

LiraX Telephony API

General API Information

API allows connect to LiraX any of your software products (External Software Product).

Integration to LiraX solves several business tasks:

- retrieving the caller name by number phone;
- retrieving the manager number, assigned to the client;
- displaying information about incoming call directly inside External Software Product;
- saving inside External Software Product history all calls and links to conversation recordings;
- creating a Contact;
- Creating a task for Contact and manager;
- Creating a note for Contact;
- Creating a Deal for contact;
- Updating a Deal
- making outgoing calls immediately from interface External Software Product.

Integration must be two-way. Therefore part requests LiraX sends in side External Software Product to specified entry points, and part requests, vice versa, External Software Product sends in side LiraX to single entry point.

Interaction is performed using the protocol HTTPS. Authorization is performed by address LiraX or External Software Product respectively and authorization key, obtained during integration settings.

Authorization and Interaction Principle

HTTPS:

- Requests to LiraX are accepted only using the protocol HTTPS. This provides sufficient security level for communication systems over the Internet.
- For the security of your data, please, implement receiving requests to side External Software Product also using HTTPS.

Key (token):

- Additionally for authorization each request to LiraX is used special key (token). Receive it can in integration settings.
- Please, implement receiving requests to side External Software Product also using key (token). Generate key and insert it in corresponding field in section integration settings.
- Keys are created once when configuration integration. If necessary, you can change the key to your side and update it in the interface LiraX.

Requests from LiraX to the External Software Product:

- Requests must send to address specified in the integration settings for receiving notifications, example <https://domain/LiraX>.
- In the request body in special field «token» must always be passed special key.
- Requests are passed in application/x-www-form-urlencoded format

Requests from the External Software Product to LiraX:

- External Software Product must send all requests to address specified in the integration settings
- In the message body External Software Product must pass specified by you in web account key (token).
- Requests are passed in application/x-www-form-urlencoded format

Responses

- All responses to requests LiraX External Software Product must send in JSON format in the response body.
- All responses to requests External Software Product LiraX sends in JSON format in the response body.

API Command List

From the External Software Product to LiraX:

- makeCall (POST,GET)
- killCall (POST,GET)t
- make2Calls (POST)
- AskQuestion (POST)
- AddTag (POST)
- DelTag (POST)
- get_sip_route_in (POST)
- set_sip_route_in (POST)
- getUserSips (POST)
- IsFreeUsers (POST)
- IsCalling (POST)
- AddCampaign (POST)
- AddPhoneCampaign (POST)
- checkContact (POST)
- getContact (POST)
- getStatInfo (POST)
- createTask (POST)
- AddTaskResult (POST)
- GetTask (POST)
- createNote (POST)
- createDeal (POST)
- updateDeal (POST)
- addBlackPhone (POST,GET)
- addBlackIP (POST,GET)
- getStages (POST)
- getShops (POST)
- getUserStatuses (POST)
- getUsers (POST)
- initStatuses (POST)
- sendMsg (POST)

- send_cloud_message (POST)
- sendSMS (POST)
- checkSMS (POST)
- EncodePhone (POST)
- DecodePhone (POST)
- set_call_lost (POST)
- get_makecall_data(POST)
- get_calls
- set_webhook
- Loading call history

From LiraX to the External Software Product:

- staton (POST)
- smsDelivered (POST)
- smsReceived (POST)
- contact (POST)
- event (POST)
- record (POST)

*All requests with Content-Type: application/x-www-form-urlencoded

API Commands and Examples of Available Scenarios

Commands from the External Software Product to LiraX

makeCall

The command is required to initiate call from the manager to the client. As a result of successful command execution, LiraX will first make a call to the manager phone, and then connect the manager with the client.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case makeCall	string	
from	employee extension making outgoing call	string	
to	number to which the outgoing call is made	string	
token	LiraX key (token) set in the integration settings	string	
idshop	Optional parameter, indicates id shop, from which will be call	integer	(list id can receive function GetShops)

Request example:

POST https://lira.voip.com.ua:8482/general

```
cmd=makeCall
from=101
to=19101234567
```

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	{"id_makecall": "\$id_makecall"}	OK, \$id_makecall – is used when receiving webhook makecall_finished
400		Invalid parameters were passed
401		Invalid key (token) was passed

killCall

The command is required to forcibly complete call from the manager to the client.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case killCall	string	
Call_id	id call, received in event makecall_finished	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=killCall

