Faxes

The fax endpoints send documents through the existing fax worker and retrieve retained fax history and documents. Every request requires a concrete `tenant` code or name, including requests using a global API key. Full keys can send faxes and download documents; full and read-only keys can retrieve metadata and delivery attempts.

Method	Endpoint	Purpose
POST	`/faxes`	Convert and queue one document for one destination.
GET	`/faxes`	List stored inbound and outbound fax history.
GET	`/faxes/{id}`	Retrieve one fax and its recent delivery attempts.
GET	`/faxes/{id}/download`	Download the fax document as PDF.

Sending a Fax

Submit JSON with these fields:

Field	Required	Description
`to`	Yes	One destination string: 1-80 digits, `*` or `#`, with at least one digit and an optional leading `+`. Strings preserve leading zeroes.
`document`	Yes	Object with `filename` and `data_base64`. Use a basename ending in `.pdf`, `.tif` or `.tiff`, maximum 255 characters. Encode the complete nonempty file as canonical base64, maximum 16 MiB before encoding.
`from`	No	Sender number, maximum 80 characters including an optional leading `+`; digits, `*` and `#` are accepted. Default empty; the existing DID/tenant routing selects or rewrites the outbound caller ID.
`from_name`	No	Sender name, maximum 255 UTF-8 characters without control characters.

Field	Required	Description
`quality`	No	`204x98` (default), `204x196` or `204x392`.
`page_size`	No	`a4`, `letter`, `legal`, or empty to retain the source size. When omitted, the tenant/global `FAXPAGEFORMAT` setting applies, or the original PDF size if unset.
`automatic_rotate`	No	JSON boolean; default `true`. Rotate landscape PDF pages when `pdftk` is installed.
`not_before`	No	Delivery date/time in `YYYY-MM-DD HH:MM:SS` using PBX server time. Defaults to database NOW(); past dates are rejected.
`status_email`	No	One email address for the existing fax status notifications.
`delete_after_send`	No	JSON boolean; default `false`. Tenant/system `ALWAYSDELETEFAX=on` forces deletion regardless of the requested value.
`routing_profile_id`	No	Default 0 uses the normal source-DID/tenant fax routing. Tenant keys may select the tenant's default FAX profile or a FAX profile assigned to one of its DIDs. Global full keys can select any existing FAX profile. Voice/SMS profiles are rejected.

Documents are converted to fax-ready Group 4 TIFF and must contain 1-500 pages. Both the uploaded file and final TIFF are limited to 16 MiB. TIFF uploads use an intermediate PDF, limited to 20 MiB. Invalid/unconvertible documents are rejected before a fax is queued. Conversion uses the existing fax helpers with bounded subprocess execution and private temporary files. Required tools are `pdfinfo`, Ghostscript, `tiffinfo` and `timeout`; TIFF input also requires `tiff2pdf`. Automatic rotation uses `pdftk` when available.

This Bash example sends `/tmp/fax.pdf` using `uuencode -m` and `jq`. The base64 data is read from a stream rather than passed as a shell argument, so larger documents avoid command-line argument limits:

```
jq -n \
  --arg to '+390212345678' \
  --arg from '+390298765432' \
  --rawfile data <(uuencode -m /tmp/fax.pdf fax.pdf | sed '1d;$d' | tr -d '\n') \
  '{to:$to,from:$from,document:{filename:"fax.pdf",data_base64:$data},quality:"204x196"}' \
  | curl --request POST 'https://pbx.example.com/pbx/openapi.php/faxes?tenant=CANISTRACCI' \
```

```
      --header 'X-API-Key: YOUR_FULL_API_KEY' \
      --header 'Content-Type: application/json' \
      --data-binary @-
```

A successful request returns HTTP `202 Accepted`:

```
{
  "id": 123,
  "tenant_id": 15,
  "status": "queued",
  "fax_status": "READY",
  "to": "+390212345678",
  "from": "+390298765432",
  "pages": 2,
  "size": 42110,
  "not_before": "2026-10-25 14:30:00",
  "delete_after_send": false
}
```

HTTP 202 confirms queuing, not successful delivery. The API uses the tenant/system fax storage configuration: database or S3/Wasabi. Older tenant schemas inherit the system settings. A fax becomes READY only after conversion and storage succeed. The existing `checksendfax.php` worker handles scheduling, routing, retries, channel limits and notifications. Poll `GET /faxes/{id}` for the latest stored status and error. Repeating a successful POST queues another fax; this endpoint does not deduplicate repeated submissions.

