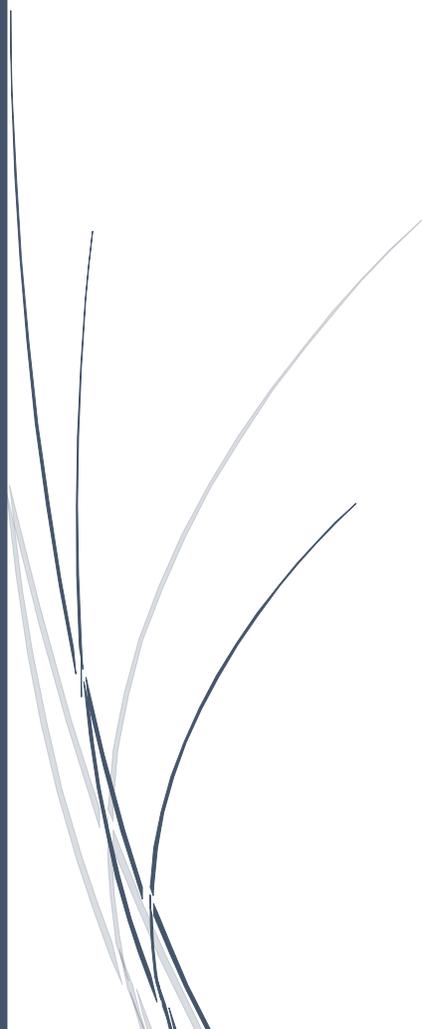




Платформа OMMG 3.0

Описание API



Георгий Хазан
Оммджи Технолоджи

Оглавление

1. Общая идеология OMMG API.....	4
2. Используемые сокращения и определения.....	5
3. Авторизация и управление устройствами.....	7
3.1. Регистрация нового устройства.....	7
3.2. Создание сессии.....	9
3.3. Регистрация пуш-уведомлений.....	10
3.4. Чтение входящих сообщений в сессии.....	11
3.5. Окончание сессии.....	12
3.6. Получение всех данных пользователя.....	13
3.7. Изменение имени пользователя.....	14
3.8. Изменение статуса сессии.....	15
3.9. Получение списка устройств.....	16
4. Работа с сообщениями.....	17
4.1. Написание нового сообщения.....	17
4.2. Редактирование сообщения.....	19
4.3. Удаление сообщения.....	20
4.4. Групповое удаление сообщений.....	21
4.5. Уведомление о наборе текста.....	22
4.6. Уведомление о доставке.....	23
4.7. Уведомление о прочтении.....	24
4.8. Получение истории сообщений.....	25
4.9. Получение последнего сообщения.....	27
4.10. Выставление реакций на сообщение.....	28
4.11. Отметка реакции прочитанной.....	29
4.12. Получение списка непрочитанных реакций.....	30
5. Работа со списком пользователей.....	31
5.1. Получение списка пользователей.....	31
5.2. Изменение информации о пользователе в списке.....	32
5.3. Удаление пользователя из списка контактов.....	33
5.4. Направление запроса о дружбе.....	34
5.5. Принятие запроса о дружбе.....	35
5.6. Отклонение запроса о дружбе.....	36
5.7. Получение списка запросов на дружбу.....	37
6. Списки видимости.....	38
6.1. Получение списков видимости.....	38

6.2. Изменение списков видимости.....	39
7. Работа с личной карточкой.....	40
7.1. Получение личной карточки пользователя.....	40
7.2. Считывание списка личных данных.....	41
7.3. Запись собственной личной карточки.....	42
8. Групповые чаты.....	43
8.1. Создание группового чата.....	43
8.2. Уничтожение группового чата.....	45
8.3. Редактирование группового чата.....	46
8.4. Получение списка групповых чатов для пользователя.....	47
8.5. Получение списка участников группового чата.....	48
8.6. Изменение списка участников группового чата.....	49
8.7. Изменение статуса в чате.....	51
9. Поддержка голосовых и видеозвонков.....	52
9.1. Инициализация приватного звонка (peer-2-peer).....	52
9.2. Инициализация группового звонка.....	53
9.3. Присоединение к групповому звонку.....	54
9.4. Поддержка присутствия в групповом звонке.....	55
9.5. Публикация оффера для SDP в групповом чате.....	56
9.6. Публикация ICE-кандидата в групповом чате.....	57
9.7. Подписка на публикации в групповом чате.....	58
9.8. Посылка SDP подписчика.....	59
9.9. Выход из группового звонка.....	60
9.10. Окончание группового звонка.....	61
10. Работа с командами.....	62
10.1. Создание команды.....	62
10.2. Уничтожение команды.....	63
10.3. Получение информации о команде.....	64
10.4. Присоединение себя к команде.....	65
10.5. Получение списка своих команд.....	66
10.6. Получение списка участников команды.....	67
10.7. Изменение команды.....	68
Приложение 1. Коды ошибок, используемые в API.....	69
Приложение 2. Расширенный формат записи событий в истории.....	70
Приложение 3. Файловые операции.....	72
1. Загрузка файла одним запросом.....	72
2. Загрузка больших файлов по частям.....	72
3. Работа с аватарками.....	73
4. Конвертация файлов.....	73

5. Предварительный просмотр.....	74
6. Получение ссылки на файл.....	74
7. Удаление файлов.....	74
8. Получение списка файлов.....	75
Приложение 4. Работа с телефонными книжками.....	76
1. Сохранение ТК на сервере.....	76
2. Считывание ТК с сервера.....	78

1. Общая идеология OMMG API

OMMG API (далее – API) использует только HTTPS запросы, направляемые на поддомен api того домена, на котором разворачивается платформа OMMG. Например, для домена yourdomain.com все запросы надо посылать на поддомен `https://api.yourdomain.com`

В описании запроса всегда указывается относительный путь к нужному URL, начиная от корня сайта `api.yourdomain.com`. Так, если в описании указано, что надо запросить `/message/send`, то запрос надо отправить на `https://api.yourdomain.com/message/send`

Все запросы имеют тип POST, содержат тело запроса в формате json, таким образом в запросе обязательно должно быть указано поле заголовка
>Content-type: application/json

Тело запроса формируется как структура данных, содержащая перечисление параметров. Таким образом, если в описании конкретного запроса указаны параметры P1 и P2, то тело запроса должно быть сформировано как JSON следующего вида:
{ "P1": value, "P2": value }

Все запросы требуют наличия строки авторизации в формате HTTP Basic Auth, если явно не указано обратное. В качестве пароля необходимо передавать md5 hash от склейки строк пароля и идентификатора устройства (globalId), использованного при регистрации устройства. Например, для пароля mypass и идентификатора устройства ios4698237469582345 надо в качестве пароля передать

```
>select md5('mypass'+ 'ios4698237469582345') as hash;
```

```
-----  
| hash  
-----
```

```
| 389d176f3034de5ef081c1403f758079  
-----
```

Ответ присылается сервером также в формате json. В случае успешного завершения возвращается код ошибки 200, а в json появляются поля "status": "ok" и "statusCode": 200.

В случае возникновения ошибки присылается код ошибки (40x, 50x), в поле status указывается непосредственная причина ошибки, а в поле statusCode – целочисленный расширенный код ошибки, полный список которых указан в Приложении 1.

В том случае, если по итогам выполнения запроса рассылается уведомление на другие устройства того же пользователя, к ним добавляется поле from_global_id, куда помещается код устройства, инициировавшего операцию. Таким образом устраняется дублирование, например, удаления сообщений. Соответственно, по наличию или отсутствию этого поля клиент может судить о том, входящее это уведомление или исходящее.

2. Используемые сокращения и определения

INT – поле целого типа, 32 бита, знаковое.

SID – серверный идентификатор, поле целого типа, 64 бита, знаковое

JID – идентификатор пользователя вида login@domain, в формате utf8

JID_ARRAY — массив JID, заключенный в квадратные скобки ([JID1, JID2, JID3])

GUID – строка из 32 шестнадцатеричных символов, записанных подряд

STRING – произвольная строка в формате utf8

TIME_STAMP – время в формате Unix time 64 (число миллисекунд с 01.01.1970)

BOOL – целое число, которое может принимать значения 0 или 1

bool – поле в json, может принимать значения false или true.

STATUS – строковая константа, код статуса пользователя:

- "1": OFFLINE;
- "3": VISIBLE;
- "5": INVISIBLE;
- "7": AWAY;
- "8": OCCUPIED

PLATFORM_TYPE – целое число, тип платформы для клиентского приложения:

- 0: Windows;
- 1: IPHONE;
- 2: Android;
- 3: MAC;
- 4: WinPhone;
- 5: Web.

CONTACT_FLAGS – целое число (32 бита), представляющее собой набор флагов для контакта в списке контактов или группового чата, участником которого вы являетесь. Смысл нижеследующих флагов не может быть изменён, все остальные битовые маски доступны для переопределения в клиентском приложении:

- 0x00000001 – контакт или групповой чат прикреплен к верху списка контактов.

RESPONSE_TYPE – целое число, тип входящего события

- 101: изменение статуса сессии;
- 102: изменение ника;
- 103: изменение аватара;
- 104: изменение списков видимости;
- 105: приватный звонок (peer-2-peer);
- 106: групповой звонок;
- 107: присоединение к групповому звонку;
- 108: выход из группового звонка;
- 109: окончание группового звонка;
- 198: вход с другого устройства;
- 199: выход с другого устройства;
- 201: приглашение в групповой чат;
- 202: присоединение к групповому чату;
- 203: выбрасывание из группового чата;
- 204: выход контакта из комнаты;
- 205: вход контакта в комнату;
- 206: изменение параметров группового чата;
- 207: удаление чата;
- 301: приглашение в ростер;
- 302: получение положительного ответа на приглашение в ростер;
- 303: получение отрицательного ответа на приглашение в ростер;
- 304: удаление контакта из списка контактов;
- 306: изменение списка контактов;
- 307: новый зарегистрировавшийся контакт из телефонной книги;

- 308: обновление телефонной книги;
- 309: обновление никнейма контакта;
- 310: изменение команды;
- 401: новое сообщение;
- 402: изменение сообщения;
- 403: удаление сообщения;
- 404: удаление всех сообщений;
- 405: уведомление о доставке сообщения;
- 406: уведомление о прочтении сообщения;
- 407: пользователь начал набор текста;
- 408: пользователь закончил набор текста;
- 409: аудио- или видео- сообщение прослушано;
- 410: уведомление о выставленной или удалённой реакции на сообщение;
- 411: уведомление о прочтении реакции.

CHAT_FLAGS – целое число (32 бита), представляющее собой набор свойств группового чата (применяется для создания, модификации или получения информации).

- 0x00000001 – публичный групповой чат (в него может войти любой посторонний контакт);
- 0x00000002 – модерлируемый групповой чат (с разделением участников на роли);
- 0x00000004 – история группового чата не копируется участникам (рекомендуется для каналов);
- 0x00000008 – скрытый чат (не выдаётся в /message/last, например);
- 0x00000010 – не генерировать идентификатор чата самостоятельно, использовать localId.

CHAT_STATUS – целое число, статус участника в групповом чате:

- 0: офлайн;
- 1: онлайн;
- 2: исключён из чата

CHAT_ROLE – строка, роль пользователя в групповом чате:

- "1": владелец чата;
- "2": администратор (может изменять состав пользователей);
- "3": модератор (может отправлять сообщения в модерлируемый чат);
- "4": участник (может отправлять сообщения в немодерлируемый чат);
- "5": посетитель (может читать сообщения в чате).

3. Авторизация и управление устройствами

3.1. Регистрация нового устройства.

