Identifying Coordination of Network Scans Using Probed Address Structure

Johan Mazel[1,2], Romain Fontugne[3], Kensuke Fukuda[1,4]
[1] NII ; [2] JFLI ; [3] IIJ Research Lab ; [4] Sokendai

***Abstract*—A great deal of work has been devoted to the study and detection of scanning. Existing detection of isolated probing, however, only provides an incomplete picture of scanning activities. Coordinated probing using several hosts, in particular, cannot be accounted for with simple scan detection that expects a single source. In this paper, we apply run-length encoding concepts to characterize the IP address structure of scanning events. We then employ graph techniques to uncover hidden coordinated network scans as communities. These coordinated events are split in accordance with the destination port and targeted network prefixes. We evaluate the sensitivity of our method with synthetic data and verify that our method outperforms the current state-of-the-art approaches for both stub and backbone network monitoring. Finally, we provide a detailed analysis of several coordinated scans occurring in real network traffic. Using these results, we verify that our method is reliable and extracts coordinated scans that are very consistent in terms of network traffic characteristics.**

I. Introduction

Network scans, also called horizontal scans in the literature, are the primary reconnaissance technique to acquire information on networks such as active hosts or services (ports). Existing large-scale studies [2], [10], [12], [15], [22], [26], [28] focus on single source probing activities. They do not study coordinated scan despite evidence of their occurrences (cf. analysis of a /0 scan [9] or the Internet Census [1] and existing tools [11], [23]). For perpetrators, this probing technique offers many advantages over single-source scans. One can probe bigger networks at a faster pace more stealthily. Furthermore, splitting probing across several sources lets scan detection techniques detect only a small number of isolated events, if any. The complete probing scope remains hidden.

Our goal is to provide a coordinated scan identification method for both stub and backbone traffic monitoring. In other words, we aim for clustering groups of previously detected single-source network scans that operate in a coordinated fashion. While existing coordinated scan identification approaches target stub networks, backbone traffic monitoring offers a broader point of view on network traffic. This means that backbone traffic monitoring captures a bigger proportion of scanning IP's activity. In return, the method needs to account for phenomena such as incomplete traffic due to asymmetric routing, which impedes stateful analysis. Several works [3], [14] have addressed the identification of coordinated scanning in the context of stub network monitoring. Gates [14] proposed a greedy aggregation method that groups scans starting with the ones with the fewest destinations while ensuring properties such as high coverage of destination IP address space. In the context of backbone traffic monitoring, unrelated small scans that target contiguous prefixes belonging to distinct entities may however be grouped together. Baldoni et al. [3] presented a method that identifies coordinated scans as connected components of a graph where each edge links two source IP addresses that generate failed connections to contiguous destination IP addresses. This neighboring criterion may however not always be verified, especially in the context of backbone traffic monitoring.

We propose a novel approach that reliably aggregates previously detected single-source network scans into coordinated events. It relies on two criteria that leverage network scans destination IP address structure. Our approach employs a graph-based mining technique to avoid the shortcoming of the greedy aggregation method [14] while our single-source scan association criteria are more reliable than that used in [3]. Moreover, our proposal exhibits a very good versatility: it can be applied on both local and backbone network traffic monitoring.

Our contribution is twofold. First, we propose a coordinated scan identification method that uses simple criteria to associate single-source scans into a graph where coordinated scans are mined as communities. Using synthetic data, we show that our method outperforms the existing work [3], [14] for random and interleaving scans. Our second contribution is to analyze coordinated scans observed in backbone traffic and describe several case studies.

II. Related work

Single-source network scan detection has generated a lot of research. Jung et al. [19] proposed a scheme that relies on the assumption that scanning IPs (also called scanners) will have a higher rate of unsuccessful connections than legitimate Internet hosts. We refer the reader to the work of Bhuyan et al. [4] for a more complete description of the state-of-the-art single-source scan detection.

This work is supported by R&D Promotion Program of the Ministry of Internal Affairs and Communication, Japan, and EU FP7/2007-2014, grant 608533 (NECOMA).

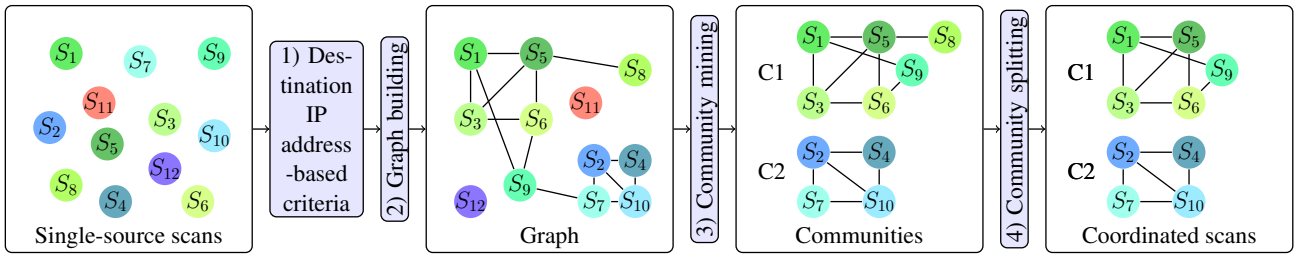


Fig. 1. Method overview (1: destination IP address-based criteria associate network scans, see Section III-A; 2: criteria-based graph is built, see Section III-B; 3: coordinated scans are mined as communities, see Section III-C; 4: communities are split, see Section III-D).

