

Postman collection: <https://documenter.getpostman.com/view/34603208/2sB3BEoW7A>

ClientTether API

Developer Guide & Reference

Client API v2 | Documentation edition 2026.10.22 | October 22, 2026

Build reliable integrations for contact intake, sales workflows, communications, scheduling and reporting. This guide provides a practical starting path, reusable request patterns and a complete reference for the two-header ClientTether API documented in this edition.

Choose your starting point

I want to...	Start here
Make my first request	1. Quick start
Connect multiple locations	2. Authentication and account context
Synchronize contacts or opportunities	4. Data model; 5. Integration recipes
Book an appointment	5. Hosted scheduling recipe; 8. Scheduling reference
Use an AI to build an integration	6. Working with AI; Markdown and OpenAPI downloads
Find a method, field or response	8. Endpoint reference
Diagnose an error	3. API conventions; 10. Troubleshooting

Contents

1. Quick start	3
2. Authentication and account context	5
3. API conventions	6
4. Data model and field dictionary	9
5. Integration recipes	15
6. Working with AI and the downloadable tools	17
7. Reusable client examples	18
8. Endpoint reference	20
9. Webhooks and workflow automation	68
10. Troubleshooting	69
11. Compatibility and reference notes	71

1. Quick start

Before you begin

You need an API access token, the web key for the account you will access, and the account timezone. Obtain credentials from your ClientTether account administrator through Settings -> API. Keep them on the server or in a private local environment. Use contact IDs and configuration IDs from that same account.

Every endpoint in this guide uses this base URL:

```
https://api.clienttether.com/v2/api
```

Every request requires both headers:

```
X-Access-Token: YOUR_API_ACCESS_TOKEN
X-Web-Key: YOUR_ACCOUNT_WEB_KEY
Accept: application/json
```

Make your first request

Start with a read-only contact lookup. Replace the example phone number with a contact you intend to look up.

```
export CT_ACCESS_TOKEN='YOUR_API_ACCESS_TOKEN'
export CT_WEB_KEY='YOUR_ACCOUNT_WEB_KEY'
curl --get 'https://api.clienttether.com/v2/api/read_client_exist' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Accept: application/json' \
  --data-urlencode 'phone=2025550147'
```

A matching response contains CT_200 and one or more record identifiers. This illustrative response shows a single match:

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "TotalRecord": 1,
  "data": [
    {
      "client_id": 100001,
      "external_id": "900001"
    }
  ]
}
```

Use the returned client_id to retrieve the record:

```
curl 'https://api.clienttether.com/v2/api/read_client_by_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Accept: application/json'
```

If no contact matches, inspect CT_515. Do not treat the presence of a data array as proof of a match; the no-match lookup can include empty IDs. When several contacts match, resolve the correct record before proceeding.

Create a contact

Create requires firstName and lastName. Send SMS permission and the new-lead alert choice explicitly. The example deliberately disables SMS permission and new-lead alerts; account automation can still apply.

```
curl 'https://api.clienttether.com/v2/api/create_client' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY" \
--header 'Content-Type: application/json' \
--data '{
  "firstName": "Jamie",
  "lastName": "Example",
  "phone": "2025550147",
  "email": "jamie@example.com",
  "smsok": 0,
  "new_lead_notification": 0
}'
```

Replace example contact details with values appropriate to your workflow. Save the returned `client_id`, then read the contact back. For ongoing syncs, use the lookup/create/update recipe in Chapter 5 rather than creating a record on every event.

Your next requests

Task	Endpoint
Read account users	GET read_user_list
Read lead sources	GET read_lead_source_list
Read sales-cycle steps	GET read_sales_cycle_list
Read Action Plans	GET read_action_plan_list
Discover scheduling pages	GET read_hosted_pages
Record an outcome	POST create_client_history_note

2. Authentication and account context

The two-header contract

All callable endpoints documented here belong to the public /v2/api interface and require X-Access-Token and X-Web-Key together. They do not require a browser login, session cookie or CSRF token. Use an API access token issued for integration access; do not substitute an application session token.

Header	Purpose	How to use it
X-Access-Token	Authorizes API access	Send the ClientTether-issued token as the header value.
X-Web-Key	Selects the account context	Use the web key for the intended account/location.
Content-Type	Declares request encoding	Use application/json for the JSON POST examples.
Accept	Requests JSON responses	Use application/json and still guard response parsing.

Authentication and account selection are separate responsibilities. A web key is a credential, not an account ID. A contact ID, user ID or proposal ID does not replace the web key.

Multi-location and enterprise integrations

Maintain an explicit mapping from each tenant or location in your system to its ClientTether account ID and web key. Select that context before performing a lookup or write. Keep record identifiers qualified by account so that two locations never share an accidental mapping.

Enterprise-level access can use `read_account_list` to discover permitted child accounts. The returned `account_web_key` is sensitive. Store it securely and use it only for the authorized location. Do not assume the account list is a recursive export of every account in the network.

Credential lifecycle

Obtain current credentials through the account administrator. When credentials stop working, verify the current token and account web key before retrying. Coordinate replacement credentials with the administrator and update your secret store. The API does not expose a login/refresh workflow in this guide.

Keep credentials private

- Store credentials in a server-side secret manager or a private local Postman environment.
- Keep them out of URLs, browser bundles, source control, shared collections and logs.
- Redact headers before submitting request examples to Support or an AI assistant.
- Use a separate configured secret for your webhook receiver; do not reuse your API token as a receiver password.

3. API conventions

Methods, encoding and parameter locations

Method	Parameter location	Typical use
GET	Path and query string	Lookups and lists. See the <code>create_quote</code> exception.
POST	JSON body	Creates, updates, sending and selected report/scheduling reads.
DELETE	Path or documented query selector	Deletion of a record or assignment.

Use the exact method in the endpoint reference. `read_available_timeslots`, `read_pipeline_list` and `read_sales_cycle_any_activity` use POST. `create_quote` uses GET but creates data. Do not infer behavior from either the name or HTTP method alone.

POST examples use `application/json`. Form-encoded requests can use `application/x-www-form-urlencoded` where represented in the OpenAPI file. Send a key in one location only; avoid mixing query and body values for the same input. Field names and identifier names matter: `firstName` is not `first_name`, and delete selectors can differ from update selectors.

Read HTTP status and ResultCode

A successful HTTP exchange can still contain an application failure. Always inspect `ResultCode`. Conversely, a routing or server failure may return non-JSON content. Read the status, attempt JSON parsing safely and preserve a redacted diagnostic if parsing fails.

ResultCode	Meaning	Integration action
CT_200	Success	Read returned IDs, warnings and any per-record results.
CT_501	Required input missing	Correct the request; inspect <code>Message</code> for the missing context.
CT_511	Invalid API request or credentials/context	Check both access headers and the operation selector.
CT_513	Access denied	Check account access with your administrator.
CT_515	No record found	Handle as an expected lookup result where appropriate.
CT_520	Duplicate entry	Perform a fresh lookup and reconcile the mapping.
CT_525	Validation or insert failure	Use <code>Message</code> to correct input; do not replay blindly.
CT_512	Record remains in use	Remove/reassign references before deletion.
CT_530	Tag unavailable to the account	Read or create an available tag before assignment.
CT_534	Scheduling block unavailable	Refresh availability and select a new block.

ResultCode	Meaning	Integration action
CT_550	Operation unavailable to the account	Contact Support about account feature access.
CT_500	Unknown error	Preserve redacted diagnostics and investigate before replaying a write.

Scheduling errors may use CT_501 with a message explaining that the hosted page, block, contact phone or address is missing. Use the actual ResultCode and Message together rather than interpreting message text as a separate numeric HTTP status.

Response shapes

Operation	Important response keys
Contact lookup	ResultCode, TotalRecord, data[]
Read one contact	ResultCode, data object
Contact list	TotalRecord, dataRecord, data[]
Create/update contact	client_id, external_id
Bulk history insert	Response[] with record, client_id, result and optional reason
Hosted pages and time slots	Direct array on success; error object on failure
Custom fields	Client_Id and capitalized Data
Proposal output	company, customer, proposals
Event create/update	Event_id; event lists use event_id

Examples are synthetic and abbreviated. Additional response fields can be present. Preserve null and empty values; do not silently replace them with zero. Numeric identifiers can be serialized as numbers or strings. Build parsers around the fields your integration needs rather than requiring every response to match a single universal envelope.

Pagination and incremental reads

Endpoint	Paging behavior
read_client_list / read_contacts	limit 1-100, offset starting at 0
read_opportunity_list	Ordinary limit 1-100; larger export mode uses mega_payload
read_all_events	100 records per page; advance offset
read_all_payments	Maximum 250 entries; no documented public paging parameter
read_client_history_notes	No documented cursor/paging parameter

For incremental contact syncs, use `last_updated_date` with a small overlap window and deduplicate by account plus `client_id`. Store your high-water mark after the destination commit succeeds. Offset scans are not transactional snapshots: concurrent changes and readback delays can affect results.

Timezones and date formats

Input/output	Convention
Event and scheduled-message date/time	Account-local date and time
History <code>utc_datetime</code>	UTC ISO timestamp, such as 2026-10-22T16:00:00Z
Contact created/modified filters	UTC date/time interpretation
Hosted slots	Use returned <code>time_zone</code> plus local/UTC fields
Calendar zone identifier	Use IANA names such as America/Denver
Plain date	YYYY-MM-DD
Plain time	HH:MM:SS

Keep the account timezone in your integration configuration. Do not interpret an account-local event time as UTC or use a fixed UTC offset year-round. For hosted booking, preserve the server-issued `block_id` and use the returned time representations for display. Day offsets cross UTC/local date boundaries; specify start and end explicitly.

Retries, request pacing and duplicate prevention

Use bounded retries for read-only network/server failures. Pace requests and handle HTTP 429; honor `Retry-After` when it is supplied. Contact Support to coordinate high-volume workloads rather than assuming a universal quota.

Do not automatically replay a create, send, booking, payment entry or destructive request after a timeout. First determine whether the original operation took effect. Maintain a local operation ledger with account, external event ID, request purpose, returned record ID and outcome. An external reference in a history note helps reconciliation but does not create server-enforced idempotency.

4. Data model and field dictionary

Understand the identifiers

Concept	Identifier	Use
Account	account_id + account web key	Account ID identifies; web key selects API context.
Contact	client_id	Parent contact record.
Opportunity	Its own client_id plus parent_id	A child record under a contact.
External record	external_id	Your integration mapping, qualified by account.
Multiple external systems	third_party_name + third_party_id	A named external mapping pair.
Assigned user	assigned_user_id	Contact owner/assignment.
Sales-cycle step	sales_cycle_id for writes	Step placement; read labels may differ.
Action Plan	action_plan_id	Automation assignment.
Hosted scheduling page	hosted_page_id / hosted_id	Availability / booking respectively.
Event	event_id / Event_id	Event list and create/update response casing differ.
Proposal	proposal_id or quote_id	Proposal/payments or configured quote engine.

Contact fields

Use this shared dictionary with `create_client`, `update_client_by_id`, `update_client_by_external_id` and `create_opportunity`. Required fields and selectors are specified with each operation. Omit optional fields you do not intend to change. Do not replay an entire read response as an update.

Identity and contact details

Field	Type	Description
external_id	string	Your external-system record ID. Keep the mapping qualified by account.
firstName	string	Contact first name; nonempty on contact creation. Case-sensitive field name.
lastName	string	Contact last name; nonempty on contact creation. Case-sensitive field name.
compName	string	Contact company name.
contact_type	integer	Contact type: 1 Client, 2 Employee, 3 Partner, 4 Vendor, 5 Other. List endpoints accept 0 for all five.
job_title	string	Contact job title.
email	string	Contact email address. Use an address you control for sending tests; URL-encode plus signs in query/form data.
phone	string	Primary/mobile phone. For US/Canada send a ten-digit national number as a string without punctuation.
extCell	string	Extension for the primary phone.
second_phone	string	Secondary phone; send as an unformatted national-number string.
extHome	string	Extension for the secondary phone.
smsok	integer	SMS permission: 0=no, 1=yes. Set explicitly; contact creation otherwise defaults to 1.
address	string	Contact street address.
city	string	Contact city.
state	string	Contact state/province. Use a two-letter code for portability; in opportunity filters this name has a different meaning.
zip	string	Postal code as a string, retaining leading zeros.
assigned_user_id	integer	User ID to assign to the contact; read the target account user list first.

Workflow and commercial details

Field	Type	Description
lead_source_id	integer	Lead source ID within the target account. May select mapped connection automation.
sales_cycle_id	integer	Sales-cycle step ID for writing a contact; read responses may call this cycle_steps_id or cycle_id.
action_plan_id	integer	Action Plan ID to assign/start. This changes automation and can remove pending actions. See workflow side effects.
tag	string	Comma-separated tag names. Contact writes find/create tags and assign them. Prefer exact existing names.
client_category	string	Configured contact category name. Use the category vocabulary defined for the account.
new_lead_notification	integer	Contact creation alert: 0 or omitted=none; 1=in-app; 2=in-app plus configured email/text alerts.
deal_size	number	Deal value; numeric. Confirm precision/currency conventions in the target account.
deal_quoted	integer	Quoted milestone flag. Writes can trigger workflow and proposal side effects.
deal_closed	integer	Closed milestone flag. Writes can trigger workflow and proposal side effects.
deal_completed	integer	Completed milestone flag. Writes can trigger workflow and proposal side effects.

Notes, mapping and dates

Field	Type	Description
whiteboard	string	Replaces the contact whiteboard content. Use <code>append_whiteboard</code> to add text instead.
append_whiteboard	string	Appends text to the existing whiteboard. Replaying this write duplicates content.
third_party_name	string	Integration/system name used together with <code>third_party_id</code> .
third_party_id	string	External identifier under <code>third_party_name</code> . Supply the pair together.
pin	string	Contact PIN value.
created	string	Contact creation timestamp, YYYY-MM-DD HH:MM:SS. Use a consistent timezone policy and check the stored UTC value.
clients_date_next	string	Contact date value, YYYY-MM-DD; API stores a UTC-midnight representation. Verify date-field automation effects.
clients_date_last	string	Contact date value, YYYY-MM-DD; API stores a UTC-midnight representation. Verify date-field automation effects.
anniversary_1	string	Contact date value, YYYY-MM-DD; API stores a UTC-midnight representation. Verify date-field automation effects.
anniversary_2	string	Contact date value, YYYY-MM-DD; API stores a UTC-midnight representation. Verify date-field automation effects.

Additional profile fields

Field	Type	Description
high_net_worth	string	Optional contact profile value for high net worth. Use the account profile vocabulary.
home_owner_status	string	Optional contact profile value for home owner status. Use the account profile vocabulary.
household_income	string	Optional contact profile value for household income. Use the account profile vocabulary.
length_of_residence	string	Optional contact profile value for length of residence. Use the account profile vocabulary.
marital_status	string	Optional contact profile value for marital status. Use the account profile vocabulary.
market_value	string	Optional contact profile value for market value. Use the account profile vocabulary.
occupation	string	Optional contact profile value for occupation. Use the account profile vocabulary.
social	string	Optional contact profile value for social. Use the account profile vocabulary.
age	string	Optional contact profile value for age. Use the account profile vocabulary.
education	string	Optional contact profile value for education. Use the account profile vocabulary.
gender	string	Optional contact profile value for gender. Use the account profile vocabulary.
presence_of_children	string	Optional contact profile value for presence of children. Use the account profile vocabulary.
presence_of_pets	string	Optional contact profile value for presence of pets. Use the account profile vocabulary.
facebook	string	Contact social-profile value.
linkedIn	string	Social profile value. Read-side labels are not a general update schema; use configured custom data if your integration requires a field not available for native update.
twitter	string	Contact social-profile value.

Contact types: 1 Client, 2 Employee, 3 Partner, 4 Vendor, 5 Other. Listing endpoints can use 0 to include all five. Contact creation defaults SMS permission to 1 when smsok is omitted; choose it explicitly based on your workflow.

Custom fields

Custom-field writes use the exact configured label as a top-level key on the contact request. Read the labels and current values through `read_custom_field_by_contact`. A misspelled native key can be treated as custom data, so send only defined native fields and intentional custom labels.

