BROKER CONSOLE · MODULE REFERENCE

Trading Solutions
the products a broker sells

The nine things a brokerage sells on top of a trading account: copying, managed and pooled money, funded-account challenges, competitions, promotions, a bot the client talks to, and the watchdog on the prices behind all of it.

Copy engine and its platform rules · MAM mandates and PAMM pools · Phased challenges, enforced on the tick · Contests and prediction campaigns · Telegram account linking · A rehearsal before every payout

PRODUCT	MODULE	CONTENTS	VERSION
HonorPro Hub	Trading Solutions	9 menus · 10 tabs	1.0 · 24 Sep 2026

Copy Trading	Engine health, the platform rules, and the accounts the desk has blocked
Social Trading	The feed beside the copy board, and the desk that moderates it
MAM	Mandates, the allocation plan, replication health and the management fee
PAMM	Offers, the pool in units, the rollover and the investor statement
Telegram Bot	The broker's own bot, who is linked to it, and what it was asked
Tournaments	Contests, the frozen starting snapshot, the board and the prize run
Prop Challenges	Challenges and storefront keys · Enforcement, on the tick
Prediction Campaigns	Promotions, the guesses, the board and the settlement
Price Alerts	The quote-stall watchdog: rules, incidents and delivery

01 THE MODULE

Nine products, and one rule about money

Every screen in this module ends in the same place: an instruction to a trading server about somebody's account or somebody's balance. That is why none of them commits on the first click. A screen that moves money rehearses first, names who would be paid and how much, and records the run — and the seat that runs it holds a second permission the seat that configures it does not.

9

MENUS

Eight are separately licensed components; the Telegram bot is part of the installation

10

TABS

Each one an address a person can paste to a colleague

6

JOURNALS

Copying, MAM, PAMM, tournaments, challenges, campaigns — each keeps its own

Rehearse, then commit

The pattern is the same on every screen that ends in a balance operation, under whichever verb that screen uses:

- **Dry run** — MAM's allocation push and fee run
- **Rehearse** — the PAMM rollover, the tournament prize table, a campaign settlement
- **Show me the plan** — every prop-attempt action

A rehearsal books nothing and is deliberately not recorded as a run. The commit that follows is keyed, so pressing it twice retries only the lines that failed and never pays anybody a second time.

Two permissions, not one

Configuring a product and paying out of it are separate grants. The tournament desk, the campaign desk, the PAMM rollover, the MAM fee run and a prop payout each refuse a seat that holds the product's own permission but not **balance**, and each says which of the two is missing rather than answering a flat refusal.

What this module does not do

Nothing here places, modifies or closes a trade. The copy engine is a separate process the gateway talks to; tick-level breach detection runs in a plugin inside the broker's MT5 server. The console's job is the policy, the record and the decision — the execution belongs to the machine that owns it.

A product this brokerage has not licensed is not drawn — not greyed out, not listed. Eight of the nine menus carry their own component gate, and **Enforcement** carries a second one inside an entitled Prop Challenges menu.

02 MENUS · COPY TRADING · SOCIAL TRADING

The engine, and the room it is watched in

Copying is bought by a client and run by an engine; the feed is where the clients talk about it. Both desks are read-and-govern rather than operate: the broker sets the rules, answers what is reported, and leaves the replication and the conversation to the people doing them.

Copy Trading `/copy-trading`

The state of the copy engine in one screen: copy lag at the median and the 95th percentile, open links, refusals in the last hour and the retry backlog, with the shard workers and what was replayed at the last start. Under it, the incidents — refusals after a full retry ladder, drawdown halts, refusal spikes — the platform rules this installation is running, and every account the desk has blocked from copying, with the reason and who blocked it.

The rules are shown here and changed in the trading desk: copying on or off, whether new masters are accepted, the maximum fee a master may charge, the minimum age of a master's history, the lot and open-position ceilings per follower, the forced drawdown stop, and which of the three sizing modes — fixed, proportional, multiplier — this brokerage allows.