Регистрация нового устройства проходит в два этапа. Сначала серверу выставляется запрос на регистрацию, в котором указывается мобильный телефон. В ответ на это сервер высылает на этот номер телефона смс с кодом, после чего клиент делает тот же самый вызов с указанием присланного кода. Код смс является временным и действует на протяжении 2 часов (если в настройках сервера не задано другое значение), после этого код надо запросить заново.

В зависимости от настроек сервера код может передаваться разными способами:

- наивысший приоритет имеет рассылка кода регистрации в виде сообщения на другие устройства пользователя, подключённые к Платформе;
- если указан параметр ring и на сервере подключён сервис зачитывания кода по телефона с помощью звонка, то будет осуществлён прямой звонок пользователю на указанный телефон;
- если не указан прямой запрет на рассылку СМС (параметр nosms), то пользователю отправляется СМС.

Вызов не требует авторизации.

Запрос:

/register

Параметры запроса:

Имя	Тип	Описание
login	STRING	Логин пользователя в виде строки
globalId	GUID	Идентификатор устройства (например, хеш IMEI)
name	STRING	Наименование устройства
platform	PLATFORM_TYPE	Тип платформы
lang	STRING	Двухбуквенный код языка для смс и голосовых звонков (например, "en" или "ru")
code	STRING	Присланный код регистрации, опционально
nosms	BOOL	запрет на рассылку СМС (по умолчанию равен 0)
ring	BOOL	использовать сервис звонков

Ответ (если параметр code пропущен):

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {"sms_sent": 1 }
}
```

Ответ (если параметр code указан):

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "password": STRING // сгенерированный пароль для устройства
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
403	Передан неверный номер телефона
604	Звонок завершился неудачей

606	Нет других активных устройств
2004	Пропущен обязательный параметр (globalId или login)
2007	Неверный тип параметра
2008	Неверный код регистрации
2009	Отсутствует код регистрации

3.2. Создание сессии.

Перед тем, как отправить любой другой запрос, необходимо войти на сервер и выставить свой статус в любое значение, отличное от STATUS.OFFLINE. Без этого все запросы будут отвергаться с кодом ошибки 3001 (неверный логин или пользователь). В дальнейшем, если интервал между запросами превысит 30 минут, сессия может быть автоматически сброшена сервером, тогда, при получении кода ошибки 3001, надо послать этот запрос на сервер, дождаться положительного ответа и послать исходный запрос ещё раз.

Запрос:

/logIn

Параметры запроса:

Имя	Тип	Описание
status	STATUS	выставляемый статус сессии, по умолчанию VISIBLE
textStatus	STRING	статусное сообщение, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "lastSid": SID,           // последний прочитанный устройством идентификатор
  события
  "user": {
    "firstName": STRING,   // имя пользователя
    "lastName": STRING,   // фамилия пользователя, опционально
    "avatar": STRING,     // хеш аватара
    "status": STATUS,     // статус сессии
    "textStatus": STRING  // статусное сообщение, опционально
  }
}
```

После успешного входа на сервер всем контактам из списка контактов, а также всем другим устройствам этого пользователя, будет разослано следующее уведомление:

```
{
  "type": "101",          // тип уведомления – смена статуса
  "from": JID,           // ваш идентификатор пользователя
  "ts": TIME_STAMP,     // серверная метка времени
  "status": STATUS,     // статус сессии
  "platform": PLATFORM_TYPE // тип платформы пользователя
}
```

Возможные сообщения об ошибках:

Код	Значение
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

3.3. Регистрация пуш-уведомлений

В том случае, если клиент стартует с платформы, поддерживающей пуш-уведомления (iOS, Mac, Android, POCA), необходимо передать на сервер специальный платформо-зависимый ключ (токен), отвечающий за доставку пушей в данное приложение.

Если на серверной стороне применяется менеджер пушей OMMG, то в поле флагов можно указывать следующие комбинации битов:

```
FLAG_PRIVATE_MESSAGE = 0x0001; // уведомлять о приходе частных сообщений
FLAG_HIGHLIGHT_MESSAGE = 0x0002; // уведомлять о подсветке имени в сообщении чата
FLAG_CHAT_MESSAGE = 0x0004; // уведомлять обо всех сообщениях в чатах
FLAG_CUT_PRIVATE = 0x0008; // обрезать текст частных сообщений
FLAG_CUT_CHAT = 0x0010; // обрезать текст сообщений в группчатах
FLAG_SOUND_PRIVATE = 0x0020; // звуковое уведомление о частных сообщениях
FLAG_SOUND_CHAT = 0x0040; // звуковое уведомление о сообщениях в группчатах
FLAG_SOUND_ACTIVATE = 0x0080; // звуковое уведомление об активации устройства
FLAG_ONLINE_MESSAGE = 0x0100; // уведомлять о выходе контактов в онлайн
FLAG_INVITE = 0x0200; // уведомлять о приходе приглашений на дружбу
FLAG_JOINED = 0x0400; // уведомлять о регистрации контакта
FLAG_INTERACTIVE_PUSH = 0x0800; // флаг поддержки интерактивных пушей (Андроид)
```

Если используется серверная платформа AWS или Firebase, то значения флагов игнорируются, и значение имеет только равенство этого флага нулю, в этом случае доставка пушей на данное устройство блокируется.

Запрос:

/setToken

Параметры запроса:

Имя	Тип	Описание
token	STRING	Токен устройства
platform	PLATFORM_TYPE	Код платформы, передаётся как строка
flags	INT BOOL	Комбинация флагов для управления пушами

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

Возможные сообщения об ошибках:

Код	Значение
1006	Такое устройство не зарегистрировано в системе
2004	Пропущен обязательный параметр
2005	Передан неверный тип платформы
2007	Неверный тип параметра
2006	Неверное сочетание флагов
3001	Неверный логин пользователя или пароль
4007	Платформа не поддерживает пуши

3.4. Чтение входящих сообщений в сессии

Для чтения входящих сообщений необходимо послать HTTP запрос, который ожидает ответа сервера не свыше 30 секунд, после чего сервер принудительно разрывает сокет. Если данные от сервера приходят раньше, то сервер посылает данные в сокет и разрывает соединение сразу же, после чего клиент должен послать ещё один такой же запрос. В блоке history возвращаются все сообщения.

Запрос:

/pollEvents

Параметры запроса:

Имя	Тип	Описание
lastSid	SID	Идентификатор последнего прочитанного сообщения или 0
nowait	BOOL	Опционально, сбрасывать соединение сразу же в любом случае

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "history": [
      {
        "type": RESPONSE_TYPE, // тип события
        ... // данные, относящиеся к данному событию
        "ts": TIME_STAMP, // серверное время сообщения
      }
    ],
    "lastSid": SID // идентификатор последнего присланного сообщения
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

3.5. Окончание сессии

Сервер автоматически переводит устройство в оффлайн, если не получает от него очередной запрос на pollEvents в течение заданного на сервере интервала времени. Однако рекомендуется по возможности насильственно завершать сессию, не дожидаясь таймаута. Также с помощью этого запроса можно отключить устройство или разрушить учётную запись пользователя целиком.

Запрос:

/logout

Параметры запроса:

Имя	Тип	Описание
removeAccount	BOOL	Надо ли уничтожить всю учётную запись, опционально
killUser	BOOL	Надо ли разрегистировать данное или другое устройство, опционально
device	GUID	Идентификатор устройства для параметра killUser, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

После окончания сессии всем контактам из списка контактов, а также всем другим устройствам этого пользователя, будет разослано следующее уведомление:

```
{
  "type": "101",           // тип уведомления - смена статуса
  "from": JID,            // ваш идентификатор пользователя
  "ts": TIME_STAMP,      // серверная метка времени
  "status": STATUS,      // статус сессии
  "platform": PLATFORM_TYPE // тип платформы пользователя
}
```

Возможные сообщения об ошибках:

Код	Значение
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

3.6. Получение всех данных пользователя

Для восстановления полностью всех данных пользователя за один запрос в Платформе есть вызов, который позволяет упаковать в один ответ разнородные данные, описанные в соответствующих разделах данного документа. Можно запросить как все данные целиком, так и любое подмножество.

Возможными параметрами запроса являются:

- chats (аналогичен вызову /chat/getList);
- roster (аналогичен вызову /roster/getList);
- privacy (аналогичен вызову /privacy/getList);
- requests (аналогичен вызову /roster/requests);
- messages (аналогичен вызову /message/last);
- vcards (аналогичен вызову /vcard/getList).

Запрос:

/execute

Параметры запроса:

Имя	Тип	Описание
get-data	[STRING]	Массив строк из перечисленных выше

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": [
    "chats": [...]           // формат ответа описан в соответствующем запросе
  ]
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

3.7. Изменение имени пользователя

Считается, что имя задаётся в формате Имя + Фамилия, однако поле фамилии может не заполняться.

Запрос:

/changeName

Параметры запроса:

Имя	Тип	Описание
firstName	STRING	Имя пользователя
lastName	STRING	Фамилия пользователя, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

После успешного изменения имени и/или фамилии всем контактам из списка контактов, а также всем другим устройствам этого пользователя, будет разослано следующее уведомление:

```
{
  "type": "102",           // тип события - изменение имени
  "firstName": STRING,    // имя пользователя
  "lastName": STRING,     // фамилия пользователя
  "ts": TIME_STAMP,       // серверное время события
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

3.8. Изменение статуса сессии

Данный запрос применяется для того, чтобы изменить текстовый статус сессии устройства или изменить режим сессии на любой, кроме STATUS.OFFLINE.

Запрос:

/changeStatus

Параметры запроса:

Имя	Тип	Описание
status	STATUS	Статус сессии, по умолчанию STATUS.ONLINE
textStatus	STRING	Статусное сообщение, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

После успешного изменения имени и/или фамилии всем контактам из списка контактов, а также всем другим устройствам этого пользователя, будет разослано следующее уведомление:

```
{
  "type": "101",           // тип уведомления - смена статуса
  "from": JID,            // ваш идентификатор пользователя
  "ts": TIME_STAMP,      // серверная метка времени
  "status": STATUS,      // новый статус сессии
  "textStatus": STRING,  // текстовый статус
  "platform": PLATFORM_TYPE // тип платформы устройства
}
```

Возможные сообщения об ошибках:

Код	Значение
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

3.9. Получение списка устройств

Данный запрос возвращает список всех устройств, присоединённых к данной учётной записи, вместе с их статусами.

Запрос:

/devicesInfo

Параметры запроса:

Отсутствуют.

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": [
    {
      "global_id": GUID,      // идентификатор устройства
      "status": STATUS      // статус устройства
    },
    {...}
  ]
}
```

Возможные сообщения об ошибках:

Код	Значение
3001	Неверный логин пользователя или пароль

4. Работа с сообщениями

4.1. Написание нового сообщения

Отсылает массив сообщений на сервер (одновременно можно передать несколько сообщений). Клиент передаёт для каждого сообщения локальный идентификатор, уникальный для данного устройства. Этот идентификатор потом используется, чтобы соотнести сообщению серверный идентификатор сообщения после успешной доставки сообщения на сервер.

Если каждое сообщение в пакете надо вставить в определённом порядке, надо внутри сообщения указать целочисленный номер в поле `order`, в противном случае сообщения могут быть вставлены в произвольном порядке.

