Offing: Following Seas — Technical Report

Oct 3, 2026 · @Kevin

What it is

Offing: Following Seas is a WordPress plugin, version 1.3.7, that shows where a boat is now. An administrator adds a vessel by its MMSI number; a shortcode then puts a marine radar display on any page, carrying the vessel's position and speed, other AIS traffic around it, nearby marinas and anchorages, and a NOAA nautical chart underneath.

The offing is the part of the sea visible from the shore. That is where the people following a boat are watching from, and they are the audience: families and friends ashore, not navigators. The display is read-only and public; nothing a visitor does changes anything.

It is one self-contained plugin, built to be installed on any WordPress site and operated by one person. There is no hosted service, no account, and no paid tier. The only outside requirement is a free API key from aisstream.io and a cron job that runs every minute.

Built with

Nothing is compiled and nothing is fetched at install time. The plugin ships as the files that run.

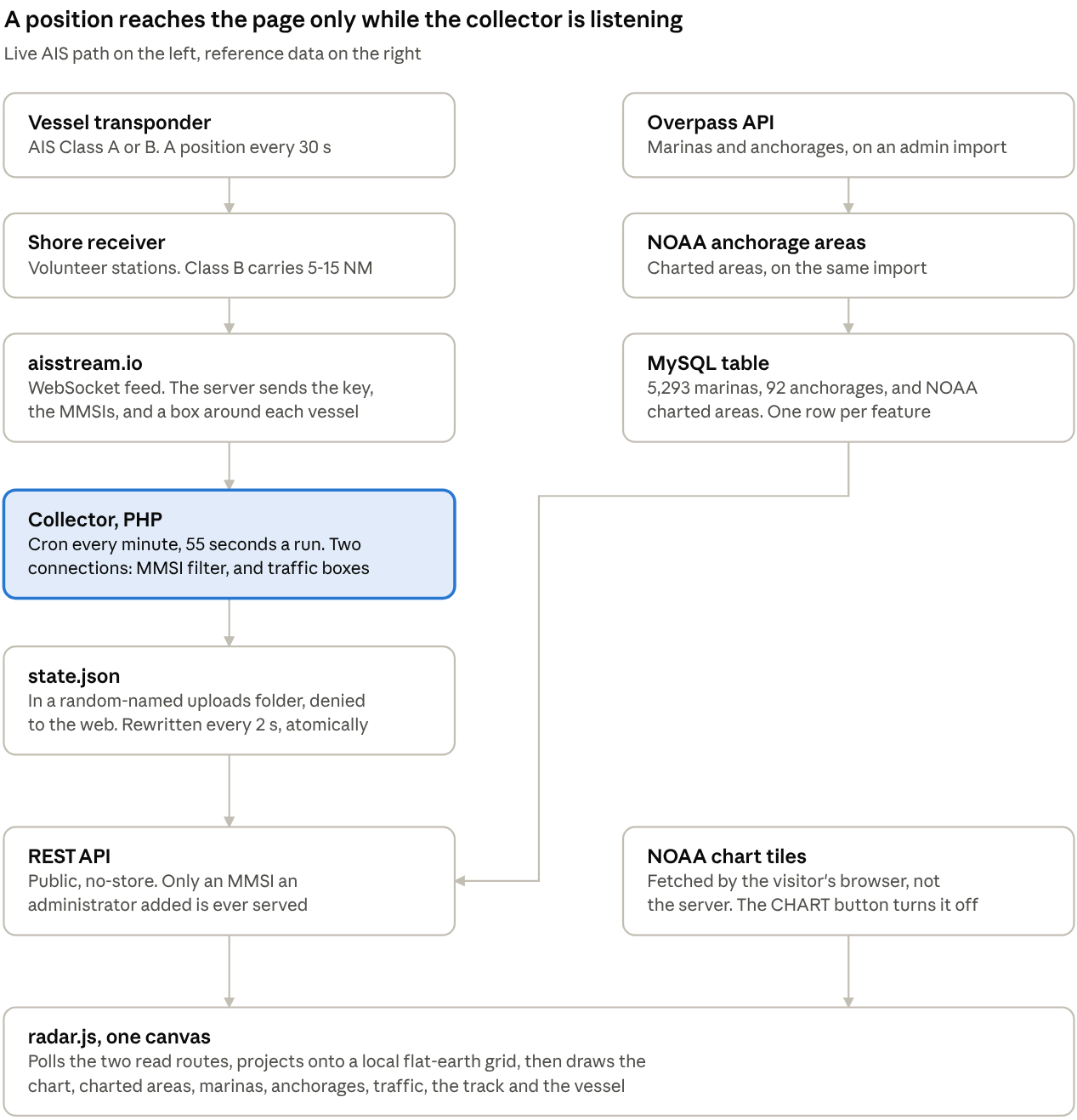
Layer	What it is
Server	PHP 7.4 or later, typed properties and strict comparisons throughout
PHP extensions	`openssl` for the TLS socket, `zlib` for `inflate_init()`
Sockets	`stream_socket_client()`, `stream_select()`, non-blocking reads
Platform	WordPress 6.0 or later
Database	MySQL / MariaDB, one table created with `dbDelta()`