```
{
  "client_id": 100001,
  "Preferred Service": "Exterior painting"
}
```

Native UTM fields

Contact reads expose `utm_source`, `utm_medium`, `utm_campaign`, `utm_content` and `utm_term`. The generic contact-write operations in this guide do not provide native UTM updates. A custom field with the same label is not the native attribution field. Use the account-supported attribution capture workflow when native UTM storage is required.

Read labels and write labels differ

Read responses can include `opportunityName`, `apptSet`, `quoted`, `closed`, `completed`, `clients_action_plan`, `clients_lead_source` and `clients_sales_cycle`. These are not automatic write keys. `assigned_user` can be serialized data rather than a nested object. Use the explicit input fields for the operation you are calling.

Workflow side effects

Changing `action_plan_id` can replace pending automation. A lead source can start a mapped connection plan, and a sales-cycle transition can start conversion follow-up. `whiteboard` replaces content; `append_whiteboard` appends it. Starting a plan and separately sending the same message can duplicate outreach. Design the integration around the account workflow rather than treating every field as passive storage.

5. Integration recipes

Create or update a lead

1. Resolve the target location and choose its web key.
2. Look up the contact using a durable external mapping or a meaningful phone/email filter.
3. For an unambiguous match, read the record and update only changed fields.
4. For a confirmed no-match, create the contact with required names and explicit consent/notification choices.
5. Store the account-qualified client_id mapping.
6. Read back important fields and record the external operation reference in your ledger.

This is a read/create/update pattern, not an atomic upsert. Coordinate concurrent writers and the account duplicate-check setting.

Create a second opportunity

1. Read the parent contact in the selected account.
2. POST create_opportunity with parent_id and opportunity_name.
3. Store the new client_id as the opportunity ID and retain the parent ID separately.
4. Use the opportunity record for the intended sales workflow and proposal association.

Do not create another unrelated parent contact merely to represent another job for the same customer.

Book an appointment through a hosted page

1. Read hosted pages and choose the intended meeting type.
2. Read the existing contact and complete required scheduling phone/address data, or supply prospect context.
3. POST read_available_timeslots with hosted_page_id, explicit start/end and the contact/prospect context.
4. Display the returned times using time_zone and the local/UTC fields. Retain block_id unchanged.
5. POST hosted_page_schedule_meeting with hosted_id, block_id and client_id, or the complete new-attendee field set.
6. Inspect the outcome. Refresh availability after an unavailable-slot response; verify the booked event before confirming completion.

Manual create_client_event does not substitute for hosted-page availability, business-hour, notice, buffer or external-calendar rules.

Give an AI conversation context

1. Resolve the account/location explicitly from your routing map.
2. Look up the contact and read its profile and history before interpreting a short response such as yes or reschedule.
3. Choose users and scheduling pages from account configuration IDs.
4. Use manual send endpoints for authorized outgoing communication; history endpoints only record it.
5. Write a concise outcome with an external reference.
6. Change the cycle/plan only when the business rule calls for it.

Keep ClientTether identifiers authoritative. Your AI event labels and external correlation IDs are integration conventions, not additional API fields.

Synchronize contacts incrementally

1. Set limit=100, offset=0 and the intended active_flag.
2. Use last_updated_date as a UTC modified-since lower bound, with an overlap window.
3. Advance offset and deduplicate returned records by account plus client_id.
4. Commit a checkpoint only after destination persistence succeeds.
5. Reconcile periodically and scan opportunities separately where needed.

This API does not define a universal deletion feed or transactional change stream.

Record and reconcile a payment

1. Identify the correct account, contact and proposal.
2. Consult existing entries and your local operation ledger.
3. POST create_proposal_payment with proposal_id, payday, amount, type, status and a reference when required.
4. Store the response and verify the record.
5. Reconcile with the accounting system; coordinate larger exports when 250 entries could be exceeded.

A recorded payment entry is not a card charge or settlement confirmation.

Send a webhook to your integration

1. Configure the webhook/Action Plan through the account settings UI.
2. Choose receiver URL, method, body encoding and a minimal explicit payload.
3. Include account/contact IDs using the tokens available to the account.
4. Configure a separate receiver authentication header or secret.
5. Store accepted events in your system and process asynchronously.
6. Use Webhook Monitor to reconcile failed or missing work.

Treat delivery order, retry scheduling and event uniqueness as explicit integration design concerns; do not assume them from an HTTP acknowledgment.

6. Working with AI and the downloadable tools

File	Use it for
PDF manual	Readable reference with navigation, request examples and operating guidance.
Markdown manual	Give an AI assistant the same guide as clean text; use it for exact code copying.
OpenAPI specification	Import structured operations, schemas and the two required access headers into development tools.
Postman collection	Inspect and send individual requests using your own account values.

Prompt for an AI integration builder

Use the attached ClientTether Markdown guide and OpenAPI specification.
 Build a server-side integration for: [describe the workflow].
 Account/location mapping: [describe the locations].
 Account timezone: [IANA timezone].
 Use only the documented /v2/api operations and both required access headers.
 Read credentials from environment variables; never embed real secrets.
 Preserve account-qualified IDs and handle ResultCode as well as HTTP status.
 Use explicit notification/consent choices and account-local scheduling times.
 Do not invent fields, endpoint methods, pagination or idempotency headers.
 For a timed-out write, reconcile its outcome before retrying.
 Ask for business choices that the supplied workflow does not establish.
 Produce the field mapping, runnable implementation and setup instructions.

Configure Postman

Import ClientTether-Client-API.postman_collection.json. Set access_token and web_key privately and fill the required record-ID variables from the target account. Run reads individually first. The collection blocks writes until allow_writes is exactly true; set it only for the intended actions, then reset it. The collection also treats create_quote as a write even though its HTTP method is GET. IDs are not populated automatically.

Use OpenAPI

The specification uses OpenAPI 3.1.0. Its security requirement lists AccessToken and WebKey in the same object, meaning both are required. The base server is the public v2 API. Compatibility aliases are included separately; prefer the canonical operations shown in this manual. Response objects permit additional fields, and configured proposal documents remain serialized account-specific data rather than a fabricated universal worksheet model.

7. Reusable client examples

The helpers below perform authenticated requests and distinguish transport failures from application outcomes. They intentionally do not retry writes or interpret every operation as the same response shape. Keep account selection explicit at the call site.

JavaScript / Node.js