Запрос:

`/message/send`

Параметры запроса:

Имя	Тип	Описание
<code>deviceType</code>	STRING	Имя устройства, с которого отсылается сообщение
<code>messages</code>	[]	Массив структур сообщений следующего формата
{		
<code>to</code>	JID	Идентификатор адресата или группового чата
<code>body</code>	STRING	Текст сообщения
<code>localId</code>	GUID	Локальный идентификатор сообщения
<code>order</code>	INT	Порядковый номер сообщения, опционально
<code>fts</code>	STRING	Строка для полнотекстового поиска, опционально
<code>persistent</code>	bool	Сохранять ли сообщение в базе, опционально, по умолчанию равняется true
}		

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": [
    {
      "sid": SID,           // серверный идентификатор сообщения
      "utc": TIME_STAMP    // серверное время приёма сообщения
      "localId": GUID      // локальный идентификатор сообщения
    },
    {...}
  ]
}
```

После приёма сообщения сервером адресату доставляется следующее уведомление:

```
{
  "type": "401",          // тип события - новое сообщение
  "body": STRING,        // текст сообщения
  "ts": TIME_STAMP,      // серверное время события
  "sid": SID,            // серверный идентификатор события
  "with": JID,           // идентификатор автора сообщения
  "localId": GUID,       // локальный идентификатор сообщения
  "deviceType": STRING   // имя устройства, с которого пришло сообщение
}
```

На остальные устройства автора сообщения также рассылается аналогичное уведомление, с той только разницей, что в поле with указывается не собственный идентификатор пользователя, а идентификатор адресата.

После того, как сервер заканчивает обработку нового сообщения и архивирует его в историю сообщений пользователя, сервер подтверждает автору сообщения доставку на сервер, высылая следующее уведомление:

```
{  
  "type": "405",           // тип события – доставка на сервер  
  "with": JID,            // идентификатор собеседника  
  "sid": SID,             // серверный идентификатор сообщения  
  "ts": TIME_STAMP       // серверное время события  
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Пользователь не является участником чата
4005	Получатель сообщения отсутствует в списке контактов

4.2. Редактирование сообщения

Любое сообщение может быть отредактировано его автором. На сервере в истории сообщений оно помечается особым образом.

Запрос:

/message/edit

Параметры запроса:

Имя	Тип	Описание
sid	SID	Серверный идентификатор сообщения
jid	JID	Идентификатор адресата
body	STRING	Новый текст сообщения

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

После редактирования и автор, и адресат сообщения получают уведомление о редактировании:

```
{
  "type": "402",           // тип события - редактирование сообщения
  "body": STRING,        // текст сообщения
  "ts": TIME_STAMP,      // серверное время события
  "sid": SID,            // серверный идентификатор события
  "from": JID,           // идентификатор автора сообщения
  "localId": GUID,       // локальный идентификатор сообщения
  "deviceType": GUID     // имя устройства, с которого пришло сообщение
}
```

Возможные сообщения об ошибках:

Код	Значение
1007	Сообщение не найдено
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

4.3. Удаление сообщения

Для выбранного собеседника может быть удалено как одно сообщение, так и вся история целиком. В случае полного удаления истории поле `exact-sid` не передаётся.

Сообщение может удаляться как для обеих сторон, так и только для отправителя, это контролируется параметром `forMe` (по умолчанию сообщение удаляется для обеих сторон). В случае группового удаления истории история всегда удаляется только для отправителя.

Запрос:

`/message/remove`

Параметры запроса:

Имя	Тип	Описание
<code>exact-sid</code>	SID	Серверный идентификатор сообщения, опционально
<code>jid</code>	JID	Идентификатор адресата, опционально
<code>forMe</code>	BOOL	Удалять сообщения только для меня, по умолчанию 0
<code>removeAll</code>	BOOL	Удаление всей истории сообщений, по умолчанию 0

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

После успешного удаления сообщения и автор, и адресат получают следующее уведомление:

```
{
  "type": "403",           // тип события - удаление сообщения
  "ts": TIME_STAMP,      // серверное время события
  "sid": SID,            // идентификатор события об удалении
  "with": JID,           // идентификатор собеседника
  "deleted": SID         // серверный идентификатор сообщения
}
```

Если производится массовое удаление, то в поле `deleted` сервер возвращает диапазон идентификаторов сообщений в формате "SID1, SID2".

Возможные сообщения об ошибках:

Код	Значение
1007	Сообщение не найдено
2004	Пропущены параметры <code>exact-sid</code> и <code>jid</code>
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

4.4. Групповое удаление сообщений

Если возникает нужна удалить сразу несколько сообщений, например, отмеченных пользователем, то можно воспользоваться функцией группового удаления. Для этого нужно передать массив идентификаторов сообщений.

Запрос:

/message/removeGroup

Параметры запроса:

Имя	Тип	Описание
forMe	BOOL	Удалять сообщения только для меня, по умолчанию 0
Messages	ARRAY	Массив структур идентификаторов сообщений
{		
sid	SID	Серверный идентификатор сообщения
localId	GUID	Локальный идентификатор сообщения
jid	JID	Идентификатор адресата
}		

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": [
    {
      "sid": SID,           // идентификатор сообщения об удалении
      "deleted": SID,      // идентификатор удалённого сообщения
      "localId": GUID     // локальный идентификатор удалённого сообщения
    }
  ]
}
```

После успешного удаления сообщения и автор, и адресат получают следующее уведомление:

```
{
  "type": "403",           // тип события - удаление сообщения
  "ts": TIME_STAMP,      // серверное время события
  "sid": SID,            // идентификатор события об удалении
  "with": JID,           // идентификатор собеседника
  "deleted": SID         // серверный идентификатор сообщения
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

4.5. Уведомление о наборе текста

Запрос:

/message/notifyTyping

Параметры запроса:

Имя	Тип	Описание
to	JID	Идентификатор адресата
finished	BOOL	Закончен ли набор текста

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

Пользователь, указанный в поле to, получает следующее уведомление о начале набора текста:

```
{
  "type": "407",           // тип события - начало набора текста
  "from": JID,            // идентификатор собеседника
  "ts": TIME_STAMP       // серверное время события
}
```

После окончания набора тип уведомления меняется:

```
{
  "type": "408",           // тип события - окончание набора текста
  "from": JID,            // идентификатор собеседника
  "ts": TIME_STAMP       // серверное время события
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

4.6. Уведомление о доставке

После получения сообщения клиент может подтвердить факт доставки сообщения на клиент, это нужно для того, чтобы отославшая сообщение сторона могла корректно отобразить факт доставки в интерфейсе (т.н. "вторая галочка").

Обратите внимание на то, что аргументом вызова является структура из нескольких пар JID/SID, который позволяет известить о доставке сообщений сразу нескольким адресатам.

Запрос:

/message/notifyDelivered

Параметры запроса:

Имя	Тип	Описание
autoDelivered	BOOL	Выставлять SID без проверки, больше ли он предыдущего значения (по умолчанию 0)
messages		Структура с идентификаторами сообщений
{		
JID	SID	Пара идентификатор адресата: идентификатор сообщения
JID	SID	Пара идентификатор адресата: идентификатор сообщения
}		

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

Собеседнику на все устройства доставляется следующее уведомление:

```
{
  "type": "405",           // тип события - уведомление о доставке
  "with": JID,            // идентификатор собеседника
  "sid": SID               // серверный идентификатор сообщения
  "ts": TIME_STAMP        // серверное время события
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Пользователь не является участником группового чата

4.7. Уведомление о прочтении

После того, как клиент показывает сообщение на экране, он обязан уведомить сервер о его прочтении, используя серверный идентификатор сообщения. Все сообщения с меньшим идентификатором также считаются прочтёнными.

При посылке данного уведомления в групповой чат количество непрочитанных сообщений рассылается только своим собственным устройствам.

Запрос:

/message/notifyRead

Параметры запроса:

Имя	Тип	Описание
jid	JID	Идентификатор адресата или группчата
sid	SID	Идентификатор прочтённого сообщения

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

Собеседнику на все устройства доставляется следующее уведомление:

```
{
  "type": "406",           // тип события - уведомление о прочтении
  "with": JID,            // идентификатор собеседника
  "sid": SID,             // серверный идентификатор сообщения
  "ts": TIME_STAMP,      // серверное время события
  "unreadCount": INT     // количество непрочитанных сообщений, опционально
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Пользователь не является участником группового чата

4.8. Получение истории сообщений

Запрос:

/message/history

Параметры запроса:

Имя	Тип	Описание
talker	JID_ARRAY или STRING	см. комментарий ниже
exact-id	SID	Идентификатор сообщения для получения, опционально
since-id	SID	Стартовый идентификатор сообщения, опционально
to-id	SID	Конечный идентификатор сообщения, опционально
since-time	TIME_STAMP	Начальное время сообщения для поиска, опционально
to-time	TIME_STAMP	Конечное время сообщения для поиска, опционально
fts	STRING	Полнотекстовый поиск по сообщениям
limit	INT	Максимальное количество сообщений в списке, по умолчанию 50
latest	BOOL	Сортировка сообщений в сторону уменьшения SID, опционально
countOnly	BOOL	Вести только подсчёт сообщений, опционально
excludeEdit	BOOL	Исключить из выдачи сообщения типа edited и deleted
team	STRING	Идентификатор команды, опционально

При запросе истории параметр talker может принимать следующие значения:

- "all": STRING – получить историю контакта полностью;
- "chats": STRING – получить только историю групповых чатов, полностью;
- "contacts": STRING – получить только историю контактов, полностью;
- JID_ARRAY – массив из одного или нескольких идентификаторов пользователей, для которых надо считать историю сообщений

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "history": [
      {
        "type": STRING,           // тип сообщения, см. таблицу ниже
        "stime": TIME_STAMP,    // серверное время сообщения
        "sid": SID,             // серверный идентификатор сообщения
        "talker": JID,          // идентификатор собеседника
        "body": STRING,         // тело сообщения
        "reactions": {
          "login": ["😊"],      // JID + все реакции, выставленные для этого сообщения
          ...                  // все JID указываются без доменной части
        },
      },
      {...}
    ]
  }
}
```

В зависимости от типа в описание события может включаться добавочное поле:

Тип сообщения	Описание	Добавочное поле	Значение
message	Входящее сообщение	"direction"	"to"
message	Исходящее сообщение	"direction"	"from"
muc-message	Сообщение из группового чата	"muc-user"	JID участника
muc-added	Участник добавлен в группчат	"muc-user"	JID участника
muc-kicked	Участник выброшен из чата	"muc-user"	JID участника

muc-left	Участник покинул группчат	"muc-user"	JID участника
sms	СМС сообщение		
deleted	Сообщение было удалено		
deleted-range	Несколько сообщений было удалено		
edited	Сообщение было изменено		
edited	Изменённое сообщение было прочитано	"listened"	1
joined	Контакт из телефонной книги присоединился к системе		

Если в запросе указано поле countOnly:1, то ответ будет иметь следующий формат:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "history": [
      {
        "talker": JID,           // идентификатор собеседника
        "count": INT,          // число сообщений
        "lastMsg": {           // последнее сообщение, аналогично предыдущему примеру
          ...
        }
      },
      {...}
    ]
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1002	Указанный пользователь не найден
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Пользователь не является участником группового чата

https://pikabu.ru/story/subbota_nachinaetsya_s_kholodnenkogo_9982390?utm_source=linkshare&utm_medium=sharing

4.9. Получение последнего сообщения

Применяется для того, чтобы проверить, какое последнее сообщение было послано в приватном или групповом чате. Результат всегда отсортирован по серверному времени сообщения в убывающем порядке.

Если оба параметра (`contacts` и `chats`) пропущены, то это эквивалентно выполнению операции над всеми контактами из ростера пользователя и групповыми чатами из списка.

