

1ST COMPUTER TELEPHONY INNOVATORS

CommLink® Lite Server Side Setup Guide

commlink-pgsql.php – for use with Asterisk Realtime with PostgreSQL

©Copyright by First Computer Telephony Innovators LLC
January 15, 2008

| | |
|---|--------------------|
| INTRODUCTION..... | 3 |
| PREREQUISITES..... | 3 |
| DOWNLOAD COMMLINK®..... | 4 |
| 1. Check for prerequisites..... | 5 |
| 2. Configure Asterisk..... | 6 |
| 3. Verify Asterisk and Apache are run as the same user..... | 7 |
| 4. Install and verify Zend Optimizer..... | 9 |
| 5. Obtain and install license file..... | 10 |
| 6. Copy CommLink® Lite files..... | 10 |
| 7. Modify constants..... | 11 |
| 8. Test CommLink® Lite installation..... | 11 |
| 9. Adding Hands-Free/Intercom Configuration..... | 12 |
| a) Configuring in a conf file:..... | 12 |
| Aastra and Adtran Phones..... | 12 |
| Polycom Phones:..... | 12 |
| b) Configuring in a database | |
| | 13 |
| Aastra and Adtran Phones..... | 13 |
| Polycom Phones:..... | 13 |

INTRODUCTION

The CommLink® php files are designed to work with Asterisk configuration files. commlink-pgsql.php works in conjunction with CommLink® Lite for Outlook plugin and standalone version which allow users to manage Asterisk voicemails, voicemail properties, configuration and message synchronization. For detailed list of features please go to <http://www.1cti.com/products/commlinklite.html>.

PREREQUISITES

Server prerequisites for CommLink® Lite installation:

- Apache – Full-featured HTTP web server software. Used by CommLink® to communicate between CommLink's® Outlook plug in and the server as well as view Asterisk web based management tools - <http://www.apache.org>.
- Asterisk – Linux based PBX (private branch exchange) software that provides all the features of high-end telephone systems, many of which are not yet available in proprietary phone systems - <http://www.asterisk.org>.
- PHP - server-side scripting language used for creating dynamic web Pages; recommended to be run as an Apache module. Used by both Asterisk and CommLink® Lite to execute specific functions to control Asterisk - <http://www.php.net>
- SoX - Sound processing program. In CommLink® php file it is used to convert voicemail audio files between gsm (raw gsm), wav (MS wav format, 16 bit linear) and WAV (MS wav format, gsm encoded - wav49) - <http://sox.sourceforge.net>.
- Perl - Programming language. In CommLink® php files it is used to move and rename voicemail files; as voicemails are marked as viewed or new, their files need to be moved between different directories - <http://www.perl.org>.
- Sed - Streams editor used to filter text. In CommLink® php files it is used to modify specific lines in voicemail.conf and voicemail.txt files; in Asterisk's static configuration, voicemail information is stored in configuration files, which commlink-pgsql.php needs to modify as voicemail settings and options are changed. It is also used to update information in voicemail.txt files - <http://sed.sourceforge.net>
- PostgreSQL - multithreaded, multi-user SQL database management system - <http://www.pgsql.com>

DOWNLOAD COMMLINK®

Download CommLink® Lite package by logging onto Asterisk server via an SSH or Telnet program such as PuTTY.

```
login as: root
root@192.168.3.200's password: █
```

Once logged in, start off by creating a downloads directory:

```
[root@asterisk1 ~]# mkdir /downloads
[root@asterisk1 ~]# cd /downloads
[root@asterisk1 downloads]# █
```

Then, download the latest CommLink® Lite server files:

```
[root@asterisk1 downloads]# wget http://www.icti.com/support/cti/cloutlookLite/
commlink-pgsql-20070925.tgz
```

