A Large Scale Study and Classification of VirusTotal Reports on Phishing and Malware URLs

EUIJIN CHOO*, University of Alberta, Canada
MOHAMED NABEEL, Palo Alto Networks, USA
DOOWON KIM, University of Tennessee, Knoxville, USA
RAVINDU DE SILVA, SCoRe Lab, Sri Lanka
TING YU and ISSA KHALIL, Qatar Computing Research Institute, Qatar

VirusTotal (VT) is a widely used scanning service for researchers and practitioners to label malicious entities and predict new security threats. Unfortunately, it is little known to the end-users how VT URL scanners decide on the maliciousness of entities and the attack types they are involved in (e.g., phishing or malware-hosting websites). In this paper, we conduct a systematic comparative study on VT URL scanners' behavior for different attack types of malicious URLs, in terms of 1) detection specialties, 2) stability, 3) correlations between scanners, and 4) lead/lag behaviors. Our findings highlight that the VT scanners commonly disagree with each other on their detection and attack type classification, leading to challenges in ascertaining the maliciousness of a URL and taking prompt mitigation actions according to different attack types. This motivates us to present a new highly accurate classifier that helps correctly identify the attack types of malicious URLs at the early stage. This in turn assists practitioners in performing better threat aggregation and choosing proper mitigation actions for different attack types.

CCS Concepts: • **Security and privacy → Network security**.

Additional Key Words and Phrases: VirusTotal Measurement, Malicious URLs, Attack Type Classifier

ACM Reference Format:
Euijin Choo, Mohamed Nabeel, Doowon Kim, Ravindu De Silva, Ting Yu, and Issa Khalil. 2023. A Large Scale Study and Classification of VirusTotal Reports on Phishing and Malware URLs. *Proc. ACM Meas. Anal. Comput. Syst.* 7, 3, Article 59 (December 2023), 26 pages. https://doi.org/10.1145/3626790

1 INTRODUCTION

VirusTotal (VT) is one of the most popular and influential online scanning services to detect and categorize malicious entities, including binaries and URLs. VT aggregates scanning results (i.e., a detection result and the category of malicious entity such as phishing and malware-hosting) from up to 95 various detection scanners and provides the security research community (academia and industry) with the aggregated results. These results are heavily utilized to label malicious entities and predict new security threats [10, 11, 24, 33, 44, 46, 49].

One of the best strategies for the security community to mitigate attacks and/or remedial actions is to promptly and accurately identify the types of attacks [10]. Unfortunately, the scanners in the VT perform as a black box. In other words, it is little known to the security community (i.e., the VT end-users) how the VT scanners decide on the maliciousness of entities and what specific attack types are (e.g., phishing or malware-hosting). Moreover, it commonly occurs that the VT scanners disagree

*The corresponding author is Euijin Choo (euijin@ualberta.ca)

2476-1249/2023/12-ART59 $15.00
https://doi.org/10.1145/3626790

Table 1. Key differences between our work and prior work analyzing VT. Only our work provides (1) a large-scale measurement for various types of malicious data, (2) insights both on scanners' detection and attack type labels, and (3) an attack type classifier to help choose proper mitigation actions for different attack types.

Work	Mal. Data Type[1]	Data Diversity[2]	# Mal. Data[3]	VT Labels Used[4]	Measurement[5]	Classifier[6]
Alex Kantchelian et al. [18]	Windows Malware	◗	-	◗	◗	Benign or Malicious
Zhu et al. [56]	Obfuscated PE malware	◗	120	◗	◗	✖
Peng et al. [35]	IRS/Paypal Phishing	◗	36	◗	◗	✖
Salem et al. [42]	Android Malware	◗	24 K	◗	◗	Benign or Malicious
Bouwman et al. [7]	Covid-19 Domains	◗	188 K	◗	◗	✖
van Liebergen et al. [50]	Malware	◗	-	◗	◗	✖
Thirumuruganathan et al. [47]	Various Types	●	2.7 M	◗	✖	Benign or Malicious
Our work	Various Types	●	1,577 M	●	●	Attack type

[1]: Malicious data type used in each paper. [2]: Diversity of the data type. [3]: The malicious dataset size. [4]: VT labels studied and utilized in each paper.
[5]: Whether any measurement study is done in the paper. [6]: Proposed classifier. –: Unknown, not specified the number of malicious data
✖: [5] no measurement, [6] no classifier ◗: [2] only a specific data type, [4] scanners' detection labels only, [5] scanners' detection labels only
●: [2] various data types [4] scanners' detection and attack type labels, [5] scanners' detection and attack type labels

with each other on their detection labels and attack type classification [35, 46, 56]. Such uncertainty about malicious entity classifications can make it challenging for the security community (i.e., the VT end-users) to have better security decisions and accurate predictions of new threats. This is because it is a typical practice in the industry to assign different severity levels for different attack types, prioritize actions based on the severity, and choose proper mitigating actions for different attack types [34]. For example, when a website is infected with malware, an initial remediation action is to check for file integrity and malicious code injections; when a website is compromised with a phishing page, an initial task is to identify pages or folders that are created recently and contain login/payment forms.

While prior work has attempted to address the challenges, they are limited in scale and diversity [7, 18, 35, 41, 42, 56]. Table 1 summarizes a comparison of our work with the prior work analyzing VT reports. Specifically, they often focus on the detection trends for specific types of entities: malware binaries [18, 41, 42, 56], IRS/Paypal phishing URLs [35], or COVID-19 related malicious domains [7], and there has been limited attention to identifying the attack types. Previous work also focuses more on malware binaries than malicious URLs. This is because analyzing URLs is significantly more challenging than analyzing malware, as URLs exhibit dynamic behaviors. For example, even though a specific URL address does not change, its purposes and contents could dynamically change over time (e.g., compromised and cleaned), which results in more challenges in analyzing malicious URLs than malware binaries. Consequently, it necessitates a longitudinal study of VT scanners specifically for malicious URLs to answer four key research questions: 1) *How do VT scanners behave for URLs?* 2) *What are the limitations of current detection and attack type labeling approaches using VT?* 3) *Do VT scanners behave differently for URLs of different attack types?* 4) *How to identify attack types of URLs from VT reports and what is the realistic distribution of different URL types?*

To answer these research questions, we analyze all the VT URL reports (of 1.58 billion distinct URLs) generated from July 2019 to Jan. 2022 (30 months). We study various characteristics in URL scan reports, such as the attack types of URLs (Section 4.1), scanners' detection specialty (Section 4.2), the stability (Section 4.3), and correlation of individual scanners (Section 4.4), and lead/lag behavior (Section 4.5). To the best of our knowledge, this is the first work of a large-scale, in-depth analysis of VT URL scan reports and the first systematic comparative study for different attack types of malicious URLs.

From our large-scale measurement study, we make the following useful **observations**. 1) Conflicting attack type labels are common in individual scanners temporally and across scanners; specifically, the labels of the phishing URLs disagree more than the ones of malware URLs. 2) Scanners specialize in different attack types, and no scanner performs well for all types of URLs. 3) The set and level of highly correlated scanners differ depending on attack types. 4) Scanners detecting phishing URLs are more correlated than those detecting malware URLs. 5) Fewer scanners correlate in their attack type labels. 6) Lead/lag relationships exist, and the set of leaders differs

depending on the attack types. 7) Using our highly accurate attack type classifier, we show malware URLs consistently dominate phishing URLs observed in VT over time. We provide an overview of our key findings and takeaways in Section 2.

Our **contributions** are summarized as the following:

- We collect large-scale VirusTotal reports for malicious URLs for 30 months and conduct a longitudinal study. We summarize and highlight our key findings in Section 2.
- We characterize VirusTotal scanners with an emphasis on scanners' specialty, stability, attack type classification, correlations, and lead/lag relationships with regard to different attack types.
- We propose an attack type classifier that takes scanners' correlations and specialties to identify the attack type of malicious URLs. Our approach achieves high accuracy compared to the baseline approaches. We apply the trained models to study the attack types reported in VirusTotal and show the realistic distribution of malware and phishing URLs.
- We provide practical suggestions using VirusTotal to compile better malicious ground truth considering attack types and characteristics of scanners.

2 OUR KEY FINDINGS

Before presenting the details of our study, we summarize our key observations in this section.

Observation 1: Conflicting Attack Type Labels. We show that conflicting attack type labels for a given URL are common, and it is due to two cases: individual VT scanners' temporal conflicts and cross-scanner conflicts (Section 4.3 and Section 4.1), although contents located in the URL did not change. While previous studies discussed the conflicting detection labels [35, 56], our finding further provides insights on attack type labels. We generally observe that phishing URLs have more conflicting labels than malware URLs. Given such conflicts, it imposes challenges for practitioners to choose the proper mitigation actions that depend on attack types. We thus emphasize the importance of identifying attack types to collect reliable corresponding ground truth. In line with it, we propose a method to quickly identify attack types given the conflicting labels (Section 5). Using the correctly identified attack type labels, we show that malware URLs dominate more than phishing URLs in VT. We suggest that practitioners utilize our classifier to quickly build reliable ground truth and choose the proper actions for different attack types.

Observation 2: Scanners' Specialty and Detection Performance. We confirm that scanners specialize in different attack types, and no scanner performs well for all types of URLs. Almost half of the scanners never detect specific types of attacks due to their specialties. We observe that almost half of the scanners never detect specific types of attacks due to their specialties (Section 4.2). While previous studies focused on each scanner's detection accuracy on one type of entity (e.g., obfuscated malware files [56], android malware [42], IRS/PayPal phishing domains [35], and COVID-19 related threats [7]), we focus on how scanners perform for different attack types. For example, we observe AegisLab WebGuard performs well in detecting malware URLs, whereas Bitdefender performs well in detecting phishing URLs. Also, similar to previous studies on malware files [18, 56], we observe that scanners often work poorly in the early reports when URLs first appear in VT and their label stabilizes over time. However, we observe that scanners reach the maximum F-1 score relatively early for URLs (near the 5th day in our dataset since their first appearance in VT), compared to $2 \sim 4$ weeks reported in previous studies on malware files or IRS/Paypal phishing URLs [18, 35]. This supports our suggestion that an independent analysis for URL reports is needed to derive the optimal time period for groundtruth collection. Finally, we observe that the performance of scanners detecting phishing URLs is less consistent than those detecting malware URLs. We recommend that practitioners collect the URL groundtruth set earlier than files (e.g., around the 5th day since their first appearance), while practitioners utilize our analysis and classifier to mitigate the effect

of unreliable and conflicting VT results. We suggest that practitioners choose a higher threshold for phishing URLs to mitigate the effect of less consistent performance of scanners.

Observation 3: Scanners' Correlation on Detection and Attack Types Classification. Some scanners are highly correlated in terms of their detection, attack type labels, temporal label similarity, and trends of label patterns (Section 4.4). Previous work focused only on the detection correlations, including label flipping patterns [56] or overlapping sets between blocklists [13]. However, detection correlation does not necessarily mean that scanners are indeed correlated if their attack labels are different. Instead, we measure the correlation in various aspects with regard to attack types. Specifically, we observe that the set and number of highly correlated scanners are different depending on attack types. Also, we observe that scanners detecting phishing URLs are more correlated than those detecting malware URLs. Finally, fewer scanners correlate in terms of their labels on attack types than detection itself. Concretely, 27% of scanners that detect phishing URLs and 5% of scanners that detect malware URLs have a high correlation on co-detected URLs; while only 3% of scanners that detect phishing URLs and 3% of scanners that detect malware URLs have a high correlation on attack label assignments for given URLs. We suggest that, instead of directly using positive counts, practitioners utilize our analysis of scanner correlations for different attack types and proportionally weigh less the counts from correlated scanners to obtain better ground truth.

