



Presentación

Nombre del estudiante:
Jamiel Santana.

Nombre del profesor:

Tema:
Instalación de Asterisk.

Materia:
Fundamentos de Asterisk.

Centro educativo:
Instituto tecnológico de las américas.

➤ Instalación de las dependencias:

- Lo primero que tenemos que hacer es asegurarnos de tener el SO con todos los paquetes actualizados, esto lo haremos con el comando **yum update -y**.

```
[jamiel@localhost ~]$ sudo yum update
Complementos cargados:fastestmirror, langpacks, yumnotify
Loading mirror speeds from cached hostfile
epel/x86_64/metalink
* base: centos.brisanet.com.br
* epel: mirrors.kernel.org
* extras: centos.itsbrasil.net
* remi-php73: fr2.rpmfind.net
* remi-php74: fr2.rpmfind.net
* remi-safe: fr2.rpmfind.net
* updates: centos.brisanet.com.br
base
epel
extras
mariadb
remi-php73
remi-php74
remi-safe
systemsmangement_Uyuni_Master_CentOS7-Uyuni-Client-Tools
updates
(1/9): epel/x86_64/group_gz
(2/9): remi-php73/primary_db
(3/9): remi-php74/primary_db
(4/9): systemsmangement_Uyuni_Master_CentOS7-Uyuni-Client-Tools/primary
```

- Continuamos con las demás instalaciones de las dependencias, ahora las dependencias de EPEL, esto se hace con el comando **yum -y install epel-release**

```
[jamiel@localhost ~]$ sudo yum -y install epel-release
[sudo] password for jamiel:
Complementos cargados:fastestmirror, langpacks, yumnotify
Loading mirror speeds from cached hostfile
* base: centos.brisanet.com.br
* epel: mirror.us-midwest-1.nexcess.net
* extras: centos.itsbrasil.net
```

- Para no tener problemas con Selinux el cual es el firewall de CentOS, lo desactivaremos, simplemente tendremos que acceder al fichero **/etc/selinux/config** y editamos la línea Selinux y la ponemos en disabled.

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of three values:
#   targeted - Targeted processes are protected,
#   minimum - Modification of targeted policy. Only selected processes are protected.
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

- Lo siguiente es asegurarnos que nuestra zona horaria se encuentra en el mismo lugar del servidor, para confirmar esto haremos uso del comando **timedatectl**.

```
[root@localhost jamiel]# timedatectl
Local time: dom 2021-02-14 11:17:43 AST
Universal time: dom 2021-02-14 15:17:43 UTC
RTC time: dom 2021-02-14 11:17:43
Time zone: America/Santo_Domingo (AST, -0400)
NTP enabled: no
NTP synchronized: no
RTC in local TZ: no
DST active: n/a
```

- Como paso siguiente instalaremos NTP en nuestro servidor.

```
[root@localhost jamiel]# yum install -y ntp
Complementos cargados:fastestmirror, langpacks, yumnotify
Loading mirror speeds from cached hostfile
epel/x86_64/metalink
* base: centos.brisanet.com.br
* epel: mirror.atl.genesisadaptive.com
* extras: centos.itsbrasil.net
* remi-php73: fr2.rpmfind.net
* remi-php74: fr2.rpmfind.net
* remi-safe: fr2.rpmfind.net
* updates: centos.brisanet.com.br
base
```

- Ahora iniciaremos el servicio para confirmar que funciona correctamente, con el comando **systemctl start ntp** y con el comando **systemctl enable ntpd**, para que siempre inicie con el sistema.

```
[root@localhost jamiel]# systemctl enable ntpd
Created symlink from /etc/systemd/system/multi-user.target.wants/ntpd.service to /usr/lib/systemd/system/ntpd.service.
[root@localhost jamiel]# systemctl start ntpd
[root@localhost jamiel]#
```

