PRÁCTICA 2

Configuración de sistemas de telefonía IP

Diego Sanz Martín y María Domínguez Valdivieso

Instalación de Asterisk

La instalación de Asterisk en Debian server 10 consistirá en:

- Instalación de librerías necesarias.
- Obtener el programa comprimido de Asterisk de la web oficial.
- Descomprimir, configurar e instalar el paquete.

```
$ sudo apt -y install git curl wget libnewt-dev libssl-dev libncurses5-dev
subversion  libsqlite3-dev build-essential libjansson-dev libxml2-dev  uuid-dev
```

```
Configurando libsqlite3-dev:amd64 (3.27.2-3+deb10u1) ...
Configurando uuid-dev:amd64 (2.33.1-0.1) ...
Configurando zlib1g-dev:amd64 (1:1.2.11.dfsg-1) ...
Configurando g++-8 (8.3.0-6) ...
Configurando libncurses5-dev:amd64 (6.1+20181013-2+deb10u2) ...
Configurando gnupg (2.2.12-1+deb10u1) ...
Configurando libpng-dev:amd64 (1.6.36-6) ...
Configurando g++ (4:8.3.0-1) ...
update-alternatives: utilizando /usr/bin/g++ para proveer /usr/bin/c++ (c++) en modo automático
Configurando build-essential (12.6) ...
Configurando libslang2-dev:amd64 (2.3.2-2) ...
Configurando libnewt-dev:amd64 (0.52.20-8) ...
Procesando disparadores para man-db (2.8.5-2) ...
Procesando disparadores para libc-bin (2.28-10) ...
usu@debian:~$
```

```
$ sudo apt policy asterisk
```

```
usu@debian:~$ sudo apt policy asterisk
asterisk:
  Instalados: (ninguno)
  Candidato:  1:16.2.1~dfsg-1+deb10u2
  Tabla de versión:
     1:16.2.1~dfsg-1+deb10u2 500
        500 http://deb.debian.org/debian buster/main amd64 Packages
```

```
$ cd /usr/src/

$ sudo curl -O http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-18-current.tar.gz
```

```
usu@debian:/usr/src$ sudo curl -O http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-18-cur
rent.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 26.6M  100 26.6M    0     0  1907k      0  0:00:14  0:00:14 --:--:-- 4194k
usu@debian:/usr/src$
```

```
$ sudo tar xvf asterisk-18-current.tar.gz
```

```
asterisk-18.3.0/utils/db1-ast/recno/rec_search.c
asterisk-18.3.0/utils/db1-ast/recno/rec_seq.c
asterisk-18.3.0/utils/db1-ast/recno/rec_utils.c
asterisk-18.3.0/utils/db1-ast/recno/recno.h
asterisk-18.3.0/utils/expr2.testinput
asterisk-18.3.0/utils/extconf.c
asterisk-18.3.0/utils/frame.c
asterisk-18.3.0/utils/frame.h
asterisk-18.3.0/utils/muted.c
asterisk-18.3.0/utils/smsq.c
asterisk-18.3.0/utils/stereorize.c
asterisk-18.3.0/utils/streamplayer.c
asterisk-18.3.0/utils/utils.xml
usu@debian:/usr/src$
```

```
$ cd asterisk-18*/

$ sudo contrib/scripts/get_mp3_source.sh
```

```
usu@debian:/usr/src/asterisk-18.3.0$ sudo contrib/scripts/get_mp3_source.sh
Redirecting to URL 'https://svn.digium.com/svn/thirdparty/mp3/trunk':
A    addons/mp3
A    addons/mp3/MPGLIB_README
A    addons/mp3/common.c
A    addons/mp3/huffman.h
A    addons/mp3/tabinit.c
A    addons/mp3/Makefile
A    addons/mp3/README
A    addons/mp3/decode_i386.c
A    addons/mp3/dct64_i386.c
A    addons/mp3/MPGLIB_TODO
A    addons/mp3/mpg123.h
A    addons/mp3/layer3.c
A    addons/mp3/mpglib.h
A    addons/mp3/decode_ntom.c
A    addons/mp3/interface.c
Se exportó la revisión 202.
```

```
$ sudo contrib/scripts/install_prereq install
```

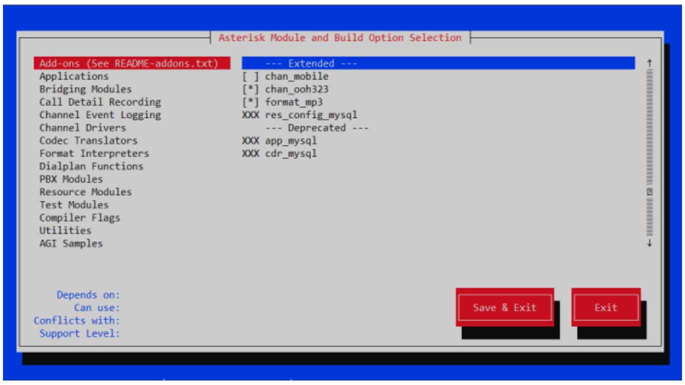
```
Configurando libgvc6 (2.40.1-6) ...
Configurando graphviz (2.40.1-6) ...
Configurando dh-autoreconf (19) ...
Configurando odbcinst1debian2:amd64 (2.3.6-0.1) ...
Configurando unixodbc-dev:amd64 (2.3.6-0.1) ...
Configurando odbcinst (2.3.6-0.1) ...
Configurando debhelper (12.1.1) ...
Configurando dh-strip-nondeterminism (1.1.2-1) ...
Configurando vpb-driver-source (4.2.61-1) ...
Procesando disparadores para libglib2.0-0:amd64 (2.58.3-2+deb10u2) ...
No se encontró ningún archivo de esquemas: sin hacer nada.
Procesando disparadores para libc-bin (2.28-10) ...
Procesando disparadores para systemd (241-7~deb10u7) ...
Procesando disparadores para man-db (2.8.5-2) ...
Configurando libgmime-2.6-dev (2.6.23+dfsg1-4) ...
Configurando libgmime-3.0-dev (3.2.1-1) ...

#############################################
## install completed successfully
#############################################
```

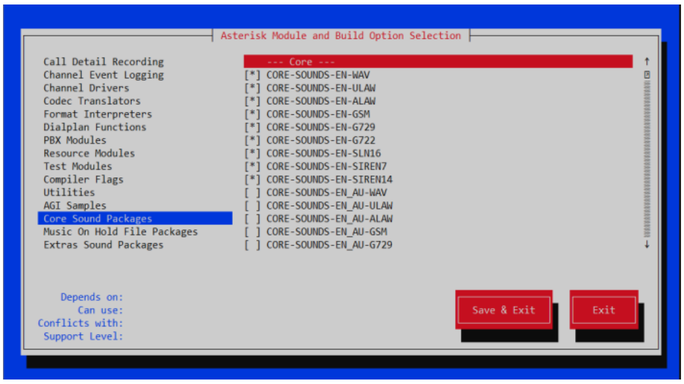
```
$ sudo ./configure
```

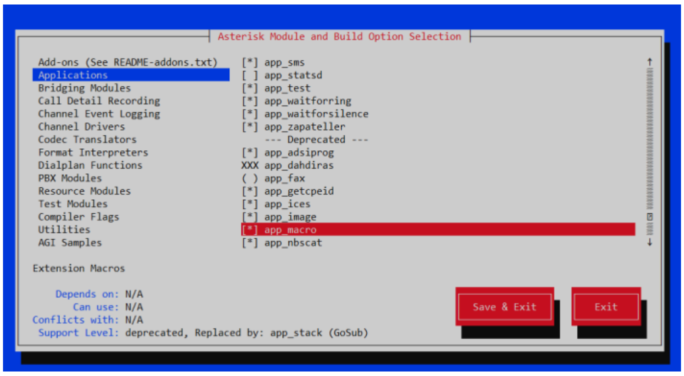
```
usu@debian: /usr/src/asterisk-18.3.0
configure: Menuselect build configuration successfully completed

               .$$$$$$$$$$$$$$$=..
            .$7$7..          .7$$7:.
          .$$:.                 ,$7.7
        .$7.     7$$$$           .$$77
     ..$$.       $$$$$            .$$$7
    ..7$   .?.   $$$$$   .?.       7$$$.
   $.$.   .$$$7. $$$$7 .7$$$.      .$$$.
 .777.   .$$$$$$77$$$77$$$$$7.      $$$,
 $$$~      .7$$$$$$$$$$$$$7.       .$$$.
.$$7          .7$$$$$$$7:          ?$$$.
$$$          ?7$$$$$$$$$$I        .$$$7
$$$       .7$$$$$$$$$$$$$$$$      :$$$.
$$$       $$$$$$7$$$$$$$$$$$$    .$$$.
$$$        $$$   7$$$7  .$$$    .$$$.
$$$$             $$$$7         .$$$.
7$$$7            7$$$$        7$$$
 $$$$$                        $$$
  $$$$7.                       $$  (TM)
   $$$$$$$.           .7$$$$$$  $$
     $$$$$$$$$$$$7$$$$$$$$$.$$$$$$
       $$$$$$$$$$$$$$$$.

configure: Package configured for:
configure: OS type  : linux-gnu
configure: Host CPU : x86_64
configure: build-cpu:vendor:os: x86_64 : pc : linux-gnu :
configure: host-cpu:vendor:os: x86_64 : pc : linux-gnu :
usu@debian:/usr/src/asterisk-18.3.0$
```

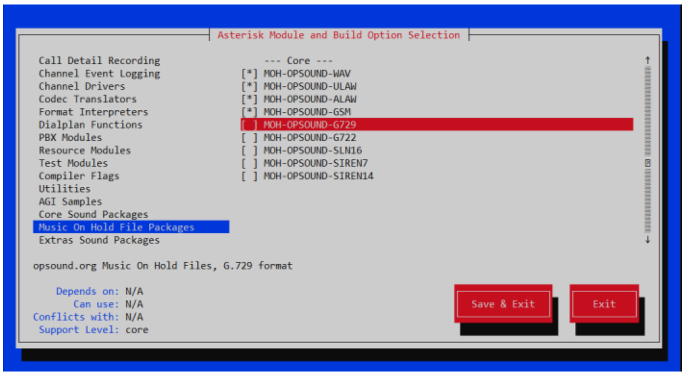
Seleccionamos los siguientes elementos:

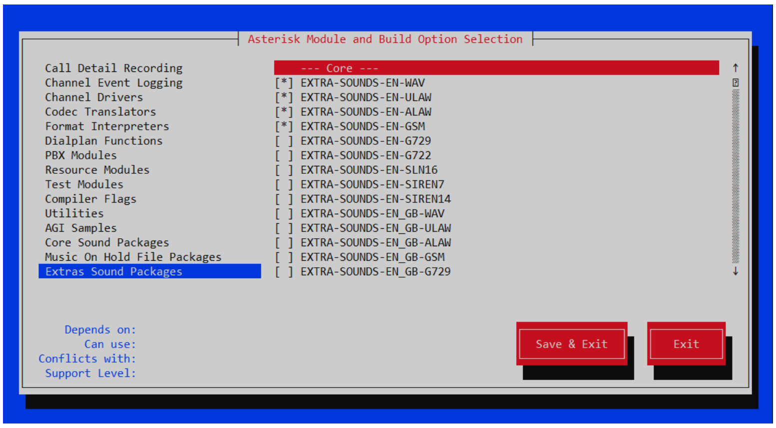
```
$ sudo make menuselect
```



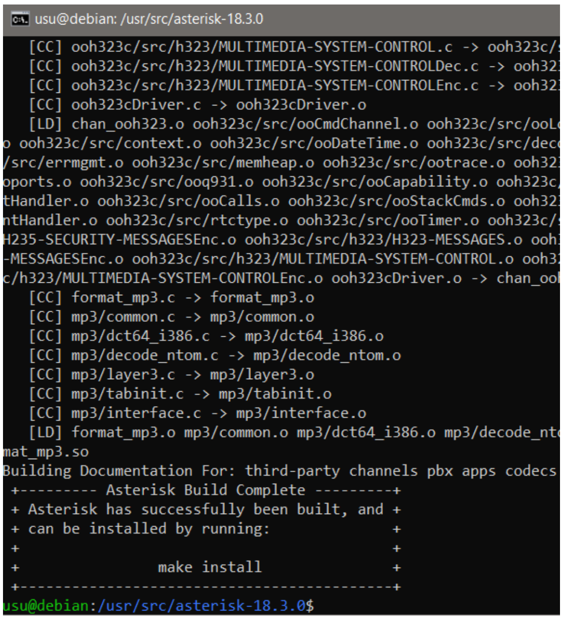




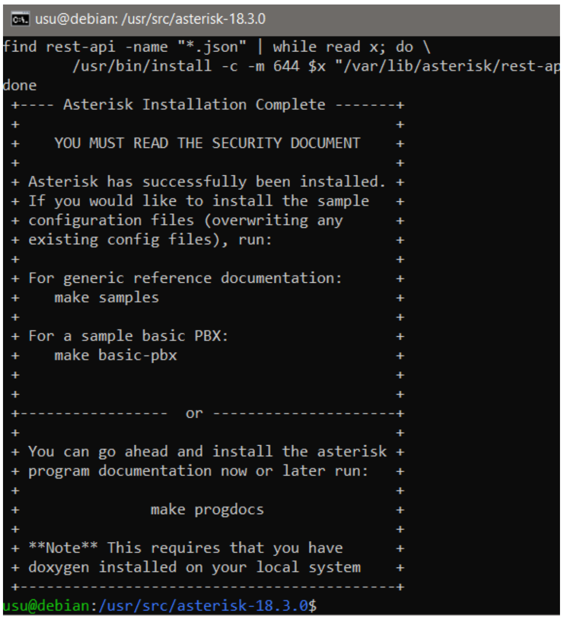





```
$ sudo make
```




```
$ sudo make install
```



Configuración

1. Archivos de configuración

```
usu@debian:/etc/radcli$ ls /etc/asterisk/
acl.conf                 cel_odbc.conf            iaxprov.conf             res_ldap.conf
adsi.conf                cel_pgsql.conf           indications.conf         res_odbc.conf
agents.conf              cel_sqlite3_custom.conf  logger.conf              resolver_unbound.conf
alarmreceiver.conf       cel_tds.conf             manager.conf             res_parking.conf
alsa.conf                chan_dahdi.conf          meetme.conf              res_pgsql.conf
amd.conf                 chan_mobile.conf         mgcp.conf                res_pktccops.conf
app_mysql.conf           cli_aliases.conf         minivm.conf              res_snmp.conf
app_skel.conf            cli.conf                 misdn.conf               res_stun_monitor.conf
ari.conf                 cli_permissions.conf     modules.conf             rtp.conf
ast_debug_tools.conf     codecs.conf              motif.conf               say.conf
asterisk.adsi            confbridge.conf          musiconhold.conf         sip.conf
asterisk.conf            config_test.conf         muted.conf               sip_notify.conf
calendar.conf            console.conf             ooh323.conf              skinny.conf
ccss.conf                dbsep.conf               osp.conf                 sla.conf
cdr_adaptive_odbc.conf   dnsmgr.conf              oss.conf                 smdi.conf
cdr_beanstalkd.conf      dsp.conf                 phone.conf               sorcery.conf
cdr.conf                 dundi.conf               phoneprov.conf           ss7.timers
cdr.conf.copia           enum.conf                pjproject.conf           stasis.conf
cdr_custom.conf          extconfig.conf           pjsip.conf               statsd.conf
cdr_manager.conf         extensions.ael           pjsip_notify.conf        stir_shaken.conf
cdr_mysql.conf           extensions.conf          pjsip_wizard.conf        telcordia-1.adsi
cdr_odbc.conf            extensions.lua           prometheus.conf          test_sorcery.conf
cdr_pgsql.conf           extensions_minivm.conf   queuerules.conf          udptl.conf
cdr_sqlite3_custom.conf  features.conf            queues.conf              unistim.conf
cdr_syslog.conf          festival.conf            res_config_mysql.conf    users.conf
cdr_tds.conf             followme.conf            res_config_sqlite3.conf  voicemail.conf
cel_beanstalkd.conf      func_odbc.conf           res_config_sqlite.conf   vpb.conf
cel.conf                 hep.conf                 res_corosync.conf        xmpp.conf
cel.conf.copia           http.conf                res_curl.conf
cel_custom.conf          iax.conf                 res_fax.conf
```

Sip.conf

Una configuración básica del registro de terminales y usuarios consistiría en la utilización de una plantilla para los usuarios con una configuración común:

- *Type-friend* permite al usuario realizar y recibir llamadas.
- *Context* establece el contexto de llamadas que puede realizar el usuario.

```
usu@debian: /etc/asterisk
  GNU nano 3.2                                              sip.conf

[general]
context=public                  ; Default context for incoming calls. Defaults to 'default'
allowoverlap=no                 ; Disable overlap dialing support. (Default is yes)
udpbindaddr=0.0.0.0             ; IP address to bind UDP listen socket to (0.0.0.0 binds to all)
tcpenable=no                    ; Enable server for incoming TCP connections (default is no)
tcpbindaddr=0.0.0.0             ; IP address for TCP server to bind to (0.0.0.0 binds to all interfaces)
transport=udp                   ; Set the default transports.  The order determines the primary default transport.
srvlookup=yes                   ; Enable DNS SRV lookups on outbound calls

qualify=yes                     ; Monitorear conexión de softphones
language=es                     ; Idioma defecto es
disallow=all                    ; Desactivamos codificadores
allow=alaw, ulaw

[my-codecs](!)                    ; a template for my preferred codecs
        disallow=all
        allow=ilbc
        allow=g729
        allow=gsm
        allow=g723
        allow=ulaw

[ulaw-phone](!)                   ; and another one for ulaw-only
        disallow=all
        allow=ulaw

[usuario](!)                      ;plantilla usuario
        type=friend
        host=dynamic
        context=ctx-prueba

[ext1000](usuario)
        username=user1000
        secret=casterisk

[ext1003](usuario)
        username=user1003
        secret=casterisk
```

Extensions.conf

En este archivo definiremos las reglas que va a tener una llamada.

```
usu@debian: /etc/asterisk
 GNU nano 3.2                    extensions.conf

[ctx-prueba]
exten => 1000,1,Dial(SIP/ext1000,15,tT) ;nº ext, priority, app
exten => 1003,1,Dial(SIP/ext1003,15,tT)
```

Buzón de correo

El buzón de voz se controla mediante:

- *VoiceMail()* que permite dejar un mensaje de voz en el buzón cuando la llamada no ha sido atendida.
- *VoiceMailMain()* es un menú que permite interactuar con los mensajes del buzón.

Para configurar los buzones de voz se utiliza el fichero voicemail.conf:

```
usu@debian: ~
 GNU nano 3.2            /etc/asterisk/voicemail.conf            Modificado

[general]
format=wav49|gsm|wav
serveremail=asterisk
attach=yes
skipms=3000
maxsilence=10
silencethreshold=128
maxlogins=3
emaildateformat=%A, %B %d, %Y at %r
pagerdateformat=%A, %B %d, %Y at %r
sendvoicemail=yes ; Allow the user to compose and send a voicemail while inside

[default]
1000 => 123
1003 => 123
1005 => 123
1007 => 123
1009 => 123
```

Ejecutando el comando *voicemail show users* en el terminal CLI se obtiene todos los mensajes de los usuarios:

```
usu@debian: ~
debian*CLI> voicemail show users
Context    Mbox  User                    Zone       NewMsg
default    1000                                          3
default    1003                                          0
default    1005                                          1
default    1007                                          0
default    1009                                          0
5 voicemail users configured.
debian*CLI>
```

El fichero extensions.conf se configura para cada extensión las acciones que se realizarán. Una configuración inicial podría ser:

```
usu@debian: /etc/asterisk
 GNU nano 3.2                    extensions.conf

[ctx-prueba]
exten => 1000,1,Dial(SIP/ext1000,15,tT) ;nº ext, priority, app
exten => _3000,1,VoiceMailMain(1000)
exten => 1003,1,Dial(SIP/ext1003,15,tT)
exten => _3003,1,VoiceMailMain(1003)
```

La configuración avanzada que se plantea es:

```
exten => _3XX,1,VoiceMailMain(100X)
;------------NUMEROS-----------
exten => _100X,1,Dial(SIP/ext${EXTEN},20,Ttm)
    same => n,VoiceMail(${EXTEN})
    same => n,HangUp()
```

Cuando se llama a una extensión se utilizan expresiones regulares (ej: _300X) para crear una extensión que dinámicamente pueda igualar a múltiples extensiones (3000-3009) y también hacemos uso de variables “${ }”

Redirección de llamadas

Para establecer redirecciones personalizadas se utiliza el fichero features.conf. Nosotros hemos decidido usar la configuración predeterminada.

```
usu@debian: ~
GNU nano 3.2                    /etc/asterisk/features.conf.copia                    Modificado

[featuremap]
;blindxfer => #1              ; Blind transfer  (default is #) -- Make sure to set the T and/or t option in the Dial() or Queue() app call!
;disconnect => *0             ; Disconnect  (default is *) -- Make sure to set the H and/or h option in the Dial() or Queue() app call!
;automon => *1                ; One Touch Record a.k.a. Touch Monitor -- Make sure to set the W and/or w option in the Dial() or Queue() app call!
;atxfer => *2                 ; Attended transfer  -- Make sure to set the T and/or t option in the Dial() or Queue()  app call!
;parkcall => #72              ; Park call (one step parking)  -- Make sure to set the K and/or k option in the Dial() app call!
;automixmon => *3             ; One Touch Record a.k.a. Touch MixMonitor -- Make sure to set the X and/or x option in the Dial() or Queue() app call!
```

Para que las redirecciones tengan efecto se deberá añadir a la función Dial( ) la sigla Tt.

```
;------------NUMEROS-----------
exten => _100X,1,Dial(SIP/ext${EXTEN},20,Ttm)
    same => n,VoiceMail(${EXTEN})
    same => n,HangUp()
```

