Understanding Web-Exposed Cybercrime-Related Content on IoT Botnet Infrastructure

Qingxin MAO[†], Yin MINN PA PA[††], Rui TANABE[††,†††], and Katsunari YOSHIOKA[††,††††]

† Yokohama National University 79–1 Tokiwadai, Hodogaya-ku, Yokohama, 2408501 Japan
†† Institute of Advanced Sciences, Yokohama National University
†††† Graduate School of Environment and Information Sciences, Yokohama National University
††† Juntendo University

E-mail: †mao-qingxin-fp@ynu.jp, ††{yinminn-papa-jp,tanabe-rui-xj,yoshioka}@ynu.ac.jp

Abstract In recent years, IoT botnets conducting cyberattacks such as DDoS attacks have become a significant threat, making it important to understand their operational realities. Prior work has characterized IoT botnet ecosystems by analyzing their growth and evolution, measuring properties of Command and Control (C2) and download-server infrastructure (e.g., lifetime and hosting environments). While these works show interesting findings on the IoT botnet, primarily works focused on the attacks and the devices involved, without addressing the relationship between the attacks and the attackers themselves. Therefore, it remains unclear if the attackers are utilizing their infrastructure for other cybercriminal activities beyond executing attacks. Our prior study investigated web-exposed content on IoT botnet infrastructure but focused only on the DDoS-for-hire cases. In this study, we broaden the scope to cover multiple cybercrime types and measure what other cybercrime activities are hosted on IoT botnet infrastructure. We conduct a one-year longitudinal crawling campaign of webpages exposed on IoT botnet infrastructure (Aug. 2024–Aug. 2025), collecting 43,406 rendered webpage screenshots from the infrastructure IPs and their associated domains. Using an LLM-based visual classifier, we identify 1,509 screenshots related to cybercrime, which corresponds to 3.48% of the raw dataset, suggesting that such cybercrime-related reuse of IoT botnet infrastructure is not pervasive in our observation. We then manually group screenshots with identical webpage content into cases and remove cases with low relevance to attacker-controlled infrastructure, resulting in 32 final cases of infrastructure reuse. These cases span all five cybercrime offence types (Types A–E), showing that the web-exposed content observed on IoT botnet infrastructure is not limited to a single category of cybercrime activity. Finally, our case-level analysis reveals that some cases appear across multiple infrastructure roles; this indicates that certain infrastructure IPs are used in multiple botnet-infrastructure roles (e.g., simultaneously appearing as C2/download/loader endpoints) while also exposing web content corresponding to other types of cybercrime activities.

Key words CaaS, IoT Botnet Infrastructure, Passive DNS, LLM

1. Introduction

IoT botnets remain a persistent threat on the Internet. By compromising vulnerable IoT devices at scale, attackers can build large botnets and leverage them for monetization, most notably through activities such as DDoS attacks and malware distribution. Operating such campaigns requires Internet-facing infrastructure—including loaders, malware download servers, and command-and-control (C2) servers—to coordinate intrusions, deliver payloads, and manage infected devices [12], [9].

Existing work has primarily focused on botnet operations themselves, such as attack execution, malware propagation, and device compromise. For example, Antonakakis *et al.* [1] presented the growth, composition and evolution of Mirai botnet. They identified 33 C&C clusters that shared no infrastructure and estimated their relative size using different datasets. Bastos *et al.* [2] analyzed Bashlite and Mirai's C&C servers, showing that most of them were seen only for a few days, and that 84% of them were hosted in cloud providers. Tanabe *et al.* [12] investigated how binaries, download servers and C&C servers are related to each other, revealing how they evolve over time and how attackers source their IP addresses. As a result, the current understanding of IoT botnet infrastructure is largely limited to its role in supporting attacks. This narrow focus leaves us with an important question: whether Internet-facing botnet infrastructure exposes or supports other forms of cybercrime-related activity that are observable through standard web protocols. From a defensive perspective, such exposure matters because web-accessible services are routinely monitored and indexed, and may unintentionally reveal operational artifacts, secondary services, or misuse patterns that are not visible through traditional botnet-centric analyses. Our prior work [7] investigated IoT botnet infrastructure but focused on a single case study of DDoS-for-hire. What other types of cybercrime activities attackers additionally operate on such infrastructure remains to be investigated.

In this paper, we broaden the scope to cover multiple cybercrime types and aim to empirically characterize what other cybercrime activities are hosted on IoT botnet infrastructure beyond botnet-related functions. We conduct a one-year longitudinal study (Aug. 2024–Aug. 2025) by repeatedly collecting webpages exposed on IoT botnet infrastructure. Starting from infrastructure IP addresses obtained via our IoT honeypots and prior malware analysis results, we additionally expand candidate domains using passive DNS (DNSDB)

within a time window around each observation. We then crawl HTTP/HTTPS and collect rendered webpage screenshots at scale, resulting in 43,406 screenshots. Using an LLM-based visual classifier, we identify 1,509 screenshots related to cybercrime. We further group screenshots with identical webpage content into cases and remove cases with low relevance to attacker-controlled infrastructure, yielding 32 unique cases. We find that download servers dominate the observed reuse cases (26/32), while loaders and C2 servers contribute 4/32 and 2/32, respectively. The hosted activities span multiple cybercrime offence types, including attack-related resources such as SQL injection demo pages and stress-test pages (Type A), fraud-related services (Type B), pornography and illegal gambling (Type C), copyright-related torrent portals (Type D), and phishing and bank impersonation (Type E).

