

Описание HTTP API. 1/7

Порядок взаимодействия:

С помощью API вы можете получать события, происходящие на вашей АТС Sipuni, такие как начало звонка, ответ оператора, перевод звонка, завершение звонка. Благодаря API вы сможете интегрировать Sipuni в любую вашу систему.

В настройках системы вы указываете URL адрес, например — [http\[s\]://example.com\[:port\]/\[pbx_prefix\]/](http[s]://example.com[:port]/[pbx_prefix]/), и нажимаете кнопку «Сохранить», после этого на него начинают приходить сообщения с АТС в случае различных событий.

Ваш адрес должен отвечать в случае успеха корректной JSON строкой, содержащей обязательное поле success. В случае успешной обработки success будет true, в противном случае — false. В целях безопасности, в случае если корректный ответ не получен, отправка сообщений может быть приостановлена. Также в JSON могут быть любые другие параметры ответа.

Минимальный корректный ответ:

```
{“success”: true}
```

Вы также можете реализовать запросы от сервиса к АТС, такие как заказ вызова, обратившись к соответствующему API.

HTTP запросы:

Входные параметры передаются, используя методы GET или POST. Все строки имеют кодировку символов UTF-8, если не указано иное, и передаются в запросе после применения URL кодирования. Порядок параметров в запросе является произвольным.

Обязательные параметры для всех видов запросов:

- event – тип запроса
- call_id – уникальный идентификатор вызова (сохраняется неизменным при переводе), является строкой произвольного формата (с использованием URL кодировки)
- src_num – адрес абонента инициализировавшего вызов (сохраняется при переводе)
- src_type – тип адреса (1 - внешний, 2 - внутренний)
- dst_num – адрес назначения – может быть пустым при запросе на «умную» переадресацию».
- dst_type – тип адреса назначения (1 - внешний, 2 - внутренний)
- timestamp – время события (начала/завершения вызова, перевода, ответа), представляет собой Unix timestamp (UTC).

Адрес (в определении параметров, перечисленных выше и далее по тексту в данном документе) является телефонным номером либо строкой. Адрес для внешних номеров обязательно должен быть номером, внутренний адрес может являться как номером так и строкой произвольного формата в URL кодировке (например SIP аккаунт).

Виды запросов:

а. **Call** – *событие начала звонка* event = 1

Событие генерируется при инициировании, либо при переводе вызова. При переводе сохраняется src_num и call_id текущего вызова.

Как только вызов поступает на устройство, приходит event=1.

Для каждого события Call, используемого как оповещение о начале/переводе вызова должно обязательно генерироваться событие Hang-up.

Обязательные параметры:

- is_inner_call – флаг определяет, пришел ли данный вызов из вне системы (например, с городского номера) (если звонок внутри системы приходит is_inner_call=1, если из вне этот флаг не приходит)

По сути, можно рассматривать как 2 разных события:

(event = 1 – инициализация звонка, обычно первое событие звонок приходит в систему)

(event = 1, is_inner_call=0 – звонок внутри системы, например идет дозвон на короткий номер)

Дополнительные параметры:

Если подключен колтрекинг Roistat и системе удастся отследить звонок, то приходят также следующие параметры:

- roistat – ид посетителя roistat_visit_id

- roistat_number – (подмененный) номер на который звонил посетитель

- roistat_market – метка вида source_medium_campaign_keyword

b. **Hang-up – событие окончания звонка** event = 2

Обязательные параметры:

- status:

ANSWER, - вызов отвечен

BUSY, - абонент занят

NOANSWER, - абонент не ответил после определённого таймаута

CANCEL, - вызов сброшен

CONGESTION, - перегрузка сети

CHANUNAVAIL – абонент недоступен (например sip абонент не зарегистрирован в сети)

- call_start_timestamp – время начала вызова

- call_answer_timestamp – время ответа на вызов, в случае отсутствия ответа, значение данного параметра должно быть равно 0.

