A study of IoT malware activities using association rule learning for darknet sensor data

Ozawa, Seiichi

Ban, Tao

Hashimoto, Naoki

Nakazato, Junji

Shimamura, Jumpei

(Citation)
International Journal of Information Security,19(1):83-92

(Issue Date)
2020-02

(Resource Type)
journal article

(Version)
Accepted Manuscript

(URL)
https://hdl.handle.net/20.500.14094/90006848

Eur. Phys. J. C manuscript No.
(will be inserted by the editor)

A Study of IoT Malware Activities Using Association Rule Learning for Darknet Sensor Data

Seiichi Ozawa [a,1], **Tao Ban** [b,2], **Naoki Hashimoto** [1], **Junji Nakazato** [3], **Jumpei Shimamura**[4]

[1] Kobe University, 1-1 Rokko-dai, Nada-ku, Kobe 657-8501, Japan
[2]National Institute of Information and Communications Technology, Japan
[3]Security Research Laboratory, Fujitsu Laboratories LT, Japan
[4]clwit Inc., Japan

Received: date / Accepted: date

Abstract Along with the proliferation of IoT (Internet of Things) devices, cyberattacks towards these devices are on the rise. In this paper, we present a study on applying Association Rule Learning (ARL) to discover the regularities of these attacks from the big stream data collected on a large scale darknet. By exploring the regularities in IoT-related indicators such as *destination ports*, *type of service* (ToS), and *TCP window sizes*, we succeeded in discovering the activities of attacking hosts associated with well-known classes of malware programs. As a case study, we report an interesting observation of the attack campaigns before and after the first source code release of the well-known IoT malware *Mirai*. The experiments show that the proposed scheme is effective and efficient in early detection and tracking of activities of new malware on the Internet and hence induces a promising approach to automate and accelerate the identification and mitigation of new cyber threats.

Keywords cybersecurity · machine learning · IoT malware · association rule learning, · darknet traffic analysis

1 Introduction

Information technologies (IT) have brought drastic changes in our life and many people have enjoyed new benefits from the Internet. In recent years, in addition to this IT revolution, the great progress of the Internet of Things (IoT), where various services and devices are connected to the Internet, is about to bring us further revolution. However, along with sophistication of IT and IoT systems, cyberattacks exploiting new system vulnerabilities are becoming serious these days. In particular, the impact of a recent IoT malware *Mirai* was enormous. Mirai is a worm-type malware that finds an IoT device with similar vulnerability for self-replication. IoT devices infected by Mirai can then be manipulated by attackers to perform Distributed Denial of Service (DDoS) attacks by seeding a large number of packets to target hosts.

In order to deal with such large-scale cyberattacks in a timely fashion, it is necessary to devise a means that is capable of observing cyberattacks occurring on the Internet from a global view. For this purpose, the use of the darknet, *a.k.a* a *network telescope*, has been studied for many years [1, 2]. Darknet is an unused address space. It is considered that no communication occurs on the darknet because it is not connected to any device, nevertheless, a remarkable amount of packets is monitored on a yearly basis. These packets are mainly caused by scan activities or backscatter from DDoS-attacked hosts; thus, it can be considered that most packets observed in the darknet have close relationship with malware. Therefore, through the analysis of darknet packets, it is possible to reveal the characterizing features of the cyberattacks on the Internet.

In this research, we analyze the behavior of scan attacks from packets observed in darknet. In particular, we focus on TCP SYN packets to characterize scan attacks, searching for statistically reliable regularities from those packets. For this purpose, we apply association rule learning to SYN packets and discuss the dynamic features of malware that performs scan attacks. As for the destination port information, there have been reported several prior works analyzing SYN packets. Ban et al. [1] and some researchers [3, 4] applied association rule learning to destination ports of SYN packets, and they discovered several association rules related to Carna botnet and other malware. These rules are currently used as signatures to identify the hosts that perform network scans. In this paper, in addition to the analysis on destination ports, we also explore the regularity on other indicators such as the *window size* and *type of service*.

[a]e-mail: ozawasei@kobe-u.ac.jp, Tel/Fax.: +81-78-8036466
[b]e-mail: bantao@nict.go.jp

This paper is organized as follow. Section 2 briefly explains the darknet analysis and association rule learning based on FP-tree/FP-growth algorithms. In Section 3, we present the rationality for mining over the IoT-related indicators. In Section 4, the proposed analysis is applied to find useful traffic patterns on a specific scanning attack from a large set of TCP SYN packets collected before and after the Mirai outbreak. Section 5 gives our conclusions of this paper and future work.

2 Association Rule Learning

This section briefly explains association rule learning, a commonly applied technique for discovering interesting relationships hidden in a database.

2.1 Frequent Pattern Mining

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 [5, 6]. Following the original definition in [5], the problem of association rule learning is defined as follows.

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

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

$$X \to Y, \text{ for } X, Y \subseteq I, X \cap Y = \emptyset. \tag{1}$$

The item sets 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.,

$$\mathrm{conf}(X \Rightarrow Y) = \mathrm{supp}(X \cup Y)/\mathrm{supp}(X). \tag{2}$$

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

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

1. Frequent pattern mining: Each of the item sets will satisfy the minimum support, i.e., occurs at least as frequently as S.
2. Strong association rule generation: By definition, rules created from frequent item sets with minimum support must satisfy a minimum confidence constraint.