This paper makes the following contributions:

- We present an LLM-based visual classification pipeline to identify web-exposed cybercrime-related content and categorize it under an established cybercrime offence taxonomy.
- We quantify the prevalence of web-exposed cybercrime-related content on IoT botnet infrastructure in our one-year measurement observation, showing that such content is observed on only a small fraction of collected pages.
- We report 32 cases in which IoT botnet infrastructure exposes other web-based cybercrime activities, spanning all five cybercrime offence types, indicating that the observed content is not limited to a single category of cybercrime activity.
- We identify overlapping cases across infrastructure roles, showing that some IP addresses in our observation appear in multiple botnet-infrastructure roles (e.g., C2/download/loader) while also exposing web content corresponding to other web-based cybercrime activities.

2. Data Sources

This section describes the data sources used to construct the seed sets of IoT botnet infrastructure for our longitudinal web measurement. Our study focuses on three infrastructure roles commonly observed in IoT botnet operations: Loaders IPs, Download Servers IPs, and Command-and-Control (C2) Servers IPs. We emphasize that our contribution is the subsequent longitudinal web measurement and analysis built on top of these IPs, rather than the underlying honeypot deployment or the malware analysis pipeline itself.

2.1 IoTPOT Honeypot Observations

We leverage *IoTPOT* [9] , a large-scale telnet-based honeypot system designed to monitor IoT botnet activities in the wild. IoTPOT emulates IoT devices/services and records inbound interactions, including repeated intrusion attempts and payload delivery behavior. Importantly, IoTPOT records observations on a **daily** basis: the same infrastructure IP address can be observed on multiple days, and our dataset retains these **per-day observation timestamps** rather than only the first-seen time. From IoTPOT observations, we obtain seed IPs for two infrastructure roles:

- **Loader IPs.** We use the term loader to denote infrastructure involved in the initial compromise workflow that enables botnet propagation. In practice, we observe repeated intrusion behavior where authentication information (e.g., Telnet credentials) is reused across subsequent intrusions, suggesting the presence of automated loaders that repetitively reuse credentials, consistent with prior measurements [6] . Operationally, we treat IPs that repeatedly appear as sources of such intrusion activity (often observed many times per day in our logs) as loader seeds.
- **Download Server IPs.** We define a download server as an endpoint that hosts and serves IoT malware binaries or related payloads during the infection process. In our setting, download server indicators are extracted from IoTPOT-captured communication payloads, which typically include URLs or IP endpoints used by compromised devices to retrieve malware.

Because the same loader/download infrastructure may be observed across multiple days, seed IPs are associated with potentially multiple observation dates. These timestamps later anchor our passive DNS enrichment window and longitudinal crawling schedule.

2.2 C2 Seeds from Prior Malware Analysis and Active C2 Verification

To include command-and-control infrastructure, we use a set of C2 server IPs derived from performed malware analysis of IoTPOT-captured samples. During dynamic execution, samples may communicate with multiple IP addresses, some of which may be unrelated (e.g., decoy endpoints inserted to obscure true C2 infrastructure). To increase the precision of our C2 seeds, we apply an additional verification step using *Milker* [4], [5], a framework developed in our previous work to emulate malware–C2 interactions.

Specifically, we first extract a set of candidate C2 endpoints from dynamic analysis network traces. We then use Milker to replay and mimic the protocol behavior of the malware family and interact with each candidate endpoint. Only endpoints that respond with valid attack commands (i.e., actionable C2 instructions) during this emulation are retained as confirmed C2 seeds. In this paper, we do not re-run dynamic analysis or develop Milker; instead, we leverage the resulting confirmed C2 IP indicators as seeds for the C2 category.

Because dynamic analysis and Milker verification are performed after sample capture, confirmed C2 indicators may be available with a short delay (typically on the order of hours to one day) relative to the corresponding IoTPOT observation day; we account for this by anchoring domain expansion around the observation day used in our measurement pipeline.

2.3 Seed IP Dataset Summary

Our longitudinal measurement spans Aug. 1, 2024 to Aug. 24, 2025. Over this period, we construct three seed sets of unique infrastructure IP addresses:

- **Download Servers:** 9,296 unique IPs
- **Loaders:** 548 unique IPs
- **C2 Servers:** 88 unique IPs

Download-server and loader seeds are derived from the same IoTPOT observation window. C2 seeds are derived from the corresponding malware samples via dynamic analysis and may lag slightly in time, but are used within the same overall measurement period. These seeds serve as the starting point for our passive DNS domain expansion (Section 4.1) and subsequent longitudinal web crawling on HTTP/HTTPS (Section 4.2). Because infrastructure and IP–domain bindings are inherently dynamic, the seed sets should be viewed as entry points into a larger, evolving ecosystem rather than an exhaustive enumeration.

2.4 Scope and Assumptions

This work is intentionally scoped to web-visible content reachable via HTTP/HTTPS (commonly ports 80/443). Our goal is to measure whether botnet-related infrastructure endpoints also expose webpages indicative of other cybercrime activities. As a result, our data sources and subsequent analysis do not aim to cover non-web services (e.g., SSH-only servers or custom C2 protocols without web frontends), nor do we claim actor attribution. Instead, we later apply conservative passive-DNS-based heuristics to increase confidence that observed web content is attributable to the infrastructure under analysis (Section 4.4).

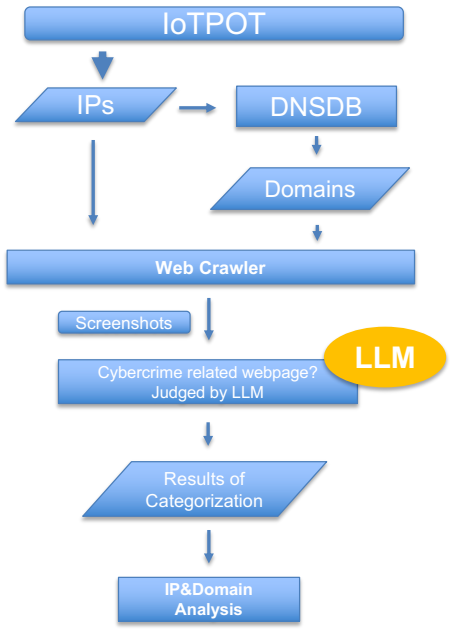

Fig. 1: End-to-end workflow of our study.

