Non-Intrusive IP Traceback for DDoS Attacks

Vrizlynn L. L. Thing
vlt@doc.ic.ac.uk

Morris Sloman
mss@doc.ic.ac.uk

Naranker Dulay
nd@doc.ic.ac.uk

Imperial College London
180 Queen’s Gate, London, SW72HR

ABSTRACT

The paper describes a Non-Intrusive IP traceback scheme which uses sampled traffic under non-attack conditions to build and maintains caches of the valid source addresses transiting network routers. Under attack conditions, route anomalies are detected by determining which routers have been used for unknown source addresses, in order to construct the attack graph. Results of simulation studies are presented. Our approach does not require changes to the Internet routers or protocols. Precise information regarding the attack is not required allowing a wide variety of DDoS attack detection techniques to be used. Our algorithm is simple and efficient, allowing for a fast traceback and the scheme is scalable due to the distribution of processing workload.

Keywords

Distributed Denial of Service Attacks, IP Traceback

1. INTRODUCTION

Attackers use spoofed source addresses to hide their identity and location in Distributed Denial of Service (DDoS) attacks [1]. Some service providers do perform ingress filtering to check for valid source IP addresses coming into access routers, but this is not completely effective. Recent studies show source address spoofing is still a major network problem [2], [3]. Traceback mechanisms [4-6] trace the true source of the attackers to stop the attack at the point nearest to its source to reduce waste of network resources and to find the attackers’ identities.

Packets belonging to a particular source-destination pair typically follow a relatively static path through the network as routing tables are not updated very frequently under normal conditions. When an attacker spoofs a legitimate source address, the packet may pass through routers which are not on the normal source-destination routing path and this anomaly can be used to determine the attack path. Based on this rationale, we propose a Non-Intrusive IP traceback scheme. Our scheme builds and maintains caches of valid source addresses for routers in the network from sampled traffic under non-attack conditions. Under attack conditions, we determine which routers have been used for unknown source addresses, to construct the attack graph within an administrative domain. The strengths of this scheme are its scalability due to the distribution of processing workload and speed due to the simple computation for the attack graph construction. There is no need to modify existing routers, victim or internet protocols to support the traceback, so it is “non-intrusive”unlike many other existing techniques [4-6]. This scheme supports the tracing of both internal (e.g. zombies within the victim network) and external attackers.

2. KEY ASSUMPTION

Our design makes the key assumption that end-to-end routes are relatively stable as indicated by analysis of 40000 end-to-end routes between 37 Internet sites, in [7]. Prevalence of a dominant route (i.e. the route that appears most often) is computed as the ratio of the number of times the dominant route is observed to the total number of traceroutes measuring a particular path. The median value of prevalence is 82%, 97% and 100% at host, city and autonomous system granularity respectively. This indicated that Internet paths were strongly dominated by a single route. Although the time periods over which routes persisted demonstrated a wide variation, ranging from seconds to days, about 2/3 of the Internet paths had routes persisting for either days or weeks.

Routing stability based on data captured from the National Internet Measurement Infrastructure (NIMI) and a set of 189 public traceroute servers was studied in [8]. Of the NIMI paths, 78% always exhibited the same route, and 86% of the routes had a prevalence of 90% or higher. For the public servers, the corresponding figures are 73% and 85% respectively. It was also shown that routes often persist for at least a day. In general, 1/3 of the Internet routes and 1/6 of the NIMI routes are short-lived.

A study on routing fluctuations [9] concluded that the vast majority of Internet routing instability stems from only a small number of unpopular destinations. Popular destinations, which are responsible for the bulk of the Internet, have remarkably stable routes lasting days or weeks, probably due to the fact that they have reliable and well-managed connections to the Internet.

The above studies showed that the Internet routes exhibit relatively high stability so our approach to caching routing information in white lists should not result in very frequent and erratic changes to the lists.