2.2 Frequent Pattern Mining Using FP-tree

The first step in association rule learning involves searching in a power set of all possible combinations of items, whereas the size of this set grows exponentially in the number of items n in $\mathscr{I}$. The key to an efficient search algorithm is the so-called a priori property: All nonempty subsets of a frequent item set must also be frequent. Thus for an infrequent item set, all its supersets must also be infrequent. One of the currently fastest and most popular algorithms for frequent item set mining is the Frequent Pattern growth (FP-growth) algorithm [6–8]. It is based on a prefix tree representation of the given database. By using a prefix tree data structure - the so-called FP-tree - FP-growth can save considerable amounts of memory for storing the transactions. The basic idea of the FP-growth algorithm can be described as a recursive elimination scheme as follows.

1. In the first pass, derive the set of frequent items and their support counts. Delete all items from the transactions which do not satisfy the minimum support constraint. All frequent items are stored in a header table in descending order of their frequency.
2. In the second pass, build an FP-tree by inserting instances into a tree with a root node labeled as ‘null’. To speed up the processing of the FP-tree, items in each transaction are sorted in the same order as in the header table. All nodes referring to the same item are indexed by a list so that all transactions containing the item can be accessed and counted by traversing this list. The header elements to the list are associated with the corresponding items in the header table.
3. Recursive mining of the FP-tree can grow large item sets directly, without generating candidate items and testing them against the entire database. Start from the bottom of the header table, build the conditional item base for the length-1-pattern, which consists of a set of prefix paths in the FP-tree co-occurring with the suffix item. Then, a conditional FP-tree is created, with counts projected from the original tree corresponding to the set of instances that are conditional on the attribute, with each node getting sum of its children counts. Recursive growth ends when no individual items conditional on the attribute meet the minimum support threshold, and processing continues on the remaining header items of the original FP-tree.
4. Once the recursive process has completed, all large item sets satisfying the minimum support constraint is found, and association rule creation begins.

2.3 Association Rule Generation from Frequent Item sets

Association rules can be generated based on the frequent item sets in the following steps.

1. For each frequent item set l, generate all nonempty subset of l.
2. For every nonempty subset s of l, output the rule "$s \rightarrow (l - s)$" if its confidence is higher than minimum confidence threshold C.

Since the rules are generated from frequent item sets, all association rules created in such a way automatically satisfy the minimum support.

3 Methodology

Darknet monitoring provides a global overview of the attack campaigns happened on the Internet. Meantime, by mining and discovering behavioral regularities of the attacking hosts (AHs), darknet monitoring complements conventional malware countermeasures in the following aspects. First, discovery of prevalent attack patterns may lead to further insight into their nature and hence enable appropriate countermeasure against them. Second, the emergence of new attack patterns may be the symptom of pandemic incidents whose early detection and take-down could lead to prevention of heavy loss. Finally, knowledge of attacks can be used to improve the performance of monitoring systems so that more pertinent malware information can be collected using reduced system and network resources.

3.1 Darknet Traffic Analysis

A darknet is a routed but unused IP address space on the Internet. A simple deployment of darknet could be realized by assigning all unused IP addresses in a network to a network interface card (NIC) on a designated server with firewall rules specified to restrain the NIC from sending any egress traffic to the Internet. Due to the absence of legitimate hosts or open services on a darknet, any traffic observed on a darknet is considered as aberrant: it is either caused by some malicious intents (e.g., probing packets from malware infected hosts or reflection packets from DDoS attacked servers) or by misconfiguration (e.g., packets sent from a PC to connect to a printer with misconfigured IP address). Darknet monitoring consists in the fact that most kinds of self-spreading malware engage an exploitation phase sending out scanning packets to the Internet in the aim of searching for the next potential victims.

Attacks towards the darknet arrives in the form of network packets. See Fig. 1 for an illustration of how packets are monitored on the darknet. The majority of traffic observed on the darknet is composed of three types of packets:

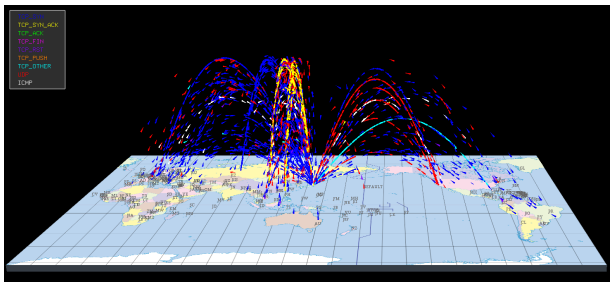

Fig. 1 Illustration of attacking packets towards the darknet.

TCP SYN packets, TCP SYN-ACK packets, and UDP packets (shown as blue, yellow, and read rockets in Fig. 1.) The first two types of packets are associated with the Transmission Control Protocol (TCP), one of the main protocols of the Internet protocol suite. TCP provides reliable, ordered, and error-checked delivery of contents between applications running on hosts communicating via an IP network. TCP uses a three-way handshake to establish the connection between hosts. First, a client sends a SYN packet to the server to initialize a connection. Second, in response to the SYN packet from the client, the server replies with a SYN-ACK packet. In the final step, the client sends an ACK packet back to the server. The last type of packets are associated with the User Datagram Protocol (UDP) where computer applications can send messages to other hosts on an Internet Protocol (IP) network without prior communications such as three-way handshake dialogue like TCP. UDP is suitable for time-sensitive applications where error checking and correction are either not necessary or are performed in the application.

