

FreeSWITCH

mid-registering

for Fun, Profit, Scale and Push

Giovanni Maruzzelli
gmaruzz@OpenTelecom.IT

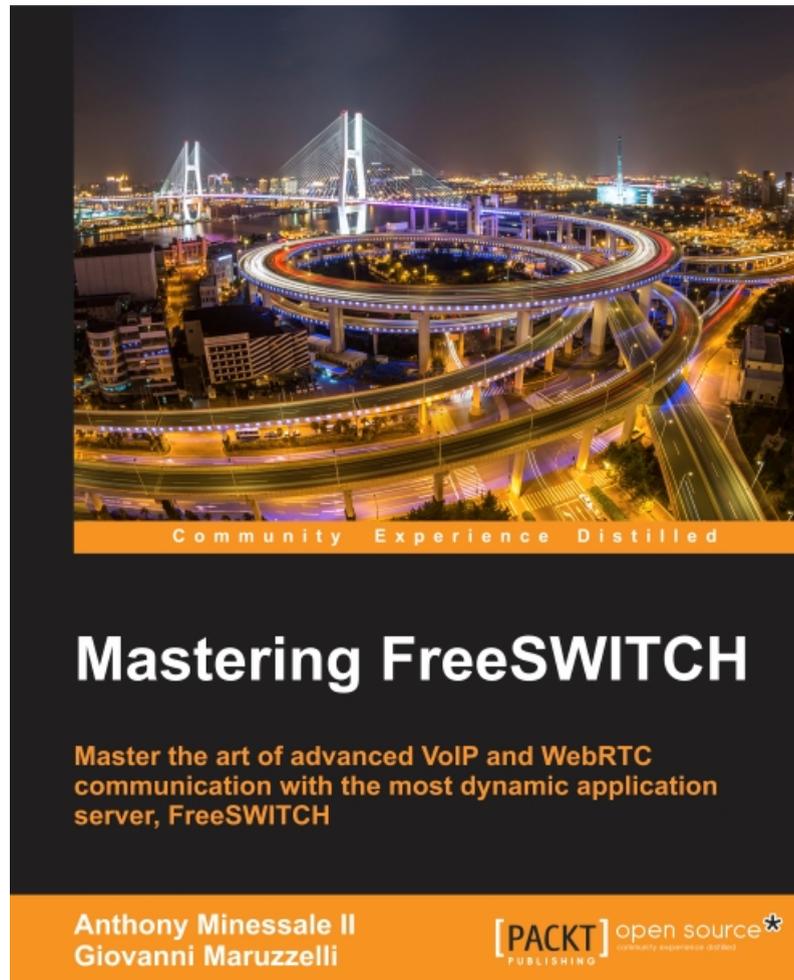
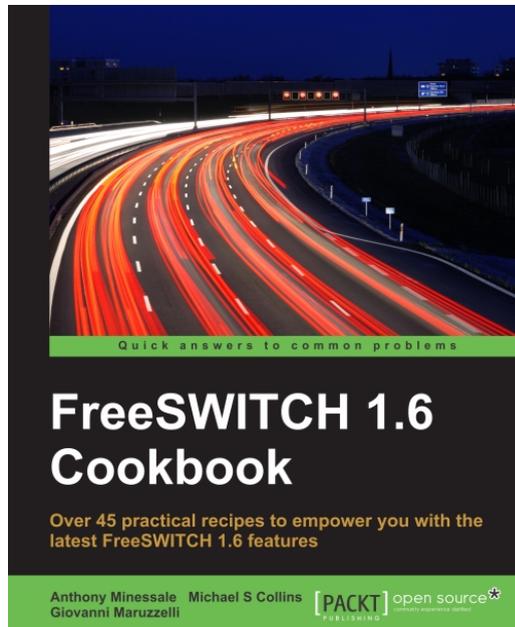
Giovanni Maruzzelli

- Carrier / Enterprise Grade Projects
- SIP / WebRTC / High Availability / Scaling
- Principal in OpenTelecom.IT
- Building the Internet since 1994
- With OpenSource since 1996
- Dealt with NCSA for Mosaic rights, before Netscape era
- Bought MySQL License N. 1 (Monty and Axel sent me a parchment, before selling to Oracle)