Observation 4. Lead/Lag Relationships among Scanners for Each Attack Type. Lead/lag relationships exist among highly correlated scanners for each attack type (Section 4.5). For example, `Webroot` and `alphaMountain.ai` have a high correlation for phishing URLs, while Webroot always detects earlier than alphaMountain.ai. Meanwhile, the set of leaders is different depending on the attack types (e.g., the top 5 leaders are Sophos, OpenPhish, PhishLabs, Netcraft, and Segasec for phishing URLs; Kaspersky, Fortinet, Webroot, Sophos, Segasec for malware URLs). Early detection is important due to the short-liveness nature of malicious URLs. We recommend that researchers utilize our analysis of specialty, correlation, and lead/lag relationships to choose a set of scanners to form groundtruth of URLs with a specific attack type.

3 DATA COLLECTION AND PRELIMINARIES

This section describes our dataset and the terminologies used throughout the paper. We collect two types of datasets: (1) VirusTotal (VT) Feed and (2) Ground Truth URL and Corresponding VT Report Dataset. We collect VT Feed to study the characteristics of VT scan reports and individual VT scanners' behaviors for URLs in the wild; Ground Truth set to study those for URLs with regard to different attack types. Table 2 summarizes our dataset.

3.1 VirusTotal (VT) Feed

VirusTotal (VT) is an aggregation service that interacts with 95 scanners and aggregates their scanning results (i.e., detection and attack type labels) for URLs queried by users. VT provides a feed of scan reports for all URLs queried by all users to premium VT service subscribers. We collect the scan reports for all URLs submitted to VT from Jul. 2019 to Jan. 2022 (30 months) through the subscription. Each scan report contains the aggregated results of up to 95 URL scanners listed in Appendix A[1]. 5 million unique scan reports, including benign and malicious ones, are generated daily on average.

VT Report Fields. We are interested in the following fields in each scan report: '*url*', '*scan_date*', '*first_seen*', '*scan_id*', '*positives*', '*Response content SHA-256*', and '*scans*'. The field '*scans*' contains the name of the scanners, and two subfields: '*detected*' (whether a URL is malicious or not according to the scanner) and '*result*' (the attack type indicated by the scanner, such as malicious, phishing, malware, suspicious, mining, not recommended, and spam sites). The '*scan_id*' represents a unique

[1]Each URL is not always scanned by the same set of scanners.

Table 2. Summary of our collected dataset. (For all URLs, we use all VirusTotal scan reports between 07/2019 - 01/2022)

	Type		# URLs*	URL Collection Period
VT	General Feed		1,577 M	Jul. 2019 – Jan. 2022 (30 months)
	Fresh Feed		224 M	
Ground Truth	Manual GT	Benign	421	Aug. 30 2020 – Sep. 3 2020 & Oct. 10, 2021
		Malicious	352	
	Phishing	APWG	9,186	Apr. 20, 2021
		SiteAdivsor	6,237	Mar. 1, 2021
		SiteAdivsor	763	Jul. 30, 2021
	Malware	APWG	223	Apr. 20, 2021
		SiteAdivsor	5,823	Mar. 1, 2021
		SiteAdivsor	658	Jul. 30, 2021

*Distinct URLs. Each distinct URL has multiple scan reports over our study period.

scan ID number. When a URL is submitted to VT for scanning, VT checks if the URL has already been scanned before. If a URL has been scanned before, VT either 1) simply returns the previous scan report with the same '*scan_id*' or 2) rescans the URL and produces an updated scan report with a new '*scan_id*'. We observe that VT returns previous reports when the URLs were recently scanned unless the user explicitly requests to rescan. Because multiple duplicated reports with the same '*scan_id*' can lead to biased results, we only extract distinct scan reports with a unique '*scan_id*'. The field '*positives*' is the number of scanners that detect a particular URL as malicious. The field '*Response content SHA-256*' is the SHA-256 hash of contents (such as html or files) located in the URL at the scan date. An example VT report is in Appendix B and the detailed information for fields in a VT report is in [51, 52].

VT General Feed. VT Feed has approximately 1 or 2 million distinct scan reports each day where a URL is marked as malicious by at least one scanner. As we focus on malicious URLs in this study, we filter out the URLs that have been never marked as malicious by any VT scanners during our study period, which gives us 1,577 million (1.57B) URLs along with their scan reports over time.

VT Fresh Feed. To better understand how VT reacts to URLs over time [56], we further extract only VT *fresh* URLs from VT General Feed that are first observed and scanned during our observation period, which is called *VT Fresh Feed*. In other words, we collect only URLs whose *first_seen* is later than our first observation time ($first\_seen \geq$ Jul. 1st, 2019). Finally, we obtain 224 million URLs along with their scan reports over time.

3.2 Ground Truth URL and Their VT Report Dataset Collection

To analyze VT scanners' behavior with regard to different attack types, we first collect groundtruth URLs (Section 3.2.1) and corresponding periodic VT reports (Section 3.2.2).

3.2.1 Ground Truth URL Collection

We build 3 groundtruth URL datasets as follows. We first build *Manual GT URLs* by manually labeling URLs sampled from VT Fresh Feed (Section 3.2.1 (1)). To avoid bias on URL sets and evaluations, we additionally collect and manually label malicious URLs from two public intelligence sources that are not included in the list of VT scanners: Anti-Phishing Working Group (APWG) (Section 3.2.1 (2)) and SiteAdvisor (Section 3.2.1 (3)).

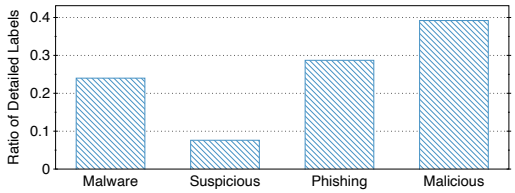


Fig. 1. The ratio of top 4 detailed labels for VT Fresh.

Figure 1 presents the ratio of the top 4 detailed labels over the total number of scan results for VT Fresh. The figure shows that the non-generic attack types of most malicious URLs in VT are either phishing or malware. We thus focus on phishing and malware ground truth.

(1) **Manually Labeled URLs (Manual GT URLs).** We collect Fresh URLs first scanned at the groundtruth collection date, then choose the sample URLs using a stratified sampling-based approach [5, 19] that helps better reflect the realistic data distribution. Then, domain experts *manually* labeled the URLs as follows. The experts label a URL as malicious if it has any malware signals (e.g., malware is hosted in the URL) or phishing signals (e.g., a squatting domain [30], or mimicking popular websites – for example, malicious.com with the login image of paypal.com [35]). A URL is considered benign when the URL has neither malware nor phishing signals, and has been in operation for more than 3 months [46] as malicious URLs are highly unlikely to survive more than 3 months [20].

Following previous studies [46, 47], all URLs are labeled by two experts, and URLs with conflicting labels are excluded for better confidence in labeling quality. The experts repeatedly checked the same set of URLs for 3 days to check content changes over time, then we continue to automatically check the content changes using the hash contents collected from VT. The detailed process is described in Appendix C. 773 URLs (Benign: 421, Malicious: 352) are successfully labelled after filtering URLs that have conflicts between two experts, whose contents have changed over time, or that have neither malicious nor benign signals. We use the benign URLs in this dataset as benign ground truth for our analyses.

(2) **Anti-Phishing Working Group (APWG) URLs.** APWG is a community-based service where the URLs are labeled by domain experts from multiple institutions [2]. We collect the latest 10K URLs from APWG and filter out invalid URLs (e.g., malformed URLs) and non-fresh URLs (i.e., the URL's first appearance timestamp in VT (*first-seen*) is older than our data collection period). Attack types of URLs are manually labelled following the same process as our Manual GT URL collection (Section 3.2.1 (1)). This results in 9K phishing and 223 malware URLs.

(3) **McAfee SiteAdvisor URLs.** McAfee SiteAdvisor (SA) is a service providing reputation reports for URLs [27]. In addition to detailed comments, SA reports include an attack category and one of the four risk levels: *unverified*, *low*, *medium*, and *high-risk*. We employ detailed SA threat reports assisting in manual labeling the attack types of URLs along with the same rubric in Section 3.2.1 (1). Concretely, we collect random samples of URLs from VT Fresh Feed to manually label their attack types with guidance from SA. As being interested in malicious URLs, we choose URLs having at least one phishing or malware label in VT and SA *medium* and *high-risk*. This results in 7K phishing and 6.5K malware URLs altogether. The data collection is split between two timestamps, where 90% of them are collected in Mar. 2021 and the remaining 10% in Jul. 2021.

3.2.2 VT Reports Collection for Ground Truth URLs

In practice, it is crucial to detect short-lived malicious URLs as early as possible for their threats to be quickly contained. To fully understand the behaviors of each scanner and URLs, we track scan reports from the very first appearance of URLs in VT and study scanners' behavior over time. To do so, after collecting the groundtruth URL datasets, we take two approaches: 1) submit the URLs to

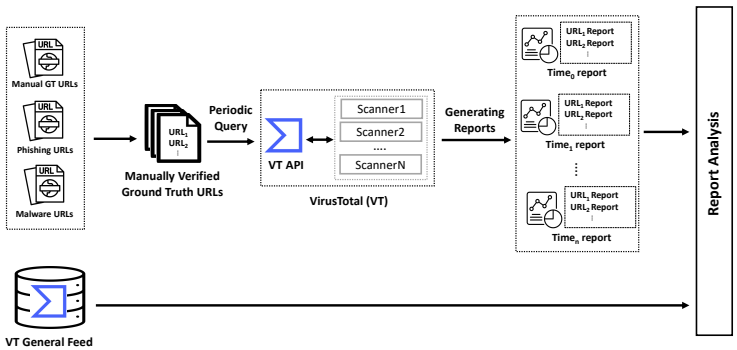


Fig. 2. Virustotal reports collection workflow

VT and request to rescan them periodically (**prospective study**); and 2) conduct a **retrospective study** to extract reports for the URLs from VT General Feed. Figure 2 illustrates the workflow of our data collection.

A few factors need to be considered when selecting the proper time granularity for building the periodic reports. *First*, the status of malicious URLs changes rapidly. They could be taken down after attacks [11], cleaned after being compromised [46], or re-registered after take-down to reuse for new attacks [53]. Recent research suggests that only a few malicious URLs have a lifetime of more than a month [20], while most malicious URLs have a few days or even a few hours [20, 31]. *Second*, it has been shown that even though a scanner may update its malicious URL list shortly after detection of new malicious URLs [32], VT does not necessarily promptly update the scanner's result in its database [35]. We collect periodic reports daily and hourly based on these observations and our empirical analysis.

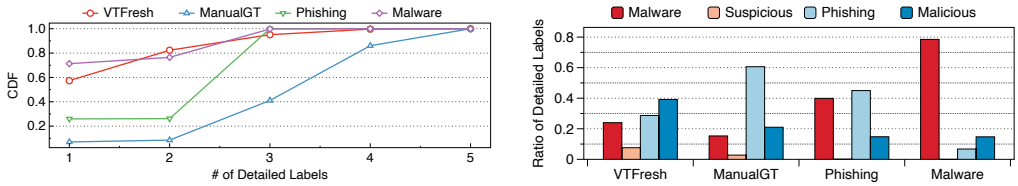
3.3 Terminologies and Notations

Note that each distinct URL may have multiple scan reports over our study period. We sort the scan reports by timestamp (i.e., *scan_date*) for each URL and represent the data per scanner as a time series (i.e., a sequence of chronologically ordered data points). Each data point corresponds to the scanner's label of the URL for a given time frame. To see the long-term trend of scanners and URLs, we present results using the daily time granularity throughout the paper.

Labels. We use two types of labels for each time frame: a binary label (*detected* field in VT reports) and a detailed label (*result* field in VT reports). A binary label indicates whether or not a scanner detects a URL as malicious, encoded as 1 or 0; a detailed label means an attack type label assigned by scanners such as "malware site" and "phishing site".

If there are multiple scan reports in a day, we assume the scanner detected a URL as malicious as long as it detected at least once within a day (i.e., the scanner's binary label at the day is 1). We observe that although a scanner may have both 0 and 1 for binary labels within a day, reports with "1" as binary labels have a single detailed label per scanner within a day. We thus use it as the detailed label for that day.