The statistics of different packets arrived at a darknet composed of approximately 300,000 IP addresses in 2017 is shown in Table 1. As seen in Table 1, the monthly average number of TCP SYN packets was rapidly increased from 2016 to 2017 and it trend was kept even in 2018. TCP SYN packets carries the most interesting information not only for the fact that they constitute the majority of the darknet traffic but also for the reason that they carry essential and reliable information about the malware attack campaigns. When scanning the Internet to search for the next victim, an AH tends to send a TCP SYN packet to a target host (TH). After that the AH have to receive the replying SYN-ACK packet from the TH to confirm the existence of the TH. Therefore, it is unlikely that obfuscation techniques such as IP-address spoofing is applied in this process.

It is worthwhile to note that to prevent designated attacks towards itself, a darknet generally does not reply to any received packets. Therefore, for most of the time, only communication initializing packets can be observed on the darknet. This renders the darknet traffic more fragmented, lack-

Table 1 The statistics of packets captured on a /16 darknet sensor from July 1st, 2016 to July 31st, 2018. The first figure in each cell corresponds to the total of darknet packets and the second figure in a bracket corresponds to the monthly average over a year.

Period	7/1/2016-12/31/2016	1/1/2017-12/31/2017	1/1/2018-7/31/2018
TCP Total	4,886,278,355 (26,700,975)	16,172,516,882 (44,429,991)	8,798,643,714 (41,699,733)
TCP SYN	4,674,026,073 (25,541,126)	14,637,259,441 (40,212,251)	7,799,946,543 (36,966,571)
TCP SYN-ACK	183,161,569 (1,000,883)	1,409,423,383 (3,872,042)	695,719,791 (3,297,250)
TCP Others	29,090,713 (158,966)	125,834,058 (345,698)	302,977,380 (1,435,912)
UDP	350,030,217 (1,912,733)	1,503,095,713 (4,129,384)	855,714,758 (4,055,520)

ing application level information. On the other hand, when the scale of the darknet reaches a certain size, packets sent from a single AH towards a series of THs has a large chance to be captured, revealing the lateral regularities of the malware. Occurrence of a remarkable number of AHs with great similarity in lateral regularities and temporal correlation, indicates an attack campaign happened on the darknet.

3.2 Association Rule Mining in Darknet Traffic

The activity of sending TCP SYN packets towards a designated IP address to confirm its existence and determine its vulnerability is known as a *network vulnerability scan* (hereafter referred as a scan). Scans monitored on the darknet spread along two dimensions: destination ports and destination hosts. First, an AH tends to probe multiple destination ports on a TH to identify network services running on the TH and exploit vulnerabilities therein. Then, it tends to replicate the same exploitation towards a range of hosts by simply changing the probed destination IP. As aforementioned, a darknet could only monitor a comparable small portion of all the scans from an AH, however, for the sake of highly replicated nature of scans from the same malware, deterministic characteristics can be derived from the monitored data by advance data mining tools.

The regularities in the scanning packets can be explored at various levels. First, at packet level, scans towards a certain vulnerability confined to a specific destination port may have different probing pattern. Second, at target host level, an attack can be featured by a series of packets sent to the TH in a predefined order towards a combination of destination ports. Then, at the network-level, there might be precoded rules to select the next TH. Finally, at meta-level, the strength, frequency, and rhythm of the packets may reveal the existence of an AT. All of the above information can be indicators to characterize the AHs and in turn reveals the trend of emerging network attacks and status of malware contamination.

Association rule learning can be applied to transaction sets defined upon indicators of regularities in malware communication. Among the many fields that can be found in the IP and TCP headers of the packets, we chose three indictors of interest to investigate: *destination port*, *sequence number*, and *window size*. Then, we define a *transaction set* for each indicator of interest. Each transaction in the set contains a series of unique indicator values observed in the communication from an AH towards the darknet during a 24-hour period. Finally, association rule learning is applied to the transaction set to extract the most significant correlation between the indicators.

Note that due to the dynamic IP address allocation mechanisms using Dynamic Host Configuration Protocol (DHCP), the packets from a single IP address in a long run may contain communication of multiple AHs. On the other hand, because the scale of the darknet under discussion constitutes to a comparatively small portion of the IPV4 space, the chance for multiple independent hosts scanning the same darknet is small enough to be ignored. It is reasonable to take all the packets launched by an AH during a 24-hour period as from a single source.

3.2.1 Destination Ports

Network ports, which provide 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 host's IP address, completes the destination address for a communication session. Network ports on a host, i.e., the destination port in the communication, are usually probed by malware to determine open services before exploitation of known vulnerability on the service. Due to the close interdependence between a service and its hosting port, vulnerabilities listed in open vulnerability databases [9] are often specified by a port number rather than the service hosted on the port.

Years ago, it was more common for malware programs to probe a range of THs on a specific port with known vulnerabilities. Recent analysis on the scanning activities reveal that more malware programs tend to probe a couple of destination ports once at a time. This may attribute to the fact that, nowadays there are much more types of devices with similar vulnerabilities hosted on different ports are connected to the Internet, e.g. the IoT devices. Fortunately, the

increase in number of distinct sets of destination ports renders port combination a more deterministic identifier of the probing malware.