Books and Speeches

Conferences:

- AstriCon (USA and SPAIN)
- ClueCon (USA)
- FOSDEM (Belgium)
- FreeSWITCH_CN (China)
- IIT (India)
- Kamailio World (Germany)
- OpenSIPS Summit (Nederland)



Terms

- **REGISTER**, the SIP request (method) sent by the phone to the server, announcing phone's SIP identity and IP address
- **REGISTRAR**, inside the server, the logical function that deals with REGISTER requests, filling and querying a DB containing identities and addresses
- **USER LOCATION**, inside the server, the logical function that deals with knowing which address a user can be reached at (where to send a call to him), querying that same DB containing identities and addresses

Requirements

- PBX needs to know the IP address(es) of each one of its users, so it can send them calls
- EG:
 - a call arrives to PBX
 - PBX verifies if callee is one of its own
 - then checks if it's "registered" or not
 - if callee is registered, PBX sends the call to the callee's phone(s) address(es)
 - if callee is not registered, PBX answers the call as voice mail

REGISTER and EXPIRES

- phone sends a REGISTER to the server, telling its own identity and address, and proposing a tentative “expiration” time
- server store in the user location DB the identity/address info and then answers to the phone telling an authoritative expiration time
- phone MUST re-REGISTER again to the server before the expiration time chosen by the server has passed, failing which at expiration time the server will de-register (delete) the phone from the user location DB

Rationale

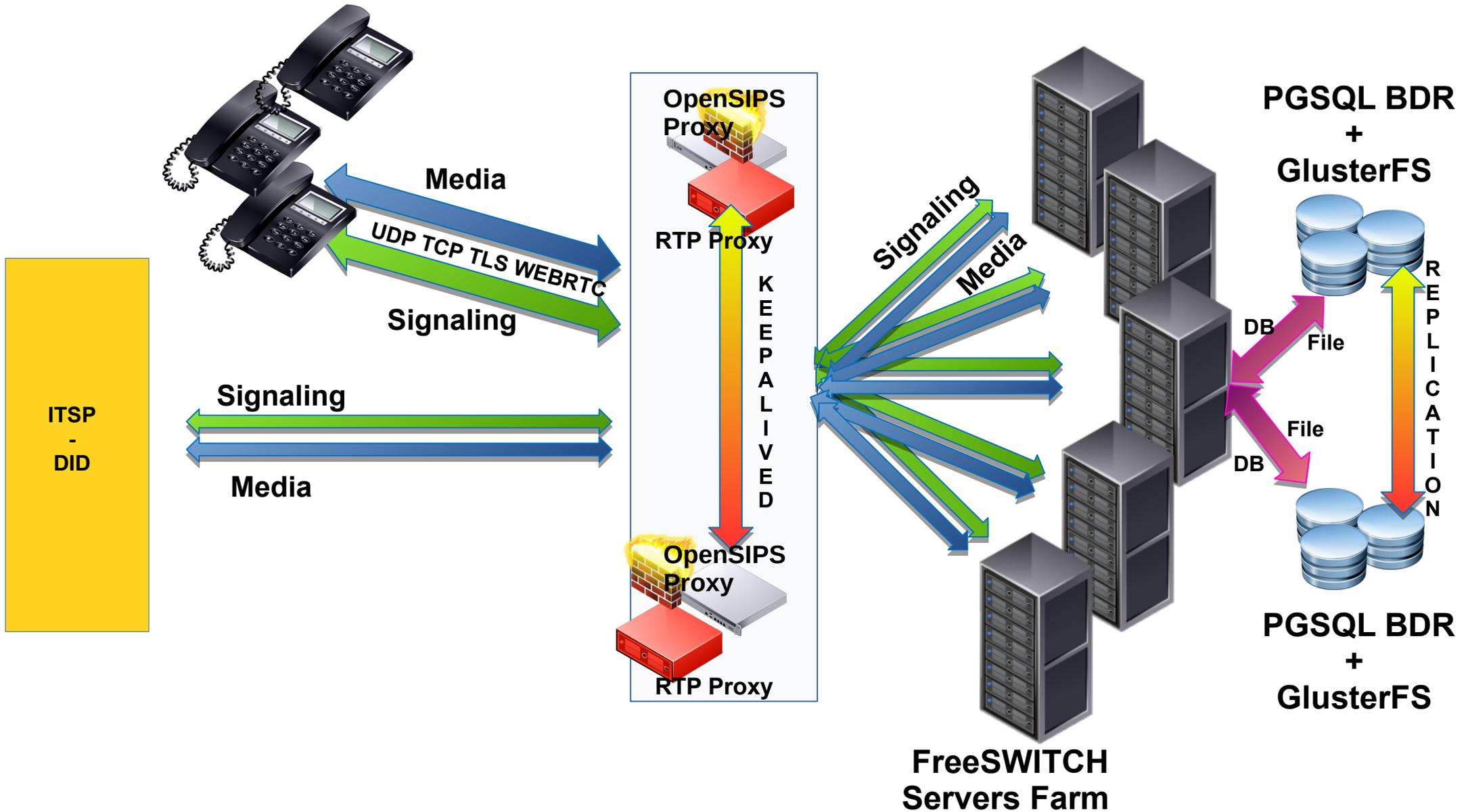
- REGISTER transaction management is a relatively heavy task, particularly when a robust number of phones are registering very often
- a **B2BUA**, like FreeSWITCH or Asterisk, is optimized for advanced media services and application scripting
- a **PROXY**, like OpenSIPS or Kamailio, is optimized for big numbers of signaling operations