Notation. For a given scanner s and a URL u, its time series is represented as a binary sequence $BL_{s,u} = [bl_{t_1}, bl_{t_2}, ..bl_{t_n}]$ or a detailed label sequence $DL_{s,u} = [dl_{t_1}, dl_{t_2}, ..dl_{t_n}]$ where t_i is the i^{th} time frame, $bl_{t_i} \in \{0, 1\}$, and $dl_{t_i} \in \{$ 0, "phishing sites", "malicious sites", "malware sites", "suspicious sites", "spam sites", "mining sites", "not recommended sites" $\}$.



(a) The CDF of the number of detailed labels

(b) The ratio of top 4 detailed labels over the total number of scan reports

Fig. 3. The statistics of detailed labels (attack types) for each URL set

4 MEASUREMENT STUDY ON VIRUSTOTAL

4.1 Analysis of Attack Type

Recall that each VT scanner assigns an attack type (i.e., detailed label) for a malicious URL. This section characterizes attack types assigned by scanners for each URL set based on the detailed labels.

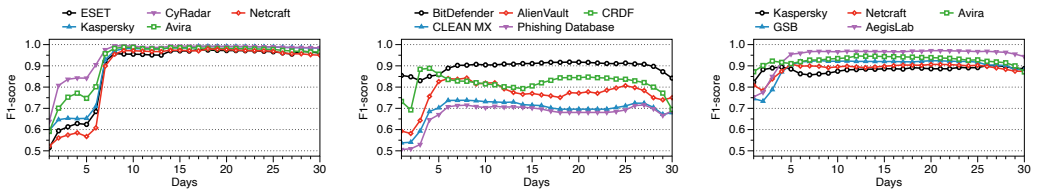
Figure 3(a) shows the CDFs (i.e., the portion of URLs) (y-axis) of the number of the detailed labels (x-axis) for each URL set. Figure 3(b) shows the ratio of the top 4 detailed labels over the total number of scan reports. Each bar presents a detailed label, the x-axis presents URL sets, and the y-axis presents the ratio of each detailed label. Both figures clearly show the different trends. That is, phishing URLs tend to have more different labels than malware URLs. Specifically, 75% of phishing URLs have 3 or more, but only 25% of the malware URLs have 3 or more labels (Figure 3(a)). While 78.5% of labels for malware URLs are malware, only 45% of labels for phishing URLs are phishing (Figure 3(b)). This means that given the conflicting labels, especially for phishing URLs, it would be hard to assign one attack type to the URL.

We observe two cases where a URL has multiple detailed labels. *First*, different scanners assign different detailed labels to the same URL. Specifically, 84.3% of URLs with multi-labels are due to this case. For example, http://faceasdasdasd.000xxxxxxxxx[redacted].com/ is always marked as "phishing" by `AegisLab`, `Fortinet`, `Kaspersky`, `Phishtank`, `Avira`, `CLEAN MX`, `Phishing Database`, `ESET`, `OpenPhish`, `G-Data`, `Emsisoft`, and `Google Safe Browsing`; as "malware" by `Sophos`, BitDefender, and `SCUMWARE.org`; and as "malicious" by `AlienVault`, `CRDF`, `Netcraft`, `CyRadar`, and `Forcepoint ThreatSeeker`. One possible reason is that scanners often have their detection specialties. In Section 4.2, we analyze scanners' specialty. *Second*, some scanners change their detailed labels for the same URL. We observe that 15.7% of URLs with multi-labels are due to such scanners quickly switching their detailed labels. In Section 4.3, we will study individual scanners' conflicting labels more in detail and show that 50% of scanners have URLs for which they keep changing the detailed labels.

Takeaway. There largely exist conflicting detailed labels for given URLs due to two cases: individual scanners' temporal and cross-scanner conflicts. Also, phishing URLs tend to have more conflicting labels than malware URLs. Given such conflicting labels, assigning one type of attack would be challenging. We analyze individual scanners' behavior in assigning attack types in more detail in Section 4.3 and propose a method to assign a final attack type given such conflicts in Section 5.

4.2 VT Scanners' Detection Specialties

This section aims to answer our research question: *Do VT scanners behave differently for different types of URLs?* We study scanners' detection specialties by measuring scanners' detection performance for different types of URLs. Recall that we have three ground-truth dataset described in Section 3.2: (1) manually labeled URLs (manual GT) including malicious and benign URLs, (2) phishing URLs (Phishing), and (3) malware URLs (Malware). *Detection Specialty* is defined as a certain



(a) Manual GT (b) Phishing (c) Malware

Fig. 4. F-1 score trends of top 5 scanners over daily timestamp for each ground truth

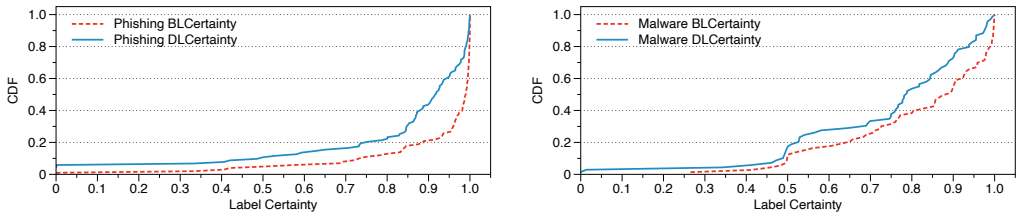
type of entity in which the scanner is specialized for detection. For instance, the scanner's detection specialty is phishing when it is capable of specifically detecting phishing attacks, not malware URLs.

We, particularly, focus on whether or not the scanner properly detects the groundtruth URLs as malicious and use the F-1 score ($\frac{2\times\text{precision}\times\text{recall}}{\text{precision + recall}}$) to accommodate the unbalanced datasets [39]. Recall that we collect the periodic scan reports for the groundtruth URLs and represent the scanner's labels for a URL over time as a time series (Section 3.3). We compute each scanner's F-1 score on each day given the groundtruth URLs. Figure 4 shows each scanner's F-1 score (the y-axis) trends over 30 days (the x-axis). We only show the top 5 scanners based on their maximum F-1 score for clarity.

We find the four interesting observations. *First*, no scanner performs well for all URL types, and thus the top 5 scanners are different for different URL types. For example, `BitDefender` works well for phishing URLs but poorly on malware URLs. In fact, 51%, 40%, and 57% of scanners cannot detect phishing, malware, and malicious manual GT URLs (i.e., 0 true positives), due to their specialties, respectively. For example, malware scanners such as `Malware Domain Blocklist`, `malwares.com URL checker`, and `Malwared` never detect phishing URLs in our dataset. *Second*, scanners often perform poorly in the early reports in VT. In general, we observe that top 5 scanners reach the maximum F-1 score near the 5th day since the first appearance in VT, which is relatively earlier than scanners' behavior for malware files (2 ∼ 4 weeks reported in previous studies [18, 35]). *Third*, there are scanners that do not change their label once they detect certain types of URLs, resulting in continuously high F-1 scores (e.g., `ESET`, `CyRadar`, `Netcraft`, `Kaspersky`, `Avira` in Figure 4(a), `BitDefender` in Figure 4(b), and `AegisLab` in Figure 4(c)). *Finally*, top scanners for phishing URLs (Figure 4(b)) have relatively lower F-1 scores than top scanners for malware URLs (Figure 4(c))). Moreover, the F-1 scores of top scanners for phishing URLs (Figure 4(b)) are less consistent over time than the F-1 scores of top scanners for malware URLs (Figure 4(c))). For example, one of the top scanners for phishing URLs, `CRDF`, quickly reaches its maximum F-1 score, and then the score continuously decreases. This is because scanners have more conflicting labels over time for phishing URLs than for malware URLs, as discussed in Section 4.1.

Note that a scanner consistently having high F-1 scores may not necessarily be a good scanner, if the status of URLs changes (e.g., compromised and cleaned). Indeed, we observe scanners not changing their decision about some URLs that are once detected then become NX (non-existent). For example, bstange.alinaalexandrovxxxxxxx[redacted].ro in VT Fresh becomes NX URL but 2 scanners such as `Fortinet` and `Webroot` still mark it as "phishing" or "malicious". However, as mentioned in Section 3.2, we build our groundtruth with URLs whose contents did not change for 30 days. Therefore, we argue that Figure 4 provides a reliable analysis of scanners' actual detection performance.

Takeaway. Scanners specialize in different attack types. Threshold-based approaches without considering such specialties may result in less accurate groundtruth. Further, scanners perform poorly in the early reports and reach the maximum F-1 score relatively earlier for URLs than for malware files. Finally, scanners' detection performances are relatively less consistent for phishing



(a) Phishing (b) Malware

Fig. 5. Scanner label certainty scores for phishing and malware URLs

URLs than for malware URLs. In Section 6, we provide recommendations on how VT users take these phenomena into account to build a better groundtruth set.

4.3 VT Scanners' Label Stability

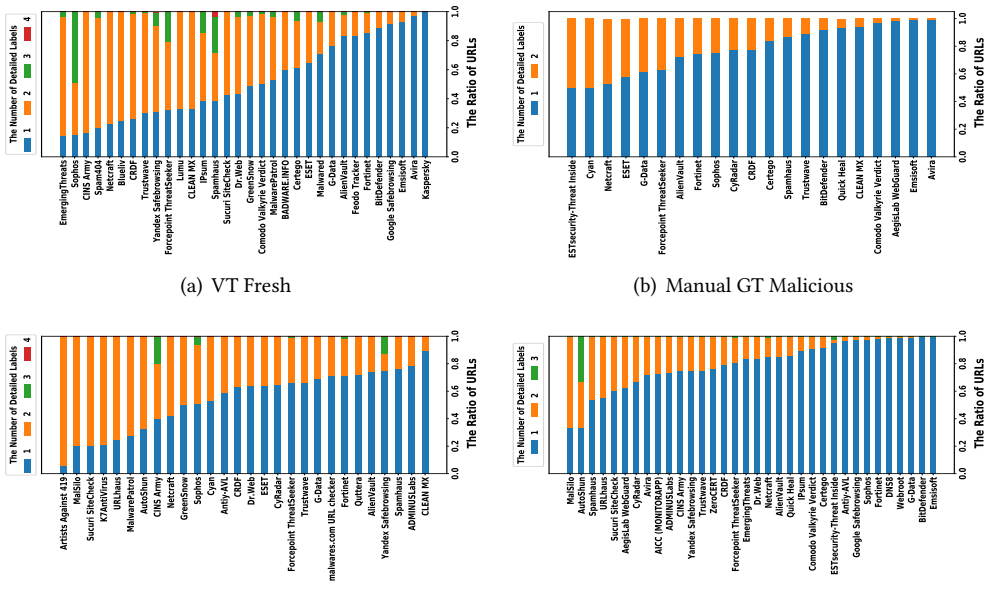
In Section 4.2, we observe the F-1 scores change over time, indicating that the scanners change their labels for given URLs. This motivates us to study the stability of binary and detailed labels of scanners. Specifically, we measure the stability of scanners' labels for malicious URLs by two certainty scores: binary and detailed label certainty scores. A binary label certainty (*BLCertainty*) is defined as how *certain* a scanner is about its detection (i.e., malicious or benign); a detailed label certainty (*DLCertainty*) is defined as how *certain* a scanner is about its detailed label (i.e., an attack type).

Binary Label Certainty (BLCertainty). A binary label certainty of scanner s for URL u measures how much time s labels u as malicious over time. For example, assume that s has 4 reports for two URLs, u_1 and u_2 where s's binary sequences are $BL_{s,u_1} = [0, \mathbf{1}, \mathbf{1}, 0]$ and $BL_{s,u_2} = [0, \mathbf{1}, \mathbf{1}, \mathbf{1}]$, respectively. Then, the binary label certainties of s for u_1 and u_2 is $certainty_b(s, u_1) = 0.5$ and $certainty_b(s, u_2) = 0.75$, respectively. Then, the binary label certainty score, *BLCertainty*, of s is computed as the average of binary label certainties for all URLs, i.e., *BLCertainty(s)* = $(0.5+0.75)/2 = 0.625$.

