



Open Asterisk and Trixbox Connector

Installation and Configuration

Table of contents

License.....	3
Legalese.....	3
Document History.....	3
Introduction.....	4
Installation.....	4
Components / Theory of operation.....	5
asteriskLogger.php.....	5
pre_install.php.....	5
AsteriskJS.php.....	5
dialin.js.....	6
dialout.js.....	6
callListener.php.....	6
callCreate.php.....	6
Configuration settings.....	7
System Settings.....	7
Per User Settings.....	8

License

This document © 2009 abcona active business solutions (www.abcona.de), and all yet unknown contributors.

This document is released to the public under the terms of the GPL, version 2. This means that you are free (free as in free speech) to distribute and modify this document, as long as you share your improvements with the community.

You should have received a copy of the GNU General Public License along with this document; if not, see <<http://www.gnu.org/licenses/>>

Legalese

- SugarCRM is a trademark of SugarCRM Inc. in the United States, the European Union and other countries.
- Asterisk is a registered trademark of Digium, Inc.

Document History

Date	Author	Change
25.06.09	Andreas Wollschlaeger aw@abcona.de	Initial version for release 1.1.0.0

Introduction

The YAAI module (which is supposed to stand for „Yet Another Asterisk Integration“) provides basic integration features between SugarCRM and Asterisk telephony software. Once installed, YAAI provides one-click dial from within SugarCRM to contacts, accounts etc., as well as sophisticated call monitoring / logging.

Please be aware that YAAI was developed with a „works for us“ attitude, so – depending on your local environment – the scripts and tools may need to be modified, adapted, even rewritten... „your mileage may vary“.

Installation

Installation is done straightforward using the „Module Loader“ from within SugarCRMs Admin section.

- Do NOT install YAAI when a previous version of the module is already installed. This will not upgrade the module, but clobber SugarCRM's module repository. Always deinstall the previous version before! Trust me, I have learned this the hard way :-)
- Before installation, please carefully read the Release Notes delivered with YAAI. Please note that YAAI does some very unclean modifications of SugarCRMs *Users* module, which may be highly version dependent and unfriendly to other extension modules.
- For installation to succeed, you need to have a working installation of Asterisk with properly configured Manager API. YAAI has been tested against Asterisk 1.4, other versions may or may not work.

See the section on configuration variables for required settings of YAAI.

Components / Theory of operation

In this section, we shall provide an overview on the different components of YAAI, along with a brief description of their interaction. This should give you an idea how YAAI works, but is surely no complete description of the system. However, all is written in PHP, so „use the source“....

asteriskLogger.php

This script provides the glue between Asterisk and SugarCRM. This script is not part of the SugarCRM GUI, it is supposed to run „standalone“.

- The most basic invocation is (from within an Linux shell)

```
php <path-to-SugarCRM>/Asterisk/asteriskLogger.php
```

This will run the script in *foreground mode*, echoing its operational messages to the console. However, for production use, this script would be run as a background service. Please check the YAAI distribution for contributed startup scripts.

asteriskLogger.php registers itself as a client to Asterisk's Manager API, so it receives notifications on Asterisk's activities. Most notably, the events we are interested in are „Dial“, „NewcallerID“, „Hangup“ and „Link“.

asteriskLogger.php uses the information received from these events for two purposes:

- Maintain the state of current calling activity in database table „asterisk_log“.
→ The information held in asterisk_log is used by SugarCRMs GUI to display call notifications in a popup.
- On receiving the „Hangup“ event, a record is created in SugarCRM's Calls module, providing information about the callers ID, date and time of the call, etc.

References:

For a description of Asterisk's Manager API, see:

<http://www.voip-info.org/wiki/view/Asterisk+manager+API>

<http://www.voip-info.org/wiki/view/asterisk+manager+events>

pre_install.php

This script is called during installation of the YAAI module. The tasks performed during installation are:

- creation of table asterisk_log in database
- installation of AsteriskJS.php as an „after_ui_frame“ hook in SugarCRM.
This causes AsteriskJS.php (and in turn, dialin.js resp dialout.js) to be included on all pages rendered by SugarCRM's GUI.

