EasyTrace: an Easily-Deployable Light-Weight IP Traceback on an AS-Level Overlay Network

Hongcheng Tian, Jun Bi, Wei Zhang and Xiaoke Jiang
Network Research Center, Tsinghua University
Department of Computer Science, Tsinghua University
Tsinghua National Laboratory for Information Science and Technology (TNList)
{tianhc, zw, justok}@netarchlab.tsinghua.edu.cn, junbi@tsinghua.edu.cn

***Abstract*—IP traceback can be used to find the origins and paths of attacking traffic. However, so far, no Internet-level IP traceback system has ever been deployed because of deployment difficulties. In this paper, we present an easily-deployable light-weight IP traceback based on flow (EasyTrace). In EasyTrace, it is not necessary to deploy any dedicated traceback software and hardware at routers, and an AS-level overlay network is built for incremental deployment. We theoretically analyze the quantitative relation among the probability that a flow is successfully traced back various AS-level hop number, independently sampling probability, and the number of packets that the flow comprises.**

I. INTRODUCTION

Distributed Denial of Service (DDoS) attacks continue to pose major threats to the Internet. Attackers can launch attacking traffic from various locations in the Internet to exhaust the bandwidth or computing resources at the victim. Attackers often forge source addresses to escape detection, such as SYN flooding, DNS amplification, SMURF, etc.

The objective of IP traceback [1] is to find the origins and attacking paths of malicious traffic. But most IP traceback approaches are difficult to be deployed in the Internet, because dedicated software or hardware needs to be deployed at routers, or they are not able to be incrementally deployed. As far as the authors know, there is no Internet-level IP traceback system that is currently deployed.

In this paper, we propose an easily-deployable light-weight IP traceback system that is based on flow (EasyTrace). EasyTrace uses existing xFlow (sFlow, NetFlow and IPFIX) function and BGP information to implement traceback, instead of deploying any traceback software or hardware at routers. EasyTrace builds an AS-level overlay network among deployed ASs by the upstream logical neighbor discovering in order to support incremental deployment. It should be emphasized that, EasyTrace can confirm the ingress interface(s) of the BGP router(s) through which some attacking flows enter the deployed AS. Theoretical analyses show that the probability that a flow is successfully traced back various AS-level hop number quantitatively depends on two factors: independently sampling probability, and the packet number that the flow comprises.

EasyTrace has three deployment incentives: (1) Deployed AS can supply the traceback service to other ASs, end-users or intrusion detection systems as a charged service; (2) For an deployed stub AS, when its users are attacked, EasyTrace can identify which interfaces some attacking traffic enters the stub AS from, and then actions at the interfaces (such as packet filtering or traffic constraint) can be taken to protect its users; (3) If a transit AS can supply more services, such as a traceback service, it is more attractive to potential customer ASs.

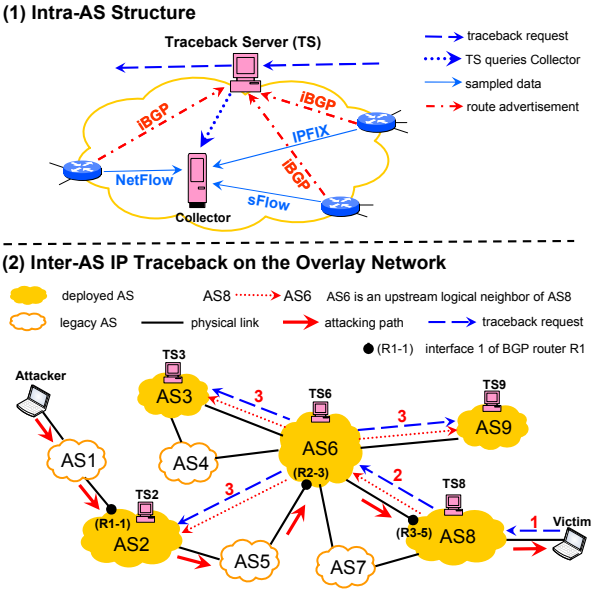

Fig. 1. Light-Weight IP Traceback (EasyTrace)

II. LIGHT-WEIGHT IP TRACEBACK

A. Assumption and Definition

We identify one assumption: Every deployed AS registers and opens its ASN and IP address of its Traceback Server (TS) on a special website. The registration incentive is that the traceback can be supplied to other ASs, end-users or intrusion detection systems as a charged service. Thus, every TS knows all deployed ASs and IP addresses of their respective TSs.

Given any two deployed ASs, *ASi* and *ASj*, if there exists a route from *ASi* to *ASj* without transiting any other deployed ASs, *ASj* is referred to as a *downstream logical neighbor* of *ASi*. And *ASi* is referred to as an *upstream logical neighbor* of *ASj*.

