



VoloMP XAPI Documentation

1. Welcome to VoloMP XAPI

Overview

The VoloMP XAPI enables outside applications to make remote procedure calls to your VoloMP server on behalf of a VoloMP user. All methods are available to your developers over the XML-RPC based interface.

Example

Method parameters are presented as an associative array as in this PHP example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'delivery_queue',
        'args'   => array('user_id'=>11),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

where this call has one parameter named `user_id` with value 11.

Example with multiple parameters is:

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'delivery_queue',
        'args'   => array('user_id'=>11, 'date_limit'=>7),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

Getting Started

Make sure your programming language has support for XML-RPC

<http://www.xmlrpc.com/>

Server

Your XAPI server is at

<http://yourserver.com/xapi/server.php>

2. List of all XAPI methods

- [delivery_queue](#)
- [delivery_queue_test](#)
- [delivery_queue_autoresponder](#)
- [message_logs](#)
- [message_ip_stats](#)
- [recipients_count](#)
- [recipient_get_info](#)
- [recipient_modify](#)
- [recipients_subscribe](#)
- [recipients_unsubscribe](#)
- [send_message](#)
- [send_test_message](#)
- [save_template](#)
- [send_template](#)
- [send_draft](#)
- [templates](#)
- [drafts](#)
- [get_ip_space](#)
- [get_ip_space_presets](#)
- [ip_space_preset_create](#)
- [ip_space_preset_delete](#)
- [get_filter_presets](#)
- [mailing_lists](#)
- [mailing_lists_create](#)
- [mailing_lists_update](#)
- [mailing_lists_delete](#)
- [mailing_lists_empty](#)
- [mailing_lists_merge](#)
- [mailing_lists_split](#)
- [headers](#)
- [footers](#)
- [job_details](#)
- [job_cancel](#)
- [job_pause](#)
- [job_resume](#)
- [file_export](#)
- [file_import](#)
- [file_import_unsubscribe](#)
- [tracked_links_actions](#)
- [easy_links_keywords](#)
- [suppression_files](#)
- [banned_add](#)
- [throttling_configuration](#)
- [throttling_available_ranges](#)
- [throttling_isp_list](#)
- [throttling_available_parameters](#)
- [throttling_modify](#)
- [throttling_delete](#)

3. XAPI methods

delivery_queue

Get messages delivery queue.

Parameters

```
!      user_id – web user id (optional)
!      fields – array of fields (optional)
◦      Possible fields to select:
◦
'MsgID', 'Owner', 'Start', 'End', 'List/Campaign', 'List', 'Lists', 'Campaign', 'Subject', 'Tra
cking', 'T', 'E', 'A', 'Queued', 'Proc', 'Processed', 'Deliv', 'Delivered', 'Bounc', 'Bounced', 'C
ompl', 'RemoveComplaints', 'AOLComplaints', 'SPAMCOPComplaints', 'OtherComplai
nts', 'Opnd', 'Opened', 'Canceled', 'Clicks', 'ClickPerc', 'DataOut', 'DataIn', 'Unsubscribe
d', 'NormalSize', 'AOLSize', 'IPSpace', 'DomainSpace', 'RecurringID', 'RecurringCount
er'
!      all_data – 1/0 (optional), 0 – show data fields from fields parameter (if
set) or data from Customer->Global->Customize page, 1 – show all data
!      date_limit – date limit (optional). If not given the value is retrieved from
web customer settings
◦      6h – last 6 hours
◦      12h – last 12 hours
◦      1 – last day
◦      3 – last 3 days
◦      7 – last 7 days
◦      14 – last 14 days
◦      [int value]d – last [int value] days
◦      tm – this month
◦      lm – last month
◦      -months_number – specific month, put -2 for the whole month starting
2 months ago, etc.
◦      all – all
```

Response

```
array (
!   paused – array (messages by each queue),
!   delivering – array (messages by each queue),
!   preparing – array (messages by each queue),
!   in_queue_for_processing – array (messages by each queue),
!   scheduled – array (messages by each queue),
!   finished – array (messages by each queue),
!   canceled – array (messages by each queue),
!   error – array (messages by each queue)
)
```

Example

```

$result = xu_rpc_http_concise(
    array(
        'method' => 'delivery_queue',
        'args'   => array(
            'user_id'=>11,
            'fields' =>
array('MsgID','Owner','Start','List/Campaign','List','Campaign','Subject','
Tracking','T','E','A','Queued'),
            'date_limit'=>'30d'
        ),
        'host'   => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'   => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);

```

delivery_queue_test

Get test messages delivery queue.

Parameters

! *The parameters list is same as in **delivery_queue** method*

Response

```
array (  
    !   paused - array (messages by each queue),  
    !   delivering - array (messages by each queue),  
    !   preparing - array (messages by each queue),  
    !   in_queue_for_processing - array (messages by each queue),  
    !   scheduled - array (messages by each queue),  
    !   finished - array (messages by each queue),  
    !   canceled - array (messages by each queue),  
    !   error - array (messages by each queue)  
)
```

delivery_queue_autoresponder

Get autoresponder messages delivery queue.

