Frequent Sequential Pattern Discovery for Data Screening

Hisashi Tsuruta, Takayoshi Shoudai, and Jun'ichi Takeuchi

***Abstract*—To early detect and defend the threats in the Internet caused by botnet, darknet monitoring is very important to understand various botnet activities. However, common illegal accesses by ordinary malwares makes such detection difficult. In this paper, in order to remove such accesses by ordinary malwares from the results of network monitoring, we propose a data screening method based on finding frequent sequential patterns which appear in given traffic data. Besides, we apply our method to traffic data observed in darknet and report the results.**

***Index Terms*—incident detection, frequent pattern mining, sequential pattern, data screening, darknet monitoring.**

I. Introduction

THE rapid growth of the high-speed Internet access service and mass storage media brings us not only the benefits of society but also many harmful effects. A notorious example of them is the social damages caused by various computer viruses. The bot worm is a typical example of criminal computer viruses, which is an Internet software controlled by a bot herder. Its infection spreads in a computer network. A bot herder can control the infected computers as a network, which is called botnet, and cause many incidents such as DDos attacks, sending a huge spam mails and so on. To construct efficient countermeasures against these incidents, many researches are studying effective methods for early detection of tendencies of such incidents [1], [2], [3].

Our purpose is knowledge discovery to detect signs of incidents. If we could prevent them, the safety and the confidence of the Internet would be increased. Many researchers extract tendencies of particular senders or patterns and analyze them to detect evidence of new attacks. For example, Kim et al. [6] proposed a Flow-based method for abnormally detectors and Fukushima et al. [5] proposed a method which focuses on the average number of packets sent by a source address and its frequency of appearances to find the subtle attacks.

On the other hand, when it comes to anomaly detection, illegal packets caused by the well-known malwares make it harder. Then, we focus on finding attack patterns of the well-known malwares. Majority of them could often be detected easily by ignorant people about network incidents. Therefore we define them as a class of time-span sequence patterns which is easily detected by computers. Furthermore we introduce a method to discover a set of frequent patterns which appear in the darknet observation data and delete packets caused by them. Many researchers are interested in frequent pattern discovery from data having structures such as web data [7], chemical compounds [8], and so on.

In this paper, we utilize a data observed in the darknet, which is a network that cannot be accessed through conventional means. Most of all packets which the darknet receives are illegal, so they could be considered as traces of malicious attacks. Therefore we might detect attack patterns by malwares in the darknet access record.

We think of each packet as a three tuple (transmitter's address, transmitter's port, receiver's port), and call it an *event*. Moreover We call an event with its received time an *incident*. The observation data, called an *incident database*, is a set of incidents. We regard each item of an event as a string and propose a pattern class of sequences of string patterns with time delays. The following pattern is an example of incident patterns, called *event pattern delay sequences (EPD sequences)* in this paper.

$$\begin{aligned} & (??.???.*.*, *, 445) \\ \xrightarrow{0.50} \quad & (??.*.*.*, ?345, ?????) \\ \xrightarrow{0.50} \quad & (??.*.*.*, ?345, ?????) \end{aligned}$$

In the above example, '?' stands for any one constant symbol, and '$*$' stands for any string whose length is at least 1. Each of the three items in a round bracket respectively denotes transmitter's address, transmitter's port number and receiver's port number. The real number over a right arrow means the maximum time delay between the first pattern and the second pattern (or the second pattern and the third pattern)[1].

In this paper, firstly we introduce a class of EPD sequences in order to represent common illegal packets, and formally discuss a computational problem of finding EPD sequences in a given incident database. Next we give an effective heuristic algorithm to discover EPD sequences in an incident database. Lastly we propose an automatic data screening method and report experimental results on darknet traffic data.

II. Preliminaries

Let X be a set of distinct *events*. For an event $e \in X$, let t be a time when the event e occurs. We call a pair (t, e) an *incident*, and a set of incidents an *incident database*.

Definition 1 (ED sequences): Let r be a positive integer and $T_{\max}$ a positive real number. Let $a_1, a_2, \ldots, a_r$ be r events (not necessarily distinct). And let $\tau_1, \tau_2, \ldots, \tau_{r-1}$ be $r-1$ positive real numbers which are less than or equal to $T_{\max}$. Then we call $\pi = (a_1, \tau_1, a_2, \tau_2, \ldots, \tau_{r-1}, a_r)$ an

Manuscript received January 7, 2011; revised February 7, 2011.

H. Tsuruta, T. Shoudai, J. Takeuchi are with Department of Informatics, Kyushu University, 744 Motooka, Nishi-ku, Fukuoka 819-0395, Japan, and with Institute of Systems, Information Technologies and Nanotechnologies (ISIT), 2-1-22 Momochihama, Sawara-ku, Fukuoka 814-0001, Japan, e-mail: {hisashi.tsuruta, shoudai, tak}@inf.kyushu-u.ac.jp

[1]This pattern is discovered during a period of receiving first 10,000 packets in the darknet observation data of 23 January, 2009. This pattern's cover rate in the same range is about 10.18%.

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

$(r, T_{\max})$-*event delay sequence* (abbreviated to $(r, T_{\max})$-*ED sequence*).

Definition 2 (Matching of ED sequences): Let D be an incident database and $\pi = (a_1, \tau_1, a_2, \tau_2, \ldots, \tau_{r-1}, a_r)$ an $(r, T_{\max})$-ED sequence. For a subset D' of D with $|D'| = r$, we say that π *matches* D' if the following conditions hold: Let $((t_1, e_1), (t_2, e_2), \ldots, (t_r, e_r))$ be a sorted sequence of the incidents in D' with respect to t_i $(1 \leq i \leq r)$, i.e., $t_1 \leq t_2 \leq \cdots \leq t_r$. Then,

(a) for all i $(1 \leq i \leq r)$, $e_i = a_i$, and

(b) for all i $(1 \leq i \leq r-1)$, $t_{i+1} - t_i \leq \tau_i$.

