# DroneWatch AI

Multi-Sensor Early-Warning · Software Intelligence Layer for Drone Detection

TECHNICAL BRIEF | UNCLASSIFIED / FOR DISCUSSION | DEMONSTRATOR

Prepared by Devaland Marketing SRL (Romania) · 2026 · office@devaland.com

**Honesty statement.** DroneWatch AI is a working **concept demonstrator**. Drone-threat signals are simulated; the authorised air-traffic registry is a live real ADS-B feed. It is not an operational system, it does not detect real drones, and no performance metric in this document is a field-validated claim. This brief is for technical discussion and partnership only.

## 1. Executive summary

Commercial drones now threaten airports, critical infrastructure, public events and borders. Traditional counter-drone detection relies on expensive, specialised hardware, which makes wide-area and layered coverage hard to afford and hard to scale. **DroneWatch AI is a software intelligence layer** that fuses many weak, distributed signals (radio-frequency, acoustic, telemetry, public registries and human reports), applies AI correlation, and raises a graded, explainable alert only when independent sources agree. It makes existing and low-cost sensors smarter rather than requiring new hardware, and it pushes a scored track into the operator's existing command system.

## 2. The problem

- Rapid proliferation of low-cost UAS creates a persistent, low-altitude threat to protected sites.
- Dedicated counter-UAS sensors are costly, so coverage is sparse and fragmented.
- Single sensors are noisy: RF alone, acoustic alone or radar alone each generate false alarms.
- Operators are flooded with unfused, unexplained detections and cannot act on them with confidence.

## 3. Concept and value

DroneWatch does not replace sensors or effectors. It is the **fusion and AI layer** that sits above heterogeneous sensing infrastructure. It aggregates weak signals no single sensor can act on, correlates them across sources, cross-references a registry of authorised traffic, scores the threat, and hands a cued, explainable track to the customer's command and control. The design principle throughout is **false-alarm suppression through multi-source agreement**.

## 4. System architecture

A modular pipeline of five stages, each independently replaceable:

| Stage | Function |
|---|---|
| 1. Ingestion | Normalises inputs from each source into a common event schema. |
| 2. Aggregation | Merges all sources into one time-ordered fused event stream per zone. |

| | |
|---|---|
| 3. Correlation / detection | Two-tier engine: deterministic rules first, AI narrative second. |
| 4. Alerting | Raises a single, de-duplicated alert on escalation and pushes it onward. |
| 5. Monitoring | Live operator console: radar scope, tactical map, feed and alert log. |

## 5. Detection logic and false-alarm discipline

Escalation is deliberately conservative. Only **unexplained** indicators raise the threat:

- A single sensor blip never alerts (it holds at LOW).
- A registered, authorised flight, even a strong one, never raises the threat.
- **HIGH requires at least two independent sensors to agree AND an unknown identity**, that is a contact with no matching authorised flight in the registry.
- Each alert fires once on the upward transition, latched, so it never flaps.
- The AI layer may only enrich or soften the assessment, never raise a threat the sensors do not support.

This multi-source-agreement rule is the core of the value proposition: fusion exists to reduce false alarms. The real-world false-alarm rate can only be proven with field data, which is the next engineering phase.

## 6. AI correlation layer

The correlation stage is two-tier, and the split is what makes the system suitable for a disconnected environment. **Tier 1** is a deterministic rule engine. It computes the threat level and confidence score from the fused window using explicit rules, with no model of any kind on the decision path. A HIGH assessment requires at least two independent indicators and a contact with no corresponding authorised flight; a registered, authorised flight is benign context and never raises the level.

**Tier 2** rewrites that assessment into one factual operator sentence, so the operator is not reading raw signal data. It is optional by construction: the model receives the already-computed threat level as a fixed value, is instructed not to change it or invent capability, runs only on escalation, and its output is cached. If it is unavailable, the deterministic rationale stands and the system is otherwise unaffected.

**The Tier 2 model runs locally.** As of 3 August 2026 it is an open-weight model served on the same host and reached over the loopback interface. There is no external API call, no API key, and no data leaves the machine. For accredited or disconnected deployments this is a requirement, not a preference.

