



IP Телефония и как она  
работает

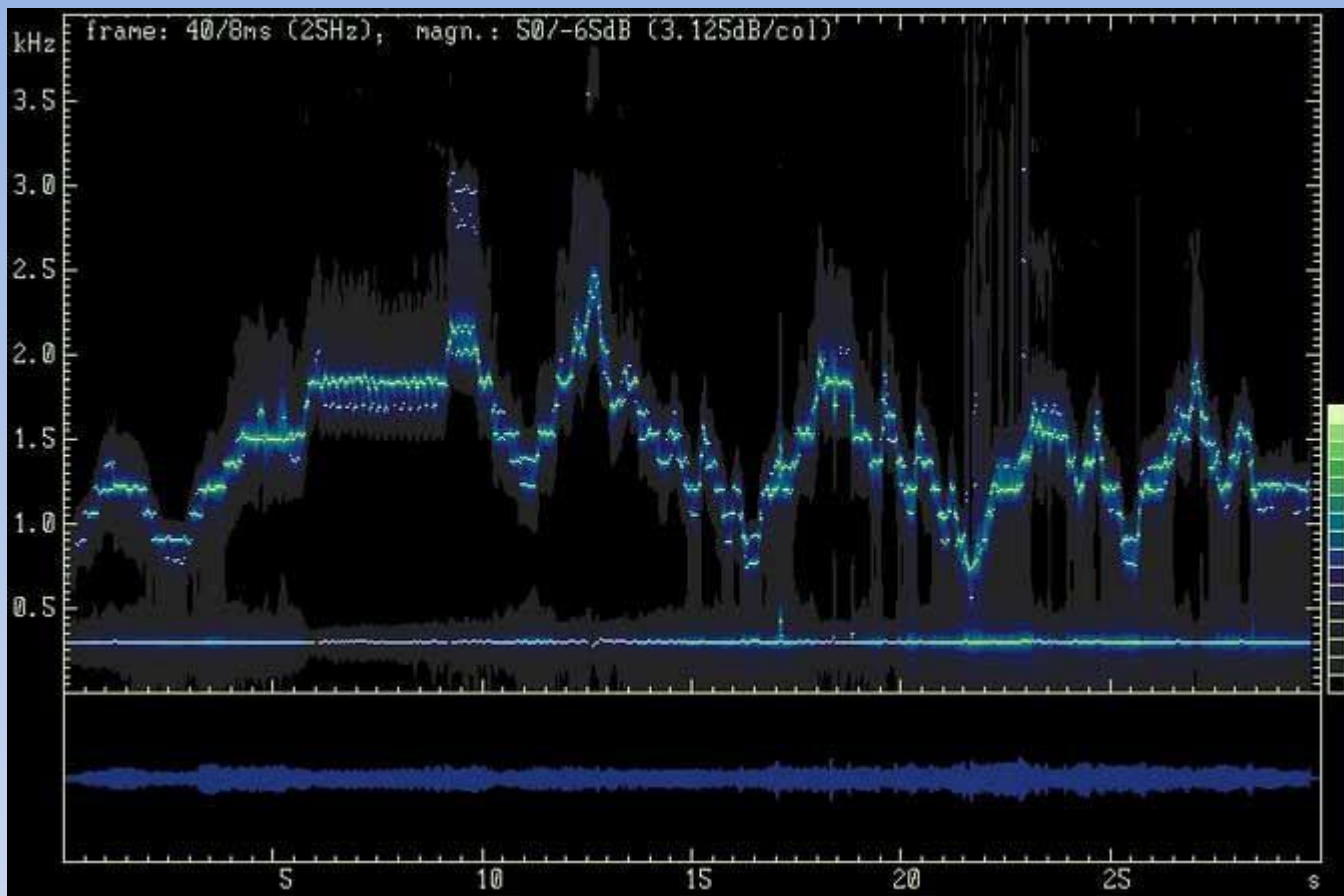


## 1928 год: Технологические истоки VoIP

В 1925 году компания AT&T, которая тогда была, по сути, единственным игроком рынка телефонных коммуникаций, открыла подразделение Bell Laboratories, в чьи задачи входило изобретение и дальнейшая разработка технологий связи, которые бы позволили бы компании улучшить свой сервис. В 1928 году Гомер Дадли (Homer Dudley) создал первый электронный синтезатор голоса, получивший название вокодер. Вокодер анализировал звуки, издаваемые ртом и связками человека, и умел воспроизводить их, что выливалось в аналог речи. Принцип работы вокодера можно сравнить с нынешними технологиями передачи пакетов, в ходе работы которых на одном телефоне записываются образцы звука, воспроизводимые затем на другом устройстве. Позднее во время Второй мировой войны вокодер использовался для передачи зашифрованных сообщений, а всемирную славу получил десятки лет спустя, когда его стали использовать музыканты вроде Kraftwerk.