- Usamos el comando **systemctl status ntpd**, para poder ver el estado del servicio.

```
[root@localhost jamiel]# systemctl status ntpd
● ntpd.service - Network Time Service
   Loaded: loaded (/usr/lib/systemd/system/ntpd.service; enabled; vendor preset: disabled)
   Active: active (running) since dom 2021-02-14 11:22:49 AST; 5min ago
   Process: 10311 ExecStart=/usr/sbin/ntpd -u ntp:ntp $OPTIONS (code=exited, status=0/SUCCESS)
   Main PID: 10312 (ntpd)
   Tasks: 1
   CGroup: /system.slice/ntpd.service
           └─10312 /usr/sbin/ntpd -u ntp:ntp -g

feb 14 11:22:49 localhost.localdomain ntpd[10312]: Listen normally on 4 virbr0 192.168.122.1 UDP 123
feb 14 11:22:49 localhost.localdomain ntpd[10312]: Listen normally on 5 lo ::1 UDP 123
feb 14 11:22:49 localhost.localdomain ntpd[10312]: Listen normally on 6 ens33 fe80::32e9:9483:b9f2:8e76 UDP 123
feb 14 11:22:49 localhost.localdomain ntpd[10312]: Listening on routing socket on fd #23 for interface updates
feb 14 11:22:49 localhost.localdomain ntpd[10312]: 0.0.0.0 c016 06 restart
feb 14 11:22:49 localhost.localdomain ntpd[10312]: 0.0.0.0 c012 02 freq_set kernel 0.000 PPM
feb 14 11:22:49 localhost.localdomain ntpd[10312]: 0.0.0.0 c011 01 freq_not_set
feb 14 11:22:56 localhost.localdomain ntpd[10312]: 0.0.0.0 c61c 0c clock_step +0.841745 s
feb 14 11:22:57 localhost.localdomain ntpd[10312]: 0.0.0.0 c614 04 freq_mode
feb 14 11:22:58 localhost.localdomain ntpd[10312]: 0.0.0.0 c618 08 no_sys_peer
```

- El hostname lo cambiaremos a pbx.jamiel.com

```
GNU nano 2.3.1                                Fichero: /etc/hosts
27.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1        localhost localhost.localdomain localhost6 localhost6.localdomain6
127.0.0.1    pbx.jamiel.com
10.0.0.14    pbx.jamiel.com
```

- Instalaremos más dependencias como se muestran en la imagen.

```
login as: jamiel
jamiel@10.0.0.14's password:
Last login: Sun Feb 14 14:01:41 2021
[jamiel@pbx ~]$ sudo yum -y install wget vim net-tools
[sudo] password for jamiel:
```

```
[jamiel@pbx ~]$ sudo yum -y groupinstall "Development Tools"
Complementos cargados:fastestmirror, langpacks, yumnotify
No existe un archivo de grupos instalados.
Maybe run: yum groups mark convert (see man yum)
```

```
[jamiel@pbx ~]$ sudo yum -y install libedit-devel sqlite-devel psmisc gmime-devel ncurses-devel libtermcap-devel sox newt-devel libxml2-devel libtiff-devel audiofile-devel
gtk2-devel uuid-devel libtool libuuid-devel subversion kernel-devel kernel-devel-$(uname -r) git subversion kernel-devel crontabs cronie cronie-anacron wget vim
Complementos cargados:fastestmirror, langpacks, yumnotify
Loading mirror speeds from cached hostfile
* base: centos.brisanet.com.br
```

- Ya instaladas estas dependencias vamos a instalar PJSIP:
- Jansson es una biblioteca de C para codificar, decodificar y manipular datos JSON. Descárguelo e instálelo en el servidor CentOS 7 ejecutando los siguientes comandos:

```
[jamiel@pbx src]$ cd /usr/src/
[jamiel@pbx src]$ sudo git clone https://github.com/akheron/jansson.git
[sudo] password for jamiel:
Cloning into 'jansson'...
remote: Enumerating objects: 24, done.
remote: Counting objects: 100% (24/24), done.
remote: Compressing objects: 100% (16/16), done.
remote: Total 6492 (delta 6), reused 14 (delta 4), pack-reused 6468
Receiving objects: 100% (6492/6492), 1.70 MiB | 1014.00 KiB/s, done.
Resolving deltas: 100% (3894/3894), done.
[jamiel@pbx src]$
[jamiel@pbx src]$ cd jansson/
[jamiel@pbx jansson]$ sudo autoreconf -i
libtoolize: putting auxiliary files in AC_CONFIG_AUX_DIR, `.'.
libtoolize: copying file `./ltmain.sh'
libtoolize: Consider adding `AC_CONFIG_MACRO_DIR([m4])' to configure.ac and
libtoolize: rerunning libtoolize to keep the correct libtool macros in-tree
[jamiel@pbx jansson]$ clear
[jamiel@pbx jansson]$ sudo ./configure --prefix=/usr/
checking for a BSD-compatible install... /bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /bin/mkdir -p
checking for gawk... gawk
[jamiel@pbx jansson]$ make && make install
```

➤ Descargando e instalando PJSIP:

