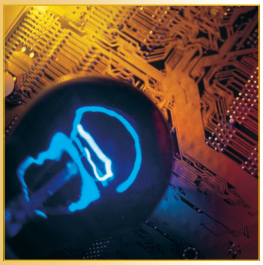

Tracing Network Attacks to Their Sources

An IP traceback architecture in which routers log data about packets and adjacent forwarding nodes lets us trace IP packets to their sources, even when the source IP address is forged.

Tatsuya Baba and Shigeyuki Matsuda
NTT Data Corporation

As the Internet becomes increasingly important as a business infrastructure, the number of attacks on it, especially denial-of-service attacks such as TCP SYN flooding,[1] Teardrop,[2] and Land,[2] grows. Because of the weak security in TCP/IP, we must take responsibility for protecting our own sites against network attacks.

Although access-control technologies, such as firewalls, are commonly used to prevent network attacks, they cannot prevent some specific attacks, including TCP SYN flooding. Consequently, more companies are deploying intrusion detection systems (IDS). (See the sidebar, "Technologies for Preventing Network Attacks," page 23, for a discussion of current access-control and detection systems.) IDSs detect network attacks; however, they do not let us identify the attack source. This is especially problematic with denial-of-service attacks, for example, because the attacker doesn't need to receive packets from the target host and thus can remain hidden.

Several efforts are in progress to develop source-identification technologies to trace packets even when an attacker forges its IP address. In this article, we describe some proposed IP traceback architectures, including our own, which we have implemented in a prototype. In our system, routers log data about traversing packets as well as information about other nodes in the packet's path.[3] We use a distributed management approach to enable tracing across networks with different access policies.

Methods of IP Traceback

The purpose of IP traceback is to identify the true IP address of a host originating attack packets. Normally, we can do this by checking the source IP address field of an IP packet. Because a sender can easily

 1089-7801/02/$17.00 ©2002 IEEE

forge this information, however, it can hide its identity. If we can identify the true IP address of the attack host, we can also get information about the organization, such as its name and the network administrator's e-mail address, from which the attack originated. With IP traceback technology, which traces an IP packet's path through the network, we can find the true IP address of the host originating the packet. To implement IP traceback in a system, a network administrator updates the firmware on the existing routers to the traceback support version, or deploys special tracing equipment at some point in the network.

Existing IP traceback methods can be categorized as *proactive* or *reactive* tracing.

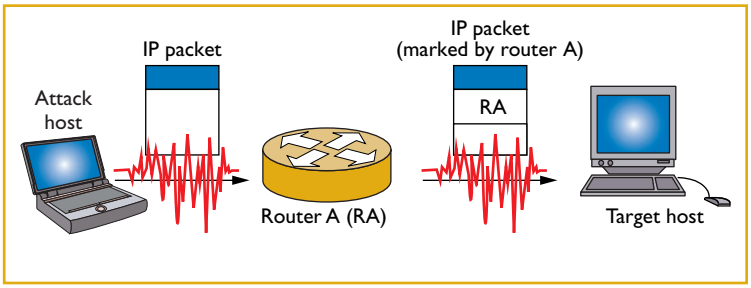

Figure 1. Packet marking. As packets travel through the network, they gather and store information about the routers they traverse.

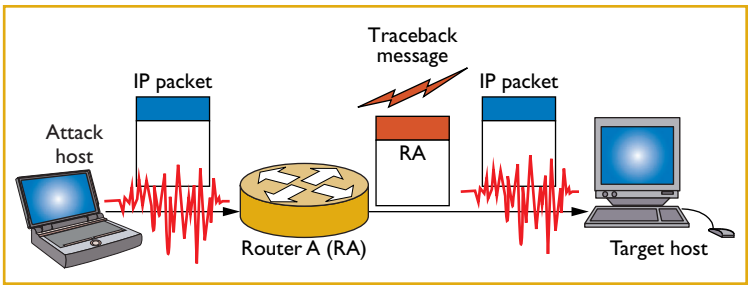

Figure 2. ICMP traceback message. Routers create and send messages containing packet information to the packet's destination.

Proactive Tracing

Proactive tracing prepares information for tracing when packets are in transit. If packet tracing is required, the attack victim (or target) can refer to this information to identify the attack source. Two proactive tracing methods — *packet marking* and *messaging* — have been proposed.

Packet marking. In packet marking, which is illustrated in Figure 1, packets store information about each router they pass as they travel through the network. The recipient of the marked packet can use this router information to follow the packet's path to its source. Routers must be able to mark packets, however, without disturbing normal packet processing.

With IP's *record route option,* for example, the IP packet can store router addresses in its option field. In another proposed approach, the router writes its identifier probabilistically in the packet's IP header identification field.[4] Each marked packet contains information in its identification field about only one or two routers on the attack path. In a flooding-style attack, however, the target network receives many attack packets and can collect enough information to identify the attack path. The identification field is used to reassemble fragmented packets. Because few fragments are created on the Internet, however, modifying the identification field rarely affects normal packet processing.

Messaging. In messaging approaches, routers create and send messages containing information about the forwarding nodes a packet travels through.

Figure 2 illustrates the Internet Engineering Task Force's proposed method, the Internet control message protocol (ICMP) traceback message.[5] A router creates an ICMP traceback message, which contains part of a traversing IP packet, and sends the message to the packet's destination. We can identify the traversed router by looking for the corresponding ICMP traceback message and checking its source IP address. Because creating an ICMP traceback message for every packet increases network traffic, however, each router creates ICMP traceback messages for the packets it forwards with a probability of 1/20,000. If an attacker sends many packets (for example, in a flooding-style attack), the target network can collect enough ICMP traceback messages to identify its attack path.