Parameters

! *The parameters list is same as in **delivery_queue** method*

Response

```
array (  
    !   paused - array (messages by each queue),  
    !   delivering - array (messages by each queue),  
    !   preparing - array (messages by each queue),  
    !   in_queue_for_processing - array (messages by each queue),  
    !   scheduled - array (messages by each queue),  
    !   finished - array (messages by each queue),  
    !   canceled - array (messages by each queue),  
    !   error - array (messages by each queue)  
)
```

message_logs

Get message logs.

Parameters

! *message_id* – message id (required)

Response

```
array (
    ! delivery_analysis - download link,
    ! open_link - download link,
    ! track_links - array (download links),
    ! unsub_link - array (download link),
    ! complaints_link - array (download link)
)
```

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'message_logs',
        'args'   => array('message_id'=>1511880),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```


message_ip_stats

Get message IP stats.

Parameters

! *message_id* – message id (required)
 ! *type* – stats type (optional), possible values: 'opened', 'delivered', 'clicks', 'bounced'

Response

array ()

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'message_ip_stats',
        'args'   => array('message_id'=>1511880/*, 'type' =>
'delivered'*/),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

recipients_count

Count recipients.

Parameters

- ! *user_id* – web user id (required)
- ! *lists* – array of list id(s) (required). You can retrieve this array with *mailing_lists* XAPI function
- ! *inputFilter* – advanced selection filter

Response

int recipients count

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'recipients_count',
        'args' => array
            (
                'user_id'=>1,
                'lists'=>array(113,45),
                'inputFilter'=>"(field(EMail) not like
'%@test.com' and field(email) != 'john.smith@example.com') "
            )
        ,
        'host'    => $host,
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        //        'debug' => 2,
        'user'    => $user,
        'pass'    => $pass
    )
);
```

recipient_get_info

Get recipient's info.

Parameters

- ! *user_id* – web user id (required)
- ! *email* – recipients email (required)

Response

array (info => array(), lists => array)

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'recipient_get_info',
        'args'   => array(
            'user_id' => '11',
            'email'  => 'volotest@mailinator.com'
        ),
        'host'    => $host,
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'debug'   => 2,
        'user'    => $user,
        'pass'    => $pass
    )
);
```

recipient_modify

Modify recipient's info and lists subscription.

Parameters

! *user_id* – web user id (required)
 ! *email* – recipients email address
 ! *lists* – array of list names => boolean (optional). You can retrieve this array with *recipient_get_info* or *mailing_lists* XAPI function. true for subscribe, false for unsubscribe
 ! *info* – array of fields => values to modify (optional). You can retrieve this array with *recipient_get_info* or *mailing_lists* XAPI function

Response

int 1/0

Example

//subscribe to aol_test and unsubscribe from yahoo_test and google_test

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'recipient_modify',
        'args' => array(
            'user_id' => '11',
            'email' => 'volotest@mailinator.com',
            'info' => array('first_name' => 'John', 'last_name'
=> 'Smith', 'bcount' => 30),
            'lists' => array('aol_test' => true, 'yahoo_test'
=> false, 'google_test' => false)
        ),
        'host' => $host,
        'uri' => '/xapi/server.php',
        'port' => 80,
//      'debug' => 2,
        'user' => $user,
        'pass' => $pass
    )
);
```

recipients_subscribe

Subscribe recipients to lists.

Parameters

! *user_id* – web user id (required)
 ! *emails* – array of recipient emails (required)
 ! *list_names* – array of list names (required). You can retrieve this array with *mailing_lists* XAPI function.

Response

```
array (email_status => array(), stats => array(), subscribe_statuses
=> array())
```

//email_status values are keys in subscribe_statuses array

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'recipients_subscribe',
        'args'   => array(
            'user_id' => '11',
            'list_names' => array('aol_test', 'yahoo_test'),
            'emails' => array('volotest1@mailinator.com',
                'volotest2@mailinator.com')
        ),
        'host'   => $host,
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        //      'debug' => 2,
        'user'   => $user,
        'pass'   => $pass
    )
);
```

recipients_unsubscribe

Unsubscribe recipients from lists.

Parameters

! *user_id* – web user id (required)
 ! *emails* – array of recipient emails (required)
 ! *list_names* – array of list names (required). You can retrieve this array with *mailing_lists* XAPI function.