Some approaches directly analyze traffic. Treurniet [25] uses network traffic's distributed nature and protocol behavior to find anomalies. Anomalies are then iteratively extracted as group of sessions. Coordinated scans are extracted after network scans. This means that some elements of coordinated network scans may be wrongly considered as isolated scans while extracted coordinated scans may be incomplete. Baldoni et al. [3] propose a collaborative architecture to identify coordinated scans. Local sensors generate alarms that are analyzed by a central entity to find distributed activities. Baldoni et al., however, assume that scanning activities generate failed connections to contiguous destination IP addresses. This is a restrictive hypothesis, especially for backbone traffic.

Another existing approach is aggregating previously detected scans to uncover coordinated scans. Bou-Harb et al. proposed several techniques to merge similar probing activities. One [8] is a statistic-based technique that uses many criteria such as a probing technique, probing strategy, and malware activity, while another [6] leverages probing temporal patterns. These criteria however do not ensure that similar scans target the same region of the IP address space. Similar single-sources scans may thus correspond to several perpetrators using the same tool on different targets. For HTTP scanning detection, Xie et al. [27] use a bipartite graph that associates source IP addresses with HTTP resource paths. Coordinated HTTP scans are then extracted as clusters from this graph. This approach is however difficult to translate to network scans because the only available information is network and transport protocol headers. Relying on only this may cause a high number of false positives. Gates [14] proposed a coordinated scan identification method for stub network monitoring. It applies a set coverage technique to destination IP address sets of previously detected network scans. It greedily merges small scans together while ensuring a small overlap between scans. Unrelated small scans may thus be erroneously grouped into coordinated events. Furthermore, for single-point backbone traffic monitoring, a subset of all sources of a coordinated scan may be observed. This means that even if a network is targeted, observed probing packets may only reach a subset of this network. This clearly makes their coverage-related criteria (see Section IV-D2) unfit for single point backbone traffic monitoring. For a detailed survey of coordinated scan identification techniques, we refer the reader to the works of Bhuyan et al. [4] and Bou-Harb et al. [7].

The state-of-the-art most similar to our proposal is that of Baldoni et al. [3]. Our network scan association criteria are however more reliable. Our approach can thus be applied in both stub and backbone traffic monitoring scenarios. Other existing methods do not exhibit such versatility.

III. Mining Coordinated Scans

Coordinated scanning is initiated by many sources controlled by a single entity. It increases scanning speed and reduces detection odds while maintaining a small probing footprint. The goal of this work is to *uncover coordinated scans* from *previously detected single-source scans*. In other words, we aim to find groups of network scans, which, as a whole, target a large number of IP addresses in an orchestrated manner. The rationale of our method is to use criteria to compare single-source scans and extract groups of events whose behavior appears to be coordinated. Figure 1 illustrates the overview of our proposed algorithm. First, we apply two criteria (destination IPs overlap-based and destination IPs structure-based, see Section III-A) to build a graph representing relationships between scans (see Section III-B). We then extract groups of strongly connected nodes from the graph (see Section III-C) representing coordinated events. Finally, we perform a post-processing splitting step that aims at extracting consistent groups of scans (see Section III-D).

A. How to correlate network scans

The first question that arises when one intends to correlate network scans is what criteria will determine whether two events are actually associated. The next two subsections describe the criteria we use.

1) Destination IP addresses overlap-based criterion: To acquire knowledge on a network, scanners send probing packets to all the IP addresses inside the targeted network. The optimal strategy is to send packets to each destination without probing the same destination more than once. The perpetrator of a coordinated network scan divides the probing load among multiple scanners and ensures that they do not target the same IP address. The perpetrator may however attempt to hide the coordination between his scanners by purposely generating small overlaps between targeted IP addresses of controlled scanners. To account

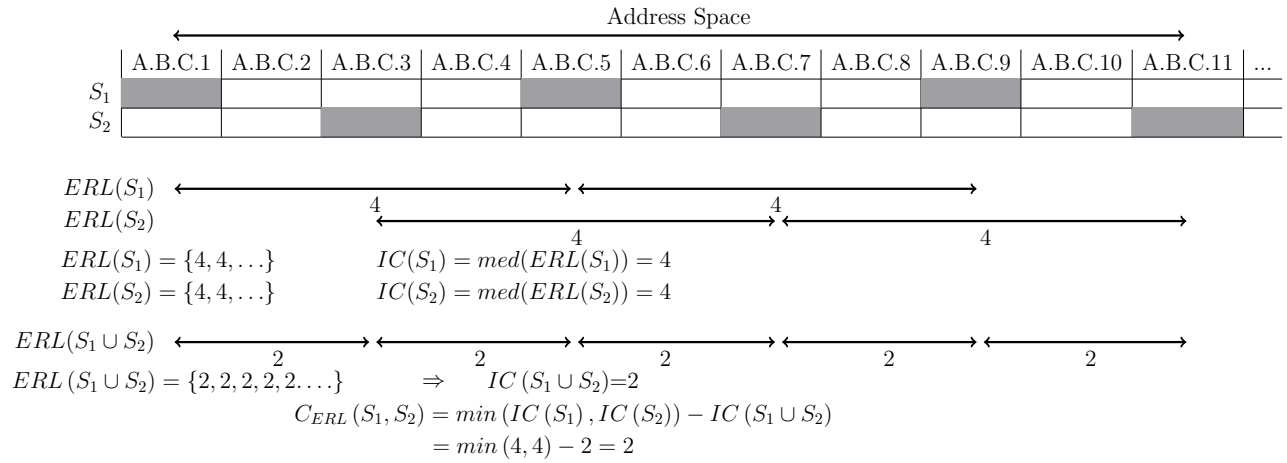


Fig. 2. IP addresses probed (grey cells) by two scans S_1 and S_2 and associated Empty Run-Length ERL, IP address Coverage IC and ERL-based criterion C_{ERL} values. The notation A.B.C.* means that all IP addresses are located in the same /24 network.

