BaitBlock: Measuring and Mitigating Phishing Propagation in YouTube Live Chats

Name	*Net ID*	*Email*	*Program*
Andy Wu	cw4483	cw4483@nyu.edu	B.S. Computer Engineering

Project Option

Option Two: Platform Measurement Project (Select a platform and harm type, develop an automated detector, and analyze the system.)

1 Summary

YouTube Live remains a popular target for impersonation and fraud. Attackers commit phishing attacks in chat rooms with cloned or misleading accounts that mimic influencers. While phishing attacks are well studied, YouTube Live introduces a new terrain that is fast-moving, loosely moderated, and dominated by a young demographic. This has created a new phishing style where the lure used in the attacks is almost always cryptocurrency-themed (e.g, cryptocurrency “airdrops,” NFT releases, or giveaway events).

BaitBlock will serve as a lightweight Chrome Extension designed to identify and flag potential scam messages in real time. It will leverage the YouTube Data API to collect live chat data from streams categorized under Finance, Web3, and Influencer. Through rule-based pattern detection, combining linguistic cues, repeated character sequences, and URL heuristics, BaitBlock will surface suspicious messages and quantify their prevalence across sampled events. The project aims not only to expose emerging scam techniques but to measure how effectively YouTube’s moderation systems respond under live conditions. Hopefully, BaitBlock’s findings will ultimately contribute to practical safety recommendations.

2 Connection to Trust and Safety Themes

BaitBlock aligns directly with the course’s core themes of Abuse Vectors and Content Moderation Infrastructures. It situates itself within the growing field of platform abuse measurement and identifies how deception flows through systems designed for engagement and scale. By focusing on impersonation and financial deception, BaitBlock examines the archetypal adversarial dynamic of automated abuse and reactive moderation that is frequently covered in lectures.

From a trust and safety engineering perspective, BaitBlock also intersects with lectures on User-Facing Defenses and Auditing Safety Interventions. The extension functions both as a diagnostic tool and as an audit mechanism, evaluating whether YouTube’s defenses meaningfully deter scams. Through this lens, I believe the project embodies the spirit of the course.

3 Measurement Plan

BaitBlock’s methodology will center on structured data collection and rule-based detection. Using the YouTube Data API, I will gather live chat messages and relevant metadata (e.g., timestamp, channel ID, user ID) from selected streams in the Finance, Web3, and Influencer categories. The collection window will target active events with high engagement to capture authentic adversarial dynamics.

The detection logic will first be implemented solely based on a mix of heuristic and linguistic signals. For instance, BaitBlock will flag messages containing phrases like “limited-time airdrop,” “send crypto to verify,” or “claim NFT reward” and/or external URLs or wallet addresses. Repetition rates and user activity patterns will also be analyzed to identify automation and coordinated behavior. The detector’s performance will be manually annotated to measure false positives and false negatives to calibrate the sensitivity of the rule-based filter.

More importantly, this annotation will serve as training data for a small supervised neural learning model. Over the course of this project, the model will distinguish between benign promotional chatter and harmful scam material with varying confidence. The final detection logic should combine results from the learning model and the rule-based system.

After scraping the chatroom, BaitBlock should store all flagged and unflagged comments to be cross-referenced with YouTube's visible moderation events, such as auto-deleted messages and shadow bans, to assess the platform's and BaitBlock's responsiveness.

4 Anticipated Challenges

Several challenges are anticipated. First, YouTube's Data API imposes rate limits that constrain real-time scraping. Popular coatrooms in live conditions also update extremely quickly. Efficient sampling with YouTube's Data API and an efficient backend is required. Second, rule-based detection will struggle with nuance. Attackers constantly adapt phrasing to evade filters, leading to false negatives. Conversely, overly strict patterns could produce false positives, flagging legitimate promotional content. Front-end should be deployed with Vite and React due to its simplicity and Chrome support. Data parsing should be handled by Python scripts integrated with C++ libraries (e.g, NumPy, SpaCy) for speed and modularity.

Ethically, BaitBlock must avoid collecting or storing identifiable user information beyond what is publicly available, in accordance with platform policies. To mitigate these risks, I will anonymize user IDs, exclude private metadata, and focus solely on publicly visible chat data. Finally, platform moderation visibility is limited. YouTube often removes content silently. This opacity may complicate analysis, but will itself become a key metric of system transparency.

5 Expected Outcomes

By semester's end, BaitBlock will produce a functional Chrome Extension with an intuitive front-end that unintrusively flags chatroom comments. The back-end will be capable of flagging potential scam messages in YouTube Live chats and deliver a dataset quantifying scam prevalence across selected categories. The project will yield an empirical understanding of how impersonation-based scams propagate in livestream contexts and where moderation systems break down. More broadly, it will demonstrate how trust and safety measurement can inform design, translating platform-level failures into actionable moderation insights. The final report will include both technical findings and ethical recommendations for improving live content integrity.