A Little Background On Trace Back

Two network tracing problems are currently being studied: *IP traceback* and *traceback across stepping-stones* (*or a connection chain*). IP traceback is to identify the origins of sequences IP packets (e.g., identify the origin of DDOS packets) when the source IP addresses of these packets are spoofed. IP traceback is usually performed at the network layer, with the help of routers and gateways. Traceback across stepping-stones is to identify the origin of an attacker through a chain of connections (e.g., connections established with *telnet*, *rlogin*, or *ssh*), which an attacker may use to hide his/her true origin when he/she interacts with a victim host. Traceback across stepping-stones is beyond the network layer, since at each intermediate host the data is transmitted to application layer in one connection, and then resent to the network in the next connection. The problem we propose to address is the latter one.

Research on IP trace back has been rather active since the late 1999 DDOS attacks [1,2,3]. Several approaches have been proposed to trace IP packets to their origins. The IP marking approaches enable routers to probabilistically mark packets with partial path information and try to reconstruct the complete path from the packets that contain the marking [4,5,6]. DECIDUOUS uses IPSec security associations and authentication headers to deploy secure authentication tunnels dynamically and trace back to the attacks' origins [7,8]. ICMP traceback (iTrace) proposes to introduce a new message "ICMP trace back" (or an iTrace message) so that routers can generate iTrace messages to help the victim or its upstream ISP to identify the source of spoofed IP packets [9]. An intention-driven iTrace is also introduced to reduce unnecessary iTrace messages and thus improve the performance of iTrace systems [10]. An algebraic approach is proposed to transform the IP traceback problem into a polynomial reconstruction problem, and uses techniques from algebraic coding theory to recover the true origin of spoofed IP packets [11]. An IP overlay network named CenterTrack selectively reroutes interesting IP packets directly from edge routers to special tracing routers, which can easily determine the ingress edge router by observing from which tunnel the packets arrive [12]. A Source Path Isolation Engine (SPIE) has been developed; it stores the message digests of recently received IP packets and can reconstruct the attack paths of given spoofed IP packets [13,14]. There are other techniques and issues related to IP traceback (e.g., approximate traceback [15], legal and societal issues [16], vendors' solutions [17]). An archive of related papers can be found at [18].

Though necessary to make attackers accountable (especially for DDOS attacks where there are a large amount of packets with spoofed source IP addresses), IP traceback has its own limitations. In particular, IP traceback cannot go beyond the hosts that send the spoofed IP packets. Indeed, a typical attacker will use a fair number of stepping-stones before he/she finally launches, for example, a DDOS attack. Thus, only identifying the source of IP packets is not sufficient to hold the attackers responsible for their actions.

Similar to IP traceback, there have been active research efforts on tracing intruders across stepping-stones. In general, approaches for traceback across a connection chain can be, based on the source of tracing information, divided into two categories: host-based and network-based. In addition, depending on how the traffic is traced, traceback approaches can be further classified into either active or passive. Passive approaches monitor and compare all the traffic all the time, and they do not select the traffic to be traced. On the other hand, active approaches dynamically control when, where, what and how the traffic is to be correlated through customized processing. They only trace selected traffic when needed. Table 1 provides a classification of existing approaches for traceback across stepping-stones.

Table 1 Classification of Existing Approaches for Traceback Across Stepping Stones

	Passive	**Active**
Host-based	DIDS CIS	Caller ID
Network-based	Thumbprinting Timing-Based Deviation-Based	IDIP Sleepy Watermark

The representatives of host-based, passive approaches to tracing back across stepping-stones are Distributed Intrusion Detection System (DIDS) [19,20] and the Caller Identification System (CIS) [21]. DIDS attempts to

proactively keep track of all the users in the network and account for all activities to network-wide intrusion detection systems. Each monitored host in the DIDS domain collects audit trails and sends audit abstracts to a centralized DIDS director for analysis. While DIDS is capable of keeping track of all users moving around the network through normal login within the DIDS domain, it seems not feasible in large-scale network such as the Internet, because of its centralized monitoring of network activities. CIS eliminates centralized control by utilizing a truly distributed model. Each host along the login chain keeps a record about its view of the login chain so far. When the user from the (n-1)th host attempts to login into the nth host, the nth host asks the (n-1)th host about its view of the login chain of that user, which should be 1,2 ... n-1 ideally. The nth host then queries host n-1 to 1 about their views of the login chain and so on. Only when the login chain information from all queried hosts matches will the login be granted at the nth host. While CIS attempts to maintain the integrity of login chain by reviewing information from hosts along the login chain, it introduces excessive overheard to the normal login process.