- PJSIP es una biblioteca de comunicación multimedia gratuita y de código abierto escrita en lenguaje C que implementa protocolos basados en estándares como SIP, SDP, RTP, STUN, TURN e ICE. Clonar el proyecto desde Github y, a continuación, compilar e instalar.

```
[jamiel@pbx jansson]$ cd /usr/src/
[jamiel@pbx src]$ export VER="2.10"
[jamiel@pbx src]$ wget https://github.com/pjsip/pjproject/archive/${VER}.tar.gz
--2021-02-14 16:29:25-- https://github.com/pjsip/pjproject/archive/2.10.tar.gz
Resolviendo github.com (github.com)... 140.82.114.4
[jamiel@pbx src]$ ls
2.10.tar.gz debug jansson kernels pjproject-2.10
[jamiel@pbx src]$ tar -xvf ${VER}.tar.gz

[jamiel@pbx pjproject-2.10]$ cd pjproject-${VER}
-bash: cd: pjproject-2.10: No existe el fichero o el directorio
[jamiel@pbx pjproject-2.10]$ cd ..
[jamiel@pbx src]$ clear
[jamiel@pbx src]$ cd pjproject-${VER}
[jamiel@pbx pjproject-2.10]$ sudo ./configure CFLAGS="-DDEBUG -DPJ_HAS_IPV6=1" --prefix=/usr --libdir=/usr/lib64 --enable-shared --disable-video --disable-sound --disable-opencore-amr
checking build system type... x86_64-unknown-linux-gnu
checking host system type... x86_64-unknown-linux-gnu
[jamiel@pbx pjproject-2.10]$ make dep
for dir in pjl原因/build pjl原因/build pj原因/build third_party/build pjmedia/build pjsip/build pjsip-apps/build ; do \
    if make -C $dir dep; then \
        true; \
    else \
        exit 1; \
    fi; \
done
make[1]: se ingresa al directorio `/usr/src/pjproject-2.10/pjlib/build'
touch ../include/pj/config_site.h
touch: no se puede efectuar `touch' sobre «../include/pj/config_site.h»: Permiso denegado
make[1]: *** [../include/pj/config_site.h] Error 1
make[1]: se sale del directorio `/usr/src/pjproject-2.10/pjlib/build'

[root@pbx pjproject-2.10]# make install
mkdir -p /usr/lib64/
cp -af /usr/src/pjproject-2.10/pjsip/lib/libpjsua2-x86_64-unknown-linux-gnu.a /usr/src/pjproject-2.10/pjsip/lib/libpjsip-ua-x86_64-unknown-linux-gnu.a /usr/src/pjproject-2.10/pjmedia/lib/libpjmedia-ua-x86_64-unknown-linux-gnu.a /usr/src/pjproject-2.10/pjmedia/lib/libpjmedia-ua-x86_64-unknown-linux-gnu.a

[root@pbx pjproject-2.10]# ldconfig
```

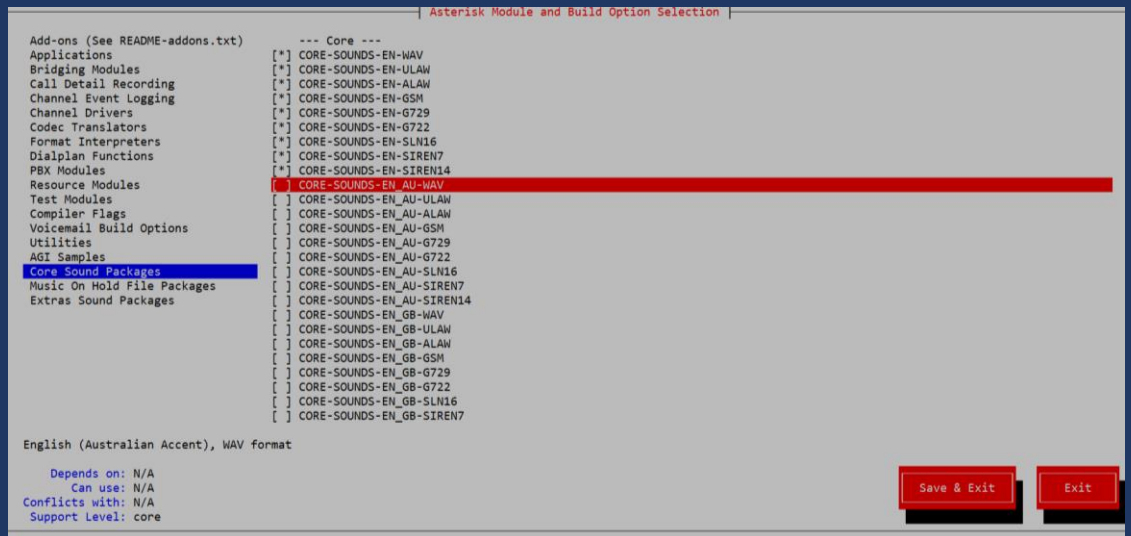
➤ Descargando e instalando Asterisk:

- Ahora que tenemos todos los paquetes de dependencia instalados, debemos estar listos para descargar e instalar Asterisk 16 en CentOS 7.