Call_id=101234234234s3

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

make2Calls

The command is required to connect between each other two subscriber. As a result of successful command execution, LiraX will first make a call to phone to1, and then connect it with to2. From required for proper call routing.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case make2Calls	string	
from	employee extension, from on behalf of which are	string	

	made outgoing calls		
to1	number, to which is made first call	string	
to2	number, to which is made second call or url audio file(mp3,wav) if only an audio file needs to be played	string	
speech	Text which must be spoken after answering the call to1. Text must consist from language(“uk “, “en “, “ru “) and itself text	string	Optional
atempo	Optional parameter, indicates speed playback from 0.5 until 2.0	string	Optional
timeout	Time in minutes (if more than 10 every 10 minutes will be retries until answering the call both subscribers). Default 0 – only one attempt	string	Optional
sucesstime	Time in seconds – How many seconds, must take place conversation, to consider, that call successful. Default 0	string	Optional
notbefore	Time in hours – Until what hour must not call. Default 08	string	Optional
notafter	Time in hours – After what hour must not call, Default 20	string	Optional
try	Number of attempts call attempts to first side with an interval 10 minutes	integer	Optional =1
vtime	Time in hours and minutes – hh:mm. Parameter can be used for IVR	string	Optional
vdate	Date – “dd.mm.YYYY”, “today” or “tomorrow”. Parameter can be used for IVR	string	Optional
vmoney	Money value of type float 0.00 with two digits after the decimal points. Parameter can be used for IVR	string	Optional
token	LiraX key (token) set in the integration settings	string	
idshop	Optional parameter, indicates id shop, from which will be call	integer	(list id can receive function GetShops)
FirstInternal	Optional parameter, indicates, that the first leg is an internal call if =1	integer	Optional
SecondInternal	Optional parameter, indicates, that the second leg is an internal call if =1	integer	Optional
SpeechNoWait	Optional parameter, if =1, indicates, that speech is played instead of ringback tones when call side to2	integer	Optional

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=make2Calls

```
from=101
```

```
to1=19101234561
```

```
to2=19101234567
```

```
speech=en Hello Customer
```

```
timeout=60
```

```
successtime=30
```

```
notbefore=18
```

```
notafter=21
```

```
token=202cb962ac59075b964b07152d234b70
```

```
cmd=make2Calls
```

```
from=201
```

```
to1=19101234567
```

```
to2=https://www.test.mp3/tttt.mp3
```

```
timeout=60
```

```
successtime=30
```

```
notbefore=18
```

```
notafter=21
```

```
try=4
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"id_make2calls":"\$id_make2calls"}	OK, \$id_make2calls – is used when receiving webhook make2calls_finished
400		Invalid parameters were passed
401		Invalid key (token) was passed

AskQuestion

The command is required to ask the subscriber a question and receive a response. As a result of successful command execution, LiraX will first make a call to phone to1, and then will play the greeting «hello», if necessary possible additionally play until 4 messages «text1-4» selected language (required if pauses are needed). Then the question is played, if enabled option voice recognition, waits correct words in response «ok», if there are no recognition, waits for a button press. «From» required for proper call routing. The behavior is similar to the make2calls command

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case AskQuestion	string	
from	employee extension, from on behalf of which are made outgoing calls	string	
to1	number, to which is made first call	string	
token	LiraX key (token) set in the integration settings	string	
hello	Optional parameter, sentence must as greeting	string	

	(type Hello)		
text1	Optional parameter, Message 0	string	
text2	Optional parameter, Message 0	string	
text3	Optional parameter, Message 0	string	
text4	Optional parameter, Message 0	string	
bye	Optional parameter, Message type goodbye, works after the question	string	
ask	Question message	String	
ok	Space-separated words, which we expect receive in positive response (example: yes OK)	String	
cburl	Optional parameter, url for webhook with message make2calls_finished, in which will be passed data with response subscriber	String	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=AskQuestion
```

```
from=101
```

```
cburl=https://mydomain.co/
```

```
tol=19101234561
```

```
token=202cb962ac59075b964b07152d234b70
```

```
hello=en Hello
```

```
text1=en A new service is offered
```

```
ask=en If you want to receive it, say yes
```

```
ok=en yes want need
```

```
bye=en Goodbye
```

Response options:

HTTP code	Body	Description
200	{"id_make2calls": "\$id_make2calls"}	OK, \$id_make2calls – is used when receiving webhook make2calls_finished
400		Invalid parameters were passed
401		Invalid key (token) was passed

AddTag

The command adds a tag to a contact

Request parameters:

Name	Description	Type/data	Note
------	-------------	-----------	------

		format	
cmd	operation type, in this case AddTag	string	
ani	Contact phone number	string	Optional
tag	tag	string	Optional
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=AddTag

ani=19101234567

tag=OK

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

DelTag

The command removes a tag from a contact

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case DelTag	string	
ani	Contact phone number	string	Optional
tag	tag	string	Optional
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=DelTag

ani=19101234567

tag=OK

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200		OK

400		Invalid parameters were passed
401		Invalid key (token) was passed

get_sip_route_in

The command is required to determine list sip numbers for extension phone

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case get_sip_route_in	string	
token	LiraX key (token) set in the integration settings	string	
phone	Extension	string	Example: 302

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=**get_sip_route_in**

phone=302

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	<pre>{ "N":2, "description":"route_in_sip.", "results":[{ "sip_name":"Test", "sip_number":"1115", "user_name":"User", "ext":"222", "priority":"0", "time_plan":"1" }, { "sip_name":"Test1", "sip_number":"1156", "user_name":"User1", "ext":"223", "priority":"0", "time_plan":"1" }] }</pre>	OK, if there are no SIPs we get "N":0, N-number SIPs, sip_name Name SIP, sip_number – SIP number, user_name – Name user, ext – Extension user, priority – current priority SIP, time_plan – id weekly plan (business-hours=1, evening=2, night=3, holiday=4)
400		Invalid parameters were passed
401		Invalid key (token) was passed

