近期因业务需要，要针对FreeSwitch进行**[性能测试](javascript:;" \t "_self)**，花了些时间了解，并实施了对应的性能测试；

整理了下**[学习](javascript:;" \t "_self)**及实施过程中遇到的一些问题及解决方法，并分享给大家。（学习过程中，发现挺多人遇到问题或暂无头绪）

主要章节:

1、**[SIPP](javascript:;" \t "_self)**概述&参考资料

2、SIPP下载安装

3、UAC

4、UAS

5、UAC+UAS

6、注意事项

7、错误记录

## [SIPP概述](javascript:;)

SIPp是一个**[测试](javascript:;" \t "_self)**SIP协议性能的工具软件。这是一个GPL的开放源码软件。

它包含了一些基本的SipStone用户代理**[工作](javascript:;" \t "_self)**流程（UAC和UAS），并可使用INVITE和BYE建立和释放多个呼叫。它也可以读XML的场景文件，即描述任何**[性能测试](javascript:;" \t "_self)**的配置文件。它能动态显示测试运行的统计数据（呼叫速率、信号来回的延迟，以及 消息统计）。周期性地把CSV统计数据转储，在多个套接字上的TCP和UDP，利用重新传输管理的多路复用。在场景定义文件中可以使用正规表达式，动态调整呼叫速率。

SIPp可以用来测试许多真实的SIP设备，如SIP代理，B2BUAs,SIP媒体服务器，SIP/x网关，SIP PBX，等等，它也可以模仿上千个SIP代理呼叫你的SIP系统。

SIPp的网址：<http://sipp.sourceforge.net/>，这里可以下载最新版的SIPp软件，并且有英文资料可供查阅。

SIP协议采用Client/Server模型。每一个请求（Request）触发服务器的一个操作；每个操作被称为方法（Method）；每个请求除了指明具体方法外，还携带了一系列的头域（Header field）,这些头域携带了各种业务信息；除了头域外，消息中还可以携带任意类型的消息体（Body），呼叫中最常使用的SDP信息就是通过消息体在Client和Server之间传递的。Server从接收到请求到处理完毕，要回复多个临时响应(Response)，和一个终结响应(Response)，终结响应有且只有一个。请求和他的所有响应构成一个事务（Transaction），一个完整的呼叫过程包含多个事务，比如呼叫建立和呼叫释放就是两个相互独立的事务。  
用户代理（User Agent）是发起或者接收呼叫的逻辑实体。用户代理客户端-**UAC**（User Agent Client），用于发起请求；用户代理服务器-**UAS**（User Agent Server），用于接收请求。**UAC**/**UAS**的划分是针对一个事务的。在一个呼叫中的多个事务里，**UAC**和**UAS**的角色是可以互换的。例如在A和B的呼叫中，A向B发起呼叫，在呼叫建立的事务中，A是**UAC**，B是**UAS**；呼叫结束时，B先挂机，在呼叫释放的事务中，B是**UAC**，A是**UAS**。换句话说，每个一般的UA都是**UAS**和**UAC**的结合体。UA的实际物理形态有：IP Phone，SoftPhone，GateWay......  
Proxy Server作为一个网络逻辑实体代理客户端转发请求或者响应；同Proxy Server类似的还有一种设备是B2BUA，顾名思义，就是背背的两个UA组成的一个逻辑实体，它作为**UAS**终结一个事务，同时作为**UAC**发起另外一个事务。Proxy Server同B2BUA相比，Proxy Server是一个事务传递过程中的中间节点，而B2BUA将一个事务转变成另一个事务。  
在SIP组网中还包括Location Server、Registrar、Redirect Server，分别负责维护地址映射表，注册管理，呼叫重定向。他们和Proxy Server 可以在同一台设备上也可以运行于不同的设备上。SIP Server是Proxy Server、Location Server、Registrar、Redirect Server的总称。  
SIP Server采用B2BUA模型。接受请求端为**UAS**端，代理转发或主动发起请求端为**UAC**端，整个SIPServr为**UAC**/**UAS**的组合体。通过**UAC**/**UAS**之间消息交互完成会话的建立、改变、结束的阶段。SIP Server协助网关交换媒体信息，但不参与会话建立后媒体流传输。

## [参考资料](javascript:;)

### sipp官网：

http://sipp.sourceforge.net/

### 官方手册：

http://sipp.sourceforge.net/doc3.3/reference.html

使用sipp对FreeSwitch进行测试（**[Linux](javascript:;" \t "_self)**环境，会议模式）：

http://blog.csdn.net/castle07/article/details/8820026

### Sipp：

http://blog.csdn.net/hanruikai/article/details/8024924

### 百问FreeSwitch.pdf

### sipp最新版本**[下载](javascript:;" \t "_self)**：

http://garr.dl.sourceforge.net/project/sipp/sipp/3.3/sipp-3.3.tar.gz

**SIPp支持四种不同的[安装](javascript:;" \t "_self)模式：**

1) 没有TLS支持与密码验证支持：

a) # tar xzvf sipp-xxx.tar.gz

b) # cd sipp-xxx

c) # makeMake

出来的sipp文件就是一个可执行的文件，只需要搭配场景xml文件与csv文件即可进行SIP[**测试**](javascript:;)，这是最常用的安装。

2) 拥有TLS支持与密码验证支持，但是不支PCAP语音播放：

a) # tar xzvf sipp-xxx.tar.gz

b) # cd sipp-xxx

c) # make ossl

这样编译出来的文件就加入了TLS至于与密码验证支持功能sipp软件了。

3) 支持PCAP Play，但是没有密码验证支持：(PCAP Play即为可以进行RTP语音，但是没有407 AUTH验证)

