3-3 Data Mining Applied to Darknet Traffic Analysis

Tao BAN

The monitoring of unused IP address space — so called darknet — provides a cost-effective way to monitor the global trends of cyber-attacks in the Internet. By monitoring a large, distributed, global-scale darknet, the NICTER project has been collecting, reporting, and mitigating tremendous malicious activities in the cyberspace for more than a decade. In this paper, we present the recent advances at the NICTER with a focus on the newly developed data mining engines that lie at its core. The experiment results show that darknet monitoring provides a good tradeoff between the monitoring cost and global knowledge acquisition for tracking the trends of cyber-threats in the Internet. Strategic countermeasure to cyber-threats can be enabled based on the discoveries reported in this paper.

1 Introduction

The proliferation of malicious software — so called malware — poses a major threat to the confidentiality, integrity, and availability of the data stored and communicated using the Internet. To address the concerns raised by malware, there is a pressing need for the development of network monitoring systems, which could provide a global-perspective overview and detailed forensic information on new cyber-threats in a timely manner. While the computation, storage, and communication costs for monitoring a densely populated network of global scale render the task impossible, the monitoring of unused address space, a.k.a. a darknet [1]–[3], usually provides a good cost-performance compromise.

A darknet, also known as network telescope, blackhole monitors, sinkholes, or background radiation monitors, is a portion of routed, allocated IP space that contains no advertised services [1]–[3]. Because of the absence of legitimate hosts on the darknet, any traffic observed on a darknet is by its presence aberrant: it is either caused by a malicious intention or a mis-configuration. Assorted works have deployed darknets in existing networks to help identify the types and sources of malicious traffic present on the larger network of which they form a part, where darknets are used to host flow collectors, backscatter detectors, packet sniffers, and so on [4][5]. Considerable improvement in detection rate and cut-down in false positive rate are reported in related work, leading to improved awareness of malicious or mistaken activities and simplified mitigation.

To facilitate early warning and mitigation of various cyber-security threats raised by malware, we have been developing and operating the NICTER (Network Incident analysis Center for Tactical Emergency Response) [2][6][7] for more than a decade. By means of monitoring over a global-scale darknet and static-dynamic analysis of hand-collected malware variants, the NICTER binds the results of both macroscopic and microscopic analysis to obtain richer information about malicious activities in the Internet and applies acquired knowledge to protect the user networks. This paper describes the recent advances of the NICTER, with a focus on the recent developed data mining techniques aiming at detection, prevention, and mitigation of emerging cyber-threats.

The rest of the paper is organized as follows. In Section **2**, we give a brief introduction of the NICTER and related research. In Section **3**, we present a study on host behavior analysis that could help to predict the future status of attacking hosts observed in the darknet. In Section **4**, we introduce an approach that can identify DDoS (Distributed Denial of Service)-attacked servers from disturbing hosts issuing SYN_ACK packets. In Section **5**, we describe a new scheme towards early detection of emerging threats. The conclusion is drawn in the final section.

2 The NICTER and related work

In this section, we give a brief introduction to the

NICTER and the related work, with a particular emphasis on the darknet monitoring aspect.

2.1 The overview of NICTER

The NICTER combines two well-known approaches to fight against malware: The macroscopic approach focuses on grasping the trend of malicious activities based on global-scale network monitoring. The microscopic approach focuses on analyzing malware specimens captured by honeypots, etc., to attain a deep understanding of their characteristics and behaviors and therefore enable quarantine and mitigation.

The NICTER's macroscopic component, a.k.a. MacS, monitors network traffic collected at distributed darknet sensors installed world-widely. According to the inherent nature of the darknet packets, the IP addresses issuing packets are treated as attacking hosts, and packets from a unique host during a short period are taken as a candidate incident. The microscopic component of the NICTER, a.k.a. MicS, makes use of honeypots and email traps to capture malware in the wild. Acquired malware specimens are fed to a malware behavior analyzer and a malware code analyzer so that a profile is learned based on their behavioral characteristics and key features.

The NICTER consists two other subsystems to fuse the results of MacS and MicS for incident handling. The so-called NemeSys (NEtwork and Malware Enchaining System) enchains the phenomena, i.e., incident candidates observed in the darknet, and their root causes, i.e., malware variants. Once the MacS observe a candidate incident, the correlation analyzer in the NemeSys outputs a list of malware variants whose profile matches the incident. Finding the root causes of the observed network attacks provides a much clearer view of happenings in the Internet and therefore lead to a better chance to mitigate the threats. Finally, the IHS (Incident Handling System) helps the operator to diagnose the results from the above analyses and file an incident report.

In the rest of the paper, we focus on the macroscopic aspect of the NICTER. Refer to [2][6][7] for detailed information about the other aspects of NICTER.