3.2.2 Time of Service

Like the combination of destination ports, many other fields in IP/TCP header could also provide hints to identify the packet issuing application. Among these is the *type of service* (ToS) field, the second byte of the IPv4 header. The ToS facility has been a part of the IP specification since the beginning, it has been little used in the past. The modern redefinition of the ToS field is presented in IETF RFC 2474 [10], and the Internet host specification [11] now mandates that hosts use the ToS facility. Additionally, routing protocols [12] have been developed which can compute routes separately for each ToS, rendering it practical for routers to consider the requested ToS when making routing decisions.

ToS can be hard-coded in the malware which often happens as many attackers prefer hand crafting the packet better than using normal socket Application Interfaces (APIs) to implement spoofing. Hence ToS is one of the promising indicators that may carries identifying information about the malware. For the same reason, Cisco standard NetFlow version 5 [13] records ToS as a field in its export datagram formats. In practice, ToS is also used as one of the features to detect DDoS and botnet activities [14].

3.2.3 TCP Window Size

Another indicator of interest is the window size field in TCP header. The *window size* is an option to increase the receive window size allowed in TCP communication above its former maximum value of 65,535 bytes [15]. For more efficient use of high-bandwidth networks, a larger TCP window size may be used. The window scale option is used only during the TCP 3-way handshake, which particularly suits the data analysis in darknet. Many malware programs have reported to abuse customized window size for Internet communication [16].

4 Experiments

In this section, we carry out a darknet analysis to find meaningful traffic patterns of specific scanning attacks using the association rule learning mentioned in Section 2. First, we explain how to make transaction sets for the FP tree algorithm. Then, we study some interesting scanning behaviors of IoT malwares such as Mirai and Hajime.

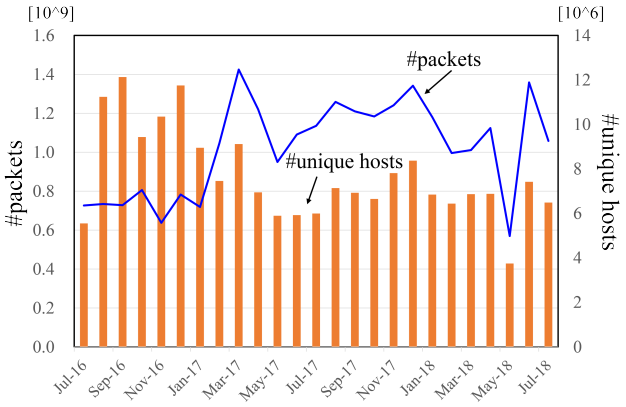

Fig. 2 The number of packets observed in the NICT /16 darknet sensor per month from July 1st, 2016 to July 31st, 2018, and the number of unique hosts sending such packets.

4.1 Data Preparation

To evaluate the proposed association rule learning, we use a large set of TCP SYN packets collected from July 1st, 2016 to July 31st, 2018 (25 months) with the NICT /16 darknet sensor. We collected 25,533,925,844 packets in total that were sent from 101,206,481 unique hosts. Figure 2 shows the number of TCP packets observed per month, and the number of unique hosts sending such packets to the darknet sensor.

In this experiment, the association rule learning is conducted day by day for all active source hosts; thus, a daily set of darknet sensor packets is first split into subsets of packets sent from each source IP and the association rule learning is applied to each of the subsets to investigate malicious behaviors of a specific host which is supposed to get infected with malwares. As mentioned in 3.2, we focus on *destination port*, *TCP sequence number*, and *TCP window size* to discover useful association rules among all active hosts. Therefore, we define three types of transaction sets for darknet sensor packets. Let $\mathscr{D}^k(d) = \{T_1^k(d), T_2^k(d), \cdots, T_N^k(d)\}$ $(k \in \{\text{destination port}, \text{TCP sequence number}, \text{TCP window size}\})$ be the database of the kth attribute on Day d where $T_j^k(d)$ is the kth transaction set of packets sent from the jth host on Day d. The association rule learning is applied to each of the three databases $\mathscr{D}^k(d)$ everyday from July 1st, 2016 to July 31st, 2018.

4.2 Experimental Setup

As mentioned in 2.1, there are two parameters to be preset: a minimum support threshold, S, and a minimum confidence threshold, C. If these parameters are set to a smaller value, a lot of association rules will come out and it makes difficult

for us to understand the behaviors of targeted source hosts. Therefore, we try to find suitable values of C and S so that only significant rules can be found. Here, we use the following parameter values: $C = 1,000$ and $S = 90$.

According to [17], Mirai can be fingerprinted by the following feature:

$$\text{sequence number} \;=\; \text{destination IP}. \tag{3}$$

This feature is used as a signature of *Mirai* and judge that a host got infected with *Mirai* if over 90% packets sent by this host reveal the feature in Eq. (3). On the other hand, another type of IoT malware called *Hajime* can be fingerprinted by the following feature:

$$\text{TCP window size} \;=\; 14,600. \tag{4}$$

Thus, we use this feature to define the signature of *Hajime*: a host is judged if over 90% packets sent by this host reveal the feature in Eq. (4).

Tables 2(a)-(c) show the obtained association rules for destination ports, sequence numbers, and TCP window sizes, respectively. Type represents the type of malware infecting a host. Here, M, H, and U correspond to Mirai, Hajime, and Unknown, respectively.