a) # tar xzvf sipp-xxx.tar.gz

b) # cd sipp-xxx

c) # make pcapplay

4) 支持PCAP 声音文件播放，而且支持密码验证支持：(支持407 auth验证支持)

a) # tar xzvf sipp-xxx.tar

b) # cd sipp-xxx

c) # make pcapplay\_ossl

本文使用安装方法4：

# tar xzvf sipp-3.3.tar.gz

# cd sipp-3.3

# make pcapplay\_ossl

**注：make不成功时，确认是否安装：libpcap、libpcap-devel、ncurses-devel、ncurses**

**见错误记录1、2；**

确认安装成功：

./sipp -h

## 使用说明

用**[SIPp](javascript:;" \t "_self)**做**[测试](javascript:;" \t "_self)**的时候需要准备以下几个文件：

### [**UAC**](javascript:;)端：uac.sh,  uac.xml,  uac.csv

* uac.sh：调用sipp命令，并传入相应参数脚本文件，模拟UAC（例如主叫），当然也可不准备此文件直接输入sipp命令执行程序，但是写成文件执行更加方便可靠；
* uac.xml：根据需要编写的uac侧的sip信号流程；
* uac.csv：用于uac.xml需要引入的相应数据；

### UAS端：reg.sh,  reg.xml,  uas\_noreg.sh,  uas\_noreg.xml,  uas.csv

* reg.sh：调用sipp命令，并传入相应参数的脚本文件，模拟UAS注册，当然也可不准备此文件直接输入sipp命令执行程序，但是写成文件执行更加方便可靠；
* reg.xml：根据需要编写的注册流程，主要配合uas流程使用；
* uas\_noreg.sh：调用sipp命令，并传入相应参数的脚本文件，模拟UAS（例如被叫），当然也可不准备此文件直接输入sipp命令执行程序，但是写成文件执行更加方便可靠。
* uas\_noreg.xml：根据需要编写的uas侧的sip信号流程。
* uas.csv：用于reg.xml和uas.xml中需要引入的相应数据。

注意：uas中，reg.xml和uas\_noreg.xml中的内容不能合并在同一个xml中，或者会导致收不到invite；

**使用下述命令，可以直接导出场景模板，再进行二次修改；**

./sipp -sd branchc > branchc.xml

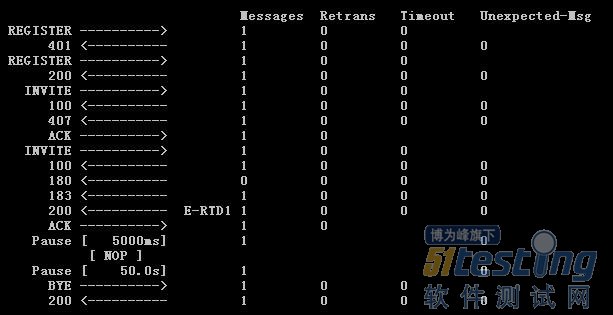
## 详细说明

### 1、UAC

调试uac流程时，推荐调试方案：uac主叫+eyebeam被叫（或其它VOIP工具）；

待uac和uas单方都确认调试通过后，再使用：uac主叫+uas被叫；

**uac.xml流程：**

[](http://www.51testing.com/batch.download.php?aid=43330)

#### uac.sh文件内容如下：

./sipp -i 192.168.146.17 -sf uac.xml -inf uac.csv 192.168.146.11:5060 -l 1 -trace\_msg -trace\_screen -trace\_err -p 12345 -m 1 -aa

参数解释：

-i: Set the local IP address for 'Contact:','Via:', and 'From:' headers.

Default is primary host IP address.

为'Contact:','Via:', 和 'From:' 头部信息设置本地IP，在脚本中用[local\_ip]引入

-sf: Loads an alternate xml scenario file.

To learn more about XML scenario syntax, use the -sd option to dump embedded scenarios.

 They contain all the necessary help.

引入脚本文件，根据需要模拟的呼叫流程编写

-inf: Inject values from an external CSV file during calls into the scenarios.

First line of this file say whether the data is to be read in sequence (SEQUENTIAL), random (RANDOM), or user(USER) order.

Each line corresponds to one call and has one or more ';' delimited data fields. Those fields can be referred as [field0], [field1], ... in the xml scenario file.

Several CSV files can be used simultaneously (syntax:-inf f1.csv -inf f2.csv ...)