Caller ID, described in [22], is a host-based, active approach used by the US Air Force. If an attacker breaks into a host via a sequence of stepping-stones, the Air Force, having the knowledge of the same attacks, breaks back into these hosts in the reverse order and eventually identifies the origin of the attacker. Caller ID introduces less overhead than DIDS and CIS. But its manual approach makes it difficult to trace short intrusion in today's high-speed network. Besides it legal complications, Caller ID also has the drawback that one must perform manual tracing on the host where the intruder is active, which is easily-noticed by the intruder. In addition, it may not work if the attacker fixes the vulnerabilities after he/she breaks into the stepping-stones.

The fundamental problem with host-based tracing approaches is that they depend on every host in the connection chain. If one host is compromised and is providing misleading correlation information, the whole tracing system is fooled. Thus, it is very difficult to deploy host-based tracing systems on the Internet.

In contrast, network-based approaches to traceback across stepping-stones do not require the participation of monitored hosts. They are usually based on the properties of network connections. For example, the application-level content of chained connections is invariant across the connection chain, and thus can be used for traceback purpose. The early network-based traceback techniques are passive, including thumbprint [22], time-based scheme [23], and deviation-based approach [24]. Thumbprint is the pioneering correlation technique that utilizes a small quantity of information to summarize connections [22]. Ideally it can uniquely distinguish a connection from unrelated connections and correlate those related connections in the same connection chain. While thumbprinting can be useful even when only part of the Internet implements it, it depends on clock synchronization to match thumbprints of corresponding intervals of connections. It also is vulnerable to retransmission variation. This severely limits its usefulness in real-time tracing.

The timing-based scheme is based on the distinctive timing characteristics of interactive traffic, rather than connection contents [23]. It pioneered new ways of correlating encrypted connections. In addition, it does not require tightly synchronized clocks, and it is robust against retransmission variation. Similarly, the deviation-based approach uses the minimum average difference between the sequence numbers of two TCP connections (i.e., the *deviation* between the two connections) to determine whether to correlation the two connections or not [24]. This notion of deviation considers both timing characteristics and the TCP sequence numbers, but does not depend on the TCP payload. Similar to the timing-based approach, the deviation-based approach does not require tight clock synchronization, and is robust against retransmission variations.

Although the above two approaches use different techniques to identify the invariants for tracing stepping-stones, both of them have the same set of limitations. First, both are intended for detecting interactive stepping-stones, and may not apply to machine-driven attacks launched through a chain of stepping-stones. Second, the performance will both degrade, if the attacker actively evades the detection. For example, an attacker can easily defeat the tracing systems by deploying a simple client at an intermediate host, which introduces meaningless content (e.g., 10 x's followed by 10 backspaces) while forwarding the commands from the attacker. In addition, the time-based scheme [23] cannot distinguish between stepping stones used by intruders and those resulting from normal activities, while the deviation-based method [24] is not directly applicable to encrypted or compressed connections. The techniques we propose to study are to address these limitations.

The active network-based approaches dynamically controls when, where, what and how connections are correlated through customized packet processing. Due to the active nature, such approaches concentrate on the connections of concern (e.g., the intrusive connections identified by intrusion detection systems), and thus require less resource than the passive ones. Existing active network-based approaches for traceback across stepping-stones are Intrusion Identification and Isolation Protocol (IDIP) [25] and Sleepy Watermark Tracing (SWT) [26]. IDIP is an application

layer protocol that coordinates intrusion tracking and isolation. The IDIP boundary controllers collaboratively locate and block the intruder by exchanging intrusion detection information, i.e., attack descriptions. While it does not require any boundary controller to record any connections for correlation, its intrusion tracing is closely coupled with intrusion detection. The effectiveness of IDIP depends on the effectiveness of intrusion identification through the attack description at each boundary controller. Therefore, IDIP requires each boundary controller to have the same intrusion detection capability as the IDS at the intrusion target host.

Sleepy Watermarking Tracing (SWT) applies the ideas of active networking [27,28,29] and watermarking [30] to traceback across stepping-stones [26]. SWT is an active approach; that is, it does not introduce any overhead when there is no intrusion detected. When intrusions are detected, the target will inject a watermark into the backward connection of the intrusion, and wake up and collaborate with intermediate routers along the intrusion path. SWT provides a highly efficient and accurate source tracing on interactive intrusions through chained telnet or rlogin. However, SWT is limited to unencrypted connections. In addition, a sophisticated intruder, after knowing the deployment of SWT systems, may remove the injected watermark and thus defeat the tracing system.

