

FirearmDistributors.com Bulk API

Documentation. Also online at <https://www.firearmdistributors.com/api-docs#bulk-api>

Bulk API

API v1 Beta Current and supported. Document updated 10 August 2026. [See the changelog](#)

This is a beta test system. It is live and stable enough to build on, and we are actively refining it. If you run into a problem, or you have a feature you would like us to add, please tell us. That feedback shapes what ships next. Email support@firearmdistributors.com.

Built for dealers who want to power their own website, price book, or internal systems with our pricing and inventory data: a full catalog dump, incremental updates, single item lookups, and a shared product image dump. \$99 per month, billed separately from the website subscription, no trial.

Read-only. The Bulk API never places, changes, or cancels orders, and it never will. Ordering stays in the website, where it is tied to your own distributor credentials. An API key cannot authenticate an order.

Getting started in three steps

1. Subscribe to the Bulk API. Contact support and we will send you a checkout link.
2. Go to [your API Keys page](#) and create a key. The key is shown once and never again, so store it before leaving the page.
3. Call `/api/v1/status` to confirm the key works, then pull your first catalog dump.

```
curl -H "Authorization: Bearer fd_your_key_here" \
https://www.firearmdistributors.com/api/v1/status
```

Authentication

Every request carries your key in the Authorization header. Keys begin with `fd_`. Send the key as a header, never in the query string, so it stays out of server logs and browser history.

```
Authorization: Bearer fd_<your key>
```

Your API Keys page lets you create up to three active keys, rotate a key, and revoke one immediately. Rotating creates the replacement first and then revokes the old key, so you are never left without a working key. You can also restrict a key to specific IP addresses: enter one per line, as a plain address, an IPv4 range in CIDR form such as `203.0.113.0/24` , or a literal IPv6 address. A key with an allowlist is refused from any other address.

We store only a hash of your key, so we cannot recover it for you. If a key is lost or exposed, revoke it and create a new one.

Rate limits

Limits are per account, per endpoint, and reset at 00:00 UTC. They are deliberately tight because the catalog dump is designed to be pulled once a day, not polled.

Endpoint	Calls per day	Intended use
<code>/api/v1/catalog/dump</code>	2	Your daily full sync, plus one retry
<code>/api/v1/images</code>	2	Product images, changes slowly
<code>/api/v1/updates</code>	12	Intraday freshness between dumps
<code>/api/v1/item</code>	100	Spot checks and troubleshooting
<code>/api/v1/status</code>	24	Health and quota checks
<code>/api/v1/catalog/download</code>	Not counted	Downloading a file you already requested

Successful responses carry `X-RateLimit-Limit` and `X-RateLimit-Remaining` . Over the limit you get HTTP 429 with a `reset_at` timestamp. Requests that fail authentication or hit the limit do not count against your quota, so a rejected call never costs you a call.

Need more? The limits are adjustable per key. Tell us what your integration needs.

How to keep your data in sync

This is the part worth reading carefully, because it determines whether your catalog stays correct.

The nightly full dump is your source of truth. Pull it once a day and reconcile against it. The `/api/v1/updates` endpoint is a freshness hint for the hours in between, not a complete change

log.

Two specific reasons the dump has to be authoritative:

- **Delistings do not appear in updates.** When a distributor stops carrying an item, we zero its quantity, but that change does not surface through the updates feed. It appears in the next full dump with `quantity` of 0. If you sync only from updates, you will keep listing items that are no longer available.
- **A distributor feed run marks every one of its rows as touched**, whether or not the price or quantity actually moved. A single day of updates can therefore exceed what 12 calls of 10,000 rows can return. Updates will keep you fresh on recent activity, but they cannot reconstruct a full day.

A good integration looks like this: download the full dump every morning and reconcile your catalog against it, call updates a few times during the day to catch recent price and stock movement, and use the item endpoint when a customer is looking at a specific product and you want the freshest number.

Endpoint reference

GET /api/v1/catalog/dump

Returns metadata and a temporary download link for your most recent nightly dump. It does not return the file itself. Dumps build every night at 04:00 Eastern.

```
{
  "generated_at": "2026-08-10T19:42:08+00:00",
  "row_count": 400306,
  "size_bytes": 15935771,
  "url": "https://www.firearmdistributors.com/api/v1/catalog/download?acct=...&sig=...",
  "url_expires_at": "2026-08-10T20:00:00+00:00"
}
```

The link is valid for 15 minutes. Downloading does not count against your quota, so if a download fails you can request a fresh link. Because the link authenticates on its own signature, it keeps working for up to 15 minutes even after you revoke the key that created it. Treat a live link like the key itself.