Response

```
array (email_status => array(), stats => array(), subscribe_statuses
=> array())
```

```
//email_status values are keys in subscribe_statuses array
```

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'recipients_unsubscribe',
        'args'   => array(
            'user_id' => '11',
            'list_names' => array('aol_test', 'yahoo_test'),
            'emails' => array('volotest1@mailinator.com',
                'volotest2@mailinator.com')
        ),
        'host'   => $host,
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        // 'debug' => 2,
        'user'   => $user,
        'pass'   => $pass
    )
);
```

send_message

Send new message.

Parameters

! *user_id* – web user id (required)

! *lists* – array of list id(s) (required). You can retrieve this array with *mailing_lists* XAPI function

! *extrafile* – full path to file (optional, instead of lists, noquery should be set to 1)

! *noquery* – 1/0 (optional) value 1 is required if sending to recipients in *extrafile*

! *header_footer* – list id (optional). It indicates one list from which we retrieve header / footer.

! *header* – header id (optional)

! *footer* – footer id (optional)

! *from* - mailing from (required). You can also get it with *mailing_lists* XAPI function

! *reply_to* - mailing Reply-To (optional)

! *subject* – mail subject (required)

! *text* – text content (optional)

! *html* – html content (optional)

! *aol* – aol content (optional)

! *send_text* – yes/no (required)

! *send_html* – yes/no (required)

! *send_aol* – yes/no (required)

! *text_from_html* – 1/0 (optional) – creates text message from HTML content (if *send_text* is chosen)

! *delivery_time* – delivery time (optional). If you want to schedule mailing you must setup in this format "MM/DD/YYYY hh:mm:ss"

! *recurring_mail* – 1/0 (optional)

! *recurring_pause* – time in seconds (optional)

! *recurring_startlist* – list id (optional)

! *recurring_rotate_ip* – 1/0 (optional)

! *recurring_subjects* – text, tab limited (optional)

! *campaign* - campaign name (optional)

! *subid* - tracking SubID (optional)

! *rotating_subjects* – array with rotating subjects (optional)

! *rotating_subjects_count* – perform Subject rotation after this amount of sent emails (optional)

! *rotating_froms* – array with rotating from emails (optional)

! *rotating_froms_count* – perform From rotation after this amount of sent emails (optional)

! *seed_emails* – array with seed emails (optional)

! *seed_count* – resend to seed addresses after this amount of mails sent to the same domain as seed address domain (optional)

! *ip_space* – ip space (optional). To setup different then default *ip_space*

! *ip_space_preset* – ip space preset name (optional). To setup different then default *ip_space*. You can't use both *ip_space* and *ip_space_preset* at the same time. Please use only one of them or none.

```

!      inputFilter – advanced selection filter
!      filter_preset – advanced selection filter preset name (optional). You can't
use both inputFilter and filter preset at the same time. Please use only one of
them or none.
!      tracked_links_domain - Domain used for links. Possible values: 'sdomain',
'shost', 'src_ip', array('other' => CustomDomainName), array('preset' =>
IPGroupPresetName)
!      tracked_links – track links (optional). You can track links from the mail
content. The parameter is array with tracked link settings. You can retrieve data
for tracked links with tracked_links_actions XAPI function and for easy links
with easy_links_keywords XAPI function
!      openers_action_id – openers action_id (optional)
!      default_openers_action – 1/0 (optional). If openers_action_id not set and
default_openers_action is set to 1 it will use Default message open action from
Customer->Global->Customize page
!      suppression_lists - array of suppression lists id(s) (optional).
!      suppression_files – array of suppression file id(s) (optional). You can
retrieve this array with suppression_files XAPI function
!      adstation – array (optional) - (
▪      listid
      )
!      adrevolution – array (optional) - (
▪      listid
      )
!      aws – array (optional) - (
▪      AWSAccessKeyId - username,
▪      AWSSecretKey - password,
▪      url – SMTP relay address
      )

```

Response

```

int      message id

```


Example

```
$result = xu_rpc_http_concise(
array(
    'method' => 'send_message',
    'args' =>
        array
        (
            'user_id'=>15,
            'lists'=>array(23, 43, 20),
            'header_footer'=>43,
            'from'=>"test@testdomain.com",
            'subject'=>'API test without rec',
            'html'=>'<p>This is API test message</p>.<br>
<a href=\"http://www.google.com\">track1</a><br>
<a href=\"http://www.yahoo.com\">track2</a>',
            'send_text'=>'no',
            'send_html'=>'yes',
            'send_aol'=>'no',
            'delivery_time'=> '08/20/2011 06:20:00',
            'recurring_mail' => 1,
            'recurring_pause' => 60,
            'recurring_startlist' => 43,
            'recurring_rotate_ip' => 1,
            'recurring_subjects' => 'test subj 002    test subj 003',
            'rotating_subjects' => array('Subject 1', 'Subject 2',
'Subject 3'),
            'rotating_subjects_count' => 2,
            'rotating_froms' => array('test1@example.com',
'test2@example.com', 'test3@example.com'),
            'rotating_froms_count' => 3,
            'seed_emails' => array('test1@example.com',
'test2@example.com', 'test3@example.com'),
            'seed_count' => 2,
            //'ip_space'=>'*.your_ip_space.kom',
            'ip_space_preset'=>'test2',
            //'inputFilter'=>"(field(EMail) like '%@gmail.com' and
field(email) != 'john@example.com')",
            'filter_preset' => 'presetname1',
            //'tracked_links_domain' => 'sdomain',
            //'tracked_links_domain' => 'shost',
            //'tracked_links_domain' => 'src_ip',
            //'tracked_links_domain' => array('other' =>
'example.com'),
            //'tracked_links_domain' => array('preset' =>
'presetname1'),
            'tracked_links'=>array
            (
                1=>array
                (
                    "type"=>"easy",
                    "domain"=>"http://www.google.com",
                    "easy_id"=>1
                ),
                2=>array
                (
                    "type"=>"track",
                    "domain"=>"http://www.yahoo.com",
                    "action_id"=>3
                )
            ),
            'suppression_lists' => array(58,27),
            'suppression_files'=>array(1)
            //'aws' => array(
            //'AWSAccessKeyId' => 'username',
```

