Defending Cyber-Physical Systems against DoS Attacks

Abdullah Yasin Nur
Center for Advanced Computer Studies
University of Louisiana at Lafayette
Lafayette, LA 70504
Email: ayasinnur@louisiana.edu

Mehmet Engin Tozal
School of Computing and Informatics
University of Louisiana at Lafayette
Lafayette, LA 70504
Email: metozal@louisiana.edu

***Abstract*—Recent advances in Cyber-Physical Systems (CPSs) promote the Internet as the main communication technology for monitoring, controlling and managing the physical entities as well as exchanging information between the physical entities and human users. On the other hand, the Internet introduces a variety of vulnerabilities that may put the security and privacy of CPSs under risk. The consequences of cyber-attacks to CPSs might be catastrophic because they are usually part of human habitat. One of the most perilous threats in the Internet is the Denial of Service (DoS) attack and its variations such as Distributed DoS (DDoS). In this work-in-progress, we propose a novel probabilistic packet marking scheme to infer forward paths from an attacker to a victim site and delegate the defense to the upstream Internet Service Providers (ISPs). Our results show that the victim site can construct a forward path from the attacker after receiving 23 packets on the average.**

I. Introduction

Cyber-physical systems (CPS) are composite systems formed by interdependent or interacting physical entities. Unlike the traditional physical systems, the entities forming CPSs interact with computational elements that are integrated in and/or separated out. These computational elements are typically responsible for monitoring, controlling and managing the physical entities as well as exchanging information between the physical entities and human users. Examples of CPSs are rail transport systems, traffic monitoring systems, power plants, production facilities and chemical pipeline networks.

Although the communication between the physical entities can be carried over centralized/decentralized private networks, the Internet provides a greater connectivity, control, management and integration opportunities for CPSs. Furthermore, the recent advances in Internet of Things (IoT) elevates the Internet as the main communication medium between the physical entities. On the other hand, the Internet introduces a variety of vulnerabilities that may put the security and privacy of CPSs under risk as well. The consequences of cyber-attacks to CPSs might be catastrophic because they are usually part of human habitat. An unauthorized access to a chemical pipeline network may cause property damages and health risks for humans. An attack to a traffic control system may cause numerous accidents in a city resulting in property damage and casualties.

One of the most perilous threats in the Internet is the Denial of Service (DoS) attack and its variations. The objective of a DoS attack is to exhaust the resources of a system until the system fails to provide its usual services in a timely fashion. Typically, a DoS attack involves flooding a system by legitimate-looking traffic and making the system break down completely, work in less capacity, or fail to serve on time. However once the source of the attack is determined, it is easy to defend the system by blocking the traffic coming from the attacking site. A more severe type of the DoS attack is the Distributed Denial of Service (DDoS) attack where a large number of hosts simultaneously attack a victim site. DDoS attackers plan their attack in advance by compromising multiple hosts that are scattered in the Internet through a common vulnerability. Then, they use all compromised hosts to flood the victim site. Preventing DDoS attacks by filtering the rogue traffic is difficult because it is challenging to distinguish the rogue traffic from the legitimate traffic. Besides, it will induce too much overhead on the traffic filtering mechanism of the target system. Moreover, the attacker may forge attack sites by using IP address spoofing. A better approach to defend a system against DDoS attacks is to delegate the defense to the Internet Service Providers on the path between the attack and victim sites. Therefore, it is necessary to be able to infer the forward path from an attack site to the victim site.

Inferring the path between two hosts from the destination site in the Internet is called the IP traceback. The IP traceback is a challenging problem because it is not directly supported by the IP protocol. A solution to the IP traceback problem is packet marking. Packet marking is blemishing the packets with routers' IP address info while the packet traverses the routers from the source toward the destination. There are two different packet marking schemes; Probabilistic Packet Marking (PPM) [1] and Deterministic Packet Marking (DPM) [3]. In PPM the routers on a path probabilistically infuse their IP address information into the packet. On the other hand, in DPM only the ingress routers on a path marks every packet passing through it with its IP address information. In this work-in-progress we propose a novel packet marking scheme based on probabilistic packet marking.

We modify the record route feature of the IP protocol. Ideally, each router forwarding a packet checks the record route option of the packet. If the record route option is enabled, the router inserts one of its IP addresses into the options field of the packet header. Due to the restrictions regarding the size of the IP packet header [4], at most nine IP addresses can be stored in the options field. Hence, if the options field of the IP packet header is full, the router skips inserting its own IP address.

