Coordinated Web Scan Detection Based on Hierarchical Correlation

Jing Yang[1,2], Liming Wang[1(✉)], Zhen Xu[1], Jigang Wang[3], and Tian Tian[3]

[1] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{yangjing,wangliming,xuzhen}@iie.ac.cn
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3] Zhongxing Telecommunication Equipment Corporation, Nanjing, China
{wang.jigang,tian.tian1}@zte.com.cn

Abstract. Web scan is one of the most common network attacks on the Internet, in which an adversary probes one or more websites to discover exploitable information in order to perform further cyber attacks. For a coordinated web scan, an adversary controls multiple sources to achieve a large-scale scanning as well as detection evasion. In this paper, a novel detection approach based on hierarchical correlation is proposed to identify coordinated web campaigns from the labelled malicious sources. The semantic correlation is used to identify the malicious sources scanning the similar contents, and the temporal-spatial correlation is employed to identify malicious campaigns from the semantic correlation results. In both correlation phases, we convert the clustering problem into the group partition problem and propose a greedy algorithm to solve it. The evaluation shows that our algorithm is effective in detecting coordinated web scan attacks, since the metric Precision for detection can achieve 1.0, and the metric Rand Index for clustering is 0.984.

Keywords: Web security · Coordinated scan · Hierarchical correlation · Cyber security

1 Introduction

Web scan is one of the most common web attacks on the Internet. During a web scan attack, an adversary probes one or more websites to discover exploitable information in order to perform further cyber attacks. According to the adversaries' intension, web scan attacks can be classified into three categories, i.e., web vulnerability scan, sensitive information scan and webshell scan. As a reconnaissance method, scanning is very important for subsequent attacks. For example, it is reported that more than 3,500 websites were added unauthorized code by attackers using automated scripts to scan these websites and find exploitable bugs in January, 2016 [1].

J. Li et al. (Eds.): SPNCE 2019, LNICST 284, pp. 388–400, 2019.
https://doi.org/10.1007/978-3-030-21373-2_30

To achieve large-scale or comprehensive web reconnaissance, an adversary utilize multiple sources to scan the responding websites in a coordinated web scan. Furthermore, employing multiple sources can make the scan activities remain stealthy, and accordingly avoid detection. Coordinated web scan campaigns of different scan types vary greatly in the access patterns. Coordinated scan sources for web vulnerabilities are probably similar with legitimate web crawlers due to the significant temporal synchronicity in their time series. The reason is that adversaries usually employ multiple sources to scan simultaneously in order to gather as much information as possible in a short time duration. While with regarding to sources in a coordinated scan for sensitive information or webshell, the access patterns are totally different. Since a few requests sent in these scan activities, adversaries may control multiple sources to scan different targets alternately.

Detection of single-source web scan activity is similar to web application attacks detection, and there are plenty of methods proposed [2–8] in the literature. However, those methods cannot identify the correlation of multiple web scan sources. The coordinated port scan detection has been addressed in many works [9–14], which put the emphasis on measuring the relativity between different sources. However, profiling web scan activities is quite a contrast to profiling port scan activities. The searching space for a port scan is limited and predictable due to the limited range of networking ports, but that for web scan is unlimited. Consequently the methods for coordinated port scan detection can not be applied to coordinated web scan detection. Jacob et al. [15] proposed `PUBCRAWL` to achieve malicious web crawlers detection and crawling campaign attribution. Their method can be used for coordinated web vulnerability scan detection, but it is not suitable for the other two scan types.

In this work, a novel detection approach is proposed to identify coordinated web scanners from the labelled malicious sources. We employ a hierarchical correlation model to comprehensively analyze the similarity of different sources. For the labelled malicious scan sources and the corresponding web traffic logs, the semantic correlation is deployed to aggregate malicious sources into groups. Sources in each group are semantically similar with each other. Then we employ the temporal-spatial correlation to each group in order to find the coordinated scanners. In both correlation phases, we convert the clustering problem into the group partition problem and propose a greedy algorithm to solve it. Our evaluation is carried out on a large dataset collected from a web hosting service provider with about 25 million web traffic log entries. We respectively quantify the capabilities of detection and clustering of our method. The evaluation results show that our algorithm is effective in detecting coordinated web scan activities, since the metric Precision for detection can achieve 1.0 at the best, and the metric Rand Index for clustering is 0.984.