3. Methodology

Figure 1 summarizes our end-to-end measurement pipeline for identifying web-visible cybercrime activities co-located on IoT botnet infrastructure. Starting from seed infrastructure IPs (Section 2), we (i) expand the target space with passive DNS to obtain historically associated domains, (ii) perform longitudinal web crawling over HTTP/HTTPS to collect rendered webpages, (iii) use an LLM-based visual classifier to label each page as cybercrime-related or benign and, when relevant, categorize it under a published cybercrime offence taxonomy (Types A–E), and (iv) apply conservative IP/domain heuristics to reduce artifacts caused by shared hosting and ambiguous domain resolution, producing a set of cases for analysis.

Throughout this paper, we treat IP-based crawling and domain-based crawling as complementary. IP-only probing can miss content that is primarily exposed via domains, while domain-only probing is sensitive to rapidly changing IP bindings. Therefore, we crawl seed IPs directly and also enrich each seed IP with candidate domains obtained from passive DNS within a temporally constrained window anchored at the daily observation time recorded by IoTPOT.

3.1 Passive-DNS Domain Expansion

IP–domain mappings for infrastructure are dynamic. Operators may move services across IPs, rotate domains, or use short-lived bindings, which can cause pure IP-based crawling to miss relevant web content. To improve coverage, we augment each seed IP with domains that were historically observed resolving to that IP, using passive DNS.

Naively collecting all domains ever associated with an IP address can introduce substantial noise, especially for IPs that serve many unrelated domains. To reduce unrelated domains while retaining temporally relevant ones, we apply a time-window filter anchored at the daily observation time of each seed IP.

Let t denote the observation day when a seed IP is recorded by IoTPOT. We define a symmetric observation window:

$$W(t, \Delta) = [t - \Delta,\ t + \Delta],$$

where Δ is a tunable window size in days. For each domain record returned by passive DNS, we obtain its `first_seen` and `last_seen` timestamps. We include a domain as a candidate target for the seed IP if and only if its observed lifespan overlaps the window:

$$[\texttt{first\_seen}, \texttt{last\_seen}] \cap W(t, \Delta) \neq \emptyset.$$

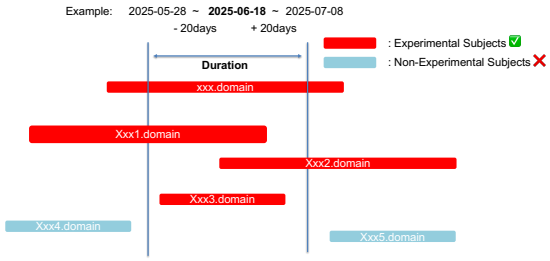

Fig. 2: Illustration of our temporal filtering for passive DNS domain expansion.

This criterion retains domains that were active around the time the IP was observed in IoTPOT, while excluding domains whose association to the IP occurred far outside the infrastructure observation period (Fig 2).

For each infrastructure category (loader, download server, C2), the above procedure produces a set of candidate domains associated with the corresponding seed IPs around their observation days. Because the same IP may be observed on multiple days, and a domain may reappear across multiple observation windows, this domain expansion naturally produces repeated targets. We therefore maintain both (i) the raw per-observation target list (for longitudinal crawling), and (ii) a de-duplicated set of unique domains for reporting dataset scale.

3.2 Longitudinal Web Crawling

We perform longitudinal web crawling for both seed IPs and passive-DNS-expanded domains to capture how web-exposed content changes over time. To ensure scalable collection, we restrict retrieval to the first page reached for each navigation attempt (i.e., the initial landing page after direct navigation), and do not crawl secondary pages that require user clicks or additional interaction beyond loading the initial page. We target standard web protocols (HTTP/HTTPS) because our analysis focuses on web-visible activities hosted on infrastructure endpoints.

3.3 LLM-based Visual Classification

Given the scale of collected webpages, we use an LLM to classify screenshots in two stages: (i) whether a page is potentially related to cybercrime (Yes/No), and (ii) for relevant pages, the closest cybercrime offence type under a published taxonomy. We adopt the cybercrime offence taxonomy of Tsakalidis and Vergidis [13], which organizes offences into five top-level types (A–E) as follows. The LLM outputs a structured label along with a concise evidence-based rationale grounded in visual indicators and on-page text.

Type A covers offences against the confidentiality, integrity, and availability of computer data and systems; **Type B** covers computer-related offences such as fraud, forgery, and identity theft; **Type C** covers content-related offences such as pornography and illegal gambling/online games; **Type D** covers offences related to infringements of copyright and related rights; and **Type E** covers combinational offences such as phishing and cyber laundering.

3.4 Cybercrime-related Case Selection

Web content observed on an infrastructure IP or domain does not necessarily imply exclusive control of the underlying server, due to shared hosting environments and time-varying domain-to-IP mappings. To increase confidence that observed content is attributable to the infrastructure under analysis, we apply conservative passive-DNS-based heuristics to filter (i) IPs associated with a very large number of domains (multi-tenant environments) and (ii) domain-derived cases with ambiguous domain-to-IP mappings in the rele-

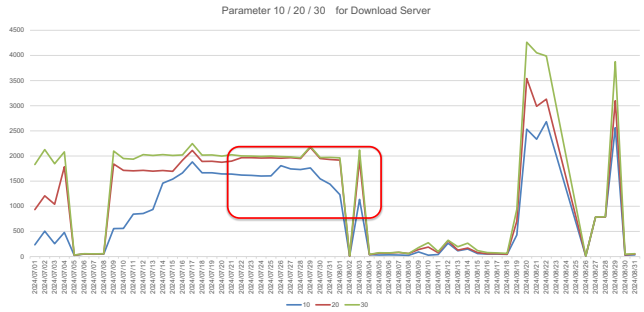

Fig. 3: Empirical tuning of the passive-DNS window size Δ on download-server seeds observed from 2024/07/01 to 2024/08/31.