```
        //      'AWSSecretKey' => 'password',
        //      'url' => 'SMTP relay address'
        //)
    ),
    'host'      => 'www.your_domain.com',
    'uri'       => '/xapi/server.php',
    'port'      => 80,
    'user'      => 'admin',
    'pass'      => 'your_pass'
)
);
```

Example with AdStation

```
$result = xu_rpc_http_concise(
array(
    'method' => 'send_message',
    'args' => array
    (
        'user_id'=>11,
        'lists'=>array(45),
        'header_footer'=>45,
        'from'=>'%%$adstation.friendlyfrom% <%%$adstation.from%%@%%
$_shost%%>',
        'subject'=>'%%$adstation.subject%%',
        'html' => '%%$adstation.content%%',
        'send_text'=>'no',
        'send_html'=>'yes',
        'send_aol'=>'no',
        // 'delivery_time'=> '08/20/2011 06:20:00',
        'ip_space'=>'*.your_ip_space.kom',
        'suppression_lists' => array(58,27),
        'suppression_files'=>array(1),
        'adstation' => array(
            'listid' => '1234',
            'action' => 4
        )
    ),
    'host'    => 'www.your_domain.com',
    'uri'     => '/xapi/server.php',
    'port'    => 80,
    'user'    => 'admin',
    'pass'    => 'your_pass'
);
```

Example with AdRevolution

```
$result = xu_rpc_http_concise(
array(
    'method' => 'send_message',
    'args' => array
    (
        'user_id'=>11,
        'lists'=>array(45),
        'header_footer'=>45,
        'from'=>'%%$adrevolution.friendlyfrom%% <%%$adrevolution.from%
%%$$_sdomain%%>',
        'subject'=>'%%$adrevolution.subject%%',
        'html' => '%%$adrevolution.content%%',
        'send_text'=>'no',
        'send_html'=>'yes',
        'send_aol'=>'no',
        'delivery_time'=> '08/20/2011 06:20:00',
        'ip_space'=>'*.your_ip_space.kom',
        'suppression_lists' => array(58,27),
        'suppression_files'=>array(1),
        'adrevolution' => array(
            'listid' => '1234',
            'action' => 6
        )
    ),
    'host'      => 'www.your_domain.com',
    'uri'       => '/xapi/server.php',
    'port'      => 80,
    'user'      => 'admin',
    'pass'      => 'your_pass'
)
);
```

save_template

Save template.

Parameters

! *template_id* – (optional)

! *The rest of parameters list is same as in **send_message** method*

Response

int template id

send_test_message

Send test message.

Parameters

! *test_recipients* – array of email addresses (required)

! *The rest of parameters list is same as in **send_message** method*

Response

array message id

send_template

Send template.

Parameters

! *template_id* – template id (required)
 ! *inputFilter* – advanced selection filter (optional)
 ! *ip_space* – ip space (optional). To setup different then default *ip_space*
 ! *ip_space_preset* – ip space preset name (optional). To setup different then default *ip_space*. You can't use both *ip_space* and *ip_space_preset* at the same time. Please use only one of them or none.

Response

array message id

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'send_template',
        'args' => array
            (
                'template_id' => 38
            )
    ),
    'host'      => 'www.your_domain.com',
    'uri'       => '/xapi/server.php',
    'port'     => 80,
    'user'     => 'admin',
    'pass'     => 'your_pass'
);
```

send_draft

Send draft.

Parameters

! *draft_id* – draft id (required)

Response

array message id

Example

```

$result = xu_rpc_http_concise(
    array(
        'method' => 'send_draft',
        'args' => array
            (
                'draft_id' => 644
            )
        ,
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);

```


templates

Get templates list.

Parameters

! *user_id* – web user id (optional)
 ! *column* – 'name'/'campaign' – (optional), if set to 'campaign' this method will return campaign names instead of template subjects

Response

array template id => template name

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'templates',
        'args' => array(
            'user_id'=>'11'
        ),
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);
```

drafts

Get drafts list.

Parameters

! *user_id* – web user id (optional)

Response

array draft id => draft name

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'drafts',
        'args' => array(
            'user_id'=>'11'
        ),
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);
```

get_ip_space

Get IP Space.