Next, extract the downloaded file:

```
[root@asterisk1 downloads]# tar -xzf commlink-pgsql-20070925.tgz
[root@asterisk1 downloads]# ls
commlink-pgsql commlink-pgsql-20070925.tgz
```

extracted package

downloaded package

Review the extracted files:

```
[root@asterisk1 commlink-pgsql]# ls
commlink-pgsql.inc  phpinfo.php
commlink-pgsql.ini  README.TXT
commlink-pgsql.php  ZendOptimizer-3.3.0a-linux-glibc21-i386.tar.gz
MD5SUM.TXT         ZendOptimizer-3.3.0a-linux-glibc23-x86_64.tar.gz
```

Lastly, verify that the files were downloaded and extracted correctly by using the md5sum checksum function; each checksum of the downloaded files should match values from MD5SUM.TXT file:

```
[root@asterisk1 commlink-pgsql]# md5sum *
78d09e3af354a67190dc8efb90322cce commlink-pgsql.inc
cc1470ccbeb8f0048cde13627ef7ac43 commlink-pgsql.ini
ddd7950e6f369c503971d2448ad73d1f commlink-pgsql.php
bbf562b742305deaed30185a2a6eaa47 MD5SUM.TXT
9dccf462d245f55ac3e0cdb0e5401f5b phpinfo.php
f1fc5b36f55b6a636476c5167768ee2f README.TXT
68a46ab6f8d1bf976388ad5a3618503e ZendOptimizer-3.3.0a-linux-glibc21-i386.tar.gz
ea54486b554305a92273e354b7cab15a ZendOptimizer-3.3.0a-linux-glibc23-x86_64.tar.gz
gz
[root@asterisk1 commlink-pgsql]# cat MD5SUM.TXT
78d09e3af354a67190dc8efb90322cce commlink-pgsql.inc
cc1470ccbeb8f0048cde13627ef7ac43 commlink-pgsql.ini
ddd7950e6f369c503971d2448ad73d1f commlink-pgsql.php
9dccf462d245f55ac3e0cdb0e5401f5b phpinfo.php
f1fc5b36f55b6a636476c5167768ee2f README.TXT
68a46ab6f8d1bf976388ad5a3618503e ZendOptimizer-3.3.0a-linux-glibc21-i386.tar.gz
ea54486b554305a92273e354b7cab15a ZendOptimizer-3.3.0a-linux-glibc23-x86_64.tar.gz
```

INSTALLATION INSTRUCTIONS

1. Check for prerequisites

To find the prerequisites try queering the rpm repository or run the “which” command. If any of below commands do not return results then proceed to install rpm packages for them.

- check httpd

```
[root@asterisk1 /]# rpm -qa | grep httpd
httpd-suexec-2.0.52-32.ent.centos4
httpd-2.0.52-32.ent.centos4
[root@asterisk1 /]# which httpd
/usr/sbin/httpd
```

- check php

```
[root@asterisk1 /]# rpm -qa | grep php
php-mbstring-4.3.11-2.8
php-odbc-4.3.11-2.8
php-4.3.11-2.8
php-pear-4.3.11-2.8
php-imap-4.3.11-2.8
php-snmp-4.3.11-2.8
php-pgsql-4.3.11-2.8
php-mysql-4.3.11-2.8
php-gd-4.3.11-2.8
tbm-phpsysinfo-2.5.2.24-1
[root@asterisk1 /]# which php
/usr/bin/php
```

- check sox

```
[root@asterisk1 /]# rpm -qa | grep sox
sox-12.17.5-3
[root@asterisk1 /]# which sox
/usr/bin/sox
```

- check perl

```
[root@asterisk1 /]# rpm -qa | grep perl
perl-DBI-1.40-8
perl-libwww-perl-5.79-5
perl-DBD-MySQL-2.9004-3.1
perl-Digest-SHA1-2.07-5
perl-Time-HiRes-1.55-3
perl-Term-ReadKey-2.30-2.e14.rf
...
[root@asterisk1 /]# which perl
/usr/bin/perl
```

- check sed

```
[root@asterisk1 /]# rpm -qa | grep sed
sed-4.1.2-6.e14
[root@asterisk1 /]# which sed
/bin/sed
```