Lag p50 / p95 · Open links · Refusals per hour · Platform rules · Blocked accounts

Social Trading `/social-trading`

The moderation desk for the feed inside the terminal, in five parts. **Feed** is every post with what a reader never sees — hidden, muted, blocked, report count. **Reports** is the queue a reader put there, for a post or for a conversation. **Blocks** is who may not write, why, and until when. **Analysis** counts what looks like a pattern — posting bursts, velocity, duplicates, keyword groups, and the built-in detectors for phone numbers, wallet addresses, IBANs, handles and domains — and shows the evidence beside each reading and the threshold it crossed. **Settings** holds the keyword lists and thresholds.

Four instruments, deliberately distinct: delete is final, hide is a curtain the author still sees with its reason, mute stops an account composing into feeds, and a block stops them writing at all. Every one of them takes a reason, and every one lands in the journal.

Feed · Reports · Blocks · Analysis · Settings · Journal

Direct messages are unmoderated by design. There is no route in this product that lists them, and the only door into a conversation is the reported-thread queue — which opens on a thread only after **a participant in it** asked the desk to look.

The engine is not in the gateway

Masters, subscriptions, fees and blocks live in the copy engine's own registry, a separate process the gateway calls. An installation without one answers **not available here**; one whose engine is configured but silent answers **unreachable**. The two are never merged, because an empty board and a broken tunnel are different facts.

Who may post

Only a public master may post, and that is rechecked against the live board on every attempt rather than from a cache. A ticket attached to a post must be one from the author's own history, checked without confirming that anybody else's ticket exists.

03 MENUS · MAM · PAMM

One manager's accounts, or one account's owners

The two managed-money products answer different questions. Under a MAM mandate each client keeps their own trading account and the manager's trades are replicated into it on an allocation. In a PAMM pool there is one trading account, and an investor holds units of it rather than positions.

MAM `/mam`

A mandate is one manager's authority over a set of accounts, and the screen is five tabs of it. **Accounts** attaches and detaches them, with a per-account risk multiplier and the record of everyone who has left and why. **Allocation** computes what each account's subscription should say — by equity, balance, fixed lot or percent — and shows the plan, per account, before a push; the push itself can be a dry run. **Summary** is replication health: open positions, net lots by symbol and realized result over a window, with balance operations excluded. **Fees** invoices the management fee for a closed period, previews it per account, and keeps the ledger of every run. **Journal** is the record behind all of it.

MAM is a policy layer over the copy engine, not a second engine, and it names the places the engine cannot do what a mandate asks. The one that matters commercially: the engine has no per-subscription symbol filter, so an exclusion recorded here is **not enforced**, and a push touching such an account is refused until an operator acknowledges that against their name.

Accounts · Allocation · Summary · Fees · Journal · Dry run

PAMM `/pamm`

An offer carries its fees — management, performance with an optional high water mark, and entry — the length of its trading interval, the minimum investment and the share the manager must keep of their own pool. **Pool** lists the investors in units, their share and value and what they have accrued. **Requests** is where a deposit or withdrawal is filed; it moves nothing. **Rollover** closes an interval: value the pool, charge the fees, redeem, subscribe, and book one netted balance operation per account. **Statement** rebuilds an investor's history interval by interval.

Pool · Requests · Rollover · Statement · Journal · Rehearse

Money moves only at the boundary of an interval, in one batch. A rollover is identified by **the boundary it closes**, not by the click that asked for it: asking twice returns the run already recorded and moves nothing. The one exception is an opt-in immediate withdrawal, at the penalty the offer publishes.

The manager's own stake

Fees are charged by burning units, never against the manager's own holding. A partial redemption by the manager is capped so it cannot take them under the minimum share the offer promises; if a rollover does, the offer is marked breached and refuses new money until it is restored.