for this behavior, our algorithm allows overlapping between sets of destination IP addresses of scanning events. Let us consider two network scans S_i and S_j and their destination IP addresses sets. We define the overlap Ω between S_i and S_j as the Jaccard index J, a standard measure of set similarity:

$$\Omega(S_i, S_j) = J(S_i, S_j) = \frac{|S_i \cap S_j|}{|S_i \cup S_j|}. \quad (1)$$

We define a threshold T_Ω to control the overlap between S_i and S_j: $\Omega(S_i, S_j) \leq \mathrm{T}_\Omega$. This is our first criterion.

2) Destination IP address-space structure-based criterion: As explained above, the perpetrator of a coordinated network scan splits the probing load among several controlled scanners. To aggregate previously detected network scans, we analyze the structure of the destination IP address sets of each scan. Several scanning patterns have been documented such as consecutive [2], interleaving [23], or reversed byte order [9]. The emergence of high-speed scanning tools [11], [16], [20] that use pseudorandom patterns and studies of darknet traffic [8], [13] show that non-consecutive scanning constitutes a non-negligible proportion of probing events. Specifically, Bou-Harb et al. [8] found that 66% of events in 2013 were random, while Fukuda et al. [13] showed that, in November 2006, 10-15% behaved randomly. Leonard et al. [21] also proved that the carefully designed non-consecutive probing pattern they used [20] minimizes detection odds by state-of-the-art IDS (Snort and Bro). We thus hypothesize that coordinated scans use non-consecutive patterns.

The goal of probing is to reach as many destinations as possible in an IP address range in order to have a complete picture of an entity's network. In the context of coordinated probing, each source thus probes a subset of the considered range. We first define an index derived from run-length encoding called *empty run-length* or ERL. It measures the gap between two IP addresses. ERL values of a network scan S, noted ERL(S), are the successive ERL values between targeted destination IP addresses. To assess the improvement in address coverage that the merging of two scans provides, we then define an index called IP address space Coverage (IC) based on the index defined above. The IC value of a scan S is obtained by computing the median of its ERL values and is defined as:

$$IC(S) = med(ERL(S)). \quad (2)$$

Thanks to the median operator, IC is robust against unusually high ERL values caused by missing packets. IC takes positive integer values (i.e. $IC > 0$). Figure 2 displays a generic example with two interleaving network scans. The empty run-length values are: $ERL(S_1) = \{4, 4, 4, \ldots\}$, $ERL(S_2) = \{4, 4, 4, \ldots\}$, and $ERL(S_1 \cup S_2) = \{2, 2, 2, \ldots\}$, and the corresponding IC values are $IC(S_1) = IC(S_2) = 4$ and $IC(S_1 \cup S_2) = 2$. The value of $IC(S_1 \cup S_2)$ is lower than the two scans considered in isolation, meaning that these two scans operate in such a way that the coverage of their union is better than that of events considered separately. We leverage this characteristic of IC to build the index for our second criterion. C_{ERL} is defined as:

$$C_{ERL}(S_i, S_j) = \min(IC(S_i), IC(S_j)) - IC(S_i \cup S_j). \quad (3)$$

We consider that two network scans S_i and S_j are correlated when $C_{ERL}(S_i, S_j) > 0$. In Figure 2, $IC(S_1) = IC(S_2) = 4$ and $IC(S_1 \cup S_2) = 2$. These two scans operate in a coordinated manner. Here $C_{ERL}(S_1, S_2) = 2$, our criterion is thus verified.

B. Graph building

We build a graph (see step 2 of Figure 1) where each vertex represents a previously detected network scan and each edge associates two network scans that satisfy the two criteria defined in Sections III-A1 (Ω) and III-A2 (C_{ERL}). The generated graph is unweighted and undirected. This graph represents probed address relationships between detected scans.

C. Community extraction

In the generated graph, coordinated scans are represented as groups of vertexes that are well connected with each other. Ideally, inside each group of vertexes, every pair of vertexes should be connected, thus forming a clique. However, traffic monitoring sometimes only allows partial observation of probing activities. Coordinated scans may thus be represented as densely connected vertexes, i.e. communities. Community mining has been extensively studied and several methods are available. In our case, we use the Louvain algorithm [5] (see step 3 of Figure 1) to partition the graph into communities.

D. Community splitting

After the community mining, scan groups represent events that target the same region of the IP address space in a complementary manner. We may, however, only partially observe single-source scans of a coordinated event. Unrelated single-source scans that probe the same prefix and whose destinations exhibit a small overlap (see Section III-A1) with the targets of a coordinated event, may thus be associated with this group. We cope with this issue by splitting communities into group of scans with consistent network traffic characteristics (see step 4 of Figure 1).