Llamadas en grupo

En el fichero extension.conf añadimos las extensiones para crear/entrar a una llamada grupal, de forma que a la extensión 4XX acceden los usuarios y a la 5XX los administradores (necesitan contraseña). Esta configuración se realiza en el archivo confbridge.conf.

```
usu@debian: /etc/asterisk
GNU nano 3.2            /etc/asterisk/extensions.conf

[ctx-prueba]
;------------CONFIG-----------
exten => _1XX,1,Goto(menu,s,1)                                     ;MENU
exten => _3XX,1,VoiceMailMain(100X)                                ;VoiceMail MENU
exten => _4XX,1,ConfBridge(1,default_bridge,default_user)          ;Conferencia User
exten => _5XX,1,ConfBridge(1,default_bridge,admin_user)            ;Conferencia Admin
```

```
usu@debian: ~
GNU nano 3.2       /etc/asterisk/confbridge.conf

[general]

[admin_user]
    type=user
    pin=7777
    marked=yes
    admin=yes
    music_on_hold_when_empty=yes
    announce_user_count=yes

[default_user]
    type=user
    ;pin=1234
    wait_marked=yes
    end_marked=yes
    music_on_hold_when_empty=yes
    announce_user_count=yes
```

Lenguaje castellano

```
$ sudo apt-get install asterisk-prompt-es

$ sudo asterisk-core-sounds-es asterisk-core-sounds-es-gsm asterisk-core-sounds-es-wav asterik-core-sounds-es-g722
```

Si realizamos una llamada al menú de buzón de voz podemos comprobar que el idioma se ha instalado correctamente y las locuciones se realizan en español

```
usu@debian: ~
 Using SIP RTP CoS mark 5
    > 0x7f4a90027510 -- Strict RTP learning after remote address set to: 192
  -- Executing [300@ctx-prueba:1] VoiceMailMain("SIP/ext1000-00000004", "100
    > 0x7f4a90027510 -- Strict RTP switching to RTP target address 192.168.
  -- <SIP/ext1000-00000004> Playing 'vm-login.slin16' (language 'es')
    > 0x7f4a90027510 -- Strict RTP learning complete - Locking on source ad
  -- <SIP/ext1000-00000004> Playing 'vm-password.slin16' (language 'es')
  -- <SIP/ext1000-00000004> Playing 'vm-youhave.slin16' (language 'es')
  -- <SIP/ext1000-00000004> Playing 'digits/3.slin16' (language 'es')
  -- <SIP/ext1000-00000004> Playing 'vm-messages.slin16' (language 'es')
  -- <SIP/ext1000-00000004> Playing 'vm-INBOX.slin16' (language 'es')
  -- <SIP/ext1000-00000004> Playing 'vm-onefor.slin16' (language 'es')
```

Música en espera

La música en espera la configuramos desde el archivo musiconhold.conf, en el que indicamos la ruta en la que se encuentran los archivos mp3 que debe reproducir.

```
usu@debian: /etc/asterisk
  GNU nano 3.2              musiconhold.conf

[default]
    mode=files
    directory=/var/lib/asterisk/mohmp3
    random=yes
```

En el archivo extensions.conf definimos una extensión desde la que podemos escuchar la música con MusicOnHold(). Para añadir esta música a una llamada de dial se deberá añadir la sigla m a Tt:

```
usu@debian: /etc/asterisk
  GNU nano 3.2                         extensions.conf

exten => _6XX,1,Answer()
    same => n,MusicOnHold(mp3,60)
    same => n,Wait(5)
```

```
;------------NUMEROS-----------
exten => _100X,1,Dial(SIP/ext${EXTEN},20,Ttm)
    same => n,VoiceMail(${EXTEN})
    same => n,HangUp()

exten => _200X,1,Dial(SIP/ag${EXTEN},20,Ttm)
    same => n,HangUp()
```

Descargamos de una librería mp3 un politono para probar la configuración.

```
Seleccionar usu@debian: ~
usu@debian:/var/lib/asterisk/mohmp3$ sudo wget https://s103.123apps.com/acutter/d/s10346vScUMo_mp3_ncdvVpWs.mp3
[sudo] password for usu:
--2021-03-17 21:45:49--  https://s103.123apps.com/acutter/d/s10346vScUMo_mp3_ncdvVpWs.mp3
Resolviendo s103.123apps.com (s103.123apps.com)... 159.69.57.46
Conectando con s103.123apps.com (s103.123apps.com)[159.69.57.46]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 3881730 (3,7M) [application/octet-stream]
Grabando a: “s10346vScUMo_mp3_ncdvVpWs.mp3”

s10346vScUMo_mp3_ncdvVpWs.mp 100%[===========================================>]   3,70M  15,0MB/s
   en 0,2s

2021-03-17 21:45:50 (15,0 MB/s) - “s10346vScUMo_mp3_ncdvVpWs.mp3” guardado [3881730/3881730]
```

Call Center

Para emular el Call Center primero integramos la figura de los agentes en el fichero sip.conf asignándole el contexto ventas:

```
;-------------------------
[ag2000](agente-ventas)
    username=ag2000
    secret=ventas

[ag2001](agente-ventas)
    username=ag2001
    secret=ventas

[ag2002](agente-ventas)
    username=ag2002
    secret=ventas
```

En extensions.conf se crea un menú en la extensión 1XX utilizando Festival para la conversión de texto a voz con la función Goto( ) y WaitExten( ).

Una vez dentro del menú se seleccionará la extensión deseada y se direccionará la llamada. Nuestros agentes se encuentran en el contexto “ventas” por lo que marcando la extensión 2 el menú nos añadirá a la cola de la extensión 200.

```
;___________________________________________________________________________
[menu]
exten => s,1,Answer(500)
    same => n,Wait(3)
    same => n,Festival(pulse uno para acceder al departamento de compras)
    same => n,Festival(pulse dos para acceder al departamento de ventas)
    same => n,Festival(pulse tres para acceder al departamento de logistica)
    same => n,WaitExten(5)
    same => n,Goto(menu,s,1)

exten => 1,1,Festival(conectando con agente disponible)
    same => n,Goto(compras)

exten => 2,1,Festival(conectando con agente disponible)
    same => n,Goto(ventas,200,1)

exten => 3,1,Festival(conectando con agente disponible)
    same => n,Goto(logistica)

[ventas]
exten => 200,1,Queue(cola-ventas)
exten => _200X,1,Dial(SIP/ag${EXTEN})

[logistica]
exten => 400,1,ConfBridge(1,default_bridge,default_user)
```

En agents.conf se asigna un grupo de prioridad a cada agente:

```
usu@debian: /etc/asterisk
  GNU nano 3.2          agents.conf

[general]
    autologoff= 15
    wrapuptime = 0
    musiconhold = default

    group = 1
    agent = 2100,200,Ventas 1
    agent = 2101,200,Ventas 2
```

Y en queues.conf configuramos las características de la cola (música en espera, tiempo entre locuciones, registro de agentes…)

```
usu@debian: /etc/asterisk
  GNU nano 3.2          queues.conf

[general]
    persistentmembers = yes
    monitor-type = MixMonitor

    language=es
    autofill = yes

[cola-ventas]
    musiconhold = default
    strategy = ringall
    timeout = 15
    retry = 5
    wrapuptime = 0

    maxlen = 0
    announce-holdtime = no
    periodic-announce = queue-periodic-announce
    periodic-announce-frequency=20

    member => SIP/ag2000
    member => SIP/ag2001
    member => SIP/ag2002
    member => Agent/@1
```

En el CLI mediante el comando “queue show” nos mostrará los agentes disponibles de la cola (*Not in use*)

```
usu@debian: ~
debian*CLI> queue show
cola-ventas has 0 calls (max unlimited) in 'ringall' strategy (38s holdtime, 5s talktime), W:0, C:2, A:0, SL:0.0%, SL2:0.
0% within 0s
   Members:
      Agent/@1 (ringinuse enabled) (Invalid) has taken no calls yet
      SIP/ag2000 (ringinuse enabled) (Not in use) has taken 1 calls (last was 358 secs ago)
      SIP/ag2001 (ringinuse enabled) (Not in use) has taken no calls yet
      SIP/ag2002 (ringinuse enabled) (Not in use) has taken 1 calls (last was 1260 secs ago)
   No Callers
```

Cuando un agente conteste nuestra llamada se mostrará ocupado (*in call*):

```
usu@debian: ~
debian*CLI> queue show
cola-ventas has 0 calls (max unlimited) in 'ringall' strategy (29s holdtime, 5s talktime), W:0, C:2, A:0, SL:0.0%, SL2:0.
0% within 0s
   Members:
      Agent/@1 (ringinuse enabled) (Invalid) has taken no calls yet
      SIP/ag2000 (ringinuse enabled) (Not in use) has taken 1 calls (last was 1298 secs ago)
      SIP/ag2001 (ringinuse enabled) (Not in use) has taken no calls yet
      SIP/ag2002 (ringinuse enabled) (in call) (Not in use) has taken 1 calls (last was 2200 secs ago)
   No Callers
```

La llamada resultante se mostrará en el CLI de la siguiente forma:

```
usu@debian: ~
debian*CLI>
  == Using SIP RTP CoS mark 5
       > 0x7fd638039ea0 -- Strict RTP learning after remote address set to: 192.168.1.143:8000
    -- Executing [100@ctx-prueba:1] Goto("SIP/ext1000-00000014", "menu,s,1") in new stack
    -- Goto (menu,s,1)
    -- Executing [s@menu:1] Answer("SIP/ext1000-00000014", "500") in new stack
       > 0x7fd638039ea0 -- Strict RTP switching to RTP target address 192.168.1.143:8000 as source
    -- Executing [s@menu:2] Wait("SIP/ext1000-00000014", "3") in new stack
    -- Executing [s@menu:3] Festival("SIP/ext1000-00000014", "pulse uno para acceder al departamento de compras") in new stack
       > 0x7fd638039ea0 -- Strict RTP learning complete - Locking on source address 192.168.1.143:8000
    -- Executing [s@menu:4] Festival("SIP/ext1000-00000014", "pulse dos para acceder al departamento de ventas") in new stack
    -- Executing [s@menu:5] Festival("SIP/ext1000-00000014", "pulse tres para acceder al departamento de logistica") in new stack
    -- Executing [s@menu:6] WaitExten("SIP/ext1000-00000014", "5") in new stack
    -- Executing [2@menu:1] Festival("SIP/ext1000-00000014", "conectando con agente disponible") in new stack
    -- Executing [2@menu:2] Goto("SIP/ext1000-00000014", "ventas,200,1") in new stack
    -- Goto (ventas,200,1)
    -- Executing [200@ventas:1] Queue("SIP/ext1000-00000014", "cola-ventas") in new stack
    -- Started music on hold, class 'default', on channel 'SIP/ext1000-00000014'
  == Using SIP RTP CoS mark 5
    -- Called SIP/ag2002
  == Using SIP RTP CoS mark 5
    -- Called SIP/ag2001
  == Using SIP RTP CoS mark 5
    -- Called SIP/ag2000
[Apr 20 00:17:31] WARNING[949][C-00000006]: channel.c:6236 request_channel: No channel type registered for 'Agent'
[Apr 20 00:17:31] WARNING[949][C-00000006]: mp3/interface.c:218 decodeMP3: Junk at the beginning of frame 49443303
    -- SIP/ag2001-00000016 is ringing
    -- SIP/ag2000-00000017 is ringing
    -- SIP/ag2002-00000015 is ringing
       > 0x7fd640021260 -- Strict RTP learning after remote address set to: 10.0.2.16:58852
    -- SIP/ag2002-00000015 answered SIP/ext1000-00000014
    -- Stopped music on hold on SIP/ext1000-00000014
    -- Channel SIP/ag2002-00000015 joined 'simple_bridge' basic-bridge <859f709d-4243-4bb3-987c-05af94581eef>
    -- Channel SIP/ext1000-00000014 joined 'simple_bridge' basic-bridge <859f709d-4243-4bb3-987c-05af94581eef>
       > Bridge 859f709d-4243-4bb3-987c-05af94581eef: switching from simple_bridge technology to native_rtp
       > Remotely bridged 'SIP/ext1000-00000014' and 'SIP/ag2002-00000015' - media will flow directly between them
       > 0x7fd640021260 -- Strict RTP learning after remote address set to: 10.0.2.16:58852
debian*CLI>
```

