

5 практическая работа

Установка Asterisk и добавление клиентов.

1 Установка Asterisk.

1.1 Подготовка среды для развертывания программной АТС Asterisk.

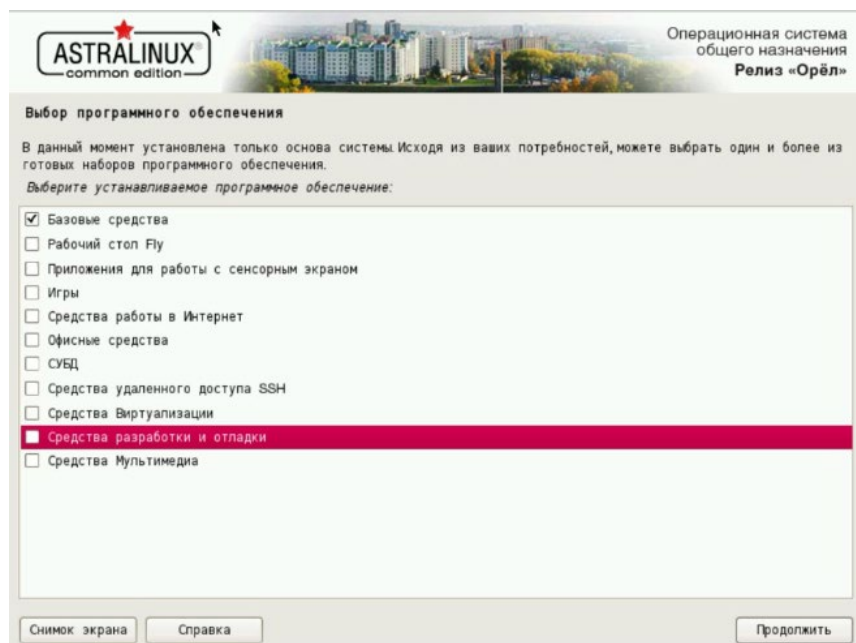
Подготовьте в среде виртуализацию VirtualBox виртуальную машину, предоставляющую следующие системные требования:

- архитектурой x64;
- оперативная память — не менее 3 ГБ;
- объем свободного дискового пространства — не менее 16 ГБ.
- Тип сетевого подключения - Мост

И установите на подготовленную виртуальную машину операционную систему Astra Linux Common Edition (актуальную версию (**alce-current.iso**) ОС можно скачать по ссылке <https://download.astralinux.ru/astra/stable/orel/iso/>) или (<https://easyastra.ru/resources/astralinux.php?ysclid=muecsxv4lt278541040>). **Во время установки снимите все галочки при выборе программного обеспечения (кроме «Базовые средства»).** В конце установите GRUB правильно выбрав раздел для установки.

Index of /astra/stable/orel/iso/

File Name <u>↓</u>	File Size <u>↓</u>
Parent directory/	—
alce-2.12.46.6-17.04.2023_15.09.iso	5625704448
alce-2.12.46.6-17.04.2023_15.09.iso.gost	102
alce-2.12.46.6-17.04.2023_15.09.iso.md5	70
alce-current.iso	5625704448
alce-current.iso.gost	102
alce-current.iso.md5	70
orel-current.iso	5625704448
orel-stable.iso	5625704448



Установка системного загрузчика GRUB на жёсткий диск

Пожалуйста, убедитесь, что данная система будет единственной на этом компьютере. Если это действительно так, то можно спокойно устанавливать системный загрузчик GRUB в основную загрузочную запись первого жёсткого диска.

Внимание! Если программе установки не удалось обнаружить другую операционную систему, имеющуюся на компьютере, то изменение основной загрузочной записи приведёт к тому, что эту операционную систему некоторое время нельзя будет загрузить. Позднее можно будет настроить GRUB для её загрузки.

Установить системный загрузчик GRUB в главную загрузочную запись?

☐ Нет

☒ Да

Снимок экрана

Справка

Вернуться

Продолжить

Установка системного загрузчика GRUB на жёсткий диск

Пришло время научить только что установленную систему загружаться. Для этого на загрузочное устройство будет установлен системный загрузчик GRUB. Обычно он устанавливается в главную загрузочную запись (MBR, Master Boot Record) первого жёсткого диска. При желании можно установить GRUB в любое другое место на диске, либо на другой диск, либо вообще на дискету.