Layer	What it is
Live state	A JSON file under `wp-content/uploads`, not the database
Browser	Vanilla ES2020, no framework, no bundler, no npm
Drawing	One HTML `<canvas>` element, 2D context
Styling	Hand-written CSS, custom properties, `dialog` elements
Build step	None
Third-party PHP	None. No Composer, no vendor folder
Third-party JS	None. No library is loaded from any CDN

The WebSocket client is written from scratch against RFC 6455 rather than pulled in as a dependency, because WordPress hosting rarely allows Composer and a plugin that needs a vendor folder cannot be installed from the Plugins screen.

WordPress APIs used: the Settings API, the REST API (`register_rest_route`), the Shortcode API, `wp_enqueue_script` / `wp_enqueue_style`, the options and transients APIs, `$wpdb` with `dbDelta()`, `wp_mail()`, the HTTP API (`wp_remote_post`, `wp_remote_get`), `admin-post.php` and `admin-ajax.php` actions, nonces and capability checks, the translation functions with a shipped `.pot` file, `register_activation_hook` and `uninstall.php`.

How it works



architecture and data flow · two paths into one page

The live path runs top to bottom on the left and is the one that can fail quietly: nothing reaches the page unless the collector is running. The reference data on the right is imported once by an administrator and then sat in the database. Chart tiles are the only outside request a visitor's own browser makes.

The WebSocket client

`PCAISR_WebSocket` is a hand-written RFC 6455 client in 286 lines, with no dependency beyond `openssl` and `zlib`. It exists because AISStream only publishes a WebSocket feed, and PHP has no WebSocket client of its own.

What it implements:

- **Transport.** `stream_socket_client()` to `ssl://` for `wss://` or plain TCP for `ws://`, with SNI and peer verification left at PHP's defaults. The socket is switched to non-blocking after the handshake so a read never stalls the run.
- **Handshake.** `GET` with `Upgrade: websocket`, `Connection: Upgrade`, a 16-byte random `Sec-WebSocket-Key` from `random_bytes()`, and `Sec-WebSocket-Version: 13`. The server's `Sec-WebSocket-Accept` is recomputed from the key and the RFC's magic GUID and compared; a mismatch throws rather than continuing.
- **Compression.** `permessage-deflate` with `client_max_window_bits` is offered, but only honoured if `inflate_init()` exists on the host. Receive-side inflation runs with context takeover across messages, which is what AISStream negotiates.
- **Framing.** Complete frame parsing: the 7-bit, 16-bit and 64-bit payload lengths, continuation frames reassembled into one message, and client-to-server frames masked as the RFC requires.
- **Control frames.** Ping is answered with pong carrying the same payload. An unsolicited pong is accepted. A close frame is read for its status code and reason, both kept for the status display, and answered before the socket is dropped.
- **End of stream.** `feof()` after a read marks the connection closed with a reason, so the collector can distinguish a server close from an idle connection.

`poll()` performs up to 256 non-blocking reads of 64 KB, appends to an internal buffer, parses out whatever complete messages are present, and returns them as an array of strings. It never blocks and never throws for an empty read, so the collector's loop stays predictable.

The handshake bytes are deliberately frozen. Changing header order or casing has broken the connection with this service before, so 1.3.x reproduces the 1.2.0 request exactly.

The collector

`PCAISR_Collector::run($seconds, $mode)` is the only thing that talks to the feed. It holds two WebSocket connections, because AISStream allows three per key and 200 MMSIs per filter.

The target connection subscribes to the whole world with the tracked MMSIs as a filter:

```
{
  "APIKey": "...",
  "BoundingBoxes": [[[-90, -180], [90, 180]]],
  "FilterMessageTypes": ["PositionReport", "StandardClassBPositionReport",
"ExtendedClassBPositionReport", "ShipStaticData", "StaticDataReport",
"AidsToNavigationReport", "StandardSearchAndRescueAircraftReport",
"LongRangeAisBroadcastMessage"],
  "FiltersShipMMSI": ["367778730"]
}
```

It re-sends the subscription on the same socket when the vessel list changes, throttled to one change every two seconds.

The traffic connection subscribes to bounding boxes instead of MMSIs, one box per tracked vessel, sized by the traffic radius setting: the latitude span is `radius / 60` degrees and the longitude span is `radius / (60 × cos(latitude))`, floored at a cosine of 0.05 so the box stays finite near the poles. It only opens once a vessel has a stored position, since there is no centre before that, and it re-subscribes when a vessel gains its first fix, is removed, or moves more than 2 NM — never more often than once every five seconds.