We organize this paper as following: Sect. 2 presents our insight on coordinated web scan attacks. Related work is introduced in Sect. 3. Section 4 gives an overview of our approach and more details on the hierarchical correlation model.

The evaluation of our approach is in Sect. 5. Finally we make a discussion and conclude the paper in Sect. 6.

2 Coordinated Web Scan

As an important reconnaissance approach, a web scan is used by an adversary to gather information about the responding websites. We summarize web scan attacks into three categories according to the adversaries' intension.

- **Web Vulnerability Scan**. A typical website usually has three layers, i.e., the web server, some third-party web application frameworks and the business application. Web vulnerabilities consist of web application vulnerabilities (e.g., SQL or code injections, Cross-Site Scripting) and web server vulnerabilities (e.g., IIS, Apache, Tomcat). The web application vulnerabilities in known third-party web application frameworks are most concerned by adversaries, for instance, the Apache Struts framework vulnerabilities including CVE-2017-5638 and CVE-2018-11776. To scan web vulnerabilities of a website, adversaries need to crawl the structure of the website and determine which query URLs may be vulnerable. Hence they always employ some automated tools to make a comprehensive scan, in which the amount of requests is large and the time duration of scanning is long compared to the benign users' traffic. IBM AppScan, HP WebInspect, Acunetix Scanner and Nikto are the most popular web application vulnerabilities scan tools. Adversaries usually perform web vulnerability scan attacks aiming at a handful of websites.
- **Sensitive Information Scan**. The misconfiguration of a website may lead to open access of sensitive information about the web application and sensitive files on the web server. Typical sensitive information includes backup files, configuration files, password files and administrative interfaces. It is not necessary to crawl the structure of the website for scanning sensitive information, and an adversary can employ a black list and an open-sourced web crawler to perform a scan attack. For this type of scan, the amount of requests is small and the time duration of scanning is short. Adversaries usually perform large-scale sensitive information scan attacks on the Internet.
- **Webshell Scan**. A webshell is a malicious backdoor uploaded by an adversary to control a compromised website. Compared to compromising a website, it is easier to determine whether or not a website has contained a known webshell. An adversary can perform a webshell scan attack with a black list on URLs of known webshells, such as "/plus/mytag_js.php?aid=9090","/plus/90sec.php". As same as the sensitive information scan, the amount of requests is small and the time duration of scanning is short in a webshell scan attack, and the scan scale is large.

Table 1 presents the comparison of the three web scan types from four aspects, including the scan scale, request amounts, time duration and whether specialized scanner tools are needed.

Table 1. Comparison of web scans.

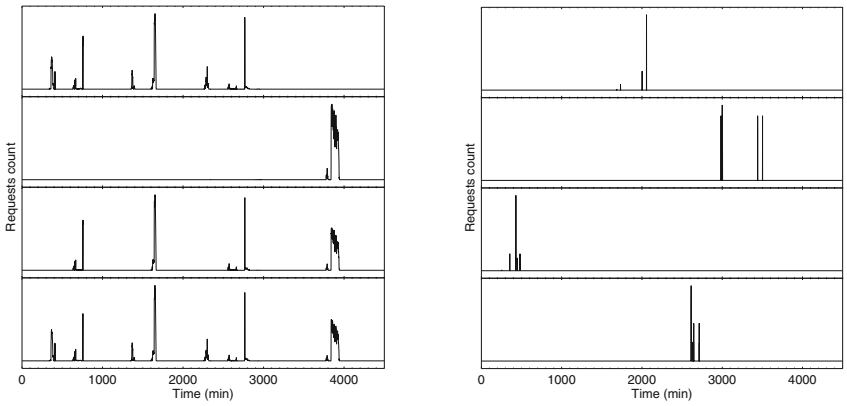
Type	Scale	Requests amounts	Time duration	Specialized tools
Web vulnerability scan	Small	Large	Long	Yes
Sensitive information scan	Large	Small	Short	No
Webshell scan	Large	Small	Short	No