In our probabilistic scheme, a router always inserts its IP address as long as there is room in the IP options field. On the other hand, if the IP options field is full, a router probabilistically restarts record routing by clearing the options field of the IP packet header. The destination site gradually joins the recorded IP addresses in the options field to construct a forward-paths graph from all sources toward the destination. Basically, the destination site starts with an empty forward-paths graph and fills the graph with subpaths that are reported via record route. After receiving a packet, the destination site aligns the sequence of the IP addresses in the record route with respect to the current snapshot of the graph and implants

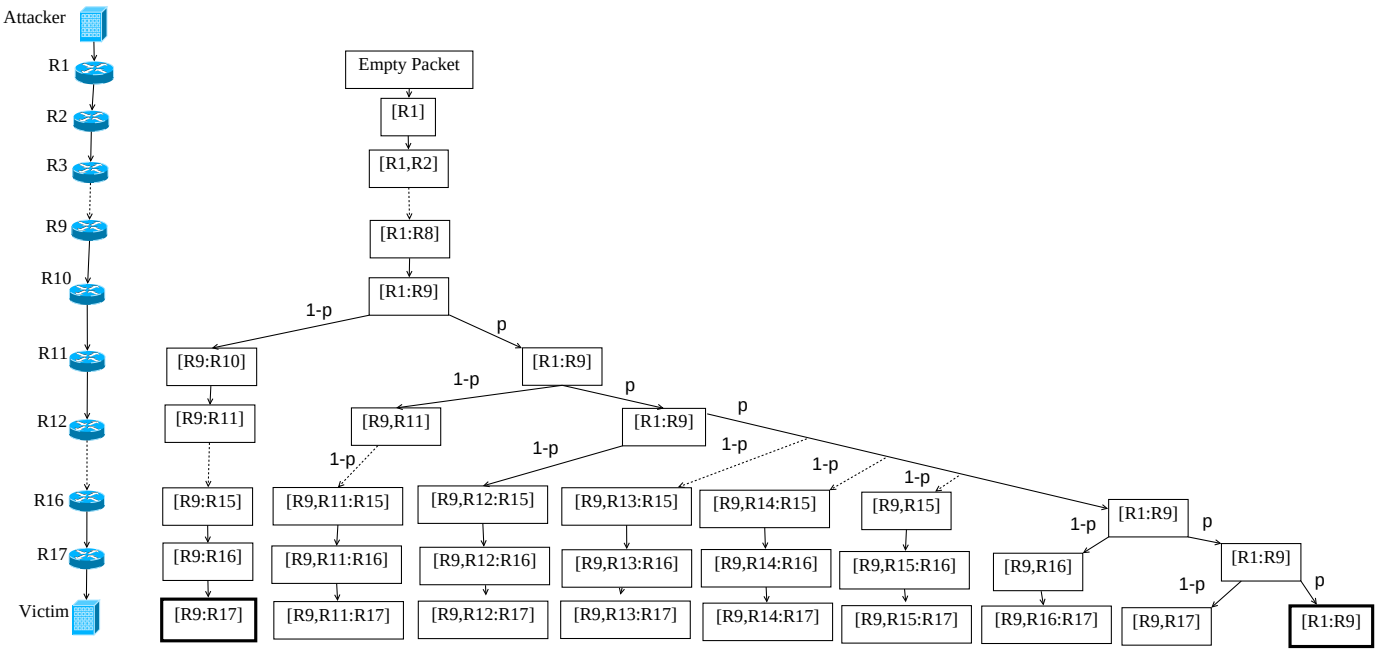

Fig. 1. All possible record route options field for an example path between the attacker and the victim sites

the new route in the graph. Our experimental results show that the proposed method is able to construct a single forward path by using 23 packets on the average. Our approach constructs the forward paths toward a target site with low overhead to defend the system against the variants of DoS attacks.

II. Methodology

In this study, we exploited the record route feature of the IP protocol and the probabilistic packet marking (PPM) technique with modifications. IP address protocol provides record route option to record the route of an internet packet. An IP packet with record route option enabled solicits each router that handles the packet to append one of its IP address into the options field. Hereby, the destination host can get the list of the routers appeared on the forward path. Note that the maximum number of IP addresses that could be stored in the IP options field is nine [4]. Therefore, the destination host is expected to receive the IP addresses of the first nine hops even if the path between the two hosts is longer than nine.