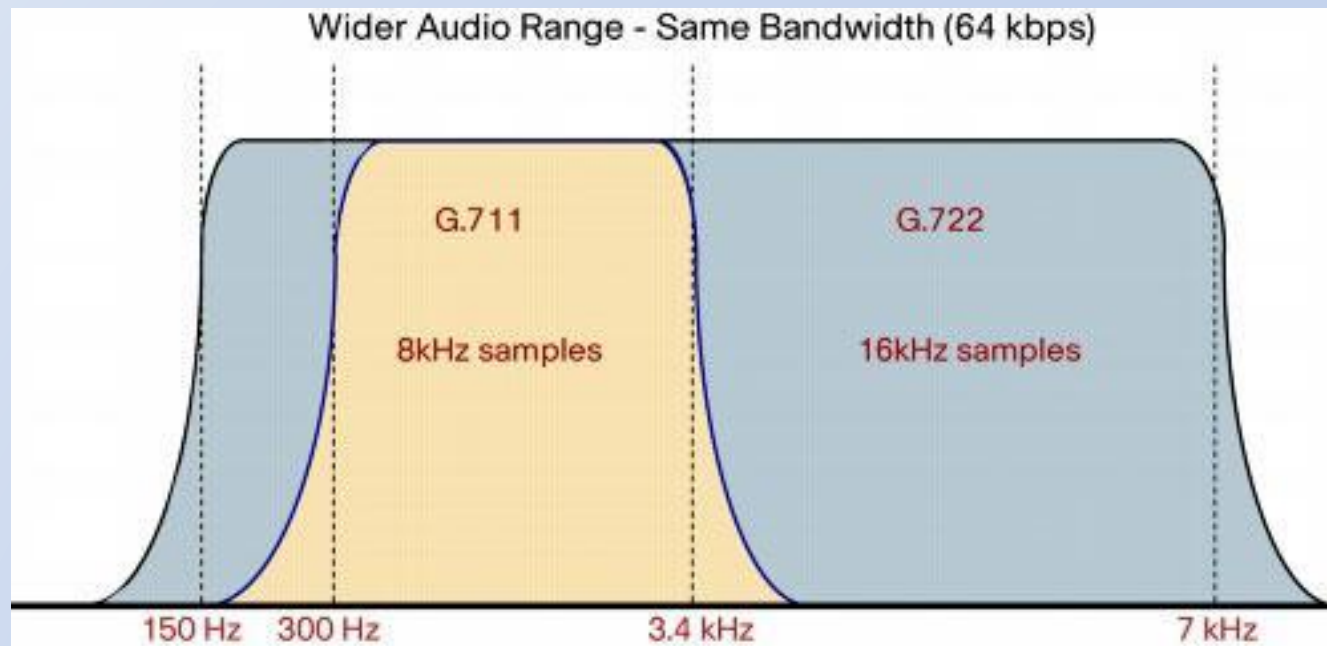






## 1973 год: Передан первый пакет голосовых данных

Финансируемая правительством США лаборатория Массачусетского технологического университета (MIT) под названием Lincoln Lab, разработала новую технологию (как это часто бывает и сейчас, разработка осуществлялась в интересах национальной безопасности). Боб МакОули (Bob McAuley) и Чарли Радар (Charlie Radar) создали формат пакета данных, который можно было передавать по сети ARPANET. Технология использовала для передачи голоса так называемый кодировщик с линейным предсказанием (LPC — linear predictive coder). В 1974 году команда успешно провела тестовый сеанс конференц-связи через LPC с офисом компании Culler Harrison, Inc., расположенном в Калифорнии. В 1982 году ученым удалось использовать LPC для связи через локальную кабельную сеть, мобильную радиосеть и интерфейс публичной телефонной сети (PSTN).