As seen from Tables 2(a)-(c), for destination ports, we found 26 association rules for 25 months, while only 9 and 2 rules were found for ToS and window sizes, respectively. Interestingly, Tables 2(b) and (c) show that Mirai and Hajime left their fingerprinting features in association rules on ToS and window sizes as well.

In the following subsections, we discuss what kind of features on malware activities can be known from the proposed darknet analyses with association rule mining.

4.3 Study on Darknet Traffic around Mirai Outbreak

To see how the proposed rule mining method works in the darknet analysis, let us focus on the period from July 1st, 2016 to October 30th, 2016, which was around the outbreak of a notorious IoT malware *Mirai*. In this analysis, we collect 2,188,183,040 packets which were sent from 9,640,067 unique hosts on average.

As seen in Fig. 2, there are 3 rules in total related to Mirai during this period: 1 rule on destination ports, 1 rule on ToS, and 1 rule on TCP window size. Figure 3 illustrates the transitions in the number of hosts associated with these 3 rules. Interestingly, about 15,000 hosts matched with a rule of window size, (1320, 2376) $\rightarrow$ 792, first appeared on August 2nd, and the number reached up about 37,000 hosts, and disappeared suddenly on September 4th, only 3 days before the first source code of *Mirai* was opened in a community forum. Then, this rule never appeared after September 4th.

On the other hand, another two rules on destination ports and ToS came out on September 5th and 15th, respectively. The rule on destination port reveals a steady Mirai feature which continuously showed up until the end of our observation (see Table 2(a)). This is because this rule is related to the scanning activity to find vulnerable IoT devices for an intrusion purpose. On the other hand, as discussed in 3.2.2, the ToS header information is useful to identify a malware type (i.e., a Mirai variant). Therefore, it is considered that an early version of Mirai adapted to IoT devices was launched in South East Asia, mainly in Vietnam (see the country information in Table 2(b)), around September 15th, 2016, and was stopped at the end of September, 2016.

From the above discussions, we can infer the following story about the early stage of the Mirai pandemic. It is assumed that the original Mirai code was testified from around August 2nd to around September 4th, 2016. Then, the source code was posted in a dark web forum with the feature on TCP window sizes erased after which it was distributed to attackers.

4.4 Study on Darknet Traffic after Mirai Outbreak

We continue to analyze TCP SYN packets collected with the NICT /16 darknet sensor for one year after the Mirai outbreak. The number of collected packets is 23,345,742,804 packets in total which were sent from 91,566,414 unique hosts.

From Table 2, we can see that there are 19 rules in total related to Mirai after the Mirai outbreak; 16 rules on destination ports, 3 rules on ToS, and no rule on TCP window size. On the other hand, the rules related to another type of IoT malware called *Hajime* emerged after February 2017.

Figure 4 shows the time lines of association rules extracted from July 1st, 2016 to July 31st, 2018. In Fig. 4, the red and blue arrows show the periods that the rules with Mirai and Hajime features in Eqs. (3) and (4) emerged. The green arrows show the periods for the rules that do not match with the features in Eqs. (3) and (4).

In Fig. 4, soon after the Mirai outbreak (i.e., September 2016), we can see that various Mirai variants emerged and they used destination ports 22 (SSH), 2222, 80, 8080 (HTTP), 5358, 6789, 19058, 23231, 37777 other than 23 and 2323 (Telnet). For example, the port scan to 22 and 2222 were observed from the late December 2016, and this was also reported on other sources on Mirai activity (e.g., Mirai Scanner [18]). The rules related to Hajime emerged from the mid-February 2017 and it is well known that Hajime shuts the access to ports 23, 7547, 5555, 5358 out. Interestingly, we can see from Fig. 4 that Mirai rules using such destination ports disappeared after several Hajime variants got activated from the mid February 2017 to the late July 2017. Then, different types of Mirai using destination ports 80, 81, 8080, 8081 and other ports emerged after October 2017.

