2012 sipX-CoLab Hackfest

Meet with the SIPfoundry Engineering Team to share your ideas for our sipXecs development efforts –
Join the sipXecs Hackfest during the second day of our sipX-CoLab conference!
SIPfoundry
sipX-CoLab
Collaboration. Communication. Education. Fort Collins, Colorado.
March 5 – 6, 2012
Hackfest
March 6, 2012

Teams

Da' Gangstas! - Joegen, Douglas

Nightlosers - George, Mircea

TBD - Laurentiu, Jan Vincent Liwanag

Ideas on Hackfest

Integrate homer, a sip stack trace utility into sipXecs

Coordinator: Douglas from **Da' Gangstas!**
Summary: Embed Homer in sipXecs to capture message call flow

Description:
Whether you're a newbie or veteran SIP engineer, when there's a problem getting a call through the system, you need to see the call flow to all components. sipXsipViewer is a nice utility to see the call flow, but it can be cumbersome to collect the logs. Homer offers web based sip call flow complete with db query interface.

*Resources

- http://www.sipcapture.org/

Goals:

- Find active or past calls and see visual of call flow.
- Interface should be accessible from sipXconfig admin UI
- Optional install, e.g. not a required install for systems that do not want the overhead.

Developer Setup in less than 10 minutes

Coordinator: Douglas
Summary: Get development environment setup quickly and reduce setup mistakes.

Description:
This idea was sparked because Douglas and George will be coming to hackfest w/o their workstations but instead laptops. Instead of not participating in hackfest, the day can be spent optimizing the steps to building a workstation. Once the goal is met, they can participate in other projects.

This project is particularly important going forward because 4.6 exposes more plugin APIs then ever before. Lowering the barrier to assembling a build system is crucial to extending sipXecs and therefore adoption of sipXecs..

Goals:

- System dependencies installed
- Full source tree compiled
- RPMs built for all distributions
- Eclipse installed with setup instructions (Eclipse is preferred IDE for Java apps)
- Easy to follow instructions.

Non-Goals:

- Not creating a "wrapper script" for system setup, but leverage standard tools and checks
- Not creating a virtual machine for setup, virtual machines get out of date fast and pain to maintain.

Develop plugin to store voicemails on other type of storage but local filesystem

Coordinator: George
Summary: Develop sipXivr plugin to store and retrieve voicemails from a ftp server

Description:
Porting sipXivr to Spring in 4.6 opens the door for various plugins like multiple voicemail storage support. More, these plugins could be developed in dynamic languages as JRuby or Groovy. Easy as implementing the right API, rewriting mailbox manager definition, packing it as a plugin (jar file) and deploying it in sipXecs.

Goals:

- Voicemails stored on chosen storage (could be ftp server or db)
- Voicemails retrieved from chosen storage

sipXecs administration via XMPP

Coordinator: ? *(George if time permits, idea originally suggested by Joegen some time ago)*
Summary: Develop XMPP bot to perform admin functionality (similar with sipXconfig admin functionality)

Description:
Thanks to sipXecs 4.6 architecture and extensive usage of Spring (in projects like sipXconfig), any java based component could register as an XMPP user and interpret messages as commands. We'll walk together through all the steps required for learning sipXconfig to "talk" XMPP and perform common administration commands (like the one performed from CLI or sipXconfig UI).

*Resources

- http://static.springsource.org/spring-integration/docs/2.0.x/reference/html/xmpp.html

Goals:

- Query services status via XMPP command
- Start / stop services via XMPP command

Support phone configuration reload without reboot

Coordinator: ? *(George added idea but may not have time to work in it)*
Summary: Change sipXconfig mechanism to allow configuration reload without reboot for phones that support this option

Description:
After altering phone config admin have to schedule a phone reboot in order to pick up changes. This is done by sending a SIP message containing check-sync;reboot=true sequence. However, some phones does not require a reboot (check-sync;reboot=false) for reloading changes but currently this is not supported by sipXconfig. The goal of this session is to make sipXconfig core to recognize such phones.

*Resources

- http://track.sipfoundry.org/browse/XX-9292

Goals:

- Add new abstract phone in sipXconfig
- Change sipXconfig UI for phones that doesn't need reboot
- Deploy and test with such phone (have to get one of these)

Conference management via XMPP

Coordinator: Mircea
Summary: Extend sipXimbot to include conference management commands

Description:
Given new REST api, recently added in 4.4 and 4.6 that performs conference management commands like: record, get record duration, get if record is in progress we can extend the current set of sipXimbot commands (@call, @xfer, @conf)

Goals:

- add @record that automatically starts recording when the group chat is an audio conference ongoing
- add @record stop that stops the current recording (if any)

Scriptable Authenticator and Redirector Plugin using Google V8

Coordinator: Joegen
Summary: Create a plugin that would allow administrators to come up with custom authenticators and redirectors using a JavaScript API

Description:
Currently, one may customize sipXecs transaction processing and routing behavior by implementing custom C++ plugin. The goal of this project is to extend the plugin architecture to support extensions written in JavaScript.

Goals:

- Expose an API that would allow manipulation of SIP message from within JavaScript
- Expose an API that would expose application layer configuration such as fastdb/mongodb/sipxconfig hooks.

Add jingle protocol (google talk) support

Coordinator: ? *(proposed by Mike)*
Summary: Integrate and configure FreeSWITCH mod_dingaling in sipXecs

Description:
Integrate FreeSWITCH mod_dingaling and allow such extensions to be configurable from sipXecs admin UI

*Resources

- http://wiki.freeswitch.org/wiki/Dingaling
- http://qxip.net/mediawiki/index.php/SIPX_JINGLE_HACK

Goals:

- Include mod_dingaling
- Configure FS dial plan and sipXecs extensions via sipXconfig

OpenACD v2 Integration

Coordinator: Laurentiu, JV
Summary: Upgrade sipXecs to use OpenACD v2 and enable new features in v2.

Description:
sipXecs currently comes with OpenACD v1. Effort has been put to integrate OpenACD v2 with the current sipXecs. However, there are no sipXconfig settings to enable the new features of v2. It is the goal of this hackfest to enable such new features found in v2 i.e. multi channel support per agent.

Goals

- Produce a build of sipX with OpenACD v2
- Integrate new configuration of OpenACD v2 into sipXconfig