Устройство для установки системного загрузчика:

Указать устройство вручную

/dev/sda (ata-VBOX_HARDDISK_VB0fec03a3-35a16c7a)

Снимок экрана

Справка

Вернуться

Продолжить

1.2 Подготовка системы (Дальнейшее выполнение практической работы происходит в гостевой ОС (Astra Linux) установленной на виртуальной машине).

Отключаем SeLinux и Firewalld (может выдавать ошибку, значит данные пакеты не установлены, просто пропускаем данный пункт):



В консоли Linux:

```
sudo systemctl disable ufw
sudo systemctl stop ufw
```

Команда отключает автозагрузку firewalld

Подготавливаем папку для работы:



В консоли Linux:

```
sudo mkdir -p /usr/src/asterisk
cd /usr/src/asterisk
```

Создаем папку, в которой будем работать с исходными файлами Asterisk `mkdir` – создать папку [опция `-p` – задает создание всех несуществующих вышележащих каталогов]
`cd` – переход в указанный каталог

Для того чтобы виртуальная машина получала ip адрес, отредактируйте файл настройки сети `/etc/network/interfaces`:



В консоли Linux:

```
sudo nano /etc/network/interfaces
```

Внесите в содержимое файла:

```
allow-hotplug eth0 #автоматически выполнять перезапуск интерфейса при его падении
auto eth0           #поднимать интерфейс автоматически при старте системы
iface eth0 inet dhcp #получить адрес по DHCP
```

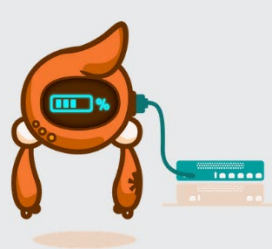
После внесения изменений в файл перезапустите службу:
`sudo systemctl restart networking.service`

1.3 Получение исходников Asterisk (в работе используется 20 версия Asterisk).

Скачайте 20 версию программной АТС Asterisk (скачивание происходит в виртуальной машине) по ссылке и размещаем скаченные файлы в ранее подготовленной директории (`/usr/src/asterisk`): Получить ссылку для скачивания исходных файлов Asterisk можно по ссылке: <https://www.asterisk.org/downloads/>

Download Asterisk

Download the currently supported versions of Asterisk and various Asterisk-related open source projects.



Asterisk Communications Framework

Asterisk is an open source toolkit for building communications applications. For a more detailed explanation, check out the [Get Started](#) section. For more information on how to use Asterisk, see the [Configuration](#) and [Operation](#) sections of the wiki.

[Download Latest LTS Version \(22.1.0\)](#)

[Change Log](#)
[* All Asterisk Versions](#)

Asterisk 20 - LTS



Download Latest - 20.11.0



ChangeLog

При наличии ссылки на архив с приложением и установленной утилитой **wget**, скачать пакет можно из командной строки:



В консоли Linux:

```
sudo wget https://ссылка на пакет.tar.gz
```

Пример: `wget https://downloads.asterisk.org/pub/telephony/asterisk/asterisk-20-current.tar.gz`

wget – утилита для загрузки файлов из Интернета. Файлы с расширением tar.gz – архивы.

Файлы будут загружены именно в ту директорию, в которой выполнена команда wget

Распаковываем все полученные файлы:



В консоли Linux:

```
sudo tar -xvf путь к архиву
```

Пример: `-tar xvf asterisk-20-current.tar.gz`

tar – консольный архиватор, xvf – его параметры (указывают на действия которые произведет архиватор)

-x – извлечь файлы из архива

-v – показывает список обрабатываемых файлов

-f – указывает используемый архивный файл

Данные разархивируются в папку с именем архива без расширения, которая располагается в одной с архивом директории.

1.4 Установка Asterisk.

Устанавливаем зависимости для компиляции пакетов



В консоли Linux:

```
sudo apt update
```

```
sudo apt -y install ssh
```

```
sudo [SRC_PATH]/contrib/scripts/install_prereq install
```

```
sudo [SRC_PATH]/contrib/scripts/get_mp3_source.sh
```

В исходниках Asterisk есть готовый скрипт для установки всех зависимостей, его можно запустить с ключом test – показать все зависимости или install – установить все зависимости

[SRC_PATH] – путь к папке с разархивированными файлами (/usr/src/asterisk/asterisk-*).

