**Executive summary**

* The aggregated data shows **very low impression counts (zero)** alongside large numbers of requests and IDFAs ,this is the primary signal of problematic traffic (requests that never resulted in impressions).
* Traffic **spikes** (large increases in total\_requests / unique\_idfas in short windows) coincide with much higher IVT estimates ,apps that experienced earlier spikes were flagged earlier.
* **User-Agent diversity is extremely low** in the data snapshot (unique\_uas values are tiny, e.g., 17–26) while unique\_idfas are tens of thousands, that mismatch strongly suggests **UA reuse / device spoofing**.

Conclusion: apps were marked IVT earlier when they showed sudden traffic spikes combined with low UA diversity and zero impressions (i.e., many requests but no real ad renders). Apps that only later showed those patterns (spikes / zero impressions / UA reuse) got flagged later.

**Key evidence & patterns (from your uploaded files)**

* **Impressions per IDFA = 0** (many rows show impressions = 0, impressions\_per\_idfa = 0). This means requests are being generated but ads are not being rendered/delivered.
* **Very small unique\_uas** values (e.g., 17–26 across aggregates) despite unique\_idfas in thousands, implies many devices share the same UA string. This is abnormal for real traffic and is a strong spoofing indicator.
* **idfa\_ip\_ratio ≈ 1** in aggregated rows, suggests devices appear from many IPs rather than all coming from a single IP, which could imply distributed bot networks or mobile proxies (the ratio alone ≈1 is not sufficient to exonerate; must check IP geography / ASN).
* **IVT spikes correlate with request spikes**: days/hours with sudden jumps in total\_requests or unique\_idfas show much larger IVT values in the data you've provided. Example: 2025-09-12 had a huge total\_requests number and a high IVT estimate.

**Hypotheses why some apps were flagged IVT earlier vs later**

1. **Traffic spike timing** ,apps that experienced sudden large increases in requests (or very large hourly spikes) were flagged earlier because automated detectors weight abrupt volume anomalies heavily. (Supported by the hourly/daily numbers in the files.)
2. **UA-string reuse / low UA diversity** , apps where many IDFAs shared the same UA made fraud detectors treat the traffic as spoofed sooner.
3. **Impression mismatch (requests > impressions)**, when many requests produce no impressions (impressions = 0), detectors treat them as invalid; apps showing this earlier were flagged earlier.
4. **Staggered/fuzzy labeling by different rulesets or thresholds**, some apps may be monitored by slightly different IVT rules (or human review lag), so the same pattern could trigger at different times depending on threshold configuration. (Operational explanation ,check detection rule configs.)

**Recommended immediate actions (operational & forensic)**

1. **Forensic per-app drilldown** (priority): run a per-app analysis for these signals:
   * *requests / idfa* time series (hourly)
   * *impression vs request* ratio time series
   * *unique\_uas per hour* and distribution of UA strings
   * *idfa reuse across UAs* (many idfas sharing same UA)
   * *IP analysis*: top IPs, geolocation, ASNs, reverse DNS
   * *SDK & app versions* correlated with spikes (did an SDK release or campaign start at the time?
2. **Block / rate-limit suspicious sources**: IP ranges or UA strings that drive the abnormal volume while you investigate.
3. **Verify impression pipeline**: check server-side impression logging (are impressions being recorded but not exported?), and ensure ad SDK is reporting impressions correctly.
4. **Add monitoring & alerts**: set alert thresholds for large hourly growth in requests (e.g., >3× baseline) and for very low impressions\_per\_idfa.
5. **Enrich detection**: incorporate UA diversity, request-to-impression ratio, and sudden new device counts as weighted inputs for IVT scoring.
6. **Manual review**: for apps that get flagged, sample raw logs (requests + headers + timestamps) and playbacks to confirm bot patterns (regular intervals, identical headers, etc.).
7. **Communicate to partners**: provide evidence (hourly spikes, UA reuse, zero impressions) when appealing a false-positive IVT label.

**Quick remediation playbook (what to tell ops)**

* Immediately throttle or rate-limit any IP ranges or UA-strings that produce > X% of hourly requests (X = 5–10% depending on baseline).
* Create a temporary rule: if impressions\_per\_idfa < 0.01 and requests\_per\_idfa > 1.5 and unique\_uas < 100 for an hour → mark traffic as suspicious and block for further review**.(it’s a assumption** **)**
* Validate ad SDK versions & server logs for time windows corresponding to spikes.
* Keep a 7-day rolling baseline to detect spikes above normal variance (>3σ).