Probabilistic packet marking schemes involve the routers that handle an IP packet to embed information into the packet. Typically the embedded information reflects the identity of the router or the link that the packet passes through. Our scheme requires a router to probabilistically modify the options field of an IP packet. Similar to the IP record route, a router appends its IP address into the options field of a packet when there is room. Different from the IP record route, a router rewrites the options field of a packet with probability 1-p when the packet is full. Equivalently, the router skips rewriting the options field of a full packet with probability p. A router rewriting the packet, swaps the first IP address in the options field by the last IP address and erases the rest of the options field. Then, it appends its own IP address in the options field.

Algorithm 1 presents the rewrite pseudocode. The algorithm expects the record route options field of an IP packet as a vector of IP addresses as well as the IP address of the current router that handles packet. It returns the modified record route options vector of the IP packet. At line 1, the algorithm stores the last IP address in the record route options vector into a temporary variable. It erases the elements of the vector at line 2. Finally, it appends the last IP address stored in the temporary variable and the IP address of the current router at third and fourth lines.

Algorithm 1 Rewrite Algorithm

Input: $RROptions$ ▷ IP record route options field
Input: IP ▷ Current router IP address
Output: $RROptions$ ▷ updated IP record route options field

1: $temp = RROptions.get(9)$
2: $RROptions.erase()$ ▷ erase all IP addresses
3: $RROptions.append(temp)$
4: $RROptions.append(IP)$

Figure 1 shows an example path consisting of 17 routers between an attacker and a victim host. In accordance with our rewriting scheme, the first nine routers append their IP addresses in the record route options field of the packet, $[R_1, R_2, \ldots, R_8, R_9]$. The tenth router on the path rewrites the packet with probability 1-p or skips rewriting with probability p. In case it rewrites the packet, the eleventh router receives $[R_9, R_{10}]$ in the record route options field and appends its IP address with probability 1 since there is room in the field. As a result, the next router receives $[R_9, R_{10}, R_{11}]$ in the options field. On the other hand, if the tenth router does not rewrite the packet the eleventh router receives $[R_1, R_2, \ldots, R_8, R_9]$ in the record route options field. Again, the eleventh router rewrites the packet with probability (1-p) or skips rewriting with probability p. If it rewrites then the next router receives $[R_9, R_{11}]$ in the record route options field. Otherwise, the next router receives $[R_1, R_2, \ldots, R_8, R_9]$. This procedure repeats itself until the packet reaches to the victim site.

The tree data structure in Figure 1 demonstrates all possible record route options field for a path between the attacker and the victim sites. Each level in the tree represents packets that might be received by the corresponding router. The arrow(s) between the levels of the tree represent the probabilistic decisions that might be made by a router before forwarding

the packet to the next router. Note that each arrow is labelled by its decision probability except the for sure cases where the probability is one. Finally, the leaves of the tree represent all possible record route options fields that the victim site may receive.

The packets marked by thick lines in the tree represent the *essential* record route IP lists (essentialRR) that the victim needs to receive to rebuild the forward path correctly. The record route IP lists other than the *essentialRRs* miss one or more intermediate routers on the forward path. On the other hand, *essentialRRs* hold a complete sequence of routers on the forward path with no missing intermediate router. Therefore it is necessary for the victim site to receive all *essentialRRs* and concatenate them in the right order to build the forward path. Note that the last IP address in an *essentialRR* is the first IP address in the next *essentialRR*.

III. Reconstruction of Forward Paths

The victim site gradually joins the recorded IP addresses in the options field to construct a forward-paths graph from all sources. The victim site starts with an empty forward-paths graph and fills the graph with subpaths that are reported via record route. After receiving a packet, the victim site aligns the sequence of the recorded IP addresses with respect to the current snapshot of the graph and implants the new route in the graph. In order to form a complete forward path from a source, the victim site needs to get at least one instance of each *essentialRR*. In addition, the victim site exploits non-*essentialRRs* to form partial paths which miss one or more intermediate routers on the forward path.

When the victim site receives a new packet, it implants the reported IP addresses in the record route options field one by one. In case a reported IP address already exists in the forward-paths graph, it is skipped. Otherwise the IP address is inserted in the forward-paths graph including a link connecting the IP address to its predecessor in the record route options field. Note that the first IP address is processed without creating a new link because it does not have a predecessor.