Устанавливаем Asterisk:



В консоли Linux:

```
cd /usr/src/asterisk/asterisk-*
```

переход в папку **usr/src/asterisk/asterisk-20**

make clean – запускаем в том случае, если до этого была неудачная попытка компиляции и нужно удалить следы прошлых компиляций

```
sudo ./configure --with-jansson-bundled --libdir=/usr/lib64
```

собирает данные о системе: определяет совместимость с ОС и наличие всех зависимостей, требуемых для сборки бинарных пакетов. В случае вывода ошибки об отсутствии утилиты *gcc*, повторно запустите скрипт ***sudo [SRC_PATH]/contrib/scripts/install_prereq install***

Если увидели подобную заставку, значит с установкой все в порядке:

[illegible]

Выбор модулей Asterisk



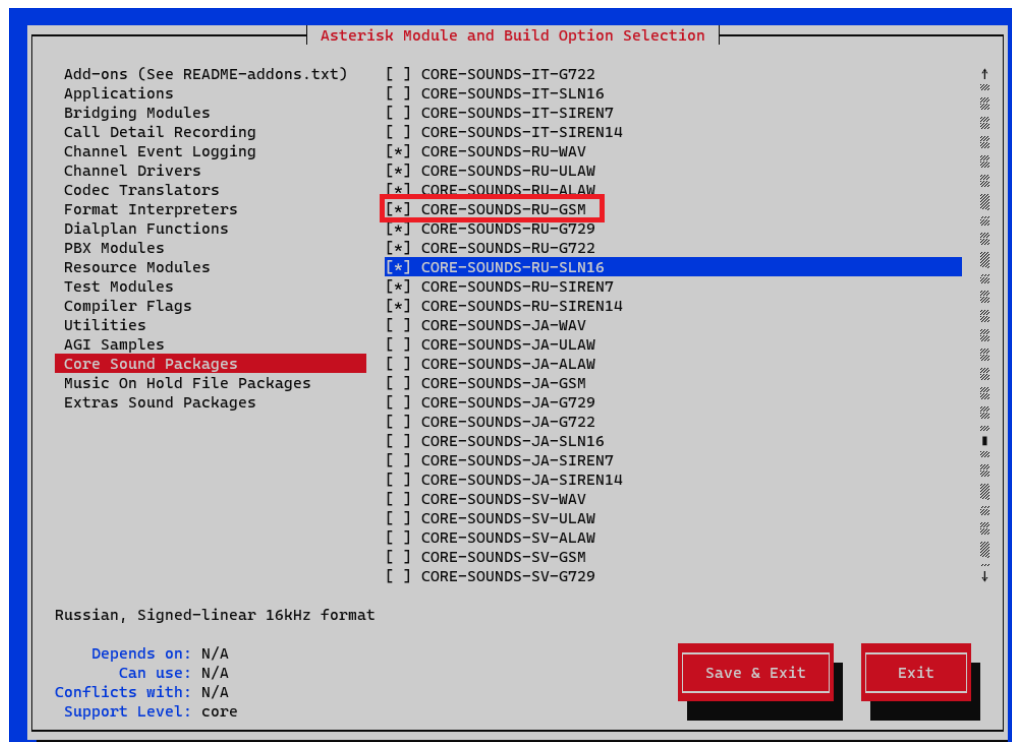
В консоли Linux:

```
sudo make menuselect
```

ВЫЗОВ СИСТЕМЫ ВЫБОРА МОДУЛЕЙ Asterisk

Настраиваем параметры компиляции:

- Выбираем “CoreSoundPackages” и выбираем там русскоязычную озвучку в формате GSM.