Дополнительные параметры:

- call_record_link – URL на файл записи разговора (в URL кодировке).

Если подключен колтрекинг Roistat и системе удастся отследить звонок, то приходят также следующие параметры:

- roistatgoogleid – ид gooogle_client_id (если доступен)

c. **Answer – ответ на вызов** event = 3

Событие посылается при ответе на вызов.

Дополнительные параметры отсутствуют.

d. **Secondary hang-up – промежуточное завершение вызова.** event = 4

Генерируется, например, при переводе звонка с подсказкой, когда номер с которого переводят звонок кладёт трубку и звонок полностью переводится на тот номер, на который переводят.

Обязательные параметры:

- status:

ANSWER, - вызов отвечен

BUSY, - абонент занят

NOANSWER, - абонент не ответил после определённого таймаута

CANCEL, - вызов сброшен

CONGESTION, - перегрузка сети

CHANUNAVAIL – абонент недоступен (например sip абонент не зарегистрирован в сети)

- call_start_timestamp – время начала вызова

- call_answer_timestamp – время ответа на вызов, в случае отсутствия ответа, значение данного параметра должно быть равно 0.

Дополнительные параметры:

- call_record_link – URL на файл записи разговора (в URL кодировке)

Пример:

Прямой вызов с переводом (это то, что отправляет АТС)

Звонок пришел в систему (на номер компании 8499999999)

http://example.com:9090/?event=1&call_id=1419783130.15593&src_num=8955555555&src_type=1&dst_num=8499999999&dst_type=1×tamp=1378296000

Начало звонка с внешнего номера 8955555555 на внутренний номер 101

http://example.com:9090/?event=1&is_inner_call=1&call_id=1419783130.15593&src_num=8955555555&src_type=1&dst_num=012345101&dst_type=2×tamp=1378296000

Номер 101 поднял трубку

http://example.com:9090/?event=3&call_id=1419783130.15593&src_num=8955555555&src_type=1&dst_num=012345101&dst_type=2×tamp=1378296010

Вызов переведён на номер 102

http://example.com:9090/?event=1&is_inner_call=1&call_id=1419783130.15593&src_num=8955555555&src_type=1&dst_num=012345102&dst_type=2×tamp=1378296020

Номер 102 поднял трубку (разговаривают 101 и 102 номер)

http://example.com:9090/?event=3&call_id=1419783130.15593&src_num=8955555555&src_type=1&dst_num=012345102&dst_type=2×tamp=1378296030

Номер 101 положил трубку (разговаривают уже 8955555555 и 102):

http://example.com:9090/?event=4&call_id=1419783130.15593&src_num=8955555555&src_type=1&dst_num=012345102&dst_type=2&call_start_timestamp=1378296000&call_answer_timestamp=1378296040×tamp=1378296040&status=ANSWER

Переведённый вызов завершён:

http://example.com:9090/?event=2&call_id=1419783130.15593&src_num=8955555555&src_type=1&dst_num=012345102&dst_type=2&call_start_timestamp=1378296000&call_answer_timestamp=1378296025×tamp=1378296050&status=ANSWER&call_record_link=ftp%3a%2f%2fmyhost.info%2faccount%2ffrecords%2f354435435436.wav

Примечание. Короткие и длинные номера:

Логины номеров в SIPUNI имеют вид <номер пользователя><короткий номер>.

И в полях src_num и dst_num приходят именно логины. Для пользователя с номером в системе 012345, короткий номер 101 будет обозначаться 012345101.

Некоторым пользователям достаточно только самого короткого номера, поэтому в запросах дополнительно добавлены поля short_src_num и short_dst_num, куда записываются не логины, а сами короткие номера.

Например, если src_num = 012345101, то short_src_num = 101. Поля различаются только короткими номерами SIPUNI. Для внешних номеров они полностью одинаковы, например, если src_num = 79031234567, то и short_src_num = 79031234567.

* API всё ещё находится на стадии тестирования и расширения функционала.