```
const API_BASE = 'https://api.clienttether.com/v2/api/';

async function ctRequest(path, { method = 'GET', query, body,
  accessToken = process.env.CT_ACCESS_TOKEN,
  webKey = process.env.CT_WEB_KEY } = {}) {
  if (!accessToken || !webKey) throw new Error('Both credentials are required');
  const url = new URL(path, API_BASE);
  if (url.origin !== new URL(API_BASE).origin ||
    !url.pathname.startsWith('/v2/api/')) {
    throw new Error('Only ClientTether public API paths are allowed');
  }
  for (const [key, value] of Object.entries(query || {})) {
    if (value !== undefined && value !== null) url.searchParams.set(key, String(value));
  }
  const headers = { 'X-Access-Token': accessToken,
    'X-Web-Key': webKey, Accept: 'application/json' };
  if (body !== undefined) headers['Content-Type'] = 'application/json';
  const response = await fetch(url, { method, headers,
    body: body === undefined ? undefined : JSON.stringify(body),
    redirect: 'error', signal: AbortSignal.timeout(30000) });
  const raw = await response.text();
  let data;
  try { data = JSON.parse(raw); }
  catch { throw new Error(`Expected JSON; HTTP ${response.status}`); }
  if (!response.ok) throw new Error(`HTTP ${response.status}`);
  // The caller interprets CT_515, bulk results and endpoint-specific shapes.
  return data;
}

const result = await ctRequest('read_client_exist', {
  query: { phone: '2025550147' }
});
if (result.ResultCode === 'CT_200') {
  const matches = (result.data || []).filter(row => row.client_id);
  if (matches.length !== 1) throw new Error('Resolve contact match explicitly');
  const contact = await ctRequest(`read_client_by_id/${matches[0].client_id}`);
  if (contact.ResultCode !== 'CT_200') throw new Error('Contact read failed');
  // Continue with the account-qualified record.
} else if (result.ResultCode === 'CT_515') {
  // Run the approved no-match/create workflow.
} else {
  throw new Error(`ClientTether ${result.ResultCode} || 'unknown outcome'`);
}
```

Python standard library

```
import json
import os
from urllib.error import HTTPError, URLError
from urllib.parse import urlencode, urljoin, urlparse
from urllib.request import Request, build_opener, HTTPRedirectHandler

API_BASE = "https://api.clienttether.com/v2/api/"

class NoRedirect(HTTPRedirectHandler):
    def redirect_request(self, req, fp, code, msg, headers, newurl):
        return None # Never forward API credentials to a redirected host.

def ct_request(path, *, method="GET", query=None, body=None,
               access_token=None, web_key=None):
    token = access_token or os.environ.get("CT_ACCESS_TOKEN")
    key = web_key or os.environ.get("CT_WEB_KEY")
    if not token or not key:
        raise ValueError("Both API credentials are required")
    url = urljoin(API_BASE, path)
    parsed = urlparse(url)
    if (parsed.scheme != "https" or parsed.netloc != "api.clienttether.com"
        or not parsed.path.startswith("/v2/api/")):
        raise ValueError("Only ClientTether public API paths are allowed")
    if query:
        url += ("&" if "?" in url else "?") + urlencode(query)
    headers = {"X-Access-Token": token, "X-Web-Key": key,
              "Accept": "application/json"}
    payload = None
    if body is not None:
        payload = json.dumps(body).encode("utf-8")
        headers["Content-Type"] = "application/json"
    req = Request(url, data=payload, headers=headers, method=method)
    try:
        with build_opener(NoRedirect()).open(req, timeout=30) as response:
            raw = response.read().decode("utf-8")
    except HTTPError as exc:
        raise RuntimeError(f"ClientTether HTTP {exc.code}") from exc
    except (URLError, TimeoutError) as exc:
        raise RuntimeError("Request failed; reconcile uncertain writes") from exc
    try:
        return json.loads(raw)
    except json.JSONDecodeError as exc:
        raise RuntimeError("Expected a JSON response") from exc

result = ct_request("read_client_exist", query={"phone": "2025550147"})
if result.get("ResultCode") == "CT_515":
    pass # Run the approved no-match/create workflow.
elif result.get("ResultCode") != "CT_200":
    raise RuntimeError(f"Application outcome: {result.get('ResultCode')}")
```

8. Endpoint reference

Every operation below uses both access headers and the same public v2 base URL. Fields marked Required are the complete operational input for the described request; conditional requirements appear in the operation guidance. Examples use synthetic IDs and data. Replace them before execution. For exact code copying, use the matching Markdown file.

Endpoint catalog

Area	Method	Operation
Contacts	GET	read_client_exist
Contacts	GET	read_client_by_id
Contacts	GET	read_client_by_external_id
Contacts	GET	read_client_by_email
Contacts	GET	read_client_list
Contacts	POST	create_client
Contacts	POST	update_client_by_id
Contacts	POST	update_client_by_external_id
Contacts	DELETE	delete_client_by_id
Contacts	DELETE	delete_client_by_external_id
Opportunities	POST	create_opportunity
Opportunities	GET	read_opportunity_list
Opportunities	POST	read_pipeline_list
History	GET	read_client_history_notes
History	POST	create_client_history_note
History	POST	create_history_notes
Messages	POST	create_manual_email
Messages	POST	create_manual_text
Events	GET	read_all_client_events
Events	GET	read_all_events
Events	POST	create_client_event
Events	POST	update_client_event
Events	DELETE	delete_client_event
Scheduling	GET	read_hosted_pages
Scheduling	POST	read_available_timeslots

Area	Method	Operation
Scheduling	POST	hosted_page_schedule_meeting
Action Plans	GET	read_action_plan_list
Action Plans	GET	read_action_plan_by_id
Action Plans	GET	read_actions_by_plan_id
Sales Cycles	GET	read_sales_cycle_list
Sales Cycles	POST	create_sales_cycle
Sales Cycles	POST	update_sales_cycle_by_id
Sales Cycles	DELETE	delete_sales_cycle_by_id
Sales Cycles	POST	read_sales_cycle_any_activity
Lead Sources	GET	read_lead_source_list
Lead Sources	POST	create_lead_source
Lead Sources	POST	update_lead_source_by_id
Lead Sources	DELETE	delete_lead_source_by_id
Tags	GET	read_tags_list
Tags	POST	create_tag
Tags	POST	create_contact_tag_assignment
Tags	GET	read_contact_tags
Tags	DELETE	delete_contact_tag_assignment
Custom Fields	GET	read_custom_field_by_contact
Users	GET	read_user_list
Users	GET	read_user_by_id
Users	POST	create_user
Users	POST	update_user_by_id
Users	DELETE	delete_user_by_id
Accounts	GET	read_account_list
Accounts	POST	update_account_by_id

Area	Method	Operation
Accounts	DELETE	<code>delete_account_by_id</code>
Proposals	GET	<code>read_proposals</code>
Payments	GET	<code>read_proposal_payments</code>
Payments	GET	<code>read_all_payments</code>
Payments	POST	<code>create_proposal_payment</code>
Calling	POST	<code>create_immediate_call</code>
Advanced Proposals	GET	<code>fetch_bid_types</code>
Advanced Proposals	GET	<code>fetch_quote_settings</code>
Advanced Proposals	GET	<code>quote_data_by_id</code>
Advanced Proposals	GET	<code>quote_workorder_data_by_id</code>
Advanced Proposals	GET	<code>create_quote</code>
Advanced Proposals	POST	<code>quote/save_overview</code>
Advanced Proposals	POST	<code>quote/save_worksheet</code>
Advanced Proposals	POST	<code>quote/save_customer_approval</code>
Advanced Proposals	POST	<code>quote/save_workorder</code>
Advanced Proposals	POST	<code>quote/save_changeorder</code>
Advanced Proposals	POST	<code>quote/save_puchlist</code>

Contacts

read_client_exist

GET https://api.clienttether.com/v2/api/read_client_exist

Look up contacts using at least one identifier or search field. All supplied filters must match. A phone-only lookup is often the simplest starting point. On CT_515, data may contain an item with empty IDs; check ResultCode and a nonempty client_id before treating the result as a match. Resolve multiple matches before updating a record.

Parameter	Location	Required?	Type	Description
client_id	query	Optional	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.

Parameter	Location	Required?	Type	Description
firstName	query	Optional	string	Contact first name; nonempty on contact creation. Case-sensitive field name.
lastName	query	Optional	string	Contact last name; nonempty on contact creation. Case-sensitive field name.
phone	query	Optional	string	Primary/mobile phone. For US/Canada send a ten-digit national number as a string without punctuation.
email	query	Optional	string	Contact email address. Use an address you control for sending tests; URL-encode plus signs in query/form data.
external_id	query	Optional	string	Your external-system record ID. Keep the mapping qualified by account.
third_party_name	query	Optional	string	Integration/system name used together with third_party_id.
third_party_id	query	Optional	string	External identifier under third_party_name. Supply the pair together.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_client_exist' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY" \
--data-urlencode 'phone=2025550147'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "TotalRecord": 1,
  "data": [
    {
      "client_id": 100001,
      "external_id": "900001"
    }
  ]
}
```

read_client_by_id

GET https://api.clienttether.com/v2/api/read_client_by_id/{id}

Retrieve a contact or opportunity by its ClientTether record ID. The path form is recommended. The query-only alternative uses id, not client_id.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_client_by_id/100001' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
```

```
--header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": {
    "client_id": 100001,
    "firstName": "Jamie",
    "lastName": "Example",
    "phone": "2025550147",
    "email": "jamie@example.com",
    "smsok": 0,
    "utm_source": null
  }
}
```

Compatibility forms: `read_client_by_id`. Use the selector map in Chapter 11 when calling a query-only form.

read_client_by_external_id

GET `https://api.clienttether.com/v2/api/read_client_by_external_id/{id}`

Retrieve a contact using your external mapping. The path form accepts the external ID. The query-only form accepts `external_id`, or `third_party_name` together with `third_party_id`. Retain the account alongside every external mapping.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_client_by_external_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": {
    "client_id": 100001,
    "firstName": "Jamie",
    "lastName": "Example",
    "phone": "2025550147",
    "email": "jamie@example.com",
    "smsok": 0,
    "utm_source": null
  }
}
```

Compatibility forms: `read_client_by_external_id`. Use the selector map in Chapter 11 when calling a query-only form.

read_client_by_email

GET `https://api.clienttether.com/v2/api/read_client_by_email/{email}`

Retrieve a contact by email in the selected account. URL-encode the email path component, including plus signs and other reserved characters.