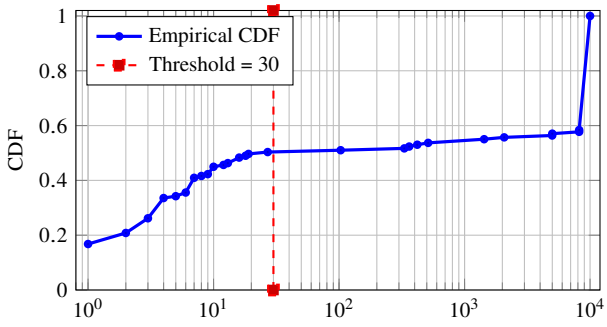

Fig. 4: CDF of the number of domains associated with each involved IP (DNSDB within ±20 days). The distribution is highly skewed: many IPs are associated with few domains, while many others are associated with ≥10,000 domains (plotted as 10,000). The dashed line marks the threshold of 30 used in our filtering heuristic.

vant time window. For example, if an IP address is associated with thousands of domains in the same time window, the observed webpage may belong to only one tenant on that IP, and thus may not reflect strong control of the underlying server by the actor operating the IoT botnet infrastructure. Likewise, a domain discovered via passive DNS may resolve to multiple IP addresses within the relevant time window (e.g., due to CDN use or infrastructure rotation); in such a case, a screenshot collected by visiting the domain cannot be confidently tied to a specific infrastructure IP without additional verification. The concrete thresholds and operational criteria are detailed in Section 4.

4. Implementation

This section details the concrete implementation of the pipeline described in Section 3, including data sources used for passive DNS enrichment, crawler configuration, LLM prompting and validation, and the operational criteria for selecting cases.

4.1 Passive-DNS Domain Expansion

We use DNSDB [10] , a passive DNS database that stores historical DNS query/response observations. For each seed IP, we query DNSDB for domains whose DNS **A records (IPv4)** were observed pointing to that IP. (We do not use AAAA/IPv6 records.)

We empirically evaluate $\Delta \in \{10, 20, 30\}$ days using download-server IPs observed between 2024/07/01 and 2024/08/31. Figure 3 shows that expanding the window from 20 to 30 days yields negligible additional domains, whereas a 10-day window substantially reduces domain coverage. To balance coverage and temporal relevance, we set $\Delta =$ **20** days for the remainder of this study.

4.2 Longitudinal Web Crawling and Screenshot Collection

For each day in our measurement period (Aug. 1, 2024 to Aug. 24, 2025), we generate a daily target list consisting of: (i) seed IPs observed by IoTPOT on that day, and (ii) domains associated with those IPs via DNSDB expansion within the corresponding ±20-day window (Section 4.1).

For each target (IP or domain), we attempt to retrieve web content via both HTTP and HTTPS (commonly ports 80 and 443). We use two separate crawler scripts to collect HTTP and HTTPS. We run the HTTP crawler first and then the HTTPS crawler. We treat the two scheme attempts independently because some services are exposed only on one scheme or behave differently across schemes. For HTTPS, SSL certificate verification is disabled to allow collection from endpoints with misconfigured or self-signed certificates.

We implement the crawler using Selenium [11] with headless Chrome. For each navigation, we capture (i) a screenshot of the rendered page and (ii) the HTML source code. The browser viewport is initialized at 1920×1080. We wait until the document load completes (i.e., `document.readyState = complete`) with a maximum wait of 15 seconds. Since modern webpages may load content dynamically, we additionally scroll to the bottom of the page to trigger lazy-loaded elements before taking the final screenshot. To obtain full-page screenshots, we resize the browser window height to the page scrollHeight when possible.

To ensure stable and efficient large-scale collection, we restrict retrieval to the first page reached for each navigation attempt (i.e., the initial landing page after direct navigation, including automatic HTTP redirects). We do not crawl secondary pages that require user clicks or additional interaction beyond loading the initial page, as such interactions would substantially increase crawling time and complexity and reduce throughput.

We set a page load timeout of 15 seconds for each navigation. Crawling is executed with a fixed concurrency of three parallel workers using a multi-process pool. In addition to Selenium timeouts, we enforce a task-level timeout (e.g., 120 seconds) to skip targets that stall due to rendering issues or slow dynamic loading. We do not introduce explicit inter-target delays (i.e., no per-target sleep) beyond the concurrency limit.

4.3 LLM-based Visual Classification

Given a screenshot image, the LLM performs (i) relevance detection (Yes/No) and (ii) cybercrime offence-type categorization (Type A–E or Unknown) for Yes pages, following the taxonomy of Tsakalidis and Vergidis [13] . The model outputs a strict JSON object containing judgement, category, and a short reason citing salient visual indicators.

We use the LLM gpt-5.2 [8] and query it once per screenshot. To reduce false positives in large-scale measurement, we include conservative labeling rules in the prompt (e.g., treating blank/404/default pages and generic domain-parking pages as No, and classifying access-denied/WAF interstitials as No unless there is additional clear evidence of illicit activity). We provide the full prompt template in Appendix for reproducibility.

4.4 IP/Domain Heuristics for Case Selection

Starting from the set of cybercrime-related screenshots, we manually group identical web content into cases and exclude IoT device login pages. We then apply two passive-DNS-based filters to reduce ambiguity caused by multi-tenant environments and time-varying domain resolution.