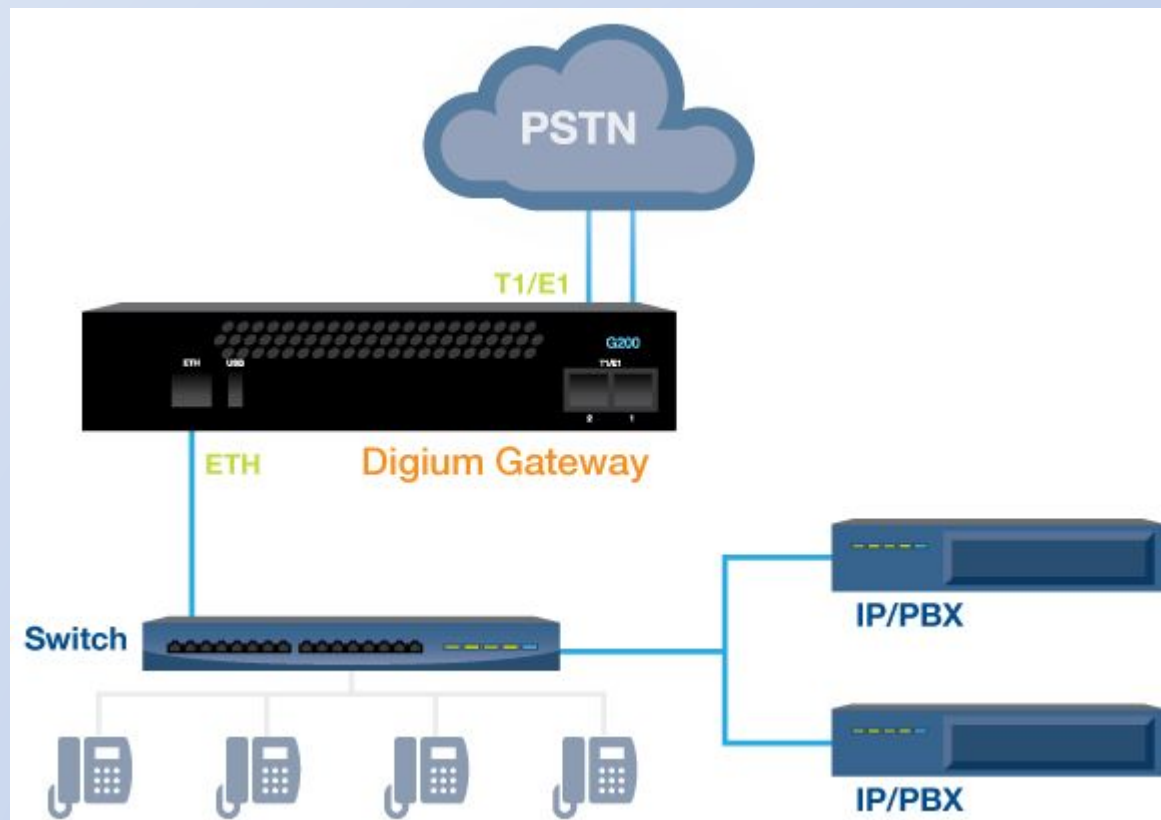
## 1988 год: Первый широкополосный аудиокодек

Международный телекоммуникационный союз (ITU-T) одобряет спецификацию аудиокодека G.722, написанного в Lincoln Labs. Он обладал битрейтом в 64 килобит в секунду, качество звука при этом было сравнимо с голосом, слышимым через обычную телефонную сеть. Поскольку битрейт кодека G.722 вдвое превосходил показатели предшественников, это позволяло добиваться звука высокого качества при тех же требованиях к пропускной способности сети.