Detailed Label Certainty (DLCertainty). A detailed label certainty of scanner s for URL u measures whether s constantly gives the same detailed label over time. Essentially, the most *certain* label of s for u will be the most common label given by s to u. We thus extract s's most common label for u and compute a detailed label certainty as the ratio of occurrences of most common labels over time. For example, assume s's detailed label sequence for u_2 is $DL_{s,u_2} = [0$, phishing, **malware**, **malware**$]$. Its most common label is "malware" and it appears twice in 4 time periods, and thus the detailed label certainty $certainty_d(s, u_2)$ is $2/4 = 0.5$. If s always gives the same detailed label, $certainty_d(s, u)$ will be the same as $certainty_b(s, u)$. Similar to *BLCertainty*, the detailed label certainty score, *DLCertainty*, of s is computed as the average of detailed label certainties for all URLs.

Results. Figure 5 shows the CDFs (i.e., the portion of scanners) (y-axis) of two label certainty scores (x-axis) of all scanners for different types of URLs. Essentially, the line in the left side means that there are more scanners with lower label certainty scores. We generally observe that scanners have lower *DLCertainty* than *BLCertainty*. This means that although a scanner have relatively stable binary labels for URLs, it changes detailed labels over time (i.e., assigning different attack types to a given URL).

To deeper understand scanners' detailed label stability, we further measure the number of detailed labels for each URL per scanner. Figure 6 shows the distribution for the number of detailed labels per scanner (only scanners having URLs with more than 1 label are shown). The x-axis represents the set of scanners. Each bar represents the number of detailed labels. The y-axis represents the ratio of URLs that scanners assign the corresponding number of detailed labels.



(a) VT Fresh (b) Manual GT Malicious

(c) Phishing (d) Malware

Fig. 6. The distribution for the number of detailed labels per scanner (only scanners having URLs with more than 1 label are shown)

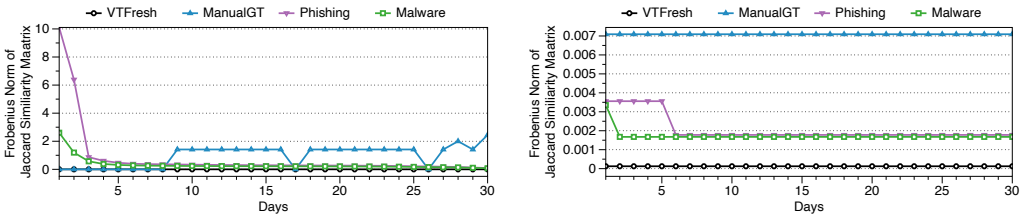
Figure 6 shows that 49 scanners assign multiple attack types to at least one of the URL sets; some scanners even assign 4 attack types to the same URL. For example, `Spamhaus` assigns 4 attack types to 3.9% of VT Fresh URLs (Figure 6(a)) and a few phishing URLs (not visible in the figure) (Figure 6(c)). Figure 6 shows that scanners considered highly reputable in the literature such as `Sophos`, `Bitdefender`, and `Kaspersky` [3, 48, 56] also assign multiple attack types to given URLs. For example, `Sophos` assigns at least 2 attack types to 85% of VT Fresh URLs (Figure 6(a)), 50% of phishing URLs (Figure 6(c)), and 2% of malware URLs (Figure 6(d)). This suggests that although using only highly reputable scanners may increase detection accuracy, assigning an attack type to a URL would still be challenging.

Interestingly, we observe different behavior of scanners assigning multiple types of attacks to the same URL. Concretely, 53% of scanners constantly change their detailed label from one to another in the beginning, and then stabilize with one type of attack. For example, `Sophos` switches its label every day for "jp-billxxxxxx[redacted].com" between "malware" and "phishing"; then later it stabilizes as "phishing". Meanwhile, 47% of scanners never stabilize their labels. For example, `Fortinet` keeps changing its label between "phishing" and "malware" for "wikixxxx[redacted].cz/wiki/ [redacted]."

Takeaway. Scanners often change their binary and detailed labels for the same set of URLs. Moreover, scanners are less "certain" about the attack types (*DLCertainty*) than the maliciousness itself (*BLCertainty*) leading to challenges in deciding an attack type for given URLs. Given these different scanners' behavior, we propose a method to assign a final attack type to each URL at a given time point in Section 5.

4.4 VT Scanners' Correlation

One may take scanners consistently having high F-1 and certainty scores as reputable for each attack type and choose thresholds or determine the attack type considering only such reputable scanners [56]. However, this section shows there exist highly correlated scanners in terms of both binary and detailed labels that may degrade threshold-based approaches for detection and produce



(a) Scanner binary label similarity Frobenius norm

(b) Scanner detail label similarity Frobenius norm

Fig. 7. Frobenius norm of Jaccard similarity of scanner's binary/detail labels over time

a bias for a majority voting-based approach for attack type detection. We analyze the pairwise correlation among scanners using two similarity measures: Jaccard similarity [17] and dynamic time warping (DTW) [6].

Scanners' Co-labeled URL Similarity. To measure the similarity in terms of co-labeled URLs, we employ Jaccard similarity for binary and detailed labels at each time point as well as over time. Specifically, we measure Jaccard similarity for binary labels by the number of co-detected URLs over the total number of URLs; Jaccard similarity for detailed labels by the number of URLs having the same detailed labels over the total number of URLs. For example, when the set of total URLs is u_1, u_2, u_3, u_4, u_5, scanner s_1 detected u_1, u_2, u_3, and scanner s_2 detected u_2, u_3, u_4, u_5, then the Jaccard similarity for a binary label is 2/5. Although s_1 and s_2 co-detected u_2 and u_3, s_1 and s_2 may have different detailed labels. And if $dl(s_1, u_2)$="malware", $dl(s_1, u_3)$ = "malware", $dl(s_2, u_2)$="malware", and $dl(s_2, u_3)$ = "phishing", the Jaccard similarity for a detailed label is 1/5 due to their different detailed labels for u_3.

We present the heatmaps of pairwise Jaccard's similarity of binary and detailed labels over all periods in Appendix D. Scanners co-detecting URLs with at least one scanner are shown in the heatmaps. A darker cell in the heatmap means high similarity, while a lighter cell means low similarity. We also compute the Frobenius norm [15] of the pairwise Jaccard similarity matrix at each time point to measure if the similarity is consistent over time. Figure 7 shows how the Frobenius norm (y-axis) changes over 30 days (x-axis). The larger norm indicates that there are more highly similar scanners in terms of detection (binary labels) or attack type assignment (detailed labels).

In general, we observe that more scanners have high Jaccard similarity for phishing URLs (more darker cells in heatmaps and the larger norm in Figure 7) than for malware URLs. Also, the Jaccard similarity of detailed labels is lower in general (lighter in heatmaps and the lower norm in Figure 7). Meanwhile, we observe a few scanners having high Jaccard similarity for detailed labels (the darkest) such as `ESTsecurity` and `Scantitan` for phishing URLs.

Figure 7(a) shows that for phishing URLs, there are more scanners having high similarity for binary labels in the beginning, then continuously the norm decreases over time. One possible reason is that shortly after detecting the phishing URLs, some scanners gradually change their label to benign, resulting in less similarity. Furthermore, while there are fewer scanners having high similarity for malware URLs, the norm is relatively consistent over time. We also observe fewer scanners having high similarity for detailed labels (and thus low norm such as 0.0036 compared to norm of 10 for binary labels) and the consistent norm.

Scanners may have high similarities due to multiple reasons. If a scanner copies others directly (e.g., a scanner uses a blacklist provided by another scanner [1]), the simple threshold-based approaches will be biased and unreliable. Meanwhile, scanners having high similarity in detection, albeit their independent methods, may indicate high confidence in detection so that the higher positive counts provide stronger signals. Scanners having high binary label similarity yet low

detailed label similarity suggest that such scanners may have independent approaches (inspecting different signals from URLs), and thus one may treat the positive counts from such scanners as the level of maliciousness. Meanwhile, scanners having both high binary and detailed label similarities suggest high correlations, and thus one may penalize the count accordingly.

Scanners' Labeling Trend Similarity. If one scanner copies another, or two scanners share similar (if not the same) features, their label trends should be similar. If one scanner copies another, the copied version's detection would be delayed with the same label trend. We thus further compare scanners' labeling patterns. To measure the similarity of scanners' binary labels' patterns, we employ dynamic time warping (DTW) distance that computes the similarity between two temporal sequences [6]. Essentially, DTW distance can measure if the evolution of labels is similar regardless of their speed. To get the final DTW distance between two scanners, we measure the DTW distance of all pair sequences for co-detected URLs and then compute the average.

We run a hierarchical clustering algorithm based on DTW distance and cut the dendrograms (Figure 18 in Appendix D) by the level. Figure 8 shows the resulting clusters for phishing and malware URLs. Note that we do not consider two scanners similar when both do not detect the URL at all over time.

(a) Phishing (b) Malware

Fig. 8. Scanner clustering using DTW distance

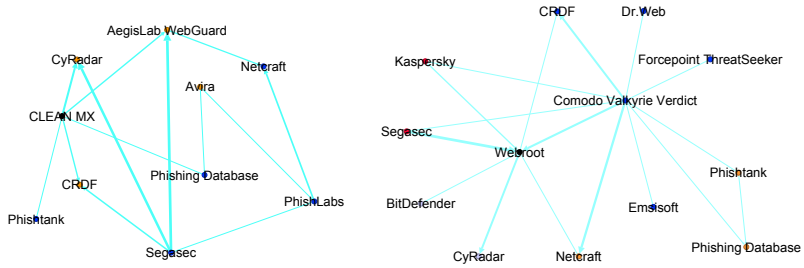
Figure 8 suggests two observations. *First*, different clusters are built for different types of URLs. For example, for phishing URLs (Figure 8(a)), `G-Data` is closely clustered with Google Safe Browsing (`GSB`), but it is closely clustered with `CLEAN MX` for malware URLs (Figure 8(b)). *Second*, scanners specialized in the same attack type get clustered together. That is, the phishing scanners (e.g., `PhishTank`, `PhishLabs`, and `Phishing Database`) clustered for phishing URLs (Figure 8(a)) and the malware scanners (e.g., `MalwarePatrol` and `Malsilo`) clustered for malware URLs. Figure 8(b)) confirm that our clustering method indeed captures meaningful clusters. Meanwhile, such clustered scanners (i.e., having highly similar trends of binary labels) suggest that some scanners may not be independent (e.g., one may refer to and utilize another scanner's labels and do delayed detection compared to another with a similar labeling trend) for a URL.

Scanners' Causal Relationship. We further examine the potential causal relationships between correlated scanners using a popular causality measure for time-series data, Transfer Entropy (TE) [25, 45]. Let $S1$ be scanner $s1$'s detailed label time-series sequence and $S2$ be scanner $s2$'s detailed label time-series sequence for a URL u. TE from $S1$ to $S2$ ($TE(S1 \rightarrow S2)$) quantifies how likely s_1's label change will influence s_2's label change, which is defined as follows.

$$TE(S1 \rightarrow S2) = H(S2_i, S1_{i-\text{lag}}, \ldots, S1_{i-1}) - H(S2_i | S1_{i-\text{lag}}, \ldots, S1_{i-1}),$$

where $S1_i$ and $S2_i$ are the detailed labels of s_1 and s_2 at time i respectively, lag is the chosen time lag, and $H(\cdot)$ represents entropy. Essentially, a larger positive TE value indicates a stronger influence

of s_1's labeling trends on s_2's labeling trends. We compute $TE(S1 \rightarrow S2)$ for all common URLs between $S1$ and $S2$ and average the TE value to represent a causal relationship between s_1 and s_2. To better understand the influence between correlated scanners, we build a causal relationship graph (Figure 9) where scanners are nodes and a directed edge from scanner s_1 to scanner s_2 is drawn, and the thickness (weight) of the edge represents the average TE value.



(a) Phishing (b) Malware

Fig. 9. Scanners' Causal Relationship Graph (Scanners having > 0.5 labelling similarity are only shown)

`CLEAN MX` is known to submit many URLs to `Phishtank` and refer to `Phishtank`'s decision [36], which is captured by our causal influence graph (Figure 9(a)). It is interesting to see that `Comodo Valkyrie Verdict` highly influences other scanners in the same cluster (e.g., `CRDF`, `Dr.Web`, `Forcepoint`, `Emsisoft`) for malware URLs (Figure 9(b)). `BitDefender` and `CyRadar` are influenced by only `Webroot`, resulting in having a high correlation and getting clustered for malware URLs (Figure 18).