First, we determine whether a community should be split. A community of scans should be split if their destination ports or their targeted network prefixes are too different. We thus build scanners' destination port distributions and check whether they are similar. Our criterion relies on the Jensen-Shannon Divergence (JSD), a standard similarity measure between probability distributions:

$$JSD_{\pi_1,\ldots,\pi_n}(P_1,\ldots,P_n) = H\left(\sum_{i=1}^{n}\pi_i P_i\right) - \sum_{i=1}^{n}\pi_i H(P_i) \tag{4}$$

where $\pi_1,\ldots,\pi_n$ are weights for the distributions $P_1,\ldots,P_n$, and H is the normalized Shannon Entropy. $JSD = 0$ when all P_i are identical and $JSD \neq 0$ when P_i differ. We here use $\pi_n = \frac{1}{n}\ \forall i \in 1\ldots n$. If $JSD_{\pi_1,\ldots,\pi_n}(P_1,\ldots,P_n) \geqslant \mathrm{T}_{JSD}$, we consider that this community is inconsistent.

Regarding targeted destination IP addresses, we first build the smallest network prefix that contains all destinations of all scans in the considered community. We then compare the prefix length of the network prefix previously built with the length of each individual scans's targeted prefix. We then define the prefix distance d_P between two prefixes NP_i and NP_j whose prefix lengths are l_i and l_j as: $d_P(NP_i, NP_j) = |l_i - l_j|$. If $d_P \geqslant \mathrm{T}_{d_P}$, we consider that the community is inhomogeneous. If either of these two criteria is verified, we split the community.

During the splitting phase, we examine every existing edge in the considered community. We reuse and adapt the above destination port and network prefix criteria. The destination port criterion remains the same but is here applied to only two scans (the ones linked by the considered edge). The network prefix criterion is slightly modified. If either scan's prefix is included in or equal to that of the other, we verify that $d_P < \mathrm{T}_{d_P}$. If there is no inclusion between prefixes, the prefix-based criterion is not met. Any edge that meets no criteria is removed.

Finally, we examine source IP addresses of scans inside communities. If all scans in one community have the same source IP address, this community represents the activity of a single scanner, so we discard the whole group.

IV. Evaluation with synthetic data

To validate our approach, we first perform a sensitivity analysis of the proposed method regarding parameters of the community splitting phase (cf. Section III-D). We then compare our method with those of Baldoni et al. [3] and Gates [14]. The evaluation relies on synthetic data due to lack of publicly available ground-truth data. Throughout the paper, we use the conservative overlap threshold $\mathrm{T}_\Omega = 0$ to avoid potential false positives.

A. Synthetic data generation

The goal of our algorithm is to extract coordinated events from previously detected network scans. We thus generate both isolated and coordinated scans. The former is composed of unrelated single-source scans, and the latter is composed of several complementary single-source scans. Parameters of isolated and coordinated scans are randomly generated. They both target full network prefixes that are generated by choosing a random IP address inside a prefix P and a random prefix length between an upper and lower bounds. The values of prefix P and prefix size bounds are explained below. In our scenario, every scanner sends a single packet to each destination IP address. The sources number n (or scanners number) inside coordinated scans follows a uniform distribution between 20 and 100. We choose to generate 300 isolated scans and a variable number of coordinated scans. We generate a large number of both types of scans, and results are similar across our experiments. The chosen values for the number of scans reduce algorithms' workload. We use two scanning algorithms for coordinated scans: randomly spread across sources as in ZMap [11] (using the Fisher-Yates shuffle) and interleaving pattern of length n (= the source number) as in NSAT [23]. Half the generated coordinated scans follow the random pattern, and the rest the interleaving pattern, There is no overlap between scanners of a coordinated scan.

B. Performance metrics

Coordinated scan identification methods extract groups of single-source scans. This is thus a classification problem with several classes. Several mappings of the found groups to the ground-truth are used in the literature. We here associate each identified scan group with the coordinated scan from the ground-truth with which it intersects the most. Formally, an extracted group of scans G_i is associated with a ground-truth coordinated scan G_j^{gt} if:

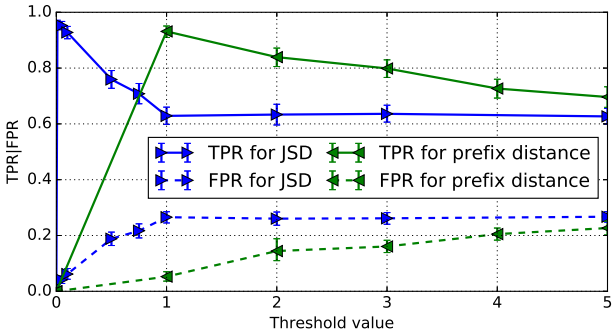


Fig. 3. Mean of True Positive Rates (TPR) (continue lines) and False Positive Rates (FPR) (dotted lines) in regards to the threshold on the Jensen-Shannon Divergence of destination port T_{JSD} (blue curves) and the threshold on the prefix distance T_{d_P} (green curves). Error bars represent a 95% confidence interval from 50 runs.

$$|G_i \cap G_j^{gt}| > |G_i \cap G_k^{gt}| \; \forall k \in \{1 \dots n\} - \{j\}. \quad (5)$$

Each identified group is only associated with a single coordinated scan from the ground-truth, and, reciprocally, each ground-truth event is associated with at most one identified group of scans. To this end, we successively associate identified and ground-truth scan groups by decreasing intersection size. The number of true positives is thus the sum of all the intersection sizes extracted with the above method. Every individual scan reported as belonging to a coordinated event, but not associated with a coordinated scan from the ground truth, is considered as a false positive. Similar methods are used in other n-class classification problem evaluation such as clustering.