В консоли Linux:

```
sudo make
sudo make install
sudo make config
```

make – утилита для автоматического определения частей большой программы, нуждающихся в перекомпилировании, и вызова команд для их перекомпиляции
make install – установка Asterisk в систему
make config – создает скрипты автозапуска



В консоли Linux:

```
sudo touch /etc/asterisk/modules.conf
```

touch – утилита создания файлов

Создаем конфигурационный файл модулей, мы заполним его позже, но

!! Без этого файла Asterisk не запустится !!



```
sudo systemctl enable asterisk
sudo systemctl start asterisk
sudo asterisk -rvvvvv
```

В консоли Linux:

systemctl enable asterisk – включает службу Asterisk в автозагрузку
service asterisk start – запускает Asterisk в фоновом режиме (как службу)
asterisk -rvvvvv – штатное подключение к Asterisk в режиме удаленной консоли
v – задает подробность вывода логов в консоль

1.5 Задание конфигурации:

Следующие конфигурационные файлы создаются в файловой операционной системы (если Вы находитесь в консоле Asterisk, предварительно нужно выйти из нее командой **exit**)

Пример создания или редактирования файла:

sudo nano ИМЯ_ФАЙЛА

CTRL+ o – сохранить, **CTRL+ x** выйти из редактора.

Задание конфигурации для системы загрузки модулей:

Создаем файл */etc/asterisk/modules.conf* с содержимым:

```
[modules]
autoload=yes
```

Задание конфигурации для системы логирования событий:

Создаем файл */etc/asterisk/logger.conf* с содержимым:

```
[general]
[logfiles]
console => notice,warning,error,verbose,dtmf
full => notice,warning,error,debug,verbose,dtmf
```

Задание конфигурации протокола RTP, использующегося для передачи медиа информации:

Создаем файл */etc/asterisk/rtp.conf* с содержимым:

```
[general]
rtpstart=36600
rtpend=39999
```

Задание конфигурации для SIP протокола:

Создаем файл */etc/asterisk/pjsip.conf* с содержимым:

```
[udp-any]
type=transport
protocol=udp
bind=0.0.0.0
```

Задание конфигурации для диалплана Asterisk:

Создаем файл */etc/asterisk/extensions.conf* с содержимым:

```
[general]
[globals]
[from-internal]
exten => _X.,1,Dial(PJSIP/${EXTEN})
```

Создаем файл `/etc/asterisk/musiconhold.conf` с содержанием:

```
[default]
mode=files
directory=/var/lib/asterisk/moh/
```

Отредактируем файл `/etc/ld.so.conf` дописав содержимое:

```
/usr/lib64/
```

Перечитываем список библиотек и делаем рестарт Астериска:

```
sudo ldconfig
sudo systemctl restart asterisk
```

1.6 Устанавливаем модуль `codec_G729` и `codec_G723`



В консоли Linux:

```
sudo cat /proc/cpuinfo
```

(если сомневаемся в разрядности, то набираем `uname -a`)

выводит из файла `cpuinfo` в консоль информацию о процессоре.
`uname -a` выводит информацию о разрядности системы.

```
root@pcl:~# cat /proc/cpuinfo
vendor_id       : GenuineIntel
cpu family      : 6
model           : 37
model name      : Intel(R) Core(TM) i5 CPU           660  @ 3.33GHz
stepping        : 5
cpu MHz         : 1199.000
cache size      : 4096 KB
physical id     : 0
siblings        : 4
cpu cores       : 2
apicid          : 5
initial apicid  : 5
fpu             : yes
fpu_exception   : yes
cpuid level     : 11
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat
                 pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx rdtscp lm constan
                 t tsc arch_perfmon pebs bts rep_good xtopology nonstop tsc aperfmperf pni pclmulqdq
                 dtes64 monitor ds_cpl vmx smx est tm2 ssse3 cx16 xtpr pdcm pcid sse4_1 sse4_2 popc
                 nt aes lahf_lm ida arat dts tpr_shadow vnmi flexpriority ept vpid
bogomips        : 6650.17
clflush size    : 64
cache alignment : 64
address sizes    : 36 bits physical, 48 bits virtual
power management:

[root@pcl ~]#
```

Тип процессора

Наборы поддерживаемых инструкций

```
root@pcl:~  
[root@pcl ~]# uname -a  
Linux pcl 2.6.32-358.el6.x86_64 #1 SMP Fri Feb 22 00:31:26 UTC 2013  
x86_64 x86_64 x86_64 GNU/Linux  
[root@pcl ~]#
```

"x86_64" говорит о том, что у нас 64-х битная система.
Это необходимо учитывать при выборе модуля для Asterisk