AsteriskJS.php

This script is installed as „logic hook“ and provides the magic® to enhance all pages in the GUI

with YAAI's functionality. Depending on users settings, it conditionally includes dialin.js and/or dialout.js.

dialin.js

This script is responsible for monitoring calls. When included, it uses AJAX calls (Whoa! Did I already mention we use AJAX?) to callListener.php (which in turn peeks into asterisk_log) to poll (and display) the current state of Asterisk's call activity. Note this script is responsible for both incoming and outgoing calls.

dialout.js

This script – when included – uses jQuery to dynamically modify the delivered HTML pages to include an dialout button for phone number fields. callCreate.php is invoked to eventually make a call.

callListener.php

This is the AJAX listener for dialin.js/dialout.js. It retrieves the current call state from YAAI's table asterisk_log and returns the result as JSON encoded data.

callCreate.php

This script is invoked to initiate an outgoing call. It simply opens a socket connection to Asterisk Manager and sends an „Originate“ action to place an outgoing call.

Configuration settings

This chapter provides information on required settings to get YAAI up and running.

System Settings

These settings affect the overall operations of YAAI (and asteriskLogger.php in particular).

To review these settings, navigate to **Admin → ASTERISK SugarCRM Connector::ASTERISK Configuration:**

ASTERISK Configuration: [Print](#) [Help](#)

Save Restore Cancel

ASTERISK Configuration

Asterisk Manager Host: Asterisk Manager Port:

Asterisk Manager Login: Asterisk Manager Password:

Dial Context: Dialout Prefix:

Dialpattern for inbound/outbound matching: Asterisk Soap-User:

Save Restore Cancel

Parameter Name	Explanation
Asterisk Manager Host	Obviously, the node where Asterisk is running
Asterisk Manager Port	The Port where the Manager API is running – Default is 5038
Asterisk Manager Login	Username configured in Asterisks manager.conf
Asterisk Manager Password	Login credentials (manager.conf)
Dial Context	Asterisk context where outgoing calls are placed. Check extensions.conf for proper value.
Dialout Prefix	Optional number prefixed to all outgoing calls (e.g. „0“ to get an external connection)
Dialpattern for inbound/outbound matching	<p>To provide proper call logging, asteriskLogger.php needs to distinguish incoming from outgoing calls. Here you may provide a regular expression to detect an internal trunk.</p> <p>This pattern is matched against the caller id sent by Asterisk, which is usually something like „SIP/xxx-yyyyy“, where xxx identifies your internal extension.</p> <p>Thus, for the popular case of internal phones to have 2 or 3 digit extensions, the pattern would be</p> <pre>^SIP/[1-9][0-9][0-9]?</pre> <p>Hint: this approach should fit most needs, for more complex scenarios, you might want to tweak asteriskLogger.php</p> <p>Hint: The pattern matching is done case-insensitive</p>
Asterisk SOAP User	AsteriskLogger uses this user to make SOAP calls to SugarCRM; this user <i>owns</i> all unassignable call records. You should provide some unprivileged dummy user for that purpose.

Per User Settings

These settings may be changed by navigating to **Admin** → **User Management** as Sugar admin user.

Asterisk options		
Asterisk Extension	Linus	Asterisk extension assigned to this user (usually a two or three digit number)
Call notification	<input type="checkbox"/>	If selected the users extension will be added to the listener and the user will be notified if a new call is coming in and also see calls going out
Magic dial buttons	<input checked="" type="checkbox"/>	If selected the user will be enabled to dial out clicking on the phone number of a contact

Parameter Name	Explanation
Asterisk Extension	The Asterisk extension assigned to that particular user. The extension is used to match inbound calls with some SugarCRM user. This entry should match an entry in Asterisk's sip.conf, without a leading "SIP".
Call notification	Activate visual call notification. You usually want this ;-)
Asterisk Manager Login	Activate magic dial buttons in SugarCRM's GUI. You usually want this, too ;-)