在通话场景中使用外部csv文件引入数据；

文件的第一行说明了后面数据的读入方式，常用的有：顺序 (SEQUENTIAL), 随机 (RANDOM), 或用户(USER)顺序；

第一行对应一个通话，它们由一个或多个’;’分隔数据字段，这些字段可以在xml场景文件中使用[field0], [field1], ...来调用；

多个csv文件，可以同时使用（语法：-inf f1.csv -inf f2.csv ...）

192.168.146.11:5060

[**Freeswitch**](javascript:;)服务端IP及freeswitch使用的端口；

-l: Set the maximum number of simultaneous calls.

Once this limit is reached, traffic is decreased until the number of open calls goes down.

Default:(3 \* call\_duration (s) \* rate).

设置同时呼叫的最大数目；

一旦达到此值，流量将被限制直到打开的通话数下降；

默认值：3\*call\_duration(s)\*rate

-p: Set the local port number.  Default is a random free port chosen by the system.

设置本地端口号，默认是由系统随机选择空闲的端口号；

-m:Stop the test and exit when 'calls' calls are processed

当设置的通话数完成时，停止测试并退出；

-aa: Enable automatic 200 OK answer for INFO, UPDATE and NOTIFY messages.

针对INFO, UPDATE 和 NOTIFY消息，进行200 OK自动回复应答；

-trace\_msg:

Displays sent and received SIP messages in <scenario file name>\_<pid>\_messages.log

在<场景文件名>\_<pid>\_messages.log中显示发送和接收的SIP消息；调试时可增加，正试**[性能测试](javascript:;" \t "_self)**时，可取消，以免[**日志**](javascript:;)量太大影响本地性能；

-trace\_screen:

Dump statistic screens in the<scenario\_name>\_<pid>\_screens.log file when quitting SIPp.

Useful to get a final status report in background mode (-bg option).

在退出SIPp时，把屏蔽上的统计信息写入<场景名>\_<pid>\_screens.log文件中；

在后台模式（-bq选项）时，这对于得到最终状态报告很有用；

-trace\_err:

Trace all unexpected messages in <scenario filename>\_<pid>\_errors.log.

跟踪所有非期望的消息到<场景文件名>\_<pid>\_errors.log；

测试时，有可能要指定发送呼叫频率，可以指定参数：

-r 1 -rp 3000：每三秒钟发一个呼，如果不指定，默认为1秒加载10用户；

性能测试过程中，一直要修改测试用户数，可考虑参照如下修改uac.sh脚本，-l及-m参数使用变更代替：

#### uac.sh

#!/bin/bash

# Author:fairylly

#执行脚本时，未输入参数，提示：please input call number!，并退出

if [[ $# -eq 0 ]]

then

        echo "please input call number!"

        exit 1

fi

#m变量使用命令行传递的第一个位置参数

m=$1

./sipp -i 192.168.146.17 -sf uac.xml -inf uac.csv 192.168.146.11:5060 -l $m -trace\_screen -trace\_err -p 12345 -m $m -aa

使用方法：./uac.sh 100，表示最大同时呼叫100个用户；

#### [**uac**](javascript:;).xml文件内容如下：

<?xml version="1.0" encoding="ISO-8859-1" ?>

<!DOCTYPE scenario SYSTEM "[**sipp**](javascript:;).dtd">

<scenario name="UAC with media">

  <send retrans="500">

     <![CDATA[

       REGISTER sip:[remote\_ip] SIP/2.0

       Via: SIP/2.0/[transport] [local\_ip]:[local\_port];branch=[branch]

       From: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>;tag=[call\_number]

       To: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>

       Call-ID: [call\_id]

       CSeq: 1 REGISTER

       Contact: sip:[field0]@[local\_ip]:[local\_port]

       Content-Length: 0

       Expires: 3600

     ]]>

  </send>

  <recv response="401" auth="true" next="1"></recv>

  <!--   send register with authentication messages -->

  <label id="1"/>

  <send retrans="500">

     <![CDATA[

       REGISTER sip:[field0]@[remote\_ip]:[remote\_port] SIP/2.0

       Via: SIP/2.0/[transport] [local\_ip]:[local\_port]

       From: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>;tag=[call\_number]

       To: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>

       Call-ID: [call\_id]

       CSeq: 2 REGISTER

       Contact: sip:[field0]@[local\_ip]:[local\_port]

       [field2]

       Content-Length: 0

       Expires: 3600

     ]]>

  </send>

  <recv response="200" next="2">

  </recv>

  <!-- invite(1) -->

  <label id="2"/>

  <send retrans="500">

     <![CDATA[

        INVITE sip:[field1]@[remote\_ip]:[remote\_port] SIP/2.0

        Via: SIP/2.0/[transport] [local\_ip]:[local\_port];branch=[branch]

        From: [field0] <sip:[field0]@[local\_ip]:[local\_port]>;tag=[call\_number]

        To: [field1] <sip:[field1]@[remote\_ip]:[remote\_port]>

        Call-ID: [call\_id]