Definition 3 (Cover rate of ED sequences): For an incident database D and an $(r, T_{\max})$-ED sequence π, we denote by $D(\pi)$ the union of all subsets of D which are matched by π, i.e.,

$$D(\pi) = \bigcup_{D' \subseteq D \text{ s.t. } \pi \text{ matches } D'} D'.$$

The *cover rate* of π for D is defined as $cover_D(\pi) = \dfrac{|D(\pi)|}{|D|}$. Let P be a set of $(r, T_{\max})$-ED sequence and $D(P) = \bigcup_{\pi \in P} D(\pi)$. The *cover rate* of P for D is defined as $cover_D(P) = \dfrac{|D(P)|}{|D|}$.

First of all, we consider the following computational problem, which plays an important role in this paper.

R-EVENT DELAY SEQUENCE COVER (R-EC)

INSTANCE: An incident database D, a cover rate σ $(0 \leq \sigma \leq 1)$, a maximum time delay $T_{\max}$, and a positive integer K.

QUESTION: Is there a set P of $(R, T_{\max})$-ED sequences such that $|P| \leq K$ and the cover rate of P for D is at least σ?

We show the following theorem.

Theorem 1: 3-EC is NP-complete.

Proof: It is easy to see that 3-EC is in NP. We construct a reduction from the following well-known NP-complete problem X3C.

EXACT COVER BY 3-SETS (X3C)

INSTANCE: A set X with $|X| = 3q$ and a collection C of 3-element subsets of X.

QUESTION: Does C contain an exact cover for X, i.e., a subcollection $C' \subseteq C$ such that every element of X occurs in exactly one member of C'?

Let $X = \{e_1, e_2, \ldots, e_n\}$ where $n = 3q$ and $C = \{c_1, c_2, \ldots, c_m\}$ where $c_i = \{a_i^{(1)}, a_i^{(2)}, a_i^{(3)}\}$ $(1 \leq i \leq m)$. We construct an incident database D as follows. Let $D' = \{((i-1)(n+2) + j, a_i^{(j)}) \mid 1 \leq i \leq m \text{ and } j = 1, 2, 3\}$ and $D'' = \{(m(n+2) + k, e_k) \mid 1 \leq k \leq n\}$. Let $D = D' \cup D''$, $\sigma = (3q+n)/(3m+n)$, $T_{\max} = n - 1$, and $K = q$.

First, we suppose that X3C returns *true*. Then we have an exact cover $C' = \{c_{i_1}, c_{i_2} \ldots, c_{i_q}\}$ $(1 \leq i_1 < i_2 < \cdots < i_q \leq m)$. Let $P = \{(a_{i_\ell}^{(1)}, T_{\max}, a_{i_\ell}^{(2)}, T_{\max}, a_{i_\ell}^{(3)}) \mid 1 \leq \ell \leq q\}$. It is easy to see that $|\bigcup_{\pi \in P} D'(\pi)| = 3q$ and $|\bigcup_{\pi \in P} D''(\pi)| = n$. Then the cover rate of P is equal to $\sigma = (3q+n)/(3m+n)$. Therefore 3-EC returns *true*.

Conversely, we suppose that 3-EC returns *true*. Then there is a set P of $(3, T_{\max})$-ED sequences such that $|P| \leq q$. Let $P = \{(\alpha_\ell^{(1)}, \tau_\ell, \alpha_\ell^{(2)}, \tau'_\ell, \alpha_\ell^{(3)}) \mid 1 \leq \ell \leq q\}$ where $\alpha_\ell^{(1)}, \alpha_\ell^{(2)}, \alpha_\ell^{(3)} \in \{e_1, e_2, \ldots, e_n\}$ and $\tau_\ell, \tau'_\ell \leq T_{\max}$. Since $|\bigcup_{\pi \in P} D(\pi)| \geq 3q + n$ and $|D''| = n$, $|\bigcup_{\pi \in P} D'(\pi)| \geq 3q$. Therefore, for each $(3, T_{\max})$-ED sequence $(\alpha_\ell^{(1)}, \tau_\ell, \alpha_\ell^{(2)}, \tau'_\ell, \alpha_\ell^{(3)})$, there is an index $f(\ell)$ $(1 \leq f(\ell) \leq m)$ such that 3 continuous time events $\{((f(\ell)-1)(n+2)+1, a_{f(\ell)}^{(1)}), ((f(\ell)-1)(n+2)+2, a_{f(\ell)}^{(2)}), ((f(\ell)-1)(n+2)+3, a_{f(\ell)}^{(3)})\}$ are matched by $(\alpha_i^{(1)}, \tau_i, \alpha_i^{(2)}, \tau'_i, \alpha_i^{(3)})$. It is easy to see that $C' = \{\{\alpha_{f(\ell)}^{(1)}, \alpha_{f(\ell)}^{(2)}, \alpha_{f(\ell)}^{(3)}\} \mid 1 \leq \ell \leq q\}$ is an exact cover for X. ■

In the next section, we give an effective heuristic algorithm to compute one of the more generalized problems within the framework of 3-EC.

III. EVENT PATTERN DELAY SEQUENCE COVER

Let N and $k_1, k_2, \ldots, k_N$ be positive integers. Let '?' and '$*$' be two special symbols. Let Σ be a finite alphabet which include neither '?' nor '$*$' (i.e., $\Sigma \cap \{\text{'?'}, \text{'}*\text{'}\} = \emptyset$). Below, an *event* is an object which consists of N strings $w_1, w_2, \ldots, w_N$ in Σ^*, each of whose length is at most k_i $(1 \leq i \leq N)$. Then we denote an event by $e = (w_1, w_2, \ldots, w_N) \in \Sigma^{k_1} \times \Sigma^{k_2} \times \cdots \times \Sigma^{k_N}$.

An *atom pattern* is a string in $\omega \in (\Sigma \cup \{\text{'?'}\})^+ \cup \{\text{'}*\text{'}\}$. For any atom pattern ω, we denote by $|\omega|$ the length of ω. For an atom pattern ω, if $\omega =$ '$*$', we can replace '$*$' with an atom pattern $\omega' \in (\Sigma \cup \{\text{'?'}\})^+$. If ω includes '?', we can replace it with a symbol in Σ. We call a set of such replacements a *substitution*. Let θ be a substitution. We denote by $\omega\theta$ the atom pattern which is obtained from ω by applying all replacements in θ to ω.