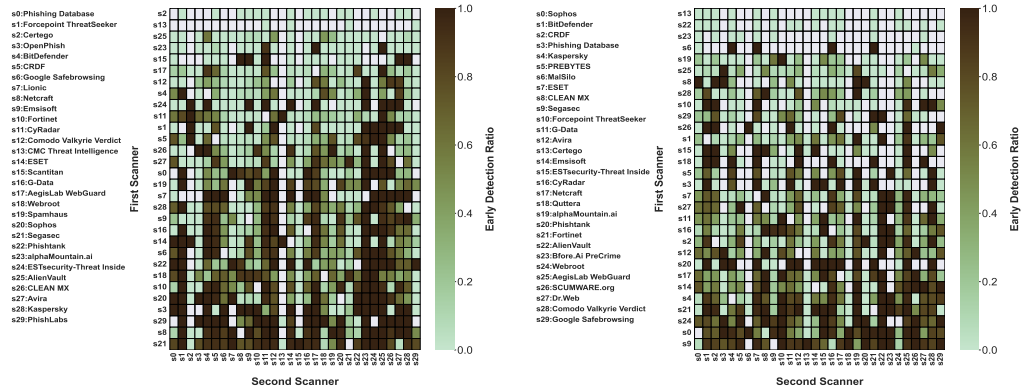
Takeaway. We observe highly correlated scanners in terms of their temporal similarity and the overall similarity in the trend of their label patterns. Scanners may have high correlations due to two reasons: scanners' specialty in the same attack type albeit their independent methods and causally related scanners (one referring to other detection methods [1, 36]). One may prefer scanners always detecting URLs earlier than others among those highly correlated ones. In the next section, we thus analyze if lead/lag relationships between scanners exist.

4.5 Lead & Lag Analysis

As malicious URLs are often short-lived, it is crucial to detect URLs as early as possible. In Section 4.4, we observe highly correlated scanners in terms of the co-labeled URLs (Jaccard similarity) and the patterns of binary label trends (DTW distance). In this section, we analyze if there is any lead/lag relationship among those correlated scanners. For example, if scanners s_1 and s_2 detect the same set of URLs yet the s_1 always detects URLs earlier than s_2, we may fairly say s_1 is a *leader* and s_2 is a lagger. We thus compare the first detection time of two scanners for co-detected URLs.

Figure 10 presents the pairwise early detection ratio matrices for phishing and malware URLs measured by the number of URLs that the first scanner (the y-axis) detected earlier than the second scanner over the total number of co-detected URLs. The matrix is sorted so that the darkest row is at the bottom. If there are no co-detected URLs, it is marked as *xxx*. Note that a scanner that does not co-detect URLs with any scanner will not appear in the matrix. Essentially, a completely dark row means that the corresponding row scanner always detects earlier than other scanners; a completely dark column indicates that the corresponding column scanner always detects later than other scanners.

First, as shown in Figure 10, we observe scanners detect relatively earlier than others (e.g., `Segasec`) and scanners detect relatively later than others (e.g., `alphaMountain.ai`). *Second*, we observe closely clustered scanners (i.e., the label trend is highly similar) where one always detects URLs earlier than another for a specific type of URLs. For example, while `Webroot` and `alphaMountain.ai`



(a) Phishing (b) Malware

Fig. 10. Early detection ratio of 1st scanner being earlier than 2nd scanner (sorted by the darkness of rows)

have similar labeling patterns (and thus closely clustered) for phishing URLs (Figure 8(a)), `Webroot` always detects URLs earlier than `alphaMountain.ai` (Figure 10(a)). Then, one may prefer `Webroot` over `alphaMountain.ai` for phishing URLs. *Third*, while `MalSilo` do not co-detect many URLs with other scanners (i.e., most cells are *xxx*), it mostly detects earlier than other scanners among those co-detected URLs. This suggests that `MalSilo` may employ an independent method that can compensate for other scanners' detection.

Meanwhile, we observe there are more scanners detecting the same set of phishing URLs than those detecting the same set of malware URLs (i.e., Figure 10(a) has fewer cells with *xxx* than Figure 10(b)). Further, more lead/lag relationships exist in phishing URLs than malware URLs (i.e., Figure 10(a) has more darker cells than Figure 10(b)). This means the approaches detecting malware URLs are more likely to be independent of other scanners than approaches detecting phishing URLs.
Takeaway. There exist lead/lag relationships among scanners having similar label patterns. Meanwhile, more scanners are correlated to detect phishing URLs than malware URLs. Along with the results in previous sections, one may consider leading and highly accurate scanners' results while penalizing the positive counts.

5 ATTACK TYPE DETECTION MODELING

Identifying if a malicious URL is involved in phishing or malware attacks is important in practice as these two attacks require different mitigation actions and malicious URLs are aggregated to threat-specific feeds [13]. As examined in Section 4, scanners often do not agree on a single attack type label. Hence, the commonly used majority voting based approach and VT label classifier utilizing VT labels as features, our baselines, are sub-optimal (see Table 4). The majority voting approach assigns the label of the majority class as the label of each URL. The VT label classifier directly uses the VT labels from the VT reports for the URLs as features for the classification. Thus, both baseline approaches do not consider scanner verdict conflicts and correlations. Instead, one needs an approach to account for scanners' dependencies and varying expertise. To this end, one approach is to learn a set of latent variables for each scanner from a large corpus of historical VT reports, capturing the scanner dependencies and expertise. Utilizing these latent variables along with other commonly available features from prior work, we construct a supervised learner to classify malicious URLs, which achieves 10-45% classification performance improvement over the baseline.

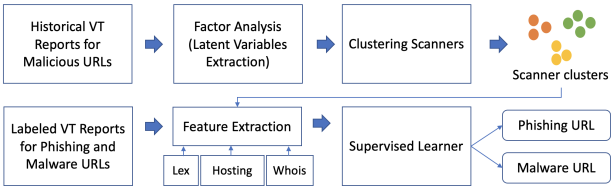


Fig. 11. Overall workflow of classifying malicious URLs as phishing or malware

Our Approach. Figure 11 shows the overall classification pipeline of our approach. Our analysis in Section 4.4 shows that scanners are highly correlated in terms of both detailed and binary labels. Further, scanners detecting phishing and malware URLs form distinct clusters. Motivated by these observations, first, we cluster similar scanners together based on the latent variables we derive. Along with the VT cluster features, we then utilize three groups of features: lexical, hosting, and WHOIS. Lexical features refer to the textual features related to URLs [26]. Phishing URLs are more likely to have lexical features impersonating popular brands than malware ones. Hosting features, capturing the differences in the hosting infrastructures utilized for these two types of attacks, extract attributes related to the IP addresses where URLs are hosted [26]. WHOIS features are extracted from WHOIS registration records for each domain [14]. The coverage of hosting and WHOIS features are 75% and 71% respectively. Table 3 shows a summary of the features used for each category, lexical, hosting and WHOIS. VT cluster features include scanner attack labels and the features derived from scanner clusters. Latent scanner features are derived from the factor analysis on randomly selected 20K recent historical VT reports with at least two positives. We use the detailed and binary labels of the scanners in each report as input features to the factor analysis. Our intuition is that these features capture scanner dependencies and varying degrees of expertise. We take the top 5 factors and cluster scanners into multiple groups. We vary the number of clusters from 5 to 20 and identify that 15 clusters produce the best downstream performance. Utilizing these clusters, we extract VT cluster features taking the scanner cluster assignment as the input and computing adjusted phishing and malware label proportions for each malicious URL. We observe that the adjusted label proportions perform better in the downstream classification task compared to the raw label proportions. A key reason for the significant performance gain is due to, as we have shown earlier, dependencies among scanners and highly correlated results at times. The adjusted label proportions consider these dependencies and compute more discriminative features to differentiate between phishing and malware URLs. Figure 11 shows the overall classification pipeline of our approach.

Model Training and Testing. To build our model, we use balanced datasets from each class collected on Mar. 1, 2021 – phishing and malware URLs – described in Section 3 where 5,823 URLs from each class are used. We train MLP (Multi-Layer Perceptron), XGBoost, Random Forest (RF), Support Vector Machine, K Nearest Neighbor, Decision Tree, Naive Bayes, Logistic Regression and Linear Discriminant Analysis. RF yields the best result and hence all the experiments are performed with RF. Randomized search based hyperparameter optimization identifies the optimal maximum depth to be 250, a maximum number of features to be 55, the number of estimators to be 200. We utilize 80-20 train-test split and Figure 12 shows the ROC curve for the two classes. Table 4 shows the offline performance metrics for the baseline models and our approach. The two baseline models utilized are as follows: majority voting and VT label classifier, a RF classifier that uses VT labels as features. While the two baseline models perform poorly, our model achieves a high accuracy, precision, recall and a low false positive rate for each class in general. We attribute the performance improvement to the inclusion of latent scanner features along with lexical, hosting and WHOIS features.

Table 3. Summary of the features used

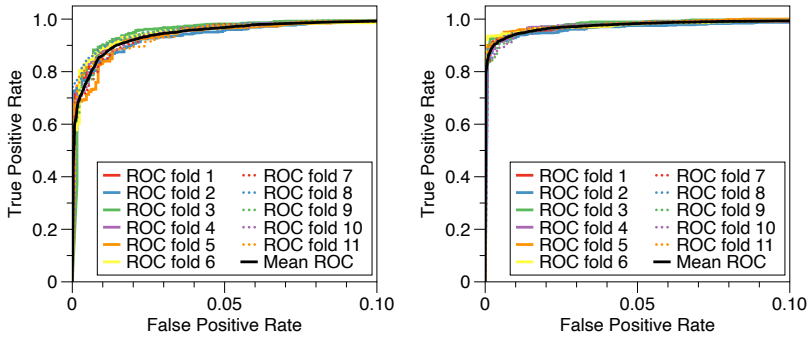
Feature Name	Description	Type
Hosting Features		
Duration	The time interval from the first seen to the last seen records for a domain	Numerical
Number of IPs	The number of IPs on which the domain is recently hosted	Numerical
Number of ASNs	The number of Autonomous System Numbers associated with the IP	Numerical
Number of queries	The number of times the domain is queried	Numerical
Number of name servers	The number of authoritative name servers used to lookup the domain	Numerical
Name server match	Does at least one authoritative NS domain match with the domain name?	Boolean
Number of SOAs	Number of SOA (Start of Authority) domains associated with the domain	Numerical
SOA match	Does at least one SOA domain matches with the domain name?	Boolean
Number of domains per IP	The average number of domains hosted on all hosting IPs of a domain	Numerical
Lexical Features		
Entropy	The Shannon entropy of the domain	Numerical
Number of popular brands	The number of popular brands that appear in malicious domains.	Numerical
The position of the first brand	The position of the first brand appearing the domain name	Numerical
Domain length	Length of the domain name	Numerical
Number of subdomains	Number of subdomains in the domain name	Numerical
Number of dashes	Number of dashes in the domain name	Numerical
Number of suspicious words	The number of suspicious words such as login, register, and secure that appear in the domain name.	Numerical
Is a generic TLD present	Is a generic TLD such as -com-, -org- and -net- present?	Boolean
Is IDN	Is it an internationalized domain name?	Boolean
Is IP	Is it an IP hostname, where the hostname is an IP address?	Boolean
Is suspicious TLD	Is the TLD of the domain in the list of TLDs with a low reputation?	Boolean
WHOIS Features		
Duration	The life time of the domain from the creation to the expiration date	Numerical
Renewed	Is the domain renewed after it was initially registered?	Boolean
Registrar	The name of the domain registrar	Categorical
Number of name servers	The number of name servers mentioned in the WHOIS record	Numerical
Status	The server status of the WHOIS record	Categorical
Is privacy protected	Is the domain registration privacy protected?	Boolean

Table 4. Attack type classification performance of the baselines and our approach.