Sipusers en MySQL

1. Instalación mysql-server

```
$ sudo apt update
$ cd /tmp
$ wget http://repo.mysql.com/mysql-apt-config_0.8.13-1_all.deb
$ sudo dpkg -i mysql-apt-config*
$ sudo apt update
$ sudo apt install mysql-server
$ mysql_secure_installation
$ mysql -u root -p
```

```
usu@debian: /etc/asterisk
mysql> CREATE USER 'asterisk'@'%' IDENTIFIED BY 'slab';
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE DATABASE asterisk;
Query OK, 1 row affected (0.02 sec)

mysql> GRANT ALL PRIVILEGES ON asterisk.* TO 'asterisk'@'%';
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> GRANT ALL PRIVILEGES ON asterisk.* TO 'usu'@'%';
Query OK, 0 rows affected (0.01 sec)
```

2. Instalación conector odbc

Instalación de msodbcsql17

Debian

Bash Copiar

```
sudo su
curl https://packages.microsoft.com/keys/microsoft.asc | apt-key add -

#Download appropriate package for the OS version
#Choose only ONE of the following, corresponding to your OS version

#Debian 8
curl https://packages.microsoft.com/config/debian/8/prod.list > /etc/apt/sources.list.d/mssql-release.list

#Debian 9
curl https://packages.microsoft.com/config/debian/9/prod.list > /etc/apt/sources.list.d/mssql-release.list

#Debian 10
curl https://packages.microsoft.com/config/debian/10/prod.list > /etc/apt/sources.list.d/mssql-release.list

exit
sudo apt-get update
sudo ACCEPT_EULA=Y apt-get install -y msodbcsql17
```

Instalación de libmyodbc8

```
$ wget https://dev.mysql.com/get/Downloads/Connector-ODBC/8.0/mysql-connector-odbc_8.0.23-1debian10_amd64.deb

$ sudo dpkg -i mysql-connector-odbc_8.0.23-1debian10_amd64.deb
```

```
usu@debian: ~/asterisk
usu@debian:~/asterisk$ wget https://dev.mysql.com/get/Downloads/Connector-ODBC/8.0/mysql-connector
-odbc_8.0.23-1debian10_amd64.deb
--2021-04-10 01:15:11--  https://dev.mysql.com/get/Downloads/Connector-ODBC/8.0/mysql-connector-od
bc_8.0.23-1debian10_amd64.deb
Resolviendo dev.mysql.com (dev.mysql.com)... 137.254.60.11
Conectando con dev.mysql.com (dev.mysql.com)[137.254.60.11]:443... conectado.
Petición HTTP enviada, esperando respuesta... 302 Found
Localización: https://cdn.mysql.com//Downloads/Connector-ODBC/8.0/mysql-connector-odbc_8.0.23-1deb
ian10_amd64.deb [siguiendo]
--2021-04-10 01:15:12--  https://cdn.mysql.com//Downloads/Connector-ODBC/8.0/mysql-connector-odbc_
8.0.23-1debian10_amd64.deb
Resolviendo cdn.mysql.com (cdn.mysql.com)... 2.17.153.148
Conectando con cdn.mysql.com (cdn.mysql.com)[2.17.153.148]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 1550976 (1,5M) [application/x-debian-package]
Grabando a: “mysql-connector-odbc_8.0.23-1debian10_amd64.deb”

mysql-connector-odbc_8.0 100%[===============================>]   1,48M  --.-KB/s    en 0,03s

2021-04-10 01:15:12 (47,5 MB/s) - “mysql-connector-odbc_8.0.23-1debian10_amd64.deb” guardado [1550
976/1550976]

usu@debian:~/asterisk$ ls
asterisk-addons  mysql-connector-odbc_8.0.23-1debian10_amd64.deb
usu@debian:~/asterisk$ sudo dpkg -i mysql-connector-odbc_8.0.23-1debian10_amd64.deb
(Leyendo la base de datos ... 60417 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar mysql-connector-odbc_8.0.23-1debian10_amd64.deb ...
Un-registering Unicode driver
Success: Usage count is 0
Un-registering ANSI driver
Success: Usage count is 0
Desempaquetando mysql-connector-odbc:amd64 (8.0.23-1debian10) sobre (8.0.23-1debian10) ...
Configurando mysql-connector-odbc:amd64 (8.0.23-1debian10) ...
Registering Unicode driver from in file
Success: Usage count is 1
Registering ANSI driver from in file
Success: Usage count is 1
```

Resultado:

```
usu@debian: /usr/lib/x86_64-linux-gnu/odbc
usu@debian:~$ cd /usr/lib/x86_64-linux-gnu/odbc/
usu@debian:/usr/lib/x86_64-linux-gnu/odbc$ ls
libesoobS.so     libnn.so             libodbcmyS.so    liboplodbcS.so
libmimerS.so     libodbcdrvcfg1S.so   libodbcnnS.so    liboraodbcS.so
libmyodbc8a.so   libodbcdrvcfg2S.so   libodbcpsqlS.so  libsapdbS.so
libmyodbc8w.so   libodbcminiS.so      libodbctxtS.so   libtdsS.so
usu@debian:/usr/lib/x86_64-linux-gnu/odbc$
```

Instalación de Add-ons MySQL

```
usu@debian:~/asterisk$ wget https://repo.mysql.com/apt/debian/pool/mysql-8.0/m/mysql-community/mysql-commun
ity-client-plugins_8.0.23-1debian10_amd64.deb
--2021-04-10 22:43:42--  https://repo.mysql.com/apt/debian/pool/mysql-8.0/m/mysql-community/mysql-community
-client-plugins_8.0.23-1debian10_amd64.deb
Resolviendo repo.mysql.com (repo.mysql.com)... 95.100.245.201
Conectando con repo.mysql.com (repo.mysql.com)[95.100.245.201]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 95432 (93K) [application/x-debian-package]
Grabando a: "mysql-community-client-plugins_8.0.23-1debian10_amd64.deb"

mysql-community-client-plugins_8.0.23-1debian10_amd 100%[=================================================
==============================================================>]  93,20K  --.-KB/s    en 0,007s

2021-04-10 22:43:42 (13,7 MB/s) - "mysql-community-client-plugins_8.0.23-1debian10_amd64.deb" guardado [954
32/95432]
usu@debian:~/asterisk$ sudo dpkg -i mysql-community-client-plugins_8.0.23-1debian10_amd64.deb
Seleccionando el paquete mysql-community-client-plugins previamente no seleccionado.
(Leyendo la base de datos ... 60062 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar mysql-community-client-plugins_8.0.23-1debian10_amd64.deb ...
Desempaquetando mysql-community-client-plugins (8.0.23-1debian10) ...
Configurando mysql-community-client-plugins (8.0.23-1debian10) ...
```

```
$ sudo apt-get install libmysqlclient-dev
$ cd /usr/src/asterisk-18.2.1/
$ sudo ./configure
$ sudo make clean
$ sudo make install
$ sudo make menuselect
```

```
                  Asterisk Module and Build Option Selection

 Add-ons (See README-addons.txt)        --- Extended ---
 Applications                       [ ] chan_mobile
 Bridging Modules                   [*] chan_ooh323
 Call Detail Recording              [*] format_mp3
 Channel Event Logging              [*] res_config_mysql
 Channel Drivers                        --- Deprecated ---
 Codec Translators                  [*] app_mysql
 Format Interpreters                [*] cdr_mysql
 Dialplan Functions
 PBX Modules
 Resource Modules
 Test Modules
 Compiler Flags
 Utilities
 AGI Samples
 Core Sound Packages


     Depends on:
        Can use:                      Save & Exit        Exit
 Conflicts with:
  Support Level:
```

```
debian*CLI> module show like realtime
Module                    Description                              Use Count  Status      Support Level
func_realtime.so          Read/Write/Store/Destroy values from a R 0          Running              core
pbx_realtime.so           Realtime Switch                          0          Running          extended
res_realtime.so           Realtime Data Lookup/Rewrite             0          Running              core
res_sorcery_realtime.so   Sorcery Realtime Object Wizard           0          Running              core
4 modules loaded
debian*CLI> module show like mysql
Module                    Description                              Use Count  Status      Support Level
app_mysql.so              Simple Mysql Interface                   0          Running        deprecated
cdr_mysql.so              MySQL CDR Backend                        0          Running        deprecated
res_config_mysql.so       MySQL RealTime Configuration Driver      0          Running          extended
3 modules loaded
```

3. Configuración de archivos Asterisk

```
usu@debian: /usr/lib/x86_64-linux-gnu/odbc
usu@debian:/usr/lib/x86_64-linux-gnu/odbc$ cat /etc/odbc.ini
[MySQL-Asterisk]
Description=MySQL connection to 'Asterisk' database
Driver=MySQL
Database=asterisk
Server=localhost
User=usu
Password=slab
Port=3306

Trace = On
TraceFile = /var/log/odbc.log

Socket=/var/run/mysqld/mysqld.sock
Driver = /usr/lib/x86_64-linux-gnu/odbc/libmyodbc8a.so
Setup = /usr/lib/x86_64-linux-gnu/odbc/libodbcmyS.so
usu@debian:/usr/lib/x86_64-linux-gnu/odbc$ cat /etc/odbcinst.ini
[MySQL]
Description = ODBC for MySQL
Driver = /usr/lib/x86_64-linux-gnu/odbc/libmyodbc8a.so
Setup = /usr/lib/x86_64-linux-gnu/odbc/libodbcmyS.so
Driver64 = /usr/lib/x86_64-linux-gnu/odbc/libmyodbc8a.so
Setup64 = /usr/lib/x86_64-linux-gnu/odbc/libodbcmyS.so
FileUsage = 1
usu@debian:/usr/lib/x86_64-linux-gnu/odbc$ cat /etc/asterisk/res_odbc.conf
[ENV]
ODBCSYSINI => /etc
ODBCINI    => /etc/odbc.ini

[MySQL-Asterisk]
enabled => yes
dsn => MySQL-Asterisk
pre-connect => yes
username => usu
password => slab
database => asterisk
usu@debian:/usr/lib/x86_64-linux-gnu/odbc$ cat /etc/asterisk/res_odbc_additional.conf
[Asterisk]
enabled => yes
dsn => MySQL-Asterisk
pre-connect => yes
username => usu
password => slab
usu@debian:/usr/lib/x86_64-linux-gnu/odbc$ cat /etc/asterisk/extconfig.conf
[settings]
sippeers => odbc,MySQL-Asterisk,sip_peers
sipusers => odbc,MySQL-Asterisk,sip_peers
usu@debian:/usr/lib/x86_64-linux-gnu/odbc$ cat /etc/asterisk/modules.conf
[modules]
autoload=yes
noload => chan_alsa.so
noload => chan_console.so
noload => res_hep.so
noload => res_hep_pjsip.so
noload => res_hep_rtcp.so

noload => app_voicemail_imap.so
noload => app_voicemail_odbc.so

noload => cdr_mysql.so
noload => cdr_csv.so
noload => cdr_custom.so

load => res_odbc.so
load => res_config_odbc.so
preload => chan_sip.so
usu@debian:/usr/lib/x86_64-linux-gnu/odbc$
```