2.2 Analysis engines at the NICTER

The recorded number of packets arriving at the darknet has been gradually increasing along with the scale of the darknet space monitored by the NICTER. Table 1 shows the basis statistic of the darknet monitored by the NICTER. In 2015, the total number of monitored darknet IP addresses sums up to 280 thousand, the number of packets collected goes to 54.51 billion, resulting in an average number of more than 213,500 thousand packets per IP throughout the year. The last column of the table shows a clear increasing trend of the average number of packets arrived at each IP during the 10-year observation period. This trend indicates the rise of scan/attacking activities in the darknet so that calls forth the need for advanced mining methods that exploit the regularity in the data for cyber-threat mitigation.

Table 1 Yearly statistics of darknet monitoring recorded by the NICTER

Year	#Packets (billion)	#IP Address (thousand)	#Packet/IP
2006	0.81	100	17,231
2007	1.99	100	19,118
2008	2.29	120	22,710
2009	3.57	120	36,190
2010	5.65	120	50,128
2011	4.54	120	40,654
2012	7.79	190	53,085
2013	12.90	210	63,655
2014	25.70	240	115,323
2015	54.51	280	213,523

We have been developing various visualization and data mining engines associated with the NICTER to facilitate incident reporting and attack mitigation. In [6], Inoue et al. introduced Atlas — a geographical traffic visualization engine that illustrates the transverse of the packet from source to destination in the map, Cube — a comprehensive 3-D traffic visualization engine rendered inside a cube, and Tap View — a host-behavior visualization engine characterizing the activities of attacking hosts during the incidents.

In [2], Inoue et al. present the primary analysis engines including Change Point Detector (CPD), Self-Organizing Map (SOM) Analyzer, and Incident Forecast (IF) Engine. Towards detecting a rapid change in monitored traffic in a timely fashion, CPD implements a time series analysis engine that uses two-stage on-line discounting learning based on the Auto-Regression (AR) model. The SOM analyzer is a clustering and visualization engine designed for classifying as well as detecting unknown malwares and their variants by means of characterization of their network behaviors. The IF is a forecasting engine for predicting the amount of traffic for future incidents several hours ahead so that prompt reactions can be enabled for the coming

incidents.

Refer to [2] for more information on analysis engines involved in the NICTER.

2.3 The spinoffs of NICTER

Visualization and analysis technologies bred by the NICTER have been applied to enhance the security operations in user networks to complement conventional security appliances such as Intrusion Detection/Prevention System (IDS/IPS) in security operation.

The DAEDALUS system[8] is developed in aim of bridging the gap between darknet monitoring and actual security operations on live networks (referred to as livenet hereafter): monitoring the global trend does not make a very direct contribution toward livenet protection. In contrast to conventional methods, wherein only the packets received from outside of the organization are observed, a distributed darknet covering IP space in multiple organizations can observe the malicious packets transmitted cross the edge of the organizations. In DAEDALUS, an inter-organization alert will be issued if a scan is detected from a host towards the darknet within the same organization and intra-organization alert will be issued if a scan is detected from a host towards the darknet in different organization; a DDoS alert will be issued if backscatter packets (TCP packets with SYN_ACK flag on) are emitted from a registered IP address under protection. Together with its visualization engine introduced in [9], DAEDALUS enables operators to visually grasp a complete overview of alert circumstances in real time, whilst providing highly flexible and tangible interactivity with the darknet traffic as well as issued alerts.

As an extension of Atlas, NIRVANA — a livenet traffic visualization engine — renders real network traffic in real-time to enable detection of network failures and misconfigured devices and hence helps to reduce the workload of network administrators. Refer to [10] for detailed information of NIRVANA.

2.4 Related work on darknet monitoring

Regarding darknet monitoring, there are a number of ongoing projects known in the literature and several monitoring systems are already in their operational phase [2][4][5][11]–[15]. Via network event monitoring, many of these project is able to perform event analysis yielding statistical data such as rapid increase of accesses on certain port numbers and so on.

3 Behavior analysis of long-term cyber-attacks

In this section, we present a brief introduction of the study conducted in [3] on behavior analysis of attacking hosts. The study is driven by the necessity to gain further understanding of the behavior of malware-infected hosts over time, to identify their temporal regularities, and to predict their future activities based on their previous behavior.

3.1 Clustering based on attacked destination ports

It is well known that the targeted destination ports are closely related to the type of an attack. Clustering is employed to analyze the destination port information so as to attacking hosts with similar activities are grouped together. According to the experiment in [3] a linkage algorithm which takes Jaccard distance defined on the set of destination ports targeted by the attacking hosts as the proximity measure reveals that the top most attacked ports are port 445, 1433, 22, 3389, 80, as in year of 2011. The following analysis is done upon on these top ports to take advantage of the coincidence in temporal behavior of similar attacks.