Parameter	Location	Required?	Type	Description
email	path	Required	string	Contact email address. Use an address you control for sending tests; URL-encode plus signs in query/form data.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_client_by_email/jamie%40example.com' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": {
    "client_id": 100001,
    "firstName": "Jamie",
    "lastName": "Example",
    "phone": "2025550147",
    "email": "jamie@example.com",
    "smsok": 0,
    "utm_source": null
  }
}
```

Compatibility forms: read_client_by_email. Use the selector map in Chapter 11 when calling a query-only form.

read_client_list

GET https://api.clienttether.com/v2/api/read_client_list

List parent contacts. Default page size is 100; use limit values from 1 to 100 and a zero-based offset. active_flag defaults to 1. Use active_flag=2 when you need an active-state-unfiltered scan. TotalRecord is the matching total; dataRecord is the current-page count. Use read_opportunity_list for opportunity records.

Parameter	Location	Required?	Type	Description
limit	query	Optional	integer	Requested maximum page size. Default/cap depends on the endpoint; see its behavior notes.
offset	query	Optional	integer	Zero-based number of records to skip; use a nonnegative integer.
contact_type	query	Optional	integer	Contact type: 1 Client, 2 Employee, 3 Partner, 4 Vendor, 5 Other. List endpoints accept 0 for all five.
start_date	query	Optional	string	Start date/time filter; on list APIs this usually filters record creation in UTC. On quote overview it is the job start date.
end_date	query	Optional	string	End date/time filter; on list APIs this usually filters record creation in UTC. On quote overview it is the job end date.

Parameter	Location	Required?	Type	Description
active_flag	query	Optional	integer	Contact listing: 1 (default) active; 0 maps to ready=2; values >=2 remove this filter . Prefer 2 for an unfiltered active-state scan.
from_date	query	Optional	string	Lower-bound date/time. Event endpoints interpret this in the account timezone. Contact-list alias for start_date.
to_date	query	Optional	string	Contact/opportunity-list alias for end_date. Avoid sending both aliases and canonical names.
lead_source_id	query	Optional	integer	Lead source ID within the target account. May select mapped connection automation.
last_updated_date	query	Optional	string	Contact last-modified lower bound, interpreted as UTC by the API.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_client_list' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --data-urlencode 'limit=100' \
  --data-urlencode 'offset=0' \
  --data-urlencode 'active_flag=2'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "TotalRecord": 1,
  "dataRecord": 1,
  "data": [
    {
      "client_id": 100001,
      "firstName": "Jamie",
      "lastName": "Example"
    }
  ]
}
```

create_client

POST https://api.clienttether.com/v2/api/create_client

Create a contact with a first and last name. Supply smsok and new_lead_notification explicitly. Lead-source mapping, routing and Action Plans can initiate additional work. Duplicate checking is an account setting; a create request is not an atomic upsert. On CT_520, run a fresh lookup rather than assuming the returned ID is the existing matching contact.

Shared input: Contact fields in Chapter 4. The table below highlights selectors and required fields; the OpenAPI request schema contains the full field set.

Parameter	Location	Required?	Type	Description
firstName	body	Required	string	Contact first name; nonempty on contact creation. Case-sensitive field name.

Parameter	Location	Required?	Type	Description
lastName	body	Required	string	Contact last name; nonempty on contact creation. Case-sensitive field name.
smsok	body	Optional	integer	SMS permission: 0=no, 1=yes. Set explicitly; contact creation otherwise defaults to 1.
new_lead_notification	body	Optional	integer	Contact creation alert: 0 or omitted=none; 1=in-app; 2=in-app plus configured email/text alerts.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_client' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "firstName": "Jamie",
    "lastName": "Example",
    "phone": "2025550147",
    "email": "jamie@example.com",
    "smsok": 0,
    "new_lead_notification": 0
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "client_id": 100001,
  "external_id": "900001"
}
```

update_client_by_id

POST https://api.clienttether.com/v2/api/update_client_by_id

Update only the fields you intend to change. Use `client_id` from the selected account. `whiteboard` replaces content; `append_whiteboard` appends content. Assigning an Action Plan or changing the sales-cycle step can change scheduled automation. Read back the fields your integration depends on after the update.

Shared input: Contact fields in Chapter 4. The table below highlights selectors and required fields; the OpenAPI request schema contains the full field set.

Parameter	Location	Required?	Type	Description
client_id	body	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.

Request example:

```
curl 'https://api.clienttether.com/v2/api/update_client_by_id' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "client_id": 100001,
    "city": "Denver"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "client_id": 100001,
  "external_id": "900001"
}
```

update_client_by_external_id

POST https://api.clienttether.com/v2/api/update_client_by_external_id

Select the record with `external_id`, or with the `third_party_name`/`third_party_id` pair. Send one selector style and only the intended changes. Keep external mappings account-qualified and confirm the selected record before writing.

Shared input: Contact fields in Chapter 4. The table below highlights selectors and required fields; the OpenAPI request schema contains the full field set.

Parameter	Location	Required?	Type	Description
<code>external_id</code>	body	Optional	string	Your external-system record ID. Keep the mapping qualified by account.
<code>third_party_name</code>	body	Optional	string	Integration/system name used together with <code>third_party_id</code> .
<code>third_party_id</code>	body	Optional	string	External identifier under <code>third_party_name</code> . Supply the pair together.

Request example:

```
curl 'https://api.clienttether.com/v2/api/update_client_by_external_id' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "external_id": "900001",
    "city": "Denver"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "client_id": 100001,
  "external_id": "900001"
}
```

delete_client_by_id

DELETE https://api.clienttether.com/v2/api/delete_client_by_id/{id}

Delete the identified contact. Confirm the account and record mapping before executing. If a request times out, check whether the deletion occurred before retrying.

Parameter	Location	Required?	Type	Description
<code>id</code>	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --request DELETE 'https://api.clienttether.com/v2/api/delete_client_by_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Compatibility forms: delete_client_by_id. Use the selector map in Chapter 11 when calling a query-only form.

delete_client_by_external_id

DELETE https://api.clienttether.com/v2/api/delete_client_by_external_id/{id}

Delete a contact using its external ID. Keep the external identifier qualified by account; use the ClientTether-ID route when a mapping is ambiguous.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --request DELETE 'https://api.clienttether.com/v2/api/delete_client_by_external_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Compatibility forms: delete_client_by_external_id. Use the selector map in Chapter 11 when calling a query-only form.

Opportunities

create_opportunity

POST https://api.clienttether.com/v2/api/create_opportunity

Create an opportunity under an existing parent contact. The parent must belong to the selected account. The new opportunity copies parent contact details and receives its own contact-style client_id. Store the parent and opportunity IDs separately.

Shared input: Contact fields in Chapter 4. The table below highlights selectors and required fields; the OpenAPI request schema contains the full field set.

Parameter	Location	Required?	Type	Description
parent_id	body	Required	integer	Existing parent contact ID for a new opportunity. Parent must belong to the target account.
opportunity_name	body	Required	string	Name of the new opportunity.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_opportunity' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "parent_id": 100001,
    "opportunity_name": "Exterior repaint"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "client_id": 100001,
  "external_id": "900001"
}
```

read_opportunity_list

GET https://api.clienttether.com/v2/api/read_opportunity_list

List opportunities using account, parent-contact, milestone, assigned-user and date filters. The usual page size is 100. mega_payload=1 requests the larger export mode, with a ceiling of 3000 in this API edition. state means a sales milestone, not a postal state; contact_id filters by the parent contact. Prefer ordinary pages for ongoing synchronization.

Parameter	Location	Required?	Type	Description
mega_payload	query	Optional	integer	1 raises the opportunity controller page ceiling from 100 to 3000; use ordinary pages for routine syncs.
limit	query	Optional	integer	Requested maximum page size. Default/cap depends on the endpoint; see its behavior notes.
offset	query	Optional	integer	Zero-based number of records to skip; use a nonnegative integer.
contact_type	query	Optional	integer	Contact type: 1 Client, 2 Employee, 3 Partner, 4 Vendor, 5 Other. List endpoints accept 0 for all five.
contact_id	query	Optional	integer	Parent contact filter in read_opportunity_list; this is not the opportunity record ID.
state	query	Optional	string	Milestone: appt, quoted, closed or completed. This is not a geographic state.
stateFromDate	query	Optional	string	Lower milestone-date bound for the selected opportunity state.

Parameter	Location	Required?	Type	Description
stateToDate	query	Optional	string	Upper milestone-date bound for the selected opportunity state.
cycle_id	query	Optional	integer	Sales-cycle step filter; for delete_sales_cycle_by_id this is the query identifier.
user_assigned_id	query	Optional	integer	Assigned-user filter in read_opportunity_list. Note spelling differs from contact write assigned_user_id.
proposalModifiedFromDate	query	Optional	string	Lower proposal-modification timestamp bound (UTC).
proposalModifiedToDate	query	Optional	string	Upper proposal-modification timestamp bound (UTC).
modified_date_from	query	Optional	string	Lower contact-modification timestamp bound (UTC).
modified_date_to	query	Optional	string	Upper contact-modification timestamp bound (UTC).
start_date	query	Optional	string	Start date/time filter; on list APIs this usually filters record creation in UTC. On quote overview it is the job start date.
end_date	query	Optional	string	End date/time filter; on list APIs this usually filters record creation in UTC. On quote overview it is the job end date.
from_date	query	Optional	string	Lower-bound date/time. Event endpoints interpret this in the account timezone. Contact-list alias for start_date.
to_date	query	Optional	string	Contact/opportunity-list alias for end_date. Avoid sending both aliases and canonical names.
active_opportunities_only	query	Optional	integer	1 requests active opportunities according to service filtering; omit for the endpoint default.
most_recent_at_top	query	Optional	integer	1 requests descending created-date ordering.
archived	query	Optional	integer	0 (default) excludes archived records; verify the effect of other values in your account.
lead_source_id	query	Optional	integer	Lead source ID within the target account. May select mapped connection automation.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_opportunity_list' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --data-urlencode 'limit=100' \
  --data-urlencode 'offset=0' \
  --data-urlencode 'state=quoted'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
```

```
"TotalRecord": 1,
"dataRecord": 1,
"data": [
  {
    "client_id": 100001,
    "firstName": "Jamie",
    "lastName": "Example"
  }
]
```

read_pipeline_list

POST https://api.clienttether.com/v2/api/read_pipeline_list

Read pipeline records using a POST body. Results include contact, account and sales-cycle context, ordered by cycle position and time in the step. The request supports limit but does not define an offset-based pagination contract.

Parameter	Location	Required?	Type	Description
contact_active	body	Optional	integer	Pipeline listing filter applied to the contact ready flag. Confirm configured state values before using beyond 1.
sales_cycle_active	body	Optional	integer	Pipeline filter for active/inactive cycle steps: 1 or 0.
contact_type	body	Optional	integer	Contact type: 1 Client, 2 Employee, 3 Partner, 4 Vendor, 5 Other. List endpoints accept 0 for all five.
limit	body	Optional	integer	Requested maximum page size. Default/cap depends on the endpoint; see its behavior notes.

Request example:

```
curl 'https://api.clienttether.com/v2/api/read_pipeline_list' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY" \
--header 'Content-Type: application/json' \
--data '{
  "contact_type": 1,
  "contact_active": 1,
  "limit": 100
}'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```

History

read_client_history_notes

GET https://api.clienttether.com/v2/api/read_client_history_notes/{id}

Retrieve history for the contact. Read-side type labels are not necessarily the same as bulk-write type values. media may be a JSON-encoded string. This endpoint does not provide a documented page cursor; plan for the complete returned history.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_client_history_notes/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```

Compatibility forms: read_client_history_notes. Use the selector map in Chapter 11 when calling a query-only form.

create_client_history_note

POST https://api.clienttether.com/v2/api/create_client_history_note

Write a plain-text history note on the contact. This records an outcome or context; it does not send an email or SMS.

Parameter	Location	Required?	Type	Description
client_id	body	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.
note_body	body	Required	string	Nonempty history note text. This records history; it does not send a message.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_client_history_note' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "client_id": 100001,
    "note_body": "Integration completed the requested follow-up. Reference EXT-1001."
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

create_history_notes

POST https://api.clienttether.com/v2/api/create_history_notes

Insert one or more history entries. Use note, sms, sms_in, email or email_in as the write type. sms and email represent outbound history; sms_out and email_out are not equivalent and become ordinary notes. Use users_name

for attribution. Inspect each item in Response, because a top-level CT_200 can contain failed records. This endpoint records communication; it does not send it.

