
The SIP Scenario Generator is a freeware program that will run on most computers.

For Windows Platforms, SIP Scenario is an executable program. For other platforms, SIP Scenario is a Perl script and requires the Perl interpreter. The Perl interpreter can be downloaded from many sites. See web sites www.activestate.com or www.perl.com for more information on Perl

There is a summary and a detail description of the SIP Scenario command options included in the SIP Scenario Package (Refer to the file readme.txt).

Background

The SIP Scenario Generator program, alias SIP Scenario, was started because we needed an inexpensive tool to debug and analyze SIP messages traces. We used the freeware tool Ethereal, but it did not display the information as a scenario diagram so it was difficult to understand the message flow. Ethereal was used to trace the ethernet interface and save the captured ethernet packets in a file. A Perl Script file then parsed the capture file and produced a text file. This text file had a very wide width depending on what was being traced, resulting in display problems. HTML formatting and browser technology solved the display problem. SIP Scenario was born.

Generating Capture Files

SIP Scenario uses a libpcap formatted ethernet trace (capture) file. Libpcap files can be generated by Ethereal, Tcpdump, and other ethernet analyzers. The capture file should contain the complete SIP message and may contain any other ethernet traffic. SIP Scenario will find the SIP messages contained in any ethernet trace.

Ethernet Protocol Handling

Ethernet, IP, UDP, and TCP protocols have characteristics that may cause problems. The ethernet frame header can be of lengths, due to different protocol such as 802.1P/Q. IP packets may be fragmented by the network and need re-assembling at the destination side. TCP packets can have several SIP message contained over a logically continuous byte stream. The TCP stream on the network does not necessarily appear as a continuous byte stream. It can be very fragmented, fragments can be sent multiple times and in any order. SIP Scenario understands these protocols and handles them properly.

Ethernet Packet Identification

Every ethernet packet (frame) in the capture file is assigned a FRAME number. The FRAME number identifies the packets position within the capture file.

SIP Call Identification

Every SIP call has a unique identification – the Call-Id header. The Call-Id is used to group the SIP messages into calls. SIP Messages with the same Call-id will be grouped to form a call. A sequence number is assigned to each call. This sequence number will be used to identify a specific call.

Filters

A capture file can get very large and may contain thousands of ethernet packets, many which may not be SIP messages. The problem now arises on how to locate the desired SIP messages.

Filters reduce the quantity of ethernet frames by either including or excluding specific ethernet frames or groups of ethernet frames.

The primary filter is to discard all non-SIP ethernet frames.

Other optional filters are based on:

- The Frame number.
Includes / Excludes a set of ethernet packets (frames) based on its frame number.
- Date/Time
Includes a set of ethernet packets (frames) based on a time range
- UDP / TCP Port numbers.
Includes a set of ethernet packets (frames) that use the specified PORT number for either source or destination.
- Call Number.
Includes / Excludes a set of calls by the call number
- IP Address.
Includes / Excludes any call that has the specified IP address in the IP header in any SIP message that is a part of the call.
- String matching.
Includes any call, where some SIP message in that call has a header that contains the specified String. Matching is case insensitive.
STRING= ray.elliott@ipc.com will find all calls related to me.
- Regular Expression Matching.
Includes any call, where some SIP message in that call has a header that matches the specified (Perl) regular expression. This more powerful than string matching. Perl Expression matching is case insensitive.
This filter could find all calls to me by
EXPRESSION= “^INVITE.*ray.elliott@ipc.com” or
EXPRESSION= “^to.*ray.elliott@ipc.com”

There are command options that control filters.

Output Description

The basic output contains four different sections

Header

The header contains a configurable Name.

The name of the input files,

the Date/time the input file was created (from ethernet capture file),

the date/time the SIP Scenario Generator was executed and

the IP address (and alias) for each column in the SIP Scenario Body

SIP Scenario Body

The SIP Scenario Body contains a line for each SIP message between two columns. The line contains the name of the SIP request or the SIP response.

Each column represents a SIP UA. There is a command that specifies the distance (gap) between columns.

At the end of each line, the call number, the frame number and time can be displayed. There are commands to control the format of this information.

Time can be display in three different ways, date/time, relative time from the start of the first SIP message, and relative between two successive SIP messages.

SIP Message Details

Every SIP message contained in the SIP Scenario is completely displayed. There is a html link from the respective SIP message in the SIP Scenario Body to the details of the SIP message.

Summary Information

The Summary information contains statistics and runtime warnings.

Output Files

There are several different output files generated.

Plain Text File

The basic output is a plain text file. This file can be read by editors and word processors.

Single HTML File

The single html file is a self-contained html file. This version is good for sending problem reports.

Triple Frame HTML Files

The triple frame html files is primary presentation package and is a two-file html package. One file contains three html frames. The first frame is used to display the column headers, the second frame is used to display the SIP Scenarios, and the third frame is used to display the details of the SIP messages. The second file contains the detailed information This version is great for debug and presentations.

HTML formatted SIP Scenarios display each call in a different color. The set of colors that are used can be specified by a command option.

I/O Commands

There are commands to specify the input capture file and the output file name and location.

There is also an include command that will include a file that is treated as a set of commands. The sip_scenario.ini file is the default include file and will always be executed if the file exists in the current directory.

Display Commands

There are a set of commands that control the output for presentation and other purposes.

Comments can be added in the SIP Scenario Body.

SIP messages can be reordered. Sometimes SIP messages do not appear in the order that a presenter may want. There is a command that reorder SIP messages.

There also is a command so that a user can add SIP messages to the trace.

Information Commands

There are several commands that display information about the SIP Scenario Generator.

- Displays the basic syntax
- Displays the comprehensive syntax
- Displays a description of the program
- Display the version of the program
- Display Licensing Information
- Display contact information
- Display Summary information.

Utility Commands

There are a set of utility commands that provide specialized features:

- Merge two capture files together with automatic time synchronization and automatic removal of duplicate SIP message.