3.2 Regression analysis on weekly attack volume time series

The task as to predict a host's attack behavior in terms of number of packets sent to the darknet based on its historical observation is approached by time series predic-

Table 2 Crossover regression performance on weekly attack-volume time series

Model trained from destination port	MSE tested on destination port				
	445	1433	22	3389	80
445	**3.61e-4**	2.17e-3	4.17e-3	8.04e-3	4.36e-3
1433	4.69e-4	**3.18e-4**	4.35e-3	7.84e-3	4.80e-3
22	8.57e-4	2.44e-3	**2.00e-3**	8.16e-3	4.32e-3
3389	6.31e-4	2.05e-3	3.74e-3	**3.77e-3**	4.03e-3
80	6.04e-4	3.20e-3	4.08e-3	8.64e-3	**1.28e-3**

Table 3 Crossover classification prediction on weekly attack-volume time series

Model trained from destination port	G-mean tested on destination port					F1-measure tested on destination port				
	445	1433	22	3389	80	445	1433	22	3389	80
445	**0.91**	0.94	0.88	0.77	0.79	0.94	0.92	0.80	0.73	0.60
1433	0.87	**0.95**	0.86	0.75	0.82	0.92	0.92	0.80	0.71	0.69
22	0.89	0.92	**0.92**	0.73	0.79	0.92	0.90	**0.89**	0.69	0.66
3389	0.78	0.94	**0.92**	**0.88**	0.85	0.91	0.90	0.78	**0.82**	0.60
80	0.76	0.91	0.88	0.77	**0.88**	**0.95**	**0.94**	0.85	0.76	**0.82**

tion. After all observed hosts are presented as a time series measured from the first week to the last week of 2011, by counting the number of packets received from the host within each week, the Support Vector Regression (SVR) [16] is selected to perform the learning and prediction.

Table 2 shows the result of the regression, where Mean Squared Error (MSE) ise used to measure the prediction performance. As can be seen in the right half of the table, the MSE values on the diagonal appear to be the minimum of each row, which means that the regression model trained from a cluster best fits the test set from the same cluster. Small MSE values along the diagonal indicate that a host future behavior is closely related to its past behavior and such relation could be learned in a quantitative sense. The comparatively large MSE values off the diagonal suggest that different type of attacks may conforms to different behavior models in terms of number of packets send to the darknet, which is in consistence with our intuition.

3.3 Qualitative prediction on the attack

In this subsection, we move on to answer the following qualitative question of the hosta: given the historical statistics of a host in the past T time slots, it will continue its attack at T+1?

This question is best modeled as a classification problem. Based on the formulation in the previous subsection, we define a binary classification problem as follows: The input vectors for the classifier are kept the same as in the regression model while the output values are transformed into binary codes, where a host is labeled +1 if it no longer launches any attack at time T+1, or −1 otherwise. This time, we apply the Support Vector Machine (SVM)[16] to solve the problem. The evaluation results are shown in Table 3. Because classification problems formed from some of the clusters appeared to be skewed, i.e., samples from the one class overwhelm those from the other, we use G-mean and F1-measure instead of accuracy to measure the generalization performance of the classifiers. As can be seen from the table, G-mean shows a similar pattern as the MSE in Table 2, indicating that the hosts belonging to the same cluster behaves in a similar way. Despite of a little variation, the F1-measure in Table 3 also supports the above conclusion.

3.4 Summary

The numerical study based on function regression and classification verifies that there is strong predictability with regards to the attack behavior for hosts that are attacking the same destination port. The result of this study can be supportive in security operation such as adaptive blacklisting.

4 Early identification of DDoS-attacked hosts

In this section, we present an effective DDoS-event detection system [17] based on the analysis of backscatters collected from the darknet. The experiments show that our approach supports fast and accurate detection of DDoS attacks. Based on the discoveries, we can not only obtain the global trend of DDoS attacks but also discover new types of DDoS attacks as well.

4.1 System framework

The proposed system extracts feature vectors for each attacking hosts from packets received from the host during a fixed short observation period, and then performs learning and prediction using supervised learning.

Our system framework is shown in Fig. 1. In the feature-extraction block shown in the left part of the figure, we first group packets observed in a darknet by source IP address. Then, for a given host, we collect all packets in a fixed period of time from the time the first packet is observed, and transform them into a feature vector. In the detection block of the right part of the figure, we feed the input data to the classifier and distinguish DDoS events

from non-DDoS events. If the classifier predicts a DDoS-attack event with high confidence, it will issue an alert to the attacked host. If the classifier predicts with low confidence, the incident is will be forwarded to human operators for justification. The justified data with correct label information is fed to the classifier to performance incremental learning. We use support vector machine (SVM)[16] as the classifier because of its outstanding generalization performance.

