Annotated Bibliography

Godfred Tekpor

February 19, 2026

References

[1] Ankit Gangwal, P. Sahithi Reddy, and C. y. k. Sagar. *Swiss Cheese CAPTCHA: A Novel Multi-barrier Mechanism for Bot Detection*, pages 1780–1789. Association for Computing Machinery, New York, NY, USA, 2025.

> The paper proposes ”Swiss Cheese CAPTCHA,” a multi-step challenge meant to stay easy for humans but harder for bots by stacking several independent obstacles under a time limit, including signals from device sensors through the Generic Sensor API. It reports two user studies showing most people can complete the challenge in about 5–6 seconds with high success rates, and it uses trajectory data to study how users recover from mistakes and learn the task. The paper also tests security by building an automated attack and finds a low success rate for that attack in their experiments. A strong point is that it evaluates both usability and security, which matters for real e-commerce defenses, while a limitation is that the sensor-based design may not work as well in desktop-heavy reseller scenarios where sensor access is limited. For a navigation-pattern risk scoring system, the main value is as an escalation tool: suspicious sessions can be routed into a stronger challenge instead of being blocked outright, and the paper offers a clear way to measure the security–friction tradeoff when adding that step to a layered defense pipeline.

[2] Christos Iliou, Theodoros Kostoulas, Theodora Tsikrika, Vasilios Katos, Stefanos Vrochidis, and Ioannis Kompatsiaris. Web bot detection evasion using deep reinforcement learning. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*, ARES ’22, New York, NY, USA, 2022. Association for Computing Machinery.

> The paper examines whether bots can learn browsing and navigation patterns that slip past behavior-based detectors built from web-log features. It models evasion as a reinforcement learning problem where a bot chooses actions to shape session features while still completing its goal, using a pre-trained detector as the environment and getting rewarded when it is not flagged. Results show simple scripted bots rarely evade, but RL-trained bots can learn evasive strategies and reach much higher evasion rates, with some settings

approaching about half of bot sessions avoiding detection after enough training. The paper then shows an attacker–defender loop: once the detector is retrained on the new behaviors, bots can train again and regain evasion capability. A key strength is the explicit adversarial setup and staged evaluation over time, while a limitation is that the experiments focus mainly on manipulating web-log features and do not fully incorporate real reseller-bot constraints like identity and payment linkages, queueing, and rate limits. For navigation-pattern risk scoring, the paper provides a clear threat model and a reusable stress-test method by training adaptive bots to minimize a risk score while still reaching add-to-cart and checkout.

[3] Christos Iliou, Theodoros Kostoulas, Theodora Tsikrika, Vasilis Katos, Stefanos Vrochidis, and Ioannis Kompatsiaris. Detection of advanced web bots by combining web logs with mouse behavioural biometrics. *Digital Threats*, 2(3), June 2021.

This paper presents a detection framework aimed at advanced web bots that spoof browser fingerprints and attempt to imitate human browsing. It builds two separate detectors: one based on session-level web-log features (navigation paths, timing, and request patterns) and another based on mouse-movement features, then fuses the two scores with a rule that relies more heavily on mouse evidence and uses the web-log score when the case is uncertain. The evaluation uses a controlled test site with human sessions plus two bot classes, moderate bots that mainly spoof fingerprints and advanced bots that also simulate humanlike browsing and mouse behavior. Results show the fused approach performs better than a web-log-only detector and remains effective under attempted behavioral imitation, with very strong reported classification metrics. A key limitation is external validity: because the study uses a test environment and simulated bots, real reseller-bot campaigns that use proxy rotation, distributed accounts, and trace replay could shift the feature distributions and reduce performance.

[4] Rizwan Ur Rahman and Deepak Singh Tomar. New biostatistics features for detecting web bot activity on web applications. *Computers & Security*, 97:102001, 2020.