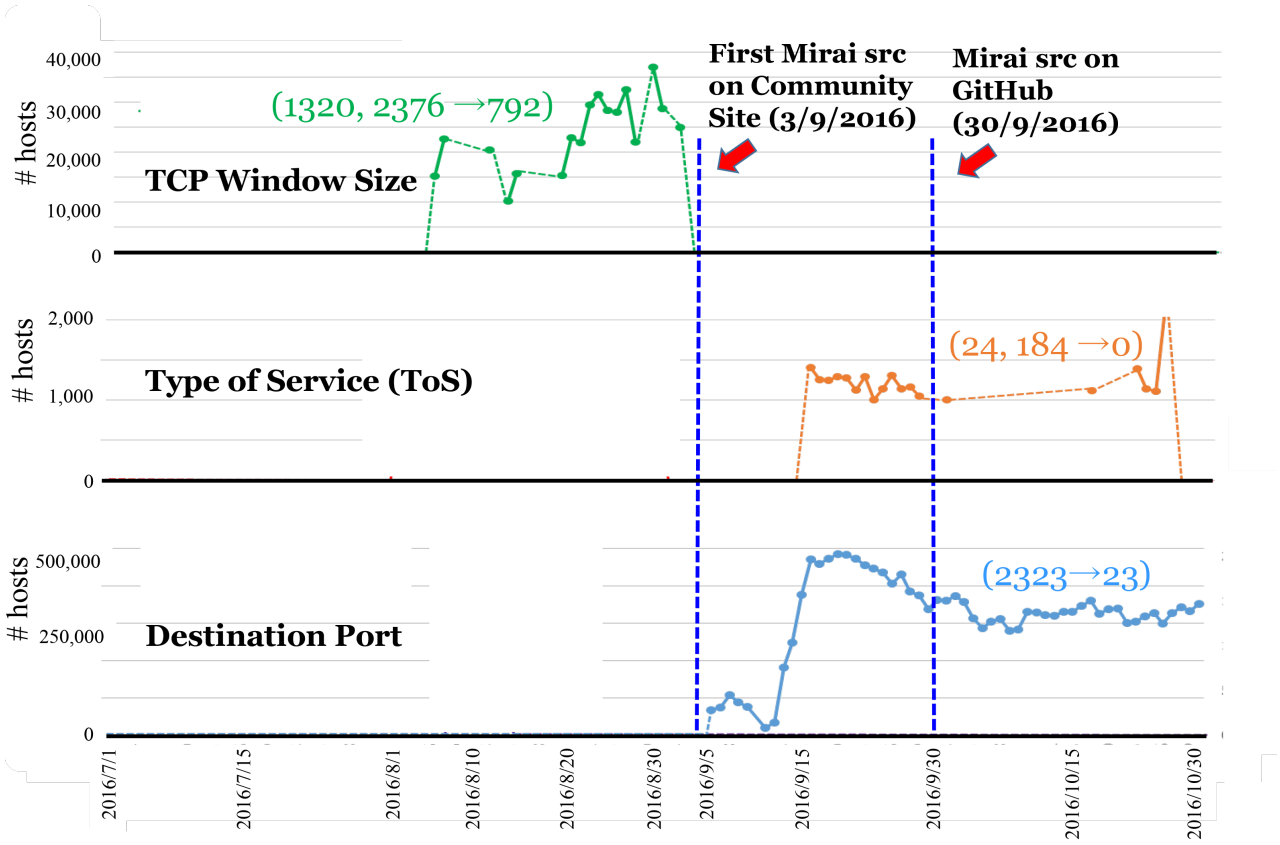

Fig. 3 Transitions in the number of hosts sending darknet SYN packets matched with the association rules for destination ports, Type of Service (ToS), and TCP window sizes extracted from the packets received with the NICT /16 darknet sensor between July 1st, 2016 and October 30th, 2016.

From the time lines of association rules, we can understand the competing trends on the emergence and disappearance of IoT malwares: Mirai vs Hajime.

5 Conclusions

In this paper, we developed a new darknet analysis using the association rule learning. In the proposed method, not only destination ports but also other TCP/IP header information are used to create transaction sets for the association rule learning. Then, the rule mining for all header information is conducted in parallel to obtain association rules.

The proposed darknet analysis was applied to a large set of TCP SYN packets collected from July 1st, 2016 to July 31st, 2018 with the NICT /16 darknet sensor. As a result, an association rule on TCP window size appeared on 2nd August and disappeared three days before the source code of Mirai was released. Almost all hosts whose scan activities are featured by the obtained association rules have a known Mirai feature: sequence number = destination IP. Therefore, we conjecture that the attackers were doing test or preparation for the actual distribution of Mirai malware about one month before the source code was opened. We also study on the trends of IoT malware variants of Mirai and Hajime after the Mirai outbreak. We found that 19 association rules on destination ports and type of service emerged in total for Mirai and Hajime, and we can understand how Mirai and Hajime competed each other from the time lines of generated/disappeared rules. The result of this paper is very encouraging for us to apply the proposed method for future attack detection.

There remain several open problems to be solved in future. The number of extracted association rules could be varied depending on the threshold values for support and confidence. If these thresholds become smaller, more association rules could be discovered. Some of them are expected to be useful, while meaningless rules might also increase in general. Therefore, a sophisticated way to find more useful rules should be further explored. In this work, we use only one /16 darknet sensor to observe attacks. However, if more darknet sensors are available, it is expected that we could discover more useful rules. This is also left as our future work.