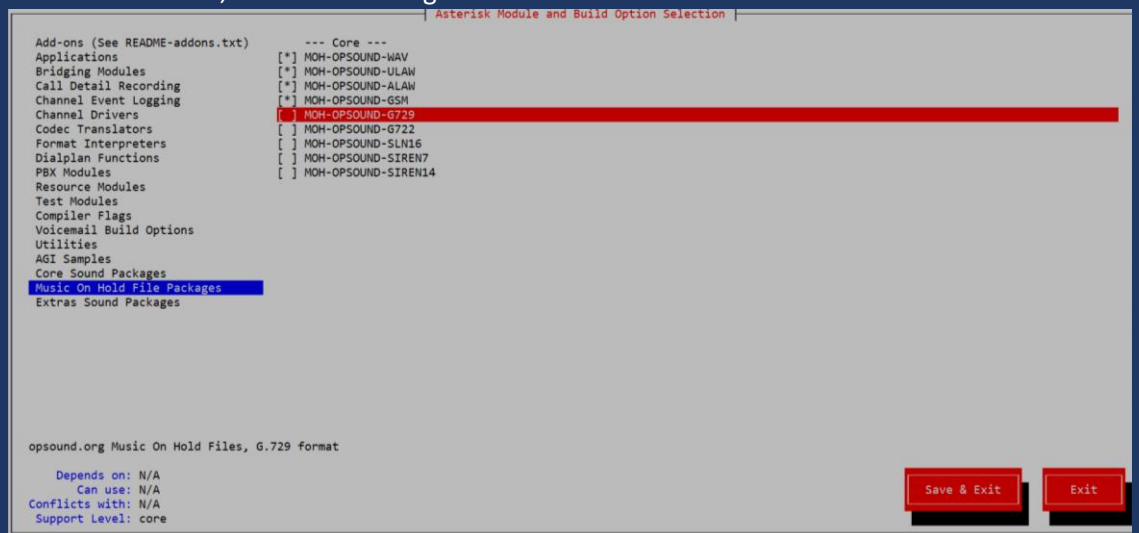
```
[root@pbx pjproject-2.10]# cd /usr/src/
[root@pbx src]# wget http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-16-current.tar.gz
--2021-02-14 17:04:04-- http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-16-current.tar.gz
Resolviendo downloads.asterisk.org (downloads.asterisk.org)... 76.164.171.238, 2001:470:e0d4::ee
Conectando con downloads.asterisk.org (downloads.asterisk.org)[76.164.171.238]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 27810532 (27M) [application/x-gzip]
Grabando a: "asterisk-16-current.tar.gz"

15% [=====
```

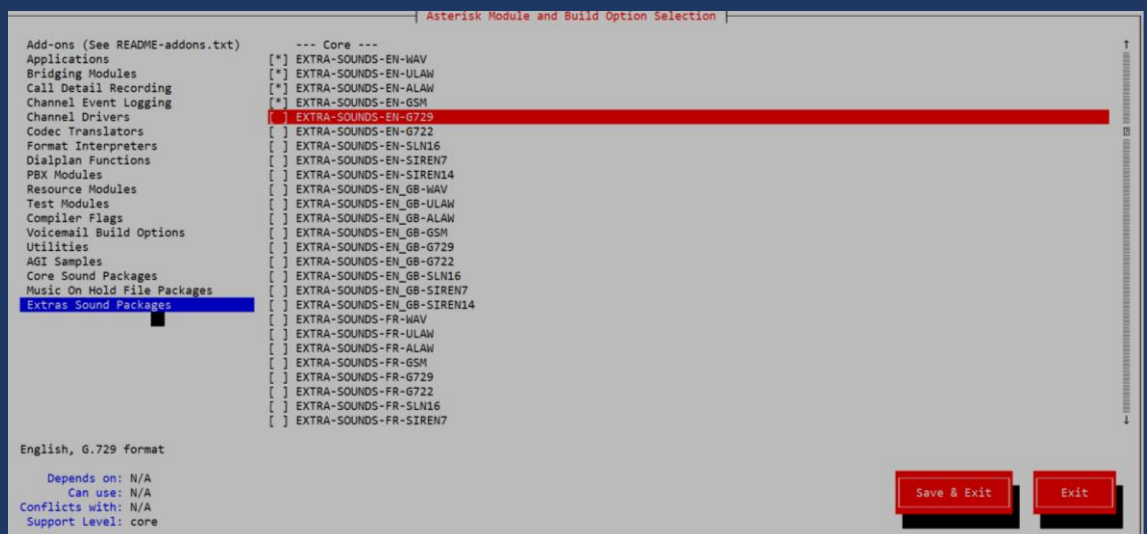
- En Core Sound Packages, seleccione los formatos de los paquetes de audio como se muestra a continuación.