Скачать необходимы кодеки можно на сайте: <http://asterisk.hosting.lv/#notes>

- codec_g723-ast200-gcc4-glibc-pentium-m.so
- codec_g723-ast200-gcc4-glibc-pentium.so
- codec_g723-ast200-gcc4-glibc-x86_64-barcelona.so
- codec_g723-ast200-gcc4-glibc-x86_64-core2.so
- codec_g723-ast200-gcc4-glibc-x86_64-core2-sse4.so
- codec_g723-ast200-gcc4-glibc-x86_64-optimon.so
- codec_g723-ast200-gcc4-glibc-x86_64-optimon-sse3.so
- codec_g723-ast200-gcc4-glibc-x86_64-pentium4.so
- codec_g729-ast200-gcc4-glibc-athlon-sse.so
- codec_g729-ast200-gcc4-glibc-atom.so
- codec_g729-ast200-gcc4-glibc-barcelona.so
- codec_g729-ast200-gcc4-glibc-core2.so
- codec_g729-ast200-gcc4-glibc-core2-sse4.so
- codec_g729-ast200-gcc4-glibc-debug.so
- codec_g729-ast200-gcc4-glibc-geode.so
- codec_g729-ast200-gcc4-glibc-optimon.so
- codec_g729-ast200-gcc4-glibc-optimon-sse3.so
- codec_g729-ast200-gcc4-glibc-pentium2.so
- codec_g729-ast200-gcc4-glibc-pentium3-no-sse.so
- codec_g729-ast200-gcc4-glibc-pentium3.so
- codec_g729-ast200-gcc4-glibc-pentium4-no-sse.so
- codec_g729-ast200-gcc4-glibc-pentium4.so
- codec_g729-ast200-gcc4-glibc-pentium4-sse3.so
- codec_g729-ast200-gcc4-glibc-pentium-m.so
- codec_g729-ast200-gcc4-glibc-pentium.so
- codec_g729-ast200-gcc4-glibc-x86_64-barcelona.so
- codec_g729-ast200-gcc4-glibc-x86_64-core2.so
- codec_g729-ast200-gcc4-glibc-x86_64-core2-sse4.so
- codec_g729-ast200-gcc4-glibc-x86_64-optimon.so
- codec_g729-ast200-gcc4-glibc-x86_64-optimon-sse3.so
- codec_g729-ast200-gcc4-glibc-x86_64-pentium4.so

В примере (у Вас может быть по-другому) для данного типа процессора и платформы подойдет модуль, например: **codec_g729-ast200-gcc4-glibc-x86_64-core2-sse4.so** и аналогичный g723.

Параметры модуля описаны в его имени:

Ast200 – версия астериск 20

gcc4-glibc-x86_64 – 64 разрядная система

core2-sse4 – процессор core2 или свежее с набором инструкций sse4



В консоли Linux:

```
cd /usr/lib64/asterisk/modules/
```

```
wget http://asterisk.hosting.lv/bin/codec_g729-ast200-gcc4-glibc-x86_64-core2-sse4.so
```

```
wget http://asterisk.hosting.lv/bin/codec_g723-ast200-gcc4-glibc-x86_64-core2-sse4.so
```

первая команда переход в каталог с модулями

вторая и третья команды выкачивают необходимые модули кодеков

Перейдите в консоль Asterisk (*sudo asterisk -rvvvvv*)



В консоли Asterisk:

```
module load codec_g729<нажимаем-tab>
```

```
core show translation
```

первая команда подгружает модуль кодека

вторая команда выводит в консоль таблицу конвертации кодеков (таблицу транскодинга)



Конфигурация
Asterisk:

Для отключения загрузки модуля кодека g723 указываем в файле modules.conf:

```
noload => codec_g723-тут-пишем-точное-имя-файла.so
```

```
noload => res_pjsip_config_wizard.so
```

modules.conf – конфигурационный файл в котором прописано какие модули должен использовать Asterisk (расположен /etc/asterisk/)

noload – параметр файла modules.conf указывающий не подгружать соответствующий модуль, указываемый после знака =>

res_pjsip_config_wizard.so – модуль мастера настройки pjsip, мы его не будем применять, поэтому отключаем

