RESEARCH ARTICLE

Anomaly detection and response approach based on mapping requests

Ming Wan1,2, Hong-Ke Zhang2, Tin-Yu Wu3* and Chi-Hsiang Lo4

1 Laboratory of Networked Control Systems, Shenyang Institute of Automation Chinese Academy of Sciences, Shenyang, China
2 National Engineering Laboratory for Next Generation Internet Interconnection Devices, Beijing Jiaotong University, Beijing, China
3 Department of Computer Science and Information Engineering, National Ilan University, I-Lan, Taiwan
4 Department of Computer Science and Information Engineering and Department of Electronic Engineering, National Ilan University, I-Lan, Taiwan

ABSTRACT

There is an increasing consensus that the locator/identifier separation of IP address is necessary to resolve the scalability issues of current Internet routing architecture. After identifiers are separated from locators, an identifier-to-locator mapping service must be employed to map identifiers onto locators. From this point, this paper proposes an anomaly detection and response approach based on mapping requests. By using the cumulative sum algorithm for change point detection, this approach introduces the anomalous traffic detection of mapping requests to diagnose the aberrant network behaviors. Once alarming, two effective response methods can be chosen to control the anomalous attack traffic in real time. Furthermore, in order to decouple the mapping request traffic from the mapping cache, this approach not only takes into account the mapping cache timeout but also puts forward a practical mapping request threshold algorithm. In particular, our simulation results show that, compared with the anomaly detection approach based on network traffic, the proposed approach is more advantageous and efficient. In addition, we also discuss the possible false positive and false negative problems, which may be caused by some accidental phenomena. Copyright © 2014 John Wiley & Sons, Ltd.

KEYWORDS
locator/identifier separation; anomaly detection and response; cumulative sum; mapping request threshold algorithm

*Correspondence
Tin-Yu Wu, Department of Computer Science and Information Engineering, National Ilan University, I-Lan, Taiwan.
E-mail: tyw@niu.edu.tw

1. INTRODUCTION

In recent years, there is an increasing consensus that locator/identifier separation is a promising solution to the overloading semantics of IP addresses [1–6]. That is to say, IP addresses are divided into two independent namespaces: a locator namespace to represent the locations of end hosts and an identifier namespace to represent the identities of end hosts. Moreover, locators are used in the network layer for forwarding packets and locating nodes, while identifiers are used in the transport and upper layers for identifying nodes.

In order to separate identifiers from locators, there is no substitute for the identifier-to-locator mapping service to map identifiers onto locators. In the designs of the mapping service, a critical challenge is how to achieve the scalability and security of a mapping service [7]. On the basis of dissimilar hypotheses, the proposed mapping services in the literatures [8–17] fall into two approaches. One is that some routers or nodes store the identifier-to-locator mappings for all identifiers. However, the scalability of these ways is relatively poor because it is rather difficult for a single router or node to store and update all identifier-to-locator mappings, while the other is that all identifier-to-locator mappings are distributed and maintained in some organized nodes, which are viewed as the RVs in the rest of the paper for ease of presentation. When a tunnel router (TR) wants to resolve a locator for some identifier, which may be structured or flat, it only needs to send a mapping request to the corresponding RV. Compared with the former, the latter is more scalable and efficient. In this paper, the proposed anomaly detection and response approach is based on the latter approach.

In fact, the major factors of locator/identifier separation can be concluded as follows [18]: First, it has been widely recognized that today’s Internet routing architecture is
facing serious scalability issues. Multi-homing, traffic engineering, and suboptimal address allocation are making the forwarding information base of the default free zone growing at an increasing and potentially alarming rate. Second, the design of today’s Internet cannot satisfy more and more mobility demands. Third, we are also experiencing a great diversity of malicious attacks, such as IP address spoofing and hijacking. In short, these serious issues are mainly caused by the overloading of IP address semantics. Besides, the locator/identifier separation and mapping service are different with the domain name system (DNS). On the one hand, the DNS associates the IP addresses with the domain names, and the locator/identifier separation only divides the whole IP address space into the locator namespace and the identifier namespace. On the other hand, in DNS, each end host needs to send the translating request to the DNS server. While in networks with locator/identifier separation, the TRs are responsible for sending a mapping request to the corresponding RV.

It is worth considering whether a new network model can provide more and better security services for the public when this network model comes to being. We discuss the characteristics of the mapping service in detail and find that we can diagnose some aberrant network behaviors by using the abnormal detection of mapping request traffic. In the existing Internet, because the number of threats has been increasing at an alarming rate and anomalous traffic events, such as the spread of worms and the distributed denial of service (DDoS) attacks, always contaminate the Internet [19], identifying and preventing network anomalies is a critical challenge for network pursuers [20, 21]. In fact, identifying and preventing anomalies requires a sophisticated real-time monitoring infrastructure. However, it is rather difficult to achieve this purpose for the Internet service providers (ISPs) or other service providers because the nature of real network traffic is high-dimensional and unpredictable, and identifying and diagnosing network anomalies in a timely fashion from the collected traffic is a very demanding task. Some anomalies may not necessarily create congestion in the network, but they can have a dramatic impact on a customer or an end user [22]. In order to detect these network anomalies, this paper proposes novel anomaly detection and response approach, which substitutes the anomaly detection of network traffic with the anomaly detection of mapping request traffic, and this approach uses the celebrated cumulative sum (CUSUM) algorithm for change point detection. Once alarming, two effective response methods can be chosen to control the anomalous attack traffic in real time. In addition, because the mapping cache in each TR may reduce the detection effectiveness of our approach, we analyze the influence of mapping cache timeout and present a practical mapping request threshold algorithm to decouple the mapping request traffic from the mapping cache. To be specific, the goal of this paper is not to clarify the cause of network anomalies but rather to provide a distinctive detection and response approach to diagnose network traffic anomaly.