During a coordinated web scan, an adversary employs multiple sources to scan the responding websites in order to improve efficiency and evade detection. Different sources in a coordinated web scan sweep websites for similar exploitable information, so there may be likenesses of semantic content between their web requests. In addition, adversaries may manipulate multiple sources to scan the target websites synchronously or alternately. If they perform scan attacks at the same time, there may be significant temporal synchronicity between their access patterns, which is the same as the distributed web crawlers. If they scan the target websites in turn, it is hard to discriminate between multiple sources within a coordinated scan and multiple adversities with the same scanner tool. However, from our observation, most of coordinated sources alternatively carrying out scan attacks are usually in a subnet of IP addresses. In other words, there is spatial similarity for the alternative coordinated web scan sources. Most of coordinated web vulnerability scan campaigns conform to the previous pattern, while most of coordinated sensitive information and webshell scan activities comply with the latter pattern.

Two examples of coordinated web scanners with different access patterns are shown in Fig. 1. Figure 1(a) illustrates the time series of four scan sources carrying out a web vulnerability scan attack. Since the time duration for a web vulnerability scan is relatively long, the temporal synchronicity is significant from their time series. The four sources in Fig. 1(b) belong to a webshell scan campaign, and they swept targets alternatively. Owing to the small volume of requests, their time series seem irrelevant. However their IP addresses are in the same /24 subnet and they used the similar source ports for scanning, indicating that they belong to the same malicious campaign.

3 Related Work

The concept of coordinated attack was first introduced by Green [9]. The author analyzed various coordinated attacks and probes, including traceroutes, NetBIOS scans, Reset scans, SFRP scans and DNS server exploit attempts. Braynov et al. [10] defined two types of cooperation for the coordinated attack: action correlation and task correlation. Gates et al. [11] developed a detection algorithm to recognize coordinated port scan activities based on the set covering technique. Zhou et al. [12] summarized different coordinated attacks including large-scale stealthy scans, worm outbreaks and distributed denial-of-service attacks, and gave a review of collaborative intrusion detection systems for detecting such



(a) Synchronous pattern (b) Alternating pattern

Fig. 1. Examples of coordinated web scanners with different access patterns

attacks. Elias et al. [13] presented an approach to fingerprint probing activity and inferred the machinery of the scan based on time series analysis techniques. Mazel et al. [14] provided a method to find the relationship of different scan sources based on the overlap and structure of destination IPs.

All of the above detection methods focus on the port scan attack. For the port scan, the scanning space is limited, which is only the range from 0 to 65535. It is practical to identify the correlation between different scanners based on the individual coverage of the whole scanning space. However, for the web scan, the scanning space is unlimited because the length of a HTTP request can be long enough. Accordingly, it is difficult to profile a web scanner's communication and predict its activities. Consequently the methods for coordinated port scan detection can not apply to coordinated web scan detection.

For coordinated web scan attacks, Xie et al. [5] introduced a clustering based approach named `Scan Hunter` to detect HTTP scanners, which can be used to identify scanners with the same scanner tool, but cannot distinguish the coordinated scan sources. Jacob et al. [15] proposed a method named `PUBCRAWL` for detecting malicious crawler campaign by identifying synchronized traffic. They also utilized time series clustering to aggregate similar time series from detected malicious crawlers. Squared Euclidean Distance is the metric for measuring the similarity between time series. Obviously, it can only detect the synchronous web scanners mentioned previously, and for the alternating web scanners it doesn't work.