From Each According to His Ability

- FS and AST are wonderful B2BUAs
 - They join calls together, and take care of content
 - MEDIA SERVER
 - APPLICATION SERVER
 - BUSINESS LOGIC SERVER
 - AUDIO/VIDEO MANIPULATION/INTERACTION/INTERPRETATION
- OSIPS and KAM are wonderful PROXIES
 - They direct the traffic, keeping track of connections
 - SECURITY
 - LOAD BALANCING
 - USER LOCATION
 - PRESENCE

Mid-Registrar

- Inside the server, the logical function that acts as a REGISTRAR toward the phones, but gets and sends info from/to (another, main) REGISTRAR
 - phone REGISTERS to the mid-registrar
 - mid-registrar sends to the (main) registrar the info he got from phone
 - if (main) registrar says to mid-registrar that the info from phone are ok, mid-registrar says ok to the phone
 - phone only deals with mid-registrar, never with (main) registrar



OpenSIPS Mid-Registrar

→ ↻ 🔒 opensips.org/docs/modules/3.1.x/mid_registrar.html

MID_REGISTRAR MODULE

Table of Contents

1. Admin Guide

1.1. Overview

- 1.1.1. Path Support (RFC 3327)
- 1.1.2. GRUU Support (RFC 5627)
- 1.1.3. SIP Push Notification Support (RFC 8599)

1.2. Working modes

- 1.2.1. Contact mirroring (default)
- 1.2.2. Contact throttling
- 1.2.3. AOR throttling

1.3. Auto-Insertion Into Future SIP Flows

1.4. Dependencies

- 1.4.1. OpenSIPS Modules
- 1.4.2. External Libraries or Applications

1.5. Exported Parameters

- 1.5.1. mode (integer)
- 1.5.2. contact_id_insertion (integer)
- 1.5.3. contact_id_param (string)
- 1.5.4. outgoing_expires (integer)

