International Journal of Artificial Intelligence, Data Science, and Machine Learning
Grace Horizon Publication | Volume 7, Issue 1, 116-119, 2026
ISSN: 3050-9262 | https://doi.org/10.63282/3050-9262.IJAIDSML-V7I1P121

Original Article

Fraud and Bot Detection in Gaming Marketplaces

Anand Ganesh
Independent Researcher, USA.

Received On: 29/12/2025 ***Revised On: 31/01/2026*** ***Accepted On: 02/02/2026*** ***Published On: 04/02/2026***

Abstract *- Cloud-scale game marketplaces face increas-ing threats from automated activity and coordinated abuse, including repeated redemption of gift coupons and single- use discount codes, as well as the creation of free or secondary accounts to inherit starter currencies or in- game assets. This paper surveys practical detection approaches suitable for real-time production environments, emphasizing observable system signals rather than model-heavy or AI-driven methods. We describe techniques for monitoring API call sequences and rates, simple behavioral signals such as keystroke latency, and graph-based linkage analysis to detect clusters of accounts exploit- ing shared vouchers or resource transfers. Operational tradeoffs including latency, privacy, and computational cost are discussed, along with strategies for mitigating abuse through tiered interventions. This work provides a framework for deploying effective fraud detection while preserving legitimate user experience in rapidly evolving game marketplaces.*

Keywords *- Game Marketplaces, Fraud Detection, Bot Detection, Gift Coupon Abuse, Discount Code Mis-Use, Account-Based Fraud, API Monitoring, Keystroke Anomaly, Operational Security.*

1. Introduction

Online game marketplaces have grown rapidly, providing players with the ability to trade digital items, in-game currencies, and account privileges. While these platforms drive engagement and gen- erate significant revenue, they are increasingly tar- geted by fraud and automated activity. Common abuse patterns include repeated redemption of pro- motional assets such as gift coupons and discount codes, creation of secondary or free accounts to inherit starter resources, and coordinated exploita-tion of system loopholes to gain unfair economic advantage. These behaviors not only cause financial losses but also undermine user trust and marketplace integrity.

Traditional detection mechanisms, such as static rule-based monitoring or simple rate-limiting, are often insufficient when adversaries operate across multiple accounts or coordinate actions to bypass basic controls. Effective defenses require a combi- nation of behavioral monitoring, linkage analysis, and transaction auditing, all implemented in a man- ner that minimizes disruption to legitimate users. In this work, we survey practical detection strate- gies for cloud-scale marketplaces that rely on observable system signals. We describe approaches for API-call pattern monitoring, lightweight behavioral analysis such as keystroke latency, and graph-based relational techniques to identify clusters of accounts involved in coupon or currency exploitation. We further discuss operational tradeoffs related to scal- ability, latency, and privacy, providing a framework for deploying effective fraud detection pipelines in real-time production environments.

2. Background & Relevant Concepts

Fraud and bot activity in online game market- places manifests in multiple forms, ranging from simple automated interactions to sophisticated, co- ordinated exploitation of platform incentives. Un- derstanding these behaviors and the system signals they generate is essential for designing effective detection mechanisms. This section introduces the core concepts that underpin practical monitoring strategies, focusing on behavioral patterns, transac- tional abuse, and relational linkages.

2.1. Types of Marketplace Abuse

One of the most common vectors of abuse in game marketplaces is the automated or manual exploitation of promotional assets (1). Gift coupons, single-use discount codes, and promotional item grants are intended to incentivize legitimate users, but adversaries frequently reuse these codes across multiple accounts. For example, attackers may cre- ate free or “throwaway” accounts to redeem promotional offers repeatedly or transfer in-game currency and items to a primary account, effectively launder- ing the benefits of these promotions. Similarly, some bots exploit starter packages or inherited resources from free accounts, rapidly inflating their economic position without engaging in legitimate gameplay.

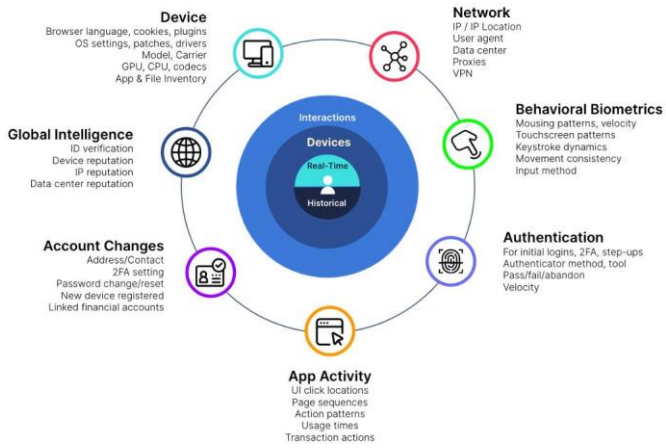

Fig 1: Behavioral Signals

Another prevalent category is API-level abuse. Automated clients can issue high-frequency requests to claim rewards, purchase limited items, or simu- late gameplay actions at speeds unattainable by hu- man users. These sequences often leave detectable patterns, such as repetitive API call orders, uniform time intervals between actions, or simultaneous ac- cess from multiple accounts with shared attributes (IP address, device fingerprints, or geolocation pat- terns).