Parameters

! *user_id* – user id (required)

Response

array (ips => domains)

Example

```

$result = xu_rpc_http_concise(
    array(
        'method' => 'get_ip_space',
        'args' => array
            (
                'user_id' => 15
            )
        ,
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);

```

get_ip_space_presets

Get IP Space presets.

Parameters

! *user_id* – user id (required)

Response

array (preset name => ip space selection)

Example

```

$result = xu_rpc_http_concise(
    array(
        'method' => 'get_ip_space_presets',
        'args' => array
            (
                'user_id' => 15
            )
        ,
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);

```

ip_space_preset_create

Create IP Space preset.

Parameters

! *user_id* – user id (required)
 ! *name* – *ip_space_preset* name
 ! *ip_space* – IP Space

Response

ip_space_preset ID

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'ip_space_preset_create',
        'args' => array
            (
                'user_id' => 15,
                'name' => 'test_123',
                'ip_space' =>
                    '*.example.com,192.168.168.*,192.168.170.*,mx1.testdomain.com'
            )
    ),
    array(
        'host' => 'www.your_domain.com',
        'uri' => '/xapi/server.php',
        'port' => 80,
        'user' => 'admin',
        'pass' => 'your_pass'
    )
);
```

ip_space_preset_delete

Delete IP Space preset.

Parameters

- ! *user_id* – user id (required)
- ! *name* – ip_space_preset name

Response

1/0

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'ip_space_preset_delete',
        'args' => array
            (
                'user_id' => 15,
                'name' => 'test_123'
            )
    ),
    array(
        'host' => 'www.your_domain.com',
        'uri' => '/xapi/server.php',
        'port' => 80,
        'user' => 'admin',
        'pass' => 'your_pass'
    )
);
```

get_filter_presets

Get filter presets.

Parameters

! *user_id* – user id (required)

Response

array (preset name => filter query)

Example

```

$result = xu_rpc_http_concise(
    array(
        'method' => 'get_filter_presets',
        'args' => array
            (
                'user_id' => 15
            )
        ,
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);

```

mailing_lists

Get mailing list definition.

Parameters

! *user_id* – web user id (required)

Response

array mailing list definition

Example

```
$result = xu_rpc_http_concise(  
    array(  
        'method' => 'mailing_lists',  
        'args'   => array('user_id'=>11),  
        'host'   => 'www.your_domain.com',  
        'uri'    => '/xapi/server.php',  
        'port'   => 80,  
        'user'   => 'admin',  
        'pass'   => 'your_pass'  
    )  
);
```


mailing_lists_create

Create mailing list.

Parameters

- ! *user_id* – web user id (required)
- ! *list_name* – list name (required)
- ! *from_name* – from name (required)
- ! *from_email_address* – from email address (required)
- ! *header* – message header id (optional)
- ! *footer* – message footer id (optional)
- ! *disabled_delivery* - 1/0 (optional)

Response

mailing list id

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'mailing_lists_create',
        'args'   => array(
            'user_id' => '11',
            'list_name' => 'List name',
            'from_name' => 'neko',
            'from_email_address' => 'from@your_domain.com',
            'header' => 9,
            'footer' => 115,
            'disabled_delivery' => 0
        ),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

mailing_lists_update

Update mailing list.

Parameters

! *user_id* – web user id (required)
 ! *list_id* – mailing list id (required)
 ! *list_name* – list name (optional)
 ! *from_name* – from name (optional)
 ! *from_email_address* – from email address (optional)
 ! *header* – message header id (optional)
 ! *footer* – message footer id (optional)
 ! *disabled_delivery* - 1/0 (optional)

Response

1/0

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'mailing_lists_update',
        'args'   => array(
            'user_id' => '11',
            'list_id' => 148,
            'list_name' => 'List name',
            'from_name' => 'neko',
            'from_email_address' => 'from@your_domain.com',
            'header' => 9,
            'footer' => 115,
            'disabled_delivery' => 0
        ),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

mailing_lists_delete

Delete mailing list.

Parameters

- ! *user_id* – web user id (required)
- ! *list_id* – mailing list id (required)

Response

job id

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'mailing_lists_delete',
        'args'   => array(
            'user_id' => '11',
            'list_id' => 148
        ),
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);
```

mailing_lists_empty

Empty mailing list.

Parameters

- ! *user_id* – web user id (required)
- ! *list_id* – mailing list id (required)

Response

job id

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'mailing_lists_empty',
        'args'   => array(
            'user_id' => '11',
            'list_id' => 148
        ),
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);
```

mailing_lists_merge

Merge mailing lists.

Parameters

- ! *user_id* – web user id (required)
- ! *inputMergeLists* – array of list id(s) (required)
- ! *inputListMergeTo* – list id merge to (required)
- ! *inputLeaveRecipients* – 1/0 (optional)

Response

array job id

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'mailing_lists_merge',
        'args'   => array(
            'user_id' => '11',
            'inputMergeLists' => array(72, 74),
            'inputListMergeTo' => 73,
            'inputLeaveRecipients' => 1
        ),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

mailing_lists_split

Split mailing list.