- check pgsq1

```
[root@asterisk1 commlink-pgsq1]# rpm -qa | grep -i pgsq1  
php-pgsq1-4.3.11-2.8
```

- check postgres

```
[root@asterisk1 commlink-pgsq1]# rpm -qa | grep -i postgres  
postgresql-libs-7.4.13-2.RHEL4.1  
postgresql-7.4.13-2.RHEL4.1  
postgresql-docs-7.4.13-2.RHEL4.1  
postgresql-server-7.4.13-2.RHEL4.1  
postgresql-odbc-7.3-8.RHEL4.1
```

2. Configure Asterisk

Need to configure manager account(s) to enable screen pops and call control for CommLink® Lite for Outlook. Depending on the environment it may be desirable to have one manager account per CommLink® Lite user or have one shared for all users. Place each manager entry in a manager conf file with values for username, secret, deny, permit, read and write. Set the username and secret which required in the Network Configuration screen on CommLink® Lite for Outlook. Set deny for all IP ranges (if desired).

Add manager account to `/etc/asterisk/manager.conf` or if `manager.conf` is regenerated then `/etc/asterisk/manager_additional.conf`:

```
[root@asterisk1 /]# cd /etc/asterisk  
[root@asterisk1 asterisk]# ls mana*  
manager_additional.conf  manager.conf  manager.conf.bak  manager_custom.conf
```

Set permit for the IP range for the users that can have access to the manager. Set the read and write for "system,call,log,verbose,command,agent,user".

```
[root@asterisk1 asterisk]# vi manager_additional.conf  
[manager]  
secret = password  
deny=0.0.0.0/0.0.0.0  
permit=192.168.0.0/255.255.0.0  
read = system,call,log,verbose,command,agent,user  
write = system,call,log,verbose,command,agent,user
```

3. Verify Asterisk and Apache are run as the same user.

- check what user asterisk runs as:

```
[root@asterisk1 /]# ps -ef | grep asterisk
root      3063      1  0 Sep24 ?        00:00:00 /bin/sh /usr/sbin/safe_asterisk -U a
sterisk -G asterisk
asterisk  3064    3063  0 Sep24 ?        00:00:16 /usr/sbin/asterisk -f -U asterisk -G
asterisk  -v -g -p -U asterisk -G asterisk
postgres 3070    2423  0 Sep24 ?        00:00:00 postgres: asteriskuser asteriskrealt
ime [local] idle
asterisk  3123      1  0 Sep24 ?        00:00:00 -bash -c cd /var/www/html/panel && /
var/www/html/panel/safe_opserver &
asterisk  3124    3123  0 Sep24 ?        00:00:00 sh /var/www/html/panel/safe_opserver
asterisk  3126    3124  0 Sep24 ?        00:00:58 /usr/bin/perl -w /var/www/html/panel
/op_server.pl
asterisk  22701   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22702   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22703   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22704   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22705   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22706   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22707   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22708   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
root      23274   23084  0 21:48 pts/0    00:00:00 grep asterisk
```

- check what user httpd runs as:

```
[root@asterisk1 /]# ps -ef | grep httpd
root      22698      1  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22701   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22702   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22703   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22704   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22705   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22706   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22707   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22708   22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
root      23282   23084  0 21:52 pts/0    00:00:00 grep httpd
```

If users for asterisk and httpd services are the same, then proceed to installation step 4; otherwise, you may choose one of the two following choices:

1. Run script to change permissions after new voice message is left.
Files and directories in `/var/spool/asterisk/voicemail` need to be accessible to httpd service for read/write. A script can be created to change permissions whenever a voicemail is left and referenced in `voicemail.conf` as `"externnotify="`.

Script sample:

```
#!/usr/bin/perl
$content = $ARGV[0];
$exten = $ARGV[1];
system ("/bin/chmod -R 770 /var/spool/asterisk/voicemail/$content/$exten");
system ("/bin/chown -R apache:apache
/var/spool/asterisk/voicemail/$content/$exten");
```

For more information go to <http://www.voip-info.org/wiki/view/Asterisk+gui+vmail.cgi>

2. Change httpd and asterisk services so that both run as non-root user, such as asterisk. This is the recommended configuration and will not require modifications of voicemail or directories. For more detailed information please refer to <http://www.voip-info.org/wiki/index.php?page=Asterisk+non-root>.