We denote by $\mathcal{P}$ the set of atom patterns. For an integer k, we denote by $\mathcal{P}^{[k]}$ the set of atom patterns in $\mathcal{P}$ of length at most k. For two atom patterns ω, ω', we write $\omega' \preceq \omega$ if there is a substitution θ such that $\omega' = \omega\theta$.

Definition 4 (Event patterns): We call a sequence of atom patterns $p = (\omega_1, \omega_2, \ldots, \omega_N) \in \mathcal{P}^{[k_1]} \times \mathcal{P}^{[k_2]} \times \cdots \times \mathcal{P}^{[k_N]}$ an event pattern. For an event $e = (w_1, w_2, \ldots, w_N)$ and an event pattern $p = (\omega_1, \omega_2, \ldots, \omega_N)$, we write $e \preceq p$ if for all i $(1 \leq i \leq N)$, $w_i \preceq \omega_i$.

In a similar way to Def. 1, we define an *event pattern delay sequence*.

Definition 5 (EPD sequences): Let r be a positive integer and $T_{\max}$ a positive real number. Let $p_1, p_2, \ldots, p_r$ be r event patterns. And let $\tau_1, \tau_2, \ldots, \tau_{r-1}$ be $r-1$ positive real numbers which are less than or equal to $T_{\max}$. Then we call $\pi = (p_1, \tau_1, p_2, \tau_2, \ldots, \tau_{r-1}, p_r)$ an $(r, T_{\max})$-*event pattern delay sequence* (*abbreviated to* $(r, T_{\max})$-*EPD sequence*).

Definition 6 (Matching of EPD sequences): Let D be an incident database and $\pi = (p_1, \tau_1, p_2, \tau_2, \ldots, \tau_{r-1}, p_r)$ an $(r, T_{\max})$-EPD sequence. For a subset D' of D with $|D'| = r$, In similar way to Def. 2, we say that π *matches* D' if the following conditions hold: Let $((t_1, e_1), (t_2, e_2), \ldots, (t_r, e_r))$ be a sorted sequence of the incidents in D' with respect to t_i $(1 \leq i \leq r)$, i.e., $t_1 \leq t_2 \leq \cdots \leq t_r$. Then,

(a) for all i $(1 \leq i \leq r)$, $e_i \preceq p_i$, and

(b) for all i $(1 \leq i \leq r-1)$, $t_{i+1} - t_i \leq \tau_i$.

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Let D be an incident database. In a similar way to Def. 3, we define the cover rate of an EPD sequence π for D and the cover rate of a set P of EPD sequences for D.

In order to consider a similar computational problem to 3-EC, we have to define an ordering on EPD sequences. The most generalized EPD sequence is $\pi_0 = (p_0, T_{\max}, p_0, \ldots, T_{\max}, p_0)$, where $p_0 = (\underbrace{*, *, \ldots, *}_{N})$. All incidents in any incident database can be covered by π_0 but it is meaningless.

Definition 7 (Size of delay sequences): For an event pattern $p \in \mathcal{P}^{[k_1]} \times \mathcal{P}^{[k_2]} \times \cdots \times \mathcal{P}^{[k_N]}$, we denote by $n_\Sigma(p)$ the number of symbols in Σ which appear in p, and by $n_{?'}(p)$ the number of '?' which appear in p. We define the size of an event pattern p as

$$size(p) = n_\Sigma(p) \times (Q+1) + n_{?'}(p),$$

where $Q = \sum_{i=1}^{N} k_i$.

Let $\pi = (p_1, \tau_1, p_2, \tau_2, \ldots, \tau_{r-1}, p_r)$ be an $(r, T_{\max})$-EPD sequence. We define the size of π as follows:

$$size(\pi) = \sum_{i=1}^{r} size(p_i) + \sum_{i=1}^{r-1} (T_{\max} - \tau_i).$$

For any $(r, T_{\max})$-EPD sequence π, $size(\pi) \geq 0$. And $size(\pi_0) = 0$ for $\pi_0 = (p_0, T_{\max}, p_0, \ldots, T_{\max}, p_0)$, where $p_0 = (*, *, \ldots, *)$.

In a similar way to the definition of R-EC, we define R-EVENT PATTERN DELAY SEQUENCE COVER (R-EPC) as follows:

R-EVENT PATTERN DELAY SEQUENCE COVER (R-EPC)

INSTANCE: An incident database D and the following four parameters:
(a) K: a maximum number of EPD sequences,
(b) σ: a minimum cover rate $(0 \leq \sigma \leq 1)$,
(c) S: a minimum size of event patterns,
(d) $T_{\max}$: a maximum time delay.

QUESTION: Is there a set P of $(R, T_{\max})$-EPD sequences that satisfy the following conditions: for all $\pi = (p_1, \tau_1, \ldots, \tau_{R-1}, p_R) \in P$,
(a) $|P| \leq K$,
(b) $cover_D(P) \geq \sigma$,
(c) $size(p_i) \geq S$ $(1 \leq i \leq R)$, and
(d) $\tau_i \leq T_{\max}$ $(1 \leq i \leq R-1)$.

We can easily see the following theorem from Theorem 1.

Theorem 2: 3-EPC is NP-complete.

IV. HEURISTIC ALGORITHMS FOR R-EPC

A. An Apriori-like method

In this section, we give a heuristic algorithm for computing R-EPC by using an Apriori-like method twice.