There are other research works such as various intrusion detection models (e.g., NIDES/STAT [31,32], data mining-based models [33,34]), intrusion detection systems (IDSs) (e.g., EMERALD [35], NetSTAT [36,37]), and interoperation of IDSs (e.g., CIDF [38], IDMEF [39]). These works are complementary to the aforementioned traceback techniques.

References

[1] J. Elliott. Distributed Denial of Service Attack and the Zombie Ant Effect. IP Professional, March/April 2000.

[2] Computer Emergency Response Team (CERT). CERT Advisor CA-2000-01 Denial-of-service developments. http://www.cert.org/advisories/CA-2000-01.html. January 2000.

[3] D. Ditrich. Distributed Denial of Service (DDoS) Attacks/Tools Resource Page. http://staff.washington.edu/dittrich/misc/ddos/, 2000.

[4] S. Savage, D. Wetherall, A. Karlin, and T. Anderson. Practical Network Support for IP Traceback. In *Proceedings of the 2000 ACM SIGCOMM Conference*, August 2000.

[5] D.X. Song and A. Perrig. Advanced and Authenticated Marking Scheme for IP Traceback. In *Proceedings of 2001 IEEE INFOCOM Conference*, 2001.

[6] K. Park and H. Lee. On the Effectiveness of Probabilistic Packet Marking for IP Traceback. In *Proceedings of 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (ACM SIGCOMM)*, pages 15 – 26, 2001.

[7] H.Y. Chang, R. Narayan, C. Sargor, F. Jou, S.F. Wu, B.M. Vetter, F. Gong, X. Wang, M. Brown, and J.J. Yuill. DECIDUOUS: Decentralized Source Identification for Network-Based Intrusions, in *Proceeding of 6th IFIP/IEEE International Symposium on Integrated Network Management*, pages 702-714, 1999.

[8] H.Y. Chang, P. Chen, A. Hayatnagarkar, R. Narayan, P. Sheth, N. Vo, C. L. Wu, S.F. Wu, L. Zhang, X. Zhang, F. Gong, F. Jou, C. Sargor, X. Wu. Design and Implementation of A Real-Time Decentralized Source Identification System for Untrusted IP Packets. In *Proceedings of the DARPA Information Survivability Conference & Exposition*, January 2000.

[9] S. M. Bellovin. ICMP Traceback Messages, Internet Draft, March 2001.

[10] A. Mankin, D. Massey, C. Wu, S. F. Wu, and L. Zhang, On Design and Evaluation of Intention-Driven ICMP Traceback, In *Proceedings of IEEE International Conference on Computer Communications and Networks*, 2001

[11] D. Dean, M. Franklin, and A. Stubblefield. An Algebraic Approach to IP Traceback. In *Proceedings of 2001 Network and Distributed System Security Symposium*, February 2001.

[12] R. Stone. CenterTrack: An IP Overlay Network for Tracking DoS Floods. In *Proceedings of 9th Usenix Security Symposium*, August 2000.

[13] A.C. Snoeren, C. Partridge, L.A. Sanchez, C.E. Jones, F. Tchakountio, S.T. Kent, and W.T. Strayer. Hash-based IP Traceback. In *Proceedings of 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (ACM SIGCOMM)*, pages 3 – 14, 2001.

[14] L.A. Sanchez, W.C. Milliken, A.C. Snoeren, F. Tchakountio, C.E. Jones, S.T. Kent, C. Partridge, and W.T. Strayer. Hardware Support for a Hash-Based IP Traceback. In *Proceedings of DARPA Information Survivability Conference & Exposition*, June 2001.

[15] H. Burch, B. Cheswick. Tracing Anonymous Packets to Their Approximate Source. In *Proceedings of the 14th USENIX Systems Administration Conference (LISA 2000)*, December 2000.

[16] S. C. Lee and C. Shields, Tracing the Source of Network Attack: A Technical, Legal and Societal Problem, In *Proceedings of the 2001 IEEE Workshop on Information Assurance and Security*, June 2001

[17] Cisco Systems, Characterizing and Tracing Packet Floods Using Cisco Routers, Aug 1999. Available at: website http://www.cisco.com/warp/public/707/22.html

[18] Silicon Defense. Traceback and Related Papers Archive. Available at: website http://www.silicondefense.com/research/itrex/archive/tracing-papers/.

