq: 1 ACK
Contact: profile0-cont2 <sips:profile0-cont2@localhost:9822>
Content-Length: 0
--------------------------------------------------------------
**********ACK SENT ***********************

**********YOUR CALL IS CONNECTED **********
************OTHER EVENT**********
```

6. To list active calls, press ENTER to display the main menu and then type **`l`**, as shown in the following:

```
Press enter to return to the Main Menu:
===================================================
MAIN MENU
===================================================
Sip UA Toolkit product id is:1-000-5-0208-0904-14-0201-000
(i) Make a Call
(c) Cancel a Call
(b) Hang up
(l) List Active Calls
(v) Modify Media Params (Re-Invite)
(h) Hold Call
(t) UnHold Call
```

```
(a) Add Streams to an existing call
(g) Add Group to an existing call
(d) Delete Streams from an existing call
(r) Register UA
(x) Delete Registration UA
(o) Send Options
(e) Send REFER
(k) Transfer This Call
(f) Send Info
(u) Send Update Request
(j) Send Unknown Request
(m) Send Message Request
(n) Send Notify Request
(z) Send Subscribe Request
(y) Abort an ongoing transaction
(q) Quit User Agent

Choice: l
Currently established calls:
Call NO 0
From: <sips:profile0-ua2@localhost:9821>
To: <sips:profile0-ua1@localhost:9801>
Call-Id: <1473349325-20401>
```

7. Press the ENTER key to display the main menu, then type **`b`** to hang up the call from the second instance of `hssUA`.

```
Press enter to return to the Main Menu:
===================================================
MAIN MENU
===================================================
Sip UA Toolkit product id is:1-000-5-0208-0904-14-0201-000
(i) Make a Call
(c) Cancel a Call
(b) Hang up
(l) List Active Calls
(v) Modify Media Params (Re-Invite)
(h) Hold Call
(t) UnHold Call
(a) Add Streams to an existing call
(g) Add Group to an existing call
(d) Delete Streams from an existing call
(r) Register UA
(x) Delete Registration UA
(o) Send Options
(e) Send REFER
(k) Transfer This Call
(f) Send Info
```

```
(u) Send Update Request
(j) Send Unknown Request
(m) Send Message Request
(n) Send Notify Request
(z) Send Subscribe Request
(y) Abort an ongoing transaction
(q) Quit User Agent

Choice: b
Currently active calls:
Call NO 0
From: <sips:profile0-ua2@localhost:9821>
To: <sips:profile0-ua1@localhost:9801>
Call-Id: <1473349325-20401>

Call No. to be terminated: 0
---- CALLED sdf_cbk_uaSendMsgOnNetwork ----
------------------------------------------------------------
Trace generated at : Wed Oct 13 21:38:02 2004
------------------------------------------------------------
BYE sips:profile0-cont1@localhost:9802 SIP/2.0
Via: SIP/2.0/TLS 127.0.0.1:9821;branch=z9hG4bK1550849381-20401
Max-Forwards: 70
From: profile0-ua2 <sips:profile0-ua2@localhost:9821>\
;tag=hssUA_1473349865-2040 1
To: profile0-ua1 <sips:profile0-ua1@localhost:9801>\
;tag=hssUA_1475500507-17109
Call-ID: 1473349325-20401
CSeq: 2 BYE
Content-Length: 0
--------------------------------------------------------------
```

8. In the window where you started the first `hssUA` instance, type **`y`** at the `Do you want to accept the Bye or not` prompt to accept the call.
9. Then the second `hssUA` instance receives a `200 OK` message from the first instance of `hssUA` in response to the `BYE` message:

```
Got a message on existing TLS connection
------------------------------------------------------------
Trace generated at : Wed Oct 13 21:38:04 2004
------------------------------------------------------------
SIP/2.0 200 OK
Via: SIP/2.0/TLS 127.0.0.1:9821;branch=z9hG4bK1550849381-20401
From: profile0-ua2 <sips:profile0-ua2@localhost:9821>\
;tag=hssUA_1473349865-20401
To: profile0-ua1 <sips:profile0-ua1@localhost:9801>\
;tag=hssUA_1475500507-17109
Call-ID: 1473349325-20401
```

```
CSeq: 2 BYE
Contact: profile0-cont1 <sips:profile0-cont1@localhost:9802>
Allow: INVITE,BYE,CANCEL,ACK,INFO,PRACK,COMET,OPTIONS,SUBSCRIBE,
\NOTIFY,\ MESSAGE, REFER,REGISTER,UPDATE
Supported: billing,prescence,*
Content-Length: 0
--------------------------------------------------------------