402		Internal error
-----	--	----------------

set_sip_route_in

The command is required to set priority sip numbers for extension phone

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case set_sip_route_in	string	
token	LiraX key (token) set in the integration settings	string	
phone	Extension	string	Example: 302
sip_number	SIP number	string	Example: 1115
time_plan	id weekly plan (business-hours=1, evening=2, night=3, holiday=4)	string	Example: 1
priority	priority SIP in extension number	string	Example: 1

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=set_sip_route_in

phone=302

sip_number=1115

priority=0

time_plan=1

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	{"result": "1"}	OK, Upon successful update, result=1, If there is no SIP result=0
400		Invalid parameters were passed
401		Invalid key (token) was passed
402		Internal error

getUserSips

The command is required to determine there is whether available phones for extension ext.

Command caches data and updates them 1 time in hour

Request parameters:

Name	Description	Type/data format	Note
------	-------------	------------------	------

cmd	operation type, in this case getUserSips	string	
token	LiraX key (token) set in the integration settings	string	
ext	Extension	string	Example: 302
web	If set > 0, then are returned only web phones	integer	Optional

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=getUserSips
```

```
ext=302
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"user_sips": [{"phone":"10000","user_id":"1111","ext":"302"}]}	OK, if there are no available SIPs we get {"user_sips":[]}
400		Invalid parameters were passed
401		Invalid key (token) was passed

IsFreeUsers

The command is required to determine, can user to Web phone accept call.

Command checks: - enabled whether Web phone; - no whether conversation with user and returns first number from free

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case IsFreeUsers	string	
token	LiraX key (token) set in the integration settings	string	
phones	List of extensions or SIP phone numbers	string	Example: 1212121, 1212123, 302
amo	If set in 1, then numbers are used as amocrm user_id	string	Optional

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=IsFreeUsers
```

```
Phones=1212121,1212123,302
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"result":"phoneFromPhones","sip":"sip_number", "ext":"number"}	OK, if all numbers busy we get {"result":null}
400		Invalid parameters were passed

401		Invalid key (token) was passed
-----	--	--------------------------------

IsCalling

Command checks, no whether conversation with SIP number and returns first number without conversation(checks registration)

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case IsFreeUsers	string	
token	LiraX key (token) set in the integration settings	string	
phones	List of extensions or SIP phone numbers	string	Example: 1212121, 1212123

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=IsCalling

Phones=1212121, 1212123

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	{"result":"phoneFromPhones","ext":"number"}	OK, if all numbers busy we get {"result":null}
400		Invalid parameters were passed
401		Invalid key (token) was passed

AddCampaign

The command is required to Create or set a campaign for calling subscribers or sending SMS.

When calling subscribers first call goes to user, and the stream goes to the subscriber

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case AddCampaign	string	
from	employee extension, from on behalf of which is performed campaign	string	
ext	employee extension, to which will be be accepted call employees (Default equal to from)	string	Optional
days	List days(0-Sunday,6-Saturday)	string	Default: 1,2,3,4,5
time	Range time	string	By default: 10:00-18:00
pdd	Time to post-processing(seconds)	integer	Default: 5
preview_timeout	Delay outgoing call, after answering the call	integer	Optional

	employee (Default 0)		
try	Number of attempts	integer	Default: 1
token	LiraX key (token) set in the integration settings	string	
type	Campaign type 1 - Phone, 2 - SMS	integer	Default: 1
phones	List of phone numbers(if list will be empty, campaign stops)	string	Example: 12121234567, 12121234568
message	Message text, for type=1 must be first word language {en,uk,ru,...} (in this case will be first call subscriber and preview_timeout is ignored)	string	Required for type=2
idshop	Optional parameter, indicates id shop, from which will be call	integer	Optional (list id can receive function GetShops)

Note: For type=1 number simultaneous calls is determined in settings campaign in account user

For preview mode the the call first goes to the user, and the stream goes to the subscriber, when message != NULL, vice versa

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=AddCampaign

from=101

days=1,2

time=10:00-11:00

timeout=60

pdd=30

try=2

idshop=2

type=2

message>Hello

Phones=12121234567,12121234568,12121234569

token=202cb962ac59075b964b07152d234b70

or

cmd=AddCampaign

from=101

days=1,2

time=10:00-11:00

timeout=60

pdd=30

try=2

idshop=2

type=1

message=en Hello

Phones=12121234567,12121234568,12121234569

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

AddPhoneCampaign

The command is required to add phone numbers in campaign for calling subscribers or sending SMS.