4 Methodology

In our detection method, we employ a hierarchical correlation model to comprehensively analyze the similarity of different sources from the semantic characteristic and the temporal-spatial characteristic. For the labelled malicious scan

sources and the corresponding web traffic logs, a semantic correlation is firstly employed to aggregate malicious sources into groups. Sources in each group are scanners with similar tools to scan similar contents. Then for each semantic similar group, a temporal-spatial correlation is utilized to find the coordinated scanners. The concept of our methodology is shown in Fig. 2.

For the semantic correlation, we use a word set to profile a scan source's visiting behavior and construct a similarity matrix with the Jaccard distance as the similarity metric. Because there is no knowledge of the number of clusters in advance, we transform the similarity matrix into the adjacency matrix with a similarity threshold and convert the problem of clustering into partitioning.

For the temporal-spatial correlation, the time series of each source in a semantic similar group are extracted at first. If two time series have overlaps, we compute the Pearson correlation coefficient to measure the similarity between them. While if two series don't overlap and the two sources are in the same subnet of IP address, we denote they are correlated and the similarity is set to 1. As same as the semantic correlation, we employ the group partition technique to identify coordinated scanners from a semantic similar group.

A simple and efficient group partition algorithm is proposed at the end of this section, which is a greedy method to divide nodes in an adjacency matrix into groups.

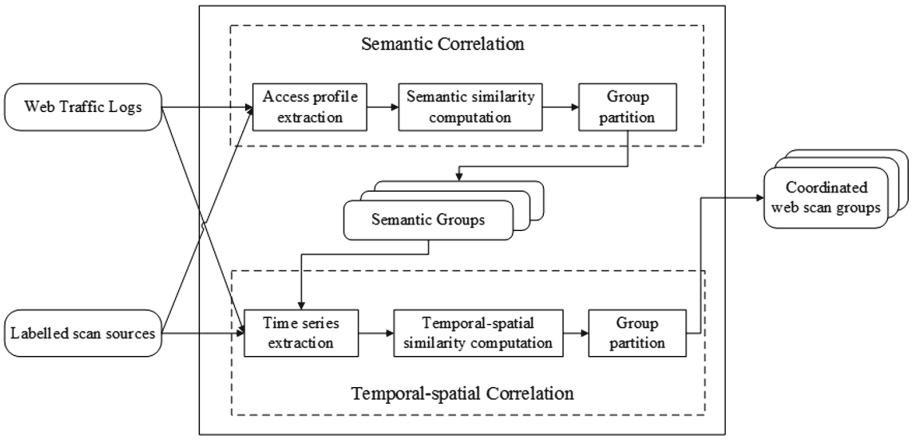


Fig. 2. Overview of our approach

4.1 Semantic Correlation

We utilize the method proposed in our previous work [8] to profile the web access behavior of a scan source. For a scan source A, we extract all requests and put all resource identifiers and query strings together into two independent text files. Different separators are applied to split the files into two word sets W_u and

W_q. Then we combine the two word sets together and get the whole word set $W = W_u \cup W_q$.

To find the groups of different scan sources with similar scan contents, we construct the semantic similarity matrix M_S. The semantic similarity $SS(i,j)$ of two sources A_i and A_j is measured by the Jaccard distance. which is denoted by the following equation:

$$SS(i,j) = 1 - J(W_i, W_j) = 1 - \frac{|W_i \cap W_j|}{|W_i \cup W_j|}. \tag{1}$$

Traditional clustering techniques require specifying the number of clusters in advance, which is not piratical in our detection. Hence we convert the clustering problem into the group partition problem by transforming the similarity matrix M_S into the adjacency matrix M_{AS}. With an assigned semantic similarity threshold λ_S, if the similarity of two sources is larger than the threshold, we define the pair of sources are adjacent, otherwise they are not. For the adjacency matrix M_{AS}, we employ our proposed partition algorithm which is detailed as following to cluster the input scan sources.