Таблица транскодинга:

gsm ulaw alaw g726 adpcm slin lpc10 ilbc g726aal2 g722 slin16 testlaw slin12 slin24 slin32 slin44 slin48 slin96 slin192

gsm	15000	15000	15000	15000	9000	15000	15000	15000	17250	17000	15000	17000	17000	17000	17000	17000	17000
ulaw	15000	-	9150	15000	15000	9000	15000	15000	15000	17250	17000	15000	17000	17000	17000	17000	17000
alaw	15000	9150	-	15000	15000	9000	15000	15000	15000	17250	17000	15000	17000	17000	17000	17000	17000
g726	15000	15000	15000	-	15000	9000	15000	15000	15000	17250	17000	15000	17000	17000	17000	17000	17000
adpcm	15000	15000	15000	15000	-	9000	15000	15000	15000	17250	17000	15000	17000	17000	17000	17000	17000
slin	6000	6000	6000	6000	6000	-	6000	6000	6000	8250	8000	6000	8000	8000	8000	8000	8000
lpc10	15000	15000	15000	15000	15000	9000	-	15000	15000	17250	17000	15000	17000	17000	17000	17000	17000
ilbc	15000	15000	15000	15000	15000	9000	15000	-	15000	17250	17000	15000	17000	17000	17000	17000	17000
g726aal2	15000	15000	15000	15000	15000	9000	15000	15000	-	17250	17000	15000	17000	17000	17000	17000	17000
g722	15600	15600	15600	15600	15600	9600	15600	15600	15600	-	9000	15600	17500	17000	17000	17000	17000
slin16	14500	14500	14500	14500	14500	8500	14500	14500	14500	6000	-	14500	8500	8000	8000	8000	8000
testlaw	15000	15000	15000	15000	15000	9000	15000	15000	15000	17250	17000	-	17000	17000	17000	17000	17000
slin12	14500	14500	14500	14500	14500	8500	14500	14500	14500	14000	8000	14500	-	8000	8000	8000	8000
slin24	14500	14500	14500	14500	14500	8500	14500	14500	14500	14500	8500	14500	8500	-	8000	8000	8000
slin32	14500	14500	14500	14500	14500	8500	14500	14500	14500	14500	8500	14500	8500	8500	-	8000	8000
slin44	14500	14500	14500	14500	14500	8500	14500	14500	14500	14500	8500	14500	8500	8500	8500	-	8000
slin48	14500	14500	14500	14500	14500	8500	14500	14500	14500	14500	8500	14500	8500	8500	8500	8500	-
slin96	14500	14500	14500	14500	14500	8500	14500	14500	14500	14500	8500	14500	8500	8500	8500	8500	8500
slin192	14500	14500	14500	14500	14500	8500	14500	14500	14500	14500	8500	14500	8500	8500	8500	8500	8500

VoxLink - Asterisk Intensive

slin192 14500 14500 14500 14500 14500 8500 14500 14500 14500 14500 14500 8500 14500 8500 8500 8500 8500 8500

pci*CLI> core show translation

Translation times between formats (in microseconds) for one second of data

Source Format (Rows) Destination Format (Columns)

gsm ulaw alaw g726 adpcm slin lpc10 g729 ilbc g726aal2 g722 slin16 testlaw slin12 slin24 slin32 slin44 slin48 slin96 slin192

gsm - 15000 15000 15000 15000 9000 15000 15000 15000 15000 17250 17000 15000 17000 17000 17000 17000 17000

ulaw 15000 - 9150 15000 15000 9000 15000 15000 15000 15000 17250 17000 15000 17000 17000 17000 17000 17000

alaw 15000 9150 - 15000 15000 9000 15000 15000 15000 15000 17250 17000 15000 17000 17000 17000 17000 17000

g726 15000 15000 15000 - 15000 9000 15000 15000 15000 15000 17250 17000 15000 17000 17000 17000 17000 17000

adpcm 15000 15000 15000 15000 15000 9000 15000 15000 15000 15000 17250 17000 15000 17000 17000 17000 17000 17000

slin 6000 6000 6000 6000 6000 - 6000 6000 6000 8250 8000 6000 8000 8000 8000 8000 8000 8000

lpc10 15000 15000 15000 15000 15000 9000 - 15000 15000 15000 17250 17000 15000 17000 17000 17000 17000 17000