To run httpd as asterisk user (assumed that asterisk user already exists) run the following commands:

- stop httpd service

```
[root@asterisk1 /]# service httpd stop
Stopping httpd: [ OK ]
```

- modify /etc/httpd/conf/httpd.conf and set:

User asterisk

Group asterisk

```
[root@asterisk1 /]# vi /etc/httpd/conf/httpd.conf
# User/Group: The name (or #number) of the user/group to run httpd as.
# . On SCO (ODT 3) use "User nouser" and "Group nogroup".
# . On HPUX you may not be able to use shared memory as nobody, and the
#   suggested workaround is to create a user www and use that user.
# NOTE that some kernels refuse to setgid(Group) or semctl(IPC_SET)
# when the value of (unsigned)Group is above 60000;
# don't use Group #-1 on these systems!
#
User asterisk
Group asterisk
```

- start httpd service:

```
[root@asterisk1 /]# service httpd start
Starting httpd: [ OK ]
```

- verify that httpd processes are ran by asterisk:

```
[root@asterisk1 /]# ps -ef | grep httpd
root      22698      1  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22701  22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22702  22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22703  22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22704  22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22705  22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22706  22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22707  22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
asterisk  22708  22698  0 16:03 ?        00:00:00 /usr/sbin/httpd
root      23282  23084  0 21:52 pts/0    00:00:00 grep httpd
```

- change owner and group of directory /var/www/html to asterisk

```
[root@asterisk1 /]# chown -R asterisk:asterisk /var/www/html
```


4. Install and verify Zend Optimizer

- Access directory where CommLink® package was downloaded and extract the Zend Optimizer package
 - o for 32 bit linux OS: ZendOptimizer-3.3.0a-linux-glibc21-i386.tar.gz
 - o for 64 bit linux OS: ZendOptimizer-3.3.0a-linux-glibc23-x86_64.tar.gz

```
[root@asterisk1 /]# cd /downloads/commmlink-pgsql
[root@asterisk1 commmlink-pgsql]# ls
commmlink-pgsql.inc  phpinfo.php
commmlink-pgsql.ini  README.TXT
commmlink-pgsql.php  ZendOptimizer-3.3.0a-linux-glibc21-i386.tar.gz
MD5SUM.TXT          ZendOptimizer-3.3.0a-linux-glibc23-x86_64.tar.gz
[root@asterisk1 commmlink-pgsql]# tar -xzf ZendOptimizer-3.3.0a-linux-glibc21-i386.tar.gz
[root@asterisk1 commmlink-pgsql]# ls
commmlink-pgsql.inc  README.TXT
commmlink-pgsql.ini  ZendOptimizer-3.3.0a-linux-glibc21-i386
commmlink-pgsql.php  ZendOptimizer-3.3.0a-linux-glibc21-i386.tar.gz
MD5SUM.TXT          ZendOptimizer-3.3.0a-linux-glibc23-x86_64.tar.gz
phpinfo.php
[root@asterisk1 commmlink-pgsql]# cd ZendOptimizer-3.3.0a-linux-glibc21-i386
```

- Execute Zends install script and follow the installation wizard prompts:

```
[root@asterisk1 ZendOptimizer-3.3.0a-linux-glibc21-i386]# ls
data          install      install-tty  LICENSE     README-ZendOptimizer
EULA-ZendOptimizer  install.sh  Inventory.xml  md5         zui_files
[root@asterisk1 ZendOptimizer-3.3.0a-linux-glibc21-i386]# ./install
```