4.2 Temporal-Spatial Correlation

For each source in the obtained semantic similar groups, we build the corresponding time series in the whole detection time window duration. With an assigned time interval Δt, the whole time window can be partitioned into $N_{\Delta t}$ time intervals. For the lth time interval, we count the number of requests N_l sent from a given source A, and the time series X of A is:

$$X = \{N_l\} \tag{2}$$

where $l \in [0, N_{\Delta t}]$. For two sources A_i and A_j, if X_i and X_j have overlaps, we compute the Pearson correlation coefficient ρ_{ij} as the temporal-spatial similarity coefficient $TS(i,j)$ of the two sources:

$$\rho_{ij} = \frac{cov(X_i, X_j)}{\sigma_{X_i}\sigma_{X_j}} \tag{3}$$

where $cov(X_i, X_j)$ is the covariance of X_i and X_j, and σ_{X_i} is the standard deviation of X_i.

If X_i and X_j have no overlap, we identify the relevance of the two sources from the spatial characteristic. If the two sources are in the same subnet of IP addresses, we denote they are correlated and the similarity $TSS(i,j)$ is set to 1, otherwise it is set to 0. The subnet space is defined by the assigned threshold λ_{IP}.

As same as the semantic correlation, we transform the similarity matrix M_{TS} into the adjacency matrix M_{ATS} by assigning the temporal-spatial similarity threshold λ_{TS}. For the adjacency matrix M_{ATS}, we employ our proposed partition algorithm which is detailed as following to get the final results.

4.3 Proposed Group Partition Algorithm

For an input adjacency matrix M_A, a greedy algorithm is presented to partition the nodes in the adjacency matrix. Our core idea is that if two nodes are connected with each other, they should be put together into one group. Specifically, assuming $node_i$ and $node_j$ are connected, if $node_i$ already belongs to the group G_k but $node_j$ is alone, then $node_j$ is added into G_k; if $node_j$ already belongs to G_k but $node_i$ is alone, then $node_i$ is added into G_k; if the two nodes have belonged to different groups, skip to the next pair of nodes. If $node_i$ is not connected to any other nodes, it is added into a new group.

Actually the known community detection algorithm `Louvain` [16] can apply to our method. The *Louvain* algorithm includes two iteratively repeated phases. The first phase is for modularity optimization, and the second phase is for community aggregation. Compared with it, our algorithm is more efficient and more suitable for our detection method, since it only has one phase and does not need to repeat.

5 Evaluation and Results

We utilized the open-source network monitor Bro [17] to collect a large volume of web traffic logs from a web hosting service provider in order to evaluate our approach. The method proposed in our previous work [8] is employed to detect malicious IPs from the dataset, and we labeled the coordinated web scan campaigns manually. Our approach is implemented in Java with the full-text search engine `Elasticsearch` for data storage.

Our evaluation is divided into two parts. At first, we measured the detection capability of our method in order to check whether our approach can differentiate between a single scanner and a coordinated scanner. Furthermore, we measured the clustering capability of our method to examine whether our approach can identify the different sources of one campaign into the same group.

5.1 Dataset

Our dataset consists of about 20 million log entries generated by 156,396 IP addresses, which involves 534 fully qualified domain names (FQDNs) from May 17 to 26, 2016. We equally divided it into the training dataset D_{train} and the testing dataset D_{test}. The training dataset, with logs in the first 5 days, is used for parameters selection, and the test dataset, with the left logs, is used for evaluating the performance of the method.

In the training dataset there are totally 1,207 detected malicious IPs. With the auxiliary of visualization tool `Kibana`, we manually analyzed the scan contents, time series synchronization and IP addresses distribution of these malicious IPs, and labelled 134 coordinated scan IPs involving 30 different groups. Among of them, 2 groups performed the web vulnerability scan attacks, 9 groups carried out the sensitive information scan attacks, and 19 groups scanned for

webshells. The testing dataset includes 1,058 malicious IPs with 149 labelled coordinated scan IPs and 38 labelled groups. In the testing dataset, only one group performed the web vulnerability scan attacks, 16 groups carried out sensitive information scan attacks, and 21 groups scanned for webshells. The distribution of numbers of group members is shown in Table 2. In most groups, the multiple IP addresses are in a /24 subnet. Only two groups contain several totally different IP addresses.