For the detection, we adopt the 17 features as listed in Table 4 from darknet packets that are sent from a single source host during a 30-second period. To make the features descriptive, we generated feature vectors only for hosts that send at least 20 packets during a 30-second observation period. We conduct the detection for a host once every 60 minutes.

Table 4 Feature extraction for DDoS-attack event detection

Number of packets observed from the host
Time intervals of packets (Average and standard deviation)
Number of Source ports
Number of packets sent from source ports (Average and standard deviation)
Number of protocol types, including the type of TCP flags
Number of destination ports attacked
Number of packets sent to destination ports (Average and standard deviation)
Number of destination IPs
Number of packets sent to destination IPs (Average and standard deviation)
The difference of destination IPs (Average and standard deviation)
Payloads size (Average and standard deviation)

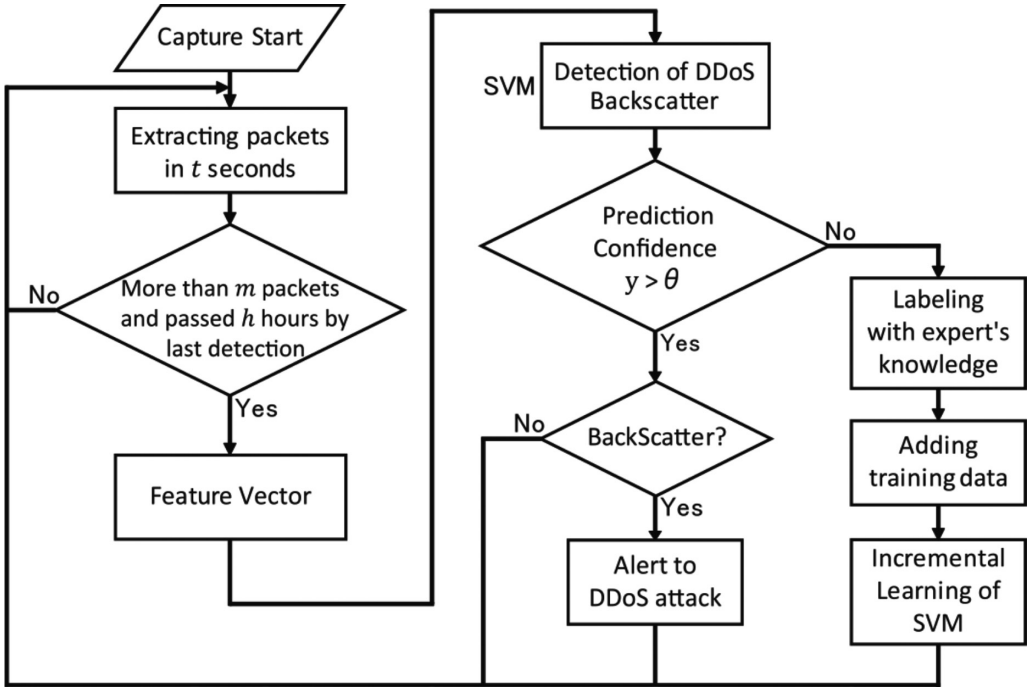


Fig. 1 Proposed framework to detect DDoS-attack events (Figure reused from [17])

Table 5 Performance evaluation of the DDoS-event detection

Week	Without incremental learning				With incremental learning			
	Precision(%)	Recall(%)	F1-Measure	Time(s)	Precision(%)	Recall(%)	F1-Measure	Time(s)
3	96.6	100	0.982	120	96.6	100	0.982	120
4	96.9	99.8	0.983	–	97.4	99.8	0.986	237
5	98.7	100	0.992	–	98.7	100	0.992	368
6	96.3	100	0.981	–	96.4	100	0.982	531
7	98.3	100	0.991	–	98.3	100	0.992	676
8	96.7	99.8	0.982	–	96.7	99.8	0.983	880

