Linux-SIPp3.3测试FreeSwitch

——fairylly

## 一、SIPP 概述

* SIPp 是一个测试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。

## 二、SIPP 下载安装

SIPp 最新版本下载：

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，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：(PCAPPlay 即为可以进行RTP 语音，但是没有407AUTH 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407auth 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

### 在cygwin下面编译需要winpcap(libpcap)的工程

先去下载winpcap

https://www.winpcap.org/devel.htm

WpdPack\_4\_1\_2

1. 将 WpdPack\Include里面的所有文件 拷贝到cygwin\usr\include
2. 将 WpdPack\Lib里面的 libpacket.a libwpcap.a拷贝到 cygwin\lib 同时 将libwpcap.a重命名为libpcap.a。

在centos下编译

## 三、UAC+UAS

用SIPp 做测试的时候需要准备以下几个文件：

UAC 端：uac.sh, uac.xml, uac.csv

uac.sh ：调用sipp 命令，并传入相应参数的脚本文件，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

### 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：



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:-inff1.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 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 200OK answer for INFO, UPDATE and NOTIFY messages.

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

-trace\_msg:

Displays sent and received SIP messages in <scenariofile 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，正试性能测试时，可取消，以免日志量太大影响本地性能；

-trace\_screen:

Dump statistic screens in the<scenario\_name>\_<pid>\_screens.logfile when quitting SIPp. Useful to get a final status report in background mode (-bgoption).

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 <scenariofilename>\_<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：

#!/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.sh100，表示最大同时呼叫100个用户；

uac.xml 文件内容如下：

附件:uac1.xml

///////////////////////////////////////////////////////

<?xmlversion="1.0"encoding="ISO-8859-1"?>

<!DOCTYPE scenario SYSTEM "sipp.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 2of the -->

<!--License, or (atyour 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., -->

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

<!----><!--Sipp 'uac' scenario with pcap (rtp)play -->

<!---->

<scenario name="UAC with media">

<label id="1"/>

<send retrans="500">

<![CDATA[

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

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

Max-Forwards: 70

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

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

From: "[field0]"<sip:[field0]@[remote\_ip]>;tag=d802bf55

Call-ID: OWI3YjdiYzljNmM2NDg1ZjZkMGNhZDVlZDUyZjE4Yzg.

CSeq: 1 INVITE

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

Content-Type: application/sdp

User-Agent: eyeBeam release 1011d stamp 40820

Content-Length: 434

v=0

o=- 9 2 IN IP4 [local\_ip]

s=CounterPath eyeBeam 1.5

c=IN IP4 [local\_ip]

t=0 0

m=audio 7080 RTP/AVP 0 8 18 101

a=alt:1 3 : xDdX1TNS kFByGAw6 192.168.49.1 7080

a=alt:2 2 : ahD+MZhk bQ6u4Ffv [local\_ip] 7080

a=alt:3 1 : wEpYSJOo MzKyFYxh 192.168.31.109 7080

a=fmtp:18 annexb=no

a=fmtp:101 0-15

a=rtpmap:18 G729/8000

a=rtpmap:101 telephone-event/8000

a=sendrecv

a=x-rtp-session-id:F105F9C332AF432E94476B1129A09CC4

]]>

</send>

<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:1ACK

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:2INVITE

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

Max-Forwards:70

Subject:Performance Test

[field2]

Content-Type:application/sdp

Content-Length:[len]

v=0

o=user1536557652353687637IN IP[local\_ip\_type][local\_ip]s=-

b=AS:84

c=INIP[local\_ip\_type][local\_ip]

t=00

a=X-nat:0

m=video 10510 RTP/AVP 34 98

a=sendrecv

a=rtpmap:98iLBC/8000

a=fmtp:98mod=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:2ACK

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

Max-Forwards:70

Subject:Performance Test

Content-Length:0

]]>

</send>

<pause milliseconds="5000"/>

<!-- <nop>

<action>

<execplay\_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:2BYE

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

Max-Forwards:70

Subject:Performance Test

Content-Length:0

]]>

</send>

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

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

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

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

<!--definition of the call length repartition table (unitis 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=1234]

153;13872;[authentication username=153 password=1234]

### 2、UAS

调试uas 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；



reg.xml

流程

uas.xml

流程：

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：

#!/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 $m-trace\_screen-trace\_err-p 12346-m $m-aa