C. Sensitivity analysis

We perform a sensitivity analysis on the parameters of the splitting phase presented in Section III-D. Throughout our experiments, we generate 50 runs for each parameter value and compute 95% confidence intervals. Small intervals thus mean that our results are consistent. Targeted prefix sizes of isolated and coordinated scans range from 16 to 24 and prefix P length is equal to 20.

Figure 3 presents the results of our sensitivity analysis. The blue curves describe our method's performance for different JSD thresholds T_{JSD}. Here, we do not use the prefix distance-based criteria. For $\mathrm{T}_{JSD} = 0$, all edges inside communities are removed. Every community thus disappears and our method does not identify any coordinated event. When T_{JSD} increases, communities that contain scans targeting different destination ports are split into consistent groups. As T_{JSD} further increases, fewer and fewer communities are split, and thus TPR decreases and FPR increases. The optimal value here is $\mathrm{T}_{JSD} = 0.01$.

The green curve depicts the performance of our method for several values of prefix distance threshold T_{d_P}. The JSD-based criterion is deactivated here. When $\mathrm{T}_{d_P} = 0$, the splitting criterion is always met, all edges inside all communities are removed. Consequently, our tool does not identify any coordinated events. Similarly to T_{JSD}, when T_{d_P} increases, our method's performance suddenly increases because our method is now able to remove edges between scans whose targeted prefix lengths (and thus, prefix) are different. When T_{d_P} continues to increase, our splitting step is less and less efficient. The optimal value here is $\mathrm{T}_{d_P} = 1$. We however use a slightly more conservative value for both thresholds in order to account for noise. We use $\mathrm{T}_{JSD} = 0.1$ and $\mathrm{T}_{d_P} = 2$.

D. Other approaches

1) Local attack graph coordinated scan identification [3]: This method is a two-step collaborative approach to identify coordinated scans. In the first step, local entities monitor stub networks and extract groups of source IP addresses that exhibit suspicious behaviors. In the second step, an algorithm merges alerts raised by local monitoring entities and performs a post-processing step to improve the extracted groups of source IP addresses. We however only use the first step of this method since we consider a single point measurement use case. This first step builds a graph, called a Local Attack Graph (LAG), where edges link source IP addresses that generate failed connections to the same port on contiguous destination IP addresses. In our case, we consider that all probing packets generate failed connections. Overlap in terms of targeted IPs is not allowed. This method then identifies coordinated scans as connected components in the previously built graph.

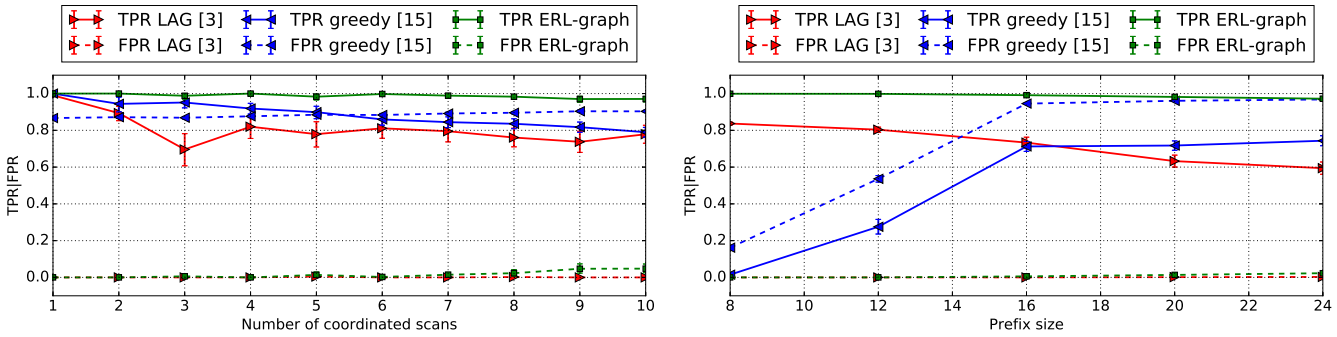
2) Greedy coordinated scan identification [14]: This method first performs a greedy aggregation of scans that target the same destination port number and exhibit a small overlap. The algorithm then ensures that an extracted group of scans verifies criteria regarding coverage, overlap, and hit rate. If it does not, the method removes individual scans from groups of events to reach these properties. Gates defines the coverage ζ of a group of scans $G = \{S_0, \dots, S_n\}$ that targets a network prefix P as $\zeta(G) = \frac{a_n - a_1 + 1}{|A|}$ where a_1 and a_n are the first and last IP addresses scanned in A (in IP-address space, not time). Hit rate $\mathcal{H}$ is defined as $\mathcal{H} = \frac{|A_C|}{a_n - a_1 + 1}$ where $|A_C|$ is the number of IP addresses targeted inside A. Gates [14] also defines the overlap as the Jaccard index J (see Section III-A1). This definition is however not suitable for a coordinated event that contains more than two scans. We thus generalize the Jaccard index and define the overlap θ of a coordinated scan G as

$$\theta(G) = \max\left(\frac{|S_i \cap \bigcup(G^i)|}{|S_i \cup \bigcup(G^i)|}\right) \quad (6)$$

where $G^i = \{S_1, \dots, S_n\} - \{S_i\} \; \forall i \in 1 \dots n$. Gates [14] does not provide any recommendation for these three thresholds. We thus choose them conservatively in order to reduce the false positive rate. Our implementation thus always ensures $\zeta(G) > 0.95$, $\mathcal{H}(G) > 0.95$ and $\theta(G) < 0.2$.