The advantages of our approach are listed as follows: First, our approach can give an alarm in advance, because the detected anomaly of mapping request traffic precedes that of network traffic. Second, the mapping request traffic, unlike the complicated and high-dimensional network traffic, is simple and single dimensional, and it is fairly easy to accomplish the anomaly detection for the ISPs or other service providers. Finally, when an alarm occurs, our approach can provide a real-time response for the victim.

Not surprisingly, because nothing is perfect, we also discuss the possible false positive and false negative problems in our approach. In this paper, our purpose is not to explore a quantitative study on the false positive or false negative rate due to the actual situations of today’s Internet but to analyze some accidental phenomena or factors, which may result in the false positive or false negative problems in our approach, and these analyses will provide a basis for our further research.

The rest of this paper is organized as follows. Section 2 introduces the background and related works for locator/identifier separation and existing network anomaly detection approaches. The basic network model and assumptions are described in Section 3. Section 4 provides the proposed detection and response approach in detail. Our simulation model and experimental results are presented in Section 5. Finally, Section 6 concludes the paper and discusses our future work.

2. BACKGROUND AND RELATED WORKS

In this section, we express some related research efforts from both academia and industry. First of all, we illustrate various locator/identifier separation proposals and some existing mapping services. And then, we describe some typical practices employed in today’s Internet for general anomaly detection.

2.1. Locator/identifier separation

In recent years, various new proposals based on locator/identifier separation have been advanced in the literatures. Broadly speaking, these proposals can be divided into two categories. One is called host-based solution, in which the identifiers are separated from locators in the hosts’ network stacks. In order to accomplish locator/identifier separation, the host-based solution must upgrade or modify the hosts’ protocol stacks. Typical proposals include Shim6 [1], Internet indirection infrastructure [2], and host identity protocol [3]. In general, this solution needs to add an additional shim layer, called host identity layer in host identity protocol. On the one hand, the host identity layer uses host identifiers to represent the identities of end hosts. On the other hand, the network layer uses routing locators to represent the locations of end hosts. Although proposals in this category have the advantages of host mobility and...
multi-homing, it is criticized that they demand the host changes and have a hard-to-deploy issue in today’s Internet. The other is named the network-based solution, which is also known as core-edge separation. Compared with the host-based solution, the network-based solution requires no change to end hosts and is relatively easy to deploy and maintain. Sample proposals include locator/ID Separation protocol (LISP) [4], Internet vastly improved plumbing [5], and SIX/ONE [6]. Generally speaking, routing objects that may be either IPv4 addresses or IPv6 addresses in edge networks are identifiers, but those in the transit core are locators. For example, in LISP, being carried out by an active working group in Internet Engineering Task Force, endpoint identifiers (EIDs) represent the identities of end hosts, and routing locators represent the locations of end hosts. Moreover, packets coming from end hosts are first tunneled at their Ingress TRs and routed to their corresponding Egress TRs, by which these packets are forwarded to their destinations. The core-edge separation solution may require other schemes to handle host mobility or multi-homing, but its big advantage in deployment is that it needs no update or awareness from user side. To sum up, both these two categories try to separate the identifiers from locators, and they have their own advantages and disadvantages, respectively.

A critical challenge with locator/identifier separation is how to design an appropriate mapping service. As set forth, the purpose of the mapping service is to distribute and maintain the identifier-to-locator mappings for all end hosts, and it is highly important to make the mapping service robust against failures and attacks. On the basis of this basic consideration, a great diversity of mapping services have been proposed by intellectual pursuits. In [5,8,9], by using some mechanisms to distribute identifier-to-locator mapping changes, some routers or nodes store and update the complete identifier-to-locator mappings for all end hosts in the whole network. Although these approaches are provided with very low map-resolution delay, a major bottleneck is that they are unscaleable because the number of all identifier-to-locator mappings is too huge to store in a single router or node. Namely, it is impossible to timely distribute mapping changes to all routers or nodes in the whole network. Different from the works in [5,8,9], various overlay topologies have been argued to distribute identifier-to-locator mapping reach-ability information [10–12]. LISP distributed hash table (DHT) [10] proposes to distribute this information by using DHTs. LISP alternative topology [11] proposes a hierarchical overlay architecture through which the reach-ability information is broadcasted in a way similar to the propagation of the Internet protocol prefix in the Internet today. Another similar approach is content distribution overlay network service (LISP-CONS) [12], whose overlay infrastructure is a strict hierarchy. However, the main drawback of these approaches is the longer resolution delay because a mapping request has to pass through the overlay topology. In order to reduce the resolution delay, the work in [13] proposes a two-level mapping system, in which nodes can first lookup the top-level DHT mapping system for the reach-ability pointer and cache this information so that future mapping requests can directly go to the corresponding bottom-level DHTs. For the self-certifying identifiers, the work in [14] distributively stores the identifier-to-locator mappings in the DNS by using a special resource record. Obviously, the benefit of this approach is that it need not build new overlay architecture, but it increases the overhead on the DNS because of the mapping updates and requests. Also, the work in [15] uses a content-addressable network to map the flat identifiers onto locators. In order to make the mapping service more mobility-efficient, a further analysis has been provided in [16]. In addition, by setting a set of identifier-to-locator mapping service providers (MSPs), the work in [17] puts forward a new idea, which lets the users of identifiers choose their preferred/trusted MSPs.

In this paper, our motive is not to discuss how to design a scalable and secure mapping service but to utilize the general network model of locator/identifier separation to propose our anomaly detection and response approach. However, the ordinary mapping service is the basic prerequisite for our approach.