        CSeq: 1 INVITE

        Contact: sip:[field0]@[local\_ip]:[local\_port]

        Max-Forwards: 70

        Subject: Performance Test

        Content-Type: application/sdp

        Content-Length: [len]

        v=0

        o=user1 53655765 2353687637 IN IP[local\_ip\_type] [local\_ip]

        s=-

b=AS:84

        c=IN IP[local\_ip\_type] [local\_ip]

        t=0 0

a=X-nat:0

        m=audio [auto\_media\_port] RTP/AVP 98

        a=sendrecv

        a=rtpmap:98 iLBC/8000

        a=fmtp:98 mod=30

     ]]>

  </send>

  <recv response="100" optional="true"></recv>

  <recv response="407" auth="true" next="3"></recv>

  <!-- ack(1) -->

  <label id="3"/>

  <send>

    <![CDATA[

ACK sip:[field1]@[remote\_ip]:[remote\_port] SIP/2.0

[last\_Via:]

From: [field0] <sip:[field0]@[local\_ip]:[local\_port]>;tag=[call\_number]

To: [field1] <sip:[field1]@[remote\_ip]:[remote\_port]>[peer\_tag\_param]

Call-ID: [call\_id]

CSeq: 1 ACK

Contact: sip:[field0]@[local\_ip]:[local\_port]

Max-Forwards: 70

Subject: Performance Test

Content-Length: 0

     ]]>

  </send>

  <!-- invite(2) -->

  <send retrans="500">

     <![CDATA[

INVITE sip:[field1]@[remote\_ip]:[remote\_port] SIP/2.0

        Via: SIP/2.0/[transport] [local\_ip]:[local\_port];branch=[branch]

        From: [field0] <sip:[field0]@[local\_ip]:[local\_port]>;tag=[call\_number]

        To: [field1] <sip:[field1]@[remote\_ip]:[remote\_port]>

        Call-ID: [call\_id]

        CSeq: 2 INVITE

        Contact: sip:[field0]@[local\_ip]:[local\_port]

        Max-Forwards: 70

        Subject: Performance Test

[field2]

        Content-Type: application/sdp

        Content-Length: [len]

        v=0

        o=user1 53655765 2353687637 IN IP[local\_ip\_type] [local\_ip]

        s=-

b=AS:84

        c=IN IP[local\_ip\_type] [local\_ip]

        t=0 0

a=X-nat:0

        m=audio [auto\_media\_port] RTP/AVP 98

        a=sendrecv

        a=rtpmap:98 iLBC/8000

        a=fmtp:98 mod=30

     ]]>

  </send>

  <recv response="100" optional="true"></recv>

  <recv response="180" optional="true"></recv>

  <recv response="183" optional="true"></recv>

  <recv response="200" rtd="true" next="4"></recv>

 <!-- ack(2) -->

  <label id="4"/>

  <send>

    <![CDATA[

        ACK sip:[field1]@[remote\_ip]:[remote\_port] SIP/2.0

[last\_Via:]

        From: [field0] <sip:[field0]@[local\_ip]:[local\_port]>;tag=[call\_number]

        To: [field1] <sip:[field1]@[remote\_ip]:[remote\_port]>[peer\_tag\_param]

        Call-ID: [call\_id]

        CSeq: 2 ACK

        Contact: sip:[field0]@[local\_ip]:[local\_port]

        Max-Forwards: 70

        Subject: Performance Test

        Content-Length: 0

    ]]>

  </send>

  <pause milliseconds="5000"/>

  <nop>

     <action>

       <exec play\_pcap\_audio="pcap/cc.pcap"/>

     </action>

  </nop>

  <pause milliseconds="50000"/>

  <!-- bye -->

  <send retrans="500">

    <![CDATA[

BYE sip:[field1]@[remote\_ip]:[remote\_port] SIP/2.0

Via: SIP/2.0/[transport] [local\_ip]:[local\_port];branch=[branch]

From: [field0] <sip:[field0]@[local\_ip]:[local\_port]>;tag=[call\_number]

To: [field1] <sip:[field1]@[remote\_ip]:[remote\_port]>[peer\_tag\_param]

Call-ID: [call\_id]

CSeq: 2 BYE

Contact: sip:[field0]@[local\_ip]:[local\_port]

Max-Forwards: 70

Subject: Performance Test

Content-Length: 0

    ]]>

  </send>

  <recv response="200" crlf="true"></recv>

  <!-- definition of the response time repartition table (unit is ms)   -->

  <ResponseTimeRepartition value="10, 20, 30, 40, 50, 100, 150, 200"/>

  <!-- definition of the call length repartition table (unit is ms)     -->

  <CallLengthRepartition value="10, 50, 100, 500, 1000, 5000, 10000"/>

</scenario>

**说明：**

pcap/cc.pcap：指定播放的录音文件，文件必须存在，方法见：四、注意事项-5、pcap语音数据包；

#### uac.csv文件内容如下：

SEQUENTIAL

152;19002;[authentication username=152 password=7d83c157ea9c4c21da1a0e7ab05c21b9]