====== Comparison function says: same call leg =====
**********FINAL RESPONSE RECEIVED **********

**********YOUR CALL IS TERMINATED **********

************OTHER EVENT**********
```

10. Continue at the first `hssUA` instance window and type **`i`** to make another call. But, to exercise SIP URI using UDP or TCP as the transport protocol, specify the following for the destination:

    ```
    Display Name: profile1-ua2
    Address: sip:profile1-ua2@localhost
    ```

    Select **`Profile 1 : <sip:profile1-ua1@localhost>`** for the caller.

    The following shows how to set up this call:

    ```
    Choice: i

    Initing CallObject Num 0

    Please enter the To header information

    Display Name: profile1-ua2

    Address("[protocol:]address"): sip:profile1-ua2@localhost

    Port: 9823

    Do you want to add URI param? [n] n

    Protocol(TCP='t'/TLS='s'/UDP='u'): u

    Please enter [Y/y] letter to have modifiedReqURI n
    List of profiles :
    Profile 0 : <sips:profile0-ua1@localhost>
    Profile 1 : <sip:profile1-ua1@localhost>

    Choose a profile : 1
    ```

```
Do you want to send Session-Expires header [y/n]: n

Do you want to send MinSE header[y/n]: n

Send Replaces Header in INVITE ? (y/n): n

Send SDP in INVITE ? (y/n): y

---- CALLED sdf_cbk_uaSendMsgOnNetwork ----

------------------------------------------------------------
Trace generated at : Mon Jan 17 16:49:19 2005
------------------------------------------------------------
INVITE sip:profile1-ua2@localhost:9823 SIP/2.0
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK1528648317-29399
Max-Forwards: 70
Allow: INVITE,BYE,CANCEL,ACK,INFO,PRACK,COMET,OPTIONS,SUBSCRIBE,
\NOTIFY, MESSAGE,REFER,REGISTER,UPDATE
Supported: timer,replaces,billing,prescence,*
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_1528648378-29399
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>
Call-ID: 1528648153-29399
CSeq: 1 INVITE
Contact: profile1-cont1 <sip:profile1-cont1@localhost:9804>
Content-Type: application/sdp
Content-Length: 517

v=0
o=hughes 10 12 IN IP4 139.85.229.21
s=SipSession with HssUA
i=SIP Session in HssUA
u=http://www.yahoo.com
e=airoy@hss.hns.com
p=+1-809-256
c=IN IP4 539.85.259.22
b=nscom:84
t=0 0
k=clear:hnscom
a=rtpmap2
m=audio 1000/2 RTP/AVP 3
i=hello
c=IN IP4 139.85.229.22
b=xy:128
k=clear:xy128
a=rtpmap:3 GSM/8000/1
a=sendrecv
```

```
m=audio 6000/2 RTP/AVP 0 99 98
i=hello
c=IN IP4 139.85.229.22
b=xy:128
k=clear:xy128
a=rtpmap:0 PCMU/8000/1
a=rtpmap:99 L17/33333/3
a=rtpmap:98 L17/77777/2
a=sendrecv
--------------------------------------------------------------
sending to ip:127.0.0.1, port:9823
SDF_TRACE: Generated At Mon Jan 17 16:49:19 2005
::::<>::::Call : 1528648153-29399 - Changing call state from Sdf_en_idle
to Sdf_en_inviteSent
Got a message on UDP

------------------------------------------------------------
Trace generated at : Mon Jan 17 16:49:19 2005
------------------------------------------------------------
SIP/2.0 100 Trying
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK1528648317-29399
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_1528648378-29399
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>\
;tag=hssUA_1531205868-29412
Call-ID: 1528648153-29399
CSeq: 1 INVITE
Contact: profile1-cont2 <sip:profile1-cont2@localhost:9824>
Content-Length: 0

====== Comparison function says: same call leg =====
SDF_TRACE: Generated At Mon Jan 17 16:49:19 2005

====== Comparison function says: same call leg =====
::::<>::::Call : 1528648153-29399 - Changing call state from
Sdf_en_inviteSent to Sdf_en_provisionalRespReceived

1xx Received for method : INVITE
**********IN PROGRESS,PLEASE HOLD ON **********

************OTHER EVENT**********

SDF_TRACE: Generated At Mon Jan 17 16:49:19 2005
::::<>::::Call : 1528648153-29399 - Changing call state from
Sdf_en_provisionalRespReceived to Sdf_en_provisionalRespReceived