3. NON-INTRUSIVE TRACEBACK

If node A spoofs node B’s address to send traffic to node C, an “incorrect” path (or anomalous intermediate routers) can be detected. The routers on the A-C path will suddenly ”see” B’s source address rather than the routers on the valid B-C path. By performing source IP address validation checks on whether transit packets are supposed to arrive at particular routers, these packets could be identified as from legitimate or illegitimate users, with a

ASIACCS’07, March 20–22, 2007, Singapore.

low false positive rate. Therefore, even seemingly legitimate packets, used in attacks would still be traceable.

In our scheme, network routers use standard flow sampling and reporting mechanisms such as Netflow [10] and IPFIX [11, 12], to update their assigned White List (WL) caching device. The required fields of the flow include the source and destination addresses from the original data packets. Each cached record consists of the above fields, the address of the router that sent the data and the time of receipt to expire the record.

The WL caching devices will update the white lists for the routers during the learning stage, i.e. only when there is no ongoing DDoS attack, to prevent spoofed source addresses from being included in the caches. We assume a DDoS attack would be detected using mechanisms such as TCP SYN flood [13]. During the attack, traffic sampling from the routers is still sent to the WL caches, but the white list generation and updates are suspended upon attack detection. The WL caching devices search for mismatches between the sampled traffic and cache data. These anomalies are sent to the Traceback Manager to generate the attack graphs.

One of the main goals of traceback is to locate the points closest to the attack sources in order to mitigate the attack by effective filtering or rate-limiting. Therefore, instead of having coverage of all routers within a domain such as a campus network, it is suffice to perform monitoring at strategic points such as nodes in the network where incoming and outgoing traffic will definitely traverse. To pin-point the strategic points, we classify attackers into internal (e.g. zombies within the victim network) or external. Ingress routers are the strategic points to perform monitoring or traffic sampling to trace external attackers. However, for the internal attackers, we have to know the network topology to perform monitoring on the routers one hop away from the victim. By reducing the number of routers participating in the traffic sampling and flow exporting, the workload and overhead traffic is significantly reduced. This is a very important enhancement considering that traceback is to be performed during the occurrence of a DDoS attack when the victim's network is under heavy load. Another advantage of this scheme is due to the small number of routers involved, a single Traceback Manager with built-in WL caching functionality could be in charge of the whole network, therefore consolidating the information storage and processing at a central point. This would allow faster processing and a global view of the traffic flows in the domain, making it easier to identify anomalous flows.

4. DEPLOYMENT CONSIDERATIONS

Our traceback approach is non-intrusive, in that it is not necessary to make any changes to the routers assisting in the traceback process. Built-in traffic sampling/monitoring and exporting tools in routers could be used to sample and report the required information to the WL caching devices. If such tools are not built in the routers, we can instead make use of monitoring devices by installing them along the network paths.

An important issue is when to suspend the learning process in order to prevent records of the attack traffic flow being included in the white list, thereby corrupting it. The DDoS attack detection mechanism triggers traceback and stops the learning process. As there will always be a finite delay in detecting an attack, the records of sampled traffic are first written in to a whitelist buffer. The interval for the buffer to confirm entries into the white list cache depends on the attack detection speed. For example, if the attack detection mechanism takes *x* secs and the time to inform the Traceback Manager of the attack takes *y* secs, the buffer flushing interval would be *x*+*y* secs.

In order to estimate white list size, we referred to [14] which shows that Amazon.com experienced 630,000 visitors in a single hour on its busiest day in 2003. By having a white list cache for a protected server in an IPv4 network, each record would need 4 bytes for the source address, 4 for the router and 8 for the tick counts in milliseconds = 16 bytes total which implies 4.8 MB needed to store 30 minutes of white list records.