Type	Baseline1* (Majority Voting)				Baseline2+ (VT Label Classifier)				Our Approach			
	Acc.	Prec.	Rec.	FPR	Acc.	Prec.	Rec.	FPR	Acc.	Prec.	Rec.	FPR
Phishing	81.72	69.90	92.98	25.43	94.04	89.33	92.54	5.24	**97.47**	**95.45**	**96.91**	**2.3**
Malware	70.10	51.59	19.93	8.12	90.24	88.13	83.58	6.10	**96.34**	**95.93**	**93.30**	**2.1**

We utilize the second set of ground truth URLs collected on Jul. 30, 2021 to test the trained model to ascertain its performance over a period of time. It contains 763 phishing and 658 malware URLs. We observe that the prediction performance degrades only less than 1% over the 4 month apart trained model and the test set. We attribute the stable performance of the model over time to temporally agnostic features utilized.

Error Analysis. We analyze *all* the misclassified URLs. For the total of 13 malware URLs misclassified as phishing URLs (e.g. support-beleid.xyz) and the total of 10 phishing URLs misclassified as malware URLs (e.g. ug62ud8jtox9jw.buzz), we observe that their average confidence scores are 0.64 and 0.68, respectively, which are quite close to the default threshold of 0.5. By increasing the classification threshold, one may reduce the misclassifications at the expense of reducing the true positive rate. For each misclassified sample, we investigate the importance of its features in classification using SHAP explainer. We observe that misclassified phishing URLs lack telltale lexical



(a) ROC Curve for Phishing (b) ROC Curve for Malware

Fig. 12. ROC Curves for Attack Types

Table 5. Performance of the attack type classification for different feature categories.

Feature Sets	Phishing				Malware			
	Acc.	Prec.	Rec.	FPR	Acc.	Prec.	Rec.	FPR
VT cluster labels	95.49	92.83	93.60	3.6	94.42	93.09	90.47	3.5
VT cluster labels + lexical	96.38	93.82	95.25	3.1	95.26	93.84	92.10	2.6
VT cluster labels + hosting	96.98	94.73	96.17	2.6	95.27	94.25	91.83	2.9
VT cluster labels + whois	96.91	94.91	95.78	2.5	95.71	94.85	92.48	2.6
VT cluster labels + lexical + hosting	97.21	95.24	96.35	2.4	95.67	94.65	92.53	2.7
All (Our approach)	**97.47**	**95.45**	**96.91**	**2.3**	**96.34**	**95.93**	**93.30**	**2.1**

features such as the presence of a popular name, the inclusion of a TLD name in the domain name, and use of suspicious keywords. The SHAP analysis of misclassified URLs suggest that having additional features (e.g. certificate features based on the TLS certificates issued for the domains of these URLs, content features extracted from the HTML content of the webpages for these URLs) may assist further distinguish phishing from malware URLs.

Ablation Analysis. As shown in Table 5, we analyze the performance with respect to different feature categories. We experiment lexical, hosting and WHOIS features separately along with VT cluster features. While the performance improves around 1% in each of these scenarios compared to only utilizing VT cluster features, 2-3% improvement when all feature categories are considered. This indicates that each feature category helps learn different aspects of phishing and malware URLs. Oftentimes, collecting WHOIS records for domains is quite challenging. In such a situation, we recommend utilizing only lexical and hosting features with only slightly dropped performance (0.3%) compared to having WHOIS features.

Feature Analysis. In addition to VT cluster features and VT labels from some of the scanners, we observe the following features to be important for the classification: renewed (whois), duration (hosting), domain length (lexical), number of suspicious keywords (lexical), number of popular brands (lexical), registrar (whois), number of SOA domains (hosting), number of name servers (hosting) and number of queries (hosting). This shows that different feature categories contribute to a strong classifier, which is inline with the findings from the ablation analysis above.

Longitudinal Results. We apply our classifier on 56,138 VT malicious URLs randomly chosen from VT General Feed between Mar. 2021 and Jul. 2021. Our predictions show that 11,922 and 44,216 are phishing and malware, respectively. Figure 13 shows the weekly percentage of these two

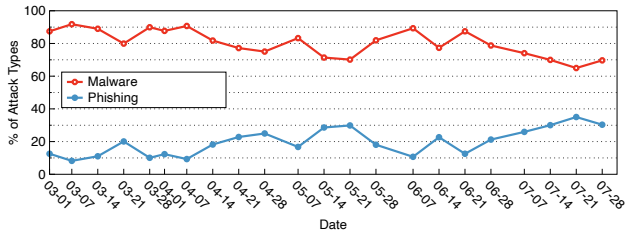


Fig. 13. Attack types proportions observed in VT General Feed over time

attack types over the 4 months. The relative proportions of these attacks have been quite stable in this quarter and the malware URLs consistently dominate phishing URLs observed in VT over time.

6 DISCUSSION

Recommendation on the usage of VT. Our studies have several important implications in using VT to build a ground truth. Based on our studies, we provide the following recommendations on how to better utilize VT to build groundtruth sets. (1) Our studies show that identifying attack types of URLs is an important consideration in building a ground truth using VT. VT users should employ our proposed classifier to identify an attack type with high accuracy at the early stage and attempt to build individual groundtruth depending on the attack type. (2) VT users should collect the URL groundtruth set earlier than files (e.g., around the 5th day since their first appearance, compared to 2 weeks for files). (3) VT users should utilize our analysis of scanner specialties and prioritize detection labels by top scanners for a specific attack type instead of using VT *positive* directly or hand-picked reputable scanners [56] to determine the level of maliciousness [8] (Section 4.2). (4) VT users should utilize our analysis of scanner correlations for different attack types and proportionally weigh less the VT *positive* counts from correlated scanners to obtain better ground truth instead of directly using VT *positive* and a fixed threshold for all types of URLs. (5) VT users should use a higher threshold to compile phishing URL ground truth compared to malware URL ground truth, given the higher correlation between scanners and their less consistent performance for phishing URLs (Section 4.4) (6) VT users should utilize our analysis of lead/lag relationship and prioritize highly accurate leading scanners to build the most accurate and reliable ground truth at the early stage (Section 4.5). We also have recommendations for a VT service provider and individual VT scanners. (1) VT may take advantage of our analysis and attempt to provide a better global metric (e.g., weighted positive count) along with the aggregated information. (2) VT may provide users more information about individual scanners, such as scanner specialties, for users to quickly refer to results from specialized scanners to build groundtruth sets for a specific attack type. (3) VT may continue to monitor the performance of scanners utilizing our analysis and proactively interact with individual scanners. For example, VT may notify scanners about their potential problem in detection, such as false positive cases or delayed detection. Individual scanners must quickly correct such problems, improve their method, and immediately update their results with VT.

Limitation on Ground truth. Collecting large-scale, ground-truth URLs is often challenging [13]. Despite our best efforts to cover various types of attacks on the ground truth, our dataset may still have two limitations. *First*, two external sources in our ground truth dataset may have a certain bias on their URL lists. However, for better confidence in the ground truth, considering noises in 2 external sources, we take the conservative approach of the additional manual verification (e.g., excluding the URLs that domain experts have disagreed on labeling from ground truth). *Second*, while it is relatively easy to judge if a URL is phishing/non-phishing and malware/non-malware, it

is hard to judge if the URL is benign by human experts. Hence, our manual GT benign URLs can be biased towards popular domains. In future work, we will study VT with less popular benign URLs.

7 RELATED WORK

VT as Ground Truth. VT has been used to build a ground truth in various domains including malware files [21, 23, 55] and IP/URLs [24, 28, 33, 44, 49] detection. In doing so, the most common approach is an unweighted threshold-based method employing a heuristically chosen number of scanners by which the entity is marked as malicious. While there is no consensus on such a number [54, 56], surprisingly, small thresholds such as 1 or 2 have been widely used in the literature [28, 44, 49, 54, 56]. Only a few papers set aggressive thresholds [21, 55]. Small thresholds often lead to high false positives [33] and high thresholds often result in low coverage [11, 35, 56]. A few studies treat the number of detecting scanners as the level of maliciousness [8, 11]. However, we show that the absolute number does not necessarily mean the level of maliciousness due to high correlations between scanners.

Threat Intelligence Aggregation. A number of recent studies measured the qualities of multiple threat intelligence sources including VT [7, 9, 12, 13, 22, 24, 31, 35, 37, 40–42, 50, 56]. Particularly, their work mainly focused on evaluating the stability (e.g., detection label dynamics) of VT malware file scanners and the detection accuracy of VT phishing URL scanners for IRS/paypal phishing URLs [7, 9, 22, 35, 50, 56]. While these researches provided insights about detection qualities of intelligent sources, their works were conducted with limited datasets (or a single snapshot of reports) in terms of diversity and scale [9, 16, 29, 35, 40, 43]. In contrast, we provide a large-scale longitudinal analysis for various types of URLs. In doing so, we analyze the specialty of scanners and their correlations for different attack types and propose a method to identify attack types of URLs.

A few studies proposed ways to aggregate different sources considering qualities [16, 18, 29, 38, 40, 43, 47]. Kantchelian *et al.* [18] proposed two machine learning models with the assumption that scanners are independent. However, we show that some scanners are highly correlated and cannot be considered independent. Sakib *et al.* [40] and Thirumuruganathan *et al.* [47] proposed ways to optimally combine malware scanners and general threat intelligence sources, respectively, with consideration of dependencies between scanners. However, Sakib *et al.*'s work cannot be applied when the ground truth is built depending solely on VT, as it violates their key assumption (i.e., scanners' detection probabilities are given). We also show each scanner's detection probability can vary over time and for different attack types due to its detecting specialty. Thirumuruganathan *et al.*'s method clustered URLs into benign or malicious by integrating noisy scan reports without such assumption [47]. However, they did not provide a systematic quantitative study on the characteristics of scan reports, which is one of our main focuses. We also show that the attack type of malicious URLs is an important factor when building ground truth, and propose a method to classify the attack types.

8 CONCLUSIONS

In this paper, we provide a large-scale analysis of VT URL scan reports spanning over two years. We show that existing approaches to determining the maliciousness and attack types are limited due to multiple factors including conflicts between scanners, and the specialty, stability, correlation, and lead/lag behavior of scanners. Our analyses show that scanners behave differently for different attack types and identifying an attack type is critical in building proper groundtruth sets using VT. We propose an approach considering such characteristics to identify the attack type of a malicious URL. We suggest that VT users first need to quickly and reliably identify attack types of malicious URLs, depending on which, VT users need to build groundtruth sets for corresponding attack types considering VT scanners' behavior with regard to a specific attack type and choose proper mitigation actions.