the Genial Thing of Mid-Registrar

**who said you must always tell the truth, all
the truth, only the truth?**

- mid-registrar lies
- then, if needed
- it lies again
- and then...
- it lies

the Genial Thing of Mid-Registrar

It's all about

expirations

and

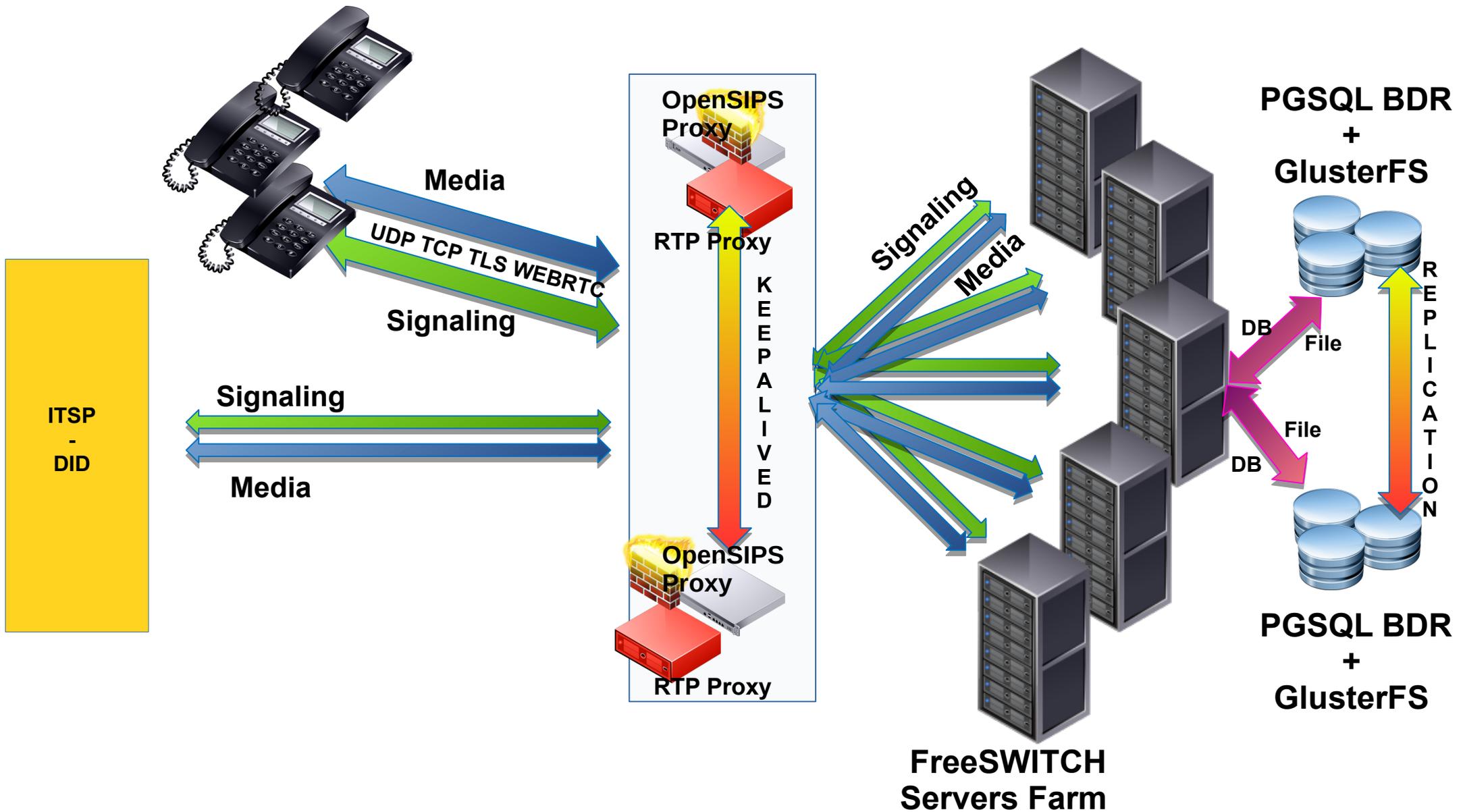
contacts (eg: addresses)

Registrations with Mid-Registrar

- phone REGISTERs its contact (identity@itsaddress) with OpenSIPS mid-registrar
- phone gets a short expiration time (120 sec) from OpenSIPS mid-registrar
- OpenSIPS mid-registrar sends to the FreeSWITCH (main) registrar the phone identity (AOR), with a long expiration time (3600 sec), and with its own (OpenSIPS) address