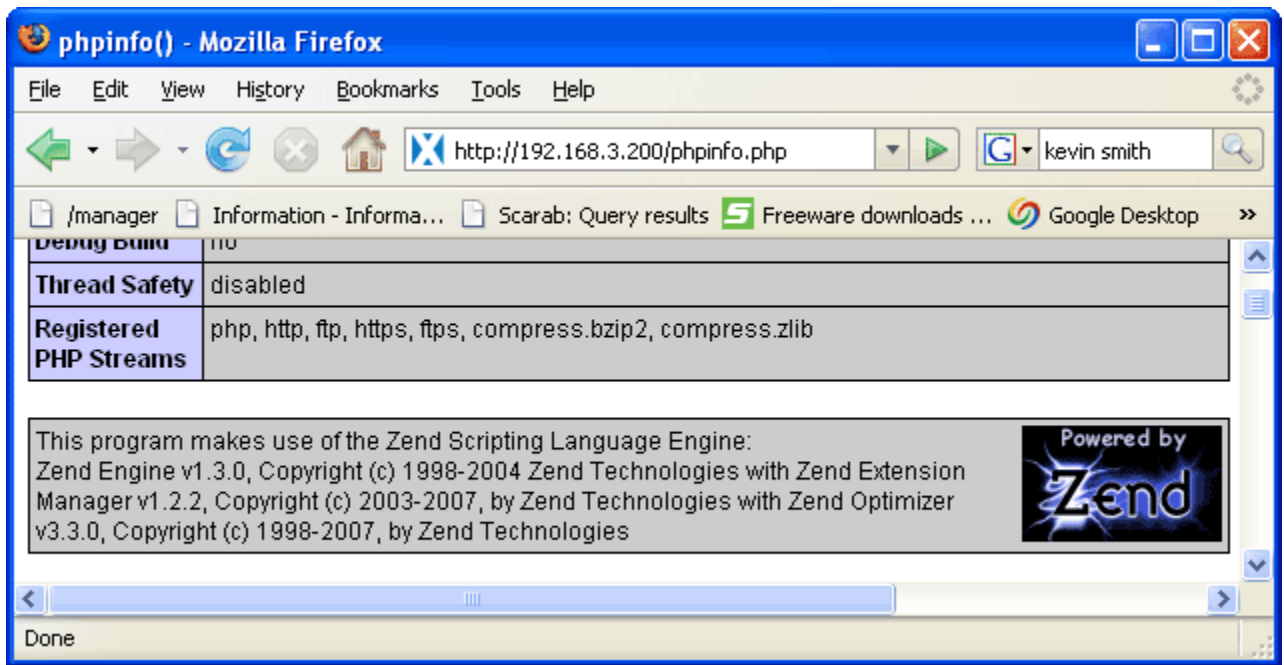
- Configure Zend Optimizer settings by editing the php.ini file in /usr/local/Zend/etc; make sure the following are set:

```
[root@asterisk1 ZendOptimizer-3.3.0a-linux-glibc21-i386]# vi /usr/local/Zend/etc/php.ini

zend_optimizer_license_path=/usr/local/Zend/etc/
display_errors = Off
log_errors = On
```

- Confirm Zend Optimizer has been installed correctly by viewing the php's setup information page. Copy phpinfo.php file from CommLink® Lite package to web server's directory, then open a web browser and review the phpinfo.php page.

```
[root@asterisk1 /]# cp /downloads/commmlink-pgsql/phpinfo.php /var/www/html/.
```



5. Obtain and install license file

- Get Zend Optimizer's host id

```
[root@asterisk1 ~]# /usr/local/Zend/bin/zendid  
M:SNCF3-9MSCE-Q3B2N-ZAXTT
```

- Compose an email to license@1cti.com, with Zend Optimizer's host ID. You will be emailed back with Zend's license file for your server.
- Check emails for message from license@1cti.com, which will include Zend's license file, and save the attached license.tgz file.
- Extract license.tgz and save cli.zl file to /usr/local/Zend/etc on the asterisk server.
- Restart Apache service

```
[root@asterisk1 ~]# service httpd restart  
Stopping httpd: [ OK ]  
Starting httpd: [ OK ]
```