E. Performance comparison

Figure 4 presents the results of our synthetic evaluation. The first part of our evaluation (Figure 4a) simulates the



(a) TPR/FPR against number of synthetic coordinated scans

(b) TPR/FPR against prefix size

Fig. 4. Mean of True Positive Rates (TPR) (continue lines) and False Positive Rates (TPR) (dotted lines) in regards to the number of coordinated scans (a) and IP address space size where synthetic scans are located (b). Red curves are obtained with LAG [3], blue curves with the greedy approach [14], and green curves with our method. Error bars represent a 95% confidence interval from 50 runs.

monitoring of a stub network and thus fits the original context of [3], [14]. We generate synthetic coordinated scans that probe network prefixes whose sizes are between 16 and 20 that completely include the monitored network. This network is actually P (cf. Section IV-A) and its prefix size is 20. This inclusion ensures that the coverage ζ and hit rate $\mathcal{H}$ of generated coordinated scans will always be 1 and thus greater than the previously set threshold of 0.95 (cf. Section IV-D2). Isolated scans target smaller network prefix sizes between 20 and 24 inside P. Figure 4a shows the evaluation results when the number of synthetic coordinated scans varies from 1 to 10. The results obtained for one coordinated scan are consistent with those by Gates [14] regarding TPR. The greedy approach is able to reliably extract a single coordinated scan. This figure however clearly shows that the performances of the greedy method degrade as the number of coordinated scans increases. This behavior may be extremely detrimental to real world performance as it is highly unlikely that a single coordinated scan occurs at any given time. The observed FPR is much bigger in our experiments than in those of Gates [14]. This is likely due to the difference in terms of isolated scans. While we completely control isolated scans' characteristics, Gates [14] extracts scans from real network traffic. These scans may thus have had a much bigger coverage of P than the events we generated. The isolated scans used by [14] may thus overlap. This may have prevented single-source scans from being merged together due to the overlapping criterion θ. Regarding Baldoni et al. [3], as the number of coordinated scans increases, we observe that its TPR decreases. This phenomenon is due to coordinated scans with an interleaving pattern. Since all coordinated scans target the same network, unexpected links appear during the graph building. Connected component extraction then merges two (or more) coordinated scans together. The results of our method are very good in the context of stub network monitoring.

Figure 4b presents the second part of our evaluation. We here define a setup similar to backbone traffic monitoring. We here randomly discard half the single-source scans of each coordinated scan to emulate the fact that single point backbone monitoring may only partially observe the traffic targeting a given prefix. Targeted prefix sizes of isolated and coordinated scans now range from 16 to 24. We here gradually decrease the size of P from a prefix length of 24 to 8. When the prefix size of P is 24, all scans will target the same network. Consequently, they will be much more spread out in the IP address space, when the prefix length of P is 8. We here generate 10 synthetic coordinated scans. The greedy method here cannot use the coverage criterion ζ due to all coordinated scans not necessarily targeting a specific network prefix. We thus here do not use ζ but keep using the hit rate criterion $\mathcal{H}$ and the overlap criterion θ. For sizes of P greater than or equal to 16, the greedy method's performances remain similar to those obtained for stub network monitoring. However, when P size is smaller than 16, performance quickly degrades. This is due to the step that tries to remove noise (single-source scans wrongly aggregated) in order to increase the hit rate $\mathcal{H}$. To this end, the greedy method performs the following step: the biggest gap is found and the set of aggregated scans is split in half. Scans located in the smallest half are removed. Here, coordinated scans tend to be grouped with more single-source events that do not overlap with them due to them being increasingly spread in the IP address space. Coordinated events thus have higher odds of being discarded during the noise removal step described above. Similarly, groups of single-source scans that generated false positives for smaller P size are spread in the IP space when P size increases and thus yield smaller FPRs. The LAG approach here yields results worse than those of the stub network scenario. It suffers from its use of a criterion on destination IP address contiguousness. The discarded single-source scans remove some contiguousness between scans belonging to the same group. Coordinated scans with interleaving patterns are thus often split into several components. Our method's results here are very satisfactory. Our method achieves higher TPR rates than both the greedy [14] and LAG approaches [3] and much lower FPR than the greedy method and similar FPR with LAG.

V. Coordinated scans in the wild

A. Dataset

We analyze network traffic traces from the MAWI repository, which is a collection of daily 15-minute-long traces captured on a backbone link connecting Japanese universities and research institutions to the Internet. Ten prefixes of lengths ranging from 16 to 24 are visible. The MAWI repository mainly consists of international traffic between universities and commercial ISPs captured since January 2001. In this work, we use Day in the Life of the Internet (DITL) traces, which are part of a worldwide effort of simultaneous traffic data collection. The MAWI repository contains DITL traces that last at least 24 hours and are captured on the same link used to perform the daily measurements. Unlike the publicly available MAWI traces where IP addresses are anonymized, our dataset contains original IP addresses to monitor scanners across different traces. We remove traffic corresponding to the IP addresses of the outage detector, Trinocular [24], because it significantly increases the workload of our analysis without providing new information.