Registrations with Mid-Registrar

- To FreeSWITCH, all identities (AORs) have only one contact each, and all at OpenSIPS address
- When FreeSWITCH needs to send a call to one identity (AOR), FreeSWITCH sends the call to OpenSIPS
- OpenSIPS checks into its user location DB, and then sends the call(s) to the actual contact(s), eg address(es), of the identity (AOR)



the Genial Thing of Mid-Registrar

- **each time phone re-REGISTERS with OpenSIPS (in less than 120 secs)**
 - mid-registrar tells nothing to FreeSWITCH
 - up to the 3600 sec expiration time
 - then, it resends to FreeSWITCH the identity registration valid for another 3600 sec
- **if phone fails to re-REGISTER with OpenSIPS in 120 sec**
 - mid-registrar sends a de-registration (deletion) to FreeSWITCH

the Genial Thing of Mid-Registrar

- **each time another, different phone REGISTERS that same identity with OpenSIPS (eg, deskphone and softphone)**
 - mid-registrar tells nothing to FreeSWITCH
 - mid-registrar stores (add) another address (contact) for that identity (AOR) in the user location DB
 - when OpenSIPS sends a call to that AOR, if there are multiple addresses (contacts) stored for that AOR, OpenSIPS sends multiple parallel (forked) calls to all of that AOR's contacts (addresses). The first call to be answered will win and go ahead (the other, not yet answered, calls will be CANCELED)
- **if that other phone fails to re-REGISTER with OpenSIPS in 120 sec**
 - mid-registrar tells nothing to FreeSWITCH
 - delete that other phone address (contact) from its identity (AOR) in user location DB
- **only if/when that AOR (identity) has no more contacts (all phones failed to re-REGISTER)**
 - mid-registrar sends a de-registration (deletion) to FreeSWITCH

PROBLEM: apps want to sleep

- mid-registering is all fine and dandy for your dad's office phone, but...
 - what smartphone owner likes his SIP app to be always on and waste the battery by sending a re-REGISTER each 120 sec?
 - how can the smartphone app gets incoming calls if it doesn't re-REGISTER?
 - BTW, the smartphone can also change network, going from WiFi to LTE, etc...

SOLUTION: Push Notification

- smartphone app sleeps
- when server wants to send a call to the smartphone app, it first sends a Push Notification to the smartphone, via Google or Apple notification services
- smartphone gets the notification and awakes the app
- app immediately re-REGISTERS to server
- server gets the registration, and immediately sends the call to the smartphone app

the Genial Thing of Mid-Registrar

- **OpenSIPS mid-registrar implements RFC 8599**
 - Push Notifications for SIP, standard compliant
- **App sends its own PN service ID as part of contact**
 - Mid-registrar store the PNID together with that contact
- **Before contact's expiration time**
 - OpenSIPS mid-registrar sends a PN to the smartphone via PN service, smartphone wakes the app, app re-REGISTER
- **When OpenSIPS needs to send a call to the app**
 - OpenSIPS mid-registrar sends a PN to the smartphone via PN service, smartphone wakes the app, app re-REGISTER, OpenSIPS sends the call

the Genial Thing of Mid-Registrar

- **actually, in this app/PN case, we want much longer expiration times**
 - we don't need to wake the app often, because we have the PNID to send PNs to
 - if that PNID changes (because so wants the PN service at Google or Apple)
 - the app will send us its new PNID (via a re-REGISTER)
- **so, we go for one week minimum expiration time toward app side**
 - and some more toward FreeSWITCH side
- **to FreeSWITCH, it's all transparent, it does not know anything**
 - to FreeSWITCH, it is an identity (AOR) like the others, and like the others it has just one contact (address), at OpenSIPS address

TIP

- Have a Script to Reload Registrations from OpenSIPS to FreeSWITCH
- (if you purge registrations from FreeSWITCH, you want to reload them from OpenSIPS immediately, as opposed to waiting for OpenSIPS to automatically resend them after the expiration time toward FreeSWITCH has timeout: it can be hours, or days)

Thank You

QUESTIONS ?

Giovanni Maruzzelli
gmaruzz@OpenTelecom.IT