Message handling. Each message is matched on its type and only the fields that type actually carries are written:

Message type	What is taken
`PositionReport`	Position, SOG, COG, true heading, navigational status, rate of turn, Class A
`StandardClassBPositionReport`	Position, SOG, COG, true heading, Class B
`ExtendedClassBPositionReport`	The above plus name, ship type, hull dimensions

Message type	What is taken
`ShipStaticData`	Name, call sign, IMO number, ship type, dimensions, maximum static draught, destination, ETA
`StaticDataReport`	Part A gives the name; Part B gives ship type, call sign, transponder vendor, dimensions
`AidsToNavigationReport`	Position, name with extension, aid type, whether the aid is virtual
`StandardSearchAndRescueAircraftReport`	Position and altitude
`LongRangeAisBroadcastMessage`	Position and navigational status

Positions are rejected if they are not numeric, out of range, exactly 0,0, or older than the position already stored. Speed over ground is discarded at the AIS not-available value of 102.2 knots, course at 360, heading outside 0–359. Hull length and beam are reconstructed from the A+B and C+D dimension offsets. Names and call signs are stripped of the `@` padding that AIS uses.

Resilience. Each connection has its own retry clock with exponential backoff starting at two seconds. A throw on either connection is logged to the feed status, the socket is dropped, and the other connection keeps running. The state file is flushed every two seconds, so a run that is killed mid-way loses at most two seconds of data.

Mutual exclusion. Every entry point — cron, WP-CLI, the REST collector URL, the page-view fallback — takes an exclusive `flock()` on `collector.lock` before starting and releases it in a `finally`. A second collector exits immediately rather than queueing.

Where data lives

Live positions are kept in a file, not the database, because the collector rewrites them every two seconds and that is not a load any site's options table should carry.

The state file. `wp-content/uploads/pcaisr-<random>/state.json`. The folder name is generated once on activation and stored in the `pcaisr_data_dir` option, so the path is not guessable. An `.htaccess` carrying `Require all denied` and `Deny from all` plus an empty `index.php` are written into it. On nginx the deny rule has to be added by hand; the settings page prints the path. Writes go to a PID-suffixed temporary file and are moved into place with `rename()`, which is atomic on the same filesystem, so a reader never sees

half a file. A failed write sets the `pcaisr_io_error` option, which the settings page shows in red — without it, a folder the server cannot write to looks exactly like a feed with no data.

The schema is versioned; anything that is not version 2 is discarded and rebuilt.

```
{
  "version": 2,
  "updated": 1759540000,
  "vessels": {
    "367778730": {
      "mmsi": "367778730", "name": "CALYPSO BREEZE", "class": "B",
      "lat": 27.6160, "lon": -82.7420, "posTime": 1759539880, "posAcc": true,
      "sog": 6.4, "cog": 143.0, "hdg": 141, "nav": 0, "rot": 0,
      "type": 37, "callsign": "WDF1234", "imo": 0,
      "length": 12, "beam": 4, "draught": 1.5,
      "destination": "SARASOTA", "eta": "10-04 18:30 UTC",
      "staticTime": 1759539600
    }
  },
  "tracks": { "367778730": [[1759539880, 27.6160, -82.7420, 6.4]] },
  "feed": {
    "heartbeat": 1759540000, "mode": "cron", "lastMessage": 1759539880,
    "connected": { "target": true, "traffic": true },
    "lastError": "", "lastErrorAt": 0, "messages": 18422
  }
}
```

Tracks. Position history is kept only for tracked vessels, capped at 720 points each. A point is skipped if it arrives within 30 seconds of the last one and the vessel has moved less than 0.03 NM, so a boat at anchor does not fill the track with noise.

The marina table. One table, `{prefix}pcaisr_marinas`, created and migrated with `dbDelta()` against a schema version in the `pcaisr_db_ver` option, because a plugin updated by upload never re-runs activation.

Column	Type	Holds
`id`	`bigint unsigned` auto	Primary key
`osm_type`	`char(1)`	`n`, `w` or `r` for node, way or relation
`osm_id`	`bigint unsigned`	The source feature's own id; unique with `osm_type`

Column	Type	Holds
`name`	`varchar(200)`	Display name
`lat`, `lon`	`double`	Centre point, indexed together
`kind`	`varchar(12)`	`marina`, `anchorage` or `charted`, indexed
`source`	`varchar(8)`	`osm` or `noaa`
`geom`	`mediumtext`	Outer ring of an area feature, as JSON, where there is one
`tags`	`text`	The source feature's attributes, as JSON
`run_id`	`bigint unsigned`	Which import run last saw the row, so a partial import removes nothing
`updated`	`datetime`	When the row was last written

Current contents on the production site: 5,293 marinas and 92 OpenStreetMap anchorages, plus NOAA charted anchorage areas.