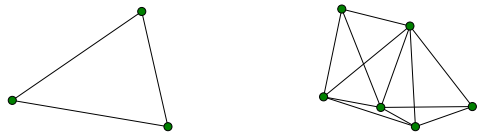
Abnormal events appearing in the MAWI repository are automatically reported in the MAWILab [12] database and then classified and annotated with a taxonomy designed for network backbone anomalies [22]. In this paper, we make use of these results and study the characteristics of traffic annotated with *network scan* labels (i.e. labels with the prefix $ntsc$). These labels ensure that corresponding traffic has a single source and a high number of destinations (> 20). Protocol header information (SYN, ACK, FIN flags for TCP and ICMP type Echo request, Netmask request and Timestamp request for ICMP) is also used to identify different types of network scan. The taxonomy ensures that each destination receives fewer than 15 packets.

To assess the reliability of MAWILab, we compare the source IP address of events annotated as network scans in the MAWI traces with the IP addresses reported by the SANS Internet Storm Center (ISC) [17] from November 2014 to March 2015. 55% of IP addresses in MAWILab are also present in ISC's suspicious domains. This shows that most IP addresses labeled as scans are also detected by the firewalls participating in the DShield project.

B. Case study

In this section, we analyze backbone traffic traces and present several case studies. TCP network scans are more prevalent than UDP and ICMP ones. We thus focus on TCP.

We analyze the 72-hour-long 2013 DITL trace captured from June 25th to 27th. MAWILab identifies 3132 single-source scans originated from 1484 distinct source IP addresses. Our method extracts 22 coordinated events that contain 844 single-source scans performed from 388 source IP addresses. Further analysis of coordinated and isolated scans' durations reveals that coordinated events last longer than isolated ones. Furthermore, more than 60% of coordinated scans last more than 24 hours. The other coordinated scans are much shorter, typically lasting less than 6 minutes. This difference is also apparent for targeted network prefixes: long-lasting scans' prefix sizes are between 0 and 16, while short-lasting scans' prefix sizes are between 7 and 22. Short lasting coordinated scans also have fewer sources (2 or 3), while long coordinated events contain between 2 and 314 scanners.



(a) SSH short coord. scan (b) SMB long coord. scan

Fig. 5. Coordinated scan case studies from 2013 DITL trace: (a) SSH coordinated scan and (b) SMB (and others) coordinated scan.

TABLE I
Examples of coordinated scans in MAWI (OC: Overall Coverage; SD: standard deviation, Dur.: duration). Destination ports: 22 - SSH ; 445 - SMB ; 5900 - VNC.

# src IP	Dst prefix size	Dst prefix OC	Dst port	Dur.	# dst addr mean $\pm$ SD	Figure graph
3	/19	0.31	22	3s	850 $\pm$ 25.7	5(a)
3	/22	0.99	22	3s	340 $\pm$ 29.5	5(a)
6	/16	0.27	445	68.2h	462 $\pm$ 248.2	5(b)
314	/16	0.47	5900	57.5h	99 $\pm$ 23.4	

Table I details some of the coordinated scans found in the 2013 DITL trace. For each coordinated scan, we provide the size of the targeted network prefix, its overall coverage (i.e. percentage of destination IP addresses reached in the considered prefix), dominant destination port, duration, mean, and standard deviation of the number of destinations targeted by scanners and the reference of the associated graph in Figure 5. The first two coordinated scans of Table I are composed of the same three contiguous IP addresses. The two targeted prefixes are located inside the same /16 network. Scanners' activity periods are synchronized: they start and finish their probing within a 50 ms window. Figure 5(b) displays the community of the third event in Table I. Despite a graph density of 0.8, this community is not a clique. Missing links between hosts are due to small overlaps Ω between scanners. The fourth coordinated scan presented in Table I contains 314 distinct source IPs. This event seems to be linked to a sudden surge in the number of scanners targeting port 5900 that is observed by ISC's TCP/UDP Port Activity service [18] from June 22nd to 26th. This high number of scanners provides strong evidence that a botnet was involved. These examples show that our method groups scanners that exhibit similar characteristics in terms of duration and number of reached destination IP addresses. Since, we only use destination IP address structure and port information, this consistency further proves that our method is reliable. The targeted destination prefix coverages of these coordinated events also show that their scanners behave in an orchestrated manner.

VI. Discussions

The evaluation presented in Section IV shows that our method reliably extracts coordinated network scans in the context of both stub and backbone traffic monitoring and outperforms LAG [3] and greedy [14] approaches. By not relying on hypothesis such as destination IP addresses contiguousness [3] and complete coverage [14], our method remains robust in the context of backbone traffic monitoring. Furthermore, we are confident that extracted coordinated scans shown in Section V are actually real coordinated events due to the consistency of their network traffic-related characteristics. This shows that our method reliably extracts coordinated scans in real data. We do not assess the impact of T_Ω on results due to the lack of space but intend to address this aspect in future work.

Our method uses previously extracted network scans as input. We thus depend on the reliability of the probing detection. Our method, however, yields low FPRs (cf. Section IV). Scanning detection may thus be tuned in a sensitive way in order to avoid false negatives without impacting our method's performance.

Our two criteria (destination IP address overlap, and structure-based) provide reliable hint regarding hidden coordination patterns between network scans. These criteria are extremely simple to understand and process. Moreover, community mining uncovers meaningful structures among network scans that clearly represent coordinated activities. Along with the introduction of new criteria, the ability of our method to work on local and backbone network traffic (cf. Section IV) also represents a significant improvement w.r.t to the state-of-the-art methods. In addition to the work of Dainotti et al. [9], this work also provides further evidence that large groups of hosts perform coordinated probing.

