A Multi-level Security Based Autonomic Parameter Selection Approach for an Effective and Early Detection of Internet Worms

Kumar Simkhada\textsuperscript{1,*}, Tarik Taleb\textsuperscript{1,†}, Yuji Waizumi\textsuperscript{1,*}, Abbas Jamalipour\textsuperscript{2}, Kazuo Hashimoto\textsuperscript{1,†}, Nei Kato\textsuperscript{1,‡}, and Yoshiaki Nemoto\textsuperscript{1,*}

\textsuperscript{1} Graduate School of Information Sciences
Tohoku University, Japan
\textsuperscript{2} School of Electrical & Information Engineering
University of Sydney, Australia
\textsuperscript{*}\{kumar,wai,nemoto\}@nemoto.ecei.tohoku.ac.jp
\textsuperscript{†}\{taleb,kh\}@aieit.ecei.tohoku.ac.jp
\textsuperscript{‡}\{kato\}@it.ecei.tohoku.ac.jp

Abstract—In light of the fast propagation of recent Internet worms, human intervention in securing the Internet during worm outbreaks is of little significance. In order to reduce the damage worms may cause, existing Intrusion Detection Systems (IDSs) need to be adaptive to the security-related requirements of their monitoring networks. This paper presents a Multi-level security based Autonomic Parameter Selector (MAPS) that can be implemented over any existing IDSs. The deployment architecture consists of a number of hierarchically placed local security managers, metropolitan security managers, and a global security manager. These security managers report events to a Worm Advisory System (WAS). WAS accordingly sets the threat level of the network. Based on this level, MAPS selects the most optimum parameters for the entire IDS to combat against the propagating worm. The MAPS architecture maintains the system performance by constantly evaluating three metrics, namely False Negative Avoidance, False Positive Avoidance, and Performance Overhead. Extensive experiments, using real network traffic and a recently proposed worm detection system, demonstrate that MAPS is capable of advising an IDS with optimum parameter values to effectively and promptly hinder further propagation of worms.

I. INTRODUCTION

Together with the emergence of high speed networks, recent times have seen the spread of some malicious Internet worms propagating with overwhelmingly high propagation rates [1]–[3]. Studies suggest that it is possible for even faster types of worms, such as Warhol worms and Flash worms to appear in the Internet [4]. If such worms do appear, the whole population of Internet users can be infected within a minute. Because human intervention requires considerably long time to carry out necessary defensive steps, relying on experts to monitor networks during worm outbreaks has little significance. Indeed, by the time experts come up with a defense solution, significant damage may have already been caused. In order to effectively and promptly defend against worms, it is necessary to automatically monitor the security of networks.

Recent literatures present several worm detection systems based on various techniques such as traffic analysis [5], taint analysis [6], data mining [7], honeypots [8], and content-based analysis [9]. The authors propose a signature-based hierarchical worm detection technique for large scale networks in [10] and a multiple-substrings signature generation approach in [11]. These systems deploy several parameters in order to enforce suitable security policies. While these research works aim to make their respective proposed systems practical, one missing point consists in requiring managers to automatically and dynamically adjust their performance parameters according to the requirements of their monitored networks.

In this paper, we present MAPS - a Multi-level Security based Automatic Parameter Selector, which supplies the implemented network IDS with suitable values for its parameters. MAPS takes account of the current Threat Level of the network. The threat level reflects the current worm activity in the network and is decided by a Worm Advisory System (WAS). It is expressed in terms of levels from one to five (thus, multi-level). Based on the current threat level, MAPS dynamically adjusts the security policy taking into account three metrics - false negative avoidance, false positive avoidance, and performance overhead. In case the IDS has to generate the worm signature, a fourth metric - signature generation time is also considered. These metrics are determined by the values of the adjustable parameters used by the corresponding IDS. MAPS selects the optimum values of these parameters that best satisfy the performance requirements under the corresponding threat level.

The rest of this paper is organized as follows. Section II discusses our prior work and highlights the performance metrics that determine the performance of an IDS. A detailed description of the major procedures of MAPS is provided in Section III. The performance evaluation of the system is presented in Section IV. Section V concludes the paper.

II. BACKGROUND AND SYSTEM PERFORMANCE METRICS

The authors proposed a signature-based worm detection system over large scale networks in [10]. The system consists of hierarchically placed local security managers, metropolitan security managers, and a global security manager, as shown in Fig. 1. The local managers collect suspicious or worm-like flows and hand them to their corresponding metropolitan managers. The suspicious flow collection is based on the fact that worms carry similar strings in their payloads. Sample strings are extracted from each incoming network flow and their occurrence frequencies in the traffic are counted. The incoming flows that contain sample strings of high occurrence frequencies are flagged suspicious and are sent to the respective metropolitan managers for further analysis. The
metropolitan managers then use clustering technique to sort only worm flows from the suspicious flow pool. The sorted worms are used to generate the signature for the propagating worm. The generated signature consists of multiple substrings that commonly exist in worm flows but not in normal flows [11].