2.2. Anomaly detection

In [23,24], this type of approach must first build a baseline model to depict the normal network traffic. Then, an alarm will be triggered when any pattern of the audit traffic departs from this baseline model. Plainly, the potential benefit of these approaches is that some undiscovered attacks can be detected because their behaviors may be different from the baseline model, but the main drawbacks are that obtaining the baseline model is prohibitively expensive and time-consuming. Also, some degree of false alarm is inevitable. Another type of anomaly detection approach aims for the change point detection, for example, adaptive threshold [25], CUSUM [26,27] wavelets [28] maximum entropy [29], and flow correlation [30]. While these approaches need to find the trade-offs among the detection probability, the false positive rate and the detection delay. In addition, the work in [25] presents and evaluates the adaptive threshold and CUSUM anomaly detection approaches for detecting transmission control protocol synchronize sequence numbers (TCP SYN) attacks. For different DDoS and worm attacks, many new algorithms have been proposed in the literatures. The work in [31] focuses on the spoofed address attack and proposes the detection approaches by exploiting spatial and temporal correlation of attack traffic. In order to detect stealthy DDoS, the work in [32] advises a real-time detection approach using time-series decomposition. The work in [33] discusses the effects of multivariate correlation analysis on the DDoS detection and proposes a covariance analysis model for detecting SYN flooding attacks. Besides, a great many principal component analysis based approaches [22,34,35] have been presented to detect DDoS attacks by decomposing the original traffic into the normal
and abnormal proportions. However, it is very difficult to guarantee the real-time performance for these approaches. Also, the approach based on sketch and principal component analysis to detect worm activities is provided in [36]. In brief, a number of different anomaly approaches have been summarized in the work in [20] and [21].

In short, the existing anomaly detection approaches only diagnose and identify the anomalous network behaviors, and they cannot assure to detect and control the attack traffic in real time. Therefore, in order to achieve this goal, we propose an anomaly detection and response approach from another perspective, which has never been considered before.

3. NETWORK MODEL

We consider a network model based on the core-edge separation [4–6]. As shown in Figure 1, the edge network is composed of a great many customer networks (CNs), which use identifiers to route packets, and transit core is composed of many provider networks (PNs), which use globally routable locators to route packets. In addition, TRs located at edge of either transit core or edge network connect CNs with PNs. Notice that, in our network model, identifiers may be either hierarchically structured [4,5,8] or flat [3,15]. At the same time, we assume all resolvers (RVs) belonging to all PNs work together for the mapping service, and each PN can own and maintain one or more RVs. It is also worth noting that we need not consider how to accomplish the mapping service by these RVs, because it does not affect our anomaly detection and response approach.

In Figure 1, assuming that an end host with EIDsrc in CN1 wants to setup a connection with another end host with EIDdst in another CN2.

Step 1: Figure 1(a) depicts that the source host sends its first packet to its TR1, using EIDsrc and EIDdst as the source and destination address of the packet, respectively.

Step 2: When TR1 whose locator is Loc1 receives the first packet from the source host, it looks up a locator for EIDdst in its local identifier-to-locator mapping cache, which stores some recent used mapping items as shown in Figure 1(b). If the cache hits (Suppose the locator for EIDdst in the mapping cache is Loc2, which is also the locator of the destination host’s TR2), go to step 6; otherwise, go to step 3.

Step 3: Suppose that the identifier-to-locator mapping item for EIDdst is stored in RV3, which is the RV 3 in Figure 1. TR1 sends a mapping request to RV3 for the identifier-to-locator mapping item as shown in Figure 1(c).

Step 4: After RV3 receives the mapping request, it sends identifier-to-locator mapping item for EIDdst to TR1 as shown in Figure 1(d).

Step 5: While receiving the mapping item for EIDdst, TR1 first stores it in its local identifier-to-locator mapping cache for possible future usage. Assume that the resolved locator for EIDdst is Loc2, which is also the locator of the destination host’s TR2.

Step 6: Then, TR1 encapsulates the received packet with a new routing header whose source address and destination address is Loc1 and Loc2, respectively. After that, TR1 sends the new encapsulated packet into transit core, and the packet will be routed and forwarded to TR2 as shown in Figure 1(e).

Step 7: After receiving the encapsulated packet, TR2 simply strips the outer header as shown in Figure 1(f). In addition, in order to reduce the lookup times for the identifier-to-locator mapping for EIDsrc, TR2 may store the locator Loc1 for EIDsrc in its local identifier-to-locator mapping cache.

Step 8: TR2 forwards the original packet to its destination host with EIDdst as shown in Figure 1(g).

4. MAPPING DETECTION AND RESPONSE APPROACH

With the drawbacks or vulnerabilities of the existing software and protocols, diversified network attacks are designed to bring the computer or network to its knees. Worse still, these attacks may seriously influence the normal use of the network and result in an enormous waste of network resources and economic losses. Therefore, it is highly necessary to study on the anomaly detection and response approach. In the networks of locator/identifier separation, when an end host wants to establish a connection to another end host, its TR must first send a mapping request to the corresponding mapping server that is viewed as a RV in the rest of the paper for ease of presentation. In this sense, it is worth considering whether the anomaly detection of mapping requests can identify or diagnose
the aberrant network behaviors, especially when a few network anomalies can have a dramatic impact on some servers or end hosts. With this infusive consideration, we convincingly demonstrate the feasibility and validity of our viewpoint by analyzing the authoritative 1998 Defense Advanced Research Projects Agency Intrusion Detection System evaluation dataset [37].