Table 2. Distribution of numbers of group members.

Num. of groups members	2	3	4	5	6	8	9	11	15	16	17	18
Num. of groups in D_{train}	15	1	5	5	2	0	0	1	0	1	1	0
Num. of groups in D_{test}	16	10	4	3	1	1	1	0	1	0	0	1

5.2 Evaluation of the Detection Capability

With regarding to the detection capability, we use four metrics: Precision (P), Recall (R), Accuracy (Acc) and F measure (F_β) to quantify the performance of our approach. The most common measure F_1 with $\beta = 1$ is chosen. Assuming a malicious entity is labelled as a coordinated scan entity, if our approach also classifies it as a coordinated entity, it is defined as a true positive, and if it is classified by our approach as a single scan entity, it is defined as a false negative. The definitions of a true negative and a false positive are similar. Given the numbers of true positives, false positives, true negatives, and false negatives as TP, FP, TN and FN, the four metrics are calculated as:

$$P = \frac{TP}{TP + FP} \tag{4}$$

$$R = \frac{TP}{TP + FN} \tag{5}$$

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

$$F_1 = \frac{PR}{P + R} \tag{7}$$

Precision can measure the portion of actual coordinated scan entities in all predicted coordinated scan entities. Recall measures the number of correctly classified coordinated scan entities out of the total number of labelled coordinated scan entities. Accuracy measures the number of correctly distinguished coordinated scanners and single scanners. F_1 score is the harmonic mean of Precision and Recall.

Parameter Selection. In our approach, there are four assigned parameters: Δt, λ_{SS}, λ_{IP} and λ_{TS}. We empirically set Δt as 1 min. For the last three threshold parameters, we designed three groups of experiments with the training dataset. In each group of experiments, two thresholds are fixed and one threshold is changed to find the best threshold value. Figure 3 illustrates the results of the three groups of experiments. In the first group, we fixed $\lambda_{IP} = 256$ and $\lambda_{TS} = 0.5$, and when $\lambda_{SS} = 0.5$ we achieved the best Precision and F_1 score. In the second group, we fixed $\lambda_{SS} = 0.5$ and $\lambda_{TS} = 0.5$, and Precision is the best when $\lambda_{IP} = 1024$ but the F_1 is the best when $\lambda_{IP} = 2048$. With an overall consideration, we chose $\lambda_{IP} = 2048$. In the last group, we fixed $\lambda_{SS} = 0.5$ and $\lambda_{IP} = 2048$, and when $\lambda_{TS} = 0.5$ Precision and F_1 are the best.

Finally the three threshold parameters are set as: $\lambda_{SS} = 0.5$, $\lambda_{IP} = 2048$ and $\lambda_{TS} = 0.5$. With these settings, our method on the training dataset can achieve: $P = 0.99$, $R = 0.793$, $Acc = 0.97$ and $F_1 = 0.846$.

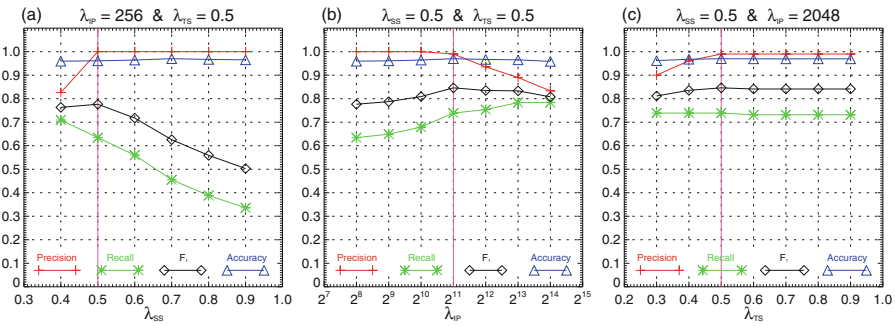


Fig. 3. Results of experiments for parameter selection.