When calling subscribers the the call first goes to the user, and the stream goes to the subscriber for preview mode, when message != NULL, vice versa

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case AddPhoneCampaign	string	
from	employee extension, from on behalf of which is performed campaign	string	
token	LiraX key (token) set in the integration settings	string	
phones	List of phone numbers	string	Example: 12121234567, 12121234568

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=AddPhoneCampaign

from=101

Phones=12121234567, 12121234568, 12121234569

token=202cb962ac59075b964b07152d234b70

checkContact

The command is required to insert New or update an existing contact, is updated Name contact and Responsible employee.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case checkContact	string	
ext	employee extension responsible for contact	string	
ani	Primary Contact phone number	string	
name	Name Contact	string	
add_phone	List additional phones	string	Optional
emails	List e-mail	string	Optional
token	LiraX key (token) set in the integration settings	string	

trackid	Google tracking id for site	string	Optional
clientid	Google Client id	string	Optional
sitename	Name site	string	Optional
source	Source	string	Optional
stype	Type Source (Medium	string	Optional
campaign	Campaign	string	Optional
Term	Keyword word	string	Optional
tag	Tag (when presence is inserted tag)	string	Optional

Request example:

```
POST https://lira.voip.com.ua:8482/general
cmd=checkContact
ext=101
ani=19101234567
Name=Dima
token=202cb962ac59075b964b07152d234b70
add_phone=380123456789,380234567890,380112345678
emails=a@a.com,b@b.com
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

getContact

The command is required to receive information about existing contact and it Responsible employee.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case getContact	string	
ani	Contact phone number	string	Optional
email	e-mail	string	Optional
token	LiraX key (token) set in the integration settings	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
cmd=getContact
ani=19101234567
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	<pre>{"result": {"name":"Dmitry","mainphone":"12345678901"," responsible":"209","sip":"111"}}</pre>	OK Name - Name contact, mainphone – first Contact phone number, responsible - Extension responsible, sip - sip number responsible
400		Invalid parameters were passed
401		Invalid key (token) was passed

getStatInfo

The command is required to receive Statistics calls by employees for period not more than month. Command can use not more often 1 time to hour

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case getStatInfo	string	
Start	Date in format “YYYY-MM-DD HH:MM:SS” - start interval	string	
Stop	Date in format “YYYY-MM-DD HH:MM:SS” - end interval	string	Optional, if absent, then for 24 hour
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=getStatInfo

Start=2021:01:01 00:00:00

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	<pre>{ "results":[{ "description":"stat_info.", "stat":[{ "login":"alex", "name":"Alexey", "count_incoming":"0",</pre>	OK

	<pre> "count_incoming_unique":"0", "count_incoming_answered":"0", "count_incoming_answered_unique":"0", "count_incoming_lost":"0", "count_incoming_lost_unique":"0", "count_outgoing":"10", "count_outgoing_unique":"4", "count_outgoing_answered":"5", "count_outgoing_answered_unique":"2", "count_outgoing_lost":"5", "count_outgoing_lost_unique":"4", "incoming_duration_talk":"0", "outgoing_duration_talk":"765" }] } </pre>	
400		Invalid parameters were passed
401		Invalid key (token) was passed

createTask

The command is required to create New task for Responsible employee and contact. If contact not exists, is created new

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case createTask	string	
ext	employee extension responsible for contact	string	
department	Name Department	string	Optional
ani	number, phone Contact	string	Required ani or email
email	e-mail Contact	string	Required ani or email
token	LiraX key (token) set in the integration settings	string	
webhook	Url for notification about result	string	In end will be appended text
text	Text tasks	string	
date	Date execution tasks	Date	Optional, Default current time +1sec. Example: "2017-01-01 17:00:00", must be more than current dates and less two year forward
type	Type	Integer	1,2,3 (Follow-up, Meeting, Call)

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=createTask

ext=101

department=Sales

ani=19101234567

text=Prepare Task

date=2017-01-01 17:00:01

type=1

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	{"idtask":"22"}	OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

AddTaskResult

The command is required to add result to task, complete task or change responsible

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case AddTaskResult	string	Required
ext	employee extension who added result	string	Required
idtask	Id Tasks	Integer	Required
newext	New responsible	string	Optional
finish	Completion tasks	bool	Optional
token	LiraX key (token) set in the integration settings	string	
text	Text result	string	Required

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=AddTaskResult

ext=101

idtask=123123

newext=191

text=Task at stage 1

finish=0

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
-----------	------	-------------

200	{"idtask":"123123"}	OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

GetTask

The command is required to add result to task, complete task or change responsible

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case AddTaskResult	string	Required
task_id	Id Tasks	Integer	Required
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=GetTask

task_id=123123

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	{ "task_id":"609", "type":"Follow-up", "date_insert":"2014-12-09 14:15:55", "date_end":"2014-12-09 15:12:07", "text":"test Task!", "ext":"101", "Name":"System", "coauthors":"1", "list":[{ "ext":"504", "Name":"Sergey Rybalkchenko" }] }	Here type – type tasks; date_insert – Date creation; date_end – Date execution; text – text tasks; ext – responsible; Name - Name responsible; coauthors - number co-assignees; list – list co-assignees
400		Invalid parameters were passed
401		Invalid key (token) was passed

createNote

The command is required to create note for existing contact from Responsible employee.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case createNote	string	
ext	employee extension responsible for contact	string	
ani	number, phone Contact	string	
token	LiraX key (token) set in the integration settings	string	
text	Text note	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=createNote
```

```
ext=101
```

```
ani=19101234567
```

```
text=News
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

createDeal

The command is required to create New deal for Responsible employee and existing contact.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case createDeal	string	
ext	employee extension responsible for contact	string	
ani	number, phone Contact	string	
token	LiraX key (token) set in the integration settings	string	
name	Name deal	string	
sum	Amount deal	Integer	
stage	Stage deal	Integer	Allowed only existing stages

status	Status deal (Optional)	Integer	0-active, 1- Successfully Completed, 2- Completed Unsuccessfully
--------	------------------------	---------	--

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=createDeal
```

```
ext=101
```

```
ani=19101234567
```

```
name=Prepare Task
```

```
sum=3030
```

```
stage=1
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"id_deal":\$integer}	OK (\$integer - integer number)
400		Invalid parameters were passed
401		Invalid key (token) was passed

updateDeal

The command is required to update existing deal with id_deal for Responsible employee and existing contact.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case updateDeal	string	
ext	employee extension responsible for contact	string	
ani	number, phone Contact	string	
token	LiraX key (token) set in the integration settings	string	
id_deal	Id deal	Integer	
name	Name deal	string	
sum	Amount deal	Integer	
stage	Stage deal	Integer	Allowed only existing stages
status	Status deal	Integer	0-active, 1- Successfully Completed, 2- Completed Unsuccessfully

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=updateDeal
```