使用方法：./reg.sh100，表示最大同时注册100个用户；

reg.xml 文件内容如下：

<?xmlversion="1.0"encoding="ISO-8859-1"?>

<!DOCTYPEscenario SYSTEM "sipp.dtd">

<scenarioname="Register">

<!--send register -->

<sendretrans="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:1REGISTER

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

Content-Length:0

Expires:3600

]]>

</send>

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

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

<labelid="1"/>

<sendretrans="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:2REGISTER

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

[field1]

Content-Length:0

Expires:3600

]]>

</send>

<recvresponse="200"></recv>

<pausemilliseconds="3000"/>

<!--definition of the response time repartition table (unitis ms) --><ResponseTimeRepartitionvalue="10,20, 30, 40, 50, 100, 150, 200"/><!--definition of the call length repartition table (unitis ms) --><CallLengthRepartitionvalue="10,50, 100, 500, 1000, 5000, 10000"/></scenario>

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：

#!/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.sh100，表示最大同时被叫100个用户；

uas\_noreg.xml文件内容如下：

<?xmlversion="1.0"encoding="ISO-8859-1"?>

<!DOCTYPEscenario SYSTEM "sipp.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 2of the -->

<!--License, or (atyour 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., -->

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

<!--<!--<!--

-->

Sipp default 'uas' scenario.

-->

-->

<scenarioname="BasicUAS responder">

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

<!--are saved and used for following messages sent. Useful to test <!--against stateful SIP proxies/B2BUAs.-->-->

-->

<recvrequest="INVITE"crlf="true"></recv>

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

<!--specified header if it was present in the last message received <!--(exceptif 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 (CRLFseperated) <!--to be used in place of the '[last\_\*]'keyword.

<send>

<![CDATA[

SIP/2.0100Trying [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/2.0180Ringing [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>

<sendretrans="500">

<![CDATA[

SIP/2.0200OK

[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=user1536557652353687637IN IP[local\_ip\_type][local\_ip]s=-

b=AS:84

c=INIP[local\_ip\_type][local\_ip]

t=00

a=X-nat:0

m=audio[auto\_media\_port]RTP/AVP98

c=INIP[local\_ip\_type][local\_ip]

b=TIAS:64000

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

a=sendrecv

a=rtpmap:98iLBC/8000

a=fmtp:98mod=30

]]>

</send>

<recvrequest="ACK"

rtd="true"

crlf="true">

</recv>

<pausemilliseconds="5000"/>

<nop>

<action>

<execplay\_pcap\_audio="pcap/cc.pcap"/>

</action>

</nop>

<pausemilliseconds="50000"/>

<recvrequest="BYE">

</recv>

<send>

<![CDATA[

SIP/2.0200OK

[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 200is lost to be

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

<pausemilliseconds="4000"/>-->-->

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

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

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

<CallLengthRepartitionvalue="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.csv 文件内容如下：

SEQUENTIAL

19002;[authentication username=19002 password=1234]

138721;[authentication username=138721 password=1234]

### 3、UAC+UAS

进行性能测试时，要使用UAC+UAS流程；建议使用两台sipp 客户端，一台担任UAC ，一台担任UAS 。

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

## 四、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>

<execplay\_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 日志中的NOTIFY

查看errors.log 日志，里面有很多针对NOTIFY 的错误，如下：

2013-11-2216:53:22:929138519802.929208:Discarding message which can't be mapped to a known SIPp call:

NOTIFY sip:19002@192.168.146.101:12346SIP/2.0

Via:SIP/2.0/UDP192.168.148.43;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:52219186NOTIFY

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 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 200OK 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

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:1REGISTER

CSeq:2REGISTER

CSeq:3INVITE

CSeq:4ACK

CSeq:5INVITE

CSeq:6ACK

CSeq:7BYE

会出现如下错误：

Reason:SIP;cause=408;text="ACKTimeout"

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:1REGISTER

CSeq:2REGISTER

CSeq:1INVITE

CSeq:1ACK

CSeq:2INVITE

CSeq:2ACK

CSeq:2BYE （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。

## 五、错误记录

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

问题描述：

#make pcapplay\_ossl

错误信息如下：

In file included from message.cpp:39:

sipp.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：

2013-11-2614:48:04:1001385448484.100398: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：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：