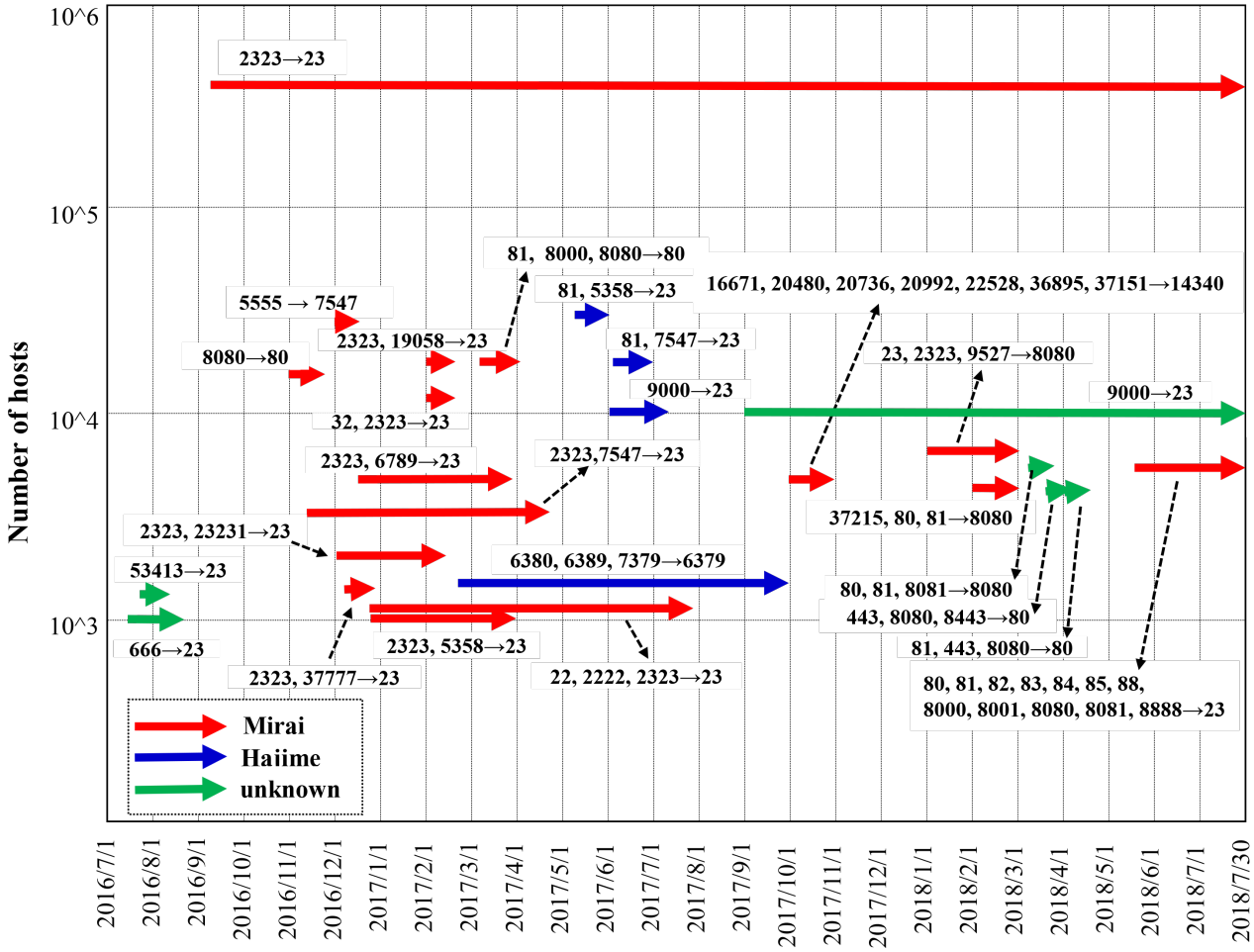

Fig. 4 Time lines of association rules extracted from July 1st, 2016 to July 31st, 2018.

Compliance with Ethical Standards: This research was funded by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (B) 16H02874 and the Commissioned Research of National Institute of Information and Communications Technology (NICT), JAPAN.

Conflict of Interest: Seiichi Ozawa has received research grants from Daiwa SB Investments ltd., LAPIS Semiconductor Co., Ltd., Mitsubishi Heavy Industries, Ltd., and Fujitsu Laboratories, Ltd. Tao Ban declares that he has no conflict of interest. Naoki Hashimoto declares that he has no conflict of interest. Junji Nakazato declares that he has no conflict of interest. Jumpei Shimamura declares that he has no conflict of interest.

Ethical approval: This article does not contain any studies with human participants performed by any of the authors.

References

1. T. Ban, M. Eto, S. Guo, D. Inoue, K. Nakao, R. Huang, "A study on association rule mining of darknet big data," *Proc. of Int. Joint Conference on Neural Networks*, pp. 1-7, 2015.
2. 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," Proc. of 2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress, pp. 341-349, 2016.
3. C. Stocker, J. Horchert, "Mapping the internet: A hacker's secret internet census," *Spiegel Online*, March 22, 2013.
4. E.L. Malecot, D. Inoue, "The Carna botnet through the lens of a network telescope", In: J. Danger, et al.(eds), *Foundations and Practice of Security*, LNCS, vol. 8352,

Springer, pp. 426-441, 2014.
5. R. Agrawal, T. Imielinski, A. Swami, “Mining association rules between sets of items in large databases,” ACM SIGMOD Record, vol. 22, no. 2, pp. 207-216, 1993.
6. J. Han, J. Pei, Y. Yin, “Mining frequent patterns without candidate generation,” ACM SIGMOD Record, vol. 29, no. 2, pp. 1-12, 2000.
7. J. HanJian, P.Y. Mao, “Mining frequent patterns without candidate generation: A frequent-pattern tree approach,” *Data Mining and Knowledge Discovery*, vol. 8, no. 1, pp. 53?87, 2004.
8. C. Borgelt, “Frequent item set mining,” *Data Mining Knowledge Discovery*, vol. 2, no. 6, pp. 437-456, 2012.
9. https://nvd.nist.gov/
10. K.Nichols, S. Blake, F. Baker, D. Black, “Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers,” IETF RFC 2119, December 1998.
11. D. Grossman, “New Terminology and Clarifications for Diffserv,” IETF RFC 3260, April 2002.
12. J. Babiarz, K. Chan, F. Baker, “ Configuration Guidelines for DiffServ Service Classes,” IETF RFC 4594, August 2006.
13. *Introduction to Cisco IOS NetFlow - A Technical Overview*, White Papers, Cisco, updated May, 2012.
14. V.L. Thing, M. Sloman, N. Dulay, “A Survey of Bots Used for Distributed Denial of Service Attacks,” *New Approaches for Security, Privacy and Trust in Complex Environments*, Springer US, Boston, MA, pp. 229-240, 2007
15. V. Jacobson, R. Braden, D. Borman, “TCP Extensions for High Performance,” IETF RFC 1323, May 1992.
16. ”Microsoft Windows TCP/IP Connection Exhaustion Denial of Service Vulnerability,” Cisco Mulitivendor Vulnerability Alerts, Alert ID: 18959, CVE-2009-1926, September, 2009.
17. M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K, Thomas, Y. Zhou, “Understanding the Mirai Botnet,” *Proc. of 26th USENIX Security Symposium*, pp. 1093-1110, 2017.
18. http://data.netlab.360.com/mirai-scanner/