Parameters

- ! *user_id* – web user id (required)
- ! *inputSplitListsSource* – list id to split (required)
- ! *inputSplitLists* – array of arrays of list ids and sizes (required)

Response

array job id

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'mailing_lists_split',
        'args'   => array(
            'user_id' => '11',
            'inputSplitListsSource' => 54,
            'inputSplitLists' => array(array('list_id' => 72,
                'size' => 200), array('list_id' => 74, 'size' => 100))
        ),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

headers

Get headers list.

Parameters

! *user_id* – web user id (required)

Response

array header id => header name

Example

```

$result = xu_rpc_http_concise(
    array(
        'method' => 'headers',
        'args' => array(
            'user_id'=>'11'
        ),
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);

```

footers

Get footers list.

Parameters

! *user_id* – web user id (required)

Response

array footer id => footer name

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'footers',
        'args' => array(
            'user_id'=>'11'
        ),
        'host'    => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);
```


job_details

Get job details.

Parameters

! *job_id* – *job id (required)*

Response

```
array (
    ! _status – status code
    ! _status_name – status name
)
```

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'job_details',
        'args'   => array('job_id'=>2987880),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

job_cancel

Cancel job

Parameters

! *job_id* – job id (required)

Response

1 or 0

Example

```
$result = xu_rpc_http_concise(  
    array(  
        'method' => 'job_cancel',  
        'args'   => array('job_id'=>2987880),  
        'host'   => 'www.your_domain.com',  
        'uri'    => '/xapi/server.php',  
        'port'   => 80,  
        'user'   => 'admin',  
        'pass'   => 'your_pass'  
    )  
);
```

job_pause

Pause job

Parameters

! *job_id* – job id (required)

Response

1 or 0

Example

```
$result = xu_rpc_http_concise(  
    array(  
        'method' => 'job_pause',  
        'args'   => array('job_id'=>2987880),  
        'host'   => 'www.your_domain.com',  
        'uri'    => '/xapi/server.php',  
        'port'   => 80,  
        'user'   => 'admin',  
        'pass'   => 'your_pass'  
    )  
);
```

job_resume

Resume job

Parameters

! *job_id* – job id (required)

Response

1 or 0

Example

```
$result = xu_rpc_http_concise(  
    array(  
        'method' => 'job_resume',  
        'args'   => array('job_id'=>2987880),  
        'host'   => 'www.your_domain.com',  
        'uri'    => '/xapi/server.php',  
        'port'   => 80,  
        'user'   => 'admin',  
        'pass'   => 'your_pass'  
    )  
);
```

file_export

File export.

Parameters

```
!      user_id – web user id (required)
!      selected_lists – array of list ids to export (required), also available:
!      -1 – all user lists
!      -2 – unsubscribed
!      -3 – temporary bounces
!      -4 – permanent bounces
!      -5 – blocked bounces
!      selected_fields – array of fields to export (optional) – id empty array all
fields will be exported
!      suppression_lists – array of suppression lists ids (optional)
!      suppression_files – array of suppression files (optional)
!      sel_field_delimiter – field delimiter - 'tab'/'comma'/'pipe' (optional)
!      chk_fields_in_quotes – check fields in quotes 1/0 (optional)
!      export_time – scheduled time (optional) 'mm/dd/yyyy hh:mm:ss'
!      opt_filename – export filename
!      inputFilter – advanced selection filter
```

Response

```
array      job ids
```

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'file_export',
        'args'   => array(
            'user_id' => '11',
            'selected_lists' => array(45,72,74),
            //'selected_lists' => array(45,72,74,-2,-3),
            'selected_fields' => array('email','track_ip'),
            'suppression_lists' => array(46,73,75),
            'suppression_files' => array('46@11','47'),
            'sel_field_delimiter' => 'tab', //comma,tab,pipe
            'chk_fields_in_quotes' => '1', // 1,0
            'export_time' => '09/29/2011 02:12:00',
            'opt_filename' => 'export-file123456',
            'inputFilter' => "(field(email) in
('test@example.com', 'test2@example.com'))"
        ),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

file_import

File import.

Parameters