Our splitting technique currently relies on destination port distribution and network prefix consistency as criteria to assess extracted communities' purity. One way to improve this step would be to use other criteria. We could for example use some criteria proposed by Bou-Harb [6], [8]. However, some criteria such as the probing rate should be avoided because they are easy to purposely falsify. We should therefore choose new criteria with extreme care.

VII. Conclusions

We propose a new method to accurately extract coordinated scans from previously detected scans. Our proposal efficiently associates previously extracted scanning events. We introduce two criteria that rely on destination IP address structure of network scans. A graph-based method then identifies scan groups as communities. Finally, we split communities into scan groups with consistent characteristics. Our synthetic data-based evaluation shows that our proposal outperforms existing approaches for random and interleaving scanning patterns in both stub and backbone traffic monitoring. An analysis of MAWI backbone traffic provides several use cases of real coordinated scans and reveals that coordinated scans last either a few minutes or tens of hours.

References

[1] Internet census. http://internetcensus2012.bitbucket.org/paper.html.

[2] M. Allman, V. Paxson, and J. Terrell. A brief history of scanning. In *Proceedings of IMC 2007*, pages 77–82, 2007.

[3] R. Baldoni, G. A. Di Luna, and L. Querzoni. Collaborative detection of coordinated port scans. In *Proceedings of ICDCN 2013*, pages 102–117, 2013.

[4] M. H. Bhuyan, D. K. Bhattacharyya, and J. Kalita. Surveying port scans and their detection methodologies. *Computer Journal*, 54(10):1565–1581, 2011.

[5] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. 2008.

[6] E. Bou-Harb, M. Debbabi, and C. Assi. A time series approach for inferring orchestrated probing campaigns by analyzing darknet traffic. In *Proceedings of ARES 2015*, pages 180–185.

[7] E. Bou-Harb, M. Debbabi, and C. Assi. Cyber scanning: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 16(3):1496–1519, 2014.

[8] E. Bou-Harb, M. Debbabi, and C. Assi. On fingerprinting probing activities. *Computers & Security*, 43:35 – 48, 2014.

[9] A. Dainotti, A. King, K. Claffy, F. Papale, and A. Pescapè. Analysis of a /0 stealth scan from a botnet. *IEEE/ACM Transactions on Networking*, 23(2):341–354, April 2015.

[10] Z. Durumeric, M. Bailey, and J. A. Halderman. An internet-wide view of internet-wide scanning. In *Proceedings of USENIX Security Symposium 2014*, pages 65–78, 2014.

[11] Z. Durumeric, E. Wustrow, and J. A. Halderman. Zmap: Fast internet-wide scanning and its security applications. In *Proceedings of USENIX Security Symposium 2013*, pages 605–620, 2013.

[12] R. Fontugne, P. Borgnat, P. Abry, and K. Fukuda. MAWILab: combining diverse anomaly detectors for automated anomaly labeling and performance benchmarking. In *Proceedings of CoNEXT 2010*, pages 1–12, 2010.

[13] K. Fukuda and R. Fontugne. Estimating speed of scanning activities with a hough transform. In *Proceedings of ICC 2010*, pages 1–5.

[14] C. Gates. Coordinated scan detection. In *Proceedings of NDSS 2009*.

[15] E. Glatz and X. Dimitropoulos. Classifying internet one-way traffic. In *Proceedings of IMC 2012*, pages 37–50, 2012.

[16] R. D. Graham. Masscan: Mass ip port scanner. https://github.com/robertdavidgraham/masscan. Accessed: 2015-05-05.

[17] ISC (Internet Storm Center). https://isc.sans.edu/feeds/daily_sources.

[18] ISC TCP/UDP Port Activ. https://isc.sans.edu/port.html?port=5900.

[19] J. Jung, V. Paxson, A. Berger, and H. Balakrishnan. Fast portscan detection using sequential hypothesis testing. In *Proceedings of SP 2004*, pages 211–225, 2004.

[20] D. Leonard and D. Loguinov. Demystifying internet-wide service discovery. *IEEE/ACM Transactions on Networking*, 21(6):1760–1773, Dec. 2013.

[21] D. Leonard, Z. Yao, X. Wang, and D. Loguinov. Stochastic analysis of horizontal ip scanning. In *Proceedings of INFOCOM 2012*, pages 2077–2085, 2012.

[22] J. Mazel, R. Fontugne, and K. Fukuda. Taxonomy of anomalies in backbone network traffic. In *Proceedings of TRAC 2014*, pages 30–36, 2014.

[23] Mixter. Nsat. http://nsat.sourceforge.net/. Accessed: 2015-02-28.

[24] L. Quan, J. Heidemann, and Y. Pradkin. Trinocular: Understanding internet reliability through adaptive probing. In *Proceedings of SIGCOMM 2013*, pages 255–266, 2013.

[25] J. Treurniet. A network activity classification schema and its application to scan detection. *IEEE/ACM Transactions on Networking*, 19(5):1396–1404, 2011.

[26] A. Wahid, C. Leckie, and C. Zhou. Characterising the evolution in scanning activity of suspicious hosts. In *Proceedings of NSS 2009*, pages 344–350.

[27] G. Xie, H. Hang, and M. Faloutsos. Scanner hunter: Understanding http scanning traffic. In *Proceedings of ASIACCS 2014*, pages 27–38.

[28] V. Yegneswaran, P. Barford, and J. Ullrich. Internet intrusions: Global characteristics and prevalence. In *Proceedings of SIGMETRICS 2003*, pages 138–147.