4.2 Experiment results

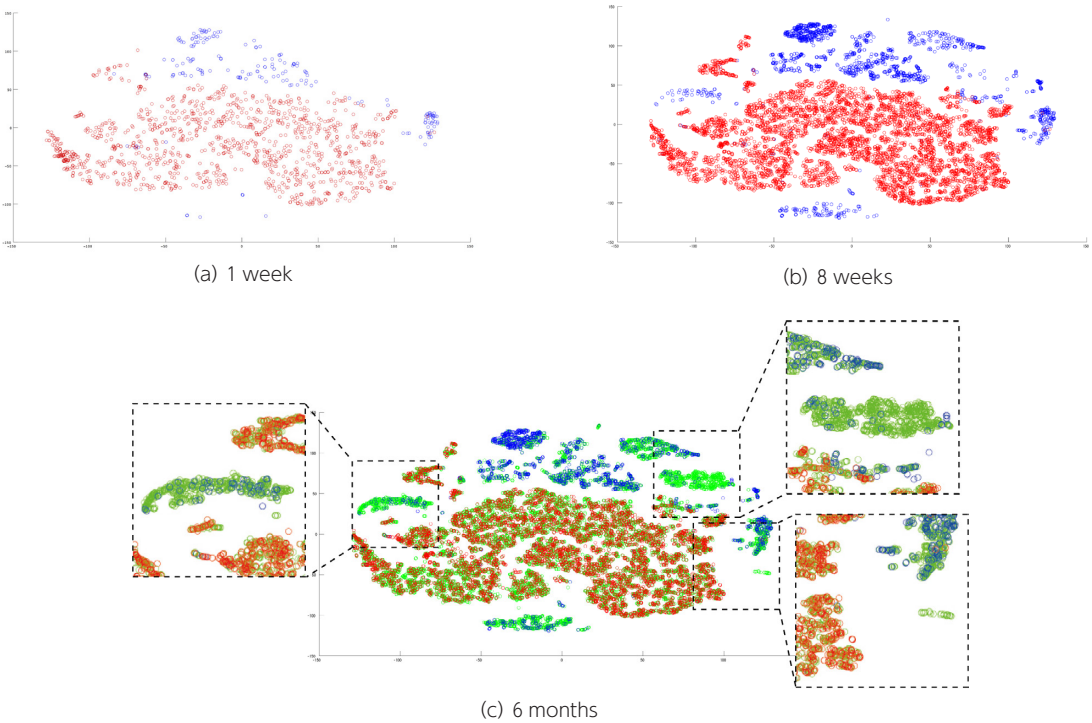
In the experiment, feature vectors created during the first 2 weeks are used for initial training to learn an SVM classifier, whilst those of the remaining 6 weeks are used for testing and retraining. The incremental learning is doe in the following process. After the initial training using the data of the first 2 weeks, feature vectors for week 3 are test against the model obtained from initial training. Then, the SVM classifier is retrained with all feature vectors from first 3 weeks. The above process is repeated for the rest of the weeks until feature vectors for week 8 are finally included in the training.

The results without and with the incremental learning are summarized in Table 5. The left half of Table 5 shows that DDoS events can be fairly-accurately detected without incremental learning. Especially, recalls reach almost one, that is, almost all the DDoS events are detected. This indicates that the 17 features and the classifiers can capture the difference between DDoS backscatters and non-DDoS backscatters, and therefore are effective for DDoS event detection. The right half of Table 5 shows that, by the incremental learning, the detection performance is further improved in all weeks except for week 5. This implies that activity patterns become diverse over time and the incremental learning enables the system to respond to the diversification.

As listed in Table 5, as long as the training and testing are done for data generated with in a few weeks, computational time is not critical. However, for such a long time monitoring project as the NICTER, an online learning scheme that could effectively treat with the incoming data will be sought as future study.

4.3 Discussions and summary

As shown in the previous subsection, the classification performance can be improved by incremental learning. This implies that new activity patterns appeared over time. To visualize such changes and diversification of activity patterns over time, we used a dimensionality-reduction method know as t-SNE [18]. By using t-SNE, the 17-D feature vectors are reduced to 2-D vectors and shown in scatter plots in Fig. 2. Figure 2(a)–(c) represent data observed during Jan. 1st to Jan. 7th (the first week), to Feb. 28th (about the first 8 weeks), and to June 31st, respectively. Red and blue indicate the DDoS events and non-DDoS events observed during the first 8 weeks, respectively. Green indicates unlabeled data collected after the first 8 weeks, which were not used in the analysis in Section **4.2**. Compared with the distribution in Fig. 2(a), both the distributions of DDoS events and non-DDoS events in Fig. 2(b) spread more widely. This means that activity patterns became more diverse over time.



(a) 1 week

(b) 8 weeks

(c) 6 months

Fig. 2 Visualization of the host activities using t-SNE. Plots represent data observed from Jan. 1st to Jan. 7th (a), to Feb. 28 (b), and to June 31 (c), in 2014, respectively. (Figure reused from [17])

Furthermore, clusters that are not in Fig. 2(a) emerge in Fig. 2(b), indicating new types of activity patterns appeared. Additionally, after the first 8 weeks, the distributions become wider and new clusters emerge, as shown in the insets in Fig. 2(c). These results reveal that the activity patterns of the hosts change over time and so that, to distinguish such new patterns, incremental learning is needed.