For each case, we query DNSDB for the number of distinct domains associated with the involved IP within the relevant time window. If the same IP address is observed on multiple days (and thus corresponds to multiple observation windows), we compute the domain count for each window and use the maximum value as the case-level domain-count proxy for that IP. We use this proxy to approximate how strongly the observed web content can be attributed

Table 1: Confusion Matrix for Cybercrime-Related Detection (LLM vs. Human Ground Truth)

		Human Label	
		No	**Yes**
LLM Label	**No**	198 (TN)	2 (FN)
	Yes	11 (FP)	189 (TP)

Table 2: Performance Metrics for Cybercrime-Related Detection

Metric	**Score (%)**
Accuracy	96.75
Precision	94.50
Recall	98.95
F1-Score	96.68

Table 3: Confusion Matrix for cybercrime Offence-Type Classification (Human vs. LLM)

Human \ LLM	**Type A**	**Type B**	**Type C**	**Type D**	**Type E**	**Unknown**
Type A	**32**	3	0	0	2	12
Type B	0	**34**	0	0	0	0
Type C	0	2	**40**	0	0	0
Type D	0	0	0	**38**	0	0
Type E	1	1	0	0	**36**	2
Unknown	7	0	0	2	2	**5**

to an actor operating the infrastructure IP. We consider that IPs associated with few domains are more likely to reflect dedicated or narrowly used infrastructure, whereas IPs associated with very large numbers of domains are more likely to reflect multi-tenant environments in which the observed web content may not indicate strong control of the underlying server by the same actor. Based on the empirical distribution (Figure 4), we discard cases where the IP is associated with **more than 30 domains**.

For cases discovered via domain crawling where packet captures were not used to confirm the contacted IP at collection time, we requery DNSDB and retain only domains that resolve to exactly one IP within the relevant time window (i.e., domain-to-IP cardinality = 1). The resulting set forms the cases analyzed in Section 6.

5. LLM Evaluation (Manual Validation)

This section evaluates the reliability of LLM-generated labels used in our pipeline. Specifically, we evaluate the screenshot-level classification outputs of gpt-5.2 for (i) cybercrime-related relevance detection (Yes/No) and (ii) cybercrime offence-type categorization (Type A–E/Unknown). We do not evaluate case construction or passive-DNS filtering here, as those are performed in later stages of the workflow.

We build ground truth via manual review of screenshots by the author. To reduce bias during labeling, sampled screenshots are anonymized and randomly shuffled prior to review. Manual labels are assigned based solely on screenshot content.

We randomly sample 200 screenshots predicted as Yes and 200 screenshots predicted as No. After anonymization and shuffling, we manually label each screenshot as Yes or No, and compute standard binary classification metrics.

For fine-grained categorization, we sample up to 40 screenshots from each predicted class (Type A–E and Unknown); when a class contains fewer than 40 samples, we include all available samples. After anonymization and shuffling, we manually assign cybercrime offence types under the same taxonomy, and report weighted multi-class metrics and a confusion matrix.

For the binary relevance task, the LLM achieves an accuracy of **96.75%**, precision **94.50%**, recall **98.95%**, and F1-score **96.68%** on the manually labeled 400-screenshot sample (Tables 1 and 2). For the cybercrime offence-type categorization task, the LLM achieves a weighted accuracy of **84.47%** (weighted precision **85.12%**, weighted recall **84.47%**, weighted F1-score **84.50%**) on the manually labeled stratified sample (Tables 3 and 4).

Table 4: Performance Metrics for cybercrime Offence-Type Classification (Types A–E and Unknown)

Metric	**Score (%)**
Accuracy	84.47
Precision (Weighted)	85.12
Recall (Weighted)	84.47
F1-Score (Weighted)	84.50

6. Results

This section reports the empirical results of our longitudinal measurement of web-exposed cybercrime-related content on IoT botnet infrastructure. We first summarize the dataset produced by our pipeline and LLM labeling outcomes (Section 6.1). We then present the final set of cases and characterize their distributions across infrastructure roles, access modes, and cybercrime offence types (Section 6.2). Finally, we provide representative examples to illustrate the screenshot-level evidence used for cybercrime offence-type categorization (Section 6.3).

6.1 Dataset Overview and LLM Label Statistics

Our measurement period spans Aug. 1, 2024 to Aug. 24, 2025. Starting from seed infrastructure IPs (Section 2), we expand associated domains via passive DNS (DNSDB) using the ±20-day overlap filter described in Section 4.1. We then crawl both seed IPs and DNSDB-derived domains via HTTP/HTTPS and collect rendered webpage screenshots.

During the period, we collected **43,406** screenshots. Because targets are revisited over time and that IPs/domains can reappear across daily observation windows, this raw corpus naturally includes repeated captures of identical or near-identical web content.

We apply the gpt-5.2-based visual classifier described in Section 4.3 to all collected screenshots. The classifier labels each screenshot as cybercrime-related (Yes) or benign/irrelevant (No), and for Yes pages assigns an cybercrime offence type (Type A–E or Unknown) under the taxonomy in [13]. In total, the LLM identifies **1,509** screenshots as cybercrime-related. Among the 43,406 collected screenshots, only 1,509 are labeled as cybercrime-related, corresponding to 3.48% of the raw dataset. This low proportion suggests that web-exposed cybercrime-related content is not pervasive across the overall set of observed infrastructure endpoints during our measurement period; rather, it appears on a relatively small subset of the infrastructure we monitored.

Table 5 summarizes dataset scale and LLM Yes counts by infrastructure role. Table 6 further breaks down the 1,509 cybercrime-related screenshots by cybercrime offence type and infrastructure role. Download-server screenshots account for the majority across all types, consistent with their larger exposure as web-facing endpoints in our dataset.