1xx Received for method : INVITE
**********IN PROGRESS,PLEASE HOLD ON **********
```

```
************OTHER EVENT**********
```

11. At the second `hssUA` instance window, observe that the `100 Trying` and `180 Ringing` messages are sent in response to the INVITE message just received. Then to accept the call, type **`y`** at the `Accept/Reject/Redirect the call?` prompt, as shown in the following:

```
Got a message on UDP

------------------------------------------------------------
Trace generated at : Mon Jan 17 16:49:19 2005
------------------------------------------------------------
INVITE sip:profile1-ua2@localhost:9823 SIP/2.0
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK1528648317-29399
Max-Forwards: 70
Allow: INVITE,BYE,CANCEL,ACK,INFO,PRACK,COMET,OPTIONS,SUBSCRIBE,
\NOTIFY,MESSAGE,REFER,REGISTER,UPDATE
Supported: timer,replaces,billing,prescence,*
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_1528648378-29399
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>
Call-ID: 1528648153-29399
CSeq: 1 INVITE
Contact: profile1-cont1 <sip:profile1-cont1@localhost:9804>
Content-Type: application/sdp
Content-Length: 517

v=0
o=hughes 10 12 IN IP4 139.85.229.21
s=SipSession with HssUA
i=SIP Session in HssUA
u=http://www.yahoo.com
e=airoy@hss.hns.com
p=+1-809-256
c=IN IP4 539.85.259.22
b=nscom:84
t=0 0
k=clear:hnscom
a=rtpmap2
m=audio 1000/2 RTP/AVP 3
i=hello
c=IN IP4 139.85.229.22
b=xy:128
k=clear:xy128
a=rtpmap:3 GSM/8000/1
a=sendrecv
m=audio 6000/2 RTP/AVP 0 99 98
```

```
i=hello
c=IN IP4 139.85.229.22
b=xy:128
k=clear:xy128
a=rtpmap:0 PCMU/8000/1
a=rtpmap:99 L17/33333/3
a=rtpmap:98 L17/77777/2
a=sendrecv
---------------------------------------------------------------
SDF_TRACE: Generated At Mon Jan 17 16:49:19 2005
::::<>::::Call : 1528648153-29399 - Changing call state from Sdf_en_idle
to Sdf_en_inviteReceived

---- CALLED sdf_cbk_uaSendMsgOnNetwork ----

-------------------------------------------------------------
Trace generated at : Mon Jan 17 16:49:19 2005
-------------------------------------------------------------
SIP/2.0 100 Trying
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK1528648317-29399
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_1528648378-29399
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>\
;tag=hssUA_1531205868-29412
Call-ID: 1528648153-29399
CSeq: 1 INVITE
Contact: profile1-cont2 <sip:profile1-cont2@localhost:9824>
Content-Length: 0
---------------------------------------------------------------
sending to ip:127.0.0.1, port:9803
SDF_TRACE: Generated At Mon Jan 17 16:49:19 2005
::::<>::::Call : 1528648153-29399 - Changing call state from
Sdf_en_inviteReceived to Sdf_en_provisionalRespSent

---- CALLED sdf_cbk_uaSendMsgOnNetwork ----

-------------------------------------------------------------
Trace generated at : Mon Jan 17 16:49:19 2005
-------------------------------------------------------------
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK1528648317-29399

From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_1528648378-29399
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>\
;tag=hssUA_1531205868-29412
Call-ID: 1528648153-29399
CSeq: 1 INVITE
```

```
Contact: profile1-cont2 <sip:profile1-cont2@localhost:9824>
Content-Length: 0
--------------------------------------------------------------
sending to ip:127.0.0.1, port:9803
SDF_TRACE: Generated At Mon Jan 17 16:49:19 2005
::::<>::::Call : 1528648153-29399 - Changing call state from
Sdf_en_provisionalRespSent to Sdf_en_provisionalRespSent
You have received a call from profile1-ua1 (sip:profile1-ua1@localhost)
Accept/Reject/Redirect the call ? (y/n/r) : y
---- CALLED sdf_cbk_uaSendMsgOnNetwork ----