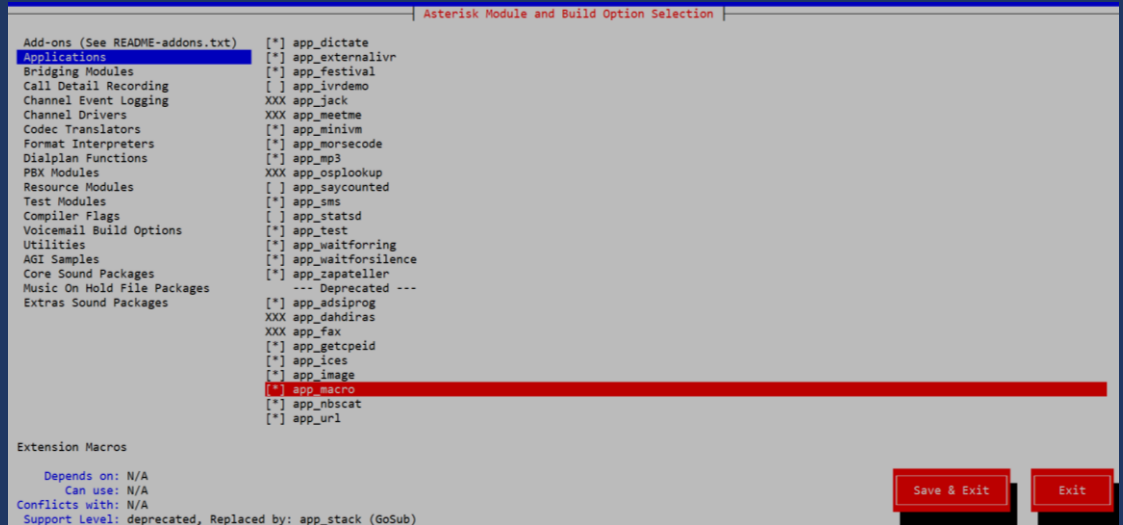
- En Music On Hold, seleccione los siguientes módulos mínimos:



- En paquetes de sonido adicionales seleccione como se muestra a continuación:



- Ahora en la sección de aplicaciones habilitamos el módulo de app_macro:



- Puede cambiar otras configuraciones que considere adecuadas. Cuando haya terminado, guarde y salga.
- Ahora ejecute el siguiente comando para descargar la biblioteca de decodificadores mp3 en el árbol de origen.

```
[root@pbx asterisk-16.16.0]# sudo contrib/scripts/get_mp3_source.sh
A addons/mp3
A addons/mp3/MPGLIB_README
A addons/mp3/common.c
A addons/mp3/huffman.h
A addons/mp3/tabinic.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.
[root@pbx asterisk-16.16.0]#
```

- Para iniciar la creación e instalación de Asterisk con módulos seleccionados, ejecute los comandos.

```
[root@pbx asterisk-16.16.0]# make
[CC] astcanary.c -> astcanary.o
[LD] astcanary.o -> astcanary
[CC] astdb2sqlite3.c -> astdb2sqlite3.o
[CC] hash/hash.c -> hash/hash.o

[root@pbx asterisk-16.16.0]# make install
Installing modules from channels...
Installing modules from pbx...
Installing modules from apps...

[root@pbx asterisk-16.16.0]# make samples
Installing adsi config files...
/bin/install -c -d "/etc/asterisk"
Installing configs/samples/asterisk.adsi
Installing configs/samples/telcordia-1.adsi
Installing other config files...
Installing file configs/samples/acl.conf.sample
```



```
[root@pbx asterisk-16.16.0]# make config
```

```
[root@pbx asterisk-16.16.0]# ldconfig
```

```
[root@pbx asterisk-16.16.0]#
```

- Cree un usuario y un grupo independientes para ejecutar servicios de asterisco y asigne los permisos correctos:

```
[root@pbx asterisk-16.16.0]# groupadd asterisk  
[root@pbx asterisk-16.16.0]#
```

- Establezca El usuario predeterminado de Asterisco en asterisco:

```
[root@pbx asterisk-16.16.0]# sudo vim /etc/sysconfig/asterisk  
# Startup configuration for the Asterisk daemon  
  
AST_USER="asterisk"  
AST_GROUP="asterisk"  
# Uncomment the following and set them to the user/groups that you  
# want to run Asterisk as. NOTE: this requires substantial work to  
# be sure that Asterisk's environment has permission to write the
```

- Establezca El usuario predeterminado de Asterisco en asterisco:

```
runuser = asterisk ; The user to run as.  
rungroup = asterisk ; The group to run as.  
[options]
```

- Reinicie el servicio de asterisco después de realizar los cambios:

```
[root@pbx asterisk-16.16.0]# sudo systemctl restart asterisk
```

- Habilite el servicio para iniciarse en el arranque:

```
[root@pbx asterisk-16.16.0]# sudo systemctl enable asterisk
```

- Pruebe para ver si usted puede conectar con el asterisco CLI:
- Tendremos que ver una salida de la siguiente forma.

```
[root@pbx asterisk-16.16.0]# asterisk -rvv  
Asterisk 16.16.0, Copyright (C) 1999 - 2018, Digium, Inc. and others.  
Created by Mark Spencer <markster@digium.com>  
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.  
This is free software, with components licensed under the GNU General Public  
License version 2 and other licenses; you are welcome to redistribute it under  
certain conditions. Type 'core show license' for details.  
=====
```

```
Connected to Asterisk 16.16.0 currently running on pbx (pid = 57810)  
pbx*CLI>
```