If you have subscribed but no dump has been generated yet, you get HTTP 404 with an explanation. Your first dump can be built on request, so contact us rather than waiting for the

overnight run.

GET /api/v1/catalog/products

The product catalog, built for powering a store: **one row per product**, not one row per distributor offer. Deduplicated on UPC, cleansed, with the full detail you need for product pages and search: title, plain-text description, manufacturer, model, MPN, category, caliber, gauge, chamber type, country of origin, an FFL flag, a hazmat flag, a primary image URL, and any additional images. It also carries the best cost, MAP, total quantity across distributors, and a distributor count as a convenience.

The intended pattern: pull this once to build and populate your catalog, then use the catalog dump, `updates`, and `item` for live per-distributor pricing and stock. Same metadata-plus-signed-link shape as the catalog dump, and a 2 per day limit.

```
curl -H "Authorization: Bearer fd_your_key_here" \
  https://www.firearmdistributors.com/api/v1/catalog/products
```

Columns: `upc`, `sku`, `internal_sku`, `mpn`, `manufacturer`, `title`, `description`, `category`, `caliber`, `gauge`, `chamber_type`, `made_in`, `ffl_required`, `is_hazmat`, `image_url`, `additional_images`, `best_cost`, `map_price`, `total_quantity`, `in_stock`, `distributor_count`, `last_synced`. Join it to the images file and the offer dump on `upc`.

GET /api/v1/updates

Rows whose pricing or stock was touched since a point in time. Maximum 10,000 rows per call, ordered oldest first.

Parameters: `since` is required, in ISO 8601 form, and cannot be older than 48 hours. `cursor_upc` is optional and is used to resume mid-page.

```
curl -H "Authorization: Bearer fd_your_key_here" \
  "https://www.firearmdistributors.com/api/v1/updates?since=2026-08-10T00:00:00Z"
```

```
{
  "count": 10000,
  "next_since": "2026-08-10T06:14:22+00:00",
  "next_upc": "816161020135",
  "rows": [ { "upc": "...", "cost": 30.09, "quantity": 4, "source": "reference", ... } ]
}
```

Always send back both `next_since` and `next_upc` on your following call. Many rows share the exact same timestamp, so the timestamp alone is not enough to resume from and using it by itself will skip rows. When there is nothing new, you get zero rows and the same cursor values back, which you simply reuse next time. A `since` older than 48 hours returns HTTP 409, which means you have fallen too far behind and should pull a full dump instead.

GET /api/v1/item

Live lookup for a single product across every distributor available to you. Takes `upc`, 8 to 14 digits. Returns HTTP 404 if the UPC is not in our catalog.

```
curl -H "Authorization: Bearer fd_your_key_here" \
  "https://www.firearmdistributors.com/api/v1/item?upc=098289001375"
```

GET /api/v1/images

Same shape as the catalog dump: metadata plus a 15 minute download link, for a shared gzipped CSV of product images.

GET /api/v1/status

Per distributor freshness, so you can see when each feed last updated, plus your quota usage for the day. Useful as a health check before a sync run and as the first thing to look at when a number seems wrong.

Catalog dump format

Gzipped CSV, one row per product and distributor combination, header row always present:

```
upc, sku, internal_sku, manufacturer, title, model_number, category,
distributor_code, distributor_name, cost, map_price, quantity, in_stock,
source, last_synced
```

Parse this with a real CSV reader. Product titles contain commas, quotes, and sometimes line breaks, so one record is not always one physical line. In a recent build, 66 of 400,306 records spanned more than one line. Splitting the file on newlines will corrupt those records. The `row_count` we report is the record count, which is what a CSV reader gives you, and it will be lower than the physical line count.

Column	Meaning
<code>upc</code>	The product identifier, and the only safe key for joining our files together
<code>sku</code>	The distributor's own SKU for this offer. The same product has a different value under each distributor
<code>cost</code>	Your cost for this offer. See <code>source</code> for whose cost it is
<code>map_price</code>	Minimum advertised price where the manufacturer sets one, otherwise empty
<code>quantity</code> , <code>in_stock</code>	Stock at the last sync. Always read alongside <code>last_synced</code>
<code>source</code>	<code>own</code> means this is your own negotiated pricing, pulled with your credentials. <code>reference</code> means it is our published reference pricing for a distributor you have not connected
<code>last_synced</code>	When we last pulled this row from the distributor. Some rows are old. Check this before treating a number as current

Reference pricing is not your pricing. Rows marked `reference` are a published benchmark, not what you would pay. Connect that distributor's account on your credentials page and those rows become `own` , showing your real negotiated cost. Quoting a customer directly from a reference row is the most common mistake an integration makes.