ext=101
ani=19101234567
id_deal=1910
name=Prepare Task
sum=3030
stage=2
status=0
token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed

addBlackPhone

The command is required to add in Black list number phone (only digits).

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case addBlackPhone	string	
phone	number phone	string	Only digits
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=addBlackPhone
phone=100111000101
token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

listBlackPhone

The command is required to receive Black list phones.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case listBlackPhone	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=listBlackPhone
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"blacklist":["phone1","phone2",...,"phoneN"]}	OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

delBlackPhone

The command is required to delete in Black list number phone (only digits).

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case delBlackPhone	string	
phone	number phone	string	Only digits
token	LiraX key (token) set in the integration settings	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=delBlackPhone
```

```
phone=100111000101
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

listBlackIP

The command is required to receive Black list IP addresses

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case listBlackIP	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=listBlackIP
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"blacklist":["IP1","IP2",..., "IPN"]}	OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

addBlackIP

The command is required to add in Black list IP address.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case addBlackIP	string	
IP	IP Address	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=addBlackPhone
```

```
IP=100.111.000.101
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200		OK

400		Invalid parameters were passed
401		Invalid key (token) was passed

delBlackIP

The command is required to delete in Black list IP address.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case delBlackIP	string	
IP	IP Address	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=delBlackPhone

IP=100.111.000.101

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

getStages

The command is required to receive list Stages deals, sorted According to Pipeline.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case getStages	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=getStages

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	{"stages":[{"stage":\$stage,"title":"..."},...]}	OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

getShops

The command is required to receive list shops.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case getShops	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=getShops
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"shops":[{"id":\$id_shop,"name":"..."},...]}	OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

getUserStatuses

The command is required to receive list statuses statuses user.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case getUserStatuses	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=getUserStatuses
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
-----------	------	-------------

200	{"statuses":[{"status":\$status,"title":"..."},...]}	OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

getUsers

The command is required to receive list users

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case getUsers	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=getUsers

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	{"users":[{"id":\$id,"Name":"...","ext":"...","active":"0 1"},...]}	OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

initStatuses

The command is required to receive current statuses operators, through Webhooks. Are used when enabling integration. Further status will be arrive by them change, previous statuses must ignore

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case initStatuses	string	
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=initStatuses

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

sendMsg

The command is required to send message employee in corporate messenger.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case sendMsg	string	
ext	employee extension, to whom is sent message	string	
ani	Name Contact, which sent message	string	
token	LiraX key (token) set in the integration settings	string	
text	Message text	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
cmd=sendMsg
ext=101
ani=19101234567
text=Prepare Task
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed

send_cloud_message

The command is required to send message in cloud messenger (Telegram, Viber, Facebook).

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case send_cloud_message	string	
client	Client ID cloud messenger	string	Required when absence ani
ani	Number phone Client(is used only Telegram)		Required when

			absence client
token	LiraX key (token) set in the integration settings	string	
text	Message text	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
cmd=send_cloud_message
client=web_t123123123_00000
text=Prepare Task
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed
402		Message not passed

sendSMS

The command is required to send SMS through GSM modem. Requests need send not more often 1 time in 5 seconds to one modem.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case sendSMS	string	
ext	employee extension, who sends SMS	string	Only digits
provider	Identifier provider (number phone in GSM modem)	string	Only digits
phone	Number Phone, to whom send message	string	
token	LiraX key (token) set in the integration settings	string	
text	Message text	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
cmd=sendSMS
ext=101
phone=19101234567
provider=380123456789
text=Prepare Task
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"id_sms":\$id_sms}	OK
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed
402		Modem not connected
403		Modem busy, repeat later

checkSMS

The command is required to check status delivery SMS through GSM modem. Requests need send not more often 1 time in 5 seconds to one modem.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case checkSMS	string	
ext	employee extension, who sends SMS	string	Only digits
provider	Identifier provider (number phone in GSM modem)	string	Only digits
id_sms	Id received command sendSMS	Integer	
token	LiraX key (token) set in the integration settings	string	