```
!      user_id – web user id (required)
!      import_into_lists - array of arrays of list ids and sizes (required)
!      import_fields - array of arrays of name, type and capitalize (required)
!      name can takes values: 'email', 'track_ip', 'first_name', 'last_name', 'site',
import_source', 'skip'
!      type: 'string', 'date', 'number'
!      capitalize: 1/0
!      field_delimiter – 'comma'/'tab'/'pipe'
!      import_file – file for import, located in Import FTP directory (required)
!      dir – folder in Import FTP directory (optional)
!      import_time - scheduled time (optional) 'mm/dd/yyyy hh:mm:ss'
!      chk_no_update - Don't import updates 1/0
!      chk_subsc_again - Import unsubscribed recipients 1/0
!      chk_ovr – Overwrite 1/0
!      chk_blocked_client_addresses - Don't import banned recipients 1/0
!      chk_blocked_client_masks - Don't import banned masks 1/0
!      chk_blocked_addresses - Don't import global banned recipients 1/0
!      chk_blocked_masks - Don't import global banned masks 1/0
!      chk_bo_temp - Don't import temporary bounces 1/0
!      chk_bo_perm - Don't import permanent bounces 1/0
!      chk_bo_bl - Don't import blocked bounces 1/0
!      chk_bo_rem_rec - Don't import removed bounces 1/0
```

Response

```
array      job ids
```

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'file_import',
        'args'   => array(
            'user_id' => '11',
            'import_into_lists' => array(array('list_id' => 68,
            'size' => 10000), array('list_id' => 69, 'size' =>
20000), array('list_id' => 70, 'size' => 30000)),
            'import_fields' => array(array('name' => 'email',
'type' => 'string', 'capitalize' => true),
array('name' => 'first_name', 'type' => 'string', 'capitalize' => true),
array('name' => 'skip', 'type' => 'skip')),
            'field_delimiter' => 'comma', // comma/tab/pipe
            'import_file' => 'botmail.txt',
            'chk_no_update' => '0',
            'chk_subsc_again' => '0',
            'chk_ovr' => '0',
            'chk_blocked_client_addresses' => '1',
            'chk_blocked_client_masks' => '1',
            'chk_blocked_addresses' => '1',
            'chk_blocked_masks' => '1',
            'chk_bo_temp' => '1',
            'chk_bo_perm' => '1',
            'chk_bo_bl' => '1',
            'chk_bo_rem_rec' => '1',
            'import_time' => '10/05/2011 18:12:00'
        ),
        'host'   => 'www.your_domain.com',
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'user'    => 'admin',
        'pass'    => 'your_pass'
    )
);
```

file_import_unsubscribe

File import unsubscribe.

Parameters

```
!      user_id – web user id (required)
!      unsubscribe_from_lists - array of list ids (required) . If the array is empty,
emails will be unsubscribed from all lists.
!      import_fields - array of arrays of name, type and capitalize (required)
      name can takes values: 'email', 'track_ip', 'first_name', 'last_name', 'site',
import_source', 'skip'
      type: 'string', 'date', 'number'
      capitalize: 1/0
!      field_delimiter – 'comma'/'tab'/'pipe'
!      import_file – file for import (required)
!      import_time - scheduled time (optional) 'mm/dd/yyyy hh:mm:ss'
!      chk_skip_insert - Skip inserts for unsubscribed recipients 1/0
!      chk_no_update - Don't import updates 1/0
!      chk_subsc_again - Import unsubscribed recipients 1/0
!      chk_ovr – Overwrite 1/0
!      chk_blocked_client_addresses - Don't import banned recipients 1/0
!      chk_blocked_client_masks - Don't import banned masks 1/0
!      chk_blocked_addresses - Don't import global banned recipients 1/0
!      chk_blocked_masks - Don't import global banned masks 1/0
!      chk_bo_temp - Don't import temporary bounces 1/0
!      chk_bo_perm - Don't import permanent bounces 1/0
!      chk_bo_bl - Don't import blocked bounces 1/0
!      chk_bo_rem_rec - Don't import removed bounces 1/0
```

Response

```
array      job ids
```


Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'file_import',
        'args'   => array(
            'user_id' => '11',
            'unsubscribe_from_lists' => array(55, 69, 70),
            'import_fields' => array(array('name' => 'email',
                'type' => 'string', 'capitalize' => true),
                array('name' => 'first_name', 'type' => 'string', 'capitalize' => true),
                array('name' => 'skip', 'type' => 'skip')),
            'field_delimiter' => 'comma', // comma/tab/pipe
            'import_file' => 'botmail.txt',
            'chk_skip_insert' => '0',
            'chk_no_update' => '0',
            'chk_subsc_again' => '0',
            'chk_ovr' => '0',
            'chk_blocked_client_addresses' => '1',
            'chk_blocked_client_masks' => '1',
            'chk_blocked_addresses' => '1',
            'chk_blocked_masks' => '1',
            'chk_bo_temp' => '1',
            'chk_bo_perm' => '1',
            'chk_bo_bl' => '1',
            'chk_bo_rem_rec' => '1',
            'import_time' => '10/05/2011 18:12:00'
        ),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

tracked_links_actions

Get tracked links actions.

Parameters

! *user_id* – web user id (required)

Response

array tracked links actions

Example

```
$result = xu_rpc_http_concise(  
    array(  
        'method' => 'tracked_links_actions',  
        'args'   => array('user_id'=>11),  
        'host'   => 'www.your_domain.com',  
        'uri'    => '/xapi/server.php',  
        'port'   => 80,  
        'user'   => 'admin',  
        'pass'   => 'your_pass'  
    )  
);
```

easy_links_keywords

Get easy links keywords.