➤ Instalación de Asterisk en Ubuntu:

- Lo primero es actualizar todos los paquetes necesarios eso lo hacemos con el comando.

```
root@Asterisk:/home/jamiel# apt-get update  
Get:1 http://security.ubuntu.com/ubuntu groovy-security InRelease [110 kB]  
Hit:2 http://do.archive.ubuntu.com/ubuntu groovy InRelease  
Get:3 http://do.archive.ubuntu.com/ubuntu groovy-updates InRelease [115 kB]  
Get:4 http://do.archive.ubuntu.com/ubuntu groovy-backports InRelease [101 kB]  
Get:5 http://security.ubuntu.com/ubuntu groovy-security/main amd64 Packages [221 kB]  
Get:6 http://do.archive.ubuntu.com/ubuntu groovy-updates/main amd64 Packages [366 kB]  
Get:7 http://do.archive.ubuntu.com/ubuntu groovy-updates/main i386 Packages [164 kB]  
Get:8 http://do.archive.ubuntu.com/ubuntu groovy-updates/main Translation-en [94.1 kB]  
Get:9 http://security.ubuntu.com/ubuntu groovy-security/main amd64 DEP-11 Metadata [4,676 B]  
Get:10 http://security.ubuntu.com/ubuntu groovy-security/main amd64 c-n-f Metadata [3,388 B]
```

- Ahora crearemos y accederos a la carpeta que le pondremos por nombre Asterisk
- Creado ya los directorios de trabajo vamos a clonar el repositorio Asterisk. Esto lo haremos con el comando.

- Automáticamente se nos descarga un archivo tar el cual descomprimiremos con el comando que se muestra en la imagen.

```
root@Asterisk:/home/jamiel/Downloads/Asterisk# ls
asterisk-16-current.tar.gz
root@Asterisk:/home/jamiel/Downloads/Asterisk# tar -zxvf asterisk-16-current.tar.gz
asterisk-16.16.2/
asterisk-16.16.2/.cleancount
asterisk-16.16.2/.gitignore
asterisk-16.16.2/.gitreview
asterisk-16.16.2/.lastclean
asterisk-16.16.2/.version
```

- Ya descomprimido este archivo nos dirigimos a la ruta `/usr/src` para descargar ciertos repositorios que se mostrarán en la imagen.

```
root@Asterisk:/usr/src# git clone git://git.asterisk.org/dahdi-linux
Cloning into 'dahdi-linux'...
remote: Counting objects: 9853, done.
remote: Compressing objects: 100% (2145/2145), done.
Receiving objects: 34% (3351/9853), 596.00 KiB | 287.00 KiB/s
root@Asterisk:/usr/src# ls
dahdi-linux  linux-headers-5.8.0-25  linux-headers-5.8.0-25-generic  linux-headers-5.8.0-44  linux-headers-5.8.0-44-generic
root@Asterisk:/usr/src# git clone git://git.asterisk.org/dahdi/tools dahdi-tools
Cloning into 'dahdi-tools'...
remote: Counting objects: 2161, done.
remote: Compressing objects: 100% (764/764), done.
Receiving objects: 62% (1340/2161), 488.00 KiB | 960.00 KiB/s
root@Asterisk:/usr/src# git clone http://gerrit.asterisk.org/libpri libpri
Cloning into 'libpri'...
warning: redirecting to https://gerrit.asterisk.org/libpri/
remote: Counting objects: 549, done
Receiving objects: 47% (1371/2916)
```