The worm detection scheme uses several parameters. During the suspicious flow collection, it has to fix appropriate values to the length of each sample string ($L_S$), number of sample strings to be extracted from each flow ($N_S$), and the repetitive occurrence threshold ($\Delta_{TH}$). Similarly, during the signature generation, it requires setting the minimum length of signature substrings ($L_{MIN}$), number of worms flows to be used to extract signature substrings ($N_W$), and the number of normal flows to check the presence of normal strings in the signature ($N_N$). These parameters are empirical in deciding the total signature generation time ($T_{SIG}$). Experimental results obtained from heuristic approaches suggest that the proposed scheme can achieve high accuracy in real world networks. However, owing to the continuously changing network state, different values of the parameters are suitable for different conditions. The primary challenge in extending the system online is to fix these values automatically.

Various metrics can be used to evaluate the performance of an IDS. In [12], Fink et al. point out some of these metrics, such as accuracy, detection rate, flexibility, overhead, error reporting, router interaction, timeliness, etc. Considering a worm detection scenario, False Negative Avoidance, False Positive Avoidance rate, and Performance Overhead rate need primary follow-ups. Apart from these three metrics, we also choose the Signature Generation Time as another performance metric when the signature generation process is involved. We next discuss these metrics and explain how they should be adjusted for an autonomic security management of networks.

A. False Negative Avoidance (FNA)

The FNA indicates the detection rate of the IDS. For each threat level $L$ ($1 \leq L \leq 5$), critical FNA rate (minimum detection rate) $\eta_L$ ($0 < \eta_1 < ... < \eta_5 < 1$) is pre-defined by the administrator. A high FNA is required when there is an active propagation of Internet worms. During such adversaries, the IDS system may have to sacrifice other two metrics to some extent.

B. False Positive Avoidance (FPA)

FPA indicates how fairly the IDS system allows normal traffic to pass undisturbed. FPA is expressed in terms of false positive rate (FPR) as, $FPA = 1 - FPR$. As there is a tradeoff between FPA and FNA, FPA rate may need relaxation if FNA rate is increased. The desired minimum FPA rate of the system at each threat level is also defined by the administrator as $\sigma_L$ ($1 > \sigma_1 > ... > \sigma_5 > 0$).

C. Performance Overhead (PO)

Cost is another prominent factor in determining the performance of an IDS. It is thus necessary to design the IDS in such a way that maximum security and accuracy can be achieved with a nominal overhead. We define PO rate as the percentage of available buffer size required for analysis. Similarly to FNA and FPA, the critical PO rate (maximum affordable PO rate) of an IDS at each threat level ($L$) is defined by the administrator as $\theta_L$ ($0 < \theta_1 < ... < \theta_5 < 1$).

There is a trade-off between FNA, FPA, and PO rates. It is always preferable to achieve high FNA and FPA at the cost a low PO. However, during emergency situations such as worm outbreaks, it is necessary to sacrifice some FPA and PO in favor of FNA. In contrast, during normal network states, FNA rate is slightly reduced in order to achieve higher FPA and lower PO rates.

D. Signature Generation Time

Signature generation time ($T_{SIG}$) is a vital performance metric for signature-based IDSs. During high threat levels, IDSs can afford relatively short time for signature generation. The value of $T_{SIG}$ for email worms such as Beagle, NetSky, and MyDoom can be fairly long (a few minutes). However, for fast spreading scanning worms such as Code Red, Blaster, and Slammer, signatures are required within a limited span of time. Signatures for scanning worms should be generated within 60 seconds in order to effectively control their propagation [4]. This indicates that an IDS system has to automatically generate signature within a significantly limited time after the worm has been detected. Indeed, we first ensure that the signature is generated within the defined $T_{SIG}$. FNA, FPA, and PO rates are then considered.

**Fig. 1.** The envisioned hierarchical worm detection architecture.
Fig. 2. A typical deployment of the proposed system.

III. A MULTI-LEVEL SECURITY BASED AUTONOMIC PARAMETER SELECTION SCHEME

In this section, we provide a detailed description of the proposed scheme. A typical deployment of MAPS is illustrated in Fig. 2. The IDS implemented at the gate of the network updates security events to a Worm Advisory System (WAS). WAS accordingly defines the threat level of the network. The MAPS-monitor uses the defined threat level to find suitable parameters to advise the IDS. Note that MAPS does not propose a new worm detection scheme but advises the implemented IDS with optimum values of its parameters in order to efficiently defend against the spreading worm. The operations of WAS and MAPS are further explained below.