6. Copy CommLink® Lite files

- There are three files that need to be copied from CommLink® Lite package:

```
[root@asterisk1 commlink-pgsql]# cp /downloads/commlink-pgsql/commlink-pgsql.php  
/var/www/html/.  
[root@asterisk1 commlink-pgsql]# cp /downloads/commlink-pgsql/commlink-pgsql.inc  
/var/www/html/.  
[root@asterisk1 commlink-pgsql]# cp /downloads/commlink-pgsql/commlink-pgsql.ini  
/etc/.
```

- Restart Apache service:

```
[root@asterisk1 ~]# service httpd restart
Stopping httpd: [ OK ]
Starting httpd: [ OK ]
```

7. Modify constants

- Verify values for constants in commlink.ini file located in /etc directory are correct:

```
[root@asterisk1 commlink-pgsql]# vi /etc/commlink-pgsql.ini
asterisk_conf_dir = /etc/asterisk/
asterisk_voicemail_context = default
asterisk_voicemail_general_inc = /etc/asterisk/vm_general.inc
AsteriskVM = /var/spool/asterisk/voicemail/
rtdb_hostname = localhost
rtdb_username = asteriskuser
rtdb_password = amp109
rtdb_databasename = asteriskrealtime
rtdb_voicemailusers_tablename = voicemail_users

#rtdb_voicemailconfig_tablename is only required if voicemail_conf_source is set
to "db"
#in which case the asterisk_voicemail_general_inc will not be used and all setti
ngs
#will be read from the database
rtdb_voicemailconfig_tablename = voicemail_config

#valid entries are "file" and "db"
voicemail_conf_source = file
```

Configuration file settings:

- asterisk_conf_dir* - location of Asterisk's conf files
- asterisk_voicemail_context* - name of context used for voicemail accounts
- asterisk_voicemail_conf* - name and location of Asterisk's voicemail.conf file
- asterisk_voicemail_general_inc* - name and location of Asterisk's vm_general.inc file
- AsteriskVM* - directory where Asterisk records voicemail
- voicemail_conf_source* - valid entries are "file" and "db"

Database settings:

- rtdb_hostname* - address of host on which Asterisk's realtime database is installed
- rtdb_databasename* - name of Asterisk's realtime database.
- rtdb_username* - username used to login to database
- rtdb_password* - database password.
- rtdb_voicemailtablename* - name of table used for voicemail user records
- rtdb_voicemailconfig_tablename* - *rtdb_voicemailconfig_tablename* is only required if *voicemail_conf_source* is set to "db"

8. Test CommLink® Lite installation.

- Access the test page by opening <http://<servername>/commlink-pgsql.php?Method=Display> and trying Get License. After clicking OK, a license number should be displayed; this indicates that CommLink® is installed and functioning correctly.

- Lastly, check web server's logs to verify that it is running correctly and no errors are reported.

```
[root@asterisk1 /]# cat /var/log/httpd/error_log
[Thu Sep 27 16:03:37 2007] [notice] suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
[Thu Sep 27 16:03:37 2007] [notice] Digest: generating secret for digest authentication ...
[Thu Sep 27 16:03:37 2007] [notice] Digest: done
[Thu Sep 27 16:03:38 2007] [notice] Apache/2.0.52 (CentOS) configured -- resuming normal operations
```

9. Adding Hands-Free/Intercom Configuration

To enable Accept button functionality available on incoming call pop-ups (Call Manager and Toast pop-up) you will need to include additional extension entries into the dial plan. I.E. for extension 113 add a hands-free extension **8113** (the new **8113** to be used in hands-free extension field in Network Configuration of CommLink® Lite Client). The hands-free extension will add a sip header to alert the phone to go to hands-free/intercom and dial the same channel as 113 extension.