- El siguiente paso seria instalar y compilar el paquete dahdi-linux.

```

root@Asterisk:/usr/src# cd dahdi-linux/
root@Asterisk:/usr/src/dahdi-linux# make
make -C drivers/dahdi/firmware firmware-loaders
make[1]: Entering directory '/usr/src/dahdi-linux/drivers/dahdi/firmware'
Attempting to download dahdi-fwload-vpmadt032-1.25.0.tar.gz
--2021-03-07 08:23:45-- http://downloads.digium.com/pub/telephony/firmware/releases/dahdi-fwload-vpmadt032-1.25.0.tar.gz
Resolving downloads.digium.com (downloads.digium.com)... 76.164.171.238, 2001:470:e004::ee
Connecting to downloads.digium.com (downloads.digium.com)[76.164.171.238]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 149360 (146K) [application/x-gzip]
Saving to: 'dahdi-fwload-vpmadt032-1.25.0.tar.gz'

dahdi-fwload-vpmadt032-1.25.0.tar.gz      100%[=====] 145.86K   759KB/s   in 0.2s

2021-03-07 08:23:45 (759 KB/s) - 'dahdi-fwload-vpmadt032-1.25.0.tar.gz' saved [149360/149360]

make[1]: Leaving directory '/usr/src/dahdi-linux/drivers/dahdi/firmware'
make -C /lib/modules/5.8.0-44-generic/build KBUILD_EXTMOD=/usr/src/dahdi-linux/drivers/dahdi DAHDI_INCLUDE=/usr/src/dahdi-linux/include DAHDI_MODULES_EXTRA=" " HOTPLUG_FIRM
WAREyes modules DAHDI_BUILD_ALL=m
make[1]: Entering directory '/usr/src/linux-headers-5.8.0-44-generic'

root@Asterisk:/usr/src/dahdi-linux# make install
make -C drivers/dahdi/firmware firmware-loaders
make[1]: Entering directory '/usr/src/dahdi-linux/drivers/dahdi/firmware'
make[1]: Leaving directory '/usr/src/dahdi-linux/drivers/dahdi/firmware'
make -C /lib/modules/5.8.0-44-generic/build KBUILD_EXTMOD=/usr/src/dahdi-linux/drivers/dahdi DAHDI_INCLUDE=/usr/src/dahdi-linux/include DAHDI_MODULES_EXTRA=" " HOTPLUG_FIRM
WAREyes modules DAHDI_BUILD_ALL=m
make[1]: Entering directory '/usr/src/linux-headers-5.8.0-44-generic'

```

- Ahora copilaremos el paquete de dahdi-tools.

```

root@Asterisk:/usr/src/dahdi-tools# autoreconf -i
libtoolize: putting auxiliary files in AC_CONFIG_AUX_DIR, 'auxdir'.
libtoolize: copying file 'auxdir/ltmain.sh'
libtoolize: putting macros in AC_CONFIG_MACRO_DIRS, 'm4'.
libtoolize: copying file 'm4/libtool.m4'
libtoolize: copying file 'm4/ltoptions.m4'
libtoolize: copying file 'm4/ltugar.m4'
libtoolize: copying file 'm4/ltversion.m4'
libtoolize: copying file 'm4/ltobsolete.m4'

root@Asterisk:/usr/src/dahdi-tools# ./configure
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
checking for gawk... no
checking for mawk... mawk
checking whether make sets $(MAKE)... yes
checking whether make supports nested variables... yes
checking whether make supports nested variables... (cached) yes
checking whether make supports the include directive... yes (GNU style)
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89...

```

- Después de esto vamos con el paquete de libpri, realizado un make y un make install dentro del directorio.
- Ahora ejecutaremos la instalación de Asterisk para eso nos dirigimos a la siguiente ruta `cd /downloads/asterisk/asterisk-16.6.0/contrib/scripts/` y ejecutamos el siguiente comando

```

root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2/contrib/scripts# ./install_prereq install
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  antitilde-common libuidate4 libvarian20

```

se nos mostrará esta pantalla durante la instalación solo daremos click enter y listo, la instalación continuará

```

This is the numeric code for the region your phone system will be operating in (eg. 61 for Australia or 33 for France). It is used to configure the default
regional standards that Voicetronix telephony hardware should comply with.

ITU-T telephone code:
61

<Ok>

```

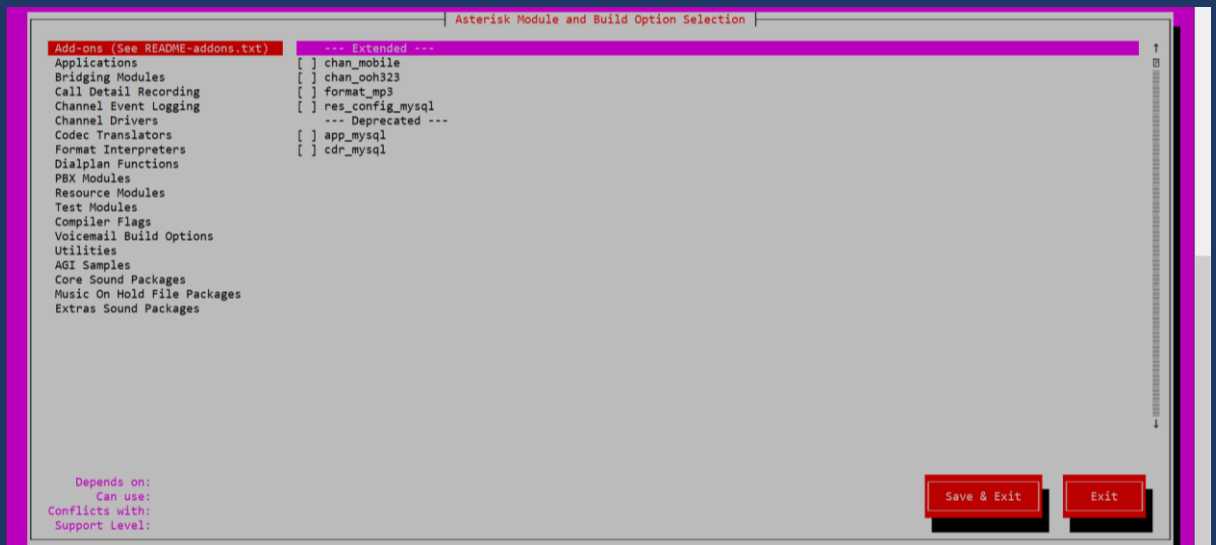
- Si la instalación se realiza correctamente veremos una salida de la siguiente forma

```
#####
## install completed successfully
#####
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2/contrib/scripts#
```