A. Worm Advisory System (WAS)

WAS defines the threat level of the network based on the events reported by the IDS or other entities, such as, firewalls, IDSs of other collaborating networks, and high-hierarchy components. It analyzes the events in timeslots. Threat level one corresponds to a normal network state when no malicious activities are reported. In contrary, threat level five implies that the system is either under attack or is in a grave danger of a worm propagation. The threat level of a network is sent to MAPS by WAS accordingly defines the threat level of the network. The MAPS-monitor uses the defined threat level to find suitable parameters to advise the IDS. Note that MAPS does not propose a new worm detection scheme but advises the implemented IDS with optimum values of its parameters in order to efficiently defend against the spreading worm. The operations of WAS and MAPS are further explained below.

B. MAPS Monitor

It is a common trend to investigate the performance of an IDS with test data before implementing it online. The MAPS monitor is provided with the results obtained when the IDS system is initially evaluated with such test data. Let Q be the set of these results. Q stores each scenario k as a quadruple $T_k = \{ P_k, \eta_k, \sigma_k, \theta_k \}$, where $P_k$ is the set of parameter values used and $\eta_k$, $\sigma_k$, and $\theta_k$ are the corresponding values of the three metrics in the test scenario $k$.

Fig. 3 depicts the flow chart of the MAPS algorithm. The current threat level of the network is sent to MAPS by WAS. The critical rates of FNA, FPA, and PO, denoted by $\eta_c$, $\sigma_c$, and $\theta_c$, respectively, are initialized to the critical values of the corresponding threat level, $\eta_L$, $\sigma_L$, and $\theta_L$, respectively. The proposed algorithm takes a greedy approach in maximizing the

$$\eta = \eta_L; \sigma = \sigma_L; \theta = \theta_L$$

$$\eta = 1; \sigma = 1; \theta = 0$$

$$Q_1 = \{ T_k : T_k \in Q ; \theta_k \leq \theta \}$$

$$Q_1$$ contains only the quadruples whose parameter can perform within the overhead $\theta$. $Q_1 = \phi$ implies that the IDS cannot function within the given transitory PO rate ($\theta$). Thus, the MAPS monitor increases $\theta$ by $\Delta \theta$. If $\theta$ is within the critical PO rate ($\theta_c$), it carries out another search from $Q$. Otherwise, $\theta$ is incremented to $(\theta_c + \Delta \theta)$ and the whole search is repeated, but with a higher $\theta_c$.

From the quadruples in $Q_1$, MAPS then searches for the quadruples whose per accuracy is more than the transitory critical FPA rate ($\sigma$). Hence, the set of the selected quadruples ($Q_2$), which fulfill the PO and FPA requirements, is expressed as

$$Q_2 = \{ T_k : P_k \in Q_1 ; \sigma_k \geq \sigma \}$$

In case $Q_2 = \phi$, the MAPS monitor decreases $\sigma$ by a fixed value. If this decrease does not affect the relation ($\sigma > \sigma_c$), MAPS searches for appropriate quadruples from $Q_1$ again. Otherwise, $\sigma$ is relaxed and $\theta$ is increased. The whole search
is repeated with a lower $\sigma_c$ and a higher $\theta_c$. This process is repeated until $(Q_2 \neq \phi)$. Similar adjustment is carried out for FNA. MAPS searches for quadruples in $Q_2$ that satisfy the FNA transitory critical rate ($\eta$). The set of the quadruples that are now chosen ($Q_3$) is expressed as,

$$Q_3 = \{T_k : T_k \in Q_2; \eta_k \geq \eta\}$$

Similarly to the PO and FPA checks, MAPS confirms whether transitory critical FNA rate is satisfied. If $Q_3 = \phi$, MAPS relaxes $\eta$ to $\eta - \Delta \eta$. If the new transitory critical FNA rate is more than $\eta_c$, it carries out next search from $Q_2$. Otherwise, $\eta_c$ and $\sigma_c$ are further reduced while $\theta_c$ is increased to start the search from the beginning. This is repeated until $Q_3 \neq \phi$. MAPS uses $Q_3$ to select the quadruple that is optimum for the current network state.

As $Q_3$ may contain multiple quadruples, a separate parameter needs to be defined to select the best quadruple from $Q_3$. We define partial gains of FNA, FPA, and PO for the quadruple $T_k$, denoted by $E_{\eta k}$, $E_{\sigma k}$, and $E_{\theta k}$, respectively, as

$$E_{\eta k} = \eta_k - \eta_L; \quad E_{\sigma k} = \sigma_k - \sigma_L; \quad E_{\theta k} = \theta_L - \theta_k$$

where $\eta_L$, $\sigma_L$, and $\theta_L$ are the critical FNA, FPA, and PO rates, respectively, for the corresponding threat level $L$. Using the partial gains of the three metrics, the total gain of $T_k$ is calculated as

$$E_k = E_{\eta k} + E_{\sigma k} + E_{\theta k}.$$  \hspace{1cm} (5)

From $Q_3$, the quadruple with the maximum value of $E$ is selected. The parameter values of the selected quadruple are sent to the IDS for deployment.