<sendretrans="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:2REGISTER

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1REGISTER 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98ilbc ，修改为98ilbc 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=INIP[local\_ip\_type][local\_ip]

t=00

a=X-nat:0

m=audio[auto\_media\_port]RTP/AVP98

c=INIP[local\_ip\_type][local\_ip]

b=TIAS:64000

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

a=sendrecv

a=rtpmap:98iLBC/8000

a=fmtp:98mod=30

## 六、SIP 协议应答码

SIP 应答码是包含了，并且扩展了HTTP/1.1应答码。并不是所有的HTTP/1.1应答码都适当应用，只有在折里指出的是适当的。其他HTTP/1.1应答码不应当使用。并且，SIP 也定义了新的应答码系列，6xx 。

1临时应答1xx

临时应答，也就是消息性质的应答，标志了对方服务器正在处理请求，并且还没有决定最后的应答。如果服务器处理请求需要花200ms 以上才能产生终结应答的时候，它应当发送一个1xx 应答。

注意1xx 应答并不是可靠传输的。他们不会导致客户端传送一个ACK 应答。临时性质的（1xx ）应答可以包含消息体，包含会话描述。

1.1100Trying

这个应答表示下一个节点的服务器已经接收到了这个请求并且还没有执行这个请求的特定动作（比如，正在打开数据库的时候）。这个应答，就像其他临时应答一样，种植了UAC 重新传送INVITE 请求。100(Trying)应答和其他临时应答不同的是，在这里，它永远不会被有状态proxy 转发到上行流中。

1.2180Ringing

UA 收到INVITE 请求并且试图提示给用户。这个应答应当出世化一个本地回铃。

1.3181Call is Being Forwarded(呼叫被转发)

服务器可以用这个应答代码来表示呼叫正在转发到另一个目的地集合。

1.4182Queued

当呼叫的对方暂时不能接收呼叫的时候，并且服务器决定将呼叫排队等候，而不是拒绝呼叫的时候，那么就应当发出这个应答。当被叫方一旦恢复接收呼叫，他会返回合适的终结应答。对于这个呼叫状态，可以有一个表示原因的短语，比如：”5calls queued;expected waiting time is 15minutes”。服务器可以给出好几个182（Queued ）应答告诉呼叫方排队的情况（比如排队靠前了等等）。

1.5183会话进度

183（Session Progress ）应答用于提示建立对话的进度信息。Reason-Phrase （表达原因的句子）、头域或者消息体可以用于提示呼叫进度的更消息的信息。

2成功信息2xx

这个应答表示请求是成功的。

2.1 200OK

请求已经处理成功。这个信息取决于不同方法的请求的应答。

3转发请求3XX

3xx 系列的应答是用于提示用户的新位置信息的，或者为了满足呼叫而转发的额外服务地点。

3.1 300Multiple Choices

请求的地址有多个选择，每个选择都有自己的地址，用户或者（UA ）可以选择合适的通讯终端，并且转发这个请求到这个地址。

应答可以包含一个具有每一个地点的在Accept 请求头域中允许的资源特性，这样用户或者UA 可以选择一个最合适的地址来转发请求。没有未这个应答的消息体定义MIME 类型。

这些地址选择也应当在Contact 头域中列出（20.10节）。不同于HTTP ，SIP 应答可以包含多个Contact 头域或者一个Contact 头域中具有一个地址列表。UA 可以使用Contact 头域来自动转发或者要求用户确认转发。不过，本规范没有定义自动转发的标准。

如果被叫方可以在多个地址被找到，并且服务器不能或者不愿意转发请求的时候，可以使用这个应答来给呼叫方。

3.2 301Moved Permently

当不能在Request-URI 指定的地址找到用户的时候，请求的客户端应当使用Contact 头域(20.10)所指出的新的地址重新尝试。请求者应当用这个新的值来更新本地的目录，地址本，和用户地址cache ，并且在后续请求中，发送到这个/这些列出的地址。

3.3 302Moved Temporarily

请求方应当把请求重新发到这个Contact 头域所指出的新地址(20.10)。新请求的

Request-URI 应当用这个应答的Contact 头域所指出的值。

在应答中的Expires(20.19节) 或者Contact 头域的expires 参数定义了这个Contact URI 的生存周期。UA 或者proxy 在这个生存周期内cache 这个URI 。如果没有严格的有效时见，那么这个地址仅仅本次有效，并且不能在以后的事务中保存。