Comparison with Different Partition Algorithms. In this section, we measure the performance of our proposed group partition algorithm. By replacing the algorithm used in the group partition part, we compare the detection performance by employing our greedy partition algorithm with that by employing the `Louvain` algorithm on the testing dataset D_{test}.

The comparison of metrics with different algorithms is shown in Table 3. With our algorithm, the precision is 1.0 with the number of true positives is 108 and no false positive, and the recall is 0.724 with 41 false negatives. The metric F_1 is 0.84 and Acc is 0.961. Among the 41 false negatives, 11 labelled groups involving 26 malicious IPs are not detected at all, and 8 labelled groups involving 15 malicious IPs are partially not detected. By examining the traffic logs of the 26 malicious IPs in 11 undetected groups, we found that most of their traffic logs were mixed by massive random resource identifiers or normal web accessing requests, which leads to a significant decrease in the semantic similarity.

The results obtained by utilizing the `Louvain` algorithm are almost the same as that obtained by our algorithm, while our algorithm is slightly better than `Louvain`. It is revealed that the proposed partition algorithm is more suitable for our method.

Table 3. Comparison of detection metrics with different algorithms.

	TP	FP	TN	FN	P	R	Acc	F_1
Our algorithm	108	0	909	41	1.0	0.724	0.961	0.84
`Louvain`	107	0	909	42	1.0	0.718	0.96	0.836

Comparison with Different Correlation Strategies. In our method, we introduce the hierarchical correlation model with a combination of semantic correlation and temporal-spatial correlation. To quantify the performance of the model, we compared the results obtained by employing hierarchical correlation with the results obtained by only employing semantic correlation, temporal-spatial correlation, spatial correlation and temporal correlation.

The results are shown in Table 4. Precision and F_1 obtained by employing hierarchical correlation are significantly better than that by only employing one correlation. It is concluded that our hierarchical correlation model is effective for coordinated web scan detection.

Table 4. Comparison of results with different correlation strategies.

	TP	FP	TN	FN	P	R	Acc	F_1
Hierarchical correlation	108	0	909	41	1.0	0.724	0.961	0.84
Only semantic correlation	123	567	342	26	0.178	0.826	0.44	0.293
Only temporal-spatial correlation	144	277	632	5	0.342	0.966	0.733	0.505
Only spatial correlation	145	532	377	4	0.214	0.973	0.493	0.351
Only temporal correlation	53	181	728	96	0.226	0.356	0.738	0.277

5.3 Evaluation of the Clustering Capability

We chose the widespread measure Rand index (RI) for evaluating the performance of clustering. With regarding to group partition, a true positive decision assigns two similar sources to the same group, and a true negative decision assigns two dissimilar sources to different groups. There are two types of errors. A false positive decision assigns two dissimilar sources to the same group. A false negative decision assigns two similar sources to different groups. RI measures the percentage of decisions that are correct, which is calculated as following:

$$RI = \frac{TP + TN}{TP + TN + FP + FN}. \tag{8}$$

With the setting of parameters as the above, our method aggregates 108 malicious sources into 28 groups, and the *RI* is 0.984. Among the 28 groups, 20 classified groups are equal with their labelled groups, and 8 classified groups are partial equal with the labelled groups. There is no group containing sources from different labelled groups. The satisfactory results prove that our method for coordinated web scan detection can be applied into practice.

6 Discussion and Conclusions

In this paper, we introduce a hierarchical correlation based methodology to distinguish coordinated web scanners. With the combination of semantic correlation analysis and temporal-spatial correlation analysis, our method can effectively detect coordinated web scan campaigns in different access patterns. Compared with `PUBCRAWL` proposed by Jacob et al. [15], which can only detect malicious web crawler campaigns with significant temporal synchronicity, our work combines the temporal synchronicity, the semantic similarity and the spatial distribution similarity, and can combat with both the synchronous and alternating coordinated web scan attacks.