------------------------------------------------------------
Trace generated at : Mon Jan 17 17:13:46 2005
------------------------------------------------------------
SIP/2.0 200 OK
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK2989135485-29520
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_2989135626-29520
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>\
;tag=hssUA_2991186882-29533
Call-ID: 2989135131-29520
CSeq: 1 INVITE
Contact: profile1-cont2 <sip:profile1-cont2@localhost:9824>
Allow: INVITE,BYE,CANCEL,ACK,INFO,PRACK,COMET,OPTIONS,SUBSCRIBE,
\NOTIFY,MESSAGE,REFER,REGISTER,UPDATE
Supported: billing,prescence,*
Content-Type: application/sdp
Content-Length: 517

v=0
o=hughes 10 12 IN IP4 139.85.229.21
s=SipSession with HssUA
i=SIP Session in HssUA
u=http://www.yahoo.com
e=airoy@hss.hns.com
p=+1-809-256
c=IN IP4 539.85.259.22
b=nscom:84
t=0 0
k=clear:hnscom
a=rtpmap2
m=audio 1000/2 RTP/AVP 3
i=hello
c=IN IP4 139.85.229.22
b=xy:128
k=clear:xy128
a=rtpmap:3 GSM/8000/1
a=sendrecv
```

```
m=audio 6000/2 RTP/AVP 0 99 98
i=hello
c=IN IP4 139.85.229.22
b=xy:128
k=clear:xy128
a=rtpmap:0 PCMU/8000/1
a=rtpmap:99 L17/33333/3
a=rtpmap:98 L17/77777/2
a=sendrecv
--------------------------------------------------------------
sending to ip:127.0.0.1, port:9803
SDF_TRACE: Generated At Mon Jan 17 17:13:46 2005
::::<>::::Call : 2989135131-29520 - Changing call state from
Sdf_en_provisionalRespSent to Sdf_en_finalSuccessSent

Got a message on UDP

------------------------------------------------------------
Trace generated at : Mon Jan 17 17:13:46 2005
------------------------------------------------------------
ACK sip:profile1-cont2@localhost:9824 SIP/2.0
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK2997825559-29520
Max-Forwards: 70
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_2989135626-29520
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>\
;tag=hssUA_2991186882-29533
Call-ID: 2989135131-29520
CSeq: 1 ACK
Contact: profile1-cont1 <sip:profile1-cont1@localhost:9804>
Content-Length: 0

--------------------------------------------------------------
====== Comparison function says: same call leg =====
SDF_TRACE: Generated At Mon Jan 17 17:13:46 2005
::::<>::::Call : 2989135131-29520 - Changing call state from
Sdf_en_finalSuccessSent to Sdf_en_callEstablished

************** ACK RECIEVED ***************

********** YOUR CALL IS CONNECTED **********
```

12. In the first `hssUA` instance window, type **`b`** to hang up the call:

```
Choice: b
Currently active calls:
Call NO 0
From: <sip:profile1-ua1@localhost:9803>
To: <sip:profile1-ua2@localhost:9823>
```

```
Call-Id: <2989135131-29520>

Call No. to be terminated: 0

---- CALLED sdf_cbk_uaSendMsgOnNetwork ----

------------------------------------------------------------
Trace generated at : Mon Jan 17 17:18:11 2005
------------------------------------------------------------
BYE sip:profile1-cont2@localhost:9824 SIP/2.0
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK3262992155-29520
Max-Forwards: 70
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_2989135626-29520
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>\
;tag=hssUA_2991186882-29533
Call-ID: 2989135131-29520
CSeq: 2 BYE
Content-Length: 0
--------------------------------------------------------------
sending to ip:127.0.0.1, port:9824
SDF_TRACE: Generated At Mon Jan 17 17:18:11 2005
::::<>::::Call : 2989135131-29520 - Changing call state from
Sdf_en_callEstablished to Sdf_en_byeSent
```

13. At the second `hssUA` instance window, when the BYE message is received, type **`y`** at the `Do you want to Accept the Bye or not?` prompt to accept the BYE so that a 200 OK message will be returned:

```
====== Comparison function says: same call leg =====
SDF_TRACE: Generated At Mon Jan 17 17:18:11 2005
::::<>::::Call : 2989135131-29520 - Changing call state from
Sdf_en_callEstablished to Sdf_en_byeReceived

********** BYE RECEIVED **********

********** YOU HAVE RECEIVED REQUEST FOR CALL TERMINATION **********
Got a message on UDP

------------------------------------------------------------
Trace generated at : Mon Jan 17 17:18:11 2005
------------------------------------------------------------
BYE sip:profile1-cont2@localhost:9824 SIP/2.0
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK3262992155-29520
Max-Forwards: 70
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_2989135626-29520
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>\
```