Options. `pcaisr_settings`, `pcaisr_vessels` (MMSI to display name), `pcaisr_secret` (64 characters, the HMAC key for share tokens and hashed IP addresses), `pcaisr_cron_key` (32 characters, authorises the collector URL), `pcaisr_data_dir`, `pcaisr_db_ver`, `pcaisr_marina_meta`, `pcaisr_marina_run`, `pcaisr_io_error`. All are autoload-off except the schema version. Admin notices pass through a 60-second transient keyed by user id.

`uninstall.php` removes the data folder, drops the table, and deletes every option listed above.

REST API

Namespace `presscare-ais/v1`, so the base is `/wp-json/presscare-ais/v1/`. The namespace keeps its original name; renaming it would break every page already published against it.

Every route answers `permission_callback: '__return_true'` and is therefore public. The access control is elsewhere and is deliberate: only an MMSI an administrator has added to `pcaisr_vessels` is ever served, and `mmsi` is normalised to 7–9 digits before it is

looked up. Anything else returns 404 `pcaisr_unknown_vessel`. There is no route that lists the vessels on a site, no route that writes vessel data, and no route that returns a position for a vessel the site does not display.

All responses carry `Cache-Control: no-store, no-cache, must-revalidate, max-age=0` and `X-Content-Type-Options: nosniff`.

Route	Method	Parameters
`/state`	GET	`mmsi` required, string, `sanitize_text_field`
`/marinas`	GET	`mmsi` required, string, `sanitize_text_field`
`/collect`	GET	`key` required, string, `sanitize_text_field`
`/share`	POST	`mmsi`, `token`, `method`, `to`, `post_id`, `website`

GET /state

The only route the radar polls. It returns the vessel, its track, the traffic around it, the feed's health, and a fresh share token.

```
{
  "ok": true,
  "now": 1759540000,
  "mmsi": "367778730",
  "name": "CALYPSO BREEZE",
  "radius": 24,
  "target": { "mmsi": "367778730", "lat": 27.616, "lon": -82.742, "posTime":
1759539880,
              "sog": 6.4, "cog": 143.0, "hdg": 141, "nav": 0, "class": "B",
"posAcc": true,
              "name": "CALYPSO BREEZE", "type": 37, "callsign": "WDF1234",
              "length": 12, "beam": 4, "draught": 1.5,
              "destination": "SARASOTA", "eta": "10-04 18:30 UTC",
"staticTime": 1759539600 },
  "track": [[1759539880, 27.616, -82.742, 6.4]],
  "vessels": [ { "mmsi": "338103756", "lat": 27.64, "lon": -82.70, "posTime":
1759539950, "sog": 11.2, "cog": 22.0, "type": 70 } ],
  "sms": false,
  "token": "1759540000.9f3c...",
  "feed": { "status": "live", "mode": "cron", "heartbeat": 1759540000,
            "lastMessage": 1759539880, "error": "" }
}
```

`target` is `null` until the vessel has been heard at all, and is present without `lat` when only identity data has arrived — a `ShipStaticData` message carries a name, call sign, type and destination but no coordinates.

`vessels` holds the surrounding traffic and is empty whenever the tracked vessel has no fix, since there is no centre to measure from. A station is included only if it has a position, that position is newer than the `max_age_min` setting, and it lies within 1.5 × the traffic radius.

`feed.status` is one of:

Status	Meaning
`nokey`	No API key saved
`error`	The last error is newer than the last message and under three minutes old
`live`	An AIS message arrived in the last two minutes

Status	Meaning
`listening`	A collector ran in the last two minutes but no message arrived
`idle`	No collector has run for two minutes

`feed.error` is populated only for `nokey` and `error`; it is an empty string otherwise, so the page never shows a stale error.

Page-view fallback. If the `web_fallback` setting is on, the key is saved, and the collector's heartbeat is more than 75 seconds old, this route also runs a short collection. Where `fastcgi_finish_request()` exists the response is flushed first and the listening happens after the visitor has their page; otherwise it runs inline and the state is reloaded before responding. Either way it is bounded by `web_seconds`, clamped to 5–25, and guarded by the collector lock.

GET /marinas

Marinas, anchorages and charted anchorage areas within the traffic radius of the vessel's last position. Returns `center: null` and an empty list when there is no fix.

```
{
  "ok": true,
  "center": { "lat": 27.616, "lon": -82.742 },
  "marinas": [
    { "id": "n123456789", "name": "Marina Jack", "lat": 27.33, "lon": -82.55,
      "kind": "marina", "src": "osm", "ring": null,
      "tags": { "phone": "+1-941-955-9488", "seamark:radio_station:channel":
"16", "fuel:diesel": "yes" },
      "_d": 17.8 }
  ]
}
```

`kind` is `marina` or `anchorage`; `src` is `osm` or `noaa`; `ring` carries the outer boundary of an area feature as an array of `[lat, lon]` pairs, or `null` for a point; `_d` is the distance in nautical miles. An area whose centre is out of range is still included if any point on its boundary is inside, which is what makes large charted anchorages appear as you approach them.