Let D be an incident database. We denote by D_i the set of all the i-th strings w_i of the incidents $(t, (w_1, w_2, \ldots, w_N))$ in D. For any i $(1 \leq i \leq N)$ and atom pattern $\omega \in \mathcal{P}^{[k_i]}$, let $freq_{D_i}(\omega) = \frac{|\{w_i \in D_i \mid w_i \preceq \omega\}|}{|D_i|}$. For any event pattern $p \in \mathcal{P}^{[k_1]} \times \mathcal{P}^{[k_2]} \times \cdots \times \mathcal{P}^{[k_N]}$, let

$$freq_D(p) = \frac{|\{e \mid \exists t \text{ s.t. } (t, e) \in D \text{ and } e \preceq p\}|}{|D|}.$$

Algorithm FIND_EPD_SEQUENCES (FES);
Input: $D = \{(t, e) \mid t > 0$ and $e \in \Sigma^{k_1} \times \Sigma^{k_2} \times \cdots \times \Sigma^{k_N}.\}$: an incident database, σ: a minimum cover rate, S: a minimum size of $(R, T_{\max})$-EPD sequences;

Let δ be a positive real number smaller than σ. This parameter δ plays an important role in this algorithm to produce a good set of $(R, T_{\max})$-EPD sequences.

1) Let $D_i^a := \{w_i \mid (t, (w_1, \ldots, w_i, \ldots, w_N)) \in D\}$ $(1 \leq i \leq N)$; For all i $(1 \leq i \leq N)$, we compute the sets A_i:
$$A_i := \{\omega \in \mathcal{P}^{[k_i]} \mid freq_{D_i^a}(\omega) \geq \delta\};$$
2) We compute the set of event patterns F (Procedure FREQ_EVENT_PATTERNS (Fig. 1)):
$$F = \{p \in \mathcal{P}^{[k_1]} \times \mathcal{P}^{[k_2]} \times \cdots \times \mathcal{P}^{[k_N]} \mid freq_D(p) \geq \delta\};$$
3) We compute the set of $(R, T_{\max})$-EPD sequences P (Procedure FREQ_PATTERN_SEQUENCES (Fig. 2)):
$$P = \{\pi = (p_1, T_{\max}, \ldots, T_{\max}, p_R) \mid freq_D(\pi) \geq \delta\};$$
4) For each $(R, T_{\max})$-EPD sequence $\pi = (p_1, \tau_1, \ldots, \tau_{R-1}, p_R)$ in P, we try to decrease each time delay τ_ℓ $(1 \leq \ell \leq R-1)$ as much as possible (Procedure UPDATE_TIME_DELAYS (Fig. 3)).
5) Output P.

B. A machine learning method

Here we define a partial order on EPD sequences and a concept of maximal EPD sequences on a given incident database. And then we give an idea of an algorithm for solving our problem.

Let $\pi = (p_1, \tau_1, p_2, \tau_2, \ldots, \tau_{r-1}, p_r)$ be an $(r, T_{\max})$-EPD sequence and $\pi' = (p'_1, \tau'_1, p'_2, \tau'_2, \ldots, \tau'_{r'-1}, p'_{r'})$ an $(r', T_{\max})$-EPD sequence. We write $\pi' \preceq \pi$ if $r' \geq r$ and there are r integers $j_1, j_2, \ldots, j_r$ with $1 \leq j_1 < j_2 < \cdots < j_r \leq r'$ such that for all i $(1 \leq i \leq r)$,

$$p'_{j_i} \preceq p_i \text{ and } \sum_{\ell=j_i}^{j_{i+1}-1} t'_\ell \leq t_i.$$

Definition 8 (Maximal EPD sequences): Let D be an incident database and σ a minimum cover rate $(0 \leq \sigma \leq 1)$. We say that π is a *maximal $(r, T_{\max})$-EPD sequence* with respect to D and σ if there is no $(r, T_{\max})$-EPD sequence π' $(\pi' \neq \pi)$ such that $\pi' \preceq \pi$ and $cover_D(\pi') \geq \sigma$.

Let P be a set of $(r, T_{\max})$-EPD sequence. we say that P is a *maximal K-set of $(r, T_{\max})$-EPD sequences with respect to D and σ* if the following four conditions hold:

1) $|P| \leq K$,
2) $cover_D(P) \geq \sigma$,
3) for any pair $\pi, \pi' \in P$ $(\pi \neq \pi')$, neither $\pi \preceq \pi'$ nor $\pi' \preceq \pi$ hold, and

Procedure FREQ_EVENT_PATTERNS$(D, \delta, N, \{A_i\})$;
Input: D: an incident database, δ: a real number, N: an integer, $\{A_i\}$: a collection of sets of atom patterns;
begin
  $F_1 := \bigcup_{1\leq i\leq N} \bigcup_{\omega\in A_i} \{p = (\omega_1, \ldots, \omega_\ell, \ldots, \omega_N) \mid$ if $\ell = i$ then $\omega_\ell = \omega$ else $\omega_\ell =$ '$*$'$\}$;
  for $k := 2$ **to** N **do begin**
    $F_k := \emptyset$;
    foreach $p, p' \in F_{k-1}$ **do begin**
      Let $p = (\omega_1, \omega_2, \ldots, \omega_N)$ and $p' = (\omega'_1, \omega'_2, \ldots, \omega'_N)$;
      if there are two indices i and j $(i < j)$ which satisfy the following conditions:
        1. $\omega_\ell = \omega'_\ell$ $(1 \leq \ell \leq N, \ell \neq i,$ and $\ell \neq j)$,
        2. $\omega_i \neq$ '$*$' and $\omega_\ell =$ '$*$' $(i+1 \leq \ell \leq N)$, and
        3. $\omega'_j \neq$ '$*$' and $\omega'_\ell =$ '$*$' $(i \leq \ell \leq j-1,\ j+1 \leq \ell \leq N)$
      then begin
        Let $p'' = (\omega''_1, \ldots, \omega''_\ell, \ldots, \omega''_N)$ be the event pattern s.t. if $\ell = j$ then $\omega''_\ell = \omega'_\ell$ else $\omega''_\ell = \omega_\ell$;
        if $freq_D(p'') \geq \delta$ **then** $F_k := F_k \cup \{p''\}$
      end
    end
  end;
  return $F := \bigcup_{1\leq k\leq N} F_k$
end;

Fig. 1. Procedure FREQ_EVENT_PATTERNS which is used at Step 2 in Algorithm FES.