In order to make the result more intuitive and comprehensible, we adopt the training data on Wednesday of the third week as the experimental data, in which the end host pascal.eyrie.af.mil suffered from the Smurf attack. Figure 2 depicts the network traffic produced by pascal.eyrie.af.mil. This figure shows that a strong outburst is caused by the Smurf attack at about the 240th minute, and the traffic volume suddenly increases to approximately 76,000 packets. Similarly, we also give the corresponding number of mapping requests for the end host pascal.eyrie.af.mil in Figure 3. Notice that the mapping cache in the TR may affect the number of mapping requests [15,38,39]. The longer the mapping cache timeout is, the smaller the number of mapping requests will be. In accordance with the advice in [39], we assume the mapping cache timeout is 60 s. When the Smurf attack happens, the number of mapping requests, which is about 10 under normal conditions, also increases to approximately 1050 in intensity, as shown in Figure 3. From these two figures, we can see that the mapping request traffic is more advantageous to measure network anomaly, and the anomaly detection based on mapping requests can indirectly identify and diagnose some aberrant network behaviors.

In the Mapping Detection and Response Approach (MDRA), identifying and diagnosing some anomalous network behaviors by judging the mapping request traffic anomaly. In particular, MDRA can detect and identify various DDoS attacks. In our network model, we assume all identifier-to-locator mapping items are distributive stored in different RVs, and each mapping item is maintained by only one RV.

Figure 4 gives an anomaly example, which is aroused by the mapping request anomaly flows. When dozens of attackers aim at overwhelming a victim with an immense volume of useless traffic, their TRs first send the mapping requests for the victim’s locator, such as Figure 4(1)–(n). From this point, we deploy MDRA to each RV, and each RV implements to detect and diagnose the mapping request traffic anomaly for the identifier-to-locator mapping items maintained by this RV. Furthermore, in order to alarm in real time, MDRA uses the celebrated CUSUM algorithm for change point detection. Once an RV generates an alarm for some identifier-to-locator mapping item, the RV can respond to control the attack traffic by cooperating with the TRs or conditionally the incoming mapping requests. In addition, because the existing mapping cache in each TR may reduce the detection effectiveness of MDRA, we analyze the influence of mapping cache timeout and present a practical mapping request threshold algorithm to address this problem.
4.1. Alarm and revocation

The CUSUM algorithm based on hypothesis testing is ascribed to the family of change point detection algorithms. The principle of change point detection is to estimate whether the observed random variables keep statistically homogeneous, and if not, it can identify the change point in time [25,27]. As is well known, CUSUM is based on the fact: if the average value of statistical process changes, stochastic probability distribution will also change [40]. Unlike the high-dimensional network traffic, the mapping request traffic is simple and single dimensional. In other words, in order to generate an alarm, we can directly apply the CUSUM algorithm to MDRA, and our ultimate purpose is to detect the mapping request traffic anomaly.

Assume the time sequence $x_1,x_2,...,x_n$ is independent identically distributed variables with the Gaussian distribution $N(0,1)$, and the time sequence $x_1,x_2,...,x_{n+1}$ is independent identically distributed variables with the Gaussian distribution $N(0,1)$, where $\nu(\nu < n)$ is an unknown change point and the value $x_i$ represents the number of mapping requests for some identifier in the $i_{th}$ time interval. Suppose there is no change point, namely, $\nu = \infty$, the statistical value of the log-likelihood ratio is

$$Z_n = \max_{1 \leq \nu < n} \left( \sum_{j=1}^{n} \left( x_i - \frac{\phi}{2} \right) \right)$$

Equation (1) describes the most ordinary CUSUM statistical value. Suppose $h$ ($h > 0$) is a chosen threshold, which may be determined empirically through experiments. If $Z_n \leq h$, $i = 1,2,...,n$, the former $n-1$ values are under normal conditions; if $Z_n > h$, anomaly happens and an alarm should be generated. Similarly, the foregoing judgment also can be understood that if an existing number $r$ satisfies \[ \sum_{i=0}^{r} x_{i+1} > (r+1) \frac{\phi}{2} \text{, where } 0 \leq r \leq n-1, \]
then the anomaly happens and an alarm should be generated.

The aforementioned equation illustrates the basic CUSUM algorithm. However, the pre-requisite is that we have assumed that $\{x_i\}$ is independent Gaussian random variables. Of course, this is not true for network traffic measurements owing to seasonality, trends, and time correlations [41]. Therefore, in order to remove such nonstationary behaviors, the work in [25] further improves the basic CUSUM algorithm, and $Z_n$ can be calculated by

$$Z_n = \begin{cases} Z_{n-1} \quad \text{if } Z_{n-1} + \frac{\alpha Z_{n-1}}{\sigma^2} \left( x_n - \bar{Z}_{n-1} - \frac{\alpha \bar{Z}_{n-1}}{2} \right) > 0 \\ \phantom{Z_{n-1}} \quad \text{otherwise} \end{cases}$$

where $\alpha$ is an amplitude percentage parameter, which intuitively corresponds to the most probable percentage of increase of the mean rate after a change has happened, and $\sigma^2$ is the variance of $\sigma$. Meanwhile, the mean $\bar{Z}_n$ can be calculated by using an exponentially weighted moving average of previous measurements [42]:

$$\bar{Z}_n = \beta \bar{Z}_{n-1} + (1 - \beta) x_n$$

where $\beta$ is the exponentially weighted moving average factor. Thus, the conditions to generate an alarm can be summarized as follows:

$$f(Z_n) = \begin{cases} 1, \text{ if } Z_n > h; \\ 0, \text{ otherwise}. \end{cases}$$

In Equation (4), $1$ indicates that the anomaly in the detected sequence $\{x_i\}$ is identified, and an alarm is generated. By contrast, $0$ indicates that the detected sequence $\{x_i\}$ is normal.