Запрос:

`/message/last`

Параметры запроса:

Имя	Тип	Описание
<code>contacts</code>	<code>JID_ARRAY</code>	Список приватных чатов, опционально
<code>chats</code>	<code>JID_ARRAY</code>	список групповых чатов, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": [{
    // структура для чата с контактом
    {
      "sid" : SID,           // серверный идентификатор сообщения
      "stime": TIME_STAMP,  // серверное время сообщения
      "type": STRING,       // тип сообщения, см. таблицу выше
      "direction": STRING,  // к нам или от нас, см. таблицу выше (опционально)
      "talker": JID,        // идентификатор собеседника
      "nick": STRING,       // имя и фамилия контакта
      "body": STRING,       // тело сообщения (опционально),
      "unreadCount": INT,   // количество непрочитанных сообщений (опционально)
      "unreadReactions": INT // количество непрочитанных реакций (опционально)
    },
    // структура для группового чата
    {
      "sid" : SID,           // серверный идентификатор сообщения
      "stime": TIME_STAMP,  // серверное время сообщения
      "talker": JID,        // идентификатор группчата
      "muc-user": JID,      // автор сообщения
      "type": STRING,       // тип сообщения, см. таблицу выше
      "nick": STRING,       // название группового чата
      "body": STRING,       // тело сообщения
      "unreadCount": INT,   // количество непрочитанных сообщений (опционально)
      "unreadReactions": INT // количество непрочитанных реакций (опционально)
    },
    {...}
  ]}
}
```

Возможные сообщения об ошибках:

Код	Значение
1002	Указанный пользователь не найден
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

4.10. Выставление реакций на сообщение

Для каждого сообщения можно выставить реакцию в виде эмодзи. Для частных чатов можно выставить реакцию от отправителя и/или получателя, для групповых чатов — по одной реакции на каждого участника группового чата. После выставления реакции она заносится для собеседника в список непрочитанных реакций (см. п. 4.12).

Запрос:

/message/reaction

Параметры запроса:

Имя	Тип	Описание
jid	JID	Идентификатор собеседника или группового чата
sid	SID	Идентификатор существующего сообщения
reaction	STRING	Строка с эмодзи или другим символом реакции
remove	BOOL	Флаг удаления реакции, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

Всем собеседникам на все устройства доставляется следующее уведомление:

```
{
  "type": "410", // тип события - выставление реакции
  "from": JID, // идентификатор собеседника, выставившего реакцию
  "sid": SID, // серверный идентификатор сообщения
  "ts": TIME_STAMP, // серверное время события
  "reactions": {
    "login": ["😊"], // JID + все реакции, выставленные для этого сообщения
    ... // все JID указываются без доменной части
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1002	Указанный пользователь не найден
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Пользователь не является участником группового чата

4.11. Отметка реакции прочитанной

Запрос:

/message/readReaction

Параметры запроса:

Имя	Тип	Описание
sid	SID	Идентификатор существующего сообщения

Ответ:

```
{  
  "status": "ok",  
  "statusCode": 200  
}
```

Всем собеседникам на все устройства доставляется следующее уведомление:

```
{  
  "type": "411",           // тип события - отметка реакции прочитанной  
  "from": JID,           // идентификатор собеседника, выставившего реакцию  
  "sid": SID,            // серверный идентификатор сообщения  
  "ts": TIME_STAMP       // серверное время события  
}
```

Возможные сообщения об ошибках:

Код	Значение
3001	Неверный логин пользователя или пароль

4.12. Получение списка непрочитанных реакций

По мере того, как другие пользователи проставляют реакции для сообщений некоторого пользователя, у него накапливается некоторое количество непрочитанных реакций. Их можно получить одним списком полностью, а также предусмотрена возможность фильтрации по собеседнику, JID которого указывается в параметре `jid`.

Запрос:

`/message/unreadReactions`

Параметры запроса:

Имя	Тип	Описание
<code>jid</code>	JID	Идентификатор пользователя, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "talker" : [           // идентификатор пользователя или чата
      {
        "sid": SID,       // идентификатор сообщения, на которое проставлена реакция
        "text": STRING,  // текст реакции
      },
      {...}
    ]
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
3001	Неверный логин пользователя или пароль

5. Работа со списком пользователей

5.1. Получение списка пользователей

Запрос:

/roster/getList

Параметры запроса:

Отсутствуют.

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": [
    {
      "jid": JID, // идентификатор пользователя
      "flags": CONTACT_FLAGS, // набор флагов пользователя
      "delivered": SID, // номер последнего доставленного события
      "markRead": SID, // номер последнего прочитанного события
      "remoteMarkRead": SID, // то же, но для того пользователя
      "remoteMarkReadTime": TIME_STAMP,
      "logoutTime": TIME_STAMP, // время последнего выхода
      "firstName": STRING, // имя пользователя
      "lastName": STRING, // фамилия пользователя
      "group": STRING, // группа в списке, если установлена
      "status": STATUS, // статус
      "textStatus": STRING, // текстовый статус
      "platform": PLATFORM_TYPE, // тип устройства
      "unreadCount": INT // число непрочитанных сообщений, опционально
    },
    {...}
  ],
}
```

Возможные сообщения об ошибках:

Код	Значение
3001	Неверный логин пользователя или пароль

5.2. Изменение информации о пользователе в списке

Для любого контакта в списке доступно хранение трёх собственных атрибутов: имя и фамилия, как их видит клиент, а не сам пользователь, а также набор флагов для данного контакта.

Запрос:

/roster/edit

Параметры запроса:

Имя	Тип	Описание
jid	JID	Идентификатор контакта в списке
firstName	STRING	Имя пользователя, опционально
lastName	STRING	Фамилия пользователя, опционально
group	STRING	Группа в списке, опционально
flags	CONTACT_FLAGS	Набор флагов контакта

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "jid": JID,           // идентификатор пользователя
    "firstName": STRING, // имя пользователя
    "lastName": STRING,  // фамилия пользователя
    "flags": CONTACT_FLAGS // набор флагов контакта
  }
}
```

После успешного изменения информации все остальные устройства этого пользователя получают следующее уведомление:

```
{
  "type": "306",           // тип события – изменение контакта
  "from": JID,            // ваш собственный идентификатор
  "jid": JID,             // идентификатор контакта в списке
  "updated": {            // список произведённых изменений
    "firstName": STRING,  // имя контакта
    "lastName": STRING,   // фамилия контакта
    "group": STRING,      // группа в списке
    "flags": CONTACT_FLAGS // набор флагов контакта
  },
  "ts": TIME_STAMP,      // серверное время события
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4005	Пользователь не входит в список контактов

5.3. Удаление пользователя из списка контактов

После удаления пользователя из списка контактов клиент теряет возможность посылать ему сообщения, а также отслеживать изменение статуса.

Запрос:

/roster/remove

Параметры запроса:

Имя	Тип	Описание
jid	JID	Идентификатор контакта в списке
removeHistory	BOOL	Удалять ли переписку с контактом, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

После успешного удаления на все устройства пользователя, а также удалённому пользователю направляется следующее уведомление:

```
{
  "type": "304",           // тип события - удаление пользователя из списка
  "jid": JID,             // идентификатор собеседника
  "ts": TIME_STAMP       // серверное время
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4005	Пользователь не входит в список контактов

5.4. Направление запроса о дружбе

Чтобы добавить произвольный контакт в свой список, клиент должен отослать запрос на дружбу. Впоследствии он может быть подтверждён или отклонён.

Данное действие регулируется настройкой `enableFriends` в файле `/var/www/ommg3_nodejs/config.json` на сервере. Если эта настройка установлена в `false`, то подтверждения запроса на дружбу не требуется, а контакт автоматически добавляется в список контактов. Если же настройка выставлена в `true`, то требуется подтвердить или отклонить запрос.

Запрос:

`/roster/invite`

Параметры запроса:

Имя	Тип	Описание
<code>jid</code>	JID	Идентификатор добавляемого контакта
<code>firstName</code>	STRING	Имя пользователя
<code>lastName</code>	STRING	Фамилия пользователя, опционально
<code>team</code>	STRING	Команда, в которую пригласить, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

После успешной обработки запроса сервер передаёт указанному пользователю уведомление о том, что ему прислан запрос на дружбу:

```
{
  "type": "301", // тип уведомления - запрос на дружбу
  "ts": TIME_STAMP, // серверное время события
  "user": { // контакт, приславший запрос на дружбу
    "firstName": STRING, // имя контакта
    "lastName": STRING, // фамилия контакта
    "jid": JID, // идентификатор контакта
    "avatar": STRING // хеш аватара
  }
}
```

Отправитель запроса получает точно такое же сообщение, но в нем в полях `firstName` и `lastName` указываются имя и фамилия того, кому прислан запрос на дружбу.

Возможные сообщения об ошибках:

Код	Значение
1002	Такой пользователь не найден
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4005	Пользователь уже входит в ваш список контактов

5.5. Принятие запроса о дружбе

Клиент получил уведомление о запросе на дружбу, показал его пользователю и получил от него подтверждение того, что он хочет принять его в список контактов.

Запрос:

/roster/inviteAffirmative

Параметры запроса:

Имя	Тип	Описание
jid	JID	Идентификатор добавляемого контакта
team	STRING	Команда, в которую пригласить, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

После этого обе стороны получают уведомление о добавлении в список контактов:

```
{
  "type": "302", // тип уведомления - добавление в ростер
  "ts": TIME_STAMP, // серверное время события
  "user": { // описание нового контакта в ростере
    "jid": JID, // идентификатор контакта
    "flags": CONTACT_FLAGS, // набор флагов контакта
    "delivered": SID, // последнее доставленное на сервер сообщение
    "markRead": SID, // последнее прочитанное мной сообщение
    "remoteMarkRead": SID, // последнее прочитанное контактом сообщение
    "remoteMarkReadTime": TIME_STAMP, // время, когда контакт прочитал моё сообщение
    "firstName": STRING, // имя контакта
    "lastName": STRING, // фамилия контакта
    "logoutTime": TIME_STAMP, // время последнего выхода контакта в оффлайн
    "avatar": STRING, // хеш аватара контакта
    "status": STATUS, // статус контакта
    "textStatus": STRING, // строковый статус
    "platform": PLATFORM_TYPE // код платформы активной сессии
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1002	Такой пользователь не найден
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

5.6. Отклонение запроса о дружбе

Если пользователь отвергает запрос о дружбе, клиент обязан уведомить об этом сервер.

Запрос:

/roster/inviteNegative

Параметры запроса:

Имя	Тип	Описание
jid	JID	Идентификатор контакта
team	STRING	Команда, в которую пригласить, опционально

Ответ:

```
{  
  "status": "ok",  
  "statusCode": 200  
}
```

После успешной обработки запроса обе стороны получают следующее уведомление:

```
{  
  "type": "305",           // тип события - отклонённый запрос о дружбе  
  "with": JID,           // идентификатор контакта  
  "ts": TIME_STAMP       // серверное время события  
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4002	Запрос на дружбу не найден

5.7. Получение списка запросов на дружбу

Запрос:

/roster/requests

Параметры запроса:

Отсутствуют.