Listing Faxes

`GET /faxes` returns newest IDs first, without a total COUNT query. It accepts:

- `limit`: 1-500, default 100.
- `before_id`: continue with the `next_before_id` from the preceding page.
- `direction`: `IN` or `OUT`.
- `from`, `to`, `status`: exact stored-value matches, for example `READY`, `SENDING`, `SUCCESS`, `FAILED` or `BUSY`. Status/error values are not translated.
- `start`, `end`: inclusive fax creation date/time bounds, in `YYYY-MM-DD HH:MM:SS` using PBX server time. These filter the creation date, not the scheduled delivery time.

```
curl --get 'https://pbx.example.com/pbx/openapi.php/faxes' \
  --header 'X-API-Key: YOUR_API_KEY' \
  --data-urlencode 'tenant=CANISTRACCI' \
```

```
  --data-urlencode 'direction=OUT' \
  --data-urlencode 'start=2026-10-25 00:00:00' \
  --data-urlencode 'end=2026-10-25 23:59:59' \
  --data-urlencode 'limit=100'
```

The list response is `{"data":[...],"limit":100,"has_more":false,"next_before_id":null}`. When `has_more` is true, repeat the same filters with `before_id` set to the returned `next_before_id`.

Fax Details and Delivery Status

```
curl 'https://pbx.example.com/pbx/openapi.php/faxes/123?tenant=CANISTRACCI' \
  --header 'X-API-Key: YOUR_API_KEY'
```

List/detail metadata contains `id`, `tenant_id`, `date`, `direction`, `from`, `from_name`, `to`, `remote_id`, `status`, `attempt`, `error`, `pages`, `size`, `filename`, `status_email`, `not_before`, `delete_after_send`, `rescheduled_as`, `routing_profile_id`, `processing_icon` and `processing_note`. The document bytes and storage credentials are excluded.

Details additionally include the newest 100 delivery-attempt records in `attempts`, ordered by attempt record ID descending. `attempts_truncated` is true if older attempt records were omitted. Each attempt contains `id`, `date`, `remote_id`, `status`, `dial_status`, `error`, `attempt`, `ecm`, `rate`, `modem` and `status_extra`.

Downloading the Document

```
curl 'https://pbx.example.com/pbx/openapi.php/faxes/123/download?tenant=CANISTRACCI' \
  --header 'X-API-Key: YOUR_FULL_API_KEY' \
  --output /tmp/received-fax.pdf
```

Downloads require a full API key and return `application/pdf` with `Content-Disposition: attachment` and the actual `Content-Length`. Ordinary incoming PDF data is returned directly; outgoing TIFF and fast-on-net TIFF are converted to PDF. Database and S3/Wasabi storage use the same download flow as the web interface. Failed or partial faxes can be downloaded when retained document data is available. If the document has been deleted, its metadata can still be retrieved while the history row exists.

Errors and Compatibility

Errors use the standard JSON `error.code`/`error.message` response. Common statuses are 400 for invalid payloads, base64 or filters; 403 for read-only sending/downloading; 404 for unknown/foreign-tenant IDs, unavailable documents or disallowed routing profiles; 405 for unsupported methods; 409 for incompatible storage or missing transactional support; 422 for document conversion failure; and 502 for remote storage/download failure. Sending requires InnoDB for the fax and tenant tables. Failed preparation or publication does not leave a fax ready for delivery.

The singular path `/fax` is an alias. Query-style routing is supported with `object=fax` (or `faxes`): POST with `action=send`; GET with `action=list`, `action=get&id=123`, or `action=download&id=123`. Every form requires `tenant`.

Revision #1
Created 2026-10-25 21:13:54 UTC by Admin
Updated 2026-10-25 21:13:54 UTC by Admin