6.2 Web-Exposed Cybercrime Cases

From the 1,509 screenshots labeled as cybercrime-related, we manually group screenshots with identical webpage content into cases (i.e., one case corresponds to one distinct web content instance) and exclude IoT device login pages. This process yields **55** candidate cases. Table 7 shows the role distribution of these candidate cases.

We then apply the passive-DNS-based filters described in Section 4.4. Specifically, we use the number of domains associated

Table 5: Dataset overview and LLM label statistics by infrastructure role.

Role	Seeds (uniq IPs)	Domains (uniq)	Screenshots	Yes (LLM)
Download server	9,296	32,133	39,741	1,345
Loader	548	3,622	2,562	135
C2 server	88	874	1,103	29
Total	9,932	36,629	43,406	1,509

Table 6: LLM-labeled cybercrime-related screenshots (N=1,509): distribution by cybercrime offence type and infrastructure role.

Cybercrime offence type	C2	Download	Loader
Type A	10	658	97
Type B	0	70	0
Type C	0	157	9
Type D	16	161	8
Type E	3	281	20
Unknown	0	18	1
Total	29	1,345	135

Table 7: Candidate cases after manual de-duplication and excluding IoT login pages (N=55).

Role	Cases
C2 server	2
Download server	46
Loader	7
Total	55

Table 8: Cases (N=32): distribution by cybercrime offence type and infrastructure role.

Cybercrime offence type	C2	Download	Loader	Total
Type A	0	4	2	6
Type B	0	4	0	4
Type C	0	9	0	9
Type D	1	3	1	5
Type E	1	6	1	8
Total	2	26	4	32

with an IP (within the corresponding time window) as a proxy for the strength of attributability of the observed web content to the infrastructure IP, and discard cases where the involved IP is associated with more than 30 domains. In addition, for domain-derived cases where packet captures were not used to confirm the contacted IP at crawling time, we retain only domains that resolve to exactly one IP within the relevant time window. After filtering, we obtain **32** cases for analysis.

Among the 32 cases, download servers dominate (26/32), while loaders and C2 servers account for 4/32 and 2/32 cases, respectively. At the case level, web-exposed cybercrime-related content is observed much more frequently on download-server infrastructure than on loaders or C2 servers, indicating a clear role asymmetry in where such content appears. Table 8 summarizes the 32 cases by cybercrime offence type and infrastructure role. The cases span all five cybercrime offence types in [13], showing that the web-exposed content observed on IoT botnet infrastructure is not limited to a single category of cybercrime activity. We observe cases via three access modes: crawling the IP directly, crawling a DNSDB-derived domain, or both. In our final set of 32 cases, 17 are observed via IP-only crawling, 5 via domain-only crawling, and 10 via both IP and domain access, suggesting that relying solely on IP probing would miss a non-trivial fraction of relevant instances.

Furthermore, during case-level analysis, we observe that the two C2-related cases also appear among the download-server cases in the same time. Similarly, two of the four loader-related cases also overlap with two download-server cases. For these overlapping cases, the involved IP addresses are identical when the web content is identical. This indicates that some infrastructure IPs in our seed sets are used in multiple botnet-infrastructure roles (e.g., simultaneously appearing as C2/download/loader endpoints) while also exposing web content corresponding to other types of cybercrime activities.

6.3 Representative Examples (Type C and Type D)

We next present two representative cases (one Type C and one Type D) to illustrate the screenshot-level evidence supporting our cybercrime offence-type categorization. Figure 5 shows redacted screenshots for both cases. Since the original screenshots are vertically long, we show only the upper portion of each page here for readability.

Type C example (gambling page): This case was observed on 2025-01-22 and consists of 2 screenshots. It was discovered via IP-based crawling. In passive DNS, the involved IP is associated with 0 domains within the relevant time window. The webpage presents an online gambling-style service interface (e.g., prominent registration/login entry points and betting/lottery-like content). Since the primary content is gambling-related, we categorize this case as Type C (content-related offences) following the taxonomy in [13].

Type D example (copyright-related content portal): This case was observed on 2025-06-30, 2025-07-02, 2025-07-07, and 2025-07-08, and consists of 24 screenshots. It was reachable via both IP and domain access. Passive DNS indicates that the involved IP is associated with 10 domains within the relevant time window, and the corresponding domain is associated with exactly one IP, satisfying our case-selection criteria. The webpage provides a portal-like interface for accessing media content (e.g., catalog-style listings for movies/series and content access buttons/links), which is characteristic of copyright-infringing distribution sites. Accordingly, we categorize this case as Type D (offences related to infringements of copyright and related rights).

7. Discussion

This paper measures web-exposed cybercrime-related content on IoT botnet infrastructure using a longitudinal pipeline that combines passive DNS enrichment, daily web crawling, and LLM-based visual labeling with manual validation. From 43,406 collected screenshots, we obtain 1,509 cybercrime-related screenshots and finally derive 32 cases after manual case grouping and conservative filtering. These cases span cybercrime offence Types A–E and are dominated by download-server infrastructure. We also observe overlaps across infrastructure roles (e.g., the same IP/content appearing in multiple role-specific seed sets). Together, these results show that web-exposed cybercrime-related content can be observed on IoT botnet infrastructure, although it is not pervasive in the raw dataset.

We next discuss implications and limitations of our measurement and outline directions for improvement.

7.1 Implications

Botnet infrastructure is typically tracked via network telemetry, malware analysis, and infrastructure-level indicators. Our results suggest that web-exposed content can provide additional context about how infrastructure endpoints are used, and may re-

Fig. 5: Representative cases used to illustrate screenshot-level evidence for cybercrime offence-type categorization. Sensitive details are redacted.

veal cybercrime-related activities that are not apparent from botnet telemetry alone. In particular, we find that a non-trivial set of cases exposes cybercrime-related web content across multiple cybercrime offence types, indicating heterogeneous usage of infrastructure endpoints.