4. Creación de tablas

```
usu@debian: ~
mysql> CREATE TABLE `sip_peers` (
    ->         `id` int(11) NOT NULL AUTO_INCREMENT,
    ->         `name` varchar(10) NOT NULL,
    ->         `username` varchar(10) NOT NULL,
    ->         `secret` varchar(15) DEFAULT NULL,
    ->         `context` varchar(40) DEFAULT NULL,
    ->         `mailbox` varchar(40) DEFAULT NULL,
    ->         `language` varchar(40) DEFAULT NULL,
    ->         `host` varchar(40) DEFAULT NULL,
    ->         `nat` varchar(40) DEFAULT NULL,
    ->         `ipaddr` varchar(15) DEFAULT NULL,
    ->         `port` int(5) DEFAULT NULL,
    ->         `qualify` varchar(40) DEFAULT NULL,
    ->         `type` varchar(40) DEFAULT NULL,
    ->         `disallow` varchar(40) DEFAULT NULL,
    ->         `allow` varchar(40) DEFAULT NULL,
    ->         `allowoverlap` enum('yes','no') DEFAULT NULL,
    ->         `allowsubscribe` enum('yes','no') DEFAULT NULL,
    ->         PRIMARY KEY (`id`)
    -> ) ENGINE=MyISAM;
Query OK, 0 rows affected, 2 warnings (0.01 sec)
```

5. Comprobamos que la configuración está lista

Comprobamos si el conector odbc está funcionando:

```
usu@debian:/var/run/mysqld$ isql -v MySQL-Asterisk usu slab
+---------------------------------------+
| Connected!                            |
|                                       |
| sql-statement                         |
| help [tablename]                      |
| quit                                  |
|                                       |
+---------------------------------------+
SQL>
```

```
usu@debian: ~
debian*CLI> odbc show all

ODBC DSN Settings
-----------------

  Name:   MySQL-Asterisk
  DSN:    MySQL-Asterisk
    Number of active connections: 1 (out of 1)
    Logging: Disabled

debian*CLI>
```

6. Insertamos usuarios

```
usu@debian: ~
mysql> INSERT INTO asterisk.sip_peers ( name, username, secret, context, host, nat, qualify, type)
    -> VALUES ('ext9000', '1001', 'casterisk', 'ctx-prueba', 'dynamic', 'yes', 'no', 'friend');
Query OK, 1 row affected (0.01 sec)

mysql>
```

Una vez que el usuario está en la base de datos realizamos una llamada a la extensión ext1000 que se encuentra definida en el archivo “sip.conf”

```
usu@debian: /usr/lib/x86_64-linux-gnu/odbc
    -- Registered SIP 'ext9000' at 192.168.1.143:43117
  == Using SIP RTP CoS mark 5
       > 0x7fbb28032490 -- Strict RTP learning after remote address set to: 192.168.1.143:8000
    -- Executing [1000@ctx-prueba:1] Dial("SIP/ext9000-00000005", "SIP/ext1000,20,Ttm") in new stack
  == Using SIP RTP CoS mark 5
    -- Called SIP/ext1000
    -- Started music on hold, class 'default', on channel 'SIP/ext9000-00000005'
       > 0x7fbb28032490 -- Strict RTP switching to RTP target address 192.168.1.143:8000 as source
[Apr 16 00:45:58] WARNING[998][C-00000004]: mp3/interface.c:218 decodeMP3: Junk at the beginning of frame
49443303
    -- SIP/ext1000-00000006 is ringing
       > 0x7fbb28032490 -- Strict RTP learning complete - Locking on source address 192.168.1.143:8000
       > 0x7fbb3000cfe0 -- Strict RTP learning after remote address set to: 192.168.1.144:35962
    -- SIP/ext1000-00000006 answered SIP/ext9000-00000005
    -- Stopped music on hold on SIP/ext9000-00000005
    -- Channel SIP/ext1000-00000006 joined 'simple_bridge' basic-bridge <8f4f09c6-1cf4-4d92-a12a-22793191c
0b9>
    -- Channel SIP/ext9000-00000005 joined 'simple_bridge' basic-bridge <8f4f09c6-1cf4-4d92-a12a-22793191c
0b9>
       > 0x7fbb3000cfe0 -- Strict RTP switching to RTP target address 192.168.1.144:35962 as source
    -- Channel SIP/ext1000-00000006 left 'simple_bridge' basic-bridge <8f4f09c6-1cf4-4d92-a12a-22793191c0b
9>
    -- Channel SIP/ext9000-00000005 left 'simple_bridge' basic-bridge <8f4f09c6-1cf4-4d92-a12a-22793191c0b
9>
  == Spawn extension (ctx-prueba, 1000, 1) exited non-zero on 'SIP/ext9000-00000005'
```

Antes de añadir el usuario guardado en mysql en sip show peers apararecen 10 sips.

```
debian*CLI> sip show peers
Name/username             Host                                    Dyn Forcerport Comedia    ACL Port     Status      Description
                      Realtime
ag2000/ag2000             192.168.1.143                            D  Yes        No             59812    OK (2 ms)
ag2001/ag2001             (Unspecified)                            D  Yes        No             0        UNKNOWN
ag2002/ag2002             (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1000/1000              192.168.1.143                            D  Yes        No             59096    OK (1 ms)
ext1003/1003              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1005/1005              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1007/1007              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1009/1009              (Unspecified)                            D  Yes        No             0        UNKNOWN
serverA                   (Unspecified)                            D  Yes        No             0        UNKNOWN
troncalB                  192.168.1.199                               Yes        No             5060     UNREACHABLE
10 sip peers [Monitored: 2 online, 8 offline Unmonitored: 0 online, 0 offline]
debian*CLI>
```

Al introducir el nuevo usuario podemos observar que también aparece en la CLI con la etiqueta Realtime: Catched RT

```
debian*CLI> sip show peers
Name/username             Host                                    Dyn Forcerport Comedia    ACL Port     Status      Description                      Realtime
ag2000/ag2000             192.168.1.143                            D  Yes        No             59812    OK (1 ms)
ag2001/ag2001             (Unspecified)                            D  Yes        No             0        UNKNOWN
ag2002/ag2002             (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1000/1000              192.168.1.143                            D  Yes        No             59096    OK (1 ms)
ext1003/1003              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1005/1005              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1007/1007              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1009/1009              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext9000/1001              192.168.1.143                            D  Yes        Yes            39405    Unmonitored                                  Cached RT
serverA                   (Unspecified)                            D  Yes        No             0        UNKNOWN
troncalB                  192.168.1.199                               Yes        No             5060     UNREACHABLE
11 sip peers [Monitored: 2 online, 8 offline Unmonitored: 1 online, 0 offline]
```

CDR MySQL

Utilizamos CDR para crear un registro de todas las llamadas que se produzcan en la PBX.

1. Creación de la tabla

```
usu@debian: ~
mysql> CREATE TABLE `cdr` (
    ->   `calldate` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
    ->   `clid` varchar(80) NOT NULL DEFAULT '',
    ->   `src` varchar(80) NOT NULL DEFAULT '',
    ->   `dst` varchar(80) NOT NULL DEFAULT '',
    ->   `dcontext` varchar(80) NOT NULL DEFAULT '',
    ->   `channel` varchar(80) NOT NULL DEFAULT '',
    ->   `dstchannel` varchar(80) NOT NULL DEFAULT '',
    ->   `lastapp` varchar(80) NOT NULL DEFAULT '',
    ->   `lastdata` varchar(80) NOT NULL DEFAULT '',
    ->   `duration` int(11) NOT NULL DEFAULT '0',
    ->   `billsec` int(11) NOT NULL DEFAULT '0',
    ->   `disposition` varchar(45) NOT NULL DEFAULT '',
    ->   `amaflags` int(11) NOT NULL DEFAULT '0',
    ->   `accountcode` varchar(20) NOT NULL DEFAULT '',
    ->   `uniqueid` varchar(32) NOT NULL DEFAULT '',
    ->   `userfield` varchar(255) NOT NULL DEFAULT '',
    ->   `peeraccount` varchar(20) NOT NULL DEFAULT '',
    ->   `linkedid` varchar(32) NOT NULL DEFAULT '',
    ->   `sequence` int(11) NOT NULL DEFAULT '0'
    -> ) ENGINE=InnoDB;
Query OK, 0 rows affected, 4 warnings (0.05 sec)

mysql>
```

Si consultamos la tabla obtenemos el registro de las llamadas realizadas (tabla recortada para poder visualizarla):