2.1. Behavioral Signals

Behavioral monitoring focuses on identifying anomalies in how users interact with the platform. Simple signals such as keystroke latency, mouse movement patterns, or response times can reveal non-human activity (2). While adversaries can at- tempt to mimic human behavior, these signals often expose subtle deviations, particularly when aggre- gated across sessions or across multiple accounts (3). For instance, a bot issuing commands at regular millisecond intervals is easier to detect than a human performing the same actions naturally. Similarly, patterns in resource utilization such as repeated attempts to redeem the same type of coupon or frequent trades of low-value items for high-value itemscan be treated as indicators of coordinated abuse. Observing these patterns over time allows the system to establish baselines for legitimate activity and detect deviations that are statistically or operationally significant.

2.2. Graph and Relational Structures

Fraud in game marketplaces is rarely isolated to single accounts. Attackers often operate networks of linked accounts, coordinating to maximize their advantage. Representing these interactions as graphs enables relational analysis. Nodes in such graphs can represent users, accounts, or in-game entities, while edges represent interactions, transactions, or resource transfers. Graph-based analysis can reveal clusters of accounts that exhibit suspiciously high levels of interaction, share promotional assets, or collectively exploit system incentives. (4)

Linkage analysis is particularly effective for de- tecting abuse that might appear innocuous at the individual level but becomes evident when viewed in the context of network interactions. For example, several accounts repeatedly redeem the same type of gift code and funnel the acquired currency to a central “hub” account. While each transaction alone may appear legitimate, graph analysis uncovers co-ordinated activity indicative of fraud.

2.3. Operational Considerations

Implementing detection mechanisms in cloud- scale marketplaces presents unique challenges. Real-time systems must process millions of events per day without introducing significant latency. This constraint favors lightweight methods for initial filtering, such as counting API call frequencies or detecting rapid sequential actions, while more com- putationally intensive analysis, such as full graph traversal, can be performed asynchronously or on sampled data. Privacy is another critical concern. Behavioral telemetry, device identifiers, and user interactions must be collected and processed in compliance with regulatory frameworks such as GDPR and CCPA. Systems should prioritize anonymization, hashing, or aggregation to minimize exposure of person- ally identifiable information while preserving signal quality for fraud detection.

Scalability requires careful design of storage and computation pipelines. Approximate counting, locality-sensitive hashing, and incremental graph updates can reduce resource consumption while retaining sensitivity to anomalous patterns. Balanc- ing sensitivity, specificity, and operational cost is essential to maintaining a functional marketplace while limiting false positives that could negatively impact legitimate users. (5)

3. Discussion

This section synthesizes practical considerations that arise when deploying the methods described above in production game marketplaces. The dis- cussion emphasizes operational tradeoffs, measure- ment strategies, adversary adaptation, privacy and compliance concerns, and the relationship between detection capability and business outcomes. The goal is to translate algorithmic detection primitives into robust, maintainable defenses that preserve legitimate user experience while minimizing fraud losses.

3.1. Effectiveness versus Operational Cost

Detection approaches present a spectrum of costs and benefits. Lightweight heuristics (API-call rate limits, simple n-gram frequency checks, single-code reuse counters) are inexpensive to compute and can be applied at edge nodes with minimal latency. These methods are effective at blocking unsophisti- cated abuse (e.g., naive scripted reuse of coupon codes) and serve as a first-line defense. In con- trast, relational analyses (full graph traversal, cross- session linkage) and deeper historical correlation require heavier compute and storage resources but detect coordinated or stealthy campaigns that evade per-session heuristics.

A production deployment should therefore adopt a tiered architecture: fast, low-cost filters operate in the request path to drop or throttle obvious abuse; mid-cost pipelines perform lightweight ag- gregation and approximate counting to mark sus- picious actors; and expensive, offline

or near-line processes run comprehensive linkage and forensic analysis for cases requiring human review. This layered approach minimizes the number of costly operations performed, improves system throughput, and concentrates expensive inference on a small, high-value subset of traffic.