```
;tag=hssUA_2991186882-29533
Call-ID: 2989135131-29520
CSeq: 2 BYE
Content-Length: 0

--------------------------------------------------------------
**********REMOTE**********
Do you want to Accept the Bye or not? (y/n) : y
---- CALLED sdf_cbk_uaSendMsgOnNetwork ----

------------------------------------------------------------
Trace generated at : Mon Jan 17 17:18:21 2005
------------------------------------------------------------
SIP/2.0 200 OK
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK3262992155-29520
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_2989135626-29520
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>\
;tag=hssUA_2991186882-29533
Call-ID: 2989135131-29520
CSeq: 2 BYE
Contact: profile1-cont2 <sip:profile1-cont2@localhost:9824>
Allow: INVITE,BYE,CANCEL,ACK,INFO,PRACK,COMET,OPTIONS,SUBSCRIBE,
\NOTIFY,MESSAGE,REFER,REGISTER,UPDATE
Supported: billing,prescence,*
Content-Length: 0
--------------------------------------------------------------
sending to ip:127.0.0.1, port:9803
SDF_TRACE: Generated At Mon Jan 17 17:18:21 2005
::::<>::::Call : 2989135131-29520 - Changing call state from
Sdf_en_byeReceived to Sdf_en_callTerminated

************** 200 OK SENT *********
********* YOUR CALL IS TERMINATED ******
```

14. In the first `hssUA` instance window, observe that a 200 OK message is received and the call is terminated:

```
Got a message on UDP

------------------------------------------------------------
Trace generated at : Mon Jan 17 17:18:21 2005
------------------------------------------------------------
SIP/2.0 200 OK
Via: SIP/2.0/UDP 127.0.0.1:9803;branch=z9hG4bK3262992155-29520
From: profile1-ua1 <sip:profile1-ua1@localhost:9803>\
;tag=hssUA_2989135626-29520
To: profile1-ua2 <sip:profile1-ua2@localhost:9823>\
```

```
;tag=hssUA_2991186882-29533
Call-ID: 2989135131-29520
CSeq: 2 BYE
Contact: profile1-cont2 <sip:profile1-cont2@localhost:9824>
Allow: INVITE,BYE,CANCEL,ACK,INFO,PRACK,COMET,OPTIONS,SUBSCRIBE,\
NOTIFY,MESSAGE,REFER,REGISTER,UPDATE
Supported: billing,prescence,*
Content-Length: 0
--------------------------------------------------------------
====== Comparison function says: same call leg =====
SDF_TRACE: Generated At Mon Jan 17 17:18:21 2005
::::<>::::Call : 2989135131-29520 - Changing call state from
Sdf_en_byeSent to Sdf_en_callTerminated