GET /collect

Runs one collection, for an external cron job that can only make an HTTP request. `key` must equal the `pcaisr_cron_key` option, compared with `hash_equals()`. The key is generated on activation and the full URL is printed on the settings page.

Returns `403` for a wrong key, `400` when no API key is saved, `{"ok": true, "skipped": "already running"}` when the lock is held, and otherwise:

```
{ "ok": true, "seconds": 55, "messages": 18422, "lastMessage": 1759539880 }
```

Run length comes from `run_seconds`, clamped to 5–300.

POST /share

Sends the page link, with the vessel's latest fix, by email or by text message.

Fields: `mmsi`, `token` from the last `/state` response, `method` of `email` or `sms`, `to` holding addresses or a phone number, `post_id` for building the link, and `website` as a honeypot — any value in it returns success without sending anything.

The token is an HMAC over a timestamp keyed with `pcaisr_secret`; an expired one returns `403` with "This page has been open too long." Email addresses are split on commas, semicolons and whitespace, de-duplicated, validated with `is_email()`, and capped by `max_recipients`. Phone numbers are reduced to digits and a leading `+` and must match 10–15 digits. Both paths are rate limited before anything is sent.

Mail goes through `wp_mail()`, so an SMTP plugin on the site is used. Text messages go through Textbelt and only when a key is saved; without one the button is not rendered and the route returns `400`.

Status codes: `200` on success, `400` for bad input, `403` for a stale token, `429` when a rate limit is hit, `500` when the mail server refuses every address, `502` when Textbelt fails.

Shortcode API

```
[ais_radar mmsi="367778730" range="3" mode="north" share="yes" chart="yes"]
```

Attribute	Accepted	Default	Effect
`mmsi`	7–9 digits	none	Which vessel to display. Must already be in the admin's vessel list, or the shortcode renders a notice instead of a radar
`range`	`0.25`, `0.5`, `0.75`, `1.5`, `3`, `6`, `12`, `24`	`3`	Opening range in nautical miles. An unrecognised value falls back to `3`
`mode`	`north`, `head`, `course`	`north`	North Up, Head Up or Course Up orientation
`share`	`yes`, `no`	`yes`	`no` hides the email, text and copy tools
`chart`	`yes`, `no`	`yes`	`no` leaves the NOAA chart off, so the visitor's browser never contacts NOAA

The MMSI is normalised and checked against the stored vessel list at render time, so a shortcode cannot be used to display an arbitrary vessel. Assets are enqueued only on a page that actually contains the shortcode. Range is stored and compared by its value rather than its index, so adding a range in a future version cannot shift a page's saved setting.

Other interfaces

Command line

```
wp ais-radar collect [--seconds=<n>] [--verbose]
```

Runs one collection. `--seconds=0` runs until stopped, which is how it would be run under a process supervisor; omitting it uses the `run_seconds` setting. `--verbose` prints connection events with timestamps. It takes the collector lock and exits with a message if another collector is running, and reports the running message count and last error when it finishes. Registered `@when after_wp_load`.

Standalone cron entry

`collect.php` in the plugin folder bootstraps WordPress and runs one collection. This is the path most shared hosting can actually use:

```
* * * * * /usr/bin/php /path/to/wp-content/plugins/following-seas/collect.php
>/dev/null 2>&1
```

From 1.3.7 the settings page builds this line from `PCAISR_DIR`, so it always names the folder the plugin is really installed in.

Admin actions

All on `admin-post.php`, all requiring `manage_options` and a matching nonce.

Action	Does
`pcaisr_add_vessel`	Adds an MMSI and display name
`pcaisr_remove_vessel`	Removes one
`pcaisr_test`	15-second capture test, reporting messages received and which connections came up
`pcaisr_probe`	Feed probe: subscribes to the Dover and Singapore straits for 15 seconds to prove the key and the network work, independently of any tracked vessel
`pcaisr_vprobe`	Vessel probe: 25 seconds on the tracked MMSIs only, printing the exact filter sent and every MMSI and message type heard. Accepts a single `mmsi` to probe one vessel. Held to 25 seconds because hosts commonly cut a request at 60

AJAX endpoints

The marina import runs tile by tile from the browser so no single request has to finish the whole job: `pcaisr_marina_start`, `pcaisr_marina_tile`, `pcaisr_marina_finish`, on `admin-ajax.php`, all nonce-checked and capability-checked. The import area is divided into 3° × 3° tiles and each tile is one request.