Depending on system setup dial plan may be configured in a conf file or database.

a) Configuring in a conf file:

If dial plan is configured to be read from system conf files you will find an entry for extension 113 in either extensions.conf or extensions_additional.conf file, with the following text:

```
exten => 113,1,Dial,SIP/113
exten => 113,n,Hangup
```

Example of Asterisk extension conf file (usually extensions.conf or extensions_additional.conf), the dial plan configuration differs depending on phone used with Hands-Free/Intercom feature.

Aastra and Adtran Phones

Below are three lines that need to be included in the dial plan to add support for Hands-Free/Intercom feature:

```
exten => 8113,1,Set(__SIPADDHEADER=Alert-Info: \;info=alert-autoanswer)
exten => 8113,2,Dial,SIP/113
exten => 8113,n,Hangup
```

Polycom Phones:

To add Hands-Free/Intercom support for Polycom phone two things need to be changed:

- 1) Edit a line in Polycom's sip.cfg with value="**Ring Answer**" and class="**4**" for

```
<alertInfo voIpProt.SIP.alertInfo.1.value="Ring Answer"
voIpProt.SIP.alertInfo.1.class="4" />.
```

- 2) Add below three lines to the dial plan to add support for hands-free extension

```
exten => 8113,1,Set(__SIPADDHEADER=Alert-Info: Ring Answer)
exten => 8113,2,Dial,SIP/115
exten => 8113,n,Hangup
```

b) Configuring in a database

If dial plan is configured to be read from the database you will find the following entries in one of the tables:

| CONTEXT | EXTEN | PRIORITY | APP | APPDATA |
|---------|-------|----------|--------|------------|
| default | 113 | 1 | Dial | SIP/113 20 |
| default | 113 | 2 | Hangup | '' |

Examples of Asterisk extension entries, which will differ depending on phone used with Hands-Free/Intercom feature.

Aastra and Adtran Phones

Example of Asterisk extension entries; below are three lines that need to be included to add support for Hands-Free/Intercom feature:

| id | context | exten | priority | app | appdata |
|------|---------|-------|----------|--------------|-----------------------------------|
| 1006 | default | 8113 | 1 | SIPADDHEADER | Alert-Info: info=alert-autoanswer |
| 1007 | default | 8113 | 2 | Dial | SIP/113 20 |
| 1008 | default | 8113 | 3 | Hangup | '' |

Sample SQL statements for inserting extension 8113:

```
Insert into <table name> (context, exten, priority, app, appdata)
values ('default', '8113', '1', 'SIPADDHEADER', 'Alert-Info: info=alert-
autoanswer')
```

```
Insert into <table name> (context, exten, priority, app, appdata)
values ('default', '8113', '2', 'Dial', 'SIP/113|20')
```

```
Insert into <table name> (context, exten, priority, app, appdata)
values ('default', '8113', '3', 'Hangup', '')
```

Polycom Phones:

To add Hands-Free/Intercom support for Polycom phone two things need to be changed:

- 1) Edit a line in Polycom's sip.cfg with value="**Ring Answer**" and class="4" for

```
<alertInfo voIpProt.SIP.alertInfo.1.value="Ring Answer"
voIpProt.SIP.alertInfo.1.class="4"/>.
```

- 2) Example of Asterisk extension entries; below are three lines that need to be included to add support for Hands-Free/Intercom feature:

| id | context | exten | priority | app | appdata |
|------|---------|-------|----------|--------------|-------------------------|
| 1006 | default | 8113 | 1 | SIPADDHEADER | Alert-Info: Ring Answer |
| 1007 | default | 8113 | 2 | Dial | SIP/113 20 |
| 1008 | default | 8113 | 3 | Hangup | '' |

Sample SQL statements for inserting extension 8113:

```
Insert into <table name> (context, exten, priority, app, appdata)
values ('default', '8113', '1', 'SIPADDHEADER', 'Alert-Info: Ring Answer')
```

```
Insert into <table name> (context, exten, priority, app, appdata)
values ('default', '8113', '2', 'Dial', 'SIP/113|20')
```

```
Insert into <table name> (context, exten, priority, app, appdata)
values ('default', '8113', '3', 'Hangup', '')
```