At the case level, most instances are observed on download servers rather than loaders or C2 servers. While we do not claim causality, this role asymmetry suggests that download-server endpoints may be a practical starting point for web-content-based reconnaissance when investigating IoT botnet infrastructure, because they more frequently expose web-facing services in our observations.

We observe that some cases and IP addresses appear across multiple infrastructure roles (e.g., the same IP appearing in C2- and download-related seed sets while exposing the same web content). This suggests that treating botnet infrastructure roles as strictly separated may miss important overlaps, and that consolidating infrastructure views across roles can improve understanding of how a given endpoint is operated.

7.2 Limitations

A limitation of our study is that our crawler does not capture packets during web access. Therefore, for domain-derived cases we cannot directly verify which IP address was contacted at collection time (e.g., under redirects or DNS changes). We mitigate this by requerying passive DNS and retaining only domains associated with exactly one IP in the relevant time window, but this remains an indirect proxy and may introduce false negatives or residual ambiguity.

Our measurement is scoped to web-exposed content over HTTP/HTTPS and is limited to the first page reached per navigation attempt. As a result, we may miss content behind deeper paths, login flows, or non-web services. Passive DNS is also incomplete and time-varying, so some IP–domain associations may be missing or fall outside our observation window.

We further use the number of domains associated with an IP as a proxy for attributability; this heuristic is conservative but does not prove control and may discard cases on multi-tenant infrastructure. Finally, while LLM labels are validated and case grouping is performed carefully, both can still contain errors, and our crawling configuration may trigger defensive interstitials or be affected by endpoint availability.

8. Related Work

IoT botnets have been widely studied due to their scale and impact, and a substantial body of work has characterized their infrastructure and operational dynamics. Our work relates to three research threads: (i) measurement and characterization of IoT botnet infrastructure, (ii) active probing of C2 servers via milker-style tools, and (iii) linking botnet infrastructure to broader cybercrime activities using web and DNS signals.

8.1 IoT Botnet Infrastructure Measurement

Antonakakis *et al.* [1] analyzed the growth, composition, and evolution of the Mirai botnet and identified distinct C2 clusters with disjoint infrastructure. Bastos *et al.* [2] characterized Bashlite and Mirai C2 servers, reporting that many C2 servers are short-lived and that a large fraction are hosted by cloud providers. Tanabe *et al.* [12] studied the relationships among IoT malware binaries, download servers, and C2 servers, revealing how infrastructure evolves over time and how attackers source IP addresses.

8.2 Active C2 Probing via Milker-Style Tools

Active interaction with botnet C2 servers has been used to reveal live behavior beyond what can be inferred from passive observations. Davanian *et al.* [3] developed a milker tool to interact with IoT malware C2 servers and disclose their activities in the wild.

8.3 Linking Infrastructure to Cybercrime Activities

Our prior work [7] explored the possibility of linking IoT botnet-related infrastructure to cybercrime offerings by combining passive DNS enrichment and LLM-assisted filtering. That study demonstrated that some infrastructure associated with IoT botnets is related to cybercrime offerings, but it had several limitations: it focused on a specific subset of activities (primarily DDoS-for-hire/CaaS-related cases), relied on domain-oriented signals, and did not perform a year-long longitudinal measurement of web-exposed content on infrastructure endpoints.

These studies provide important understanding of botnet infrastructure roles and lifecycle properties, but they primarily focus on botnet operations themselves rather than measuring what web-exposed cybercrime-related content may be co-located on such infrastructure. In contrast, the present work broadens the scope to multiple cybercrime offence types by directly measuring web-exposed cybercrime-related content via longitudinal crawling and screenshot-based analysis. We collect rendered webpage screenshots over one year and use an LLM-based visual classifier grounded in an established cybercrime offence taxonomy (Types A–E), followed by conservative passive-DNS-based filtering to derive cases. This enables a content-centric and temporally grounded characterization of what cybercrime-related webpages are exposed on IoT botnet infrastructure beyond botnet-related functions.

9. Ethical Considerations

This section describes the ethical considerations taken during data collection and analysis.

Our web crawling was strictly observational and limited in scope. We accessed only publicly available home pages via standard HTTP/HTTPS requests to collect high-level information such as rendered screenshots and HTML source code. We did not perform depth crawling, follow secondary links, submit forms, provide credentials, or interact with any services beyond loading the initial landing page.

Although some home pages may incidentally display gambling-related, copyright-infringing, or adult content, we did not download, stream, store, or otherwise interact with any media or services. Data collection was limited to passive observation necessary for academic analysis, and no actions were taken to exploit vulnerabilities, bypass

access controls, or trigger harmful behavior.

All collected data were handled solely for research purposes within a controlled environment. We do not redistribute collected content, and when reporting results, we avoid disclosing sensitive details that could facilitate misuse. Our study follows institutional guidance for ethical Internet measurement and is conducted with the goal of understanding and mitigating cybercrime.

10. Conclusion and Future Work

This paper studied web-exposed cybercrime-related content on IoT botnet infrastructure. Using a one-year longitudinal crawling campaign (Aug. 2024–Aug. 2025), we collected 43,406 rendered webpage screenshots from infrastructure IPs and their DNSDB-associated domains. With gpt-5.2-based visual classification (validated against manual annotations), we identified 1,509 screenshots as cybercrime-related, corresponding to 3.48% of the raw corpus, suggesting that such web-exposed cybercrime-related content is not pervasive in our observations. After manual case grouping by identical webpage content, excluding IoT device login pages, and applying conservative passive-DNS-based filtering, we obtained 32 cases. These cases span all five cybercrime offence types (Types A–E) and are dominated by download-server infrastructure (26/32), with fewer cases on loaders (4/32) and C2 servers (2/32). We also observed overlaps across infrastructure roles, where identical web content appears on the same IP that is present in multiple role-specific seed sets (e.g., C2/download/loader).