[19] S. Snapp et al. DIDS (Distributed Intrusion Detection System) – Motivation, Architecture, and Early Prototype. In *Proceedings of 14th National Computer Security Conference*, pages 167-176, October 1991.

[20] C. Ko, D.A. Frincke, T.Goan Jr., L.Tl. Heberlein, K. Levitt, B.Mukherjee, C. Wee. Analysis of An Algorithm for Distributed Recognition and Accountability. In *Proceedings of the 1st ACM Conference on Computer and Communication Security*, pages 154-164, 1993.

[21] H. Jung, et al. Caller Identification System in the Internet Environment. In *Proceedings of 45h USENIX Security Symposium*, 1993.

[22] S. Staniford-Chen, L.T. Heberlein. Holding Intruders Accountable on the Internet. In *Proceedings of IEEE Symposium on Security and Privacy*, May 1995.

[23] Y. Zhang and V. Paxson. Detecting Stepping Stones. In *Proceedings of 9th USENIX Security Symposium*, 2000.

[24] K. Yoda and H. Etoh. Finding a Connection Chain for Tracing Intruders. In *Proceedings of the 6th European Symposium on Research in Computer Security (LNCS 1985)*, Toulouse, France, October 2000.

[25] D. Schnackenberg, K. Djahandari, and D. Sterne. Infrastructure for Intrusion Detection and Response. In *Proceedings of DARPA Information Survivability Conference & Exposition*, January 2000.

[26] X. Wang, D. Reeves, S. F. Wu, and J. Yuill. Sleepy Watermark Tracing: An Active Network-Based Intrusion Response Framework, In *Proceedings of 16th International Conference on Information Security (IFIP/Sec 01)*, June 2001.

[27] S. Bhattacharjee, K.L. Calvert, and E.W. Zegura. Architecture for Active Networking. In *Proceedings of High Performance Networking (HPN '97)*, White Plans, NY, April 1997.

[28] K.L. Calvert, S.Bhattacharjee, and E. Zegura. Directions in Active Networks. *IEEE Communication*, 1998.

[29] R.H. Campbell, Z. Liu, M.D. Mickunas, P. Naldurg, and S. Yi. Seraphim: Dynamic Interoperable Security Architecture for Active Networks. In *Proceedings of IEEE OPENARCH 2000*, March 2000.

[30] N. Johnson, Z. Duric, and S. Jajodia. *Information Hiding: Steganography and Watermarking - Attacks and Countermeasures*. Kluwer Academic Publishers, February 2001.

[31] D. Anderson, T.F. Lunt, H. Javitz, A. Tamaru, and A. Valdes. Detecting Unusual Program Behavior Using the Statiscal Component of the Next-generation Intrusion Detection Expert System ({NIDES). *Technical Report SRI-CSL-95-06, SRI Internal, Computer Science Laboratory*, 1995.

[32] H.S. Javits, and A. Valdes. The NIDES Statistical Component: Description and Justification. Technical Report SRI International, Computer Science Laboratory, March 1993.

[33] W. Lee, S.J. Stolfo, and K.W. Mok. A Data Mining Framework for Building Intrusion Detection Models. In *Proceedings 1999 IEEE Symposium on Security and Privacy*, pages 120 – 132, May 1999.

[34] W. Lee and S.J. Stolfo. A Framework for Constructing features and Models for Intrusion Detection Systems. *ACM Transactions on Information and System Security*, 3(4):227-261, November 2000.

[35] P. A. Porras and P.G. Neumann. EMERALD: Event Monitoring Enabling Response to Anomalous Live Disturbances. In *Proceedings of the 20th National Information Systems Security Conference*, 1997.

[36] G. Vigna and R. A. Kermmerer. NetSTAT: A Network-based Intrusion Detection Approach. In *Proceedings of the 14th Annual Security Applications Conference*, December 1998.

[37] G. Vigna and R. A. Kermmerer. NetSTAT: A Network-based Intrusion Detection Approach. *Journal of Computer Security*, 7(1):37-71, 1999.

[38] C. Kahn, P. A. Porras, S. Staniford-Chen, and B. Tung. A Common Intrusion Detection Framework. *Submitted to Journal of Computer Security*, 1998.

[39] D. Curry, and H. Debar. Intrusion Detection Message Exchange Format Data Model and Extensible Markup Language (XML) Document Type Definition. *Internet Draft, draft-ietf-idwg-idmef-xml-03.txt*, February 2001.