However, a disadvantage or flaw exists in the CUSUM algorithm [43]. That is, when the anomaly or attack is over, CUSUM still continues generating the false alarms for a long time. Resulting from accumulation effect of the CUSUM algorithm, the increased amount to $Z_n$ caused by the attack traffic is much greater than the decreasing amount provided by the normal traffic. In order to resolve this issue, MDRA uses the following formula to revoke an alarm.

$$f(Z_n) = 0, \text{ if } Z_n \geq h \text{ and } x_i < \rho \bar{Z}_{2v-i}$$

where $\phi$ is an amplitude and $\phi > 1$. Assume an anomalous behavior happens at time $v$, and $x_i$ is the detected mapping request traffic in the $i_{th}$ time interval, $i > v$, $\bar{Z}_{2v-i}$ is the traffic mean of the former $2v-i$ time intervals, which can be calculated by Equation (3). The main idea of Equation (5) is that when the traffic $x_i$ is less than the traffic mean $\bar{Z}_{2v-i}$ and $Z_n \geq h$, the alarm will be revoked. In addition, in order to revoke an alarm more accurately, the condition $x_i < \rho \bar{Z}_{2v-i}$ can be improved as

$$\sum_{j=0}^{k} \{ x_{i+j} < \rho \bar{Z}_{2v-i-j} \} \geq \theta$$

where $\theta$ is a positive integer and $k > \theta > 1$. Equation (6) describes that when the number satisfies the condition $x_i < \rho \bar{Z}_{2v-i}$ is larger than $\theta$, the alarm will be revoked. At the same time, after revoking the alarm, we also reset $Z_n$ between $0$ and $h$.

4.2. Response the alarms

In order to restrain network anomalies or attacks in real time, MDRA takes defense methods to respond to the alarms. As mentioned earlier, when the CUSUM algorithm detects a change point of the mapping request traffic, MDRA will generate an alarm. Once alarming, MDRA may carry out the following two methods to control network traffic or the further attack.
(a) Cooperating with TRs

For the incoming mapping requests, which aim at resolving a locator for some victim identifier, MDRA can inform each TR who sends the corresponding mapping requests to limit the network traffic until the alarm is revoked. In the case of limiting network traffic, TRs can apply rate limiting or packet-filtering methods based on the intensity of the attacks or pre-defined response policies. Of course, the foregoing response method may need all the TRs and RVs to cooperate with each other in the whole network. In addition, MDRA can use the digital signature technique to guarantee the authenticity of the control information [44,45].

(b) Conditionally answering the mapping requests for the victim’s identifier

For the incoming mapping requests, which aim at resolving a locator for the victim’s identifier, MDRA may employ random resolution or pre-determined threshold algorithms to answer the mapping requests. Furthermore, the main idea of random resolution algorithm is that each RV can invoke a random rule to resolve a locator for the incoming mapping requests, in order to indirectly reduce the attack traffic to the victim’s identifier. As well, the primary consideration for the pre-determined threshold algorithm is that each RV first sets a threshold for the victim’s identifier, once the number of mapping requests for this identifier exceeds the pre-determined threshold value, the RV will not continue to resolve the locator for the victim’s identifier over a period. In this way, the number of attackers will be restricted, and the attack traffic can be cut down indirectly.

To sum up, each of the aforementioned methods has its advantages and disadvantages. On one hand, although the first method can efficiently restrain network anomalies or attacks, it is very difficult to promote cooperation between all RVs and TRs under current network conditions. On the other hand, it is fairly convenient to carry out the second method, but this method may discard some normal or legitimate communications. Hence, we must carefully consider implementing the aforementioned methods in accordance with actual experiences and network circumstances.

4.3. Mapping cache timeout

As stated previously, each TR needs to store the temporary identifier-to-locator mapping items in its mapping cache. Moreover, the TR sets a timer for each identifier-to-locator mapping item and associates each mapping item with a time-to-live (TTL) value, which is equal to the given mapping cache timeout. If an identifier-to-locator mapping item is used by some end host before its timer exceeds its TTL, the timer for this mapping item will be reset to zero. By contrast, if the mapping item is not used by any end host until the timer exceeds its TTL, the corresponding mapping item will be removed from the TR’s mapping cache. In this case, if a new connection using this mapping item needs to be established, the TR will resend a mapping request to the RV for this mapping item. However, the design of the mapping cache timeout has significant impact on the efficiency of MDRA. Indeed, when a packet arrives at the TR, the TR should first lookup a locator for the destination in its cache. If the mapping cache misses, the TR should send a new mapping request to the RV. We show the number of mapping cache misses per 5 min in Figure 5. The traces were collected from a border router in our campus network on 12 October 2008, and this border router has a 1 Gbps link toward ChinaNet and another 1 Gbps link toward China Education and Research Network. Furthermore, there are about 40 000 users in total and over 13 000 active users per day in our campus network [15]. From this figure, we can see that the number of mapping cache misses per 5 min depends on the mapping cache timeout. When the mapping cache timeout is smaller, the number of mapping cache misses per 5 min is larger. Namely, the larger the number of mapping cache misses is, the greater the number of resending mapping requests is. Table I shows the average number of resending mapping requests per 5 min for different mapping cache timeout. From the table, we can see that as mapping cache timeout increases, the mean number of resending mapping requests takes on a nonlinear decrease. However, the number of resending mapping requests may influence the efficiency of MDRA. On the one hand, for the normal communications, the excessive number of resending mapping requests may increase the false positives of MDRA that is because the normal mapping requests may be regarded as an anomaly when the mapping request traffic is too much to generate an alarm. But on the other hand, for the abnormal attacks, it is more effective to detect the anomaly by the resending mapping requests, because the number of the mapping requests increase, and the traffic spike can be greater, resulting in detecting the anomaly more skillfully or easily by MDRA. Nevertheless, the
smaller the mapping cache timeout is, the higher the bandwidth is needed for mapping requests, and the larger TR’s cache memory and overhead are. Hence, the selection of an appropriate mapping cache timeout is of prime importance for the efficiency of MDRA, and we should find a trade-off between them in practice.