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    [
      "jid": JID,           // идентификатор пользователя
      "firstName": STRING, // имя пользователя
      "lastName": STRING,  // фамилия пользователя
      "avatar": STRING,    // хеш аватара контакта
      "type": STRING,      // тип запроса на дружбу, см. ниже
      "created": TIME_STAMP // время создания запроса
    ]
    ...
  ]
}
```

Тип запроса на дружбу	Описание
1 - TYPE_IN	Пользователь хочет подружиться со мной
2 - TYPE_OUT	Я послал пользователю запрос на дружбу
3 - TYPE_OUT_DECLINED	Пользователь отклонил запрос на дружбу
4 - TYPE_IN_IGNORED	Пользователь добавил мою заявку в игнор
5 - TYPE_IN_POSTPONED	Отложенный входящий запрос на дружбу
6 - TYPE_OUT_POSTPONED	Отложенный исходящий запрос на дружбу
7 - TYPE_IN_DECLINED	Я отклонил запрос на дружбу

Возможные сообщения об ошибках:

Код	Значение
3001	Неверный логин пользователя или пароль

6. СПИСКИ ВИДИМОСТИ

Списки видимости применяются для того, чтобы изменять свою видимость для других контактов из своего списка контактов. Возможны два варианта:

- включение контакта в список "всегда видимых", тогда этот контакт будет видеть ваш статус даже в состоянии INVISIBLE;
- включение контакта в список "никогда не видимых", тогда этот контакт будет видеть вас в состоянии OFFLINE, даже если вы сейчас находитесь в сети.

6.1. Получение списков видимости

Запрос:

/privacy/getList

Параметры запроса:

Отсутствуют.

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    [
      {
        "jid": JID,           // идентификатор пользователя
        "privacy": "visible", // контакт "всегда видимый"
        "firstName": STRING, // имя пользователя
        "lastName": STRING,  // фамилия пользователя
        "avatar": STRING,    // хеш аватара контакта
      },
      {
        "jid": JID,           // идентификатор пользователя
        "privacy": "ignore"  // контакт "никогда не видимый"
        "firstName": STRING, // имя пользователя
        "lastName": STRING,  // фамилия пользователя
        "avatar": STRING,    // хеш аватара контакта
      }
    ]
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
3001	Неверный логин пользователя или пароль

6.2. Изменение списков видимости

Запрос:

/privacy/manage

Параметры запроса:

Имя	Тип	Описание
jid	JID	Идентификатор контакта
privacy	STRING	Строка, см. примечание ниже

Строковый параметр "privacy" может приобретать три значения:

- "visible" для всегда видимых
- "ignore" для всегда не видимых
- "none" для исключения контакта из списков видимости.

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {}
}
```

В ответ на успешно изменённый список видимости сервер рассылает всем устройствам следующее уведомление:

```
{
  "type": "104",
  "ts": TIME_STAMP,           // время совершения операции
  "user": {
    "jid": JID,               // идентификатор пользователя
    "firstName": STRING,     // имя пользователя
    "lastName": STRING,      // фамилия пользователя
    "avatar": STRING,        // хеш аватара контакта
    "privacy": STRING,       // изначальное значение параметра privacy
    "status": STATUS,        // статус
    "textStatus": STRING,    // текстовый статус
    "platform": PLATFORM_TYPE // тип устройства
    "logoutTime": TIME_STAMP, // время последнего выхода
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4001	Нельзя включать себя в списки видимости

7. Работа с личной карточкой

7.1. Получение личной карточки пользователя

Запрос:

/vcard/get

Параметры запроса:

Имя	Тип	Описание
jid	JID	Идентификатор пользователя

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "jid": STRING,           // идентификатор пользователя
    "vcard": {
      структура данных в формате JSON, определяется клиентом
    }
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

7.2. Считывание списка личных данных

Если параметр `userJids` пропущен, то это эквивалентно выполнению операции над всеми контактами из ростера пользователя.

Запрос:

`/vcard/getList`

Параметры запроса:

Имя	Тип	Описание
<code>userJids</code>	<code>JID_ARRAY</code>	Массив идентификаторов пользователей, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    // массив структур
    [
      { "jid": JID, // идентификатор пользователя
        "firstName": STRING, // имя
        "lastName": STRING, // фамилия
        "avatar": GUID // хэш аватара
      },
      {...}
    ]
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

7.3. Запись собственной личной карточки

Запрос:

/vcard/set

Параметры запроса:

Имя	Тип	Описание
vcard	STRING	Структура данных в формате JSON

Ответ:

```
{  
  "status": "ok",  
  "statusCode": 200  
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

8. Групповые чаты

Групповой чат – это объект, разделяющий общую историю между своим списком участников.

Для реализации группового чата создаётся фейковый пользователь с идентификатором следующего вида: `{GUID}@conference.yourdomain.com`. По наличию слова `@conference` можно судить о том, что контакт является групповым чатом. Соответственно, история группового чата является историей этого фейкового пользователя.

Список участников группового чата задаётся при создании и впоследствии может быть изменён. Каждому участнику группового чата ставится в соответствие так называемая роль, определяющая его возможности в этом чате. Ролей предусмотрено пять:

- `owner` (владелец) – пользователь, создавший групповой чат, имеет права делать любую операцию. Впоследствии может передать роль `owner` другому пользователю;
- `admin` (администратор) – может делать любую операцию с чатом, кроме назначения владельца;
- `moderator` (модератор) – может менять информацию о чате и писать в модерлируемые групповые чаты;
- `participant` (участник) – может писать в немодерируемый групповой чат;
- `visitor` (визитёр) – может только читать информацию в групповом чате.

Для каждого чата предусмотрено несколько свойств, которые определяют режим использования чата:

- `модерируемый`: изменение свойств чата и добавление новых сообщений в чат доступно не всем участникам, а только пользователям с ролью не ниже модератора. По умолчанию отключено. В немодерируемом чате все участники являются владельцами, т.е. могут совершать любые операции.
- `приватный`: чат доступен только по приглашению в него другим пользователем или добавлением пользователем с правами администратора или владельца. По умолчанию все чаты создаются публичными. Если чат является модерлируемым, то зашедший в публичный модерлируемый чат новый пользователь по умолчанию получает права визитёра, т.е. может лишь просматривать сообщения.
- `отдельная история`: история чата не копируется в истории его участников (по умолчанию выключена). Если для небольших чатов копирование истории участникам помогает значительно оптимизировать работу с историей, гарантируя полное её получение одним запросом, то для каналов или чатов с большим количеством участников (более 500) рекомендуется ведение истории только для фейкового контакта чата.

8.1. Создание группового чата

При создании группового чата сервер создаёт новый GUID и нового фейкового пользователя со случайным идентификатором вида `{GUID}@conference.yourdomain.com`. При необходимости между GUID и `@` может быть вставлено поле `suffix`. Владелец этого группового чата становится создавший его пользователь. При необходимости создания приватного или модерлируемого группового чата необходимо указать параметр `flags`.

При создании немодерируемого чата все участники, указанные в параметре `userJids`, также становятся владельцами этого чата, в противном случае они получают статус участников (`participant`).

Аналогично созданию сообщения (/message/send), при создании группового чата может быть указан локальный идентификатор события, чтобы остальные устройства пользователя могли отличать групповые чаты, созданные данным устройством. Если клиент рассчитан на работу с одним устройством, данный параметр может быть пропущен.

Запрос:

/chat/create

Параметры запроса:

Имя	Тип	Описание
topic	STRING	Тема чата
userJids	JID_ARRAY	Идентификаторы участников чата, опционально
flags	CHAT_FLAGS	Набор свойств чата, опционально
localId	GUID	Локальный идентификатор события, опционально
team	STRING	Идентификатор команды, опционально
suffix	STRING	Суффикс идентификатора чата, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "chatJid": JID // идентификатор созданного чата
  }
}
```

После создания чата всем участникам рассылается уведомление следующего формата:

```
{
  "type": "201", // тип события – создание чата
  "from": JID, // идентификатор создателя чата
  "to": JID, // идентификатор участника чата
  "ts": TIME_STAMP, // серверное время события
  "chat": {
    "jid": JID, // идентификатор чата
    "topic": STRING, // тема чата
    "flags": CHAT_FLAGS, // набор флагов группового чата
    "chatStatus": CHAT_STATUS, // статус участника в чате
    "markRead": SID, // последний прочитанный ID сообщения
    "created": TIME_STAMP, // дата создания чата
    "members": JID_ARRAY, // список участников
    "localId": GUID // опционально, если был указан в запросе на создание
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль

8.2. Уничтожение группового чата

Удаление модерлируемого группового чата может производиться только владельцем, в то время как немодерируемого – любым участником.

Запрос:

/chat/destroy

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор чата

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

После успешного уничтожения группового чата всем участникам рассылается сообщение следующего формата:

```
{
  "type": "207",                // тип события – уничтожение чата
  "from": SYSTEM_JID,
  "chatJid": JID,              // идентификатор уничтоженного чата
  "to": JID,                   // идентификатор участника чата
  "ts": TIME_STAMP             // время события
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Вы не являетесь участником этого группчата
4009	У вас не хватает привилегий для выполнения операции

8.3. Редактирование группового чата

Редактирование темы модерлируемого группового чата может осуществляться только участником с правами не ниже модератора. Изменение флага pinned и других произвольно назначаемых клиентом флагов может производиться любым участником группового чата.

Запрос:

/chat/edit

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор чата
topic	STRING	Новая тема чата, опционально
flags	CONTACT_FLAGS	Набор опций участника чата, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

Если одно или оба поля были изменены тема чата, то всем участникам чата направляется уведомление следующего формата:

```
{
  "type": "206",
  "from": JID, // идентификатор пользователя, произведшего изменение
  "chatJid": JID, // идентификатор чата
  "updated": {
    "topic": STRING, // новая тема чата
    "flags": CHAT_FLAGS // новый набор флагов чата
  },
  "ts": TIME_STAMP // серверное время события
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Вы не являетесь участником этого группчата
4009	У вас не хватает привилегий для выполнения операции

8.4. Получение списка групповых чатов для пользователя

Запрос:

/chat/getList

Параметры запроса:

Отсутствуют.

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": { [
    {
      "jid": JID, // идентификатор чата
      "topic": STRING, // тема чата
      "flags": CONTACT_FLAGS, // пользовательские настройки для чата
      "chatStatus": CHAT_STATUS, // статус пользователя в чате
      "chatType": CHAT_FLAGS, // флаги, с которыми был создан чат
      "created": TIME_STAMP, // время создания чата
      "markRead": SID, // последнее прочтённое сообщение из чата
      "unreadCount": INT, // число непрочитанных сообщений
      "members": [ // список участников
        {
          "jid": JID, // идентификатор участника чата
          "logoutTime": TIME_STAMP, // время последнего нахождения в сети
          "firstName": STRING, // имя
          "lastName": STRING, // фамилия
          "role": CHAT_ROLE, // роль пользователя
          "status": TYPE_STATUS, // статус пользователя, если он в сети
          "textStatus": STRING, // текстовый статус
          "platform": PLATFORM_TYPE // текущая платформа
        }
      ],
      "team": STRING, // идентификатор команды, опционально
      "type": INT // тип командного чата, опционально
    },
    {...}
  ]
}
```

Возможные сообщения об ошибках:

Код	Значение
3001	Неверный логин пользователя или пароль

8.5. Получение списка участников группового чата

Запрос:

/chat/members

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор чата

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": [
    {
      "jid": JID,                // идентификатор участника чата
      "avatar": STRING,         // хеш аватара
      "logoutTime": TIME_STAMP, // время последнего нахождения в сети
      "firstName": STRING,      // имя
      "lastName": STRING,       // фамилия
      "role": CHAT_ROLE,        // роль пользователя
      "status": TYPE_STATUS,    // статус пользователя, если он в сети
      "textStatus": STRING,     // текстовый статус
      "platform": PLATFORM_TYPE // текущая платформа
    },
    {...}
  ]
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Вы не являетесь участником этого группчата

8.6. Изменение списка участников группового чата

Этот вызов требует от участника статуса администратора или создателя чата для модерлируемых групповых чатов.

Если необходимо изменить роль уже существующих участников, то их надо перечислить в списке `addJids`, система автоматически поймёт, что добавлять их заново не нужно.

Запрос:

`/chat/modify`

Параметры запроса:

Имя	Тип	Описание
<code>chatJid</code>	JID	Идентификатор чата
<code>addJids</code>	JID_ARRAY	Добавляемые участники чата, опционально
<code>role</code>	CHAT_ROLE	Роль добавляемых участников, опционально
<code>removeJids</code>	JID_ARRAY	Удаляемые участники чата, опционально

Ответ:

```
{
  "status": "ok",
  "statusCode": 200
}
```

Если в запросе был указан параметр `addJids`, и это привело к добавлению нового пользователя, то всем пользователям группчата рассылается уведомление о присоединении к группчату:

```
{
  "type": "202",
  "from": JID, // пользователь, произведший изменение
  "chatJid": JID, // идентификатор чата
  "role": CHAT_ROLE, // роль пользователя
  "flags": CONTACT_FLAGS, // набор опций участника чата, опционально
  "team_id": STRING, // идентификатор команды, опционально
  "members": [
    {
      "jid": JID, // идентификатор нового пользователя
      "avatar": STRING, // хеш аватара
      "logoutTime": 0, // время последнего захода
      "firstName": STRING, // имя
      "lastName": STRING, // фамилия
      "status": CHAT_STATUS, // статус пользователя в чате
      "textStatus": STRING, // строковый статус
      "platform": PLATFORM_TYPE // текущая платформа
    }
  ],
  "ts": 1624524204415
}
```

Если же был указан параметр `removeJids`, и в результате были удалены один или несколько участников, то им и оставшимся участникам группчата рассылается уведомление об исключении пользователя из группчата:

```
{
  "type": "203",
  "from": JID, // пользователь, произведший изменение
  "chatJid": JID, // идентификатор чата
  "userJids": [
    JID, JID, JID // список идентификаторов удалённых пользователей
  ],
  "ts": 1624524204431
}
```

}

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр или указана неверная роль
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Вы или ни один из указанных пользователей не являетесь участником этого группчата
4004	Все указанные пользователи уже являются участниками этого группчата или имеют указанную роль
4009	У вас не хватает привилегий для выполнения операции

8.7. Изменение статуса в чате

Запрос:

/chat/online – для подключения к чату
/chat/offline – для отключения от чата

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор чата

Ответ:

```
{  
  "status": "ok",  
  "statusCode": 200  
}
```

После смены статуса в чате на остальные устройства этого же пользователя рассылается следующее уведомление:

```
{  
  "type": "204", // 204 для офлайн или 205 для онлайн  
  "from": JID, // собственный идентификатор пользователя  
  "chatJid": JID, // идентификатор чата  
  "status": CHAT_STATUS, // статус пользователя в чате  
  "textStatus": STRING, // текстовый статус пользователя  
  "platform": PLATFORM_TYPE, // текущая платформа пользователя  
  "ts": TIME_STAMP // серверное время события  
}
```

Остальные участники чата получают другое уведомление;

```
{  
  "type": "204", // 204 для офлайн или 205 для онлайн  
  "from": JID, // собственный идентификатор пользователя  
  "to": JID, // идентификатор получателя  
  "chatJid": JID, // идентификатор чата  
  "ts": TIME_STAMP, // серверное время события  
  "chat": {  
    "topic": STRING, // тема чата  
    "members": JID_ARRAY // список участников чата в онлайн  
  }  
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
4003	Вы не являетесь участником этого группчата

9. Поддержка голосовых и видеозвонков

9.1. Инициализация приватного звонка (peer-2-peer)

Запрос:
/call/p2p

Параметры запроса:

Имя	Тип	Описание
to	JID	Идентификатор вызываемого абонента
format	STRING	Тип звонка

Поле format может принимать значения "audio" или "video".

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "user": STRING,           // имя голосовой или видео сессии
    "password": STRING,      // пароль для установления сессии
    "turns": [STRING]        // адреса TURN серверов
  }
}
```

Вызываемый абонент получает следующее уведомление:

```
{
  "type": "105",             // инициализация звонка
  "from": JID,               // идентификатор пользователя, начавшего звонок
  "format": STRING,          // параметр format из вызова
  "ts": TIME_STAMP,          // серверное время события
  "data": {
    "user": STRING,           // имя голосовой или видео сессии
    "password": STRING,      // пароль для установления сессии
    "turns": [STRING]        // адреса TURN серверов
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки

9.2. Инициализация группового звонка

Запрос:

/call/startGroup

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор группового чата
format	STRING	Тип звонка

Поле format может принимать значения "audio" или "video".

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "type": "group_call_created",
    "group_id": JID          // идентификатор группового чата
    "owner_id": JID        // владелец звонка
  }
}
```

Все участники группового чата получают следующее уведомление:

```
{
  "type": "106",           // создание группового звонка
  "from": JID,            // идентификатор пользователя, начавшего звонок
  "format": STRING,       // параметр format из вызова
  "ts": TIME_STAMP,       // серверное время события
  "data": {
    "type": "group_call_created",
    "group_id": JID,       // идентификатор группового чата
    "owner_id": JID       // владелец звонка
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1011	Звонок внутри этого группового чата уже активирован
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки

9.3. Присоединение к групповому звонку

Запрос:

/call/join

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор группового чата

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "type": "join",
    "group_id": JID,           // идентификатор группового чата
    "user_id": JID           // идентификатор пользователя
  }
}
```

Все участники чата получают следующее асинхронное уведомление после того, как сервер отработает процедуру присоединения:

```
{
  "type": "107",             // присоединение к групповому звонку
  "from": JID,              // идентификатор пользователя, начавшего звонок
  "format": STRING,         // параметр format из вызова
  "ts": TIME_STAMP,        // серверное время события
  "data": {
    "publishers": JID_ARRAY // массив пользователей, присоединившихся к звонку
  }
}
```

Возможные сообщения об ошибках:

Код	ЗначениеGROUP_CALL_ALREADY_STARTED
1009	Групповой вызов не найден
1010	Групповой вызов существует, но не активизирован
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки
4003	Пользователь не является участником группового чата

9.4. Поддержка присутствия в групповом звонке

Сервер групповых звонков требует, чтобы после присоединения к звонку клиент раз в 3-4 секунды отправлял пинг-пакет на сервер для подтверждения своего присутствия. Если такой пакет не будет отослан, то через 6 секунд сервер выбросит участника из звонка автоматически.

Запрос:

/call/presence

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор группового чата

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "event_type": "presence",
    "group_id": JID,          // идентификатор группового чата
    "user_id": JID          // идентификатор пользователя
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1009	Групповой вызов не найден
1010	Групповой вызов существует, но не активизирован
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки
4003	Пользователь не является участником группового чата
4010	Пользователь ещё не присоединился к звонку
4011	Пользователь в процессе незавершённого присоединения к звонку

9.5. Публикация оффера для SDP в групповом чате

Запрос:

/call/sendSdp

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор группового чата
sdp	STRING	Строка с описанием SDP-оффера
audio	BOOL	Требуется аудио-связь
video	BOOL	Требуется видео-связь

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "event_type": "publish_answer",
    "group_id": JID,           // идентификатор группового чата
    "user_id": JID,          // идентификатор пользователя
    "meta_data": {
      "data": {              // параметры WebRTC
        {
          "audio_codec": "opus",
          "jsep": {
            "sdp": <SDP>,
            "type": "answer"
          },
          "streams": [
            {
              "disabled": true,
              "mid": "0",
              "mindex": 0,
              "type": "audio"
            }
          ]
        }
      }
    }
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1009	Групповой вызов не найден
1010	Групповой вызов существует, но не активизирован
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки
4003	Пользователь не является участником группового чата
4010	Пользователь ещё не присоединился к звонку
4011	Пользователь в процессе незавершённого присоединения к звонку
4012	Неверное содержание оффера SDP

9.6. Публикация ICE-кандидата в групповом чате

Запрос:

/call/sendIce

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор группового чата
candidate	STRING	Строка с описанием ICE-кандидата в формате SDP
sdpMid	STRING	Идентификатор медиа-стрима
sdpMLineIndex	INT	Индекс начала стрима в оффере
publisher	JID	Опционально, идентификатор пользователя, на которого ранее был сделан subscribe

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "event_type": "publish_ice_candidate_ack",
    "group_id": JID,      // идентификатор группового чата
    "user_id": JID      // идентификатор пользователя
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1009	Групповой вызов не найден
1010	Групповой вызов существует, но не активизирован
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки
4003	Пользователь не является участником группового чата
4010	Пользователь ещё не присоединился к звонку
4011	Пользователь в процессе незавершённого присоединения к звонку
4014	Неверный кандидат
4015	Пользователь пытается опубликовать кандидата до выполнения подписки

9.7. Подписка на публикации в групповом чате

Запрос:

/call/subscribe

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор группового чата
publisher	JID	Идентификатор пользователя, на публикации которого мы подписываемся

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "event_type": "subscribe_offer",
    "group_id": JID,           // идентификатор группового чата
    "user_id": JID,          // идентификатор подписывающегося пользователя
    "publisher_id": JID,     // идентификатор подписываемого пользователя
    "jsep": {
      "sdp": <SDP>,         // SDP-оффер для данного звонка
      "type": "offer"
    }
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1009	Групповой вызов не найден
1010	Групповой вызов существует, но не активизирован
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки
4003	Пользователь не является участником группового чата
4010	Пользователь ещё не присоединился к звонку
4011	Пользователь в процессе незавершённого присоединения к звонку
4016	В параметре publisher указан не присоединившийся пользователь
4017	В параметре publisher указан пользователь без подписки

9.8. Посылка SDP подписчика

Запрос:

/call/subscribeSdp

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор группового чата
publisher	JID	Идентификатор пользователя, на публикации которого мы подписываемся
sdp	STRING	Строка с описанием SDP-оффера
audio	BOOL	Требуется аудио-связь
video	BOOL	Требуется видео-связь

Ответ:

```
{  
  "status": "ok",  
  "statusCode": 200  
}
```

Возможные сообщения об ошибках:

Код	Значение
1009	Групповой вызов не найден
1010	Групповой вызов существует, но не активизирован
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки
4003	Пользователь не является участником группового чата
4010	Пользователь ещё не присоединился к звонку
4011	Пользователь в процессе незавершённого присоединения к звонку
4013	Неверный ответ от сервера звонков

9.9. Выход из группового звонка

Запрос:

/call/leave

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор группового чата

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "event_type": "leave_call",
    "group_id": JID,           // идентификатор группового чата
    "user_id": JID           // идентификатор пользователя, покинувшего звонок
  }
}
```

Все участники группового чата получают следующее уведомление:

```
{
  "type": "108",              // пользователь покинул групповой звонок
  "from": JID,                // идентификатор пользователя, покинувшего звонок
  "ts": TIME_STAMP,          // серверное время события
  "data": {
    "type": "leave_call",
    "group_id": JID,          // идентификатор группового чата
    "user_id": JID           // идентификатор пользователя
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1009	Групповой вызов не найден
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки
4010	Пользователь не является участником группового чата

9.10. Окончание группового звонка

Запрос:

/call/endGroup

Параметры запроса:

Имя	Тип	Описание
chatJid	JID	Идентификатор группового чата

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "event_type": "end_call",
    "group_id": JID,           // идентификатор группового чата
    "owner_id": JID          // владелец звонка
  }
}
```