**IV. PERFORMANCE EVALUATION**

**A. Experimental Set-up**

The experiment is carried out using the results obtained during the evaluation of the worm-detection IDS proposed in [10]. During the suspicious flow collection process, the IDS extracts $N_S$ sample strings of length $L_S$ from each incoming network flow and counts the occurrence frequency of each sample string. It then uses a Repetitive Occurrence Threshold ($\Delta_{TH}$) to decide if a worm is propagating. The FNA and FPA rates for different $N_S$, $L_S$, and $\Delta_{TH}$ are obtained using a real network data that contains traces of Beagle worms. The system is assumed to set aside a maximum of 1,000 Bytes of buffer storage per each flow. The PO rate per flow is, thus, defined as

$$\theta = \frac{L_S \times N_S}{1000}.$$  \hspace{1cm} (6)

During high threat levels, the IDS needs to achieve a high FNA rate, even at the cost of some FPA and PO rates. Hence, we set the critical values of FNA, FPA, and PO rates for the five threat levels to \{0.70, 0.80, 0.85, 0.90, 0.95\}, \{0.95, 0.90, 0.85, 0.80, 0.70\}, and \{0.20, 0.40, 0.50, 0.60, 1.00\}. In the first experiment, we use the above-mentioned set-up to investigate change in threat level of the network when events are reported. We then investigate the IDS performance under these threat levels. In the second experiment, we consider a signature generation scenario. We assume that a threat level has been fixed by the WAS. The signature generation time

![Fig. 4. Adjustments to the threat level with respect to the number of events reported.](image-url)

$(T_{SIG})$ at threat levels one to five are set to 105, 90, 75, 60, and 45 seconds respectively. MAPS finds the best set of parameters that can generate signature within the $T_{SIG}$ of the current threat level. These parameters include the number of worm flows to be used for signature generation ($N_W$), minimum length of signature substrings to be extracted ($L_{MIN}$), and the number of normal flows required to check the inclusion of normal substrings in the signature ($N_N$). The corresponding FNA, FPA, and PO rates are then used to evaluate the performance of the proposed scheme. We obtain the FNA and FPA rates directly from preliminary experiments which are presented in [10]. PO rate is defined as the percentage of the worm payload that is necessary to generate the signature under the selected scenario.

**B. Experimental Results**

Fig. 4 depicts the change in the threat level when events are reported. As illustrated in the figure, threat level is increased when a large number of events are reported and when the number of reported events shows an increasing tendency in each timeslot. The stability of threat level primarily depends upon the length of each timeslot and the value of event threshold. If the timeslot length is short, the WAS can quickly respond to the reported events. However, this also degrades the stability of the threat level management because threat level is subject to changes within a short span of time. Considering the fact that a new set of parameters has to be deployed by the IDS during each threat level change, frequent changes in threat level make the IDS unstable. This phenomenon is illustrated

---

**Authorized licensed use limited to: TOHOKU UNIVERSITY. Downloaded on March 26, 2009 at 03:00 from IEEE Xplore. Restrictions apply.**
Fig. 5.  FNA, FPA, and PO rates for different threat levels.

Table I

<table>
<thead>
<tr>
<th>Threat level</th>
<th>$T_G$ (sec)</th>
<th>FNA rate</th>
<th>FPA rate</th>
<th>PO rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>105</td>
<td>1.000</td>
<td>1.000</td>
<td>0.013</td>
</tr>
<tr>
<td>2</td>
<td>90</td>
<td>1.000</td>
<td>1.000</td>
<td>0.013</td>
</tr>
<tr>
<td>3</td>
<td>75</td>
<td>0.998</td>
<td>1.000</td>
<td>0.018</td>
</tr>
<tr>
<td>4</td>
<td>60</td>
<td>0.970</td>
<td>1.000</td>
<td>0.048</td>
</tr>
<tr>
<td>5</td>
<td>45</td>
<td>0.970</td>
<td>1.000</td>
<td>0.048</td>
</tr>
</tbody>
</table>

V. Concluding Remarks

In this paper, we proposed an architecture for selecting the most optimum parameters to enable an IDS system maximum detection accuracy while maintaining minimal cost. In the proposed scheme, a worm advisory system first defines the threat level of the network. Taking into account the threat level, the proposed scheme selects the most optimum values of system parameters to best deal with the impending worm outbreaks. A high priority is given to avoid false negatives and false positives. At the same time, an overhead check is also performed before parameters are sent to the IDS for a real implementation. Through experiments, we showed that the proposed scheme is effective in reducing the cost of security monitoring, and can be of great help to administrators during worm outbreaks.

References