如果cache 的Contact 头域的值失败了，那么被转发请求的Request-URI 应当再次尝试一次。临时URI 可以比超时时间更快的失效，并且可以有一个新的临时URI 。

3.4 305Use Proxy

请求的资源必须通过Contact 头域中指出的proxy 来访问。Contact 头域指定了一个proxy 的URI 。接收到这个应答的对象应当通过这个proxy 重新发送这个单个请求。305（UseProxy ）必须是UAS 产生的。

3.5380Alternative Service

呼叫不成工，但是可以尝试另外的服务。另外的服务在应答的消息体中定义。消息体的格式在这里没有定义，可能在以后的规范中定义。

4请求失败4xx

4xx 应答定义了特定服务器响应的请求失败的情况。客户端不应当在不更改请求的情况下重新尝试同一个请求。（例如，增加合适的认证信息）。不过，同一个请求交给不同服务器也许就会成功。

4.1 400Bad Request

请求中的语法错误。Reason-Phrase 应当标志这个详细的语法错误，比如”MissingCall-ID header field”。

4.2 401Unauthorized

请求需要用户认证。这个应答是由UAS 和注册服务器产生的，当407（Proxy

Authentication Required ）是proxy 服务器产生的。

4.3 402Payment Required

保留/以后使用

4.4 403Forbidden

服务端支持这个请求，但是拒绝执行请求。增加验证信息是没有必要的，并且请求应当不被重试。

4.5 404Not Found

服务器返回最终信息：用户在Request-URI 指定的域上不存在。当Request-URI 的domain 和接收这个请求的domain 不匹配的情况下，也会产生这个应答。

4.6 405Method Not Allowed

服务器支持Request-Line 中的方法，但是对于这个Request-URI 中的地址来说，是不允许应用这个方法的。

应答必须包括一个Allow 头域，这个头域包含了指定地址允许的方法列表。

4.7 Not Acceptable

请求中的资源只会导致产生一个在请求中的Accept 头域外的，内容无法接收的错误。

4.8 407Proxy Authentication Required

这个返回码和401（Unauthorized ）很类似，但是标志了客户端应当首先在proxy 上通过认证。

这个返回码用于应用程序访问通讯网关（比如，电话网关），而很少用于被叫方要求认证。

4.9 408Request Timeout

在一段时间内，服务器不能产生一个终结应答，例如，如果它无法及时决定用户的位置。客户端可以在稍后不更改请求的内容然后重新尝试请求。

4.10 410Gone

请求的资源在本服务器上已经不存在了，并且不知道应当把请求转发到哪里。这个问题将会使永久性的。如果服务器不知道，或者不容易检测，这个资源消失是临时性质的还是永久性质的，那么应当返回一个404（Not Found ）。

4.11 413请求实体过大。

服务器拒绝处理请求，因为这个请求的实体超过了服务器希望或者能够处理的大小。这个服务器应当关闭连接避免客户端重发这个请求。

如果这个情况是暂时的，那么服务端应当包含一个Retry-After 头域来表明这是一个暂时的故障，并且客户端可以过一段时间再次尝试。

4.12 414Request-URI Too Long

服务器拒绝这个请求，因为Request-URI 超过了服务器能够处理的长度。

4.13 415Unsupported Media Type

服务器由于请求的消息体的格式本服务器不支持，所以拒绝处理这个请求。这个服务器必须根据内容的故障类型，返回一个Accept ，Accpet-Encoding, 或者Accept-Language 头域列表。UAC 根据8.1.3.5节定义的方法处理这个应答。

4.14 416Unsupported URI Scheme

服务器由于不支持Request-URI 中的URI 方案而终止处理这个请求。客户端处理这个应答参照8.1.3.5。

4.15 Bad Extension

服务器不知道在请求中的Proxy-Require(20.29)或者Require(20.32)头域所指出的协议扩展。服务器必须在Unsupported 头域中列出不支持的扩展。UAC 处理这个应答请参见8.1.3.5

4.16 421Extension Required

UAS 需要特定的扩展来处理这个请求，但是这个扩展并没有在请求的Supported 头域中列出。具有这个应答码的应答必须包含一个Require 头域列出所需要的扩展。