In conclusion, we give an overall insight on the coordinated web scan attack, and propose a novel detection approach based on hierarchical correlation. In our detection approach, the semantic correlation is used to identify the malicious entities scanning the similar contents, and the temporal-spatial correlation is employed to identify scan campaigns from the semantic correlation results. Furthermore, we propose an efficient greedy algorithm for group partition which is used in both correlation phases to aggregate sources. We evaluate our approach on a manually labeled dataset, and the results reveal that it can effectively distinguish the coordinated web scan campaigns.

Acknowledgments. This paper is supported by the National Key R&D Program of China (2017YFB0801900).

References

1. Security Newspaper. https://www.securitynewspaper.com/2016/01/23/web-reconnaissance-attack-infects-3500-websites-possibly-wordpress/. Accessed 20 Nov 2018
2. Kruegel, C., Vigna, G.: Anomaly detection of web-based attacks. In: Proceedings of the 10th ACM Conference on Computer and Communications Security, pp. 251–261. ACM (2003)
3. Valeur, F., Mutz, D., Vigna, G.: A learning-based approach to the detection of SQL attacks. In: Julisch, K., Kruegel, C. (eds.) DIMVA 2005. LNCS, vol. 3548, pp. 123–140. Springer, Heidelberg (2005). https://doi.org/10.1007/11506881_8
4. Robertson, W., Vigna, G., Kruegel, C., Kemmerer, R.A.: Using generalization and characterization techniques in the anomaly-based detection of web attacks. In: Annual Network & Distributed System Security Symposium (NDSS) (2006)

5. Xie, G., Hang, H., Faloutsos, M.: Scanner hunter: understanding HTTP scanning traffic. In: Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security, pp. 27–38. ACM (2014)
6. Shancang, L.I., Romdhani, I., Buchanan, W.: Password pattern and vulnerability analysis for web and mobile applications. ZTE Commun. **14**(S1), 32–36 (2016)
7. Mimura, M., Tanaka, H.: Heavy log reader: learning the context of cyber attacks automatically with paragraph vector. In: Shyamasundar, R.K., Singh, V., Vaidya, J. (eds.) ICISS 2017. LNCS, vol. 10717, pp. 146–163. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-72598-7_9
8. Yang, J., Wang, L., Xu, Z.: A novel semantic-aware approach for detecting malicious web traffic. In: Qing, S., Mitchell, C., Chen, L., Liu, D. (eds.) ICICS 2017. LNCS, vol. 10631, pp. 633–645. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-89500-0_54
9. Green, J., Marchette, D.J., Northcutt, S., Ralph B.: Analysis techniques for detecting coordinated attacks and probes. In: Proceedings of workshop on Intrusion Detection and Network Monitoring, pp. 1–9 (1999)
10. Braynov, S., Jadliwala, M.: Detecting malicious groups of agents. In: Proceedings of the First IEEE Symposium on Multi-Agent Security and Survivability, pp. 90–99. IEEE (2004)
11. Gates, C.: Coordinated scan detection. In: Annual Network & Distributed System Security Symposium (NDSS) (2009)
12. Zhou, C.V., Leckie, C., Karunasekera, S.: A survey of coordinated attacks and collaborative intrusion detection. Comput. Secur. **29**, 124–1402 (2010)
13. Elias, B.H., Mourad, D., Chadi, A.: On fingerprinting probing activities. Comput. Secur. **43**, 35–48 (2014)
14. Mazel, J., Fontugne, R., Fukuda, K.: Identifying coordination of network scans using probed address structure. In: Traffic Monitoring and Analysis-8th International Workshop, pp. 7–8 (2016)
15. Jacob, G., Kirda, E., Kruegel, C., Vigna, G.: PUBCRAWL: protecting users and businesses from CRAWLers. In: Proceedings of 21st Usenix Conference on Security Symposium, pp. 507–512. Usenix (2013)
16. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Etienne, L.: Fast unfolding of communities in large networks. J. Stat. Mech. Theory Exp. **2008**(10), P10008 (2008)
17. Paxson, V.: Bro: a system for detecting network intruders in real-time. In: Proceedings of 7th USENIX Security Symposium. Usenix (1998)