Table 2 Obtained association rules for (a) destination ports, (b) Type of Service (ToS), and (c) TCP window sizes. The columns 'Country' and 'Period' correspond to the major countries of hosts and the periods that an association rule emerged. 'Support' is equivalent to the number of hosts matched with an association rule and 'Confidence' is the conditional probability of an association rule holds. The column 'Type' represents the type of malwares that over 90% hosts get infected by either of Mirai (M), Hajime (H), or unknown (U). The abbreviation of each country is followed by the IOC code: Philippine (PHI), Great Britain (GBR), China (CHN), Mexico (MEX), Brazil (BRA), Korea (KOR), Thai (THA), Vietnam (VIE), Pakistan (PAK), Chinese Taipei (TPE), India (IND)

(a) Destination Ports

Association Rules	Country	Period	Support	Conf. [%]	Type
666 → 23	PHI	7/26 - 8/23/2016	2,176	93.8	U
53413 → 23	-	8/1/2016	2,343	94.7	U
2323 → 23	-	9/6/2016 -	$> 3.7 \times 10^6$	91.7	M
8080 → 80	-	11/2 - 11/21/2016	29,006	94.3	M
(2323, 7547) → 23	-	11/26 - 4/27/2017	42,620	96.5	M
5555 → 7547	GBR	12/4/2016	29,425	92.3	M
(2323, 23231) → 23	-	12/10/2016 - 2/14/2017	70,376	98.8	M
(2323, 37777) → 23	-	12/11 - 12/28/2016	7,319	96.8	M
(2323, 6789) → 23	-	12/18/2016 - 3/31/2017	45,628	94.8	M
(23, 2222, 2323) → 22	-	12/23 - 7/30/2017	160,444	99.1	M
(2323, 5358) → 23	-	1/26 - 3/28/2017	71,931	98.1	M
(2323, 19058) → 23	-	2/3 - 2/9/2017	67,036	98.6	M
(32, 2323) → 23	-	2/7 - 2/9/2017	27,699	98.9	M
(6380, 6389, 7379) → 6379	CHN	2/25 - 9/29/2017	8287	99.7	H
(81, 88, 8000, 8080) → 80	-	3/7 - 3/29/2017	107,901	96.8	M
(81, 5358) → 23	MEX, BRA	5/17 - 5/27/2017	100,706	98.9	H
(81, 7547) → 23	-	6/2 - 7/7/2017	381,981	91.1	H
9000 → 23	-	6/2 - 7/8/2017	82,888	94.4	H
9000 → 23	-	9/2/2017 -	166,745	94.6	U
(14340, 20480, 20736, 20992, 22528, 36895, 37151) → 16671	-	10/1 - 10/29/2017	24,425	100	M
(23, 2323, 9527) → 8080	KOR	1/10 - 3/8/2018	39,170	99.6	M
(80, 81, 37215) → 8080	-	1/29 - 2/22/2018	16,140	100	M
(80, 81, 8081) → 8080	CHN	3/30/2018	6,177	93.7	U
(443, 8080, 8443) → 80	CHN	4/13/2018	5,317	92.7	U
(81, 443, 8080) → 80	CHN	4/26/2018	4,811	92.6	U
(80, 81, 82, 83, 84, 85, 88, 2323, 8000, 8001, 8080, 8081) → 8888	-	5/19 - 7/31/2018	31,361	99.4	M

(b) Type of Service (ToS)

Association Rules	Country	Period	Support	Conf. [%]	Type
(208, 212, 216) → 204	THA	8/2-9/1/2016	11,167	95.3	U
(24, 184) → 0	VIE	9/7-10/2/2016	16,302	99.2	M
(8, 104) → 40	PAK	2/27-3/7/2017	11,739	99.1	H
224 → 0	TPE	2/28-3/31/2017	4,642	95.5	H
(0, 40) → 8	IND	4/16-4/27/2017	24,257	91.0	M
(104, 184) → 0	CHN	8/31/2017 -	1,054	95.0	M
(4, 184) → 0	CHN	9/1/2017 -	1,323	93.8	M
(4, 184) → 0	CHN	7/11-7/23/2018	2,020	99.5	U
44 → 0	CHN	7/11-7/23/2018	4,020	99.5	U

(c) TCP Window Sizes

Association Rules	Country	Period	Support	Conf. [%]	Type
(1320,2376) → 792	-	8/2-9/4/2016	27,188	94.6	M
14100 → 1024	CHN	1/5, 30, 2/2/2018	12,595	92.1	U