Procedure FREQ_PATTERN_SEQUENCES(F, D, δ, R);
Input; F: a set of event patterns, D: an incident database, δ: a real number, R: an integer;
begin
  $P_1 := F$;
  for $r := 2$ **to** R **do begin**
    $P_r := \emptyset$;
    foreach $\pi, \pi' \in P_{r-1}$ **do begin**
      Let $\pi = (p_1, T_{\max}, \ldots, T_{\max}, p_{r-1})$ and $\pi' = (p'_1, T_{\max}, \ldots, T_{\max}, p'_{r-1})$;
      if π and π' satisfy the following conditions: for all i $(1 \leq i \leq r-2)$, $p_i = p'_i$, and $p_{r-1} \neq p'_{r-1}$
      then begin
        Let $\pi'' := (p_1, T_{\max}, \ldots, T_{\max}, p_{r-1}, T_{\max}, p'_{r-1})$;
        if $cover_D(\pi'') \geq \delta$ **then** $P_r := P_r \cup \{\pi''\}$
      end
    end
  end;
  return P_R
end;

Fig. 2. Procedure FREQ_PATTERN_SEQUENCES which is used at Step 3 in Algorithm FES.

4) for any $\pi \in P$, there is no $(r, T_{\max})$-EPD sequence $\pi' \notin P$ such that $\pi' \preceq \pi$ and $cover_D(P - \{\pi\} \cup \{\pi'\}) \geq \sigma$.

We define R-MAXIMAL EVENT PATTERN DELAY SEQUENCE COVER (R-MEPC) as follows:

R-MAXIMAL EVENT PATTERN DELAY SEQUENCE COVER (R-MEPC)
INSTANCE: An incident database D and the following four parameters:
K: a maximum number of EPD sequences, σ: a minimum cover rate $(0 \leq \sigma \leq 1)$, $T_{\max}$: a maximum time delay.
QUESTION: Is there a maximal K-set P of $(R, T_{\max})$-EPD sequences that satisfy the following conditions: (i) $cover_D(P) \geq \sigma$ and (ii) for all $\pi = (p_1, \tau_1, \ldots, \tau_{R-1}, p_R) \in P$, $\tau_i \leq T_{\max}$.

Proposition 1: Let π and π' be $(r, T_{\max})$-EPD sequence and $(r', T_{\max})$-EPD sequence, respectively. Then, if $\pi' \preceq \pi$ then $size(\pi') \geq size(\pi)$.

Proof: First, we show that $p' \preceq p \Rightarrow size(p') \geq size(p)$. By the definition of the size of an event pattern, we have

$$\begin{aligned} & size(p') - size(p) \\ = \ & (Q+1)(n_\Sigma(p') - n_\Sigma(p)) + (n_{\cdot?\cdot}(p') - n_{\cdot?\cdot}(p)). \end{aligned}$$

From the definition of '$\preceq$', symbols in Σ which appears in p must appear in p'. Therefore we have $n_\Sigma(p) \leq n_\Sigma(p')$ anytime. Since the absolute value of the first term is larger than or equal to that of the second term, $size(p') - size(p) \geq 0$ is satisfied even if the absolute value of the second term is negative. Therefore, we have $p' \preceq p \Rightarrow size(p') \geq size(p)$.

Let $\pi = (p_1, \tau_1, p_2, \tau_2, \ldots, \tau_{r-1}, p_r)$ and $\pi' = (p'_1, \tau'_1, p'_2, \tau'_2, \ldots, \tau'_{r'-1}, p'_{r'})$. From $\pi' \prec \pi$, the following

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Procedure UPDATE_TIME_DELAYS(P, D, σ, S);
Input: P: a set of EPD sequences, D: an incident database, σ: a real number, S: an integer;
begin
  foreach $\pi = (p_1, T_{\max}, \ldots, T_{\max}, p_R) \in P$ **do begin**
    $\pi' := \pi$; $P := P - \{\pi\}$;
    $T(\pi) := \{(t_1, t_2, \ldots, t_R) \mid$ there is a subset $D' = \{(t_1, e_1), (t_2, e_2), \ldots, (t_R, e_R)\} \subseteq D$ s.t. π matches D' where $t_1 \leq t_2 \leq \cdots \leq t_R\}$;
    for $\ell := 1$ **to** $R - 1$ **do begin**
      Let $T_\ell(\pi) = \{t_{\ell+1} - t_\ell \mid (t_1, \ldots, t_\ell, t_{\ell+1}, \ldots, t_R) \in T(\pi)\}$;
      Let $T_\ell^{ord}(\pi)$ be the decreasing ordered sequence of $T_\ell(\pi)$, i.e., $T_\ell^{ord}(\pi) = (\tau_\ell^{(1)}, \ldots, \tau_\ell^{(|T_\ell(\pi)|)})$, where $T_\ell(\pi) = \{\tau_\ell^{(1)}, \ldots, \tau_\ell^{(|T_\ell(\pi)|)}\}$ and $\tau_\ell^{(1)} > \cdots > \tau_\ell^{(|T_\ell(\pi)|)}$;
      for $i := 1$ **to** $|T_\ell(\pi)|$ **do begin**
        Let π'' be the EPD sequence obtained from π' by replacing the i-th time delay with $\tau_\ell^{(i)}$;
        if $cover_D(P \cup \{\pi''\}) \geq \sigma$ **and** $size(\pi'') \geq S$ **then** $\pi' := \pi''$;
      end
    end;
    $P := P \cup \{\pi'\}$
  end;
  return P
end;

Fig. 3. Procedure UPDATE_TIME_DELAYS which is used at Step 4 in Algorithm FES.

equations hold:

$$r' \geq r \text{ and } \sum_{i=1}^{r'-1} \tau_i' \leq \sum_{i=1}^{r-1} \tau_i.$$

Then,

$$\sum_{i=1}^{r'-1} (T_{\max} - \tau_i') \geq \sum_{i=1}^{r-1} (T_{\max} - \tau_i).$$

Moreover we have

$$\sum_{i=1}^{r'} size(p_i') \geq \sum_{i=1}^{r} size(p_i).$$

From the definition of the size of an EPD sequence, we have $\pi' \preceq \pi \Rightarrow size(\pi') \geq size(\pi)$. ■

We can solve R-MEPC by specializing EPD-sequences step by step with a machine learning method proposed by Arimura et al. [4]. We omit the detail of the strategy. From Prop. 1, a maximal K-set of $(R, T_{\max})$-EPD sequences has a local optimal solution with respect to its size. In the next section, we use Algorithm FES rather than a strategy based on maximalities, in order to obtain a set of EPD sequences which has a sufficient large size.