3.2. False Positives, User Experience and Business Tradeoffs

False positives are the central practical risk for any automated fraud control system. Overzealous blocking of legitimate users erodes trust, generates support costs, and can reduce revenue from pro- motions. Conversely, permissive thresholds increase fraud losses and marketplace instability. The balance between these outcomes must be informed by clear business objectives (e.g., acceptable revenue loss, support load limits, tolerance for user churn). Mitigations include graduated response policies, where initial flags lead to soft interventions (rate limiting, temporary captchas, challenge-response verification) rather than outright account suspension. Escalation policies should preserve user experience for borderline cases: require secondary verification only when multiple independent signals concur (e.g., coupon reuse + abnormal API sequencing + nascent account with starter currency transfer). Metrics such as false positive rate at fixed recall levels, and business KPIs (e.g., promotional redemp- tion lift, chargeback rate), should guide threshold selection and policy tuning.

3.3. Coupon, Discount Code and Starter Currency Abuse - Specific Considerations

Promotion abuse has domain-specific dynamics that influence detection design. Single-use codes and gift coupons are particularly attractive to at- tackers because they provide discrete, quantifiable value and can be redeemed rapidly. Key operational detection strategies include:

- Redemption Linking: maintain an immutable audit trail linking each redemption to a (hashed) redemption context anonymized device fingerprint, session identifier, IP prefix, and time window to detect high-risk reuse patterns without retaining raw PII.
- Temporal and Topological Patterns: detect bursty redemptions of the same code across ac- counts, or sequences where many new accounts redeem identical sets of coupons then funnel currency to a handful of receivers.
- Coupon Design Hardening: move from static single-use codes to constrained redemption modalities (e.g., time-bound vouchers tied to verified accounts, multi-factor redemption flows, or nontransferable starter items) to re- duce transferability and laundering risk.
- Economic Controls: implement per-account and net-flow limits for newly created accounts (e.g., cap on immediate transfers or market listings) and require aging or minimal activity prior to full transfer rights.

These measures must be aligned with market- ing goals restrictive controls can reduce legitimate promotional uptake so a close loop between prod- uct/marketing and security teams is essential.

3.4. Evasion and Adaptive Adversaries

Adversaries continuously adapt: bots may ran- domize inter-call timing, introduce jitter into keystroke patterns, or distribute redemptions across a geographically dispersed proxy network. Detec- tion systems must therefore be resilient to evasion through diversity in signals and continual recalibra- tion. Practical defenses include:

- Signal Diversity: combine orthogonal indica- tors (API sequencing, redemption metadata, account age, transaction graphs) so evasion requires coordinated changes across many di- mensions.
- Behavioral Baselines: maintain rolling base- lines per cohort (region, platform, campaign) to prevent drift and to reduce false positives when legitimate user behavior shifts.
- Adversarial Testing: incorporate red-teaming to model evolving attack techniques and to stress detection thresholds prior to production rollout.
- Deception and Honeytokens: strategically seed unique, unadvertised coupons or bait items to observe laundering behaviors with high confidence.

Defenders must accept a probabilistic detection regime and prioritize signals that are costly for attackers to fake (e.g., long-term relational ties, economic consistency across transfers).

3.5. Privacy, Legal and Ethical Constraints

Operational telemetry for fraud detection often contains sensitive identifiers. Privacy and compli-ance requirements (GDPR, CCPA, local data protec-tion laws) place constraints on collection, retention, and use:

- Minimization and Hashing: collect minimum necessary telemetry and store identifiers as cryptographic hashes or salted pseudonyms when possible.
- Purpose Limitation and Consent: ensure users are informed of fraud prevention uses where required, and enable opt-out pathways consistent with law and product requirements.
- Retention Policies: define short retention win- dows for raw telemetry and retain only aggre- gated or anonymized signals for longer hori- zons.
- Auditability: maintain tamper-evident logs of detection decisions and human interventions for regulatory review or dispute resolution.

Design choices should be documented in a pri- vacy checklist (Appendix B) and reviewed with legal counsel before deployment.

3.6. Scalability, Observability and Engineering Tradeoffs

At cloud scale, engineering constraints shape algorithmic choices. Key patterns include:

- Approximation Techniques: use approximate

counters (HyperLogLog, Count-Min Sketch) to bound memory while maintaining statistical fidelity. (6) (7)

- Edge Filtering: perform initial filtering at CDN/edge proxies to reject trivial abuse with- out round trips to centralized services.
- Asynchronous Processing: schedule heavy graph computations in batch or streaming win- dows; maintain incremental indices to avoid full rebuilds.
- Backpressure and Graceful Degradation: design pipelines to degrade detection fidelity under overload (e.g., favor sampling or more aggressive heuristics) rather than fail open.