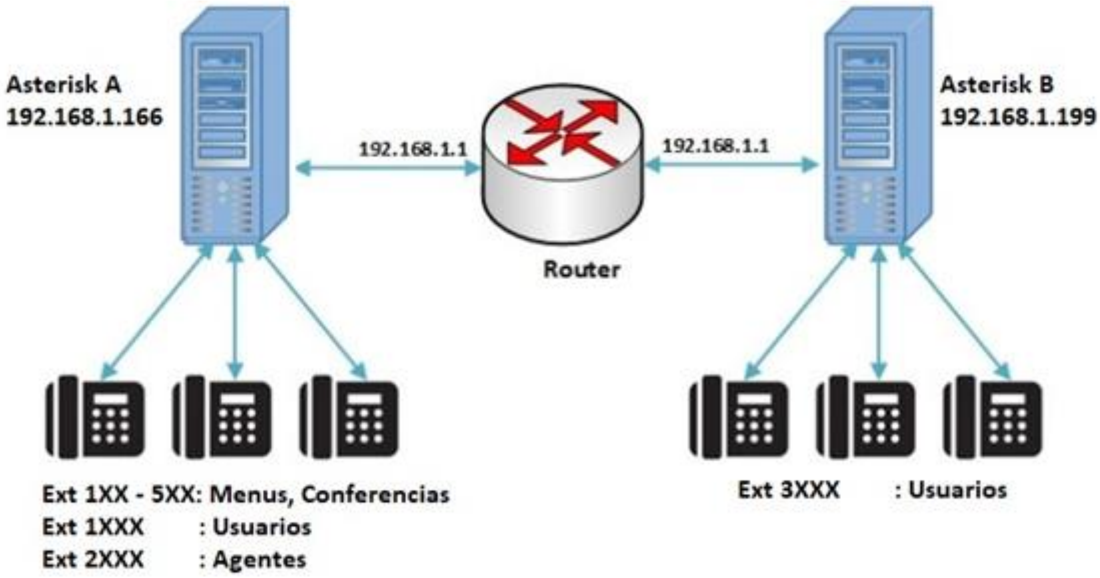
```
mysql> select * from cdr;
```

calldate	clid	src	dst	dcontext	channel	dstchannel	lastapp	lastdata
2021-04-15 19:42:36	"" <ext9000>	ext9000	1000	ctx-prueba	SIP/ext9000-00000000	SIP/ext1000-00000001	Dial	SIP/ext1000,20,Ttm
2021-04-15 19:44:41	"" <ext1000>	ext1000	1005	ctx-prueba	SIP/ext1000-00000002		Dial	SIP/ext1005,20,Ttm
2021-04-15 22:32:20	"" <ext9000>	ext9000	1000	ctx-prueba	SIP/ext9000-00000000		Dial	SIP/ext1000,20,Ttm
2021-04-15 22:32:47	"" <ext9000>	ext9000	1000	ctx-prueba	SIP/ext9000-00000001	SIP/ext1000-00000002	Dial	SIP/ext1000,20,Ttm
2021-04-15 22:33:22	"" <ext9000>	ext9000	1000	ctx-prueba	SIP/ext9000-00000003	SIP/ext1000-00000004	Dial	SIP/ext1000,20,Ttm
2021-04-15 22:33:30	"" <ext9000>	ext9000	1005	ctx-prueba	SIP/ext9000-00000003		VoiceMail	1005
2021-04-15 22:45:58	"" <ext9000>	ext9000	1000	ctx-prueba	SIP/ext9000-00000005	SIP/ext1000-00000006	Dial	SIP/ext1000,20,Ttm
2021-04-16 17:28:11	"" <ext1000>	ext1000	1003	ctx-prueba	SIP/ext1000-00000000	SIP/ext1003-00000001	Dial	SIP/ext1003,20,Ttm
2021-04-16 17:54:03	"" <ext1000>	ext1000	3000	ctx-prueba	SIP/ext1000-00000001	SIP/192.168.1.199-00000002	Dial	SIP/192.168.1.199/ext
2021-04-16 17:59:43	"" <ext1000>	ext1000	1000	ctx-prueba	SIP/ext1000-00000000	SIP/ext1000-00000001	Dial	SIP/ext1000,20,Ttm
2021-04-16 17:59:43	"" <ext1000>	ext1000	1000	ctx-prueba	SIP/ext1000-00000000		VoiceMail	1000
2021-04-16 18:18:15	"" <ext1000>	ext1000	3000	ctx-prueba	SIP/ext1000-00000005	SIP/serverA-00000006	Dial	SIP/serverA/ext3000,2
2021-04-16 18:32:50	"" <ext1000>	ext1000	3000	ctx-prueba	SIP/ext1000-00000000	SIP/troncalB-00000001	Dial	SIP/troncalB/ext3000,
2021-04-16 18:34:32	"" <ext1000>	ext1000	3000	ctx-prueba	SIP/ext1000-00000000	SIP/troncalB-00000001	Dial	SIP/troncalB/ext3000,
2021-04-16 18:34:39	"" <ext1000>	ext1000	3000	ctx-prueba	SIP/ext1000-00000002	SIP/troncalB-00000003	Dial	SIP/troncalB/ext3000,
2021-04-16 18:36:55	"" <ext3000>	ext3000	1000	ctx-prueba	SIP/troncalB-00000000	SIP/ext1000-00000001	Dial	SIP/ext1000,20,Ttm
2021-04-16 18:37:35	"" <ext1000>	ext1000	3000	ctx-prueba	SIP/ext1000-00000002	SIP/troncalB-00000003	Dial	SIP/troncalB/3000,20,
2021-04-16 18:39:07	"" <ext1000>	ext1000	3000	ctx-prueba	SIP/ext1000-00000000	SIP/troncalB-00000001	Dial	SIP/troncalB/3000,20,
2021-04-16 18:39:27	"" <ext1000>	ext1000	3000	ctx-prueba	SIP/ext1000-00000000		Hangup	
2021-04-17 21:04:41	"" <ext1000>	ext1000	600	ctx-prueba	SIP/ext1000-00000000		Stasis	hello-world
2021-04-17 23:12:23	"" <ext1000>	ext1000	1003	ctx-prueba	SIP/ext1000-00000001	SIP/ext1003-00000002	Dial	SIP/ext1003,20,Ttm
2021-04-17 23:12:42	"" <ext1000>	ext1000	300	ctx-prueba	SIP/ext1000-00000003		VoiceMailMain	100X
2021-04-17 23:13:48	"" <ext1000>	ext1000	300	ctx-prueba	SIP/ext1000-00000004		VoiceMailMain	100X

Interconectar con otra PBX

1. Esquema general

Para la interconexión de las PBX hemos decidido asignar las siguientes extensiones a cada PBX




```
;-----------------------
[ext1000](usuario)
    username=1000
    secret=casterisk

[ext1003](usuario)
    username=1003
    secret=casterisk

[ext1005](usuario)
    username=1005
    secret=casterisk

[ext1007](usuario)
    username=1007
    secret=casterisk

[ext1009](usuario)
    username=1009
    secret=casterisk
```

```
;-------------------------
[ext3000](usuario)
    username=3000
    secret=casterisk

[ext3003](usuario)
    username=3003
    secret=casterisk

[ext3005](usuario)
    username=3005
    secret=casterisk

[ext3007](usuario)
    username=3007
    secret=casterisk

[ext3009](usuario)
    username=3009
    secret=casterisk
```

En la PBX B únicamente se configurarán los usuarios para comprobar que la PBX A puede comunicarse a través del troncal con la PBX B

2. Conexión del troncal

Sip.conf

En el archivo sip.conf creamos el troncal que unirá las PBX de forma que asignamos a cada troncal la dirección ipv4 de la PBX a la que nos conectaremos.

```
[troncal->B]
    type=friend
    host=192.168.1.199
    disallow=all
    allow=alaw
    context=ctx-prueba
```

```
[troncal->A]
    type=friend
    host=192.168.1.166
    disallow=all
    allow=alaw
    context=ctx-prueba
```

Extensions.conf

Desde extensions.conf enrutaremos todas las llamadas hacia los usuarios 300X hacia la PBX B

```
;------------PBX=>B-----------
exten => _300X,1,Dial(SIP/troncal->B/${EXTEN},20,Ttm)
    same => n,VoiceMail(${EXTEN})
    same => n,HangUp()
```

```
;------------PBX=>A------------
exten => _100X,1,Dial(SIP/troncal->A/${EXTEN},20,Ttm)
    same => n,VoiceMail(${EXTEN})
    same => n,HangUp()
```

3. Comprobación en CLI

Server A

```
debian*CLI> sip show peers
Name/username             Host                                    Dyn Forcerport Comedia    ACL Port     Status
Description                      Realtime
ag2000/ag2000             (Unspecified)                            D  Yes        No             0        UNKNOWN
ag2001/ag2001             (Unspecified)                            D  Yes        No             0        UNKNOWN
ag2002/ag2002             (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1000/1000              192.168.1.86                             D  Yes        No             61352    OK (1 ms)
ext1003/1003              192.168.1.86                             D  Yes        No             51014    OK (1 ms)
ext1005/1005              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1007/1007              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext1009/1009              (Unspecified)                            D  Yes        No             0        UNKNOWN
troncal->B                192.168.1.199                               Yes        No             5060     OK (1 ms)
```

Server B

```
debian*CLI> sip show peers
Name/username             Host                                    Dyn Forcerport Comedia    ACL Port     Status
Description                      Realtime
ext3000/3000              192.168.1.86                             D  Yes        No             50999    OK (1 ms)
ext3003/3003              192.168.1.86                             D  Yes        No             51006    OK (1 ms)
ext3005/3005              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext3007/3007              (Unspecified)                            D  Yes        No             0        UNKNOWN
ext3009/3009              (Unspecified)                            D  Yes        No             0        UNKNOWN
troncal->A                192.168.1.166                               Yes        No             5060     OK (2 ms)
```

2. Llamada

Al realizar la llamada en la CLI de Asterisk podemos comprobar que se utiliza el troncal para la llamada

```
    -- Executing [3000@ctx-prueba:1] Dial("SIP/ext1000-00000006", "SIP/troncalB/3000,20,Ttm") in new stack
  == Using SIP RTP CoS mark 5
    -- Called SIP/troncalB/3000
    -- Started music on hold, class 'default', on channel 'SIP/ext1000-00000006'
       > 0x7f1aac037e30 -- Strict RTP switching to RTP target address 192.168.1.143:4000 as source
       > 0x7f1ab400c230 -- Strict RTP learning after remote address set to: 192.168.1.199:16964
    -- SIP/troncalB-00000007 is making progress passing it to SIP/ext1000-00000006
[Apr 20 02:39:55] WARNING[1464][C-00000005]: mp3/interface.c:218 decodeMP3: Junk at the beginning of frame 49443303
       > 0x7f1ab400c230 -- Strict RTP switching to RTP target address 192.168.1.199:16964 as source
       > 0x7f1aac037e30 -- Strict RTP learning complete - Locking on source address 192.168.1.143:4000
       > 0x7f1ab400c230 -- Strict RTP learning complete - Locking on source address 192.168.1.199:16964
    -- SIP/troncalB-00000007 answered SIP/ext1000-00000006
    -- Stopped music on hold on SIP/ext1000-00000006
    -- Channel SIP/troncalB-00000007 joined 'simple_bridge' basic-bridge <726f93d6-4ebb-40d4-8856-9090444a2af9>
    -- Channel SIP/ext1000-00000006 joined 'simple_bridge' basic-bridge <726f93d6-4ebb-40d4-8856-9090444a2af9>
    -- Channel SIP/ext1000-00000006 left 'simple_bridge' basic-bridge <726f93d6-4ebb-40d4-8856-9090444a2af9>
    -- Channel SIP/troncalB-00000007 left 'simple_bridge' basic-bridge <726f93d6-4ebb-40d4-8856-9090444a2af9>
  == Spawn extension (ctx-prueba, 3000, 1) exited non-zero on 'SIP/ext1000-00000006'
debian*CLI>
```

API REST

Primero se instala wscat para poder conectarse a una API de WebSocket

```
$ apt-get install npm

$ npm install -g wscat
```

Después se instala curl

```
$ apt-get install curl
```

1. Configuración de Asterisk

Http.conf

```
usu@debian: ~
    /etc/asterisk/http.conf

[general]
    servername=Asterisk
    enable=yes
    bindaddr = 0.0.0.0
```

Ari.conf

```
usu@debian: /etc/asterisk
 GNU nano 3.2                  ari.conf

[general]
    enabled = yes       ; When set to no, ARI support is disabled.
    pretty = yes

[usu]
     type = user
     read_only = no
     password = capi
```