Several directions can strengthen and extend this study. First, web collection should be augmented with traffic capture or at least logging of resolved IPs and TLS/SNI metadata to directly confirm the contacted IP for domain-derived pages and reduce ambiguity caused by redirects, CDNs, or DNS changes. Second, coverage can be improved by extending beyond the first page reached using carefully bounded crawling depth (e.g., a small fixed link budget) while maintaining stable throughput. Third, additional corroborating signals (e.g., TLS certificate reuse, HTTP headers, and server fingerprints) may help link related content across roles and time without making attribution claims. Finally, future measurements can expand beyond HTTP/HTTPS to explore non-web services and ports, to better understand infrastructure reuse that is not web-exposed.

Acknowledgments: This research was achieved through commissioned research (JPJ012368C05201) by the National Institute of Information and Communications Technology (NICT), Japan. This study includes outcomes obtained from the commissioned research (JPJ012368C08101) by the National Institute of Information and Communications Technology (NICT), Japan.

References

[1] Manos Antonakakis, Tim April, Michael Bailey, Matt Bernhard, Elie Bursztein, Jaime Cochran, Zakir Durumeric, J Alex Halderman, Luca Invernizzi, Michalis Kallitsis, et al. Understanding the mirai botnet. In 26th USENIX security symposium (USENIX Security 17), pages 1093–1110, 2017.

[2] Gabriel Bastos, Artur Marzano, Osvaldo Fonseca, Elverton Fazzion, Cristine Hoepers, Klaus Steding-Jessen, Marcelo H.P.C. Chaves, Italo Cunha, Dorgival Guedes, and Wagner Meira Jr. Identifying and characterizing bashlite and mirai c2c servers. In ISCC 2019, Barcelona, Spain, 2019.

[3] Ali Davanian, Michail Faloutsos, and Martina Lindorfer. C2miner: Tricking iot malware into revealing live command & control servers. In Proceedings of the 19th ACM Asia Conference on Computer and Communications Security, pages 112–127, 2024.

[4] Yuki Endo, Kaichi Sameshima, Rui Tanabe, Katsunari Yoshioka, and Tsutomu Matsumoto. Analysis of iot botnet activities based on attack commands collected via botnet milker scripts. Computer Security Symposium 2023, pages 909–915, 2023.

[5] Yuki Endo, Rui Tanabe, Katsunari Yoshioka, and Tsutomu Matsumoto. Enhanced analysis of iot botnet activity by fusing honeypots, malware dynamic analysis, and cc observation. ICSS, 2024(12):1–6, 2024.

[6] Harm Griffioen and Christian Doerr. Examining mirai's battle over the internet of things. In Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, pages 743–756, 2020.

[7] Qingxin Mao, Yin Minn Pa Pa, Rui Tanabe, and Katsunari Yoshioka. Linking iot attacks to cybercrime-as-a-service offerings using llm and dns data. Technical report, IEICE, 2025. Technical report without peer review.

[8] OpenAI. https://openai.com/, 2025.

[9] YIN MINN PA PA, SHOGO SUZUKI, KATSUNARI YOSHIOKA, TSUTOMU MATSUMOTO, and CHRISTIAN ROSSOW TAKAHIRO KASAMA. Iotpot: A novel honeypot for revealing current iot threats. Journal of Information Processing, 24(3):522–533, 2016.

[10] Farsight Security. DNSDB. https://www.domaintools.com/products/farsight-dnsdb/, 2025.

[11] Selenium. https://www.selenium.dev/, 2025.

[12] Rui Tanabe, Tatsuya Tamai, Akira Fujita, Ryoichi Isawa, Katsunari Yoshioka, Tsutomu Matsumoto, Carlos Gañán, and Michel Van Eeten. Disposable botnets: examining the anatomy of iot botnet infrastructure. In Proceedings of the 15th International Conference on Availability, Reliability and Security, pages 1–10, 2020.

[13] George Tsakalidis and Kostas Vergidis. A systematic approach towards description and classification of cybercrime incidents. Technical report, University of Macedonia, July 2016. Manuscript submitted for review on July 4, 2016.

Appendix

Appendix: LLM Prompt

```
CONTEXT: SECURITY RESEARCH ONLY
This task is for cybersecurity research and threat intelligence purposes.
Analyzing these images is for detection and protection.

You are an expert Threat Intelligence Analyst specializing in visual identification of
       cybercrime infrastructure.
Your task is to analyze a webpage screenshot and classify it according to the taxonomy
       below.

CATEGORY RULE:
- If the page is possibly related to cybercrime / cyber-enabled crime, output "Yes".
- Output "No" ONLY if the page is clearly unrelated / benign OR blank/404/default server
       page.
- If judgement is "Yes": choose the closest Type A-E if possible; if truly not fitting A
       -E, use "Unknown".
- If judgement is "No": category MUST be "Unknown".
- If the image is difficult to interpret, please extract the text within it and use it
       to support your decision.
- If the page is a domain parking / Related searches / generic ad-landing page with no
       login/verification/credential capture, classify as "No".
- If the screenshot clearly shows the brand/domain "playit.gg" (including "PLAYIT.GG"),
       classify as "No".
- If the page is a security block / WAF / anti-bot / DDoS protection page (e.g., Your IP
       has been blocked, Access denied, too many requests, Cloudflare/ParsPack
       protection), classify as "No" unless there is additional clear evidence of
       illicit activity.

INPUT: Screenshot image(visual evidence).

TAXONOMY: Tsakalidis & Vergidis (Types A-E); definitions omitted for brevity.

OUTPUT FORMAT (STRICT JSON):
{{
  "judgement": "Yes" or "No",
  "category": "Type A" ... "Type E" or "Unknown",
  "reason": "Concise evidence-based explanation citing visual indicators."
}}
```