Parameters

! *user_id* – web user id (required)

Response

array easy links keywords

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'easy_links_keywords',
        'args'   => array('user_id'=>11),
        'host'   => 'www.your_domain.com',
        'uri'    => '/xapi/server.php',
        'port'   => 80,
        'user'   => 'admin',
        'pass'   => 'your_pass'
    )
);
```

suppression_files

Get suppression files list.

Parameters

! *user_id* – web user id (required)

Response

array suppression files list

Example

```
$result = xu_rpc_http_concise(  
    array(  
        'method' => 'suppression_files',  
        'args'   => array('user_id'=>11),  
        'host'   => 'www.your_domain.com',  
        'uri'    => '/xapi/server.php',  
        'port'   => 80,  
        'user'   => 'admin',  
        'pass'   => 'your_pass'  
    )  
);
```

banned_add

Add banned email or mask.

Parameters

! *emails* – comma separated emails and/or masks (required)

! *user_id* – web user id (optional, if not set *user_id* email/mask will be banned for all user accounts)

! *do_mask_unsub* – submits job into banned queue (optional) – it will only work with *user_id* and if there are some masks in *emails* parameter (emails with at least one %)

Response

```
array (
    'saved_emails' => ,
    'saved_masks' => ,
    'duplicates' => ,
    'bad_entries' => ,
    'job_id' =>
)
```

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'banned_add',
        'args' => array(
            'user_id' => 11,
            'emails' => 'testname1@example.com, te%st%2@example.com',
            'do_mask_unsub' => 1
        ),
        'host' => 'www.your_domain.com',
        'uri' => '/xapi/server.php',
        'port' => 80,
        'user' => 'admin',
        'pass' => 'your_pass'
    )
);
```

throttling_configuration

Throttling configuration.

Parameters

Response

```
array      (throttling_id => array(
                data => array(
                    max_conn,
                    max_conn_per_ip,
                    max_rate,
                    max_rate_per_ip
                ),
            ipspace,
            isp
        )
    )
```

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'throttling_configuration',
        'args'   => array(),
        'host'    => $host,
        'uri'     => '/xapi/server.php',
        'port'    => 80,
        'debug'   => 0,
        'user'    => $user,
        'pass'    => $pass
    )
);
```

throttling_available_ranges

Throttling - available ranges.

Parameters

Response

```
array (
    available_ranges
)
```

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'throttling_available_ranges',
        'args' => array(),
        'host' => $host,
        'uri' => '/xapi/server.php',
        'port' => 80,
        'debug' => 0,
        'user' => $user,
        'pass' => $pass
    )
);
```

throttling_isp_list

Throttling – ISP list.

Parameters

Response

```

array (
    ISP => array(
        details,
        name
    )
)

```

Example

```

$result = xu_rpc_http_concise(
    array(
        'method' => 'throttling_isp_list',
        'args' => array(),
        'host' => $host,
        'uri' => '/xapi/server.php',
        'port' => 80,
        'debug' => 0,
        'user' => $user,
        'pass' => $pass
    )
);

```


throttling_available_parameters

Throttling - available parameters.

Parameters

Response

```

array (
    parameter => array(
        description,
        title,
        units
    )
)

```

Example

```

$result = xu_rpc_http_concise(
    array(
        'method' => 'throttling_available_parameters',
        'args' => array(),
        'host' => $host,
        'uri' => '/xapi/server.php',
        'port' => 80,
        'debug' => 0,
        'user' => $user,
        'pass' => $pass
    )
);

```

throttling_modify

Throttling – add or update

Parameters

```
!      id – throttling id (optional, required for update)
!      ipspace – select from throttling_available_ranges
!      isp – select from throttling_isp_list
!      data – array with parameters from throttling_available_parameters
```

Response

```
array (
    'affected' => ,
    'id' => ,
    'overrides' => array()
)
```

Example

```
$result = xu_rpc_http_concise(
    array(
        'method' => 'throttling_modify',
        'args' => array
            (
                'id' => 478,
                'ipspace' => '192.168.161.1-53',
                'isp' => 'GI',
                'data' => array
                    (
                        'max_conn' => 1,
                        'max_conn_per_ip' => 2
                    )
            )
        ,
        'host' => $host,
        'uri' => '/xapi/server.php',
        'port' => 80,
        'debug' => 0,
        'user' => $user,
        'pass' => $pass
    )
);
```

throttling_delete

Throttling – delete

Parameters

! *id – throttling id (required)*

Response

1/0

Example

```

$result = xu_rpc_http_concise(
    array(
        'method' => 'throttling_delete',
        'args' => array
            (
                'id' => '478'
            )

        'host'    => $host,
        'uri'      => '/xapi/server.php',
        'port'     => 80,
        'debug'    => 0,
        'user'     => $user,
        'pass'     => $pass
    )
);

```