This article introduces "biostatistics" style features for bot detection that emphasize human interaction signals rather than only server-side request patterns. The authors argue that many bot detectors are too tailored to specific websites or attack types, so they propose more general features derived from how humans complete tasks online. Examples include timing-related signals such as time spent on pages, time spent in form fields, and patterns that capture how users move through forms (for example, focus changes, field navigation behaviors, and typing dynamics). The goal is to detect bots by identifying interaction traces that differ from natural human behavior, especially in workflows like registration and checkout where automation is common. The paper also discusses efficiency and scalability considerations when processing large volumes of sessions, proposing methods to structure and group behavior patterns for classification. Credibility-wise, this is a peer reviewed security publication that offers a strong conceptual foundation: behavioral authenticity is harder to fake perfectly than headers or IP patterns. The main tradeoff is implementation complexity, since collecting these signals often requires client-side instrumentation and careful privacy handling. For my reseller bot project, these features map directly to checkout abuse, giving me a justified feature set for detecting scripted purchasing behavior.

[5] Rizwan Ur Rahman and Deepak Singh Tomar. A new web forensic framework for bot crime investigation. *Forensic Science International: Digital Investigation*, 33:300943, 2020.

This paper shifts focus from prevention to investigation and attribution, proposing a structured web forensics framework for bot-enabled crimes. The authors argue that bot activity is widespread and increasingly linked to harmful outcomes such as automated abuse, illegal scraping, and other forms of web exploitation. Their framework organizes an investigation into multiple phases, guiding how to identify bot-driven incidents, collect relevant evidence, reconstruct sessions, and analyze behavioral traces in a way that supports a defensible conclusion. A key point is that basic access logs often provide incomplete visibility, so investigators should capture and correlate richer artifacts (for example, timelines of actions, repeated behavioral signatures, and contextual indicators) to understand what occurred and how automation was used. The paper illustrates the framework through a case-based evaluation on a web application scenario, showing how structured collection and analysis can clarify whether bots were involved. Credibility-wise, it is published in a digital forensics venue, which makes it a strong source for audit and evidence practices. Its limitation is that frameworks can be high-level, so implementation details must be translated into concrete logging and reporting requirements. For my reseller bot project, this supports adding an audit trail and investigator-ready summaries that explain why a session was flagged and what signals drove the decision.

[6] Athena Stassopoulou and Marios D Dikaiakos. Web robot detection: A probabilistic reasoning approach. *Computer Networks*, 53(3):265–278, 2009.

This article proposes a web robot detection approach that relies on server access logs and session-level classification. The authors first transform raw log requests into browsing sessions using rules that split visits based on inactivity thresholds, then compute behavioral features from each session. They classify sessions as human or robot using a Bayesian network, which is useful because it produces not only a label but also a probability score that reflects confidence. The paper emphasizes that single indicators (like user agent strings) are weak on their own, so combining multiple signals improves reliability. The evaluation uses real log data and reports strong detection performance, while also acknowledging failure cases such as short sessions, ambiguous browsing patterns, or traffic types that do not fit the training distribution. From a credibility standpoint, this is a peer reviewed journal article with a clear, reproducible modeling pipeline and a realistic data source. The main limitation is that it is log-centric, so it may miss modern client-side indicators (JavaScript execution, mouse movement, typing cadence). For my reseller bot project, this paper supports building a baseline risk scoring model from session features, then using the score to trigger mitigation like throttling, step-up checks, or queue placement.

[7] Takamasa Tanaka, Hidekazu Niibori, Shimpei Nomura, Hiroki Kawashima, Kazuhiko Tsuda, et al. Bot detection model using user agent and user behavior for web log analysis. *Procedia Computer Science*, 176:1621–1625, 2020.

This paper focuses on detecting bots in web traffic when bots can spoof user agent strings. The authors argue that relying on user agent alone is unreliable, so they combine user agent information with user behavior features extracted from web logs. They define visits as cookie-based sessions with an inactivity timeout and propose a detection model that improves bot identification by incorporating patterns of navigation and request behavior. A major practical contribution is the discussion of labeling: they suggest using whether JavaScript executed as a signal for human activity, while acknowledging that labeling can be imperfect due to network conditions and because advanced bots may execute JavaScript. The paper frames bot filtering as essential for preventing bots from corrupting analytics and decision-making systems built on web log data. Credibility-wise, it is published as a conference proceedings style paper, so it is shorter and less detailed than a full journal study, but it still presents a clear and realistic approach grounded in production-style data. For my reseller bot project, this directly supports two design choices: treat user agent as a weak feature, and prioritize behavioral plus client capability signals to catch automation that tries to blend in.