Closing a mandate

A mandate cannot be closed while any account is still attached: closing is bookkeeping, detaching is what stops the copying. A detach must state, with no default offered, whether the copied positions close now or run on.

04 MENUS · TELEGRAM BOT · PRICE ALERTS

What a client may ask, and what the desk is told

Two messaging products that are not the same product. The bot is the conversational half — a client asks their own account a question. The watchdog is the outbound half — the desk is told when the prices stop. Neither delivers for the other, and the console says so on both screens.

Telegram Bot `/telegram-bot`

Four tabs. **The bot** holds its identity — token, username, brand name — the four commands that can be switched on and off individually (*/balance*, */equity*, */positions*, */quote*), the webhook secret, and a test message to prove the token works before anybody links anything. **Who is linked** lists every chat holding an account, with the moment it was linked and last heard from, and revokes one. **What it was asked** is the rolling record of the questions — never the answers, because nobody needs a second copy of a client's balance on the gateway. **Journal** is the settings and linking history.

Identity and kill switch · Per-command switches · Webhook secret · Links · Asks · Journal

The bot token never comes back over the wire: the console is told only whether one is set. The webhook secret is shown once, on minting, and minting a new one stops the old one at that moment. **Switching the bot off keeps every link** — the desk turns it off during an incident and back on after, and nobody has to link again.

How an account is linked, and why it is built this way

A bot that answers "balance" to whoever messages it is a data breach with a friendly interface. Telegram proves nothing about who is typing, so the only question this product asks is which account a chat has proved it holds:

- **The code is minted inside the client's own session.** It is reachable only behind their access token, so it names an account the requester already proved they hold. There is no way to link one from this console, and that is the point.
- **96 bits, hashed, single-use, ten minutes.** The code is stored only as a hash peppered with the installation's own secret, and asking for a new one replaces the last rather than leaving two doors open.
- **Guessing is throttled per chat, never per account.** Throttling the account would let a stranger lock a client out of linking their own. Five wrong codes and that chat waits fifteen minutes.
- **The webhook is authenticated, and refuses everything while no secret is set.** A URL is not a credential.
- **A group chat is refused outright**, and an unlinked chat is given an instruction and never a figure. No route here takes a login from a message — the chat id is the only input that reaches an account read.