## 1991 год: Выпущено первое публичное приложение VoIP

Основатель Autodesk Джон Уокер (John Walker) отправился в Европу, чтобы подготовить расширение своей компании. На тот момент для запуска голосовых приложений нужна была скорость не меньше 64 кбит/с, но Брайан Уайлс создает схему прореживания/расширения, с помощью которой можно снизить необходимую пропускную способность до 32 кбит/с. Он выпускает программу в свободное обращение под названием NetFone. Эта разработка, позднее известная как Speak Freely, является первым софтверным VoIP-телефоном. Уокер использует программу для проведения совещаний с разработчиками компании.



## 1996 год: первое серверное PBX-решение

Калифорнийская компания Virtual PBX выпускает первое серверное PBX-решение, в котором реализованы функции поиска и следования (find me/follow me), а также контроля с помощью веб-портала (у «обычных» PBX такой функциональности не было и много лет спустя). В отличие от клиентских решений, серверные PBX были легко масштабируемы, кроме того, для их использования не требовалось покупать или арендовать железо. Что интересно, первый серверный PBX не использовал VoIP, а с помощью медного провода соединялся с PSTN-сетью, давая возможность совершения звонков внутри одной компании.







# BUILD CUSTOM PHONE APPS THAT **talk to asterisk**

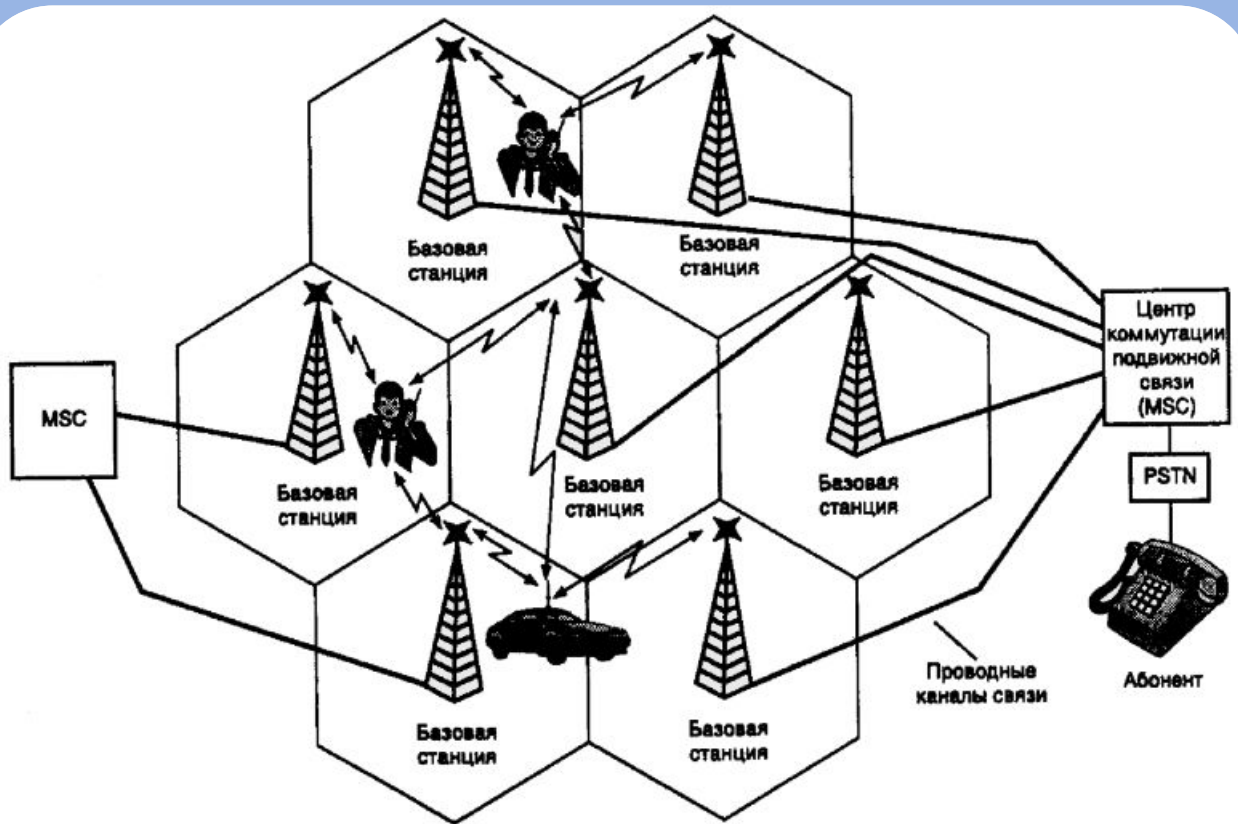


Asterisk and Digium Phones.  
with the Digium Phone App Engine API

Phones for Asterisk

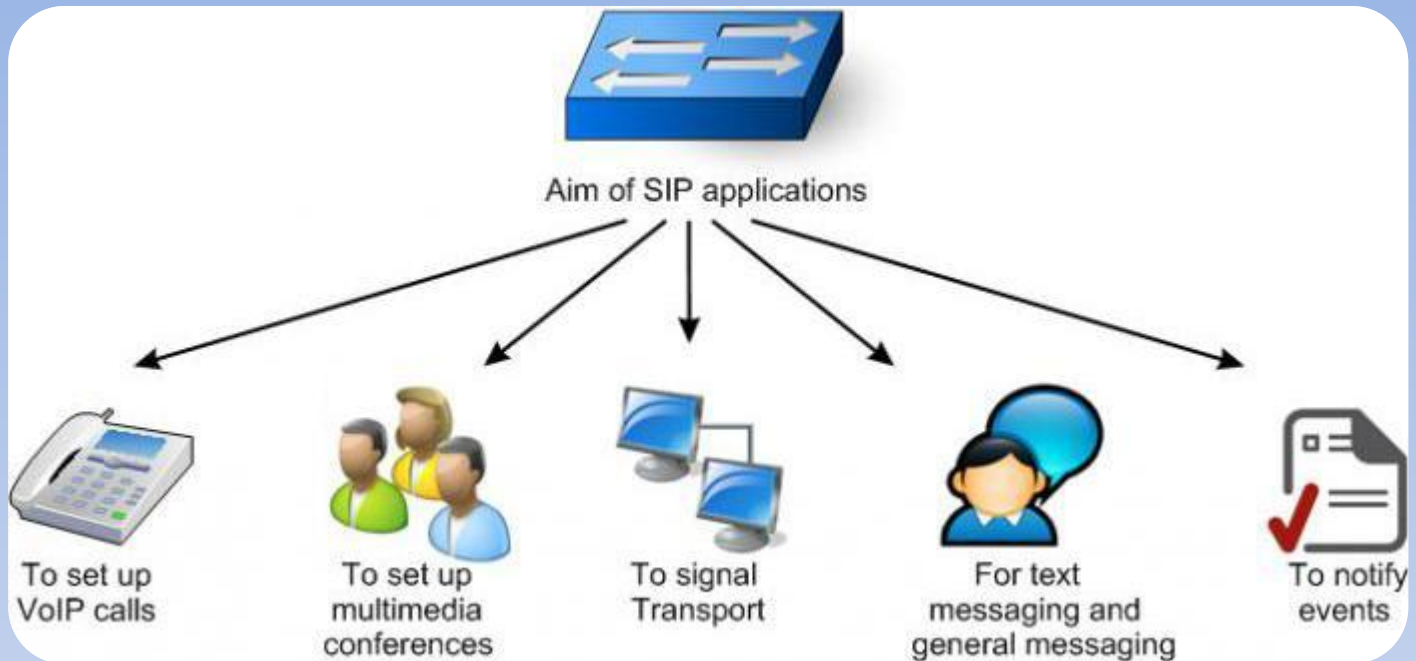
## **1999 год: Asterisk, первый IP-PBX**

Марк Спенсер (Mark Spencer) владел собственной компанией, занимавшейся техподдержкой Linux. Марк столкнулся с необходимостью установки PBX для разговоров с сотрудниками и для налаживания контактов между ними. Денег на покупку такого решения не хватало, поэтому он разработал собственный IP-PBX и назвал его Asterisk. Марк решил, что программа будет открытой, что помогло Asterisk завоевать популярность. В итоге компания Спенсера перешла с поддержки Linux на работу по консалтингу пользователей Asterisk, особенно относительно выбора железа для работы программы. За многие годы с момента создания Asterisk над проектом успели поработать тысячи программистов.