Request example:

POST https://lira.voip.com.ua:8482/general

cmd=checkSMS

ext=101

id_sms=1910123

provider=380123456789

token=202cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200	{"sms_status":\$id_status}	OK, 1- passed provider, status unknown; 2- passed provider, goes delivery; 3- delivered; 4- Time delivery expired
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed

EncodePhone

The command is required to check receive encrypted number.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case EncodePhone	string	
phone	Number phone	string	Only digits
token	LiraX key (token) set in the integration settings	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=EncodePhone
```

```
phone=12123123123
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"phone":\$encoded}	OK
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed

DecodePhone

The command is required to receive decrypted number.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case EncodePhone	string	
phone	Number phone	string	Only digits
token	LiraX key (token) set in the integration settings	string	

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=DecodePhone
```

```
phone=12123120960170061170799
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{"phone":\$decoded}	OK
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed

set_call_lost

The command is required to set number phone, as missed call, for CallBack

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case set_call_lost	string	required
phone	Number phone	string	Only digits, required
ext	Number phone employees for CallBack	string	Only digits, required
time_plan	1 - Working time, 2 - Evening, 3 - Night, 4 - Days off	digits	Time interval is determined in weekly plan, Default 1
max_try	Maximum Number of attempts connection with subscriber (with operator not is considered)	digits	Default 3
interval	Initial interval between attempts in minutes, multiple 6. Subsequent interval equal to initial multiply to number of attempts	digits	Default 6
minutes	Number minutes delay	digits	Default 0
hours	Number hours delay	digits	Default 0
days	Number days delay	digits	Default 0
weeks	Number weeks delay	digits	Default 0
months	Number months delay	digits	Default 0
info	Text, which will be be spoken employee. Format: language and text. Example: "ru Call Back service to missed call, wait connection with subscriber"	string	
token	LiraX key (token) set in the integration settings	string	required

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=set_call_lost
```

```
phone=12123120960
```

```
ext=222
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed
402		Problems with server

get_makecall_data

Command allows receive information about call using id_makecall, received command makecall

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case get_makecall_data	string	required
id_makecall	Received in result makecall	string	Only digits, required
token	LiraX key (token) set in the integration settings	string	required

Request example:

```
POST https://lira.voip.com.ua:8482/general
```

```
cmd=get_makecall_data
```

```
id_makecall=e5251de22f99e071
```

```
token=202cb962ac59075b964b07152d234b70
```

Response options:

HTTP code	Body	Description
200	{ "id_makecall":"e5251de22f99e071", "call_start":"2021-10-28 16:02:32", "call_record":"https://files.lirax.net/voicerec.wav", "duration":"5" }	OK
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed
402		Problems with server

get_calls

Command allows log calls in json format (maximum 5000) for range time. For Commands possible use method GET

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case get_calls	string	required
date_start	Start range	string	In SQL format, required
date_finish	End range	string	In SQL format, required interval must be less 48hours
call_type	type call, {-1,0,1}, respectively {extension, incoming, outgoing }, if is absent, then all	numerical	optional
ani	Who called	string	optional
dnis	Where called	string	optional
offset	Skip in output number rows	numerical	optional
token	LiraX key (token) set in the integration settings	string	required

Request example:

```
GET https://api.lirax.net/general?cmd=get_calls&date_start=2023-10-04 10:00:00&date_finish=2023-10-04 11:00:00&token=-----
```

Response options:

HTTP code	Body	Description
200	[{"call_id":"6708","ani":"203", "dnis":"380123123", "start":"2023-10-04 10:21:14", "connect":"2023-10-04 10:21:25", "record": " https://files.lirax.net/voice_record2/x_10_21_14. wav ", "type":"1","duration": "163","disconnect_side":"0", "call_call_center_id":"6" }, { "call_id":"8326","ani":"380111111" ,"dnis":"380 123123", "start":"2023-10-04 10:04:07", "connect":"","record":"","type":"1","duration":"0" ,"disconnect_side":"- 1","call_call_center_id":"0" }]	OK connect – if not answered will be empty row and record – url recording conversation disconnect_side – who hung up handset, -1 system(possible by timeout), 0 – who called, 1 – to whom called call_call_center_id – what call center processed call (if there is)
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed
402		Problems with server

set_webhook

Command allows log calls in json format (maximum 5000) for range time. For Commands possible use method GET

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case set_webhook	string	required
type	«0» — incoming, «1» — outgoing	numerical	required
event_type	init - when beginning call, trying - when beginning outgoing leg, connect - when answering handset subscriber, release - when end of outgoing leg, disconnect - when end of call	string	required
json	Template json, which will be used in body webhook. Here can use macros [ani] — who calls, [dnis] — where, [date] — current time, [starto] — start incoming leg, [starta] — start answering the call subscriber, [durationo] — duration incoming leg, [durationa] — duration outgoing leg, [anidnis] — number operator for incoming call, [type] — type call «-1» — extension, «0» — incoming, «1» — outgoing example: { "call_type": "[type]", "event_date": "[date]", "number_a": "[ani]", "number_b": "[dnis]", "call_duration": "[durationa]" }	string	Can be absent, if method POST, then Body will be used as template
header_field	Additional field header, which LiraX will be insert in webhook example: Authorisation: 123123123	string	optional
method	GET POST	string	Optional Default GET
url	Url for webhook example: https://test.test.test.com/webhook?a=1&b=2	string	required
token	LiraX key (token) set in the integration settings	string	required