Price Alerts `/alerts`

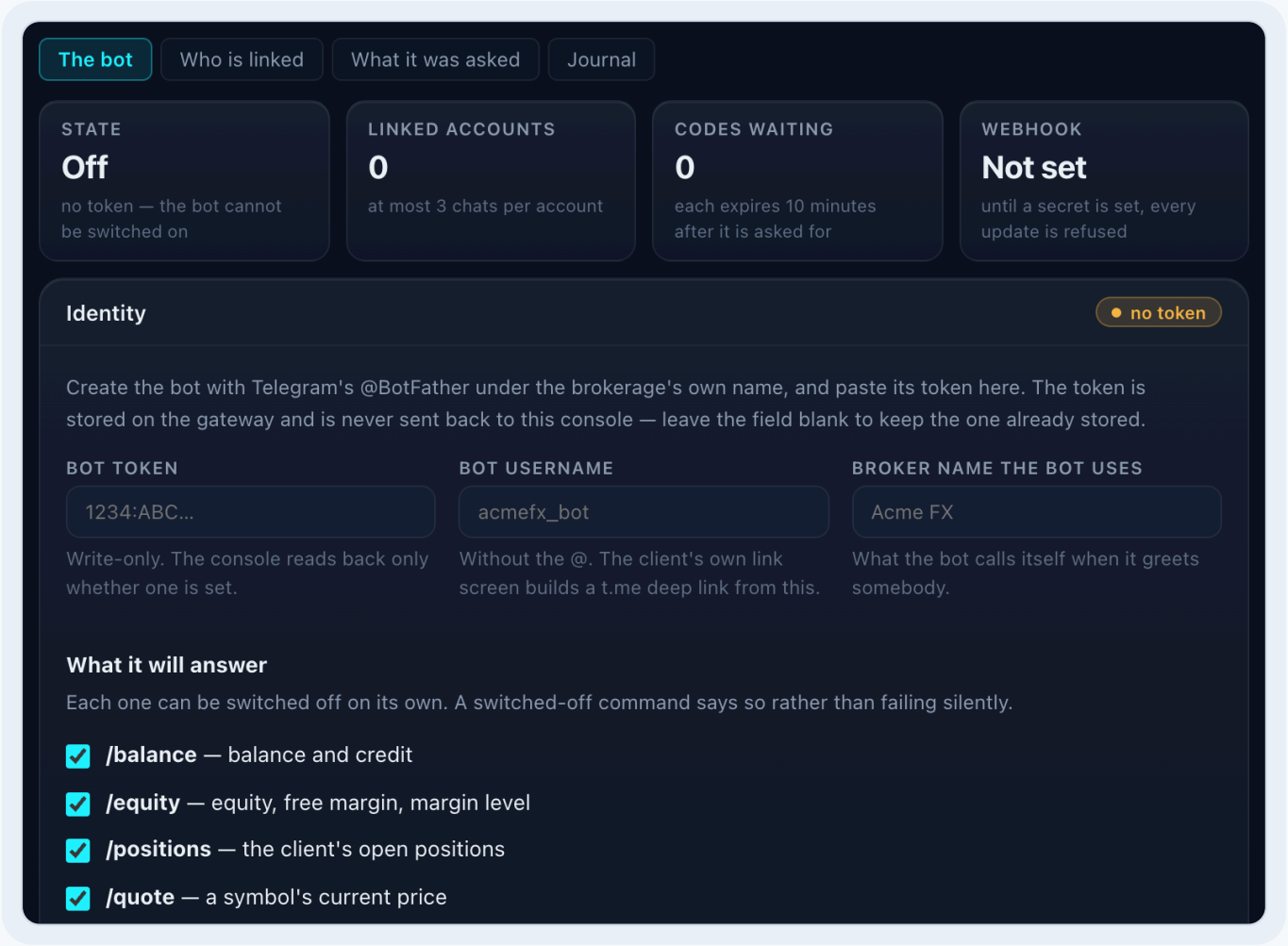
The quote-stall watchdog. **Rules** is what is watched and how patient the desk is with it — an explicit symbol list or the whole book minus exclusions, a default and per-group patience, the poll interval, and the weekend close and open so a Sunday reopen does not alarm on Friday's last price. **Incidents** is what it raised, kept in this Hub so the tab still answers while the gateway is down. **Delivery** shows which channels are bound and tests them.

List or whole book · Patience by group · Tick age per symbol · Incidents · Delivery

05 LINKING AN ACCOUNT

Proved once, inside the client's own session

The bot's own tab as the desk sees it. Four figures answer the only questions worth asking of it: whether it is on, how many chats hold an account, how many codes are live, and whether the webhook is authenticated.



Trading Solutions · Telegram Bot, on a pilot with no bot configured. Off, nothing linked, no webhook secret — the state a brokerage starts in, stated rather than dressed up.

What the four tiles answer

The state, and why it cannot be switched on before a token exists. How many chats hold an account, against the ceiling of three per account — a phone, a tablet, a desktop, not a channel. How many link codes are live, each ten minutes old at most. And whether the webhook carries a secret, because until one is set every update is refused.

Write-only by construction

The token field is write-only: the console is told whether one is set and never what it is, and a blank save keeps the stored one so a save cannot wipe it by accident. Forgetting it deliberately also switches the bot off, because a bot with no token cannot answer. Each of the four commands is switched on its own, and a switched-off command says so rather than failing silently.

06 MENUS · TOURNAMENTS · PREDICTION CAMPAIGNS

A contest anybody can be shown the arithmetic for

Both products end with a broker paying money to a client in public. Both are built so that the answer to "why did they win" is a number the desk can point at, frozen at a moment nobody can move afterwards.