V. APPLICATION TO A SCREENING METHOD FOR INTERNET ACCESS LOGS

A. A screening method

In this section, we propose a screening method for removing irregular packets which are supposed to be occurred by well-known malwares.

We assume that any event e is an element of $\Sigma^{k_1} \times \Sigma^{k_2} \times \cdots \times \Sigma^{k_N}$.

Algorithm SCREENING (SCR);
Input: $D = \{(t, e) \mid t > 0$ and $e \in \Sigma^{k_1} \times \Sigma^{k_2} \times \cdots \times \Sigma^{k_N}.\}$: an incident database, σ: a minimum cover rate, S: a minimum size of $(R, T_{\max})$-EPD sequences;
Output: D': screened incident database;

1) Let P be an output of Algorithm FES for inputs D, σ, and S;
2) Output $D - D(P)$;

Algorithm SCR takes exponentially large time depending on the size of a given incident database $|D|$. To overcome this difficulty, we divide a given database into some smaller databases whose sizes are specified previously.

Algorithm LARGE_SCREENING (LSC);
Input: $D = \{(t, e) \mid t > 0$ and $e \in \Sigma^{k_1} \times \Sigma^{k_2} \times \cdots \times \Sigma^{k_N}.\}$: an incident database, σ: a minimum cover rate σ, S: a minimum size of $(R, T_{\max})$-EPD sequences, s: a size of small databases;
Output:D': screened incident database.

1) Let $((t_1, e_1), (t_2, e_2) \ldots, (t_{|D|}, e_{|D|}))$ be the time-sorted sequence of D, i.e., $t_1 \leq t_2 \leq \cdots \leq t_{|D|}$; Let $\kappa := \lfloor |D|/s \rfloor$;
2) For all i $(1 \leq i \leq \kappa)$, let $D_i := \{(t_{s\cdot(i-1)+1}, e_{s\cdot(i-1)+1}), \ldots, (t_{s\cdot i}, e_{s\cdot i})\}$; If $\kappa \cdot s < |D|$, let $D_{\kappa+1} := \{(t_{s\cdot\kappa+1}, e_{s\cdot\kappa+1}), \ldots, (t_{|D|}, e_{|D|})\}$;
3) For each D_i $(1 \leq i \leq \kappa + 1)$, let D_i' be an output of Algorithm SCR for inputs D_i, σ, and S;
4) Output $D' := \bigcup_{1 \leq i \leq \kappa+1} D_i'$;

B. Experiments on Internet access logs

A typical Internet access log (darknet observation data) includes the time, transmitter's IP address, transmitter's port number, and receiver's port number. The data has no receiver's IP address. This is owing to the concealment of the darknet. Let $\Sigma = \{0, 1, 2, \ldots, 9\}$. We divide the transmitter's address into four parts according to the form of IP address, and those parts are called *address*-1, *address*-2, *address*-3, and *address*-4 from the head, respectively. In addition to those addresses, we have two port numbers, which are called