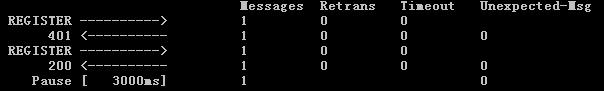
153;138721;[authentication username=153 password=7d83c157ea9c4c21da1a0e7ab05c21b9]

UAS  
调试**[uas](javascript:;" \t "_self)**流程时，推荐调试方案：eyebeam主叫（或其它VOIP工具）+uas被叫；

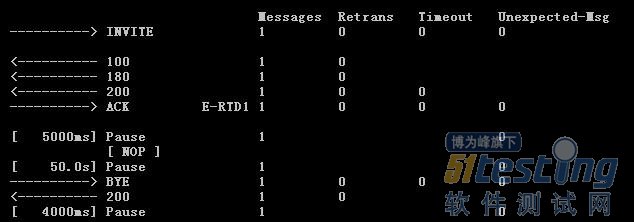
待uac和uas单方都确认调试通过后，再使用：uac主叫+uas被叫；

### 注册

#### reg.xml流程

[](http://www.51testing.com/batch.download.php?aid=43331)

uas.xml流程：

[](http://www.51testing.com/batch.download.php?aid=43332)

**reg.sh文件内容如下：**

./sipp -i 192.168.146.101 -sf reg.xml -inf uas.csv 192.168.148.43:5060 -l 1 -trace\_msg -trace\_screen -trace\_err -p 12346 -m 1 -aa

参数解释同uac流程；

[**性能测试**](javascript:;)过程中，一直要修改测试用户数，可考虑参照如下修改reg.sh脚本，-l及-m参数使用变更代替：

#### reg.sh

#!/bin/bash

# Author:fairylly

# 执行脚本时，未输入参数，提示：please input call number!，并退出

if [[ $# -eq 0 ]]

then

        echo "please input call number!"

        exit 1

fi

# m变量使用命令行传递的第一个位置参数

m=$1

./sipp -i 192.168.146.101 -sf reg.xml -inf uas.csv 192.168.148.43:5060 -l 1 -trace\_screen -trace\_err -p 12346 -m 1 -aa

使用方法：./reg.sh 100，表示最大同时注册100个用户；

#### reg.xml文件内容如下：

<?xml version="1.0" encoding="ISO-8859-1" ?>

<!DOCTYPE scenario SYSTEM "[**sipp**](javascript:;).dtd">

<scenario name="Register">

    <!-- send register -->

    <send retrans="500">

     <![CDATA[

       REGISTER sip:[remote\_ip] SIP/2.0

       Via: SIP/2.0/[transport] [local\_ip]:[local\_port];branch=[branch]

       From: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>;tag=[call\_number]

       To: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>

       Call-ID: [call\_id]

       CSeq: 1 REGISTER

       Contact: sip:[field0]@[local\_ip]:[local\_port]

       Content-Length: 0

       Expires: 3600

     ]]>

  </send>

  <recv response="401" auth="true" next="1"></recv>

  <!--   send register with authentication messages -->

  <label id="1"/>

  <send retrans="500">

     <![CDATA[

       REGISTER sip:[field0]@[remote\_ip]:[remote\_port] SIP/2.0

       Via: SIP/2.0/[transport] [local\_ip]:[local\_port]

       From: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>;tag=[call\_number]

       To: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>

       Call-ID: [call\_id]

       CSeq: 2 REGISTER

       Contact: sip:[field0]@[local\_ip]:[local\_port]

       [field2]

       Content-Length: 0

       Expires: 3600

     ]]>

  </send>

  <recv response="200"></recv>

  <pause milliseconds="3000"/>

  <!-- definition of the response time repartition table (unit is ms)   -->

  <ResponseTimeRepartition value="10, 20, 30, 40, 50, 100, 150, 200"/>

  <!-- definition of the call length repartition table (unit is ms)     -->

  <CallLengthRepartition value="10, 50, 100, 500, 1000, 5000, 10000"/>

</scenario>

### UAS

#### uas\_noreg.sh文件内容如下：

./sipp -i 192.168.146.101 -sf uas\_noreg.xml -inf uas.csv 192.168.148.43:5060 -l 1 -trace\_msg -trace\_screen -trace\_err -p 12346 -m 1 -aa

参数解释同uac流程；

性能测试过程中，一直要修改测试用户数，可考虑参照如下修改uas\_noreg.sh脚本，-l及-m参数使用变更代替：

#### uas\_noreg.sh

#!/bin/bash

# Author:fairylly

# 执行脚本时，未输入参数，提示：please input call number!，并退出

if [[ $# -eq 0 ]]

then

        echo "please input call number!"

        exit 1

fi

# m变量使用命令行传递的第一个位置参数

m=$1

./sipp -i 192.168.146.101 -sf uas\_noreg.xml -inf uas.csv 192.168.148.43:5060 -l $m -trace\_screen -trace\_err -p 12346 -m $m -aa