g729 15000 15000 15000 15000 15000 9000 15000 - 15000 15000 17250 17000 15000 17000 17000 17000 17000 17000

ilbc 15000 15000 15000 15000 15000 9000 15000 15000 - 15000 17250 17000 15000 17000 17000 17000 17000 17000

g726aal2 15000 15000 15000 15000 15000 9000 15000 15000 15000 - 17250 17000 15000 17000 17000 17000 17000 17000

g722 15600 15600 15600 15600 15600 9600 15600 15600 15600 - 9000 15600 17500 17000 17000 17000 17000 17000

slin16 14500 14500 14500 14500 14500 8500 14500 14500 14500 14500 6000 - 14500 8500 8000 8000 8000 8000

testlaw 15000 15000 15000 15000 15000 9000 15000 15000 15000 15000 17250 17000 - 17000 17000 17000 17000 17000

slin12 14500 14500 14500 14500 14500 8500 14500 14500 14500 14500 14000 8000 14500 - 8000 8000 8000 8000

slin24 14500 14500 14500 14500 14500 8500 14500 14500 14500 14500 14500 8500 14500 8500 - 8000 8000 8000 8000

slin32 14500 14500 14500 14500 14500 8500 14500 14500 14500 14500 14500 8500 14500 8500 8500 - 8000 8000 8000

slin44 14500 14500 14500 14500 14500 8500 14500 14500 14500 14500 14500 8500 14500 8500 8500 8500 - 8000 8000

slin48 14500 14500 14500 14500 14500 8500 14500 14500 14500 14500 14500 8500 14500 8500 8500 8500 - 8000 8000

slin96 14500 14500 14500 14500 14500 8500 14500 14500 14500 14500 14500 8500 14500 8500 8500 8500 - 8000 8000

slin192 14500 14500 14500 14500 14500 8500 14500 14500 14500 14500 14500 8500 14500 8500 8500 8500 8500 -

pci*CLI>

После установки кода g729 видим его в таблице транскодирования. До этого его не было в этой таблице.

1.7 Создание учетной записи SIP

В файле `pjsip.conf` прописываются следующие параметры (дописываем к секции `[udp-any]` созданной в предыдущем пункте):

- Создаем endpoint (sip-канал) и вспомогательные секции `auth` и `aor`
- Задаем пароль
- сети, из которых запрещаем подключение (все сети)
- сети, из которых разрешаем подключение (наша локальная сеть)
- указываем контекст
- выставяем параметр NAT
- запрещаем все кодеки
- разрешаем необходимые кодеки
- прописываем опцию проверки доступности устройства (`keepalive`)
- прописываем опцию возможности прямого медиа (запрещаем эту возможность)



Конфигурация
Asterisk:

```
[301]
type=endpoint
context=default
disallow=all
allow=ulaw
transport=udp-any
auth=auth301
aors=301
callerid=Vasiliy Petrov <301>
direct_media=no
force_rport=yes
rtp_symmetric=yes
deny=0.0.0.0/0.0.0.0
permit=адрес локальной сети/маска сети
language=ru

[auth301]
type=auth
auth_type=userpass
```

```
password=AsterPASS
username=301
```

```
[301]
type=aor
max_contacts=1
qualify_frequency=60
```

Создается SIP аккаунт «301», с подключением из подсети адрес локальной сети/маска сети который Вы указали.

type=endpoint основная секция для любого SIP-канала, она ссылается на несколько вспомогательных секций

transport=udp-any – транспорт говорит с какого адреса и порта работать с этим каналом, мы уже создали его ранее.

auth=auth301 – авторизационные данные.

aors=301 – Address Of Record(адрес записи) – определяет где брать контактные данные для связи с этим SIP-каналом. **max_contacts=1** указывает на то что абонент должен предоставить контактные данные при регистрации.