**Small local models are not reliably obedient, so their output is verified rather than trusted.** In a 25-run test across all threat levels the model never altered the threat level, which by construction it cannot. However 7 of those 25 narratives were unfit for an operator display: some omitted the threat level and drifted into speculation, others described detector confidence as "high" alongside a LOW assessment. A validation gate was therefore added. A generated sentence is accepted only if it names the level the rules computed, and no other level; anything else is discarded and the deterministic rationale is shown instead. Repeating the test after this change returned 30 conforming outputs from 30, with no non-conforming output reaching the operator. The same verify-the-output discipline is published as open source at github.com/MariusGithub13/ro-cited-answers.

## 7. Inputs: simulated now, real on integration

### Simulated in the demonstrator

- RF control/video-link detection
- Acoustic rotor signature
- Drone telemetry entry
- Human observer reports

### Live real today

- Authorised air-traffic registry via a live ADS-B feed (real aircraft, real positions)
- Plotted in real time on the tactical map

Each simulated source is replaced, one at a time, by a real sensor over standard messaging (MQTT or Kafka), keeping the fusion engine sensor-agnostic.

## 8. Integration with existing systems

Being usable means speaking the interfaces operators already run. DroneWatch ingests real sensors (RF direction-finding, acoustic arrays, radar, EO/IR cameras, Remote ID receivers, ADS-B) and integrates with existing command systems through recognised standards:

| Domain | Standards / interfaces |
|---|---|
| Identification | Remote ID (ASTM F3411), ADS-B |
| Air / drone traffic | ASTERIX (Eurocontrol), U-space / UTM |
| Military C2 | STANAG 4586 (UAS control), STANAG 4609 (EO/IR imagery), Link 16 / STANAG 5516 |
| Friendly forces | NFFI, to suppress alerts on own assets |

**Output** is a cued, scored track handed to the customer's C2 for a human to act on. DroneWatch stays strictly **defensive**: detection and early-warning only. Any mitigation is left to a separate, authorised, human-in-the-loop system.

## 9. Technology and deployment

Built on a mature, auditable stack: Python and FastAPI, containerised with Docker, real-time data pipelines, deterministic correlation rules with a locally hosted open-weight model for operator-facing explanation, and a browser-based operator console.

**Implemented today, and verifiable in the running demonstrator:** TLS in transit; **role-based access enforced server side**, where an observer can read the picture and run the scripted scenario but is refused sensor ingest, and the refusal itself is logged; an **append-only audit trail** of every non-polling console action, recording user, role, a stable visitor identifier, source address and outcome; and a Tier-2 explanation model that runs wholly on the host with no external API call.

**Intended for a site deployment and not yet built:** on-premise or air-gapped installation per site with a central fusion node, encryption at rest, and a persistence layer. Apart from the audit trail the demonstrator holds its state in memory. **Nothing here has been through a security accreditation and none is claimed**; the architecture is built so that such a process would be possible, which is a different statement.

The current demonstrator runs on independently owned and monitored infrastructure.

## 10. Development roadmap

| Phase | Content | Status |
|---|---|---|
| 1 Prototype | Fusion, correlation, alerting, live console, live ADS-B | Built |
| 2 Sensor integration | Real RF, acoustic, radar, EO/IR, Remote ID at one site | Next |
| 3 Advanced detection | ML models, behavioural analysis, false-positive reduction | Planned |
| 4 Operational C2 | Command interface, integration with security infrastructure | Planned |
| 5 Detection grid | Distributed, edge and cross-border coverage | Planned |

## 11. Partnership opportunity

DroneWatch provides the AI fusion and correlation layer that sensor and integrator companies often lack. The most efficient route to a fielded system is a partnership:

- **We bring:** the software intelligence layer, the AI correlation, the operator console and the integration engineering.
- **A partner brings:** real sensors, an accreditation path, and a customer channel.
- **Funding:** EU Defence Innovation Scheme (EUDIS) and European Defence Fund calls fund exactly this kind of dual-use software layer.
- **First step:** a pilot at one real site with one or two real sensor types plus the live feeds, to measure the real false-alarm rate.

## 12. Team

DroneWatch AI is built by a former **Army Engineer** (Military Technical Academy, Bucharest) who now builds and operates production AI systems on his own infrastructure. The perspective is defensive early-warning throughout, from someone who understands both the operational world and the software.

## 13. The demonstrator

A live, login-walled demonstrator with a one-click hostile-drone scenario, a radar scope, a live map of real air traffic, audible alerts and an explainable threat assessment is available for a guided walkthrough on request at **https://dronewatchai.eu**.