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

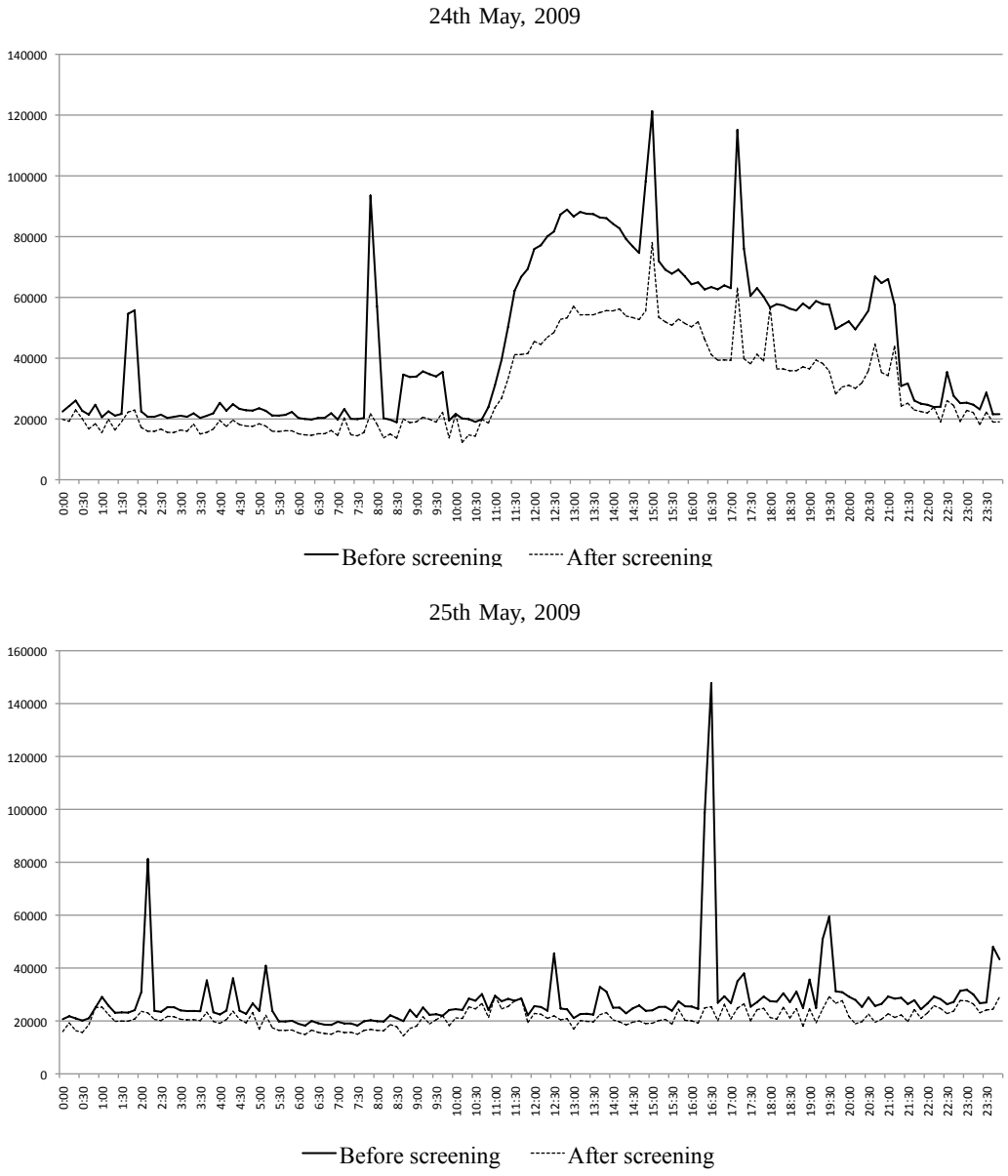

Fig. 4. The solid line (resp. dotted line) means the number of packets before (resp. after) screening every ten minutes. The numbers of packets after screening are 4,258,470 (24th May, 2009, remaining rate 66.98%) and 3,039,185 (25th May, 2009, remaining rate 75.55%), respectively.

port-1 and *port*-2. Then, we have a set of atom data D_i^a $(1 \leq i \leq 6)$ (defined in Algorithm FES) which includes all atom data about *address*-i $(1 \leq i \leq 4)$ and *port*-i $(i = 1, 2)$. Each *address*-i is a string in $\{0, 1, \ldots, 255\}$, and *port*-i is a string in $\{0, 1, \ldots, 65535\}$. Then an event consists of 6 strings whose lengths are at most $3, 3, 3, 3, 5, 5$, respectively. In the following experiment, we consider only $(3, T_{\max})$-EPD sequences. The second parameter $T_{\max}$ is specified later.

In the screening algorithm LSC, if an input incident database is larger than s which is the maximum quantities of packets in each data, we divide it to some data whose quantities of packets are s. We call the time span between the first received packet and the last received packet in each data the *attack period* of the data. When the attack period is short, many packets are received in a short period of time.

- Experimental Data.
  The observation data and the quantity of packets, which we used in each experiments, are shown in Table I.

TABLE I
DARKNET OBSERVATION DATE AND QUANTITIES OF RECEIVED PACKETS

Observation day	Total packets received
8th Nov, 2006	4,039,197
10th Sep, 2008	2,864,698
23rd Jan, 2009	2,884,381
2nd Apr, 2009	10,261,071
24th May, 2009	6,358,187
25th May, 2009	4,022,849
21th Jun, 2009	2,275,630
12th Jul, 2009	6,691,671

- Parameters for screening.
  - Cover rate: $\sigma = 0.1$.
  - Maximum time delay: $T_{\max} = 0.5$ (second).
  - Minimum size of event patterns: $S = 69$. If an event pattern has at least 3 symbols in Σ, the size of the event pattern is more than or equal to 69.

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

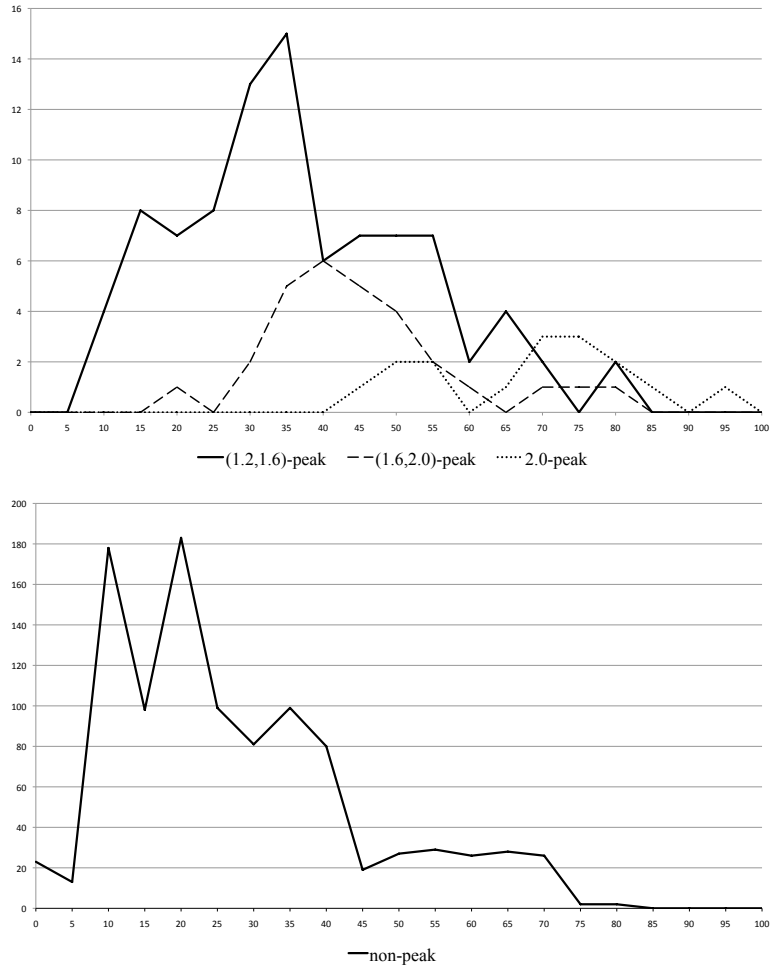

Fig. 5. In the upper graph, the solid line, broken line, and dotted line show the numbers of peaks against their cover rates of the $(1.2, 1.6)$-peak set, $(1.6, 2.0)$-peak set, and 2.0-peak set, respectively. Similarly, the lower graph describes the line for the non-peak set (i.e., $(0, 1.2)$-peak set). The average cover rates are 38.44%, 47.73%, 70.72%, and 29.90% for the $(1.2, 1.6)$-peak set, $(1.6, 2.0)$-peak set, 2.0-peak set, and non-peak set, respectively.