4.4. Mapping request threshold algorithm

Because all end hosts under the same CN or TR share the same mapping cache, there is a distinct issue, which may make MDRA more inefficient or powerless. If a great many puppet hosts who belong to the same CN or TR want to launch attacks against the victim in a short period, the TR only needs to send a mapping request to the RV before its mapping cache timeout as shown in Figure 6. Consequently, this network anomaly would not be characterized precisely by the mapping request traffic. In MDRA, we must consider well over the possibility of the aforementioned issue.

In order to decouple the mapping request traffic from the mapping cache, one possible algorithm is to use the network traffic whose destination is the victim. Taken a step further, every time the network traffic from one TR to the victim exceeds a limited value, the TR will send a mapping request for the victim’s locator. Although this algorithm can represent the characteristics of network traffic more profitably, it may impose the additional overhead on the mapping system and can also increase the false positive rate of the detection approach based on mapping requests. Therefore, we provide a mapping request threshold algorithm to solve this issue. When an identifier-to-locator mapping item is live in the TR’s mapping cache, the TR will count up the number of other new end hosts who first lookup this mapping item in a specified time. If this amount exceeds the pre-determined mapping request threshold, which can be defined empirically by real experiments, the TR will immediately inform the corresponding RV of the mapping request amount. The benefit of this algorithm is that it can conspicuously decrease the mapping request times to the RV and simultaneously reduce the burden on the mapping system. The mapping request threshold algorithm is illustrated as follows:

\[
d(y_n) = \begin{cases} 
  y_n & \text{if } y_n \geq m; \\
  0 & \text{otherwise}
\end{cases}
\]  

where \(d(y_n)\) is the mapping request amount, which is sent to the RV by the TR; \(y_n\) is the lookup number of other new end hosts, which is counted up by the TR in the \(n_{th}\) time interval; \(m\) is the pre-determined mapping request threshold in an observation period.

5. PERFORMANCE EVALUATIONS

In this section, we evaluate our approach MDRA in detail by a small-scale DDoS simulation in NS-2, and our main purpose is to discuss the obvious advantages of MDRA. At the same time, in order to evaluate our approach more completely and justly, we also analyze the possible false positive and false negative problems in MDRA. In particular, we cannot use the actual network traffic downloaded from some famous databases [46–48] to evaluate our approach, because networks of locator/identifier separation have not been deployed in today’s Internet yet, and there is no mapping service up till the present moment. Taking a step further, we first illustrate the advantages of MDRA by comparing with the general anomaly detection approach based on network traffic, including alarm in advance, detection efficiency, and traffic control. Next, we analyze the feasibility of the mapping request threshold algorithm and discuss the possibility of generating DDoS attacks against the RVs. At last, we qualitatively discuss the possible false positive and false negative problems caused by some accidental phenomena or factors.

For a start, in order to illustrate the authenticity and universality of our DDoS simulation, we analyze the Cooperative Association for Internet Data Analysis “DDoS Attack 2007” dataset, which describes the end host 71.126.22.64 suffered from a DDoS attack on 4 August 2007. Furthermore, this DDoS attack was launched at the
23rd minute and lasted about 1 hour, and the attack traffic suddenly increased from 200 Kbps to 80 Mbps in few minutes. Figure 7 depicts the network traffic produced by the end host 71.126.22.64. From the figure, we can see that when the DDoS attack starts, the network traffic increases to approximately 170,000 packets per 10 s in intensity. Similarly, we also give the corresponding number of mapping requests for the end host 71.126.22.64 in Figure 8. This figure shows that a strong outburst is also caused by the DDoS attack, and the ultimate number of the mapping requests suddenly reaches approximately 400 and lasts about 5 min. According to the characteristics of this real DDoS attack dataset, we give a correspondingly small-scale DDoS attack, which is launched under the mapping service, and we believe that our simulation can really describe the benefits of MDRA because of the comprehensible characteristics.

In order to simulate a DDoS attack under networks of locator/identifier separation, we first accomplish our network model in NS-2. Literally speaking, we assume there are six PNs in transit core and nine CNs under each of the first five PNs, and each CN takes charge of one TR and one end host. In addition, another end host is regarded as a victim under the sixth PN. We suppose the background traffic comes from 30 legitimate end hosts, and each communication initiated at intervals of 1 s by each legitimate end host lasts 2 s. Meanwhile, the legitimate transmission rate, which is Poisson distributed is 1 Mbps. Notice that, in order to simulate a large traffic spike or surge caused by some legitimate end host, the transmission rate from the legitimate end host at 12 s is set to 5 Mbps. Besides, the DDoS attack from 15 malicious attackers starts at 23 s and lasts 5 s, and the average attack rate of each end host is 1 Mbps. The simulation begins at 5 s and ends at 35 s, and Figure 9 shows the network traffic received by the victim per 0.5 s.

5.1. Alarm in advance

Compared with the general anomaly detection approach based on network traffic, MDRA can generate an alarm in advance. In networks of locator/identifier separation, when an end host wants to establish a connection to another end host, it must first send a mapping request to the RV in order to resolve a locator for the destination host. Therefore, once a DDoS attack or other attacks happen, the change point of mapping request traffic detected by the RV is anterior to that of attack traffic, Figure 10 shows the different arrival time between mapping requests and attack flow belonging to the 15 DDoS attackers in our simulation. Furthermore, the arrival time of the mapping request is gleaned from the RV who maintains the victim’s identifier-to-locator mapping, and the arrival time of attack flow is collected by the victim. As described in Figure 10, the arrival time of the mapping request is shorter than that of attack flow, and the average arrival time of the mapping request and attack flow is at about 24.05 s and at 24.29 s, respectively. So, the average time difference of arrival is 0.24 s, which can be roughly regarded as the alarm delay.
between MDRA and the detection approach based on network traffic. Although the alarm delay in real network environments may be surprisingly small, it is worth believing that this small delay is enough for the RVs to take the precaution to defend the further DDoS attacks.