Extensions.conf

En el archivo extensions.conf configuramos que la extensión 600 ejecute la función Stasis() que conecta el API con Asterisk

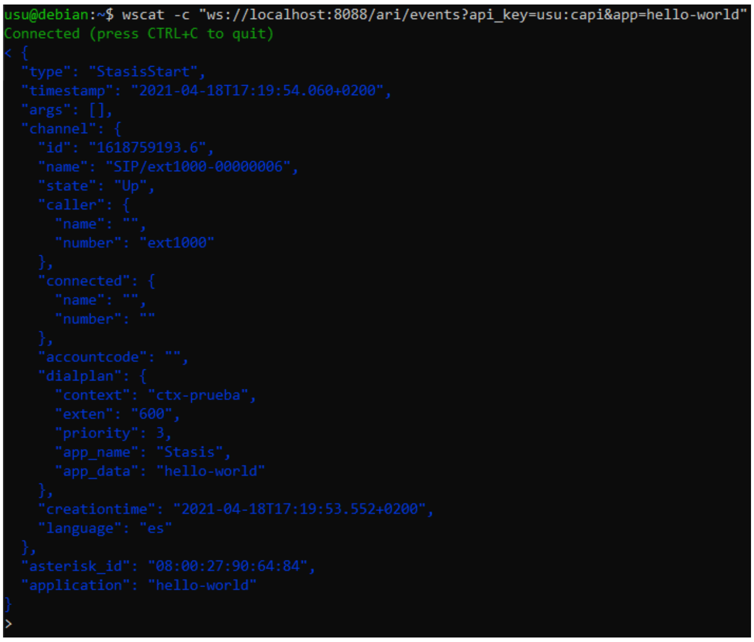
```
exten => 600,1,NoOp()
     same => n,Answer()
     same => n,Stasis(hello-world)
     same => n,Hangup()
```

2. Conexión con Asterisk mediante wscat

Mediante wscat conectamos el Api “hello-world” a Asterisk.

```
$ wscat -c "ws://localhost:8088/ari/events?api_key=usu:capi&app=hello-world"
```

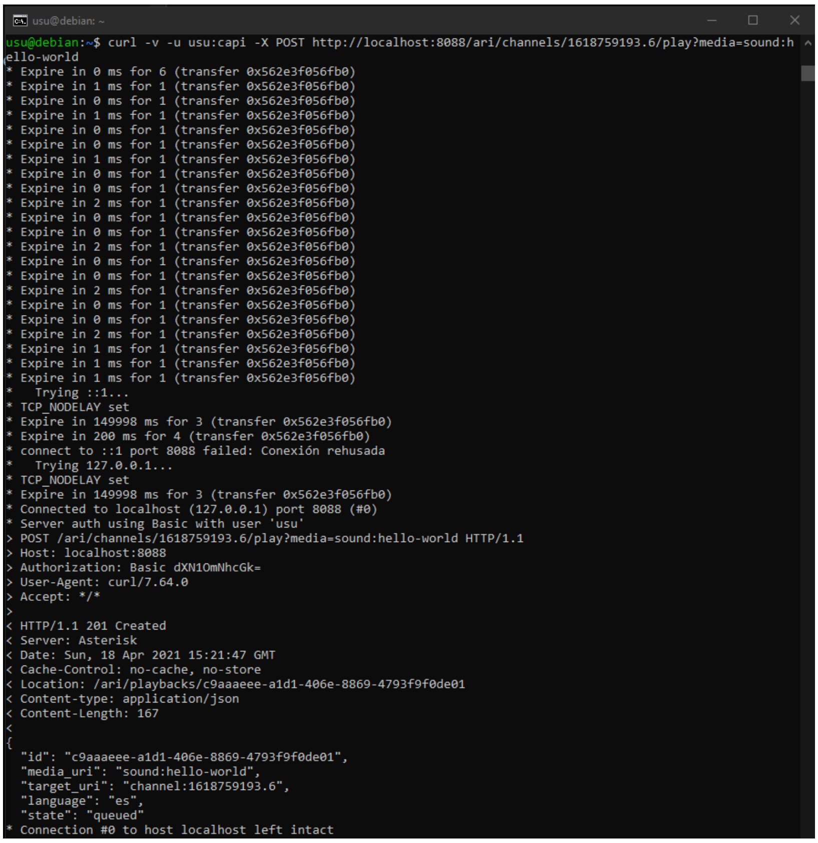
De esta forma al establecer la conexión con Asterisk y realizar la llamada en este caso desde la extensión ext1000, obtenemos la siguiente salida en wscat:



3. Post de la acción play media con curl

Mediante el uso de curl enviaremos un POST al API generada en Asterisk de forma que con la id obtenida en el wscat podemos reproducir “hello world”

```
$ curl -v -u usu:capi -X POST
http://localhost:8088/ari/channels/1618693481.0/play?media=sound:hello-world
```



En wscat se obtiene la salida de la función curl:


```
< {
  "type": "PlaybackStarted",
  "timestamp": "2021-04-18T17:21:47.896+0200",
  "playback": {
    "id": "c9aaaeee-a1d1-406e-8869-4793f9f0de01",
    "media_uri": "sound:hello-world",
    "target_uri": "channel:1618759193.6",
    "language": "es",
    "state": "playing"
  },
  "asterisk_id": "08:00:27:90:64:84",
  "application": "hello-world"
}
< {
  "type": "PlaybackFinished",
  "timestamp": "2021-04-18T17:21:48.817+0200",
  "playback": {
    "id": "c9aaaeee-a1d1-406e-8869-4793f9f0de01",
    "media_uri": "sound:hello-world",
    "target_uri": "channel:1618759193.6",
    "language": "es",
    "state": "done"
  },
  "asterisk_id": "08:00:27:90:64:84",
  "application": "hello-world"
}
```


En el CLI de Asterisk la llamada se visualiza así:

```
 Creating Stasis app 'hello-world'
  == WebSocket connection from '127.0.0.1:34700' for protocol '' accepted using version '13'
    -- Registered SIP 'ext1000' at 192.168.1.144:46273
       > Saved useragent "Zoiper rv2.10.12.3-mod" for peer ext1000
[Apr 17 23:04:27] NOTICE[6208]: chan_sip.c:24996 handle_response_peerpoke: Peer 'ext1000' is now Reachable. (4ms / 2000ms)
       > Saved useragent "Zoiper rv2.10.12.3-mod" for peer ext1003
  == Using SIP RTP CoS mark 5
       > 0x7fa42801d290 -- Strict RTP learning after remote address set to: 192.168.1.144:39116
    -- Executing [600@ctx-prueba:1] NoOp("SIP/ext1000-00000000", "") in new stack
    -- Executing [600@ctx-prueba:2] Answer("SIP/ext1000-00000000", "") in new stack
       > 0x7fa42801d290 -- Strict RTP switching to RTP target address 192.168.1.144:39116 as source
    -- Executing [600@ctx-prueba:3] Stasis("SIP/ext1000-00000000", "hello-world") in new stack
       > 0x7fa42801d290 -- Strict RTP learning complete - Locking on source address 192.168.1.144:39116
    -- <SIP/ext1000-00000000> Playing 'hello-world.slin16' (language 'es')
debian*CLI>
```

IoT

1. Servidor IoT

Para la realización del servidor IoT se ha decidido reutilizar parte de la practica GP12 de Servicios Telemáticos, de tal forma que se ha unificado el generador de valores para los sensores con el servidor web.

Además, se ha eliminado el uso de MySQL para la realización de consultas.

Iroom.py

En este script de Python mediante el uso de *flask* se visualizan unas plantillas de la página principal, login y sensores. También incluye la introducción aleatoria de valores en las rutas de los sensores.

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

from flask import Flask, url_for, session, render_template, Response, request, flash,
redirect, abort, jsonify
from flask_restful import Resource, Api
from random import randint
import json
import time

app = Flask(__name__)
app.config.from_object(__name__)
app.config.from_envvar('IROOM_SETTINGS', silent=True)
#SHELL -> export IROOM_SETTINGS=/home/usu/Iroom/config/iroom.cfg
api = Api(app)

@app.route('/')
def main():
    return render_template('index.html')

@app.route('/sensors')
def sensors():
    return render_template('sensors.html')

@app.route('/login', methods=['GET', 'POST'])
def login():
    error = None
    if request.method == 'POST':
            if request.form['username'] != app.config['USERNAME']:
                    error = 'Invalid username'
            elif request.form['password'] != app.config['PASSWORD']:
                    error = 'Invalid password'
            else:
                    session['logged_in'] = True
                    flash('Has entrado en la sesion')
                    return redirect(url_for('sensors'))
    return render_template('login.html', error=error)

@app.route('/logout')
def logout():
    session.pop('logged_in', None)
    flash('Has salido de la sesion')
    return redirect(url_for('main'))

class temperature(Resource):
    def get(self):
            value = randint(18,23)
            return {'temperature': value}

class humidity(Resource):
    def get(self):
            value = randint(29,61)
            return {'humidity': value}
```

```
class light(Resource):
    def get(self):
            value = randint(0,100)
            return {'light': value}

class sound(Resource):
    def get(self):
            value = randint(20,80)
            return {'sound': value}

class motion(Resource):
    def get(self):
            value = randint(0,1)
            return {'motion': value}

class red(Resource):
    def put(self, id):
            print ("Color rojo:"+str(id))
            return {'red': id}

class green(Resource):
    def put(self, id):
            print ("Color verde:"+str(id))
            return {'green': id}

class blue(Resource):
    def put(self, id):
            print ("Color azul:"+str(id))
            return {'blue': id}


api.add_resource(temperature, '/temperature')
api.add_resource(humidity, '/humidity')
api.add_resource(light, '/light')
api.add_resource(sound, '/sound')
api.add_resource(motion, '/motion')
api.add_resource(red, '/red/<int:id>')
api.add_resource(green, '/green/<int:id>')
api.add_resource(blue, '/blue/<int:id>')