Observability is critical: exposure of pipeline latency, queue lengths, and false positive/negative trends enables rapid triage. Diagnostic tooling should support replay of historical sessions for offline forensics.

3.7. Measurement, Metrics and Business Impact

Detection performance must be evaluated both technically and economically. Technical metrics in- clude precision and time-to-detect. Business met- rics translate these into outcomes: fraudulent value prevented, reduction in refund/chargeback volume, and impact on legitimate conversion or retention. A small increase in precision at high recall may yield large cost savings; conversely, aggressive blocking can reduce promotional return on investment. Time-to-detect is especially salient for coupon and starter currency abuse: detecting a fraudulent redemption within minutes can enable reversal or freezing of subsequent transfers, while days-long detection windows may render remediation inef- fective. Therefore, measurement should track both detection latency and the expected recoverable value per detection.

3.8. Mitigation Policies and Human-in-the-Loop Operations

Automated detection should be coupled with pragmatic mitigation policies. Recommended tiers include:

1) Soft Actions: impose rate limits, require second-factor confirmation for high-value re- demptions, or insert captcha challenges.
2) Intermediate Actions: temporarily hold dis- puted funds or items pending manual review, or require identity verification for account upgrades.
3) Hard Actions: suspend or ban accounts with high-confidence evidence of coordinated abuse.

Human analysts are essential for handling nu- anced cases, tuning heuristics, and adjudicating appeals. Tools should present concise evidence (redemption timelines, transaction graph snippets, cross-account comparisons) to enable fast, consis- tent decisions.

4. Conclusion

This paper surveys practical approaches for de- tecting fraud and automated activity in cloud-scale game marketplaces, focusing on operationally feasi- ble methods rather than model-heavy AI techniques. Techniques such as API-call pattern fingerprinting, keystroke latency anomaly detection, and graph- based linkage analysis provide complementary in- sights into suspicious activity, including coupon reuse, discount code abuse, and starter currency exploitation. We highlight tradeoffs between la- tency, computational cost, and detection coverage, and emphasize the importance of layered, tiered enforcement strategies to maintain user experience while mitigating losses. By combining lightweight real-time signals with deeper relational analysis and carefully designed mitigation policies, marketplaces can significantly reduce abuse while preserving the integrity of promotional programs. Future work will extend these methods to more adaptive adversaries and explore privacy-preserving operational pipelines to safeguard user data without compromising detec- tion efficacy.

References

[1] E. Lee, J. Woo, H. Kim, A. Mohaisen, and H. K. Kim, "You are a game bot!: Uncovering game bots in mmorpgs via self-similarity in the wild," in *NDSS (Network and Distributed System Security) Symposium*, 2016. [Online]. Available: https://www.ndss-symposium. org/wp-content/uploads/2017/09/you-are-game-bot-uncovering-game-bots-mmorp.pdf

[2] M. Fahey, "Keystroke lag alerts amazon security staff that recent us-based it hire was actually in north korea," PC Gamer (online), 2025. [Online]. Available: https://www.pcgamer.com keystroke-lag-alerts-amazon-security-staff-that-recent-us- gs-via-s

[3] A. R. Kang, S. H. Jeong, A. Mohaisen, and H. K. Kim, "Multimodal game bot detection using user behavioral characteristics," *SpringerPlus*, vol. 5, p. 523, 2016. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/ PMC4844581/

[4] Y. Yang, Q. Wu, B. He, H. Peng, R. Yang, Z. Hao, and Y. Liao, "Sebot: Structural entropy guided multi-view contrastive learning for social bot detection," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2024.[Online]. Available: https://arxiv.org/abs/2405. 11225

[5] P. Indyk and R. Motwani, "Approximate near- est neighbors: Towards removing the curse of dimensionality," in *Proceedings of the 30th An- nual ACM Symposium on Theory of Computing (STOC)*, 1998, pp. 604–613.

[6] G. Cormode and S. Muthukrishnan, "An improved data stream summary: the count–min sketch and its applications," *Journal of Algo- rithms*, vol. 55, no. 1, pp. 58–75, 2005.

[7] P. Flajolet, E Fusy, O. Gandouet, and F. Meunier, "Hyperloglog: the analysis of a near-optimal cardinality estimation algorithm," in *Proceedings of the 2007 International Conference on Analytic Algorithms and Combinatorics (AofA)*, 2007, pp. 127–146.[Online]. Available: https://algo.inria.fr/flajolet/Publications/FlFuGaMe07.pdf