According to the basic communication procedures of our network model in Section 3, the alarm delay $T_d^u$ can be summarized as follows:

$$T_d^u = T_d^m + T_d^t + T_d^e$$  \hspace{1cm} (8)

where $T_d^m$ is the mapping request delay begins from the time the TR sends a mapping request to the RV and ends with the time the TR receives the corresponding mapping answer from the RV. $T_d^t$ is the packet transmission delay from the TR of the source host to the destination host, and $T_d^e$ is the encapsulation delay. We show the mapping request delay of the 15 DDoS attackers in Figure 11, and the average mapping request delay is 0.19 s in our simulation. From Figure 11, we also observe that the mapping request delay is the major portion of the alarm delay. Of course, the mapping request delay may be different if we design another mapping service, but it will have no influence on our conclusion. That is, the essential characteristics of the identifier-to-locator mapping service can give us a great support to detect anomalies in real time.

5.2. Detection efficiency

Compared with the general detection approach based on network traffic, MDRA is more efficient to identify and diagnose the aberrant network behaviors because the network traffic is complicated and high-dimensional, and the background noises, which consist of various worthless and undesirable network data, have a strong impact on the detection efficiency. Especially, the detection approach based on network traffic generates false alarms frequently. By contrast, the mapping request traffic is simple and single dimensional, and it is extremely convenient to detect network anomalies for the ISPs or other service providers without decreasing the dimension of network traffic. In addition, MDRA can eliminate the influence caused by some large traffic spike or surge, which may consist of some huge data flows sent by one or several legitimate users over the relatively short period. Like DDoS attacks, because burst traffic and high volume are the characteristics of such spike or surge, it is not easy for current techniques to distinguish it merely by statistical characteristics of network traffic.

Figure 12 plots the alarm points in the network traffic received by the victim, and Figure 13 depicts the alarm points in the corresponding mapping request traffic for the victim. Both of them adopt the alarm and revocation algorithm proposed in Section 4.1 to generate and revoke an alarm, that is to say, we analyze the network traffic and the corresponding mapping request traffic by using the same alarm and revocation algorithm. From these two
figures, we can observe that the alarm points caused by the large traffic spike or surge in Figure 12 are not shown in Figure 13, while the alarm points caused by the DDoS attack will certainly show in these figures. At the same time, we also find that the first alarm point caused by the DDoS in Figure 13 is recognizably earlier than that in Figure 12.

By analyzing the authoritative 1998 Defense Advanced Projects Agency Intrusion Detection System evaluation dataset, Figure 14 depicts the alarm points generated by MDRA in the network traffic produced by pascal.eyrie.af.mil. Similarly, Figure 15 shows the corresponding alarm points in mapping request traffic for pascal.eyrie.af.mil. From these two figures, we also can see that the Smurf attack can be detected to generate alarms in either network traffic or mapping request traffic. Furthermore, we also find that in Figure 14 some normal network traffic may be regarded as an abnormal behavior, which does not exist in Figure 15.

To sum up the aforementioned arguments, our approach is more efficient to identify and diagnose network anomalies. In other words, compared with the anomaly detection based on network traffic, our approach can further reduce the false positive rate.

5.3. Traffic control

When network anomalies or attacks have been detected in advance, a powerful real-time response is essential in protecting the victim. In this paper, we have proposed two response methods, and both of them can defend the malicious attacks by controlling the attack traffic. For ease of presentation, we carry out the second method in our simulation. We show the change of the attack traffic under different random resolution values or pre-determined thresholds in Figure 16, and we suppose the time period of the random resolution values or pre-determined thresholds is 5 s. From this figure, we can see that when there is no response, namely, we take no action to prevent the DDoS attack, the attack traffic is the largest. However,
when the random resolution value or pre-determined threshold decreases from 12 to 4, the attack traffic is also significantly reduced. Hence, we can draw a conclusion that our response method can efficiently cut down the attack traffic by conditionally answering the mapping requests. Nevertheless, as previously stated, this method also has its drawbacks. That is, this method may discard some normal or legitimate communications: the smaller the random resolution value or pre-determined threshold is, the larger the negative effect is. In short, we may carefully consider implementing the response methods in accordance with actual experiences and network circumstances.

5.4. Feasibility of mapping request threshold algorithm

The feasibility of mapping request threshold algorithm, we implement another simulation. In this simulation, we assume 20 malicious attackers who belong to the same TR simultaneously launch a DDoS attack at 23 s, and other assumptions are the same with the aforementioned simulation. In addition, we would not simulate the large traffic spike or surge mentioned earlier, because our objective focuses on analyzing the feasibility of mapping request threshold algorithm. Figure 17 illustrates the network traffic received by the victim per 0.5 s.

We show the different number of mapping requests under two circumstances in Figure 18. In the first circumstance, called in general conditions in this figure, we show the number of mapping requests without a mapping request threshold algorithm. In the second circumstance, we use a mapping request threshold algorithm to provide the number of mapping requests for the RV. As shown in Figure 18, when the DDoS attack does not take place, the results under these two circumstances nearly overlap. When the DDoS attack happens, the aggregate number of DDoS mapping requests without and with mapping request threshold algorithm is 1 and 19, respectively. Notice that, in our simulation, the pre-determined mapping request threshold is set to 4 per 0.5 s. Hence, we can conclude that mapping request threshold algorithm is feasible to decouple the mapping request traffic from the mapping cache. In particular, we also find that the alarm in this simulation may delay 0.5 s, which is an observation period because the TR needs this period to count up the lookup number of other new end hosts. Therefore, in order to detect the network anomaly in real time, we should properly adjust the mapping request threshold and the observation period according to the actual experiments and experiences.