Все участники группового чата получают следующее уведомление:

```
{
  "type": "109",              // окончание группового звонка
  "from": JID,                // идентификатор пользователя, закончившего звонок
  "ts": TIME_STAMP,          // серверное время события
  "data": {
    "type": "end_call",
    "group_id": JID,         // идентификатор группового чата
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
1009	Групповой вызов не найден
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
3001	Неверный логин пользователя или пароль
3002	Ошибка сервиса звонков, описание в тексте ошибки
4003	Пользователь не является участником группового чата

10. Работа с командами.

10.1. Создание команды.

Запрос:

/team/create

Параметры запроса:

Имя	Тип	Описание
name	STRING	Имя команды
id	STRING	Идентификатор команды, только английские буквы и _
date	SID	Время создания команды в формате SID
description	STRING	Описание команды
photo	URL_STRING	URL закачанной картинке, опционально
members	JID_ARRAY	Идентификаторы участников группы

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "id": STRING,
    "name": STRING,
    "date": SID,
    "photo": URL_STRING,
    "members": JID_ARRAY
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
2010	Команда с таким идентификатором уже существует
3001	Неверный логин пользователя или пароль

10.2. Уничтожение команды

Запрос:

/team/destroy

Параметры запроса:

Имя	Тип	Описание
id	STRING	Идентификатор команды

Ответ:

```
{  
  "status": "ok",  
  "statusCode": 200  
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
2010	Неверный идентификатор команды
3001	Неверный логин пользователя или пароль

10.3. Получение информации о команде

Запрос:

/team/info

Параметры запроса:

Имя	Тип	Описание
id	STRING	Идентификатор команды

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "data": {
    "id": STRING,
    "name": STRING,
    "description": STRING,
    "photo": URL_STRING,
    "members": JID_ARRAY
  }
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
2010	Неверный идентификатор команды
3001	Неверный логин пользователя или пароль

10.4. Присоединение себя к команде

Запрос:

/team/join

Параметры запроса:

Имя	Тип	Описание
id	STRING	Идентификатор команды

Ответ:

```
{  
  "status": "ok",  
  "statusCode": 200  
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
2010	Неверный идентификатор команды
3001	Неверный логин пользователя или пароль

10.5. Получение списка своих команд

Запрос:

/team/list

Параметры запроса:

Отсутствуют.

Ответ:

```
{
  "status": "ok",
  "statusCode": 200,
  "list": [
    idstr: STRING,          // идентификатор команды
    ...
  ]
}
```

Возможные сообщения об ошибках:

Код	Значение
3001	Неверный логин пользователя или пароль

10.6. Получение списка участников команды

Запрос:

/team/members

Параметры запроса:

Имя	Тип	Описание
id	STRING	Идентификатор команды

Ответ:

```
{  
  "status": "ok",  
  "statusCode": 200,  
  "members": JID_ARRAY  
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
2010	Неверный идентификатор команды
3001	Неверный логин пользователя или пароль

10.7. Изменение команды

Запрос:

/team/modify

Параметры запроса:

Имя	Тип	Описание
id	STRING	Идентификатор команды
name	STRING	Название команды
description	STRING	Описание команды
remove	JID_ARRAY	Список идентификаторов для удаления из команды
admin	JID_ARRAY	Список администраторов команды
members	JID_ARRAY	список простых участников команды

Все параметры запроса, кроме id, являются необязательными.

Ответ:

```
{  
  "status": "ok",  
  "statusCode": 200  
}
```

Возможные сообщения об ошибках:

Код	Значение
2004	Пропущен обязательный параметр
2007	Неверный тип параметра
2010	Неверный идентификатор команды
3001	Неверный логин пользователя или пароль

Приложение 1. Коды ошибок, используемые в API

Код ошибки	Значение
1001	Модуль не найден (невозможно загрузить код на сервере)
1002	Пользователь не найден
1003	Страница не найдена
1004	Данные о пользователе не найдены
1005	Указанный групповой чат не существует
1006	Указанное устройство не найдено
1007	Сообщение не найдено
1008	Стикер не найден
1009	Групповой вызов не найден
1010	Групповой вызов существует, но не активизирован
2001	Неверный тип данных (для списков приватности)
2002	Неверный тип акции (для списков приватности)
2003	Неверный статус пользователя
2004	Требуемый параметр отсутствует или пропущен
2005	Неверный тип платформы устройства
2006	Неверная маска флагов
2007	Неверный тип параметра
2008	Неверный код регистрации
2009	Пропущен код регистрации
2010	Неверный идентификатор команды
3001	Неверный логин или пароль
4001	Попытка добавить в друзья самого себя
4002	Попытка добавить в друзья несуществующий контакт
4003	Неверный или несуществующий пользователь группового чата
4004	Ошибка приглашения в групповой чат
4005	Указанный пользователь отсутствует в списке контактов
4006	Запрошенное действие запрещено
4007	Платформа не поддерживает рассылку пуш-нотификаций
4008	Пользователь не авторизован
4009	Недостаточно прав в групповом чате для выполнения операции
4010	Пользователь не является участником группового чата
4011	Пользователь в процессе незавершённого присоединения к звонку
4012	Неверное содержание оффера SDP
4013	Неверный ответ от сервера звонков
4014	Неверный кандидат
4015	Пользователь пытается опубликовать кандидата до выполнения подписки
4016	В параметре publisher указан не присоединившийся пользователь
4017	В параметре publisher указан пользователь без подписки

Приложение 2. Расширенный формат записи событий в истории.

Простые текстовые сообщения записываются в историю как обычный текст. Для того, чтобы вставить в текст сообщения файл или другой объект, используется расширенный формат записи. Расширенный формат состоит из открывающего тега `[extra ver="1" type="xxxx" len="NNN"]`, где `len` – длина последующего блока данных в байтах, собственно блока данных в формате JSON и закрывающего тега `[/extra]`. Содержимое блока данных варьируется в зависимости от поля `type`. Ниже перечислены все возможные варианты расширенного формата:

- стикер (`gallery_id` - номер коллекции стикеров, `sticker_id` - номер стикера в коллекции):
`[extra ver="1" type="file" len="124"]`
`{"data":[{"name":"sticker.png", "filesize":49430, "link":"stickers\s\1\6\256", "gallery_id":1, "sticker_id":6}]}`
`[/extra]`
- ссылка на файл с картинкой:
`[extra ver="1" type="file" len="217"] {"data":[{"name":"i1.jpg", "filesize":37111, "domain":"files.yourdomain.com","link":"000000000000001000010d4001600163730303132343831363140666c6f6469756d2e7275373931363535313834383840666c6f6469756d2e7275/i1.jpg"}]}`
`[/extra]`
- ссылка на аудио сообщение:
`[extra ver="1" type="audio" len="408"]`
`{"data":[{"name":"filename.mp3", "filesize":1695956, "duration":1220, "listener": [JID, ...], "domain":"files.yourdomain.com", "link":"00005af3107b23c70000d640018003e373932303833343232363140737065616b636861742e696f7b61363264663635632d396561302d343237652d393831662d3837653363313166306534667d40636f6e666572656e63652e737065616b636861742e696f/filename.mp3"}]}`
`[/extra]`
- ссылка на видео сообщение:
`[extra ver="1" type="video" len="270"]`
`{"data":[{"filesize":1754040, "name":"filename.mp4", "domain":"files.yourdomain.com", "link":"00044364c5bc17df000056a001800183730303030303033303640737065616b636861742e696f373936303535363133363340737065616b636861742e696f/filename.mp4"}]}`
`[/extra]`
- пересылка контакта
`[extra ver="1" type="contact" len="206"]`
`{"firstName":"xxxx","lastName":"xxxxxx","avatar":"https://files.yourdomain.com/avatars/42845af4e84e662e04adee51dc8973cc756dae28a182a4d3","mobile":["xxxxxxxxxx"],"home":[],"work":[],"other":[],"email":[]}`
`[/extra]`
- географическое местоположение
`[extra ver="1" type="location" len="75"]`
`{"lat":"54.639845159887","long":"36.13243536095706","name":"","address":""}`
`[/extra]`

- для подсветки пользователя в чате используется следующий формат:
`[user="7xxxxxxxxx@yourdomain.com"]Xxxxxx Xxxxx[/user]`
- событие "Пользователь только что зарегистрировался в системе", которое начинает историю любого зарегистрировавшегося контакта из вашей телефонной книги, записывается в историю самим сервером в формате
`[just-registered]`

Приложение 3. Файловые операции

В состав платформы OMMG 3.0 входит специальный файловый сервер, который позволяет хранить файлы, ссылаться на них, асинхронно конвертировать аудио- и видео- сообщения в стандартный формат, а также вести права на просмотр любого файла.

Файловая подсистема реализована на языке PHP и не является непременной частью платформы OMMG, так как расширенный формат сообщений требует только включение ссылок на файлы, которые могут быть физически размещены в любом другом месте.

1. Загрузка файла одним запросом

Для загрузки файла в один приём, без возможности отображения прогресса закачки, применяется следующий POST запрос на <https://files.yourdomain.com/upload.php> со следующими параметрами:

- `userfile` – строковое имя файла, например, `sample1.txt`;
- `local_id` – строка, локальный идентификатор файла, впоследствии используемый файловой системой для доставки оповещения об окончании конвертации файла, буде такая потребуется;
- `to_jid` – получатель файла, контакт или групповой чат, для которого устанавливаются права видимости (всем остальным этот файл будет недоступен);
- `device_type` – строковое имя текущего устройства;
- `pic=1` – опционально, используется для загрузки картинок и генерации `preview`;
- `audio=1` – опционально, указывает на то, что файл является аудио-сообщением
- `video=1` – опционально, указывает на то, что файл является видео-сообщением

Результатом выполнения такого запроса является json вида

```
{ "status": "ok", "domain": "files.yourdomain.com", "public_link": Url },
```

где `public_link` – путь на загруженный файл относительно домена, указанного в `domain`.

2. Загрузка больших файлов по частям

Для того, чтобы загружать большие файлы по частям, используется POST запрос на https://files.yourdomain.com/upload_chunk.php. Формат и параметры этого запроса полностью идентичны предыдущему, однако он позволяет собирать файл по кусочкам, используя поле заголовка `X-Content-Range`. Первым запросом передаётся начало файла и идентификатор сессии закачки:

```
POST /upload_chunk.php HTTP/1.1
Host: yourdomain.com
Content-Length: 50000
Content-Type: application/octet-stream
Content-Disposition: attachment; filename="myvideo.mp4"
X-Content-Range: bytes 0-50000/8723987623
Session-ID: 78379476837465
```

Вторым и последующими запросами в уже созданную сессию дописываются другие куски файла:

```
POST /upload_chunk.php HTTP/1.1
Host: yourdomain.com
Content-Length: 50000
Content-Type: application/octet-stream
Content-Disposition: attachment; filename="myvideo.mp4"
X-Content-Range: bytes 50001-100000/8723987623
Session-ID: 78379476837465
```

В тот момент, когда файл будет полностью передан, сервер скомпилирует итоговый файл, разместит его в хранилище и выдаст ответ, описанный в разделе 1 данного приложения.

3. Работа с аватарками

Аватар – это небольшое изображение пользователя, размером до 320x320. Чтобы загрузить аватар, надо послать POST запрос на https://files.yourdomain.com/upload_avatar.php со следующими параметрами:

- `userfile` – поле HTTP-формы, отвечающее за файл;
- `gc` – опциональный идентификатор группового чата, если аватар выставляется для группового чата;
- `team` – опциональное имя команды, если загружается локальный аватар в команде.