REFERENCES

[1] 2021. EST Security. https://en.estsecurity.com/product/alyac.
[2] APWG. 2021. Anti-Phishing Working Group. https://www.apwg.org/.
[3] Daniel Arp, Michael Spreitzenbarth, Malte Hubner, Hugo Gascon, Konrad Rieck, and CERT Siemens. 2014. Drebin: Effective and explainable detection of android malware in your pocket.. In *NDSS*, Vol. 14. 23–26.
[4] AT&T. 2021. Open Threat Exchange. https://cybersecurity.att.com/open-threat-exchange.
[5] Paul N Bennett and Vitor R Carvalho. 2010. Online stratified sampling: evaluating classifiers at web-scale. In *19th ACM CIKM*. 1581–1584.
[6] Donald J Berndt and James Clifford. 1994. Using dynamic time warping to find patterns in time series.. In *KDD workshop*, Vol. 10. 359–370.
[7] Xander Bouwman, Victor Le Pochat, Pawel Foremski, Tom Van Goethem, Carlos H Gañán, Giovane Moura, Samaneh Tajalizadehkhoob, Wouter Joosen, and Michel van Eeten. 2022. Helping hands: Measuring the impact of a large threat intelligence sharing community. In *Proc. of the 31st USENIX Security Symposium*.
[8] Onur Catakoglu, Marco Balduzzi, and Davide Balzarotti. 2016. Automatic extraction of indicators of compromise for web applications. In *Proc. of the 25th Web Conference*. 333–343.
[9] John Charlton, Pang Du, Jin-Hee Cho, and Shouhuai Xu. 2018. Measuring Relative Accuracy of Malware Detectors in the Absence of Ground Truth. In *IEEE MILCOM 2018*. 450–455.
[10] Hyunsang Choi, Bin B Zhu, and Heejo Lee. 2011. Detecting malicious web links and identifying their attack types. In *2nd USENIX WebApps*.
[11] Euijin Choo, Mohamed Nabeel, Mashael AlSabah, Issa Khalil, Ting Yu, and Wei Wang. 2022. DeviceWatch: A Data-Driven Network Analysis Approach to Identifying Compromised Mobile Devices with Graph-Inference. *ACM Transactions on Privacy and Security* (2022).
[12] Pang Du, Zheyuan Sun, Huashan Chen, Jin-Hee Cho, and Shouhuai Xu. 2018. Statistical estimation of malware detection metrics in the absence of ground truth. *IEEE TIFS* 13, 12 (2018), 2965–2980.
[13] Álvaro Feal, Pelayo Vallina, Julien Gamba, Sergio Pastrana, Antonio Nappa, Oliver Hohlfeld, Narseo Vallina-Rodriguez, and Juan Tapiador. 2021. Blocklist babel: On the transparency and dynamics of open source blocklisting. *IEEE TNSM* (2021).
[14] Mark Felegyhazi, Christian Kreibich, and Vern Paxson. 2010. On the Potential of Proactive Domain Blacklisting. In *3rd USENIX LEET*. 6.
[15] Roger A Horn and Charles R Johnson. 2012. *Matrix analysis*. Cambridge university press.
[16] Médéric Hurier, Guillermo Suarez-Tangil, Santanu Kumar Dash, Tegawendé F Bissyandé, Yves Le Traon, Jacques Klein, and Lorenzo Cavallaro. 2017. Euphony: Harmonious unification of cacophonous anti-virus vendor labels for android malware. In *2017 IEEE/ACM 14th MSR*. 425–435.
[17] Paul Jaccard. 1912. The distribution of the flora in the alpine zone. 1. *New phytologist* 11, 2 (1912), 37–50.
[18] Alex Kantchelian, Michael Carl Tschantz, Sadia Afroz, Brad Miller, Vaishaal Shankar, Rekha Bachwani, Anthony D. Joseph, and J. D. Tygar. 2015. Better Malware Ground Truth: Techniques for Weighting Anti-Virus Vendor Labels. In *Proc. of the 8th ACM AISEC*. 45–56.
[19] Namit Katariya, Arun Iyer, and Sunita Sarawagi. 2012. Active evaluation of classifiers on large datasets. In *12th IEEE ICDM*. 329–338.
[20] Sungjin Kim. 2020. Anatomy on Malware Distribution Networks. *IEEE Access* 8 (2020), 73919–73930.
[21] David Korczynski and Heng Yin. 2017. Capturing malware propagations with code injections and code-reuse attacks. In *2017 ACM CCS*. 1691–1708.
[22] Marc Kührer, Christian Rossow, and Thorsten Holz. 2014. Paint It Black: Evaluating the Effectiveness of Malware Blacklists. In *RAID*, Angelos Stavrou, Herbert Bos, and Georgios Portokalidis (Eds.). 1–21.
[23] Bum Jun Kwon, Jayanta Mondal, Jiyong Jang, Leyla Bilge, and Tudor Dumitraş. 2015. The dropper effect: Insights into malware distribution with downloader graph analytics. In *22nd ACM CCS*. 1118–1129.
[24] Vector Guo Li, Matthew Dunn, Paul Pearce, Damon McCoy, Geoffrey M Voelker, and Stefan Savage. 2019. Reading the tea leaves: A comparative analysis of threat intelligence. In *28th USENIX Security Symp*. 851–867.
[25] Brian Lindner, Lidia Auret, Margret Bauer, and Jeanne WD Groenewald. 2019. Comparative analysis of Granger causality and transfer entropy to present a decision flow for the application of oscillation diagnosis. *Journal of Process Control* 79 (2019), 72–84.
[26] Justin Ma, Lawrence K. Saul, Stefan Savage, and Geoffrey M. Voelker. 2009. Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs. In *Proc. of the 15th ACM KDD*. 1245–1254.
[27] McAfee. [n.d.]. Site Advisor. https://www.mcafee.com/siteadvisor
[28] Najmeh Miramirkhani, Timothy Barron, Michael Ferdman, and Nick Nikiforakis. 2018. Panning for Gold.Com: Understanding the Dynamics of Domain Dropcatching. In *Proc. of the 2018 Web Conference*. 257–266.
[29] Aziz Mohaisen and Omar Alrawi. 2014. Av-meter: An evaluation of antivirus scans and labels. In *DIMVA*. 112–131.

[30] Abdallah Moubayed, MohammadNoor Injadat, Abdallah Shami, and Hanan Lutfiyya. 2018. Dns typo-squatting domain detection: A data analytics & machine learning based approach. In *IEEE GLOBECOM*.

[31] Adam Oest, Yeganeh Safaei, Penghui Zhang, Brad Wardman, Kevin Tyers, Yan Shoshitaishvili, and Adam Doupé. 2020. Phishtime: Continuous longitudinal measurement of the effectiveness of anti-phishing blacklists. In *29th {USENIX} Security Symposium*. 379–396.

[32] Adam Oest, Penghui Zhang, Brad Wardman, Eric Nunes, Jakub Burgis, Ali Zand, Kurt Thomas, Adam Doupé, and Gail-Joon Ahn. 2020. Sunrise to sunset: Analyzing the end-to-end life cycle and effectiveness of phishing attacks at scale. In *29th USENIX Security Symp.* 361–377.

[33] Alina Oprea, Zhou Li, Robin Norris, and Kevin Bowers. 2018. Made: Security analytics for enterprise threat detection. In *Proc. of the 34th Annual Computer Security Applications Conference*. 124–136.

[34] PANW. 2022. PANW Advanced URL Filtering. https://www.paloaltonetworks.com/network-security/advanced-url-filtering.

[35] Peng Peng, Limin Yang, Linhai Song, and Gang Wang. 2019. Opening the blackbox of virustotal: Analyzing online phishing scan engines. In *Proc. of the Internet Measurement Conference*. 478–485.

[36] Phishtank. 2021. Phishtank. https://www.phishtank.com/.

[37] Li Qiang, Jiang Zhengwei, Yang Zeming, Liu Baoxu, Wang Xin, and Zhang Yunan. 2018. A Quality Evaluation Method of Cyber Threat Intelligence in User Perspective. In *2018 IEEE TrustCom/BigDataSE*. 269–276.

[38] Sivaramakrishnan Ramanathan, Jelena Mirkovic, and Minlan Yu. 2020. BLAG: Improving the Accuracy of Blacklists. In *NDSS*.

[39] Takaya Saito and Marc Rehmsmeier. 2015. The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PLOS ONE* 10, 3 (03 2015), 1–21.

[40] Muhammad N Sakib, Chin-Tser Huang, and Ying-Dar Lin. 2020. Maximizing accuracy in multi-scanner malware detection systems. *Computer Networks* 169 (2020), 107027.

[41] Aleieldin Salem. 2021. Towards Accurate Labeling of Android Apps for Reliable Malware Detection. In *11th ACM CODASPY*. 269–280.

[42] Aleieldin Salem, Sebastian Banescu, and Alexander Pretschner. 2021. Maat: Automatically Analyzing VirusTotal for Accurate Labeling and Effective Malware Detection. *ACM TOPS* 24, 4 (2021), 1–35.

[43] Marcos Sebastián, Richard Rivera, Platon Kotzias, and Juan Caballero. 2016. Avclass: A tool for massive malware labeling. In *RAID*. 230–253.

[44] Mahmood Sharif, Jumpei Urakawa, Nicolas Christin, Ayumu Kubota, and Akira Yamada. 2018. Predicting impending exposure to malicious content from user behavior. In *Proc. of the 2018 ACM CCS*. 1487–1501.

[45] Riccardo Silini and Cristina Masoller. 2021. Fast and effective pseudo transfer entropy for bivariate data-driven causal inference. *Scientific reports* 11, 1 (2021), 8423.

[46] Ravindu De Silva, Mohamed Nabeel, Charitha Elvitigala, Issa Khalil, Ting Yu, and Chamath Keppitiyagama. 2021. Compromised or Attacker-Owned: A Large Scale Classification and Study of Hosting Domains of Malicious URLs. In *30th {USENIX} Security Symposium*.

[47] S. Thirumuruganatha, M. Nabeel, E. Choo, I. Khalil, and T. Yu. 2022. SIRAJ: A Unified Framework for Aggregation of Malicious Entity Detectors. In *2022 IEEE Symposium on Security and Privacy*. 507–521.

[48] Kurt Thomas, Elie Bursztein, Chris Grier, Grant Ho, Nav Jagpal, Alexandros Kapravelos, Damon McCoy, Antonio Nappa, Vern Paxson, Paul Pearce, et al. 2015. Ad injection at scale: Assessing deceptive advertisement modifications. In *IEEE S&P*. 151–167.

[49] Ke Tian, Steve T. K. Jan, Hang Hu, Danfeng Yao, and Gang Wang. 2018. Needle in a Haystack: Tracking Down Elite Phishing Domains in the Wild. In *Proc. of the Internet Measurement Conference 2018*. 429–442.

[50] Kevin van Liebergen, Juan Caballero, Platon Kotzias, and Chris Gates. 2023. A Deep Dive into the VirusTotal File Feed. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 155–176.

[51] VirusTotal, Subsidiary of Google. [n.d.]. Free Online Virus, Malware and URL Scanner. https://www.virustotal.com/.

[52] VirusTotal, Subsidiary of Google. [n.d.]. VirusTotal URL Scan Reports. https://developers.virustotal.com/v2.0/reference/url-report.

[53] Thomas Vissers, Jan Spooren, Pieter Agten, Dirk Jumpertz, Peter Janssen, Marc Van Wesemael, Frank Piessens, Wouter Joosen, and Lieven Desmet. 2017. Exploring the Ecosystem of Malicious Domain Registrations in the .eu TLD. In *RAID*. 472–493.

[54] Haoyu Wang, Zhe Liu, Jingyue Liang, Narseo Vallina-Rodriguez, Yao Guo, Li Li, Juan Tapiador, Jingcun Cao, and Guoai Xu. 2018. Beyond Google Play: A Large-Scale Comparative Study of Chinese Android App Markets. In *Proc. of the Internet Measurement Conference*. 293–307.

[55] Wei Yang, Deguang Kong, Tao Xie, and Carl A Gunter. 2017. Malware detection in adversarial settings: Exploiting feature evolutions and confusions in android apps. In *33rd ACSAC*. 288–302.

[56] Shuofei Zhu, Jianjun Shi, Limin Yang, Boqin Qin, Ziyi Zhang, Linhai Song, and Gang Wang. 2020. Measuring and Modeling the Label Dynamics of Online Anti-Malware Engines. In *29th USENIX Security Symposium*.