5. SIMULATIONS

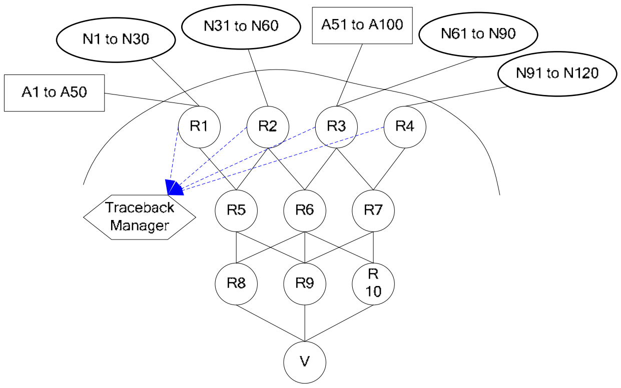

Figure 1: Simulation scenario

We have carried out simulations in ns-2. During the learning phase, nodes generate legitimate traffic to the target/victim and the Traceback Manager builds the white list. When the attack traffic is started, the white list updating is suspended and traceback is started, but the legitimate nodes continue to generate new or existing flows' traffic at a probability (to simulate random traffic).

Figure 1 shows 100 attackers and 120 legitimate nodes. The attackers send attack traffic with randomly spoofed addresses in the range of 1 to 10000 (which includes the addresses of legitimate nodes). The strategic points are R1, R2, R3 and R4, which are the entry points to the network. The links from the legitimate nodes and the attackers into the network are set to 10Mbps with a propagation delay of 30ms to reflect the Internet delays. The internal links are set to 100Mbps with a propagation delay of 10ms. During the learning phase, each of the legitimate nodes, N1 to N100, sent traffic to the victim V at the rate of 5 pkts/sec. R1 to R4 sampled traffic at a probability of 0.01 and sent them to the Traceback Manager. The learning period was set to 20 secs. We ran 3 sets of simulations and the attacks were started at the 20th sec with rates of 20, 50 or 100 pkts/sec, per attack node. During the attack, all the legitimate nodes (including N101 to N120 which were simulating new legitimate requests) generated traffic with a “decide to send” probability[1] of 0.5 at a rate of 5 pkts/sec per node. The attack lasted for 1.5 secs.

R1 and R3 were successfully detected. Table 1 shows the statistics collected, of the number of mismatch packets detected. The time is from the start of the attack and the results are displayed as R*X*(*Y*), where *X* refers to the router’s ID and *Y* refers to the number of mismatch packets detected. The time, *t*, taken to first detect mismatch packets for both R1 and R3, was 140ms, 80ms and 70ms for attack rates of 20, 50 and 100 pkts/sec, respectively. At *t* ms, a total of 3, 3 and 4 sampled packets were received by the Traceback Manager, of which 2, 2 and 3 were mismatch packets, for the attack rates of 20, 50 and 100 pkts/sec, respectively.

Table 1: Mismatched packets

Attack Rate pkts/sec	*t ms*	0.5 sec	1 sec	1.5 sec (attack stopped)

20	R1(1)
	R3(1)
		R1(5) R3(4)	R1(11)
			R3(13)
				R1(15)
				R3(17) R4(1)

50	R1(1)
	R3(1)
		R1(15)
		R3(10) R4(1)
			R1(24)
			R3(26) R4(1)
				R1(34)
				R3(39) R4(2)

100	R1(1)
	R3(2)
		R1(23)
		R3(24)
			R1(43)
			R3(56)
				R1(69)
				R3(91) R2(1)

The results show false positives were detected. R2 (for attack rate of 100 pkts/sec) and R4 (for attack rates of 20 and 50 pkts/sec) were detected for mismatched packets. This is due to sampling from the new legitimate traffic not found in the white list. We also observe that as time progresses, false positives started appearing. However, the difference between the number of mismatch packets sampled for R1,R3 and R2,R4 widens too. At 0.5 sec, the smallest-gap ratio (worst case) was 1/10. At 1.5 sec, the smallest-gap ratio was 1/15, 1/17 and 1/69 for attack rates of 20, 50 and 100 pkts/sec. Therefore, threshold values can be safely set so that false positives are ignored in a real-world scenario.