Tournaments `/tournaments`

A contest is scored on return, profit or volume, over a window, with entry either closed at the start — everyone measured over the same days — or open until the end, each entrant measured from their own join. **Overview** carries the terms and the public board's address. **Participants** enters and removes accounts, disqualifies and reinstates. **Leaderboard** is the worked table: starting equity, equity now, funding, profit, return, volume, trades and place. **Flags** is what looked wrong. **Finalize** rehearses the prize table and then pays it. **Journal** records every step.

Starting the contest freezes every entrant's equity, and that number never changes again. A second start is refused, because re-taking the snapshots would silently re-base every score onto whatever the market did in between. Every deposit, withdrawal, credit and correction is stripped out of the score, so funding a bigger account cannot win — commission stays in, because it is a real trading cost.

Return · profit · volume | Frozen snapshot | Flags | Rehearse then pay

Prediction Campaigns `/campaigns`

A promotion in which a client guesses where an instrument will be at a fixed moment. **Participants** previews who is eligible against the live book — minimum balance, minimum trades, groups in and groups out — and refuses a group pattern matching nothing on the server by naming it, because an empty list and a typo look identical on a screen. **Predictions** is every guess with when it was first entered, when it was last changed and how many times. **Leaderboard** ranks by distance from the price, and can be run against a hypothetical price before settlement. **Settle** fixes the price, names every winner and the amount before the operator commits, and keeps the payout ledger. **Journal** holds the rest.

Closest or exact | Bid · ask · mid | Prize table | Payout ledger

Frozen the moment somebody enters

Once one client has entered a campaign, the symbol, the three moments, the rule, the tolerance, the price side, the prize and the eligibility can no longer be changed — a client based their guess on them. The title and the description stay editable for the campaign's life.

Eligibility is checked at entry and never rechecked at settlement: a client who entered legitimately and then withdrew their balance still won. Retroactive disqualification is how a promotion becomes a complaint.

Struck out, not deleted

A disqualified entrant keeps their row, their score and the reason, and simply holds no place. Deleting a cheat from the board is how a broker gets accused of inventing the winner.

Editing a prediction moves it to the back of the tie queue, so nobody can hold an early place and swap to somebody else's answer at the last second.

07 MENU · PROP CHALLENGES

The challenge, and the rules that fail it

Two tabs of one menu, on purpose. A challenge is a product with a price, phases and a funded account at the end; enforcement is the machinery that decides, on the tick, that somebody has failed one. They belong beside each other because a rule read off a different screen is a rule nobody reconciles.

Challenges `/prop`

A challenge carries the account size, the funded size, the leverage and groups, the price and the reset price, the profit split, the payout cycle and up to five phases. Each phase states its profit target, its daily and overall drawdown limits and its minimum and maximum trading days — a phase with no way to fail is refused, because that is a funded account, not a phase. **Attempts** sells one, shows its progress against all three bars, issues the funded account once it has passed, sells a reset, or cancels it. **Funnel** is how many reached and passed each phase. **Journal** is every event on every attempt.

The terms are frozen onto the attempt when it starts, so editing the challenge afterwards never moves a trader's goalposts. Storefront keys let a third-party shop site sell and manage attempts with no console session; a key's secret is shown once, can be rotated with a grace window, and no MT5 group name ever leaves that surface.

Phases · Attempts · Funnel · Storefront keys · Plan then commit

Enforcement `/enforcement`

The supervision document — who is watched, on what limits, anchored to what, and what happens to them on a breach — plus the inbox the MT5 plugin posts breaches into and the incident record behind each one. An incident carries what was read, what the gateway asked for, what it read back afterwards, and a timeline through detected, acted and confirmed. Releasing a breached account takes a reason and returns it **paused**, never active: its anchors are stale, and reactivating it on the old numbers would breach it again on the next tick.