Parameter	Location	Required?	Type	Description
notes	body	Required	array	Array of history entries; see the nested field table below.

notes[] property	Requirement	Description
client_id	Required	Target contact ID.
type	Required	note, sms, sms_in, email or email_in.
utc_datetime	Optional	UTC ISO timestamp; use an explicit valid value for imported history.
subject	Optional	History subject.
body_content	Optional	History content.
users_name	Optional	Attribution name; defaults to the account owner.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_history_notes' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "notes": [
      {
        "client_id": 100001,
        "type": "note",
        "utc_datetime": "2026-10-22T16:00:00Z",
        "subject": "Integration outcome",
        "body_content": "Completed requested follow-up. Reference EXT-1001.",
        "users_name": "Integration User"
      }
    ]
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "Response": [
    {
      "record": 1,
      "client_id": 100001,
      "result": "success"
    }
  ]
}
```

Messages

create_manual_email

POST https://api.clienttether.com/v2/api/create_manual_email

Send or schedule an email to a contact. Omit `email_date` and `email_time` to send immediately, or provide both in account-local time to schedule. The contact needs a valid email address. API acceptance and final delivery are separate outcomes. This endpoint documents the fields listed below; CC, BCC and `exclude_contact` are not part of this request contract.

Parameter	Location	Required?	Type	Description
<code>client_id</code>	body	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.
<code>email_subject</code>	body	Required	string	Nonempty email subject.
<code>email_date</code>	body	Optional	string	Scheduled email date in account timezone, YYYY-MM-DD. Supply with <code>email_time</code> ; omit both for immediate sending.
<code>email_time</code>	body	Optional	string	Scheduled email time in account timezone, HH:MM:SS.
<code>email_body</code>	body	Required	string	Nonempty email body. Prefer plain text for portability; do not assume signature/HTML handling across implementations.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_manual_email' \  
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \  
  --header "X-Web-Key: $CT_WEB_KEY" \  
  --header 'Content-Type: application/json' \  
  --data '{  
    "client_id": 100001,  
    "email_subject": "Appointment follow-up",  
    "email_body": "Thank you for speaking with our team."  
  }'
```

Response example (selected fields):

```
{  
  "ResultCode": "CT_200",  
  "Message": "Success"  
}
```

create_manual_text

POST https://api.clienttether.com/v2/api/create_manual_text

Send or schedule an SMS to an eligible contact. Omit both schedule fields for immediate sending. `text_date` requires `text_time`. Confirm the contact phone, SMS permission and the account texting setup. API acceptance does not prove final delivery.

Parameter	Location	Required?	Type	Description
<code>client_id</code>	body	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.

Parameter	Location	Required?	Type	Description
text_date	body	Optional	string	Scheduled SMS date in account timezone, YYYY-MM-DD; requires text_time. Omit both schedule fields to send immediately.
text_time	body	Optional	string	Scheduled SMS time, HH:MM:SS, in account timezone.
text_body	body	Required	string	Nonempty SMS body. The contact needs an eligible mobile number and SMS permission.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_manual_text' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "client_id": 100001,
    "text_body": "Thank you for contacting our team."
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Events

read_all_client_events

GET https://api.clienttether.com/v2/api/read_all_client_events

Read appointment, call and contact-reminder events for a contact. Supply `client_id` or `external_id`. Event dates and times are returned in the account timezone. This is an event list, not a general queue of every pending Action Plan action.

Parameter	Location	Required?	Type	Description
client_id	query	Optional	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.
external_id	query	Optional	string	Your external-system record ID. Keep the mapping qualified by account.
from_date	query	Optional	string	Lower-bound date/time. Event endpoints interpret this in the account timezone. Contact-list alias for <code>start_date</code> .

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_all_client_events' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --data-urlencode 'client_id=100001'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": [
    {
      "event_id": 100004,
      "client_id": 100001,
      "event_type": "appt",
      "event_date": "2027-01-12",
      "event_time": "10:00:00",
      "event_end_date": "2027-01-12",
      "event_end_time": "11:00:00",
      "user_id": 100008
    }
  ]
}
```

read_all_events

GET https://api.clienttether.com/v2/api/read_all_events

Read account events from `from_date`, with a zero-based offset and pages of 100 records. `from_date` uses account-local time and defaults to today. Pagination fields are `Total_Count` and `Offset`.

Parameter	Location	Required?	Type	Description
<code>from_date</code>	query	Optional	string	Lower-bound date/time. Event endpoints interpret this in the account timezone. Contact-list alias for <code>start_date</code> .
<code>offset</code>	query	Optional	integer	Zero-based number of records to skip; use a nonnegative integer.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_all_events' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --data-urlencode 'from_date=2026-10-01 00:00:00' \
  --data-urlencode 'offset=0'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": [
    {
      "event_id": 100004,
      "client_id": 100001,
      "event_type": "appt",
      "event_date": "2027-01-12",
      "event_time": "10:00:00",
      "event_end_date": "2027-01-12",
      "event_end_time": "11:00:00",
      "user_id": 100008
    }
  ]
}
```

create_client_event

POST https://api.clienttether.com/v2/api/create_client_event

Create an Appointment, Call or Contact Reminder. Supply a start date; calls and appointments also require a start time. For appointments, send both end fields and ensure the end follows the start. Times are account-local and past starts are rejected. A missing or invalid user_id can assign the account owner. Use hosted-page booking when availability, notice or calendar rules must apply.

Parameter	Location	Required?	Type	Description
client_id	body	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.
event_type	body	Required	string	Call, Appointment/appt, or Contact Reminder/cappt (case-insensitive).
event_date	body	Required	string	Event start date, YYYY-MM-DD, in the target account timezone.
event_time	body	Optional	string	Event start time, HH:MM:SS, in the target account timezone. Required for Call/Appointment when providing a date.
event_end_date	body	Optional	string	Appointment end date, YYYY-MM-DD. Supply with event_end_time.
event_end_time	body	Optional	string	Appointment end time, HH:MM:SS. End must be after start.
event_note	body	Optional	string	Event notes text.
user_id	body	Optional	integer	User/calendar ID. Invalid or omitted user may fall back to the account owner on event creation.
event_manual_action_plan_id	body	Optional	string	Manual-schedule Event Plan ID. Other plan types may be skipped with a Warning in a successful response.
template_id	body	Optional	string	Appointment invitation template ID; affects calendar/invitation content.
email_list	body	Optional	string	Comma-separated invitee email addresses. Validate notifications in the test account.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_client_event' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "client_id": 100001,
    "event_type": "Appointment",
    "event_date": "2027-01-12",
    "event_time": "10:00:00",
    "event_end_date": "2027-01-12",
    "event_end_time": "11:00:00",
    "event_note": "Integration-created appointment"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "Event_id": 100004
}
```

```
}

```

update_client_event

POST https://api.clienttether.com/v2/api/update_client_event

Change an existing event using event_id. Send only the fields to change. Date/time changes must remain valid. Changing a manual Event Plan can remove its associated scheduled actions. Inspect any Warning and read the resulting event back.

Parameter	Location	Required?	Type	Description
client_id	body	Optional	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.
event_id	body	Required	integer	ClientTether event ID returned as Event_id on creation/update or event_id in event lists.
event_type	body	Optional	string	Call, Appointment/appt, or Contact Reminder/cappt (case-insensitive).
event_date	body	Optional	string	Event start date, YYYY-MM-DD, in the target account timezone.
event_time	body	Optional	string	Event start time, HH:MM:SS, in the target account timezone. Required for Call/Appointment when providing a date.
event_end_date	body	Optional	string	Appointment end date, YYYY-MM-DD. Supply with event_end_time.
event_end_time	body	Optional	string	Appointment end time, HH:MM:SS. End must be after start.
event_note	body	Optional	string	Event notes text.
user_id	body	Optional	integer	User/calendar ID. Invalid or omitted user may fall back to the account owner on event creation.
event_manual_action_plan_id	body	Optional	string	Manual-schedule Event Plan ID. Other plan types may be skipped with a Warning in a successful response.
email_list	body	Optional	string	Comma-separated invitee email addresses. Validate notifications in the test account.
template_id	body	Optional	string	Appointment invitation template ID; affects calendar/invitation content.

Request example:

```
curl 'https://api.clienttether.com/v2/api/update_client_event' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "event_id": 100001,
    "event_note": "Updated arrival instructions."
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "Event_id": 100004
}
```

delete_client_event

DELETE https://api.clienttether.com/v2/api/delete_client_event/{id}

Delete the identified event. Use event_id in the query-only alternative. Review any linked manual Event Plan before canceling an event that participates in automation.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --request DELETE 'https://api.clienttether.com/v2/api/delete_client_event/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Compatibility forms: delete_client_event. Use the selector map in Chapter 11 when calling a query-only form.

Scheduling

read_hosted_pages

GET https://api.clienttether.com/v2/api/read_hosted_pages

List the scheduling pages available in the account. Success returns an array directly. Select a page using hosted_page_id and its meeting metadata. users can contain serialized configuration data; obtain user details separately when your integration needs display names.

No request parameters beyond the two access headers.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_hosted_pages' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
[
  {
    "hosted_page_id": 100005,
    "page_slug": "example-estimate",
    "meeting_name": "Estimate",
    "description": "Estimate meeting",
    "aptLength": 60,
    "wrapupTime": 15,
  }
]
```

```
"advNotice": 24,
"users": "[100008]",
"type": "contact"
}
]
```

read_available_timeslots

POST https://api.clienttether.com/v2/api/read_available_timeslots

Read available slots with a POST request. Supply `hosted_page_id` and either `client_id` or the complete prospect phone/address context. Existing contacts still need the required phone/address information. Set start and end explicitly; defaults are 1 and 7. These are day offsets, not a duration. Use the returned `time_zone` and `UTC` fields when presenting a time. Success returns a slot array directly.

Parameter	Location	Required?	Type	Description
client_id	body	Optional	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.
hosted_page_id	body	Required	integer	Hosted scheduling-page ID returned by <code>read_hosted_pages</code> . Availability field name differs from booking <code>hosted_id</code> .
start	body	Optional	integer	First day offset from today. Default 1. Set explicitly; 0 requests a window beginning today.
end	body	Optional	integer	Last day offset from today. Default 7. This is an end offset, not a number of days to add to start.
mobile	body	Optional	string	Prospective attendee mobile number for hosted scheduling.
streetAddress	body	Optional	string	Prospective attendee street address; hosted scheduling uses this spelling rather than <code>address</code> .
city	body	Optional	string	Contact city.
state	body	Optional	string	Contact state/province. Use a two-letter code for portability; in opportunity filters this name has a different meaning.
zip	body	Optional	string	Postal code as a string, retaining leading zeros.

Request example:

```
curl 'https://api.clienttether.com/v2/api/read_available_timeslots' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "hosted_page_id": 100001,
    "client_id": 100001,
    "start": 0,
    "end": 2
  }'
```

Response example (selected fields):

```
[
  {
    "block_id": "OPAQUE_TOKEN_FROM_SERVER",
    "title": "Block 1",
```

```
{
  "class": "block-available",
  "time_zone": "America/Denver",
  "start_date": "2026-10-23T10:00:00",
  "end_date": "2026-10-23T11:00:00",
  "start_date_utc": "2026-10-23T16:00:00",
  "end_date_utc": "2026-10-23T17:00:00"
}
```

hosted_page_schedule_meeting

POST https://api.clienttether.com/v2/api/hosted_page_schedule_meeting

Book a fresh availability block. Send `hosted_id`, the opaque `block_id` and either `client_id` or all required new-attendee details. Use `client_id` in the public request. Never construct a `block_id` or substitute a hand-built time. If the response indicates that the slot is no longer available, refresh availability and ask for a new selection.