5 Early detection of emerging threats

Traffic data captured on a darknet contain valuable forensic information of programming techniques that are exploited to scan the Internet. In this section, we describe the application of association rule learning to characterize the behavior of attacking hosts observed in the darknet [19].

5.1 Association rule learning

The problem of association rule learning was originally proposed in the context of market basket data in order to find frequent groups of items that are purchased together [20]–[22]. Following the original definition in [20], the problem of association rule learning is defined as follows.

Let $D = \{T_1, T_2, ..., T_N\}$ be a set of N transactions called the database. Let $I = \{i_1, i_2, ..., i_M\}$ be the universal set of all M items present in the database. Each transaction in D has a unique transaction ID and contains a subset of the items in I. The support s(X) of a set of item (for short itemset) X is defined as the number/proportion of transactions in the database, which contain the itemset.

Frequent pattern mining is to determine all patterns $P \subset I$ that are present in at least a fraction S of the transactions. The fraction S is referred to as the minimum support. It can be expressed either as an absolute number, or as a fraction of the total number of transactions in the database.

An association rule is defined as an implication of the form

$$X \rightarrow Y, \text{ for } X, Y \subseteq I, X \cap Y = \phi \quad (1)$$

The itemsets X and Y are called antecedent and consequent of the rule respectively. The confidence of a rule is presented by the conditional probability, P(Y|X), i.e.,

$$\text{conf}(X|Y) = s(X \cap Y)/s(X) \quad (2)$$

To select interesting rules from the set of all possible rules, rules that satisfy both a minimum support threshold, S_0, and a minimum confidence threshold, C_0, are called strong.

In general, association rule learning can be done in two steps:

1) Frequent pattern mining: Search for itemsets that satisfy the minimum support in a power set of all possible combination of items. Efficient algorithm such as Apriori [20] and FP-tree [21] exist which make use of the following Apriori Property: *All nonempty subsets of a frequent itemset must also be frequent. Thus for an infrequent itemset, all its supersets must also be infrequent.*
2) Strong association rule generation: For each frequent itemset l, generate all nonempty subset of l. For every nonempty subset s of l, output the rule s→(l-s) if its confidence is higher than minimum confidence threshold C_0. Since the rules are generated from frequent itemsets, all association rules created in such a way automatically satisfy the minimum support.

5.2 Application to attacking-host behavior characterization

Discovery of behavior regularities of the attacking host may complement existing malware countermeasures in the following aspects. First, discovery of prevalent attack patterns may lead to further insights into the mechanism of the attack and thus enables countermeasure for the attack. Second, the emergence of new attack patterns/graphs may be the symptom of pandemic incidents whose early detection and takedown could lead to prevention of heavy loss. Finally, such information can be used to improve the performance of monitoring systems so that more pertinent malware information can be collected using limited system and network resources.

In the following we present an example of association rule learning that exploit the correlation among attacked destination ports. Network ports, which provide essential identifying information for open services, are the entry points to any networked device. The port number, identified by a 16-bit number, together with a device's IP address, completes the destination address for a communication session. The open ports on a device are usually probed by malware to determine available services before exploitation of known vulnerability on the service.

Discovered strong association rules with regards to the destination ports could provide useful information in the following aspects. First, because different malware pro-

Table 6 Frequent itemsets related to destination port 80, obtained from 1-day traffic of a /16 darknet sensor

ID	Dest. Port 1	Dest. Port 2	Dest. Port 3	Dest. Port 4	Occurrence
1	80				2,932
2	80	8			747
3	80	443			786
4	80	13	443		715
5	80	8	13		741
6	80	8	443		713
7	80	13	443		712
8	80	8	13	443	711

The network services on involved ports are as follows. Port 8: service unassigned, port 13: the daytime protocol, port 80: hypertext transfer protocol (HTTP), port 443: hypertext transfer protocol over TLS/SSL (HTTPS).

grams usually exploit different combinations of vulnerable ports, the destination ports may provide deterministic information to identify the specific malware or offer hints to the intent of the attacker. Therefore, frequent pattern mining can be an efficient approach to automated malware signature extraction. Second, frequently probed port sets can reveal the most vulnerable services and therefore provide valuable clues for malware diagnosis.

5.3 Mining high-order correlation among destination ports

To discover the correlation among destination ports, the mining problem on destination ports is formulated by defining the set of unique port numbers probed by an attacking IP in one day as the transactions in the database. Table 6 shows the frequent itemsets learned from a 1-day traffic trace of a /16 sensor. The minimum support is set to 700. Eight frequent itemsets, which are related to the well-known port 80, are selected from a pool of 610 frequent itemsets. Because of the popularity of port 80 used for hosting web service, many attacks tend to probe this port. As shown in the table, 2,932 hosts had attacked port 80 on the day. Many ports are probed together with port 80, among which are ports 8, 13, and 443. In the table, all the frequent itemsets that are highly related to these 4 ports are shown, with the number of their occurrences shown on the last column. Obviously, ports 8, 13, and 443 have a strong correlation, i.e., they tend to be probed at the same time.