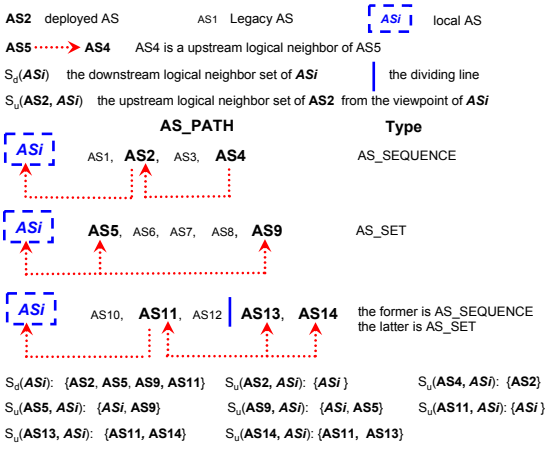

Fig. 2. An example of pre-processing in *TSi* (the Traceback Server of *ASi*).

B. Intra-AS Structure

Fig. 1 shows intra-AS structure in EasyTrace. The BGP routers enable xFlow function on interfaces to other ASs, to sample ingress traffic and send sampled traffic data to Collector which uniformly processes xFlow data to serve traceback. Each deployed AS has a TS in function, which communicates with TSs of other ASs (or end-users, etc.) and Collector of the local AS. TS runs BGP protocol and learns iBGP routes from iBGP peers, but never advertises any route. TS logs its entire historical best routes into a Historical Route Information Base (HRIB).

C. Building an AS-Level Overlay Network

The overlay network enables every TS to know who the upstream logical neighbors of its local AS are. Thus, the attacking flow can be traced back over hop-by-hop upstream logical neighbor. In the following portion, assuming that *ASi* is any one deployed AS.

1) *Pre-processing in each Traceback Server (TS)*

The AS_PATH implies the upstream and downstream logical neighbor relations between the deployed ASs. By searching all AS_PATHs of its HRIB, *TSi* maintains two types of sets: the downstream logical neighbor set of *ASi* (denoted by $S_d(ASi)$), and the upstream logical neighbor set of each deployed AS which appears in the AS_PATHs in its HRIB from the viewpoint of *ASi*. Note that *TSi* knows all deployed ASs, and AS_PATH is composed of one or more path segments [2]. Every path segment may be of type AS_SEQUENCE or AS_SET. In general, there are three typical combinations of path segments for AS_PATH (listed in Fig.2). Fig. 2 shows an example of the pre-processing.

2) *Upstream Logical Neighbor Discovering*

By II.C.1, *TSi* maintains the downstream logical neighbor set of *ASi*, $S_d(ASi)$. *TSi* sends a query request to the TS of every member in $S_d(ASi)$, asking who the upstream logical neighbors of *ASi* are. Assuming that *ASj* is any one member in $S_d(ASi)$. *TSj* of *ASj* will receive the query request from *TSi*. By II.C.1, *TSj* maintains the upstream logical neighbor set of *ASi* from the viewpoint of *ASj*, $S_u(ASi, ASj)$. *TSj* will respond to *TSi* with $S_u(ASi, ASj)$. When *TSi* receives responses from the TSs of all members in $S_d(ASi)$, *TSi* will get the upstream logical neighbor set of *ASi*, $S_u(ASi)$.

D. Sampling and Logging Attacking Flows

Traffic data, generated and sent to Collector by sFlow and NetFlow/IPFIX, is different in granularity, the former is packet-level while the latter is flow-level. Collector will simulate the processing of NetFlow/IPFIX at routers and aggregate sFlow data to flow-level data. Thus, the content of flow records stored in Collector includes *IP address of the router, ingress interface index of the router*, *source/destination address*, *source/destination port* and *protocol*.

E. Traceback on the Overlay Network

Fig. 1 illustrates the traceback process. If AS8 has sampled the attacking flow, the flow record stored in the Collector of AS8 can prove that R3-5 forwards the attacking flow. Furthermore, AS8 knows it directly connects to AS6 through R3-5. So AS6 can be identified to be in the attacking path. Similarly, if AS2, AS6 and AS8 have all sampled the attacking flow, the reconstructed path is [AS1, AS2-R1-1, AS5, AS6-R2-3, AS8-R3-5].

III. Evaluation

Assume that a m-packet flow transits r deployed ASs ($r \geq 1$) along the same path and m packets are independently sampled in turn with probabilities $p_r, \ldots, p_2, p_1$, respectively. Then for this flow, the probability that EasyTrace can successfully trace back the flow h AS-level hops ($1 \leq h \leq r$, is sum of deployed ASs) from destination address to source address of the flow is

$$\prod_{k=1}^{h}[1-(1-p_k)^m] \qquad (1 \leq h \leq r) \qquad (1)$$

IV. Conclusion

A light-weight IP traceback approach is proposed to probabilistically find the origin ASs and the paths of attacking flows. Deployment difficulties are the most challenging for most traceback approaches, but EasyTrace, based on existing protocol and functions (such as BGP, sFlow, NetFlow and IPFIX), easily solves the difficulties not only in intra-AS deployment but also in inter-AS incremental deployment.

Acknowledgment

Supported by National Science Foundation of China under Grant 61073172, Program for New Century Excellent Talents in University, Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant 200800030034.

References

[1] S. Savage, D.Wetherall, A. R. Karlin, and T. Anderson, Practical network support for IP traceback, in Proceeding of ACM SIGCOMM, 2000.

[2] Y. Rekhter, T. Li and S. Hares, A Border Gateway Protocol 4 (BGP-4), RFC 4271, Jan. 2006.