6. CONCLUSIONS

We have implemented a non-intrusive traceback technique based on the rationale that packets relating to a particular source-destination flow follow a relatively static path through routers. If an attacker spoofs a legitimate user’s address, an “incorrect” path can be detected.

Simulations conducted showed routers forwarding attack packets were successfully traced. We achieved detection rate of 140ms, 80ms and 70ms for attack rate of 20, 50 and 100 pkts/sec. We observed that as the attack rate increases, the detection is faster and the difference in the number of mismatch packets from attack and new legitimate traffic increases. This allows a threshold to be set to ignore false positives.

Due to the differences in the way our system and the other existing traceback techniques are triggered, quantitative analysis and comparison are not practical. However, we presented a qualitative analysis comparing our scheme with other traceback techniques. Our approach is non-intrusive, not requiring any changes to be made to the Internet routers and precise information regarding the attack is not required so we can use a wide variety of DDoS attack detection techniques. The logging and computation tasks are shifted to the WL caching devices and Traceback Manager, and therefore relieving the victim from additional burden. Changes to the original data packets are also not required. As the learning phase is conducted before the attack, once the attack is detected, mismatch checking can be conducted at once to determine routers carrying attack traffic. Our algorithm is also simple and efficient, allowing for a fast generation of the attack graph and is scalable due to the distribution of processing workload.

7. ACKNOWLEDGEMENTS

We gratefully acknowledge the support from the EU funded Diadem Distributed Firewall FP6 IST-2002-002154.

8. REFERENCES

1. K. J. Houle and G. M. Weaver, "Trends in Denial of Service Attack Technology", Oct. 2001, CERT Coordination Center, http://www.cert.org/archive/pdf/DoS_trends.pdf.
2. Robert Beverly and Steven Bauer, "The Spoofer Project: Inferring the Extent of Source Address Filtering on the Internet", USENIX SRUTI: Steps to Reducing Unwanted Traffic on the Internet Workshop, Jul. 2005.
3. David Moore, et al., "Inferring Internet Denial-of-Service Activity", ACM Transactions on Computer System (TOCS), May 2006, **24**(2), pp. 115-139.
4. Alex C. Snoeren, et al., "Hash-Based IP Traceback", ACM Sigcomm, Aug. 2001.
5. Steve Bellovin, Marcus Leech, and Tom Taylor, "ICMP Traceback Messages", IETF Internet Draft, Version 4, Feb. 2003 (Work in progress).
6. Stefan Savage, et al., "Practical Network Support for IP Traceback", ACM Sigcomm, Aug. 2000.
7. Vern Paxson, "End-to-end routing behavior in the Internet", ACM Sigcomm, Aug. 1996.
8. Yin Zhang, Vern Paxson, and Scott Shenker, "The Stationarity of Internet Path Properties: Routing, Loss, and Throughput", ACIRI Technical Report, 2000.
9. Jennifer Rexford, et al., "BGP Routing Stability of Popular Destinations", ACM SIGCOMM IMW (Internet Measurement Workshop), Nov. 2002.
10. Cisco IOS Netflow, "http://www.cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html".
11. B. Claise, "IPFIX Protocol Specification", IETF Internet Draft, Version 19, Sept. 2005 (Work in progress).
12. G. Sadasivan, et al., "Architecture for IP Flow Information Export", IETF Internet Draft, Version 9, Aug. 2005 (Work in progress).
13. Haining Wang, Danlu Zhang, and Kang G. Shin, "Detecting SYN flooding attacks", IEEE INFOCOMM, 2002.
14. Keith Regan, "Holiday E-Tail Sales Set Records Despite Performance Woes", E-Commerce Times, http://www.ecommercetimes.com/story/32491.html, Dec. 2003.

[1] Legitimate traffic during attack is generated at 5 pkts/sec. However, a random generator is used to determine whether to generate each packet, with a probability of 0.5.