Then our algorithm compares the newly constructed path, $\mathcal{P}_n$, with the existing paths, $\mathcal{P}$, between the first and the last IP addresses in the record route options field of the received packet. Let *addr($\mathcal{P}_i$)* be the set of the IP addresses of path $\mathcal{P}_i$. In case $addr(\mathcal{P}_n) \subset addr(\mathcal{P}_i)$ where $\exists \mathcal{P}_i \in \mathcal{P}$ then $\mathcal{P}_n$ is a partial path and it is removed from the graph. In case $addr(\mathcal{P}_n) \supset addr(\mathcal{P}_i)$ where $\exists \mathcal{P}_i \in \mathcal{P}$ then $\mathcal{P}_i$ is a partial path and it is removed from the graph. In other cases, all paths are preserved because the new path is a load balancing case of an existing forward path.

IV. Experimental results

In this section we demonstrate the efficiency of our algorithm using a real world dataset. Specifically, we used the iPlane interface-level atlas dataset [2] consisting of more than 15 million (15,865,206) path traces collected from multiple vantage points. Note that we only included the loop-free path traces that reach to their specified destinations. The minimum and maximum hop lengths in our dataset are one and 31, respectively. The average hop length is 16.23 and the distribution is symmetric-like. We randomly pick a pair of IP addresses in the topology and run our Matlab emulation procedure which mimics the methodology given in Section II. In addition, we gradually construct the forward-paths graph from the attack IP address toward the victim IP address.

According to our preliminary results a victim site can construct the forward path from an attacker site after receiving

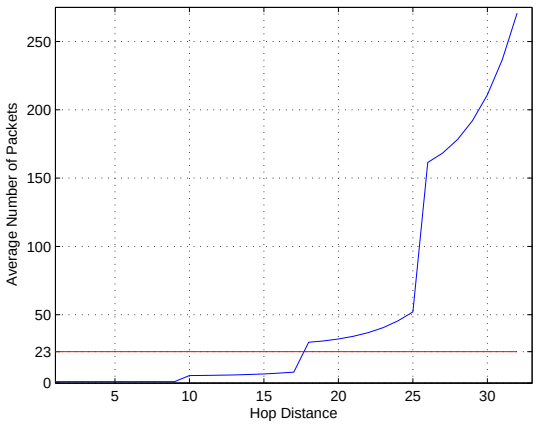

Fig. 2. Average number of packets needed to construct a forward path per hop distance

23 packets on the average. However, the number of packets needed to build a forward path depends on the hop distance between the attack and the victim sites. Figure 2 show the average number of packets needed to construct a forward path with respect to the hop distances between the attack and victim sites. Obviously, the number of packets needed to build a forward path increases non-linearly on the average as the distance increases. Particularly, it reaches to 236 at hop distance 31. Considering that the diameter of the Internet is around 31 in our iPlane snapshot, the required packet overhead of our approach is still acceptable. Moreover, the figure demonstrates jumps at hop distance 9, 17 and 25. Because the number of the required *essentialRRs* increases after the boundaries 9, 17 and 25 meaning that more packets are needed to construct a forward path.

We are working on the DDoS experiment for the same scenario. The 23 packets on the average in DoS gives us an upper bound for the DDoS scenario because the partial paths toward the victim site can be constructed using the packets from multiple attack sites.

V. Conclusions

Recent advances in Cyber-Physical Systems (CPSs) promote the Internet as the main communication technology for monitoring, controlling and managing the physical entities as well as exchanging information between the physical entities and human users. On the other hand, the Internet introduces a variety of vulnerabilities that may put the security and privacy of CPSs under risk. One of the most perilous threats in the Internet is the Denial of Service (DoS) attack and its variations. In this work-in-progress, we propose a novel probabilistic packet marking scheme to infer forward paths from the attacker to the victim site and delegate the defense to the upstream Internet Service Providers (ISPs). Our results show that a victim site can construct the forward path from an attacker site after receiving 23 packets on the average. We are working on a mathematical model to represent our approach and analyse its theoretical implications.

References

[1] S. Savage, D. Wetherall, A. Karlin, T. Anderson, "Network support for IP traceback," in IEEE/ACM ToN, vol. 9, issue 3, pp. 226-237, 2001.
[2] "iPlane project" http://iplane.cs.washington.edu/
[3] A. Belenky, N. Ansari, "IP traceback with deterministic packet marking" IEEE Communications Letters, vol. 7, no. 4, pp. 162164, April 2003.
[4] "RFC 791" https://tools.ietf.org/html/rfc791