使用方法：./uas\_noreg.sh 100，表示最大同时被叫100个用户；

#### uas\_noreg.xml文件内容如下：

<?xml version="1.0" encoding="ISO-8859-1" ?>

<!DOCTYPE scenario SYSTEM "[**sipp**](javascript:;).dtd">

<!-- This program is free software; you can redistribute it and/or      -->

<!-- modify it under the terms of the GNU General Public License as     -->

<!-- published by the Free Software Foundation; either version 2 of the -->

<!-- License, or (at your option) any later version.                    -->

<!--                                                                    -->

<!-- This program is distributed in the hope that it will be useful,    -->

<!-- but WITHOUT ANY WARRANTY; without even the implied warranty of     -->

<!-- MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the      -->

<!-- GNU General Public License for more details.                       -->

<!--                                                                    -->

<!-- You should have received a copy of the GNU General Public License  -->

<!-- along with this program; if not, write to the                      -->

<!-- Free Software Foundation, Inc.,                                    -->

<!-- 59 Temple Place, Suite 330, Boston, MA  02111-1307 USA             -->

<!--                                                                    -->

<!--                 Sipp default 'uas' scenario.                       -->

<!--                                                                    -->

<scenario name="Basic UAS responder">

  <!-- By adding rrs="true" (Record Route Sets), the route sets         -->

  <!-- are saved and used for following messages sent. Useful to test   -->

  <!-- against stateful SIP proxies/B2BUAs.                             -->

  <recv request="INVITE" crlf="true">

  </recv>

  <!-- The '[last\_\*]' keyword is replaced automatically by the          -->

  <!-- specified header if it was present in the last message received  -->

  <!-- (except if it was a retransmission). If the header was not       -->

  <!-- present or if no message has been received, the '[last\_\*]'       -->

  <!-- keyword is discarded, and all bytes until the end of the line    -->

  <!-- are also discarded.                                              -->

  <!--                                                                  -->

  <!-- If the specified header was present several times in the         -->

  <!-- message, all occurences are concatenated (CRLF seperated)        -->

  <!-- to be used in place of the '[last\_\*]' keyword.                   -->

  <send>

    <![CDATA[

      SIP/2.0 100 Trying

      [last\_Via:]

      [last\_From:]

      [last\_To:];tag=[call\_number]

      [last\_Call-ID:]

      [last\_CSeq:]

      Contact: <sip:[local\_ip]:[local\_port];transport=[transport]>

      Content-Length: 0

    ]]>

  </send>

  <send>

    <![CDATA[

      SIP/2.0 180 Ringing

      [last\_Via:]

      [last\_From:]

      [last\_To:];tag=[call\_number]

      [last\_Call-ID:]

      [last\_CSeq:]

      Contact: <sip:[local\_ip]:[local\_port];transport=[transport]>

      Content-Length: 0

    ]]>

  </send>

  <send retrans="500">

   <![CDATA[

SIP/2.0 200 OK

[last\_Via:]

[last\_From:]

[last\_To:];tag=[call\_number]

[last\_Call-ID:]

[last\_CSeq:]

Contact: <sip:[local\_ip]:[local\_port];transport=[transport]>

Content-Type: application/sdp

Content-Length: [len]

    v=0

    o=user1 53655765 2353687637 IN IP[local\_ip\_type] [local\_ip]

    s=-

    b=AS:84

    c=IN IP[local\_ip\_type] [local\_ip]

    t=0 0

    a=X-nat:0

    m=audio [auto\_media\_port] RTP/AVP 98

    c=IN IP[local\_ip\_type] [local\_ip]

    b=TIAS:64000

    a=rtcp:6001 IN IP[local\_ip\_type] [local\_ip]

    a=sendrecv

    a=rtpmap:98 iLBC/8000

    a=fmtp:98 mod=30

    ]]>

  </send>

  <recv request="ACK"

        rtd="true"

        crlf="true">

  </recv>

  <pause milliseconds="5000"/>

  <nop>

    <action>

      <exec play\_pcap\_audio="pcap/cc.pcap"/>

    </action>

  </nop>

  <pause milliseconds="50000"/>

  <recv request="BYE">

  </recv>

  <send>

    <![CDATA[

      SIP/2.0 200 OK

      [last\_Via:]

      [last\_From:]

      [last\_To:]

      [last\_Call-ID:]

      [last\_CSeq:]

      Contact: <sip:[local\_ip]:[local\_port];transport=[transport]>

      Content-Length: 0

    ]]>

  </send>

  <!-- Keep the call open for a while in case the 200 is lost to be     -->

  <!-- able to retransmit it if we receive the BYE again.               -->

  <pause milliseconds="4000"/>

  <!-- definition of the response time repartition table (unit is ms)   -->

  <ResponseTimeRepartition value="10, 20, 30, 40, 50, 100, 150, 200"/>

  <!-- definition of the call length repartition table (unit is ms)     -->

  <CallLengthRepartition value="10, 50, 100, 500, 1000, 5000, 10000"/>

</scenario>

**说明：**

pcap/cc.pcap：指定播放的录音文件，文件必须存在，方法见：四、注意事项-5、pcap语音数据包；

#### [**uas**](javascript:;).csv文件内容如下：

SEQUENTIAL

19002;[authentication username=19002 password=7d83c157ea9c4c21da1a0e7ab05c21b9]