Filters

Filter	Purpose
`pcaisr_ws_url`	Override the AISStream endpoint. Intended for testing against a local stub
`pcaisr_overpass_url`	Override the Overpass API endpoint, for a self-hosted or mirror instance
`pcaisr_noaa_anchorages_url`	Override the NOAA anchorage-areas endpoint
`pcaisr_chart_tiles`	Replace the chart tile source: its URL template, zoom offset, and minimum and maximum zoom

Outside services

Five, four of them contacted by the server and one by the visitor's browser. Nothing is contacted at all until an administrator sets the plugin up.

aisstream.io

`wss://stream.aisstream.io/v0/stream`, the live AIS positions. The server sends the API key, the MMSI numbers the administrator added, and a map area around each vessel's last position. No visitor data is sent. The key is free: sign in with GitHub, create a key. Three connections per key, 200 MMSIs per filter. No paid tier exists and no payment is involved at any point. Terms · Privacy

OpenStreetMap Overpass API

`https://overpass-api.de/api/interpreter`, the marina and anchorage lists. Contacted only while an administrator runs the import. The request carries the map area to fetch and, in the user agent header, the plugin version and the site's address, as the Overpass usage policy asks. The query collects `leisure=marina`, `seamark:harbour:category=marina`, `seamark:type=anchorage` and `anchor_berth`, as nodes, ways and relations. Data is © OpenStreetMap contributors under the Open Database License. API policy · Copyright

NOAA Office of Coast Survey — chart tiles

`https://gis.charttools.noaa.gov/arcgis/rest/services/MarineChart_Services/NOAACharts/MapServer/tile/{l}/{y}/{x}`, the Chart Display Service, requested by the **visitor's browser**. NOAA therefore receives the visitor's IP address and which tiles were viewed. This is the only visitor-facing outside request the plugin makes, and it is the reason the CHART button and the `chart="no"` shortcode option exist: with the chart off, no request is made. The service's zoom level 0 corresponds to web zoom 2, so the plugin applies an offset of 2 and requests zoom 3 to 18. Tiles are redrawn in a night palette client-side. Public domain.

NOAA Office of Coast Survey — charted anchorage areas

`https://gis.charttools.noaa.gov/arcgis/rest/services/NavigationChartData/Anchorage_Areas/MapServer/0/query`, the ENC feature `ACHARE`. Fetched by the server during the import, carrying only the map area, with `maxAllowableOffset=0.0005` to keep the geometry manageable and paging 500 features at a time. Fields used: `OBJECTID`, `OBJL`, `THEMELAYER`, `INFORM`, `OBJNAM`. A failed request removes nothing already stored. Public domain. Chart information · Privacy

Textbelt

`https://textbelt.com/text`, optional, used only if an administrator saves a Textbelt key. When a visitor chooses to send the page link as a text message, the number they typed and the message are sent to Textbelt to deliver. With no key saved, the button is not rendered and nothing is sent. Textbelt's shared free key allows one message a day and strips links, which is why the paid key is the only useful configuration. Terms and privacy

Email

Not an outside service as such: email goes through `wp_mail()`, so it uses whatever the site already uses. If an SMTP plugin is installed, that is the path.

All five are disclosed in `readme.txt` under `== External services ==`, which WordPress.org requires.

The browser client

`assets/radar.js`, one file, no dependencies. It polls `/state`, polls `/marinas` less often, and draws everything onto a single canvas.

Projection. A local flat-earth projection centred on the tracked vessel: north–south is `(lat − lat0) × 60` nautical miles, east–west is `(lon − lon0) × 60 × cos(lat0)`. Over a 24 NM radius the error is far below what AIS resolution can express, and it keeps the maths cheap enough to redraw on every animation frame.

Display modes. North Up, Head Up and Course Up. In the two rotating modes the whole scene — including the chart tiles — is rotated about the vessel, so the chart stays aligned with the picture.

Ranges. 0.25, 0.5, 0.75, 1.5, 3, 6, 12 and 24 NM, with range rings and bearing graticules drawn to suit.

Layers drawn, bottom to top: NOAA chart tiles recoloured for a night palette; charted anchorage areas as dashed polygons; range rings and bearing scale; marinas as an M in a ring; anchorages as a solid anchor; AIS traffic coloured by vessel type; the tracked vessel's own track; the vessel itself with a true-motion vector; labels.

Labels. Drawn in priority order — the selected target first, then the nearest — with a collision test against everything already placed, so a crowded anchorage does not turn into illegible overlapping text. A label that cannot be placed is dropped rather than drawn on top of another.

Interaction. Click a target, marina or anchorage for a details dialog; traffic details include closest point of approach. A cursor readout gives range, bearing and position under the pointer. Toggles for traffic, marinas, anchorages, the chart, and the chart key. The key dialog explains the buoys, lights, depth figures and area boundaries that NOAA draws, and links to U.S. Chart No. 1.

Formatting. Position in degrees and decimal minutes and in decimal degrees; speed in knots, mph and km/h; course and heading; navigational status; and the age of the last fix.

Status. A feed chip reports what the collector is doing. From 1.3.7 it distinguishes a collector that is listening from one that has stopped, and says how long since it last ran.