- Ya instalado vamos a compilar Asterisk.

```
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# ./configure --with-pjproject-bundled
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
```

```
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# make menuselect
CC="cc" CXX="g++" LD="" AR="" RANLIB="" CFLAGS="" LDFLAGS="" make -C menuselect CONFIGURE_SILENT="--silent" makeopts
```



```
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# make
CC="cc" CXX="g++" LD="" AR="" RANLIB="" CFLAGS="" LDFLAGS="" make -C menuselect CONFIGURE_SILENT="--silent" makeopts
make[1]: Entering directory '/home/jamiel/Downloads/Asterisk/asterisk-16.16.2/menuselect'
make[1]: 'makeopts' is up to date.
make[1]: Leaving directory '/home/jamiel/Downloads/Asterisk/asterisk-16.16.2/menuselect'
[CC] astcanary.c -> astcanary.o
[LD] astcanary.o -> astcanary
[CC] astdb2sqlite3.c -> astdb2sqlite3.o
[CC] hash/hash.c -> hash/hash.o
[CC] hash/hash_bigkey.c -> hash/hash_bigkey.o
```

```
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# make install
CC="cc" CXX="g++" LD="" AR="" RANLIB="" CFLAGS="" LDFLAGS="" make -C menuselect CONFIGURE_SILENT="--silent" makeopts
make[1]: Entering directory '/home/jamiel/Downloads/Asterisk/asterisk-16.16.2/menuselect'
make[1]: 'makeopts' is up to date.
make[1]: Leaving directory '/home/jamiel/Downloads/Asterisk/asterisk-16.16.2/menuselect'
```

```
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# make samples
Installing adsi config files...
/usr/bin/install -c -d "/etc/asterisk"
```

```
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# make config
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2#
```

```
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# make config
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# make install-logrotate
if [ ! -d "/etc/asterisk/./logrotate.d" ]; then \
    /usr/bin/install -c -d "/etc/asterisk/./logrotate.d" ; \
fi
sed 's#_LOGDIR_#/var/log/asterisk#g' < contrib/scripts/asterisk.logrotate | sed 's#_SBINDIR_#/usr/sbin#g' > contrib/scripts/asterisk.logrotate.tmp
/usr/bin/install -c -m 0644 contrib/scripts/asterisk.logrotate.tmp "/etc/asterisk/./logrotate.d/asterisk"
rm -f contrib/scripts/asterisk.logrotate.tmp
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2#
```

- Realizado todo lo anterior iniciamos el servicio de Asterisk como se ve en la imagen

```
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# service asterisk start
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2#
```

- Verificamos si el servicio está corriendo con normalidad con el siguiente comando

```
root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# service asterisk status
● asterisk.service - LSB: Asterisk PBX
   Loaded: loaded (/etc/init.d/asterisk; generated)
   Active: active (running) since Sun 2021-03-07 09:54:52 AST; 1min 26s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 54298 ExecStart=/etc/init.d/asterisk start (code=exited, status=0/SUCCESS)
    Tasks: 68 (limit: 4615)
   Memory: 40.5M
    CGroup: /system.slice/asterisk.service
            └─54311 /usr/sbin/asterisk

root@Asterisk:/home/jamiel/Downloads/Asterisk/asterisk-16.16.2# netstat -tulpn | grep asterisk
tcp        0      0 0.0.0.0:2000        0.0.0.0:*           LISTEN     54311/asterisk
udp        0      0 0.0.0.0:4520      0.0.0.0:*           54311/asterisk
udp        0      0 0.0.0.0:4569      0.0.0.0:*           54311/asterisk
udp        0      0 0.0.0.0:2727      0.0.0.0:*           54311/asterisk
udp        0      0 0.0.0.0:5000      0.0.0.0:*           54311/asterisk
udp        0      0 0.0.0.0:5060      0.0.0.0:*           54311/asterisk
udp        0      0 0.0.0.0:50867     0.0.0.0:*           54311/asterisk
udp6       0      0 :::47853          :::*                 54311/asterisk
```