Supervised register · Incidents · Document · Plugin fetch · Journal

The screen states plainly when it is not protecting anything. If the plugin is not checking in, a banner says that nothing on this gateway is checking a limit on the tick and that the supervised list is a document, not a guarantee. If the plugin is running an older export than the document would produce now, it says that too.

Two rules shape the whole thing

An account nobody put under supervision is never touched — an event for an unknown or paused login is recorded and not acted on, which is what stops a plugin fault or a forged event moving a live client into a read-only group. And a supervision document that fails to validate refuses loudly rather than exporting an empty list, because an empty list would switch enforcement off for the whole brokerage in silence.

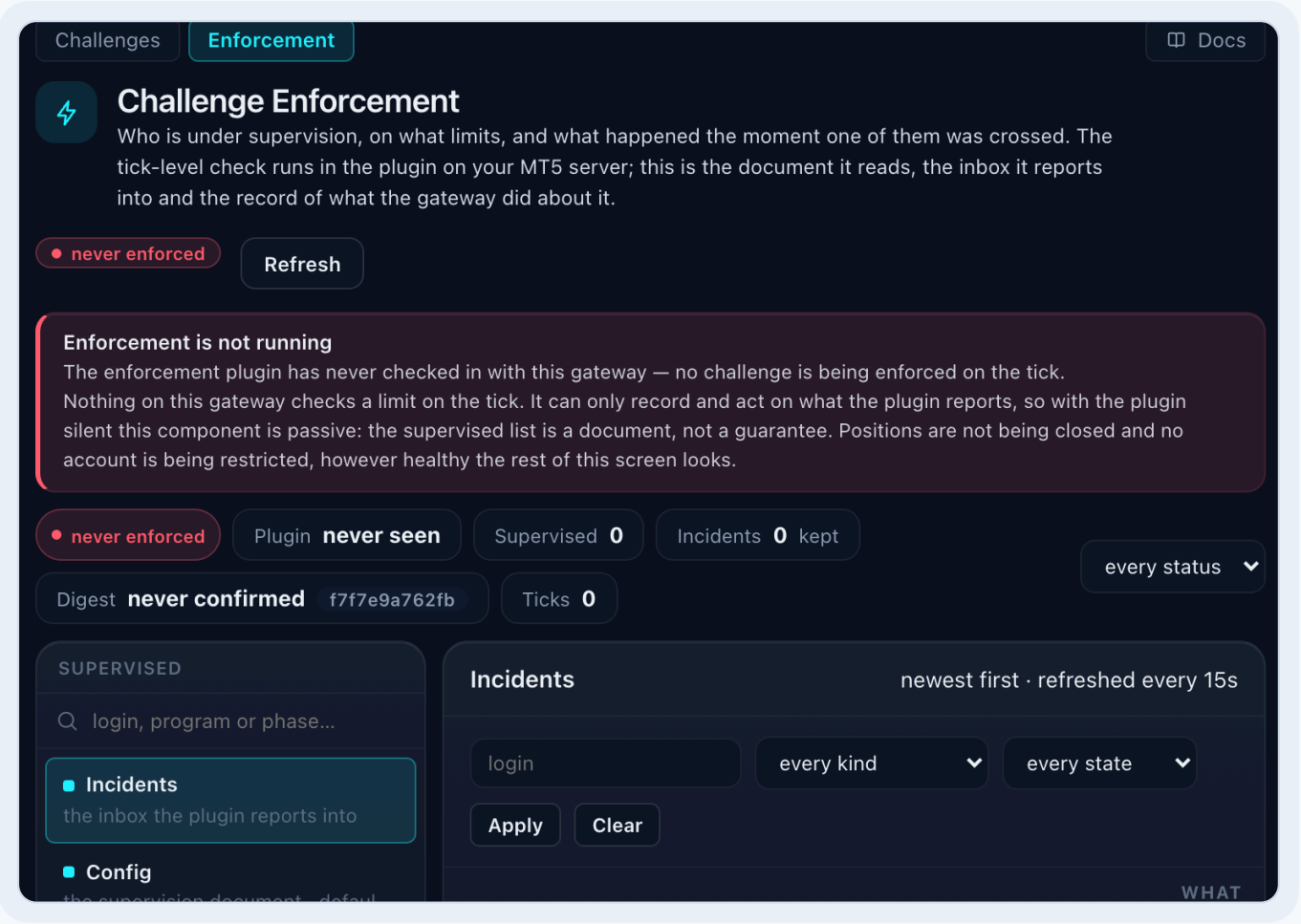
Restrict first, flatten second

On a breach the gateway restricts trading before it closes anything, which shuts the race where a breached account reopens positions as fast as they are closed. It then reads the account back rather than trusting the request: anything that did not verify is listed as still outstanding, not swallowed.

08 WHEN NOTHING IS ENFORCING

A screen that refuses to look healthy

Enforcement on an installation whose MT5 plugin has never checked in. Rather than an empty supervised list under a calm header, the screen states in full what is and is not happening to the accounts it names.



Trading Solutions · Prop Challenges · Enforcement, on a pilot with no enforcement plugin installed. The chips carry the plugin's last heartbeat, the supervised count, the incidents kept and the digest the plugin last confirmed.

Why the banner is red

Nothing on this gateway is checking a limit on the tick, so positions are not being closed and no account is being restricted, however healthy the rest of the screen looks. The supervised list is a document, not a guarantee, and the screen says so before a broker reads it as one.

An empty inbox is not "nothing happened"

The incidents inbox says as much in its own words: an empty inbox means nothing has been reported, which is the same thing as nothing happened only while the plugin is reporting. The digest beside it answers the other half — whether the plugin is enforcing the document this gateway would export now, or an older one.