## Сотовая связь

Каждая ячейка, которой выделяется своя полоса частот, обслуживается базовой станцией, состоящей из передатчика, приёмника и модуля управления. Смежные ячейки используют разные частоты, чтобы избежать интерференции или перекрёстных помех. В то же время ячейки, находящиеся на большом расстоянии друг от друга, могут использовать одинаковые полосы частот. Таким образом, основной принцип сотовой связи — это повторное использование частот в несмежных сотах.



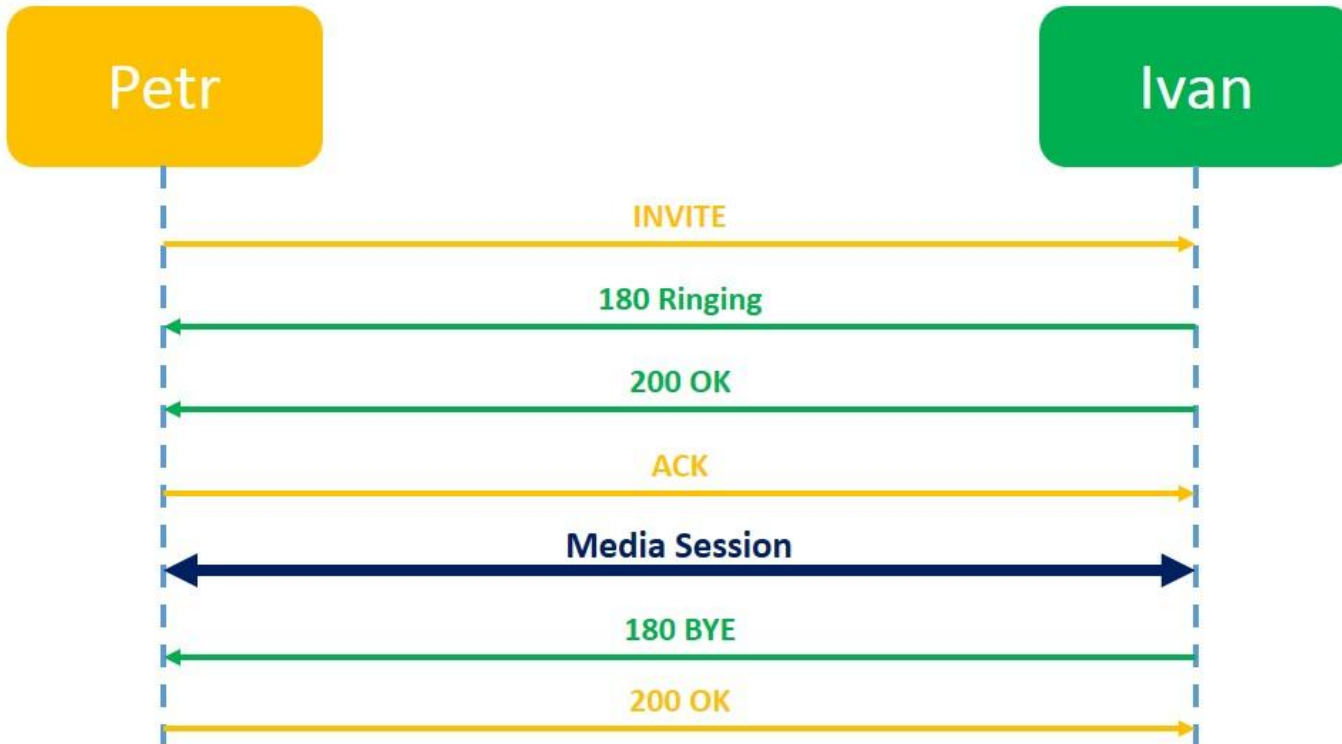
## Плюсы

1. Сравнительная простота.
2. Быстрота подключения.
3. Персональная мобильность пользователей.
4. Масштабируемость сети.