138721;[authentication username=138721 password=7d83c157ea9c4c21da1a0e7ab05c21b9]

进行**[性能测试](javascript:;" \t "_self)**时，要使用UAC+UAS流程；建议使用两台**[sipp](javascript:;" \t "_self)**客户端，一台担任**[UAC](javascript:;" \t "_self)**，一台担任**[UAS](javascript:;" \t "_self)**。

操作步骤：

1）被叫方：进行被叫号码注册操作./reg.sh 100

2）被叫方：启动被叫uas流程./uas\_noreg.sh 100

3）主叫方：启动主叫uac流程./uac.sh 100

注：100为用户数

## 注意事项

### 1、系统配置

如果您的系统不支持足够的文件描述符，在使用TCP/ TLS模式进行多人同时通话时，可能会遇到问题。

你有两种方法来克服这个限制：要么使用-max\_socket的命令行选项，或更改你的系统的限制。

增大文件描述符限制：

# vi /etc/security/limits.conf

增加：

soft nofile 1024

hard nofile 65535

# vi /etc/pam.d/login

增加：

session required /lib/security/pam\_limits.so

echo 65535> /proc/sys/fs/file-max

ulimit -n unlimited

退出连接再重新登录，使设置生效；

### 2、regsiter注册

注册时，from/to要设置一样，否则会注册不成功；

### 3、pcap语音数据包

#### 1）nop后要增加对应的pause时间

  <nop>

    <action>

      <exec play\_pcap\_audio="pcap/cc.pcap"/>

    </action>

  </nop>

 <pause milliseconds="15000"/>

pcap/cc.pcap录音文件播放需要多长时间，接下去的pause设置多少时间；如果未设置或设置太短，有可能未播放或播放不完整；

#### 2）录制pcap文件时注意开始及结束时间

使用wireshark进行抓包，抓包过滤规则：udp and src host 192.168.134.120

目的是只抓192.168.134.120发出来的包。

已经建立起通话连接后，再开始抓包；

关闭通话连接前，先结束抓包；

这样抓的包才全是rtp数据包；

如果提前开始或延后结束，vi查看抓包文件，头部或结尾会有sip协议数据；

### 4、errors.log[****日志****](javascript:;)中的NOTIFY

查看errors.log日志，里面有很多针对NOTIFY的错误，如下：

2013-11-22      16:53:22:929    1385110402.929208: Discarding message which can't be mapped to a known SIPp call:

NOTIFY sip:19002@192.168.146.101:12346 SIP/2.0

Via: SIP/2.0/UDP 192.168.148.43;rport;branch=z9hG4bKSKpUm8BB4pa1S

Route: <sip:19002@192.168.146.101:12346>

Max-Forwards: 70

From: <sip:19002@192.168.148.43>;tag=cSpHa5BgB3pHD

To: <sip:19002@192.168.148.43>

Call-ID: 9ce1be85-cdf6-1231-91b1-005056b07ef2

CSeq: 52219186 NOTIFY

Contact: <sip:mod\_sofia@192.168.148.43:5060>

User-Agent: FreeSWITCH-mod\_sofia/1.2.13+git~20131007T141815Z~cbdb174965

Allow: INVITE, ACK, BYE, CANCEL, OPTIONS, MESSAGE, INFO, UPDATE, REGISTER, REFER, NOTIFY, PUBLISH, SUBSCRIBE

Supported: timer, precondition, path, replaces

Event: message-summary

Allow-Events: talk, hold, conference, presence, dialog, line-seize, call-info, sla, include-session-description, presence.winfo, message-summary, refer

Subscription-State: terminated;reason=noresource

Content-Type: application/simple-message-summary

Content-Length: 67

Messages-Waiting: no

Message-Account: sip:19002@192.168.148.43

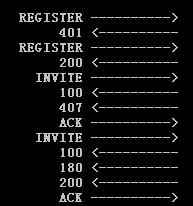
这个是因为**[sipp](javascript:;" \t "_self)**的xml脚本未对接收到的NOTIFY进行处理，可以在启动sipp时，使用-aa参数，自动对NOTIFY进行回复；

./sipp -h

说明如下：

-aa : Enable automatic 200 OK answer for INFO, UPDATE and NOTIFY messages.