Parameter	Location	Required?	Type	Description
<code>block_id</code>	body	Required	string	Opaque availability token returned by <code>read_available_timeslots</code> . Copy exactly; never decode or manufacture it.
<code>client_id</code>	body	Optional	integer	ClientTether contact/opportunity record ID. Obtain it from a <code>read</code> or <code>create</code> response in the target account.
<code>hosted_id</code>	body	Required	integer	Hosted scheduling-page ID for booking; use the same page selected for availability.
<code>meetingNotes</code>	body	Optional	string	Notes for the hosted-page appointment.
<code>emailAddress</code>	body	Optional	string	Prospective attendee email for hosted booking; differs from contact email.
<code>fullName</code>	body	Optional	string	Prospective attendee full name for hosted booking.
<code>mobile</code>	body	Optional	string	Prospective attendee mobile number for hosted scheduling.
<code>streetAddress</code>	body	Optional	string	Prospective attendee street address; hosted scheduling uses this spelling rather than <code>address</code> .
<code>city</code>	body	Optional	string	Contact city.
<code>state</code>	body	Optional	string	Contact state/province. Use a two-letter code for portability; in opportunity filters this name has a different meaning.
<code>zip</code>	body	Optional	string	Postal code as a string, retaining leading zeros.

Request example:

```
curl 'https://api.clienttether.com/v2/api/hosted_page_schedule_meeting' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "hosted_id": 100001,
    "block_id": "COPY_FRESH_BLOCK_ID_FROM_AVAILABILITY",
    "client_id": 100001,
    "meetingNotes": "Estimate appointment"
  }'
```

```
}'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Action Plans

read_action_plan_list

GET https://api.clienttether.com/v2/api/read_action_plan_list

Read the account Action Plan catalog. `action_plan_name` filters by name prefix. Plans may be grouped by type, and the available catalog can reflect enterprise sharing. Use returned IDs rather than copying IDs from another location.

Parameter	Location	Required?	Type	Description
<code>action_plan_name</code>	query	Optional	string	Optional Action Plan name prefix filter.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_action_plan_list' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --data-urlencode 'action_plan_name=Follow'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

read_action_plan_by_id

GET https://api.clienttether.com/v2/api/read_action_plan_by_id/{id}

Read one Action Plan definition. Reading a plan does not start it. To assign/start a plan for a contact, write `action_plan_id` on the contact.

Parameter	Location	Required?	Type	Description
<code>id</code>	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_action_plan_by_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": {}
}
```

```
}
```

Compatibility forms: `read_action_plan_by_id`. Use the selector map in Chapter 11 when calling a query-only form.

read_actions_by_plan_id

GET `https://api.clienttether.com/v2/api/read_actions_by_plan_id/{id}`

Read the ordered action definitions in an Action Plan. This describes the plan template, not a contact-specific execution queue.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_actions_by_plan_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```

Sales Cycles

read_sales_cycle_list

GET `https://api.clienttether.com/v2/api/read_sales_cycle_list`

List the sales-cycle steps available in the account, optionally filtered by name. Use returned IDs for contact movement and step configuration.

Parameter	Location	Required?	Type	Description
name	query	Optional	string	Name filter for sales-cycle listing; quote overview uses this as the proposal name.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_sales_cycle_list' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```

create_sales_cycle

POST https://api.clienttether.com/v2/api/create_sales_cycle

Create or find a named sales-cycle step. Set `sales_cycle_active_status` and `sales_cycle_order` explicitly; an omitted active status defaults to 0. This changes account configuration, not a contact position.

Parameter	Location	Required?	Type	Description
<code>sales_cycle_name</code>	body	Required	string	Sales-cycle step display name.
<code>sales_cycle_active_status</code>	body	Optional	integer	1 active, 0 inactive. Create service defaults to 0 when omitted; send explicitly.
<code>sales_cycle_order</code>	body	Optional	integer	Step ordering within the active/inactive path. Send explicitly when order matters.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_sales_cycle' \  
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \  
  --header "X-Web-Key: $CT_WEB_KEY" \  
  --header 'Content-Type: application/json' \  
  --data '{  
    "sales_cycle_name": "Qualified",  
    "sales_cycle_active_status": 1,  
    "sales_cycle_order": 2  
  }'
```

Response example (selected fields):

```
{  
  "ResultCode": "CT_200",  
  "Message": "Success",  
  "Sales_Cycle_id": 100006  
}
```

update_sales_cycle_by_id

POST https://api.clienttether.com/v2/api/update_sales_cycle_by_id

Update a sales-cycle step definition. To move a contact into a step, update the contact `sales_cycle_id` instead.

Parameter	Location	Required?	Type	Description
<code>sales_cycle_id</code>	body	Required	integer	Sales-cycle step ID for writing a contact; read responses may call this <code>cycle_steps_id</code> or <code>cycle_id</code> .
<code>sales_cycle_name</code>	body	Optional	string	Sales-cycle step display name.
<code>sales_cycle_active_status</code>	body	Optional	integer	1 active, 0 inactive. Create service defaults to 0 when omitted; send explicitly.
<code>sales_cycle_order</code>	body	Optional	integer	Step ordering within the active/inactive path. Send explicitly when order matters.

Request example:

```
curl 'https://api.clienttether.com/v2/api/update_sales_cycle_by_id' \  
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \  
  --data '{  
    "sales_cycle_id": 100006,  
    "sales_cycle_name": "Qualified",  
    "sales_cycle_active_status": 1,  
    "sales_cycle_order": 2  
  }'
```

```
--header "X-Web-Key: $CT_WEB_KEY" \
--header 'Content-Type: application/json' \
--data '{
  "sales_cycle_id": 100001,
  "sales_cycle_name": "Estimate scheduled"
}'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "Sales_Cycle_id": 100006
}
```

delete_sales_cycle_by_id

DELETE https://api.clienttether.com/v2/api/delete_sales_cycle_by_id/{id}

Delete a step that is no longer in use. Referenced steps can return CT_512. The query-only identifier is cycle_id; the update identifier sales_cycle_id is not interchangeable.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --request DELETE 'https://api.clienttether.com/v2/api/delete_sales_cycle_by_id/100001' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Compatibility forms: delete_sales_cycle_by_id. Use the selector map in Chapter 11 when calling a query-only form.

read_sales_cycle_any_activity

POST https://api.clienttether.com/v2/api/read_sales_cycle_any_activity

Read sales-cycle activity within the inclusive UTC date range from startDate at 00:00:00 through endDate at 23:59:59. Send the dates in a POST body.

Parameter	Location	Required?	Type	Description
startDate	body	Required	string	Activity report lower date bound, YYYY-MM-DD (UTC).
endDate	body	Required	string	Activity report upper date bound, YYYY-MM-DD (UTC).

Request example:

```
curl 'https://api.clienttether.com/v2/api/read_sales_cycle_any_activity' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
```

```
--header "X-Web-Key: $CT_WEB_KEY" \
--header 'Content-Type: application/json' \
--data '{
  "startDate": "2026-10-01",
  "endDate": "2026-10-22"
}'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```

Lead Sources

read_lead_source_list

GET https://api.clienttether.com/v2/api/read_lead_source_list

List account lead sources. Use returned IDs in contact writes and retain any automation mapping associated with the source.

No request parameters beyond the two access headers.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_lead_source_list' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```

create_lead_source

POST https://api.clienttether.com/v2/api/create_lead_source

Create or find a named lead source. Returns `Lead_Sources_id`. A new source can also receive its own source web key; keep account credentials separate from identifiers used in lead-source configuration.

Parameter	Location	Required?	Type	Description
lead_source_name	body	Required	string	Nonempty lead source display name.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_lead_source' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY" \
--header 'Content-Type: application/json' \
--data '{
  "lead_source_name": "Website integration"
}'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "Lead_Sources_id": 100006
}
```

update_lead_source_by_id

POST https://api.clienttether.com/v2/api/update_lead_source_by_id

Update an existing lead-source name using lead_source_id.

Parameter	Location	Required?	Type	Description
lead_source_id	body	Required	integer	Lead source ID within the target account. May select mapped connection automation.
lead_source_name	body	Optional	string	Nonempty lead source display name.

Request example:

```
curl 'https://api.clienttether.com/v2/api/update_lead_source_by_id' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "lead_source_id": 100001,
    "lead_source_name": "Website quote form"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "Lead_Sources_id": 100006
}
```

delete_lead_source_by_id

DELETE https://api.clienttether.com/v2/api/delete_lead_source_by_id/{id}

Delete a lead source that is no longer in use. Referenced sources can return CT_512. Use source_id for the query-only delete form.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --request DELETE 'https://api.clienttether.com/v2/api/delete_lead_source_by_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

```
}
```

Compatibility forms: `delete_lead_source_by_id`. Use the selector map in Chapter 11 when calling a query-only form.

Tags

read_tags_list

GET `https://api.clienttether.com/v2/api/read_tags_list`

List tags available to the account, including configured shared tags.

No request parameters beyond the two access headers.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_tags_list' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```

create_tag

POST `https://api.clienttether.com/v2/api/create_tag`

Create or find a tag definition. This does not assign the tag to a contact. Use `create_contact_tag_assignment` for the assignment.

Parameter	Location	Required?	Type	Description
<code>tag_value</code>	body	Required	string	Tag name/value, not tag ID. Assignment endpoints expect an existing available tag; <code>create_tag</code> can create one.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_tag' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "tag_value": "Integration imported"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "Tag_id": 100006
}
```

create_contact_tag_assignment

POST https://api.clienttether.com/v2/api/create_contact_tag_assignment

Assign an existing available tag to the contact. If CT_530 is returned, check `read_tags_list` and create the tag if appropriate. Use POST for this operation.

Parameter	Location	Required?	Type	Description
tag_value	body	Required	string	Tag name/value, not tag ID. Assignment endpoints expect an existing available tag; create_tag can create one.
client_id	body	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_contact_tag_assignment' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "client_id": 100001,
    "tag_value": "Integration imported"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

read_contact_tags

GET https://api.clienttether.com/v2/api/read_contact_tags/{client_id}

Read tag assignments for a contact. Preserve the returned response shape; tag responses can differ from the common data envelope.

Parameter	Location	Required?	Type	Description
client_id	path	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_contact_tags/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

delete_contact_tag_assignment

DELETE https://api.clienttether.com/v2/api/delete_contact_tag_assignment/{client_id}/{tag_value}

Remove a tag assignment without deleting the tag definition. URL-encode the tag_value path component.

Parameter	Location	Required?	Type	Description
client_id	path	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.
tag_value	path	Required	string	Tag name/value, not tag ID. Assignment endpoints expect an existing available tag; create_tag can create one.

Request example:

```
curl --request DELETE 'https://api.clienttether.com/v2/api/delete_contact_tag_assignment/100001/Example%20tag' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Custom Fields

read_custom_field_by_contact

GET https://api.clienttether.com/v2/api/read_custom_field_by_contact/{clientId}

Read configured custom-field labels and values for a contact. Response keys include Client_Id and capitalized Data. Write a custom field using its exact label as a top-level contact input key.

Parameter	Location	Required?	Type	Description
clientId	path	Required	integer	Identifier returned by the corresponding account operation.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_custom_field_by_contact/100001' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "Client_Id": 100001,
  "Data": []
}
```

Users

read_user_list

GET https://api.clienttether.com/v2/api/read_user_list

List users for the selected account. Use the returned IDs for contact assignments and event calendar selection.