Request example:

```
GET https://api.lirax.net/general?cmd=set_webhook&type=0&event_type=disconnect&url=https://test.test.test.com/webhook?a=1&b=2
&json={"call_type":"[type]","event_date":"[date]","number_a":"[ani]","number_b":"[dnis]","call_duration":"[durationa]"}
&token=-----
```

Response options:

HTTP code	Body	Description
-----------	------	-------------

200		Successfully
400		Invalid parameters were passed, or parameters not updated
401		Invalid key (token) was passed
402		Problems with server

Loading call history

Authorization through API

Performs authorization user in system. All methods API can be used only after authorization.

In response to request, when successful authorization, except response body is returned cookie file, containing key session, similarly work with WEB-browser. When further requests to API-methods need back pass received cookie. Time lifetime session - 15 minutes.

All requests to API occur from under user, attributes which were used when authorization through this method. When this we consider all rights user, i.e. through API must not receive more than data, than itself user can view through interfaces systems. We recommend for API create separate user for more specific settings rights connected application.

Resource URL

POST

<https://lira.lirax.net/api/private/api/auth.php>

Parameters

PARAMETER	DESCRIPTION
USER_LOGIN	require Login user.
USER_HASH	require Password user.

Calls

https://lira.lirax.net/services/call_show.php?action=show&date_start=2016-12-06

00:00:00&date_finish=2016-12-13

23:59:59&filter=call_ani_dnis&filter_value=&call_type=&call_connect=&call_export=1

date_start

Date From

date_finish

Date Until

filter

Filter by field

possible values:

"call_ani", "call_dnis", "call_ani_dnis", "call_originip", "call_destination", "call_origination",
"call_utm_source", "call_utm_medium", "call_utm_campaign", "call_utm_term"

filter_value

Values filter

call_type

possible values:

0 incoming, 1 outgoing

call_connect

possible values:

0 not was conversation, 1 was conversation

Commands From LiraX to the External Software Product

makecall_finished

Command notifies about completion task to creation call for 2 subscribers.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case makecall_finished	string	
id_makecall	Id - makecall	string	
Call_id	Call_id outgoing leg call	string	When success=0 empty
success	Answered handset subscriber	Integer	0 1
from_LiraX_token	key (token) External Software Product, specified in integration settings	string	

Request example:

```
POST https://domain/LiraX
cmd=makecall_finished
id_makecall=ef232234feb12
from_LiraX_token=303cb962ac59075b964b07152d234b70
Call_id=b964b07152d
success=1
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

make2calls_finished

Command notifies about completion task to creation call for 2 subscribers.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case make2calls_finished	string	
id_make2calls	Id - make2calls	string	
success	Was whether answering handset	Integer	0- was not, 1-

			answered handset
duration_success	Duration conversation was not less, than expected in parameter successtime	Integer	0- less, 1- more than equal to
keys	Json array pressed buttons, ivr_name – Name IVR, ivr_entry – item IVR menu, key – selected button	Json	Can be absent
from_LiraX_token	key (token) External Software Product, specified in integration settings	string	

Request example:

```
POST https://domain/LiraX
cmd=make2calls_finished
id_make2calls=ef232234feb12
from_LiraX_token=303cb962ac59075b964b07152d234b70
success=1
keys=[{"ivr_name":"Test", "ivr_entry":"0", "key":"9"}, {"ivr_name":"Test", "ivr_entry":"09", "key":"7"}]
duration_success=0
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

staton

Command notifies about about employee statuses.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case staton	string	
ext	Extension employee LiraX	string	
status	identifier status employee in personal account	Integer	0- disconnected, 1- free, >1 -busy, value according to settings
from_LiraX_token	key (token) External Software Product, specified in integration settings	string	

Request example:

```
POST https://domain/LiraX
cmd=staton
```

```
ext=910
from_LiraX_token=303cb962ac59075b964b07152d234b70
status=1
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

smsDelivered

Command notifies about about delivery SMS.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case smsDelivered	string	
id_sms	Id SMS, received command sendSMS	Integer	
from_LiraX_token	key (token) External Software Product, specified in integration settings	string	

Request example:

```
POST https://domain/LiraX
cmd=smsDelivered
from_LiraX_token=303cb962ac59075b964b07152d234b70
id_sms=1212
```

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

smsReceived

Command notifies about about receiving SMS.

Request parameters:

Name	Description	Type/data format	Note
------	-------------	------------------	------

cmd	operation type, in this case smsReceived	string	
ani	Number phone subscriber	string	
id	Id SMS, received from provider	Integer	
provider	Name provider (number phone)	string	
text	Message text	string	In UTF8
from_LiraX_token	key (token) External Software Product, specified in integration settings	string	

Request example:

POST https://domain/LiraX

cmd=smsReceived

from_LiraX_token=303cb962ac59075b964b07152d234b70

id_sms=121

ani=38012345679

provider=380112345678

text=Hello

Response options:

HTTP code	Body	Description
200		OK
400		Invalid parameters were passed
401		Invalid key (token) was passed

contact

Command for receiving information about the name of client and it responsible employee by its phone number.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case contact	string	
phone	number contact	string	
callid	unique id call	string	
diversion	external number phone LiraX through which arrived call	string	When outgoing equal to number subscriber
from_LiraX_token	key (token) External Software Product, specified in integration settings	string	

Request example:

POST https://domain/LiraX

cmd=contact

phone=19101234567
diversion=12101234567
from_LiraX_token=303cb962ac59075b964b07152d234b70
callid=D12D0EB124F4E64AF4EA-1511

Response options:

HTTP code	Body	Description
200	{ "contact_name": "Ivanov Ivan Ivanovich", "responsible": 103 }	responsible - short number responsible user, can be absent, if such no or at it not set short numbers
400	{ "error": "Invalid parameters" }	Invalid parameters were passed
401	{ "error": "Invalid token" }	Invalid key (token) was passed

event

LiraX sends in External Software Product notifications about events calls: appearance, acceptance or completion call, and also information about call when it completion.

Request parameters:

Name	Description	Type/data format	Note
cmd	operation type, in this case event	string	
type	type call	in/out	incoming / outgoing
event	type events, related with call: INCOMING - arrived incoming call (in this time the manager phone should start ringing). ACCEPTED - call successfully accepted (manager hung up handset). COMPLETED - call completed (manager or client hung up handset after conversation). CALL_COMPLETED – call completely completed.	string	
phone	number phone client	string	
diversion	external number phone LiraX through which arrived or disconnected call	string	
ext	Extension employee LiraX, in case CALL_COMPLETED and type=in, contains number, to which arrived call	string	
callid	unique id call	string	
duration	duration conversation in seconds	int	when event=COMPLETED, CALL_COMPLETED
call_duration	Duration call to moment events in seconds	int	when event=COMPLETED, CALL_COMPLETED
is_recorded	recorded whether call	0/1	when event=COMPLETED, CALL_COMPLETED
status	status incoming call: Success - successful incoming call Missed – missed incoming call status outgoing call: Success - successful outgoing call Busy - LiraX received response Busy NotAvailable - LiraX received response Subscriber unavailable NotAllowed - Calls to this direction forbidden	string	when event=COMPLETED, CALL_COMPLETED
record_link	call recording link, if LiraX can provide it immediately, if there are no - link is sent command record	string	when event=COMPLETED, CALL_COMPLETED

tid	Tracking Id		
cid	Client Id		
cs	Source		
cm	Medium		
cc	Campaign		
ct	Term		
keys	Last selection IVR menu	json	when event=CALL_COMPLETED
From_LiraX_token	key (token) External Software Product, specified in integration settings	string	

Request example:

POST https://domain/LiraX

```

cmd=event
type=out
event=COMPLETED
phone=19101234567
ext=102
callid=D10D0EB124F4E64AF4EA-1511
duration=124
is_recorded=1
status=Success
record_link=https://link/file.wav
keys=[{"ivr_name":"Test","ivr_entry":"09","key":"4"}]
from_LiraX_token=303cb962ac59075b964b07152d234b70

```

Response options:

HTTP code	Body	Description
200		OK
400	{ "error": "Invalid parameters" }	Invalid parameters were passed
401	{ "error": "Invalid token" }	Invalid key (token) was passed

record

Notification contains information about recording conversation.

Request parameters:

Name	Description	Type/ data format	Note
cmd	operation type, in this case record	string	

callid	unique id call	string	
record_link	call recording link	string	
from_LiraX_token	key (token) External Software Product, specified in integration settings	string	

Request example:

POST https://domain/LiraX

cmd=record

callid=D10D0EB124F4E64AF4EA01511

record_link=https://link/file.wav

from_LiraX_token=303cb962ac59075b964b07152d234b70

Response options:

HTTP code	Body	Description
200		OK
400	{ "error": "Invalid parameters" }	Invalid parameters were passed
401	{ "error": "Invalid token" }	Invalid key (token) was passed

ChangeLog:

2021-01-28

make2calls- added field FirstInternal

2021-02-18

getContact- added function

2021-02-19

getStatInfo- added function

2021-04-12

AskQuestion- added function

2021-04-13

AddTag- added function

2021-04-13

DelTag- added function

2021-04-14

checkContact — added field tag

2021-08-13

get_sip_route_in — added function

2021-08-13

set_sip_route_in — added function

2021-09-12

SecondInternal in function **make2calls**

2021-09-12

SpeechNoWait in function **make2calls**

2021-10-26

set_call_lost- added function

2021-10-28

get_makecall_data - added function

2021-11-19

set_call_lost - fields added info, minutes, hours, days, weeks, months

2021-12-30

createTask — added parameter email, if contact not exists, is created new

2022-08-25

Event — added parameter call_duration

2022-11-10

send_cloud_message - added function

2023-10-04

get_calls- added function

2023-10-20

AddTaskResult- added function

2024-09-24

GetTask- added function