### 5、对407进行ack确认后，还一直接收到407消息

[](http://www.51testing.com/batch.download.php?aid=43333)

发送第一个INVITE后，会接收407要求认证，发送ACK进行确认，后再发送第二个带认证信息的INVITE；

但后续还一直收到服务端发送的407；

原因是ACK数据包设置的via为：

Via: SIP/2.0/[transport] [local\_ip]:[local\_port];branch=[branch]

修改为：

[last\_Via:]

后正常；

branch=[branch]这个会重新生成一个branch，但ACK是要对接收的407进行确认，所以branch要和407数据包的branch相同，所以使用[last\_Via:]；

**注意：第二INVITE后，对接收到的200进行ACK时，也是一样设置：[last\_Via:]；**

### 6、Send数据包中的CSeq设置

Send的数据包，CSeq每个都递增，例：

       CSeq: 1 REGISTER

       CSeq: 2 REGISTER

        CSeq: 3 INVITE

        CSeq: 4 ACK

        CSeq: 5 INVITE

        CSeq: 6 ACK

        CSeq: 7 BYE

## 

会出现如下错误：

Reason: SIP;cause=408;text="ACK Timeout"

Aborting call on an unexpected BYE for call: 1-3328@192.168.146.101.

修改为下述内容后正常：

       CSeq: 1 REGISTER

       CSeq: 2 REGISTER

        CSeq: 1 INVITE

        CSeq: 1 ACK

        CSeq: 2 INVITE

        CSeq: 2 ACK

        CSeq: 2 BYE（这个用1会出现500错误，用2、3正常，不太确认原因）

### 7、UAS流程中的注册操作

UAC流程中，可以把注册和呼叫流程放在一个xml场景文件中；

但UAS流程中，放在一起，会出现invite无法正常接收的问题，

日志：Discarding message which can't be mapped to a known SIPp call

所以uas中，要把注册流程写在一个xml场景中，被叫流程写在另一个xml场景中。

### 1、make pcapplay\_ossl提示：error: curses.h: No such file or directory

**问题描述：**

# make pcapplay\_ossl

[**错误**](javascript:;)信息如下：

In file included from message.cpp:39:

[**sipp**](javascript:;).hpp:57:20: error: curses.h: No such file or directory

。。。

make[1]: \*\*\* [message.o] Error 1

make[1]: Leaving directory `/usr/local/luly/sipp-3.3'

make: \*\*\* [pcapplay\_ossl] Error 2

**解决方法：**

yum install ncurses-devel ncurses

### 2、make pcapplay\_ossl提示：error: pcap.h: No such file or directory

**问题描述：**

# make pcapplay\_ossl

错误信息如下：

send\_packets.c:44:18: error: pcap.h: No such file or directory

make[1]: \*\*\* [send\_packets.o] Error 1

make[1]: Leaving directory `/usr/local/sipp-3.3'

make: \*\*\* [pcapplay\_ossl] Error 2

**解决方法：**

yum install libpcap libpcap-devel

### 3、sipp提示：Can create thread to send RTP packets.

**问题描述：**

运行sipp时，提示：

2013-11-26      14:48:04:100    1385448484.100398: Can create thread to send RTP packets.

sipp: There were more errors, see 'uac\_30796\_errors.log' file

**解决方法：**

修改服务器线程数相关配置；

ulimit -s 1024

线程栈大小，以 Kbytes 为单位。

### 4、Sipp提示：Reason: SIP;cause=606;text="USER\_NOT\_REGISTERED"

**问题描述：**

Sipp被叫注册后，启动uas流程，uac呼叫时，返回错误：

Reason: SIP;cause=606;text="USER\_NOT\_REGISTERED"

查询服务端**[日志](javascript:;" \t "_self)**：第二次REGISTER时，没有指明Expires，从而使用默认的30S；但xml文件中有指明Expires为3600；如下：

  <send retrans="500">

    <![CDATA[

        REGISTER sip:[field0]@[remote\_ip]:[remote\_port] SIP/2.0

        Via: SIP/2.0/[transport] [local\_ip]:[local\_port]

        From: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>;tag=[call\_number]

        To: [field0] <sip:[field0]@[remote\_ip]:[remote\_port]>

        Call-ID: [call\_id]

        CSeq: 2 REGISTER

        Contact: sip:[field0]@[local\_ip]:[local\_port]

        [field1]

        Content-Length: [len]

        Expires: 3600

    ]]>

  </send>

注：以上为网上常见示例；

**解决方法：**

和1 REGISTER不同的是：Content-Length: [len]，这里指定为[len]，修改[len]为0，发送的Expires可识别；或修改Expires到Content-Length前一行，也正常；

### 5、Sipp从服务器中转传输rtp数据包时，听不到pcap录音

**问题描述：**

Sipp从服务器中转传输rtp数据包时，被叫方听不到pcap录音

**解决方法：**

原指定的编码为104 ilbc，修改为98 ilbc后正常；

uac.xml/uas.xml文件中，涉及rtp数据包传输编码指定部份，要和服务端一致；具体信息可和服务端人员确认；

    b=AS:84

    c=IN IP[local\_ip\_type] [local\_ip]

    t=0 0

    a=X-nat:0

    m=audio [auto\_media\_port] RTP/AVP 98

    c=IN IP[local\_ip\_type] [local\_ip]

    b=TIAS:64000

    a=rtcp:6001 IN IP[local\_ip\_type] [local\_ip]

    a=sendrecv

    a=rtpmap:98 iLBC/8000

    a=fmtp:98 mod=30