All of it works at phone width: the layout has been checked down to 390 px with no horizontal overflow and no element hidden to make it fit.

Features

The vessel

- Position in degrees and decimal minutes and in decimal degrees
- Speed over ground in knots, mph and km/h

- Course over ground, true heading, navigational status, rate of turn
- Age of the last fix, and a note when the vessel has been quiet a long time
- Name, call sign, IMO number, ship type, length, beam, draught
- Destination and ETA, where the transponder sends them
- Track history, up to 720 points, with a noise filter for a vessel at rest

Traffic

- Every AIS station within the traffic radius, coloured by vessel type
- Details on click, including closest point of approach
- Aids to navigation, including virtual ones, and search-and-rescue aircraft
- Silent stations dropped after a configurable age

Marinas and anchorages

- Marinas from OpenStreetMap, drawn as an M in a ring
- Distance, bearing and estimated time of arrival at the vessel's current speed
- Whatever OpenStreetMap holds for each: phone, VHF channel, fuel, berths, website
- Anchorages from OpenStreetMap, drawn as a solid anchor, on their own layer
- NOAA charted anchorage areas, drawn as areas with their boundary and chart note
- Tabbed lists for traffic, marinas and anchorages

Charting

- NOAA nautical chart under the radar, recoloured for a night palette
- Rotates with Head Up and Course Up
- CHART toggle for the visitor, `chart="no"` for the page
- Plain-language chart key, linking to U.S. Chart No. 1

Sharing

- Send the page link by email, to several addresses at once
- Send by text message from the server, where a Textbelt key is saved
- Copy the link, or hand off to the device's own share menu
- Honeypot field, share tokens, and per-IP and per-day rate limits

Administration

- Its own top-level admin menu
- Vessels added and removed by MMSI, with a copyable shortcode per vessel

- Feed status: collector mode, last run, last message, message count, last error
- 15-second capture test
- Feed probe, which proves the key and network independently of any vessel
- Per-vessel 25-second probe, printing the exact filter sent and everything heard
- Data-folder write test
- Cron warning on every admin screen when nothing is running the collector
- Marina and anchorage import, tile by tile, with progress
- Fully translatable, with a 229-string `.pot` file included

Security

What a visitor can reach. Four public REST routes. Three of them only ever return data about an MMSI an administrator has explicitly added, and the fourth needs a 32-character key. No route lists the vessels on a site, and none writes vessel data.

Administration. Every admin action checks `current_user_can('manage_options')` and verifies a nonce before doing anything. Every AJAX endpoint does the same. Settings go through one `sanitize_callback` that rebuilds the whole array from a fixed set of keys with every numeric value clamped to a range, so an unexpected field cannot reach the option.

Output. Everything echoed in the admin and in the page markup is escaped at the point of output — `esc_html`, `esc_attr`, `esc_url`, and `wp_kses` with an explicit tag list where a link or a `<code>` element is wanted inside a translated string.

Database. One table. Every query is `$wpdb->prepare()`d; the only interpolation is the table name built from `$wpdb->prefix`, annotated where the coding standard requires it.

The data folder. Named with a random suffix generated on activation, closed off with `.htaccess` and an empty `index.php`, and its path is never printed to a visitor. The write test on the settings page exists because a folder the server cannot write to is otherwise indistinguishable from a feed with no data.

Share tokens. An HMAC-SHA256 over a timestamp keyed with a 64-character secret generated on activation. Expired tokens are rejected. The form also carries a honeypot field, which returns success and sends nothing.

Rate limiting. Per-IP-per-hour and per-day caps on sends, held in a `flock()`ed JSON file. The visitor's IP address is never stored: it is HMAC'd with the site secret and only the hash is kept.

What is never stored. No visitor identity, no IP addresses in clear, no analytics, no cookies set by the plugin. Nothing about a visitor is sent to any outside service. The one outside request a visitor's browser makes is for NOAA chart tiles, which the visitor can switch off and the administrator can disable for a page.

Uninstall. `uninstall.php` removes the data folder and its contents, drops the table, and deletes every option. Nothing is left behind.

Requirements and setup

Needs	Why
WordPress 6.0+	REST and Settings APIs as used
PHP 7.4+	Typed properties
`openssl`	The TLS socket to AISStream
`zlib` with `inflate_init()`	`permessage-deflate`. Without it the connection still works, uncompressed
`stream_socket_client()` and `stream_select()` not disabled	Some shared hosts disable them; the settings page tests for this
A writable uploads folder	The state file
A cron job every minute	Continuous collection
A free aisstream.io key	The feed

Setup, in order:

1. Install and activate, by upload or by copying the folder into `wp-content/plugins/`.
2. Open **Offing** in the admin menu and paste in the AISStream key.
3. Add each vessel by MMSI. Only administrators can do this, and only a listed vessel can ever be displayed.
4. Copy the vessel's shortcode into a page, for example `[ais_radar mmsi="367778730"]`.
5. Add the cron job. The settings page prints the exact line for the folder the plugin is installed in.
6. Run the 15-second test to confirm the feed connects.