This is confirmed by the association rules shown in Table 7, which are generated from the frequent patterns in Table 6. In the table, despite of the high number of co-occurrence between ports 80 and 13, the association rule $P80 \rightarrow P13$ only has a confidence of 24.3%, failing to meet the minimum confidence requirement 80%. On the contrary, the association rule $P13 \rightarrow P80$ has a strong confidence of 94.7%. Therefore, probes to port 13 can be considered as the causal factor of the probes to port 80, e.g., if a packet directed to port 13 is observed from a host, then port 80 has a large chance to be probed.

Take the rules 5 to 7 of in Table 7 as another example. These three rules illustrate the correlation between ports 8, 80, and 443. If two of the ports are probed, the chance for the other port to be probed is over 94%. Because of the high correlation of these three ports, they can be treated as the signature of the scanning behavior.

Table 7 Association rules created from frequent itemsets in Table 6

ID	Rule	Support	Confidence
1	$80 \rightarrow 8$	747	27.5%
2	$8 \rightarrow 80$	747	4.7%
3	$80 \rightarrow 13$	715	24.3%
4	$13 \rightarrow 80$	715	94.7%
5	$80, 443 \rightarrow 8$	741	94.3%
6	$8, 443 \rightarrow 80$	741	95.5%
7	$8, 80 \rightarrow 443$	741	99.2%
8	$13, 443 \rightarrow 80$	712	95.3%
9	$80, 443 \rightarrow 13$	712	90.6%
10	$13, 80 \rightarrow 443$	712	99.6%
11	$8, 13 \rightarrow 80$	713	95.2%
12	$8, 80 \rightarrow 13$	713	95.4%
13	$13, 80 \rightarrow 8$	713	99.7%
14	$13, 8, 443 \rightarrow 80$	711	95.4%
15	$8, 80, 443 \rightarrow 13$	711	96.0%
16	$13, 80, 443 \rightarrow 8$	711	99.9%
17	$8, 13, 80 \rightarrow 443$	711	99.7%

The first three rules which do not satisfy the minimum confidence C = 80% are not considered as strong association rules.

5.4 Summary

The strong association rules discovered in the above experiments indicate that the strongly correlated destination ports could be the identifying signatures of malware variants. However, to prove this, information from other data sources are needed to give precise information of the malware programs performing the probes. In fact, the above findings are confirmed to be associated with the Carna botnet[24]. The Carna botnet was created by intruding more than 420,000 embedded devices that were accessible online with default credentials. After the intrusions, a small binary are uploaded to those devices to conduct an Internet-wide scan of the IPv4 address space. The owners of the Carna botnet claimed that the botnet was created for research purpose and they published a detailed description of how they operated, along with 9TB of raw logs of the scanning activity. According to previous work in [25], probes to ports 8, 80, and 433, and probes to ports 23 and 210 are reported as the signatures of the network scans performed by different fractions of the botnet.

In [23], the above discoveries are extended to identify the newest types of attacks at their early stage so as to facilitate proactive countermeasure of these cyber-threats.

6 Conclusions

In this paper we present the NICTER, a global-scale darknet-monitoring project, with a focus on the backend analysis engines supporting its incident report and handling. Despite the lack of overall information about the attacking hosts observed in the darknet, we show that analysis of the collected attacking traffic yields revealing interesting regularities of the attack and contributes to the countermeasure of malware. We believe strategic countermeasure to related attacks can be enabled based on these discoveries.

References

1 M. Bailey, E. Cooke, F. Jahanian, J. Nazario, D. Watson, et al., "The internet motion sensor – a distributed blackhole monitoring system," NDSS, 2005.

2 D. Inoue, K. Yoshioka, M. Eto, M. Yamagata, E. Nishino, J. Takeuchi, K. Ohkouchi, and K. Nakao, "An incident analysis system NICTER and its analysis engines based on data mining techniques," ICONIP 2008, Part I. LNCS, vol.5506, pp.579–586, 2009.

3 T. Ban, L. Zhu, J. Shimamura, S. Pang, D. Inoue, and K. Nakao, "Behavior analysis of long-term cyber-attacks in the darknet," 19th International Conference Neural Information Processing (ICONIP 2012), Part V, vol.151, no.3, pp.620–628, 2012.