No request parameters beyond the two access headers.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_user_list' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```

read_user_by_id

GET https://api.clienttether.com/v2/api/read_user_by_id/{id}

Read one user using the path identifier.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_user_by_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": {}
}
```

create_user

POST https://api.clienttether.com/v2/api/create_user

Create a user login and profile. user_type=admin creates an administrator; select the role deliberately. Usernames are lowercased and checked for availability; email availability is also checked. Supply all listed required fields and set the initial password securely at runtime.

Parameter	Location	Required?	Type	Description
user_type	body	Required	string	User role; admin creates an administrator, other values follow the normal-user path. Choose deliberately.
user_first_name	body	Required	string	User first name.
user_last_name	body	Required	string	User last name.
user_phone	body	Required	string	User phone string.
user_email	body	Required	string	User email; creation checks uniqueness.
user_username	body	Required	string	Login username; creation lowercases it and checks availability. Not an ordinary updateable field in the documented update service.
user_password	body	Required	string	Initial user password. Supply securely at runtime; never embed it in distributed examples.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_user' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "user_type": "user",
    "user_first_name": "Integration",
    "user_last_name": "Example",
    "user_phone": "2025550188",
    "user_email": "integration@example.com",
    "user_username": "integration.example",
    "user_password": "SET_SECURE_PASSWORD_AT_RUNTIME"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "User_id": 100006
}
```

update_user_by_id

POST https://api.clienttether.com/v2/api/update_user_by_id

Update the user profile using nonempty name, phone or email values. Username, password and role changes are not part of this profile-update contract.

Parameter	Location	Required?	Type	Description
user_id	body	Required	integer	User/calendar ID. Invalid or omitted user may fall back to the account owner on event creation.
user_first_name	body	Optional	string	User first name.
user_last_name	body	Optional	string	User last name.

Parameter	Location	Required?	Type	Description
user_phone	body	Optional	string	User phone string.
user_email	body	Optional	string	User email; creation checks uniqueness.

Request example:

```
curl 'https://api.clienttether.com/v2/api/update_user_by_id' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "user_id": 100001,
    "user_phone": "2025550188"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "User_id": 100006
}
```

delete_user_by_id

DELETE https://api.clienttether.com/v2/api/delete_user_by_id/{id}

Delete a user belonging to the account. Check access and assignments before deletion.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --request DELETE 'https://api.clienttether.com/v2/api/delete_user_by_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Compatibility forms: delete_user_by_id. Use the selector map in Chapter 11 when calling a query-only form.

Accounts

read_account_list

GET https://api.clienttether.com/v2/api/read_account_list

Read permitted child accounts using enterprise-level API access. This is not a guarantee of a recursive network export. Returned account_web_key values are credentials and must remain server-side.

No request parameters beyond the two access headers.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_account_list' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```

update_account_by_id

POST https://api.clienttether.com/v2/api/update_account_by_id

Update the permitted child account profile and listed preferences. Login username/password/type changes are not part of this operation. Coordinate any ClientTether-managed phone-number change with Support.

Parameter	Location	Required?	Type	Description
account_id	body	Required	integer	Account ID in the permitted hierarchy. Account-list access is enterprise-level .
account_first_name	body	Optional	string	Account login owner first name.
account_last_name	body	Optional	string	Account login owner last name.
account_owner_name	body	Optional	string	Account owner display name.
account_company_name	body	Optional	string	Account company display name.
account_addr1	body	Optional	string	Account address line 1.
account_addr2	body	Optional	string	Account address line 2.
account_addr3	body	Optional	string	Account address line 3.
account_office_phone	body	Optional	string	Account office phone.
account_phone	body	Optional	string	Account login owner phone.
account_email	body	Optional	string	Account login owner email.
account_ctphone	body	Optional	string	ClientTether-managed account number; requires coordinated telephony configuration, not just a record update.
account_return_phone	body	Optional	string	Account return-call phone setting.
account_return_text	body	Optional	string	Account return-text phone setting.

Request example:

```
curl 'https://api.clienttether.com/v2/api/update_account_by_id' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "account_id": 100001,
    "account_company_name": "Example Location"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "Account_id": 100006
}
```

delete_account_by_id

DELETE https://api.clienttether.com/v2/api/delete_account_by_id/{id}

Delete a permitted child account. This is an administrative, destructive operation; confirm the account identity and intended outcome first.

Parameter	Location	Required?	Type	Description
id	path	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.

Request example:

```
curl --request DELETE 'https://api.clienttether.com/v2/api/delete_account_by_id/100001' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Compatibility forms: delete_account_by_id. Use the selector map in Chapter 11 when calling a query-only form.

Proposals

read_proposals

GET https://api.clienttether.com/v2/api/read_proposals

Read proposal output for a contact. This endpoint returns closed or completed proposals; it is not an all-draft-proposals export. A contact can therefore exist while proposals is empty. Output uses company, customer and proposals rather than the common data property.

Parameter	Location	Required?	Type	Description
client_id	query	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_proposals' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --data-urlencode 'client_id=100001'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "company": {
    "account_id": 100009
  },
  "customer": {
    "client_id": 100001
  },
  "proposals": []
}
```

Payments

read_proposal_payments

GET https://api.clienttether.com/v2/api/read_proposal_payments

Read proposal payment entries for a contact. Both `client_id` and `created_date` are required. `created_date` is a creation-time lower bound; `pay_date` optionally filters by payment day.

Parameter	Location	Required?	Type	Description
client_id	query	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.
created_date	query	Required	string	Payment creation lower bound. Use a valid UTC date/time; <code>read_all_payments</code> requires a nonempty value.
pay_date	query	Optional	string	Optional payment-day filter, YYYY-MM-DD.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/read_proposal_payments' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --data-urlencode 'client_id=100001' \
  --data-urlencode 'created_date=2026-10-01 00:00:00'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": []
}
```



```
--header "X-Web-Key: $CT_WEB_KEY" \
--header 'Content-Type: application/json' \
--data '{
  "proposal_id": 100001,
  "payday": "2026-10-22",
  "amount": 250.0,
  "type": "Other",
  "status": "Deposit",
  "number": "EXT-PAY-1001"
}'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success",
  "data": {
    "id": 100011,
    "proposal_id": 100010,
    "amount": 250,
    "status": "Deposit",
    "number": "EXT-PAY-1001"
  }
}
```

Calling

create_immediate_call

POST https://api.clienttether.com/v2/api/create_immediate_call

Initiate an immediate agent/contact call using an existing agent and contact. Phone values must match their configured records. `client_full_name` is required but does not create a missing contact. The request has three fields; it does not accept a say instruction.

Parameter	Location	Required?	Type	Description
agent_phone	body	Required	string	Existing agent phone in the expected stored format for the newer callback implementation.
client_full_name	body	Required	string	Required by the newer callback controller; it does not create the client record.
client_phone	body	Required	string	Existing contact mobile phone in the expected stored format for the newer callback implementation.

Request example:

```
curl 'https://api.clienttether.com/v2/api/create_immediate_call' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY" \
--header 'Content-Type: application/json' \
--data '{
  "agent_phone": "2025550188",
  "client_full_name": "Jamie Example",
  "client_phone": "2025550147"
}'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
```

```
{
  "Message": "Success",
  "data": "Immediate call initiated successfully."
}
```

Advanced Proposals

fetch_bid_types

GET https://api.clienttether.com/v2/api/fetch_bid_types

Read configured bid types for the proposal engine enabled on the account.

No request parameters beyond the two access headers.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/fetch_bid_types' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

fetch_quote_settings

GET https://api.clienttether.com/v2/api/fetch_quote_settings

Read the configured proposal-engine settings. Response details depend on the account proposal configuration.

No request parameters beyond the two access headers.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/fetch_quote_settings' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

quote_data_by_id

GET https://api.clienttether.com/v2/api/quote_data_by_id

Read the configured quote. data contains quote and client objects. The quote can include worksheet, changeOrder, punchList, payments and cogs. Preserve serialized documents in the shape returned by the account.

Parameter	Location	Required?	Type	Description
quote_id	query	Required	integer	Proposal/quote ID for the configured proposal engine.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/quote_data_by_id' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY" \
--data-urlencode 'quote_id=100001'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

quote_workorder_data_by_id

GET https://api.clienttether.com/v2/api/quote_workorder_data_by_id

Read work-order data for the configured quote. Nested content follows the account proposal engine.

Parameter	Location	Required?	Type	Description
quote_id	query	Required	integer	Proposal/quote ID for the configured proposal engine.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/quote_workorder_data_by_id' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY" \
--data-urlencode 'quote_id=100001'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

create_quote

GET https://api.clienttether.com/v2/api/create_quote/{client_id}

Create a quote for an existing contact. This operation uses GET despite creating data. Do not prefetch, crawl or automatically retry the URL. Use the proposal engine configured on the account.

Parameter	Location	Required?	Type	Description
client_id	path	Required	integer	ClientTether contact/opportunity record ID. Obtain it from a read or create response in the target account.

Request example:

```
curl --get 'https://api.clienttether.com/v2/api/create_quote/100001' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY"
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

quote/save_overview

POST https://api.clienttether.com/v2/api/quote/save_overview

Update proposal overview fields and associated contact deal values. Include the current quote_type when you intend to preserve it; omitting it can clear the type and change the opportunity name. Obtain IDs and existing values from the quote first.

Parameter	Location	Required?	Type	Description
id	body	Required	integer	Record identifier required by this endpoint. Use the ID returned by its corresponding read/create operation.
name	body	Optional	string	Name filter for sales-cycle listing; quote overview uses this as the proposal name.
estimator_id	body	Optional	integer	Estimator user ID.
prod_mgr_id	body	Optional	integer	Production manager user ID.
notes	body	Optional	string	Proposal overview notes text.
quote_type	body	Optional	string	Configured quote type. Overview writes can rename the opportunity.
notes_two	body	Optional	string	Secondary quote notes.
start_date	body	Optional	string	Start date/time filter; on list APIs this usually filters record creation in UTC. On quote overview it is the job start date.
end_date	body	Optional	string	End date/time filter; on list APIs this usually filters record creation in UTC. On quote overview it is the job end date.
total_deal	body	Optional	number	Quote total; writing it also updates contact deal values.
total_deal_withouttax	body	Optional	number	Quote total excluding tax; applied according to proposal tax settings.
tax_district_id	body	Optional	integer	Configured tax district ID.