– Upper bound of packets in each divided data: $s = 10,000$. For example, the observation databases of 24th May, 2009 and 25th May, 2009 are divided into 637 and 403 files (incident databases), respectively.

We show the relationship between the quantities of screened database and the unit time in Fig.4, where the upper graph is for 24th May, 2009 and the lower graph for 25th May, 2009. Those two graphs shows the transition of quantities of packets which is received per 10 minutes in the time sequence each day and the quantities of packets which is left by screening the packets. The horizontal and vertical axes mean time sequence and the quantities of packets, respectively. The solid and dotted lines mean the quantities of packets before and after screening, respectively. A peak of the solid lines shows a mass of packets are received during a short period. In the graph for 24th May, 2009, there is no peak of the dotted line. And so, we can say attacks indicated by peaks of the solid line are caused by well-known malwares, and our screening method could detective them. Therefore, we achieved our purpose, removing of packets caused by well-known malwares. Furthermore, in the graph for 25th May, 2009, most of the peaks of the solid line are removed by the screening. But, around 16:30, a peak of the solid line still remains after screening. There are also many packets caused by complex attack patterns which cannot be detected by our method.

In Table II, we show the quantities of remained packets obtained from databases in Table I with their cover rates. The average cover rate of all databases is 36.39%.

TABLE II
QUANTITIES OF REMAINED PACKETS AFTER SCREENING AND THEIR COVER RATES

Observation day	Total remained packets	Cover rate (%)
8th Nov, 2006	2,006,190	50.33
10th Sep, 2008	1,736,558	39.38
23rd Jan, 2009	1,775,265	38.45
2nd Apr, 2009	7,336,516	28.50
24th May, 2009	4,258,470	33.02
25th May, 2009	3,039,185	24.45
21th Jun, 2009	1,797,655	21.00
12th Jul, 2009	3,109,089	53.54

Next we show the relationship between the increasing rates of peaks and their corresponding cover rates. We call a 10 min interval an m-*peak* if the quantity of the packets received for the interval is more then m times the average of two 10 min intervals before and after it. Moreover, for $m < n$, we call a 10 min interval an (m, n)-*peak* if the interval is an m-

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

peak but not an n-*peak*. Let I be a 10 min interval. Let $t(I)$ and $r(I)$ be a start time of I and the cover rate of I after screening. We call the set of all pairs $(t(I), r(I))$ of m-peaks I an m-*peak set*. We define an (m, n)-*peak set* similarly.

We obtained the $(1.2, 1.6)$-peak set, $(1.6, 2.0)$-peak set and 2.0-peak set from screened results of all data in Table I. We show the distribution of the number of peaks against their cover rates in Fig. 5, where the upper graph describes the $(1.2, 1.6)$-peak set, $(1.6, 2.0)$-peak set and 2.0-peak set, and the lower graph describes the non-peak set (i.e., $(0, 1.2)$-peak set). The horizontal and vertical axes mean the cover rates and the number of peaks, respectively. In the upper graph, the solid, broken, and dotted lines mean $(1.2, 1.6)$-peak, $(1.6, 2.0)$-peak, and 2.0-peak, respectively. For example, for the $(1.2, 1.6)$-peak set, the number of peaks whose cover rate is more than or equal to 15% and less than 20% is 8. We can say the higher the peak is, the larger the cover rate. In other words, our method succeeds to detect attacks by malwares that has simple attack patterns even if the quantities of them are large.

VI. Conclusions

In this paper, we proposed a screening method taking advantage of time-span sequential patterns. Moreover we applied our proposed method to darknet observation data and showed its effectiveness for identification of packets caused by well-known malwares. In many cases the reduction rate before and after screening is proportional to the frequency of receiving packets. But experiments showed there are some cases the reduction rate is low despite the concentrated attacks.

As future works, we should inspect to declare the relation between the reduction rate and the frequency of receiving packets on tuning appropriate establishments. Furthermore, to attain the online screening, we are considering efficient algorithms to find a set of time-span sequential patterns.

Acknowledgments

This research is partially supported by the National Institute of Information and Communications Technology (NICT) of Japan, entitled ”Research and Development for Widespread High-speed Incident Analysis”, and partially supported by the Japanese Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (C), 20500016, 2008-2010.

References

[1] nicter project. https://www2.nict.go.jp/y/y211/index.html
[2] SANS Internet Storm Center. http://isc.sans.org/
[3] JPCERT/CC. http://www.jpcert.or.jp
[4] H. Arimura, T. Shinohara, and S. Otsuki. Finding Minimal Generalizations for Unions of Pattern Languages and Its Application to Inductive Inference from Positive Data. *In Proc. the 11th STACS*, Springer, LNCS 775, pages 649–660, 1994.
[5] Y. Fukushima, Y. Hori, K. Sakurai. A Consideration of Feature Extraction for Attacks on Darknet. IEICE technical report, Vol.109, No.285, pages 37–42, 2009 (in japanese).
[6] M. Kim, H. Kang, S. Hong, S. Chung, and J.W. Hong. A Flow-based Method for Abnormal Network Traffic Detection. *Proc. IEEE/IFIP Network Operations and Management Symposium*, pages 599–612, 2004.
[7] T. Miyahara, Y. Suzuki, T. Shoudai, T. Uchida, K. Takahashi, and H. Ueda. Discovery of Frequent Tag Tree Patterns in Semistructured Web Documents. *Proc. 5th PAKDD*, Springer, LNAI 2336, pages 341–355, 2002.
[8] H. Yamasaki, Y. Sasaki, T. Shoudai, T. Uchida, and Y. Suzuki. Learning block-preserving graph patterns and its application to data mining. Machine Learning, Vol.76, No.1, pp.137–173, 2009.

ISBN: 978-988-18210-3-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)