**********FINAL RESPONSE RECEIVED **********

**********YOUR CALL IS TERMINATED **********
```

15. Press ENTER in each window to display the main menu. Then type **`q`** (for quit) to exit from the User Agent program.

# Appendix A
# SINAP/SIP User Agent and Stack Compliance

1. The SINAP/SIP User Agent is fully compliant with:

- Session Initiation Protocol (RFC 3261) (the protocol stack has been subjected to all SiPit message tests)
- Session Description Protocol (RFC 2327)
- SIP INFO Method (RFC 2976) (October 2000)
- SIP UPDATE Method (RFC 3311) (September 2002)
- HTTP Authentication (RFC 2617) (June 1999)
- The MD5 Message-Digest Algorithm (RFC 1321) (April 1992)
- Reliability of Provisional Responses in the Session Initiation Protocol (SIP), RFC 3262, June 2002
- An Offer/Answer Model with the Session Description Protocol (SDP), RFC 3264, June 2002
- SIP-Specific Event Notification, draft-ietf-sip-events-01.txt (November, 2001)
- Integration of Resource Management and SIP, RFC-3312 (October, 2002)
- Grouping of Media Lines in the Session Description Protocol (SDP), RFC 3388, December 2002
- The SIP Session Timer, draft-ietf-sip-session-timer-10.txt (November 4, 2002)
- Support for IPv6 in Session Description Protocol (SDP), RFC 3266, June 2002
- URLs for Telephone Calls (RFC 2806) (April 2000)
- The SIP Refer Method – RFC 3515
- The SIP Referred-By Mechanism, draft-ietf-sip-referredby-01.txt
- The Session Inititation Protocol (SIP) "Replaces" Header, draft-ietf-sipreplaces-03.txt
- Internet Media Type message/sipfrag, RFC 3420
- Call Transfer module - SIP Call Control – Transfer, draft-ietf-sipping-cctransfer-01.txt
- SIP Extension for Instant Messaging - RFC 3428

2. The SINAP/SIP Stack is fully complaint with the following Internet drafts:

- SIP: Session Initiation Protocol, RFC 3261, June 2002
- SDP: Session Description Protocol, RFC 2327 (April 1998)
- SDP: Session Description Protocol, draft-ietf-mmusic-sdp-new-04.txt, Nov 2001
- Reliability of Provisional Responses in SIP, RFC 3262, June 2002, J.Rosenberg, H.Schulzrinne
- Caller Callee Preferences, draft-ietf-sip-callerprefs-06.txt, July 2002, J.Rosenberg, H.Schulzrine
- SIP Call Control-Transfer draft, draft-ietf-sip-cc-transfer-05.txt, July 2001
- SIP Offer-Answer Draft, RFC 3264, June 2002, J. Rosenberg, H. Schulzrinne
- SIP Call Transfer Draft, draft-ietf-sip-cc-transfer-04.txt, February 2001, R.Sparks
- SIP Refer Draft, draft-ietf-sip-refer-05.txt, June 2002; draft-ietf-sip-referredby-00.txt, May 2002, R.Sparks
- Backward compatibility for parsing support of draft-ietf-sip-refer-00.txt, July 2001;draft-ietf-sip-refer-01.txt, September 2001, draft-ietf-sip-refer-02.txt, October 2001, draft-ietf-sip-refer-04.txt, May 2002, R.Sparks
- SIP-Specific Event Notification, RFC 3265, June 2002, Adam Roach
- The SIP PROPOSE Method, draft-onghe-mmusic-sip-propose-00.txt, June 1999, Casey Ong, Sha He
- MIME media types for ISUP and QSIG Objects, RFC 3204, June 2002,
- Eric Zimmerer, Jon Peterson, Aparna Vemuri, Lyndon Ong, F. Audet, M. Watson, M. Zonoun
- The SIP Session Timer, draft-ietf-sip-session-timer-09.txt, July 2002
- SIP Extensions for Instant Messaging, draft-ietf-sip-message-05.txt, June 2002
- IPV6 support in SDP, RFC 3266, June 2002, S. Olson, G. Camarillo, A. B. Roach
- A Common Profile for Instant Messaging, draft-ietf-impp-cpim-02.txt, November 2001
- SIP ISUP mapping draft (SIP BCP-T) - draft-ietf-sip-isup-01.txt
- SIP INFO Method, RFC 2976, October 2000, S. Donovan
- URLs for Telephone Calls, RFC 2806, April 2000, A. Vaha-Sipila
- Format for Literal IPv6 Addresses in URL's, RFC 2732, Dec 1999

- Uniform Resource Identifiers (URI): Generic Syntax, RFC 2396 (August 1998)
- SIP Event Package for Message Waiting indication, draft-ietf-sipping-mwi-00.txt, May 2002, Rohan Mahy, Ilya Slain
- Management Information Base for Session Initiation Protocol, draft-ietf-sip-mib-04.txt, Feb 2002, K. Lingle
- Conventions for the use of the Session Description Protocol (SDP) for Asynchronous Transfer Mode (ATM) Bearer Connections, RFC 3108, May 2001, R. Kumar, M. Mostafa
- The Session Initiation Protocol (SIP) "Replaces" Header, draft-ietf-sip-replaces-02.txt, April 2002, Rohan Mahy
- The Session Initiation Protocol UPDATE Method, draft-ietf-sip-updat
- SIP Extensions for Caller Identity and Privacy, draft-ietf-sip-privacy-04.txt, Feb 2002
- SIP Extensions for supporting Distributed Call State, draft-ietf-sip-state-02.txt, August 2001
- SIP Extensions for Media Authorization, draft-ietf-sip-call-auth-06.txt, May 2002
- The SIP proxy-to-proxy extensions to support DCS, draft-dcsgroup-sip-proxy-proxy-06.txt, March 2002
- SIP Extensions for Resource Management, draft-ietf-sip-manyfolks-resource-07.txt, April 2002
- draft-willis-sip-path-08, D. Willis, May 2002
- draft-ietf-sip-reason-01, H. Schulzrinne, May 2002
- draft-ietf-sip-asserted-identity-02, Jun 2002, C. Jennings
- draft-ietf-sip-privacy-general-01, Jun 2002, J. Peterson

# Index

## P



## Q



## R



## S



## T



## U



## V