if __name__ == "__main__":
    app.run(host='0.0.0.0', port=8000, debug=True)
```

Plantillas html

Las plantillas html utilizadas se adjuntarán con la carpeta /Iroom/templates. Nos centraremos en el script de Javascript de la plantilla sensors.html, en el que se visualizarán los datos en una tabla.

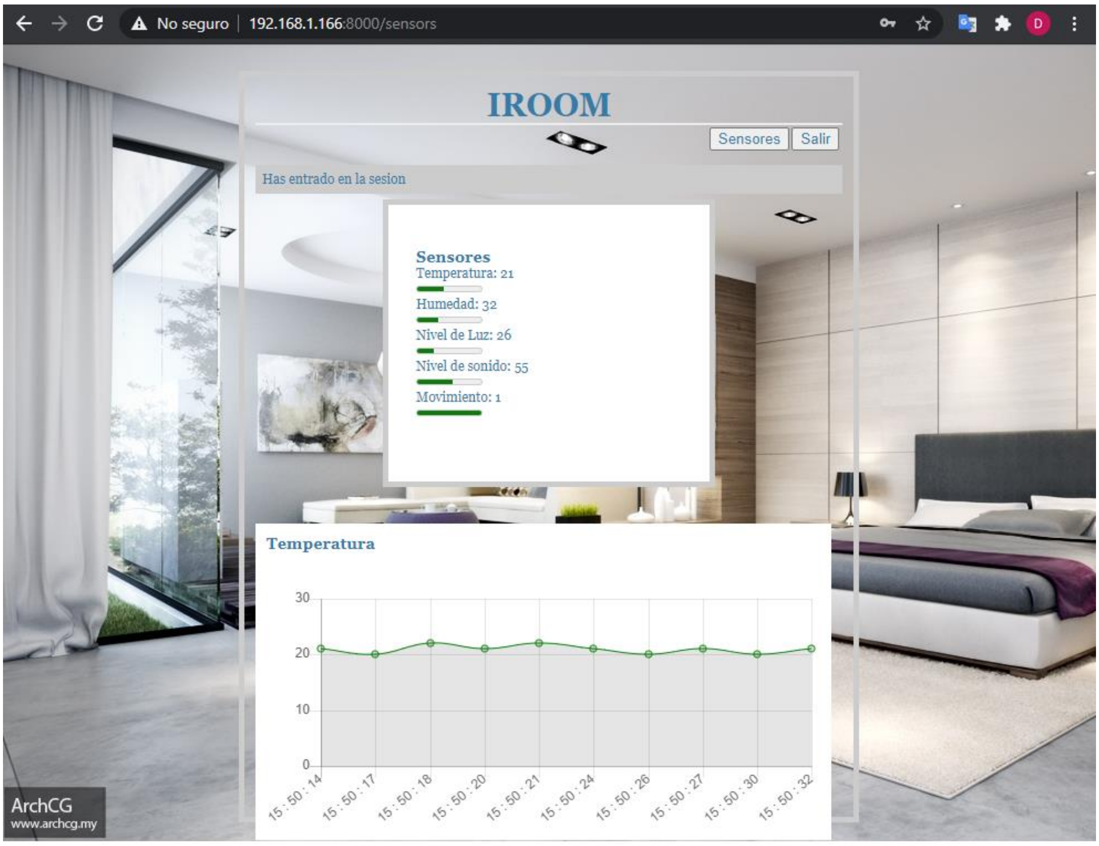
En este script la función getJson( ) obtendrá los valores de cada sensor de la url (“host-máquina/sensor”) y dependiendo de que tabla se esté mostrando añadirá el valor a la tabla.

```
<script>
var ctx_live = document.getElementById("temp_graph");
var type_sensor = ['Temperatura', 'Humedad', 'Luz', 'Sonido', 'Movimiento']
var sensor = ['temperature', 'humidity', 'light', 'sound', 'motion']
var valor_ant = 0;
var no_sens = 0;
var cont = 0;

#Configuración de la tabla
var myChart = new Chart(ctx_live, {
    type: 'line',
    data: {
    labels: [],
    datasets: [{ data: [], borderWidth: 1, borderColor:'green', label: "",}]},
    options: {
            responsive: true,
            animation: {easing: 'linear'},
```

```
            title: {display: true,  text: "",},
            legend: {display: false},scales: {yAxes: [{ticks: {beginAtZero: true,}}] }
    }
});

$(document).ready(function(){
    setInterval(getJson, 1500);
    $("#ct").click(function(){
            removeData(cont);
            myChart.update();
            valor_ant = 0;
            no_sens = 0;
            cont = 0;

            $("#sens").html(type_sensor[no_sens]);
    });
    $("#ch").click(function(){
            removeData(cont);
            myChart.update();
            valor_ant = 0;
            no_sens = 1;
            cont = 0;

            $("#sens").html(type_sensor[no_sens]);
    });
    $("#cl").click(function(){
            removeData(cont);
            myChart.update();
            valor_ant = 0;
            no_sens = 2;
            cont = 0;

            $("#sens").html(type_sensor[no_sens]);
    });
    $("#cs").click(function(){
            removeData(cont);
            myChart.update();
            valor_ant = 0;
            no_sens = 3;
            cont = 0;

            $("#sens").html(type_sensor[no_sens]);
    });
    $("#cm").click(function(){
            removeData(cont);
            myChart.update();
            valor_ant = 0;
            no_sens = 4;
            cont = 0;

            $("#sens").html(type_sensor[no_sens]);
    });

function getJson() {
      for (s in sensor) {
            $.getJSON('http://192.168.1.166:8000/' + sensor[s], function(data) {
                    key = Object.keys(data)[0];
                    value = data[key]

                    console.log(key, value)
                    if (key == 'temperature') {
                            document.getElementById('ct').innerHTML = "Temperatura: "+value;
                            document.getElementById('temperatura').setAttribute('value', 
value);
                            if(no_sens == 0) addData(value);
                    };
                    if (key == 'humidity') {
                            document.getElementById('ch').innerHTML = "Humedad: "+value;
                            document.getElementById('humedad').setAttribute('value', value);
                            if(no_sens == 1) addData(value);
                    };
                    if (key == 'light') {
```

```
                              document.getElementById('cl').innerHTML = "Nivel de Luz: "+value;
                              document.getElementById('luz').setAttribute('value', value);
                              if(no_sens == 2) addData(value);
                      };
                      if (key == 'sound') {
                              document.getElementById('cs').innerHTML = "Nivel de sonido:
"+value;
                              document.getElementById('sonido').setAttribute('value', value);
                              if(no_sens == 3) {
                                      addData(value);
                              }
                      };
                      if (key == 'motion') {
                              document.getElementById('cm').innerHTML = "Movimiento: " + value;
                              document.getElementById('movimiento').setAttribute('value',
value);
                              if(no_sens == 4){
                                      addData(value);
                              }
                      };
              });
      }
}});
```

```
Seleccionar usu@debian: ~/Iroom
usu@debian:~/Iroom$ python iroom.py
 * Serving Flask app "iroom" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://0.0.0.0:8000/ (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 140-649-794
192.168.1.138 - - [18/Apr/2021 15:52:01] "GET /humidity HTTP/1.1" 200 -
192.168.1.138 - - [18/Apr/2021 15:52:01] "GET /light HTTP/1.1" 200 -
192.168.1.138 - - [18/Apr/2021 15:52:01] "GET /sound HTTP/1.1" 200 -
192.168.1.138 - - [18/Apr/2021 15:52:01] "GET /motion HTTP/1.1" 200 -
192.168.1.138 - - [18/Apr/2021 15:52:01] "GET /temperature HTTP/1.1" 200 -
```



2. Integración de servidor con Asterisk

Función AGI( )

La integración de IoT en Asterisk se realiza con la ejecución de un script en Python. Configuramos la extensión 700 en el archivo extensions.conf.

```
exten => 700,1,Answer()
    same => n,AGI(python.py)
```

Script Python

La ruta predeterminada de ejecución de scripts de AGI es /var/lib/asterisk/agi-bin aunque si se indica la ruta completa en AGI(/*path-script*) también funcionará.

El script consiste en mediante la librería asterisk.agi (pyst2) crear una variable en Asterisk y asignarle el valor que se obtiene del servidor

```
$ sudo pip install pyst2
```

```
usu@debian: /var/lib/asterisk/agi-bin
  GNU nano 3.2                                        python.py

#!/usr/bin/env python
import sys
from asterisk.agi import *
import urllib, json

agi = AGI()
v = 111


url = ["/temperature", "/humidity", "/light", "/sound", "/motion"]

for sensor in url:
    response = urllib.urlopen("http://127.0.0.1:8000" + sensor)
    data = json.loads(response.read())
    num = data[sensor[1:]]
    agi.set_variable(sensor[1:], num)
    agi.verbose(num)
```

3. Llamada

Al realizar la llamada en el CLI podemos comprobar que se ejecuta el script y reproduce los valores correctamente.

```
debian*CLI>
  == Using SIP RTP CoS mark 5
       > 0x7fc14c036d10 -- Strict RTP learning after remote address set to: 192.168.1.142:4006
    -- Executing [700@ctx-prueba:1] Answer("SIP/ext1000-00000004", "") in new stack
       > 0x7fc14c036d10 -- Strict RTP learning after remote address set to: 192.168.1.142:4006
    -- Executing [700@ctx-prueba:2] AGI("SIP/ext1000-00000004", "python.py") in new stack
    -- Launched AGI Script /var/lib/asterisk/agi-bin/python.py
       > 0x7fc14c036d10 -- Strict RTP switching to RTP target address 192.168.1.142:4006 as source
 python.py: 20
 python.py: 57
 python.py: 80
 python.py: 34
 python.py: 1
    -- <SIP/ext1000-00000004>AGI Script python.py completed, returning 0
    -- Executing [700@ctx-prueba:3] Festival("SIP/ext1000-00000004", "temperatura") in new stack
    -- Executing [700@ctx-prueba:4] SayNumber("SIP/ext1000-00000004", "20") in new stack
    -- <SIP/ext1000-00000004> Playing 'digits/20.slin16' (language 'es')
    -- Executing [700@ctx-prueba:5] Festival("SIP/ext1000-00000004", "humedad") in new stack
    -- Executing [700@ctx-prueba:6] SayNumber("SIP/ext1000-00000004", "57") in new stack
    -- <SIP/ext1000-00000004> Playing 'digits/50.slin16' (language 'es')
    -- <SIP/ext1000-00000004> Playing 'digits/and.slin16' (language 'es')
    -- <SIP/ext1000-00000004> Playing 'digits/7.slin16' (language 'es')
       > 0x7fc14c036d10 -- Strict RTP learning complete - Locking on source address 192.168.1.142:40
06
    -- Executing [700@ctx-prueba:7] Festival("SIP/ext1000-00000004", "luz") in new stack
    -- Executing [700@ctx-prueba:8] SayNumber("SIP/ext1000-00000004", "80") in new stack
    -- <SIP/ext1000-00000004> Playing 'digits/80.slin16' (language 'es')
    -- Executing [700@ctx-prueba:9] Festival("SIP/ext1000-00000004", "sonido") in new stack
    -- Executing [700@ctx-prueba:10] SayNumber("SIP/ext1000-00000004", "34") in new stack
    -- <SIP/ext1000-00000004> Playing 'digits/30.slin16' (language 'es')
    -- <SIP/ext1000-00000004> Playing 'digits/and.slin16' (language 'es')
    -- <SIP/ext1000-00000004> Playing 'digits/4.slin16' (language 'es')
    -- Executing [700@ctx-prueba:11] Festival("SIP/ext1000-00000004", "movimiento") in new stack
    -- Executing [700@ctx-prueba:12] SayNumber("SIP/ext1000-00000004", "1") in new stack
    -- <SIP/ext1000-00000004> Playing 'digits/1.slin16' (language 'es')
    -- Executing [700@ctx-prueba:13] Hangup("SIP/ext1000-00000004", "") in new stack
  == Spawn extension (ctx-prueba, 700, 13) exited non-zero on 'SIP/ext1000-00000004'
debian*CLI>
```

Links a videos explicativos de la parte avanzada

1. Interconexión de PBX
2. Call Center
3. API Rest
4. IoT
5. MySQL