По завершении операции в теле ответа приходит json следующего содержания:
{`"status": "ok", "url": STRING`}

Для удаления аватара надо послать такой же запрос с пустым телом и параметром `remove`:
https://files.yourdomain.com/upload_avatar.php?remove=1

Аватар будет удалён для того пользователя или команды, которым авторизован вызов.

Чтобы задать в качестве аватара одну из предустановленных картинок, надо выполнить запрос https://files.yourdomain.com/set_def_avatar.php?avatar_id=NNN где NNN – номер предустановленной картинке от 1 до 40.

Для того, чтобы считать аватар другого пользователя, надо запомнить хеш аватара, который присылается в описании ростера или запроса на дружбу, и отправить запрос на <https://files.yourdomain.com/avatars/<hash>>

Для считывания аватаров группчатов зарезервирован специальный URL:
<https://files.yourdomain.com/avachat/{GUID}>, где GUID – часть идентификатора группчата до @. Обращаем ваше внимание, что символы { и } в URL должны быть закодированы как %7B и %7D соответственно.

Для локальных аватаров команды предусмотрен другой адрес:
<https://files.yourdomain.com/avateam/<teamid>/<hash>>

4. Конвертация файлов.

Иногда для размещения файлов их приходится конвертировать в единый формат, понимаемый всеми клиентами на всех платформах. Так, например, аудио-сообщения переводятся в формат MP3, а видео – MP4 с использованием кодека h264.

Для таких сообщений после загрузки файла не указывается ссылка на результат, а лишь возвращается подтверждение того, что файл принят для обработки. Впоследствии, после завершения обработки файла, файловый сервер передаёт событие завершения через сервер приложений в формате сообщения:

```
{  
  "type": "401",           // тип события - сообщение  
  "body": STRING,        // сообщение в произвольном формате  
  "ts": TIME_STAMP,      // серверное время сообщения  
  "sid": SID,            // серверный идентификатор сообщения  
  "with": STRING,        // получатель файла  
  "localId": GUID,       // локальный идентификатор файла  
}
```

```
' "deviceType": STRING,      // имя устройства
  "from_global_id": GUID     // идентификатор устройства, инициировавшего загрузку
}
```

5. Предварительный просмотр.

Для работы на мобильных клиентах важное значение имеет загрузка изображений небольшого размера для предварительного просмотра больших изображений и видеороликов. Загрузка таких изображений делается с помощью вспомогательных виртуальных URL:

- обычное изображение для предварительного просмотра имеет наибольший размер 320 точек и загружается через добавочный виртуальный URL /preview:
<https://files.yourdomain.com/linktovideo.mp4/preview>
- изображение с уменьшенным максимальным размером (меньше 320 точек) загружается через добавочный виртуальный URL /preview/NNN, где NNN – максимальный размер в точках от 1 до 320:
<https://files.yourdomain.com/linktovideo.mp4/preview/200>
- квадратное изображение с размером 320 точек, вырезанное по центру, загружается через добавочный виртуальный URL /crop:
<https://files.yourdomain.com/bigpicture.jpg/crop>
- квадратное изображение уменьшенного размера, вырезанное по центру, загружается через добавочный виртуальный URL /crop/NNN, где NNN – размер уменьшенного изображения в точках от 1 до 320:
<https://files.yourdomain.com/bigpicture.jpg/crop/300>

6. Получение ссылки на файл.

Иногда, чтобы передать загруженный файл другому пользователю или в другой групповой чат, необходимо получить копию файла с другими правами на чтение. Платформа OMMG позволяет делать ссылки на файл, избегая повторной загрузки файла, с помощью запроса на https://files.yourdomain.com/make_link.php со следующими параметрами:

- filename – оригинальная ссылка на файл;
- to_jid – идентификатор пользователя или группового чата, которому выдаются права на доступ к файлу.

В результате возвращается json следующего формата:

```
{"status": "ok", "public_link": STRING}
```

Допускается одновременная обработка нескольких файлов. Для этого их имена передаются не в параметре filename, а в теле POST-запроса, разделённые возвратом каретки (\n). В таком случае в ответе придёт json со списком URL в том же порядке, в каком присланы исходные имена файлов.

7. Удаление файлов.

Если файл больше не нужен клиенту, или удаляется сообщение, содержащее имя файла, его необходимо стереть с сервера. Делается это с помощью POST запроса на <https://files.yourdomain.com/remove.php>, в теле которого указывается список файлов (в виде URL на скачивание), разделённых возвратом каретки (\n).

8. Получение списка файлов.

Пользователь может скачать с сервера список всех залитых им на сервер файлов. Для удобства показа список может быть получен кусками, начиная с определённого файла или даты, в направлении возрастания или убывания.

Чтобы получить список файлов, надо отправить следующий GET запрос:

https://api.yourdomain.com/get_file_list.php, указав при необходимости параметры:

- `exact-id`: уникальный идентификатор файла;
- `since-id`: по возрастанию начиная с указанного номера файла;
- `to-id`: по убыванию до указанного номера файла;
- `since-time`: по возрастанию начиная с указанного номера файла;
- `to-time`: по убыванию до указанного номера файла;
- `with`: JID собеседника или группового чата, которому отправлен файл;
- `limit`: максимальное количество файлов в выдаче;
- `latest`: сортировка по убыванию (по умолчанию — по возрастанию).

Результат выдаётся в виде JSON списка:

```
{
  "data" : [
    {
      url: STRING,      // относительный путь к файлу на файловом сервере
      link: STRING,    // полный путь для скачивания файла
      size: INT,       // размер файла
      date: STRING     // дата и время заливки файла
    },
    {...}],
  "partial": BOOL     // существуют ли данные после переданного массива
}
```

Приложение 4. Работа с телефонными книжками

Платформа OMMG 3.0 включает набор вызовов для обработки телефонных книжек (в дальнейшем — ТК), т.е. списков произвольной информации о контактах пользователя (номер телефона, e-mail, номер паспорта, СНИЛС, etc.), по которым может производиться поиск пользователей для автоматического включения таких пользователей в список контактов.

Включение контакта производится как при регистрации самого пользователя, так и вследствие регистрации контакта (в этом случае пользователь получает специальное уведомление).

ТК привязывается к конкретному устройству, на котором установлен клиент. При разрегистрации устройства всё содержимое ТК удаляется с сервера.

1. Сохранение ТК на сервере

Для сохранения ТК на сервере служит POST-запрос на https://api.yourdomain.com/save_phonebook.php, передаваемый со следующими параметрами:

Имя	Тип	Описание
request	STRING	Вид запроса
action	STRING	Код команды, если указан request=pb
id	STRING	Опционально, если надо передать идентификатор строки в ТК
globalId	STRING	Опционально, если надо передать идентификатор устройства, по умолчанию – текущее устройство

В зависимости от значения параметра request, вызов делает следующие действия:

- pb: обработка ТК. В зависимости от переданного параметра action делает следующее:
 - delete: удаление одной строки ТК для идентификатора, присланного в параметре id;
 - bdracyiadRo: «секретный» код действия для удаления всей ТК;
 - YatCkDZsTMC: «секретный» код действия для удаления всей ТК и самой учётной записи пользователя, применяется для случая экстренного стирания всех данных клиента с сервера

Если параметр action не передан, то тело запроса POST интерпретируется как массив данных, который надо записать в ТК. О его формате будет рассказано ниже.

- pbwipe: удаляет из ТК данные, относящиеся к устройству, идентификатор которого передаётся в параметре globalId;
- pbkill: удаляет из учётной записи все данные об устройстве, идентификатор которого передаётся в параметре globalId, включая само устройство, токен для пушей и ТК

Данные для записи в ТК имеют следующий формат json:

```
{
  // описание устройства
  "device": {
    "name": STRING,           // имя устройства
    "globalId": GUID         // идентификатор устройства
  },

  // собственные данные
  "metadata": [
    "...": STRING,           // произвольные строки с данными
    "...": STRING,           // название переменных определяется клиентом
  ]
}
```

```

    "...": STRING
  ],
  // строки телефонной книги
  "data": [
    // строка для вставки
    {
      "localId": INT,          // идентификатор для вставки в ТК
      "name": STRING,         // уникальное имя, опционально
      "firstName": STRING,    // имя, опционально
      "lastName": STRING,     // фамилия, опционально
      "...": STRING,          // произвольные строки с данными
      "...": STRING,          // название переменных определяется клиентом
      "...": STRING
    }
    // строка для изменения
    {
      "id": INT,              // номер строки ТК
      "name": STRING,         // уникальное имя, опционально
      "firstName": STRING,    // имя, опционально
      "lastName": STRING,     // фамилия, опционально
      "...": STRING,          // произвольные строки с данными
      "...": STRING,          // название переменных определяется клиентом
      "...": STRING
    },
    // строка для удаления
    {
      "id": INT,              // номер строки телефонной книжки
      "delete": 1             // флаг удаления
    }
  ]
}

```

Результат работы [save_phonebook.php](#):

```

{
  "result": "ok",
  "version": INT,           // версия ТК, увеличивается при каждой записи
  "deviceId": GUID,        // идентификатор устройства
  "data": [
    {...},                  // массив результатов, по одной записи на каждую
    {...},                  // запись из параметра data
  ]
}

```

Для вставки строки в ТК клиентское приложение должно сгенерировать собственный идентификатор, уникальный для каждой добавляемой строки, и поместить его в поле localId. В результате вернётся пара чисел {"localId":STRING, "id":INT}, по которой приложение должно сохранить новый идентификатор строки в серверной ТК.

Для изменения или удаления строки из ТК приложение должно указать идентификатор строки из серверной ТК.

Для формирования имени строки ТК может передаваться либо параметр name, тогда поля firstName и lastName будут игнорироваться, либо, в случае его отсутствия, в имя строки будет записано firstName, и, в случае присутствия параметра lastName, его значение будет добавлено к firstName через пробел.

Имена переменных для произвольных строк внутри каждой записи ТК могут быть любыми сочетаниями английских букв и подчёркивания, за следующим исключением: поля с именами 'issms', 'id', 'name', 'lastname', 'firstname', 'localid', 'deviceid', 'globalid', 'accounts' будут проигнорированы и никогда не будут записаны на сервер.

2. Считывание ТК с сервера

Для считывания ТК с сервера используется HTTP-запрос типа GET на https://api.yourdomain.com/read_phonebook.php, передаваемый со следующими параметрами:

Имя	Тип	Описание
request	STRING	Вид запроса, должен равняться rb
deviceOnly	BOOL	Опционально, если надо только считать информацию о списке устройств

Результатом работы является JSON следующего формата:

```
{
  "result": "ok",
  "ownNumber": STRING,           // опционально – собственный номер телефона
  "ownUniqueID": STRING,        // опционально – собственный логин
  "devices": [                  // список информации об устройствах
    {
      "name": STRING,           // имя устройства
      "globalid": GUID,         // идентификатор устройства
      "lastLogin": TIME_STAMP,  // время последнего входа
      "active": BOOL,           // находится ли в онлайн
      "version": INT            // версия ТК этого устройства
    },
    {...}, {...}
  ],
  "data": [
    {
      "id": INT,                 // номер строки ТК
      "deviceId": GUID,          // идентификатор устройства
      "...": STRING,             // произвольные строки с данными
      "...": STRING,             // название переменных определяется клиентом
      "...": STRING,
      "jid": JID                 // опционально – идентификатор пользователя,
                                // ассоциированный с этой строкой ТК
    },
    {...}, {...}
  ]
}
```