4 U. Harder, M. W. Johnson, J. T. Bradley, and W. J. Knottenbelt, "Observing internet worm and virus attacks with a small network telescope," Electronic Notes in Theoretical Computer Science, vol.151, no.3, pp.47–59, 2006.

5 K. Benson, A. Dainotti, K. Claffy, and E. Aben, "Gaining insight into as-level outages through analysis of internet background radiation," in Computer Communications Workshops, INFOCOM, pp.447–452, 2013.

6 K. Nakao, K. Yoshioka, D. Inoue, and M. Eto, "A novel concept of network incident analysis based on multi-layer observations of malware activities," The 2nd Joint Workshop on Information Security (JWIS 2007), pp.267–279, 2007.

7 D. Inoue, M. Eto, K. Yoshioka, S. Baba, K. Suzuki, J. Nakazato, K. Ohtaka, and K. Nakao, "NICTEr: An incident analysis system toward binding network monitoring with malware analysis," WOMBAT Workshop on Information Security Threats Data Collection and Sharing (WISTDCS 2008), pp.58–66, 2008.

8 D. Inoue, M. Suzuki, M. Eto, K. Yoshioka, K, Nakao, "DAEDALUS: Novel application of large-scale darknet monitoring for practical protection of live networks," 12th International Symposium on Recent Advances in Intrusion Detection, LNCS 5758, pp.381–382, 2009.

9 D. Inoue, M. Eto, K. Suzuki, M. Suzuki, and K. Nakao, "DAEDALUS-VIZ: novel real-time 3D visualization for darknet monitoring-based alert system," In Proceedings of the Ninth International Symposium on Visualization for Cyber Security (VizSec 2012), pp.72–79, 2012.

10 K. Suzuki, M. Eto, and D. Inoue, "Evaluation of NIRVANA: Real network traffic visualization system," Journal of the National Institute of Information and Communications Technology, vol.58, no.3/4, pp.61–77, 2011.

11 D. Song, R. Malan, and R. Stone, "A snapshot of global Internet worm activity," 14th Annual FIRST Conference on Computer Security Incident Handling and Response, 2002.

12 D. Moore, "Network telescopes: tracking denial-of-service attacks and Internet worms around the globe," 17th Large Installation Systems Administration Conference (LISA 2003), USENIX, 2003.

13 M. Bailey, E. Cooke, F. Jahanian, J. Nazario, and D. Watson, "The Internet motion sensor: A distributed blackhole monitoring system," 12th Annual Network and Distributed System Security Symposium (NDSS 2005), 2005.

14 F. Pouget, M. Dacier, and V. H. Pham, "Leurre.com: On the advantages of deploying a large scale distributed honeypot platform," E-Crime and Computer Conference (ECCE 2005), 2005.

15 C. Leita, V. H. Pham, O. Thonnard, E. Ramirez-Silva, F. Pouget, E. Kirda, and M. Dacier, "The Leurre.com project: Collecting threats information using a worldwide distributed honeynet," WOMBAT Workshop on Information Security Threats Data Collection and Sharing (WISTDCS 2008), pp.40–57, 2008.

16 V.N. Vapnik, "The Nature of Statistical Learning Theory," Springer, 1995.

17 N. Furutani, J. Kitazono, S. Ozawa, T. Ban, J. Nakazato, and J. Shimamura, "Adaptive DDoS-event detection from big darknet traffic data," ICONIP, vol.4 pp.376–383, 2015.

18 L., Van der Maaten, and G. Hinton, "Visualizing data using t-SNE," Journal of Machine Learning Research, vol.9, pp.2579–2605, 2008.

19 T. Ban, M. Eto, S. Guo, D. Inoue, K. Nakao, and R. Huang, "A study on association rule mining of darknet big data," IJCNN 2015, pp.1–7, 2015.

20 R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in ACM SIGMOD Record, vol.22, no.2. ACM, pp.207–216, 1993.

21 J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in ACM SIGMOD Record, vol.29, no.2. ACM, pp.1–12, 2000.

22 C. Borgelt, "Frequent item set mining," Data Mining Knowledge Discovery, vol.2, no.6, pp.437–456, 2012.

23 T. Ban, S. Pang, M. Eto, D. Inoue, K. Nakao, and R. Huang, "Towards early detection of novel attack patterns through the lens of a large-scale darknet," submitted to ATC 2016.

24 C. Stocker and J. Horchert, "Mapping the Internet: A hacker's secret Internet census," Spiegel Online, 22/3, 2013.

25 E. Le Malecot and D. Inoue, "The carna botnet through the lens of a network telescope," in Foundations and Practice of Security, Springer, pp.426–441, 2014.

Tao BAN, Dr. Eng.

Senior Researcher, Cybersecurity Laboratory,
Cybersecurity Research Institute
Cybersecurity, Network Security