По аналогии с SIP-аккаунтом 301 мы могли бы создать сколько угодно подобных аккаунтов, копируя секции, однако есть способ лучше. Asterisk позволяет вынести повторяющуюся часть конфигурации в шаблон, и далее ссылаться на этот шаблон вместо повторения конфигурации (**замените ранее созданную учетную запись следующим шаблоном**):



Конфигурация
я
Asterisk:

; Шаблон для эндпоинтов

```
[office](!)
type=endpoint
context=from-internal
disallow=all
allow=ulaw
transport=udp-any
direct_media=no
force_rport=yes
rtp_symmetric=yes
rewrite_contact=yes
deny=0.0.0.0/0.0.0.0
permit=192.168.10.0/255.255.255.0
language=ru
```

; Шаблон для секций авторизации

```
[office-auth](!)
type=auth
auth_type=userpass
password=AsterPASS
```

; Шаблон для AOR

```
[office-aor](!)
type=aor
max_contacts=1
```

qualify_frequency=60

; Это эндпоинты с приклеенным шаблоном

[301](office)

auth=auth301

aors=301

callerid=Vasiliy Petrov <301>

[302](office)

auth=auth302

aors=302

callerid=Fedor Sumkin <302>

[303](office)

auth=auth303

aors=303

callerid=Ivan Karamazov <303>

[304](office)

auth=auth304

aors=304

callerid=Petr Bashirov <304>

; Авторизации с приклеенным шаблоном.

; На курсах мы используем единый пароль из шаблона для всех внутренних абонентов,

; в боевой системе применяйте уникальные и сложные пароли

[auth301](office-auth)

username=301

[auth302](office-auth)

username=302

[auth303](office-auth)

username=303

[auth304](office-auth)

username=304

; Секции AOR с приклеенным шаблоном (да, они все одинаковые, но каждый эндпоинт должен иметь собственный контакт и значит свой AoR)

[301](office-aor)

[302](office-aor)

[303](office-aor)

[304](office-aor)

Создается шаблон **[office]** для секций с типом **endpoint**, на его основе создаются SIP-каналы 301,302,303,304. Далее создается шаблон авторизации **[office-auth]** и на его основе создаются авторотационные данные для SIP-каналов 301,302,303,304.

И создается шаблон **[office-aor]** для создания записей адреса (AoR), и записи адреса для каждого SIP-канала.



В консоли Asterisk:

После того, как настройка выполнена, выполняем команду - **module reload res_pjsip**.

Проверяем, что все SIP-каналы созданы:

pjsip list endpoints – краткая форма
или **pjsip show endpoints** – расширенная форма

Проверяем, что созданы все логины-пароли:

pjsip show auths

Проверяем, что созданы все записи адреса (AoR):

pjsip list aors – краткая форма
или **pjsip show aors** – расширенная форма

1.8 Настройка IP-телефона

Настройте второй канал IP телефона и один из SoftPhone (установленный на компьютере или на мобильном телефоне) для подключения к Asterisk, как это было описано в первой практической работе, но используя учетные данные «Endpoint» (каналов) созданных в Asterisk.

При успешной регистрации телефона на сервере Вы должны увидеть в консоли такую строку:

```
localhost*CLI>
-- Added contact 'sip:301@192.168.10.50:5060' to AOR '301' with expiration of 3600 seconds
== Endpoint 301 is now Reachable
-- Contact 301/sip:301@192.168.10.50:5060 is now Reachable. RTT: 11.682 msec
-- Added contact 'sip:302@192.168.10.50:5061' to AOR '302' with expiration of 3600 seconds
== Endpoint 302 is now Reachable
-- Contact 302/sip:302@192.168.10.50:5061 is now Reachable. RTT: 11.443 msec
localhost*CLI>
```

Так же проверить зарегистрировался ли аккаунт в системе можно командой **pjsip show contacts**.

```
localhost*CLI> pjsip show contacts

Contact: <Aor/ContactUri.....> <Hash....> <Status> <RTT(ms)...>
=====
Contact: 301/sip:301@192.168.10.50:5060          7c240aadd2 Avail      9.096
Contact: 302/sip:302@192.168.10.50:5061          e5f3bff1ec Avail      7.043
Contact: 303/sip:303@192.168.10.67:5060          d166920094 Avail     13.212
Contact: 304/sip:304@192.168.10.67:5061          b6b1aa9f59 Avail      7.878

Objects found: 4
localhost*CLI>
```

Проверьте корректность проделанной работы, позвонив с одного телефона зарегистрированного на сервере Asterisk на другой телефон зарегистрированный на сервере Asterisk.