Images dump format

```
upc, sku, internal_sku, image_url
```

Join the images file to the catalog file on `upc` only. The `sku` column means different things in the two files: in the catalog it is the distributor's SKU, in the images file it is our catalog SKU. Joining on `sku` will produce wrong matches.

Download the images and host them yourself rather than linking to our URLs on a busy site. Self hosting is faster for your customers and protects you from any change on our side.

Which distributors you see

You see reference pricing for every distributor we publish, and your own pricing for any distributor whose account you have connected. Two exceptions: Lipsey's and Crow only permit their pricing to be shown to dealers who hold an account directly with them, so those rows are absent entirely

unless you have connected your own credentials for them. This applies everywhere, including this API. If you hold accounts with them, connect the credentials on your credentials page and their rows appear.

Response codes

Code	Meaning and what to do
200	Success
400	A parameter is missing or malformed. The message names the problem
401	Missing, invalid, or revoked key. Check the Authorization header
402	No active API subscription. The response includes a link to reactivate
403	Your key has an IP allowlist and this address is not on it, or a download link signature did not verify
404	Nothing to return: an unknown UPC, or no dump generated yet
405	Only GET is supported. The API is read-only
409	Your <code>since</code> is more than 48 hours old. Pull a full dump instead
410	A download link expired. Request a new one, which does not cost a call
429	Daily limit reached. The response tells you when it resets
503	Temporary problem on our side. Retry after the interval we return

A working daily sync

```
#!/bin/bash
KEY="fd_your_key_here"
BASE="https://www.firearmdistributors.com/api/v1"

# 1. Ask for today's dump and pull the download link out of the response
URL=$(curl -s -H "Authorization: Bearer $KEY" "$BASE/catalog/dump" \
  | python -c "import sys,json; print(json.load(sys.stdin)['url'])")

# 2. Download it. This does not count against your quota
curl -s -o catalog.csv.gz "$URL"
```

```
# 3. Reconcile your catalog against the file, then do the same for images
# once a day. Use /updates during the day for recent movement.
```

Schedule this for the morning, after the overnight build at 04:00 Eastern has finished. Check `generated_at` in the response: if it is not today's date, the overnight build did not run and you are looking at the previous file, which we keep in place deliberately so you always have something usable.

Versioning and what will not change

The version is in the URL. Everything documented above is **v1**, served from `/api/v1/`. The `/api/v1/status` response also returns `api_version` so your integration can log and assert which contract it is talking to.

While you are on v1, you can rely on these:

- Existing fields keep their name and meaning. We will not repurpose a column or change what a value means underneath you.
- New columns are **appended to the end** of a dump file, never inserted in the middle. Read columns by header name rather than by position and a new column costs you nothing.
- New fields may appear in JSON responses. Ignore what you do not recognise.
- Response codes keep their meaning.

Anything that would break a working integration ships as `/api/v2/` instead, and v1 keeps running while you migrate. If we ever need to retire a version we will email every active key holder first and give a migration window. Quota numbers are the one thing that can move within a version, and they only move up unless we contact you.

Changelog

Newest first. Additive changes are listed here rather than announced by email, so check this section when something looks new.

v1.0 • 10 August 2026 • First release

- Endpoints: `catalog/dump`, `catalog/products`, `catalog/download`, `updates`, `item`, `images`, `status`.
- Product-level catalog (`catalog/products`): one cleansed row per product with full detail for building a store, alongside the per-offer catalog dump.

- Nightly per-account catalog file with a 15 minute signed download link, and a shared product image file.
- Pricing for every distributor with no credentials required. Crow and Lipsey's stay out unless your account is approved for them.
- Optional account setting that removes every item requiring an FFL from your feed entirely. Ask support if you want it on.
- Per account daily quotas, with `X-RateLimit-Limit` and `X-RateLimit-Remaining` on successful responses. Rejected calls do not count against your quota.
- Self-service key management: create, rotate, revoke, and see your usage.
- `updates` uses a two-part cursor. Send back both `next_since` and `next_upc` . Many rows share one timestamp, so the timestamp alone is not enough to resume from.

Getting help and giving feedback

This is a beta system and your feedback is welcome. If something is not working, or you have a recommendation you would like us to consider, please reach out. We would rather hear it than have you work around it.

Email support@firearmdistributors.com. For a problem, include the endpoint, the time of the call, and the response code you received. Never send us your key. If you think a key has been exposed, revoke it on your API Keys page first and tell us afterwards.