09 HOW IT CONNECTS

Where each of these writes

These are the products in the console that change something outside it — what each one reads, and what it may write.

TAB	READS	WRITES
Copy Trading	The copy engine's health, rules and audit of blocks	Nothing — rules and blocks are changed in the trading desk
Social Trading	The feed store, the report queue and the live copy board	Hides, mutes, blocks, deletions and keyword settings, each with a reason
MAM	Mandates, account balances and deal history	Copy subscriptions, through the engine; the fee run's balance operations
PAMM	The offer document and the master account's equity	One netted balance operation per account, at the boundary
Telegram Bot	Its own settings, links and ask log	Settings, the webhook secret, and revoking a link
Tournaments	Entrant balances and deal history, live on every board read	The frozen start snapshot, and the prize payouts
Challenges	Attempt accounts, equity and balance, on a cadence	Accounts provisioned, funded accounts issued, payouts booked
Enforcement	The plugin's breach reports and heartbeat	Trading restrictions and group moves on a breached account
Prediction Campaigns	One quote at the fixed moment, and the client book for eligibility	One balance operation per winner, against a claim ledger
Price Alerts	Tick freshness across the watched symbols	Its own rules; incidents are pushed to this Hub

Scoped desks and installation-wide acts

Where a brokerage runs several desks over one server, per-account data — entrant lists, boards, attempt lists — is cut to the desk's own basket, while creating, starting, cancelling or settling a contest is refused to a scoped seat outright. A rank is computed over the whole field first and cut afterwards, so a desk reads "3rd of 214" and never a misleadingly small one.

Shown to the client, or not

MAM, PAMM, Tournaments, Prediction Campaigns and Prop Challenges each carry a switch deciding whether the client's terminal shows that product at all, on the same screen as the product itself. Prop Challenges also publishes to a storefront and Tournaments to a public board — by key or by address, never by opening the console.

Every one of these products keeps an append-only journal of what was done, by whom, and what the machine on the other end said back. **"Why was this account restricted on the 14th"** is a question this module can answer months later.