UAS 不应当使用这个应答除非它真的不能给客户端提供有效的服务。相反，如果在Support 头域中没有列出需要的扩展，服务器应当根据基准的SIP 兼容的方法和客户端支持的扩展来进行处理。

4.17 423Interval Too Brief

服务器因为在请求中设置的资源刷新时间（或者有效时间）过短而拒绝请求。这个应答可以用于注册服务器来拒绝那些Contact 头域有效期过短的注册请求。这个应答的用法和相关的Min-Expires 头域在10.2.8,10.3,20.23节中介绍和说明。

4.18 480Temporarily Unavailable

请求成功到达被叫方的终端系统，但是被叫方当前不可用（例如，没有登陆，或者登陆了但是状态是不能通讯，或者有”请勿打扰”的标记）。应答应当在Retry-After 中标志一个合适的重发时间。这个用户也有可能在其他地方是有效的（在本服务器中不知道）。

Reason-Phrase(原因短句) 应当提示更详细的原因，为什么被叫方暂时不可用。这个值应当是可以被UA 设置的。状态码486（Busy Here ）可以用来更精确的表示本请求失败的特定原因。

这个状态码也可以是转发服务或者proxy 服务器返回的，因为他们发现Request-URI 指定的用户存在，但是没有一个给这个用户的合适的当前转发的地址。

4.19 481Call/TransactionDoes Not Exist

这个状态表示了UAS 接收到请求，但是没有和现存的对话或者事务匹配。

4.20 482Loop Detected

服务器检测到了一个循环(16.3/4)

4.21 483Too Many Hops

服务器接收到了一个请求包含的Max-Forwards(20.22)头域是0

4.22 484Address InComplete

服务器接收到了一个请求，它的Request-URI 是不完整的。在原因短语中应当有附加的信息说明。这个状态码可以和拨号交叠。在和拨号交叠中，客户端不知道拨号串的长度。它发送增加长度的字串，并且提示用户输入更多的字串，直到不在出现484（Address Incomplete ）应答为止。

4.23 485Ambiguous

Request-URI 是不明确的。应答可以在Contact 头域中包含一个可能的明确的地址列表。这个提示列表肯囊个在安全性和隐私性对用户或者组织造成破坏。必须能够由配置决定是否以404（NotFound ）代替这个应答，又或者禁止对不明确的地址使用可能的选择列表。

给带有Request-URI 的请求的一个应答例子：

sip:lee@example.com:

SIP/2.0485Ambiguous

Contact:Carol Lee <sip:carol.lee@example.com>

Contact:Ping Lee <sip:p.lee@example.com>

Contact:Lee M.Foote <sips:lee.foote@example.com>

部分email 和语音邮箱系统提供了这个功能。这个状态码和3xx 状态码不同：对于300来说，它是假定同一个人或者服务有不同的地址选择。所以对3xx 来说，自动选择系统或者连续查找就有效，但是对485（Ambiguous ）应答来说，一定要用户的干预。

4.24 486Busy Here

当成功联系到被叫方的终端系统，但是被叫方当前在这个终端系统上不能接听这个电话，那么应答应当回给呼叫方一个更合适的时间在Retry-After 头域重试。这个用户也许在其他地方有效，比如电话邮箱系统等等。如果我们知道没有其他终端系统能够接听这个呼叫，那么应当返回一个状态码600（Busy Everywhere ）。

4.25 487Request Terminated

请求被BYE 或者CANCEL 所终止。这个应答永远不会给CANCEL 请求本身回复。

4.26 488Not Acceptable Here

这个应答和606（Not Acceptable ）有相同的含义，但是只是应用于Request-URI 所指出的特定资源不能接受，在其他地方请求可能可以接受。

包含了媒体兼容性描述的消息体可以出现在应答中，并且根据INVITE 请求中的Accept 头域进行规格化（如果没有Accept 头域，那么就是application/sdp）。这个应答就像给OPTIONS 请求的200(OK)应答的消息体一样。

4.27 491Request Pending

在同一个对话中，UAS 接收到的请求有一个依赖的请求正在处理。14.2描述了这种情况应当怎样解决。

4.28 493Undecipherable

UAS 接收到了一个请求，包含了一个加密的MIME, 并且不知道或者没有提供合适的解密密钥。这个应答可以包含单个包体，这个包体包含了合适的公钥，这个公钥用于给这个UAS 通讯中加密包体使用的。细节描述在23.2节。