5.5. Discussion on distributed denial of service attacks against the resolvers

Like DNS, even if the targets are not DNS servers, the DNS is likely to be affected by DDoS attacks because of the sudden increase of DNS queries [49]. From this point, it is worthwhile to discuss whether the mapping request traffic for some victims may overwhelm the corresponding RV, when a DDoS attack happens. However, the mapping service is distinctly different from the DNS service, because the mapping requests are from all TRs, which are the endpoints of the edge or CNs, while the DNS queries come from all legitimate or malicious end hosts. We can conclude that during the period of a DDoS attack, the mapping request traffic will be significantly smaller than the DNS queries, especially the DDoS attack traffic. The comprehensible causes of this conclusion are listed as follows:

To begin with, from the characteristics of DDoS attacks, the attack traffic will be obviously larger than the mapping request traffic generated by the DDoS attack. In extreme cases, an aggressor requires one mapping request at most, but the volume of useless traffic launched by this aggressor is really immense, such as user datagram protocol flooding.
Second, as illustrated in Section 5.4, although the mapping cache timeout has slight impact on the efficiency of MDRA, it indeed can reduce the mapping request traffic of DDoS attacks. For example in Table I, the number of resending mapping requests is 37,905 on average throughout the whole day when the mapping cache timeout is 1 min. When the mapping cache timeout is 10 min, however, this number reduces to about 14,704 on average throughout the whole day. Namely, the bigger the mapping cache timeout is, the smaller the mapping request traffic of DDoS attacks is.

Finally, our mapping cache threshold algorithm can effectively decrease the mapping request traffic of DDoS attacks without affecting the detection effects of MDRA. In botnets, a great many bots controlled to launch DDoS attacks are generally under the same edge or CN [50,51]. When plenty of puppet machines undertaking by the same TR aim at overwhelming a target, the TR only needs one message to inform the corresponding RV of the mapping request amount in an observation period. For example, assume that 100 end hosts look for some mapping items in the TR’s mapping cache during an observation period, in the worst case, the TR should send 100 mapping requests without mapping cache threshold algorithm, while the TR needs only to send a message to the corresponding RV with mapping cache threshold algorithm.

Nevertheless, when the number of the malicious attackers is large enough, the mapping request traffic still may generate a DDoS attack against the RV, or the real attack target is the RV. In order to defend such attacks, the mapping service can take defense measures proposed in DNS system. For instance, the cache servers can be utilized to improve the resilience against DDoS attacks [52]. In this case, MDRA should be deployed between the RVs and the cache servers, and the RVs must collaborate closely with the cache servers so as to efficiently detect network anomaly. In particular, how to prevent DDoS attacks against the RVs or design a more secure mapping service is outside the research range of this paper, and it can be one of the future research trends and directions.

5.6. False positive and false negative problems

Despite the fact that MDRA is possessed of the obvious advantages, there are still some drawbacks in MDRA, for example, the category of network anomaly detected by MDRA is limited because the mapping request traffic is simple and single dimensional. At the same time, MDRA also contains some possible false positive and false negative problems. In particular, our purpose is not to explore a quantitative study on the false positive or false negative rate due to the actual situations of today’s Internet, but we analyze some accidental phenomenon or factors, which may result in the false positive or false negative problems in our approach, and we believe that these analyses will provide a basis for our further research.

First, some flash events may increase the false positive rate. These flash events, also called flash crowds [53,54], are large surges of legitimate traffic focusing on some specific sites over the relatively short period, such as the flash events during the semifinals of the 1998 World Cup [55]. Obviously, when these flash events happen, the corresponding mapping request traffic is also unstable, sudden, and huge. In general, these flash events are quite similar with DDoS attacks in terms of network anomalies and traffic characteristics. However, no great loss without some gain, these characteristics can also help MDRA to diagnose and identify the flash-crowd attacks [56].

Second, several tactical attacks may escape from the detection of MDRA and increase the false negative rate. That is, some artful attackers can control large surges of puppet hosts to acquire the identifier-to-locator mapping of the victim in different period, and each puppet host maintains the identifier-to-locator mapping item in its TR by a spot of discontinuous communication with the victim, such as the command ping. When the malicious attackers want to launch a general offense to the victim, they can control these puppet hosts to flood the victim in a relatively short amount of time. Of course, these tactical attacks require the attackers to fiddle away a lot of time and effort.

Last but not the least, when the victim and the malicious attackers belong to the same CN or TR, or when many CNs is under one TR, the attack will not be detected because it does not need to send a mapping request to resolve the locator for the victim. Therefore, this instance may also add to the false negative rate.

6. CONCLUSION

This paper aims to propose an anomaly detection and response approach based on mapping requests, and the basic idea behind the proposed approach is very simple. That is, identifying and diagnosing the anomalous network behaviors by judging the mapping requests anomaly. The most significant benefit of our approach is that it can diagnose and respond the aberrant network behaviors in real time. In this paper, we first put forward our network model of locator/identifier separation and illustrate our motivation by examples. And then, we present the detailed design of the proposed approach, including alarm and revocation algorithm, real-time response methods, some considerations for mapping cache timeout, and mapping request threshold algorithm. At last, we evaluate the proposed approach in detail by the DDoS simulations in NS-2. We show that, compared with the general anomaly detection approach based on network traffic, the proposed approach is more advantageous and efficient. In addition, we also discuss the possible false positive and false negative problems for our future research. Because of its salient features and advantages, we believe that this approach is promising.
ACKNOWLEDGEMENTS

This work is supported in part by Major State Basic Research Development Program of China (973 Program) (no. 2013CB329010); National Nature Science Foundation of China (nos. 61232017, 61202428, and 61164012); National High Technology Research and Development Program of China (863 Program) (no. 2012AA041102-03); and the Fundamental Research Funds for the Central University (nos. 2013JBM004, 2013JBM013). The authors should thank the other cooperators in this project for their contributions in this paper. The authors are also grateful to the anonymous referees for their insightful comments and suggestions.

REFERENCES