Request example:

```
curl 'https://api.clienttether.com/v2/api/quote/save_overview' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "id": 100001
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

quote/save_worksheet

POST https://api.clienttether.com/v2/api/quote/save_worksheet

Save the worksheet document and totals. Use the serialized worksheet format returned for the same account proposal engine. Preserve its structure and exact worksheetTotalWithOutTax field spelling.

Parameter	Location	Required?	Type	Description
quote_id	body	Required	integer	Proposal/quote ID for the configured proposal engine.
worksheet	body	Required	string	Serialized worksheet document from the same account proposal engine; preserve the configured document structure.
worksheetTotal	body	Required	number	Worksheet total including applicable tax.
worksheetTotalWithOutTax	body	Required	number	Worksheet total excluding tax; preserve this exact capitalization.

Request example:

```
curl 'https://api.clienttether.com/v2/api/quote/save_worksheet' \
--header "X-Access-Token: $CT_ACCESS_TOKEN" \
--header "X-Web-Key: $CT_WEB_KEY" \
--header 'Content-Type: application/json' \
--data '{
  "quote_id": 100001,
  "worksheet": "REPLACE_WITH_ACCOUNT_VALIDATED_VALUE",
  "worksheetTotal": 100.0,
  "worksheetTotalWithOutTax": 100.0
}'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Document values: replace any REPLACE_WITH_ACCOUNT_VALIDATED_VALUE example string with the serialized value for the account proposal engine. Read the existing quote first; do not send the marker string.

quote/save_customer_approval

POST https://api.clienttether.com/v2/api/quote/save_customer_approval

Record approval on an existing quote worksheet. A supplied customerSignature can generate signed-proposal notifications and history. Submit only approval/signature data authorized by the customer workflow.

Parameter	Location	Required?	Type	Description
quote_id	body	Required	integer	Proposal/quote ID for the configured proposal engine.
customerApproval	body	Required	string	Nonempty approval data expected by the configured quote worksheet.
customerSignature	body	Optional	string	Signature representation expected by the configured proposal engine; only submit a real authorized signature.

Request example:

```
curl 'https://api.clienttether.com/v2/api/quote/save_customer_approval' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "quote_id": 100001,
    "customerApproval": "REPLACE_WITH_ACCOUNT_VALIDATED_VALUE"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Document values: replace any REPLACE_WITH_ACCOUNT_VALIDATED_VALUE example string with the serialized value for the account proposal engine. Read the existing quote first; do not send the marker string.

quote/save_workorder

POST https://api.clienttether.com/v2/api/quote/save_workorder

Create a work order from a configured worksheet and bid-type selection. type=0 creates it without emailing; type=1 emails the selected subcontractor. This creates a work order rather than updating one. Use the account proposal document format.

Parameter	Location	Required?	Type	Description
quote_id	body	Required	integer	Proposal/quote ID for the configured proposal engine.
worksheet	body	Required	string	Serialized worksheet document from the same account proposal engine; preserve the configured document structure.
bidTypes	body	Required	string	Serialized bid-type selections in the proposal engine format.
subcontractor	body	Required	integer	Configured subcontractor ID.
type	body	Required	integer	0 creates without emailing; 1 emails the selected subcontractor.

Request example:

```
curl 'https://api.clienttether.com/v2/api/quote/save_workorder' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "quote_id": 100001,
    "worksheet": "REPLACE_WITH_ACCOUNT_VALIDATED_VALUE",
    "bidTypes": "REPLACE_WITH_ACCOUNT_VALIDATED_VALUE",
    "subcontractor": 1,
    "type": 0
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
```

```
"Message": "Success"
}
```

Document values: replace any REPLACE_WITH_ACCOUNT_VALIDATED_VALUE example string with the serialized value for the account proposal engine. Read the existing quote first; do not send the marker string.

quote/save_changeorder

POST https://api.clienttether.com/v2/api/quote/save_changeorder

Save the configured change-order document, total and optional authorized customer signature.

Parameter	Location	Required?	Type	Description
quote_id	body	Required	integer	Proposal/quote ID for the configured proposal engine.
changeOrder	body	Required	string	Serialized change-order document in the target account proposal format.
changeOrderTotal	body	Required	number	Change-order total.
customerSignature	body	Optional	string	Signature representation expected by the configured proposal engine; only submit a real authorized signature.

Request example:

```
curl 'https://api.clienttether.com/v2/api/quote/save_changeorder' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "quote_id": 100001,
    "changeOrder": "REPLACE_WITH_ACCOUNT_VALIDATED_VALUE",
    "changeOrderTotal": 100.0
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Document values: replace any REPLACE_WITH_ACCOUNT_VALIDATED_VALUE example string with the serialized value for the account proposal engine. Read the existing quote first; do not send the marker string.

quote/save_puchlist

POST https://api.clienttether.com/v2/api/quote/save_puchlist

Save the configured punch-list document and optional authorized signature. The URL is spelled save_puchlist; the body property is punchList. Preserve both spellings exactly.

Parameter	Location	Required?	Type	Description
quote_id	body	Required	integer	Proposal/quote ID for the configured proposal engine.

Parameter	Location	Required?	Type	Description
punchList	body	Required	string	Serialized punch-list document in the target account proposal format.
customerSignature	body	Optional	string	Signature representation expected by the configured proposal engine; only submit a real authorized signature.

Request example:

```
curl 'https://api.clienttether.com/v2/api/quote/save_puchlist' \
  --header "X-Access-Token: $CT_ACCESS_TOKEN" \
  --header "X-Web-Key: $CT_WEB_KEY" \
  --header 'Content-Type: application/json' \
  --data '{
    "quote_id": 100001,
    "punchList": "REPLACE_WITH_ACCOUNT_VALIDATED_VALUE"
  }'
```

Response example (selected fields):

```
{
  "ResultCode": "CT_200",
  "Message": "Success"
}
```

Document values: replace any REPLACE_WITH_ACCOUNT_VALIDATED_VALUE example string with the serialized value for the account proposal engine. Read the existing quote first; do not send the marker string.

9. Webhooks and workflow automation

Configure the outgoing request

Configure outgoing webhooks through the account webhook/Action Plan settings. Select the receiver URL, HTTP method, header values, payload style, conditions and contact-payload option. Use HTTPS for your receiver and a separate configured authentication secret. The receiver must accept the configured encoding: JSON or form data. A configured GET places fields in its query string.

Define an explicit payload

Choose the fields your receiver needs. Include the ClientTether account and contact identifiers using the tokens available in your account. A webhook body is configured data, not automatically the same schema as a public API contact response. Contact payload fields can use names such as phoneCell and smsOK; nonempty-field selection can omit zero/false-like values. Custom payload keys can override automatic keys.

This example is a receiver contract you can configure, not a universal ClientTether event envelope:

```
{
  "schemaVersion": "1.0",
  "eventType": "followup_requested",
  "accountId": "YOUR_ACCOUNT_ID",
  "clientId": "YOUR_CONTACT_ID",
  "externalOperationId": "YOUR_OPERATION_REFERENCE"
}
```

Receive, acknowledge and reconcile

Authenticate the request, validate its account/contact context, store the event and acknowledge promptly. Process long-running business work asynchronously. Maintain a receiver-side duplicate ledger and an operator reconciliation path. Do not assume exactly-once delivery, ordering, automatic retry scheduling or a standard HMAC signature unless your configured integration contract defines them.

Use Webhook Monitor

Inspect actual outgoing URL, method, payload, headers, response and timestamp in Webhook Monitor. Compare the receiver logs with the monitor entry when diagnosing a delivery issue. Protect any authentication headers captured in logs. An HTTP acknowledgment means the receiver accepted the request; it does not prove that the downstream business workflow completed.

Prevent duplicate outreach

Decide whether ClientTether automation or your integration owns each outgoing message. For example, if a lead-source connection plan sends the initial SMS, do not also call create_manual_text for the same welcome message. Record external conversation outcomes in history and apply explicit cycle/plan transitions when the business rule requires them.

10. Troubleshooting

Symptom	What to check
HTTP 200 but an operation failed	ResultCode, Message, warnings and per-record bulk results.
Authentication rejected	Current API token plus the intended account web key.
Contact appears in the wrong location	Tenant/location-to-web-key mapping and the account-qualified record ID.
Lookup returns no match	Phone normalization, exact external mapping and unnecessary additional filters.
Repeated creates generate duplicates	Account duplicate-check setting, concurrent writers and your durable ID mapping.
Native UTM values remain empty	Use the supported attribution workflow; generic contact writes do not provide native UTM updates.
Bulk history becomes a plain note	Use documented write types; outbound history uses sms/email, and attribution uses users_name.
History exists but no message was sent	Use a manual send operation to send; history writes only record.
No new-lead alert	Explicit new_lead_notification choice and account channel preferences.
Availability begins on the wrong day	Explicit start/end offsets, UTC/local day boundary and hosted-page notice/business hours.
A selected slot is unavailable	Refresh the slot list and choose a new block_id.
An event is hours off	Account-local input time versus UTC history timestamps; use the correct IANA zone.
Proposal results omit drafts	read_proposals returns closed/completed proposal output.
Payment export is incomplete	read_all_payments has a 250-entry limit without public pagination.
Delete source/cycle selector fails	Use /{id}, or source_id/cycle_id for query-only deletes.
404 on an old copied request	Check the exact operation URL and method in this edition, then contact Support about account availability.

API Monitor

Open Hello [Your Name] -> API Monitor to inspect supported request logs. The monitoring guide describes a 45-day request window and UTC times, with requested URL, requester IP and payload. Use it alongside your own redacted operation ledger; request logging does not prove message delivery or downstream business completion.

Contact Support with useful diagnostics

Provide the account ID, endpoint and method, timestamp with timezone, redacted request, HTTP status, ResultCode/Message, returned IDs and expected versus actual outcome. For scheduling include the hosted-page ID, start/end offsets, slot timezone and chosen time. For a webhook include the monitor entry reference. Remove access tokens, web keys, passwords and unnecessary customer data.

Support: support@clienttether.com

11. Compatibility and reference notes

Canonical operations and compatibility forms

Use canonical paths in the endpoint reference for new work. Query-only forms exist for compatibility, but their selector names are not uniform. The table below shows the values to send.

Operation	Path selector	Query-only selector
read_client_by_id	/{{id}}	id
read_client_by_external_id	/{{id}}	external_id, or third_party_name + third_party_id
delete_client_by_id	/{{id}}	id
delete_client_by_external_id	/{{id}}	external_id
read_client_history_notes	/{{id}}	client_id
read_action_plan_by_id	/{{id}}	plan_id
delete_lead_source_by_id	/{{id}}	source_id
delete_sales_cycle_by_id	/{{id}}	cycle_id
delete_user_by_id	/{{id}}	user_id
delete_account_by_id	/{{id}}	account_id; path form recommended
delete_client_event	/{{id}}	event_id

Contact-name aliases

Canonical name	Compatibility alias
read_client_by_id	read_contact_by_id
read_client_by_external_id	read_contact_by_external_id
read_client_list	read_contacts
create_client	create_contact
update_client_by_id	update_contact_by_id
update_client_by_external_id	update_contact_by_external_id
delete_client_by_id	delete_contact_by_id
delete_client_by_external_id	delete_contact_by_external_id

Advanced proposal documents

Advanced proposal operations use the proposal engine configured for the account. Worksheets, bid-type selections, change orders, punch lists and signature representations are structured documents specific to that configuration. Read an existing quote and preserve the serialized representation. The top-level field names in this guide are stable request names for this edition; they do not imply that every account uses an identical worksheet design.

Scope of this guide

This guide covers the public v2 operations authenticated by both API headers. Email parsing, form-posting setup, legacy IVR screen-pop interfaces, telephony-provider callbacks and custom partner integrations use separate contracts. They are outside this downloadable API package. Feature access and available configuration depend on your account; use Support for account-specific setup.

Documentation edition

Edition 2026.10.22 organizes the public API around setup, shared models, integration recipes and endpoint reference. Examples use synthetic data to illustrate request and response formats. Keep the guide, OpenAPI file and Postman collection together when sharing them with a developer or AI assistant.

Additional resources

- [Published Postman collection](#)
- [API overview](#)
- [Endpoint reference](#)
- [Parameter reference](#)