APPENDIX

A LIST OF VIRUSTOTAL SCANNERS IN DATASET

Abusix, ADMINUSLabs, AICC (MONITORAPP), Alexa, AlienVault, alphaMountain.ai, Antiy-AVL, Armis, AutoShun, Avira, BADWARE.INFO, Baidu-International, BenkowCC, BforeAi, BitDefender, Blueliv, Certego, CINS, CMC Threat Intelligence, CRDF, C-SIRT, CLEAN MX, Comodo Valkyrie Verdict, Cyan Digital Security, CyberCrime, CyRadar, desenmascara.me, DNS8, Dr.Web, EmergingThreats, Emsisoft, ESET, ESTsecurity, Forcepoint ThreatSeeker, Feodo Tracker, FraudSense, Fortinet, G-Data, Google Safebrowsing (GSB), GreenSnow, IPSum, Hoplite Industries, Lumu, K7AntiVirus, Lionic, Kaspersky, MalBeacon, Malekal, Malsilo, Malware Domain Blocklist, Malware Domain List, MalwarePatrol, Malwarebytes hpHosts, Malwared, Malwares.com, Netcraft, NotMining, OpenPhish, Palevo Tracker, Phishlabs, Phishtank, Prebytes, Quickheal, Quttera, Rising, Sangfor, SafeToOpen, Scantitan, SCUMWARE.org, SecureBrain, Sophos, Spam404, SpyEye Tracker, Spamhaus, StopBadware, Sucuri SiteCheck, ThreatHive, Trend Micro Site Safety Center, Trustwave, urlQuery, Virusdie External Site Scan, VX Vault, Web Security Guard, Wepawet, Yandex Safebrowsing, Zeus Tracker, Zvelo, Botvrij.eu, Artists Against 419, Nucleon, Ransomware Tracker, URLhaus, Webroot, ZeroCERT, securolytics

B VIRUSTOTAL REPORT EXAMPLE

"url": "example.com", "scan_date": "2021-04-30 23:00:17", "positives": 3, "scan_id": "454..312" "first_seen": "2021-04-30 23:00:17", "Response content SHA-256": "d5a89...62a3", "scans": {"VT scanner1": {"detected": true, "result": "malicious site"}, "VT scanner2": {"detected": true, "result": "malicious site"}, "VT scanner3": {"detected": true, "result": "malware site"}, "VT scanner4": {"detected": false, "result": "clean site"}, "VT scanner5": {"detected": false, "result": "clean site"},....., "VT scanner95": {"detected": false, "result": "clean site"}}

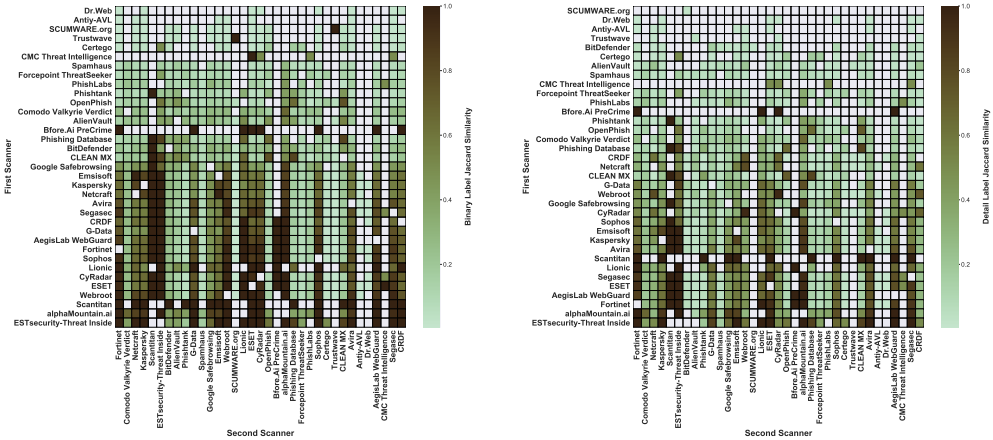
C MANUAL GT URL COLLECTION AND MANUAL LABELING PROCESS

To collect the set of URLs for Manual GT, we choose the fresh URL samples using a stratification sampling approach proposed in [5] and [19]. In doing so, we consider multiple dimensions of strata such as VT positive count and popularity (the number of VT rescan queries made in the first_seen date). The URLs are then manually labeled by 5 domain experts immediately. Specifically, experts individually visit the set of URLs using multiple browsers including Chrome, Opera, Firefox, and Safari, and manually classify the attack types. To achieve better confidence in labeling, all URLs are labeled by two experts and exclude URLs with conflicting labels. If the URL is NX, the URL is filtered from the list of URLs to analyze. If the URL is not NX, experts classify the type of URLs with the rules including the following.

- Check the URL address, forms, brand logos, redirections to identify phishing URLs.
- Check for associated files hosted in the URL to identify malware URLs. Download the file and check if the file is malware or not. In doing so, we perform the similar process to [56] and we also check the file against multiple Anti-virus engines including Sophos and McAfee desktop engine.
- Check if popular brand names or their variants being present in the URL address.
- Check the screenshots saved in the historical databases such as Internet Wayback Machine and urlscan.io.
- Check the detailed threat report by OTX [4] and McAfee WebAdvisor [27].

- If none of the above malicious indicators of compromise are present for a URL and the URL has been operational for at least 3 months, we mark the URL as benign.
- If the landing page is legitimate (e.g., known popular URLs such as https://outlook.live.com/owa/ and https://abc7news.com/weather/), we mark the URL as benign.
- Experts repeatedly check the changes of contents located in the URLs for 3 days. We further check the content changes for 30 days by collecting the content hashes from VT. If contents changed over time, we filter the URLs.

D HEATMAPS FOR SCANNERS' PAIRWISE JACCARD SIMILARITY OF BINARY AND DETAILED LABELS AND SCANNER CLUSTERING

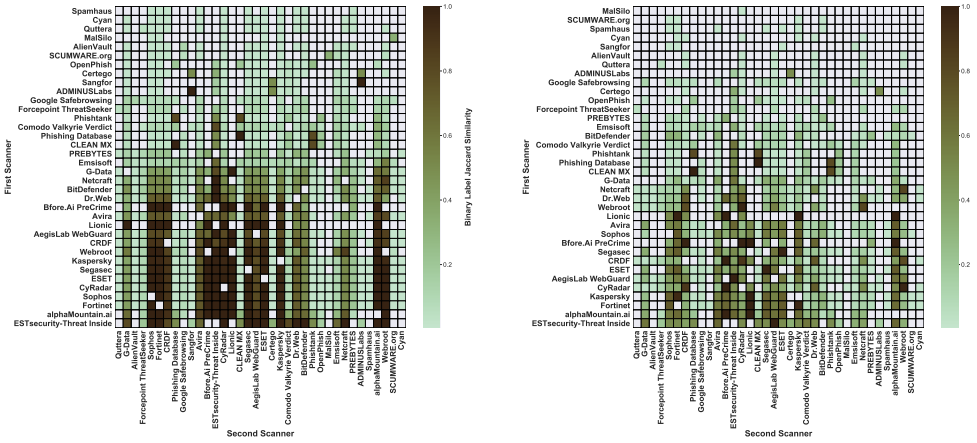


(a) Phishing - Binary Label Similarity

(b) Phishing - Detail Label Similarity

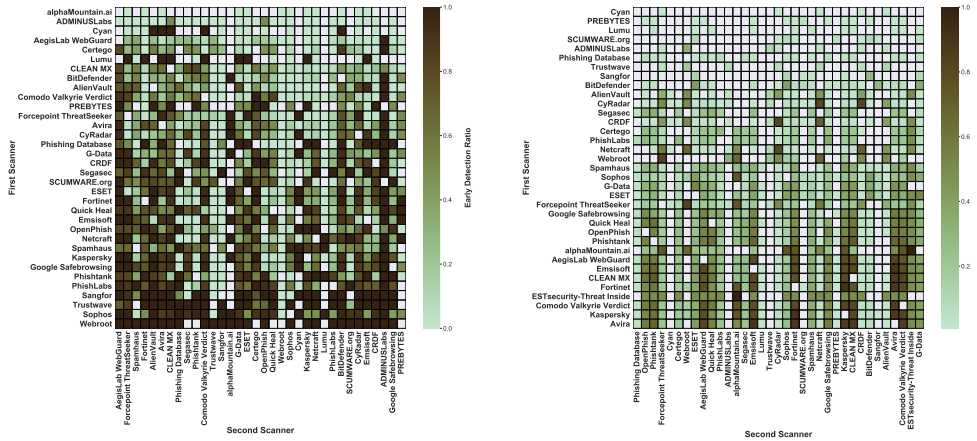
Fig. 14. Phishing - Jaccard similarity of scanners's binary and detail labels for all periods

Received February 2023; revised October 2023; accepted October 2023



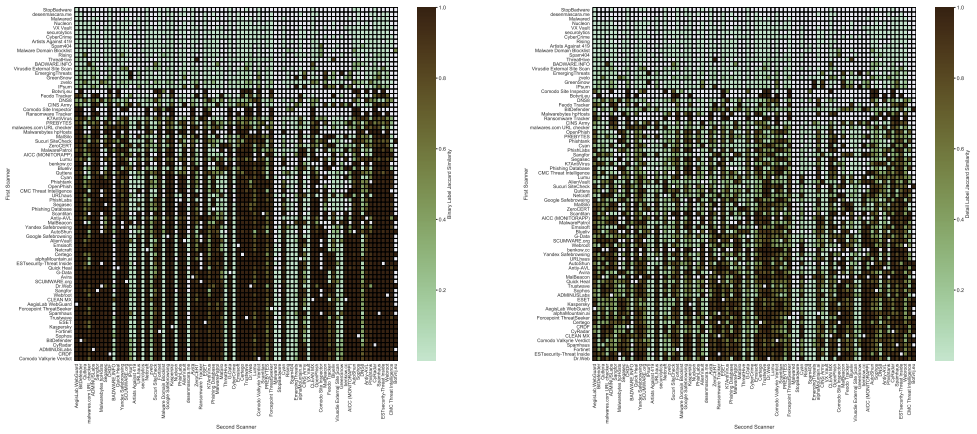
(a) Malware - Binary Label Similarity (b) Malware - Detail Label Similarity

Fig. 15. Malware- Jaccard similarity of scanners' binary and detail labels for all periods



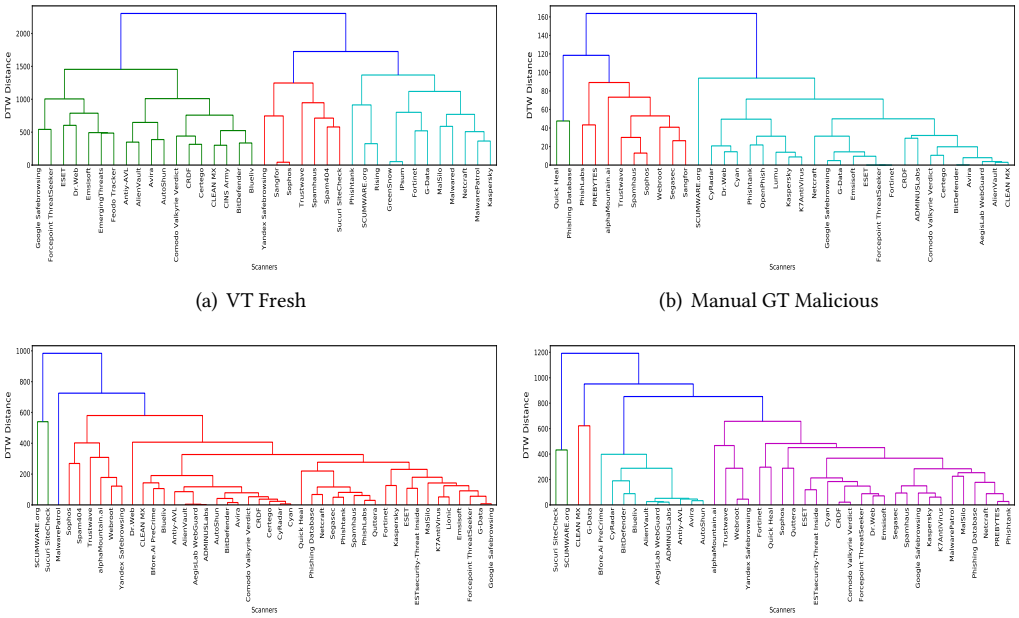
(a) Manual GT Malicious - Binary Label Similarity (b) Manual GT Malicious - Detail Label Similarity

Fig. 16. Manual GT Malicious - Jaccard similarity of scanners's binary and detail labels for all periods



(a) VT fresh URLs - Binary Label Similarity (b) VT fresh URLs - Detail Label Similarity

Fig. 17. VT fresh URLs - Jaccard similarity of scanners's binary and detail labels for all periods



(a) VT Fresh (b) Manual GT Malicious

(c) Phishing (d) Malware

Fig. 18. Scanner clustering using dynamic time warping distance