5Server Failure 5xx

5xx 应答是当服务器本身故障的时候给出的失败应答。

5.1 500Server Internal Error

服务器遇到了未知的情况，并且不能继续处理请求。客户端可以显示特定的错误情况，并且可以在几秒种以后重新尝试这个请求。

如果这个情况是临时的，服务器应当在Retry-After 头域标志客户端过多少秒钟之后重新尝试这个请求。

5.2 501Not Implemented

服务器没有实现相关的请求功能。当UAS 不认识请求的方法的时候，并且对每一个用户都无法支持这个方法的时候，应当返回这个应答。（proxy 不考虑请求的方法而转发请求）。

注意405（Method Not Allowed ）是因为服务器实现了这个请求方法，但是这个请求方法在特定请求中不被支持。

5.3 502Bad Gateway

如果服务器，作为gateway 或者proxy 存在，从下行服务器上接收到了一个非法的应答（这个应答对应的请求是本服务器为了完成请求而转发给下行服务器的）。

5.4 503Service Unavailable

由于临时的过载或者服务器管理导致的服务器暂时不可用。这个服务器可以在应答中增加一个Retry-After 来让客户端重试这个请求。如果没有Retry-After 指出，客户端必须就像收到了一个500（Server Internal Error ）应答一样处理。

客户端（proxy 或者UAC ）收到503（Service Unavailable ）应当尝试转发这个请求到另外一个服务器处理。并且在Retry-After 头域中指定的时间内，不应当转发其他请求到这个服务器。

作为503(ServiceUnavaliable) 的替代，服务器可以拒绝连接或者把请求扔掉。

5.5 504Server Time-out

服务器在一个外部服务器上没有收到一个及时的应答。这个外部服务器是本服务器用来访问处理这个请求所需要的。如果从上行服务器上收到的请求中的Expires 头域超时，那么应当返回一个408（Request TimeOut ）错误。

5.6 505 Version Not Supported

服务器不支持对应的SIP 版本。服务器是无法处理具有客户端提供的相同主版本号的请求，就会导致这样的错误信息。

5.7 Message To Large

服务器无法处理请求，因为消息长度超过了处理的长度。

6Global Failures 6xx

6xx 应答意味这服务器给特定用户有一个最终的信息，并不只是在Request-URI 的特定实例有最终信息。

6.1 600Busy Everywhere

成功联系到被叫方的终端系统，但是被叫方处于忙的状态，并不打算接听电话。这个应答可以通过增加一个Retry-After 头域更明确的告诉呼叫方多久以后可以继续呼叫。如果被叫方不希望提示拒绝的原因，被叫方应当使用603（Decline ）。只有当终端系统知道没有其他终端节点（比如语音邮箱系统）能够访问到这个用户的时候才能使用这个应答。否则应当返回一个486（Busy Here ）的应答。

6.2 603Decline

当成功访问到被叫方的设备，但是用户明确的不想应答。这个应答可以通过增加一个Retry-After 头域更明确的告诉呼叫方多久以后可以继续呼叫。只有当终端知道没有其他任何终端设备能够响应这个呼叫的势能才能给出这个应答。

6.3604Does Not Exists Anywhere

服务器验证了在请求中Request-URI 的用户信息，哪里都不存在

6.4 606Not Acceptable

当成功联系到一个UA, 但是会话描述的一些部分比如请求的媒体，带宽，或者地址类型不被接收。

606（NotAcceptable ）应答意味着用户希望通讯，但是不能充分支持会话描述。606（Not Acceptable ）应答可以在Warning 头域中包含一个原因列表，用于解释为何会话描述不能被支持。警告原因代码在20.43节中列出。

在应答中，可以出现一个包含媒体兼容性描述的消息体，这个消息体的格式根据INVITE 请求中的Accept 头域指出的格式进行规格化（如果没有Accept 头域，那么就是application/sdp），就像给OPTIONS 亲求的200(OK)应答中的消息一样。

我们希望这些媒体协商不要经常需要，并且当一个新用户被邀请加入已经存在的会话的时候，这个媒体协商可能不需要。这取决于邀请的初始化者是否需要对606（Not Acceptable ）进行处理。

这个应答只有当客户端知道没有其他终端能够处理这个请求的时候才能发出。

## 参考资料

* 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 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