7. Optional: set the import area, save, then run the marina and anchorage import and leave the page open until it finishes.

On the cron job. It is the part that matters most and the part most likely to be wrong. AIS positions arrive every 30 seconds when a vessel is under way and every 3 minutes when it is at rest, so a vessel is only seen if something is listening nearly all the time. A collector that runs for 12 seconds every 75 seconds catches roughly one report in fifteen; one that runs 55 seconds out of every 60 catches nearly all of them. The page-view fallback exists so a site with no cron still shows something, not as a substitute.

This is what broke on the production site after the 1.2.1 rename: the folder changed from `presscare-ais-radar` to `following-seas`, the existing cron line pointed at a path that no longer existed, and nothing reported it — cron ran nothing, silently, for an hour of troubleshooting. 1.3.7 adds the warning that would have caught it in seconds.

What the data can and cannot do

The limits here are in AIS itself, not in the plugin, and they set what the display can honestly promise.

Coverage is terrestrial. AISStream aggregates land-based receivers. There is no satellite AIS in the free feed. A vessel is visible only while a shore station can hear it.

Range depends on the transponder. Class B, which almost all pleasure craft carry, transmits at 2 W and typically reaches a shore receiver 5 to 15 NM away. Class A transmits at 12.5 W and reaches further. Beyond that the vessel simply goes quiet, and the page shows the last reported position with its age. That is normal offshore and is not a fault.

Reporting intervals. Class B sends a position every 30 seconds under way and every 3 minutes at rest. Static data — name, call sign, type, destination — comes roughly every 6 minutes. So a vessel can be heard and still have no position for several minutes, which is why the page now says which of the two has happened.

Fields are optional. Navigational status is not sent by Class B at all. Destination and ETA are typed in by the crew and are often blank, stale or wrong. Heading is frequently absent where there is no compass interface. The display shows what arrived and does not infer the rest.

Position accuracy. AIS carries a one-bit accuracy flag, not an error figure. A "high accuracy" flag means under 10 metres; otherwise under 100 metres. The plugin reports the flag and does not pretend to more precision.

This is not a navigation aid. It is a display for following a vessel from shore. It must not be used for navigation, collision avoidance or safety of life at sea. Positions can be minutes old, absent or wrong, charts can be out of date, and nothing here is certified for any of those purposes.

Version history

Current version: **1.3.7**.

Version	What changed
1.3.7	Cron warning on every admin screen, with the correct line built from the install path. Feed chip distinguishes a stopped collector from a starting one. A vessel heard with identity data but no position now says so. `Update on page views` can only be changed by the settings form
1.3.6	Vessel probe cut from 60 seconds to 25, since hosts that cut a request at 60 returned a 504
1.3.5	A probe button on each vessel row, testing that one MMSI
1.3.4	Vessel probe: asks the feed for your MMSIs only and reports the exact filter sent and every MMSI heard
1.3.3	Data-folder write test on the settings page. Before this, a folder the server could not write to looked exactly like a feed with no data
1.3.2	`collect.php` restored so existing cron lines keep working. Page-view updates on by default again. Connection handshake restored byte for byte to 1.2.0
1.3.1	NOAA charted anchorage areas, drawn as areas with their boundary, name and chart note. A failed NOAA request no longer removes anything stored

Version	What changed
1.3.0	Anchorages from OpenStreetMap with their own toggle and tab. Marinas redrawn as an M in a ring. 0.25 NM range, with range stored by value. Chart KEY dialog. `chart="no"` shortcode option
1.2.1	Renamed to Offing - Following Seas with its own top-level admin menu. All text made translatable, with a `.pot` file. **This release changed the plugin folder to `following-seas`, which invalidated existing cron lines**
1.2.0	NOAA chart background with a CHART toggle, rotating with Head Up and Course Up. Marina import also takes `seamark:harbour:category=marina`, de-duplicated. Label collision avoidance
1.1.0	Marinas from OpenStreetMap: Overpass import, radar icons, a Marinas tab, a details dialog
1.0.0	First release

Still open for WordPress.org

- `Contributors:` in `readme.txt` must be the real WordPress.org username
- `Tested up to:` must name the version actually tested
- Four screenshots as `screenshot-1.png` to `screenshot-4.png` in the SVN assets folder
- A 1544 × 500 banner
- Run the official Plugin Check plugin and clear what it reports

Noted for later versions

- A plain-English status line, place names instead of coordinates, and reframed wording for a quiet transponder — the interface changes for families watching from shore
- A map as the default view with the radar as a toggle
- Per-vessel home position, so the display has a centre before the first fix
- NOAA feature identify on chart clicks

- Visitor contributions, and vessel owner logins