## Минусы

1. Проблемы с качеством связи.
2. Подмена CallerID и его доступность.
3. Наличие уязвимостей в передаче данных.

## Пример установки простой SIP-сессии





## Что такое Asterisk?



Это open source приложение, работающее на Linux и FreeBSD и предназначенное для создания решений компьютерной телефонии. Asterisk обладает всеми возможностями классической АТС, поддерживает три VoIP протокола (SIP/H323/IAX), предоставляет функции голосовой почты (VoiceMail), конференций, интерактивного голосового меню (IVR), центра обработки вызовов (постановка звонков в очередь и распределение их по агентам используя различные алгоритмы), а также имеет гибкий и универсальный интерфейс для интеграции с внешними системами обработки данных (AGI). Все это делает Asterisk мощнейшей платформой для организации телекоммуникационного сервиса любого масштаба.

# Плюсы Asterisk

- Мониторинг звонков в режиме online и запись разговоров, возможность коммутировать того или иного абонента через панель управления;
- Голосовые приветствия, динамические очереди распределения входящих звонков;
- Отсутствие необходимости в покупке дополнительных лицензий и плат телефонии;
- Возможность интегрирования со сторонними системами используя АМІ/АРІ интерфейс, например CRM системами и базами данных;
- Музыка в ожидании на выбор заказчика;
- Распределение звонков, в первую очередь Asterisk - это IP-АТС со всеми возможностями VoIP телефонии.

# Что такое

## IVR?

Интерактивное Голосовое Меню (IVR) — это система предварительно записанных голосовых сообщений, выполняющая функцию маршрутизации звонков внутри call-центра, пользуясь информацией, вводимой клиентом с помощью тонального набора.

## Плюсы

### IVR

- IVR является способом снижения нагрузки на сотрудников компании. Клиенты, звонящие в компанию, самостоятельно могут выбрать маршрут прохождения звонка.
- IVR является лицом компании. Наличие интерактивного меню считается хорошим корпоративным стилем и оказывает влияние на престиж компании в глазах ее клиентов.
- Наличие IVR позволяет компании обработать входящий звонок в нерабочее время, когда все сотрудники отсутствуют на работе (например, в вечернее время или по праздникам). Система IVR может информировать клиента о графике работы офиса компании, его расположении, схеме проезда, а также записать голосовое сообщение, которое будет переправлено на электронный адрес менеджера.
- IVR позволяет организовать рекламно-информационное обслуживание клиентов, которое можно осуществлять как в момент ожидания ответа оператора (в очереди), так и по явному выбору клиента.

# Session Border Controller

SBC (Session Border Controller, пограничный контроллер сессий) – это оборудование (или ПО), устанавливаемое на границе сетей для контроля передаваемого трафика.

## Основные функции SBC

1. Скрытие внутренней топологии вашей сети от внешнего мира.
2. Разделение доверенных (внутренних) и недоверенных (внешних) сетей по различным физическим сетевым интерфейсам (разделение сетей на физическом уровне).
3. Защита сетей.
4. Обеспечение работы абонентов за NAT.
5. Разделение типов трафика по различным интерфейсам.
6. Маршрутизация SIP трафика на основе IP адресов, А и Б номеров, времени суток, доступности оборудования, стоимости минуты трафика и т.д. и т.п.
7. Возможность перемаршрутизации трафика на основе ответа окончательного оборудования.



## Мои обязанности как инженера телефонии:

- Поддержка работоспособности текущих конфигураций Asterisk и SBC;
- Настройка новых каналов связи;
- Конфигурирование Asterisk и SBC;
- Мониторинг работы каналов связи, Asterisk и SBC;
- Взаимодействие с операторами связи и заказчиками по вопросам телефонии.



*That's all Folks!*