Reactive Tracing

Reactive tracing starts tracing after an attack is detected. Most of the methods trace the attack path from the target back to its origin. The challenges are to develop effective traceback algorithms and packet-matching techniques. Various proposals attempt to solve these problems.

Hop-by-hop tracing. In hop-by-hop tracing, which is illustrated in Figure 3 (next page), a tracing program, such as MCI's DoSTracker, logs into the router closest to the attacked host and monitors incoming packets. If the program detects the spoofed packet (by comparing the packet's source IP address with its routing table information), it logs into the upstream routers and monitors packets. If the spoofed flooding attack is still occurring,

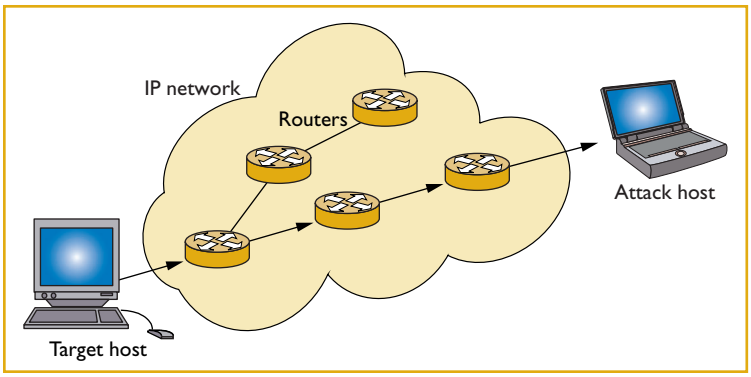

Figure 3. Hop-by-hop tracing. Hop-by-hop tracing starts at the router nearest the target host and follows the attack packet back to its source, hop by hop, during the attack.

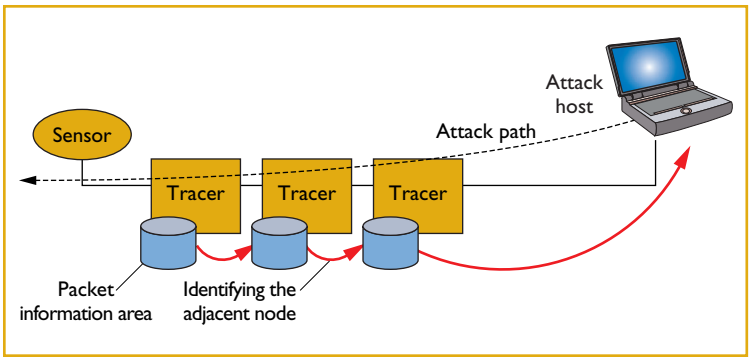

Figure 4. Basic method of our traceback approach. Forwarding nodes, or tracers, store data from an incoming packet as well as its datalink-level identifier in the packet information area, and they identify the adjacent forwarding node.

the program can detect the spoofed packet again on one of the upstream routers. This procedure is repeated recursively on the upstream routers until the program reaches the attack's actual source.

Hop-by-hop tracing with an overlay network. In hop-by-hop tracing, the more hops there are, the more tracing processes will likely be required. As a result, a packet will take longer to trace, and necessary tracing information might be lost before the process is complete. To decrease the number of hops required for tracing, one approach builds an overlay network by establishing IP tunnels between edge routers and special tracking routers and then reroutes IP packets to the tracking routers via the tunnels.[6] Hop-by-hop tracing is then performed over the overlay network.

IPsec authentication. Another proposed reactive tracing technique is based on existing IP security protocols.[7] With this method, when the IDS detects an attack, the Internet key exchange (IKE) protocol establishes IPsec security associations (SAs) between the target host and some routers in the administrative domain (for example, autonomous system boundary routers). Routers at the SA ends add an IPsec header and a tunnel IP header containing the router's IP address to traversing packets. If the attack continues and one of the established SAs authenticates a subsequent attack packet, the attack must come from a network beyond the corresponding router. The receiver checks the source IP address of the tunnel IP header to find out which routers the attack packet traversed. Repeating this process recursively, the receiver finally reaches the attack source.

Because this technique uses existing IPsec and IKE protocols, implementing a new protocol for tracing within an administrative domain is unnecessary. To trace beyond the administrative domain, however, a special collaboration protocol is needed. The IETF Intrusion Detection working group (IDWG) is discussing such a protocol.

Traffic pattern matching. A fourth proposed technique traces an attack path by comparing traffic patterns observed at the entry and exit points of the network with the network map.[8]

A Proposed Architecture for IP Traceback

Of the approaches described above, we believe that hop-by-hop tracing is the most reliable. Moreover, while most tracing techniques deal only with flooding-style DoS attacks, some attacks use only one or a few IP packets (for example, the Land or Teardrop attacks[2]). We therefore believe any solution must be able to trace the source of an attack using a single packet.

In our reactive approach, forwarding nodes (such as routers) log information about traversing packets on the Internet and then use the log data to trace each packet from its final destination back to its source, hop by hop. Information about the packets remains in forwarding nodes as packets traverse, allowing us to trace even a single attack packet to its source. Our approach goes beyond hop-by-hop tracing by using datalink-level identifiers such as Ethernet's media access control (MAC) address, ATM's virtual path identifier/virtual channel identifier (VPI/VCI), and frame relay's datalink connection identifier (DLCI) to identify nodes in the packet's path.

Intermediate forwarding nodes change a packet's (or frame's or cell's) datalink-level identifier to match their interface identifiers. Although it is

Technologies for Preventing Network Attacks

Current technologies for protecting networks against attacks focus on access control and attack detection. Although some methods can find the attacker's identity, they are unsuccessful when the attacker's true IP address is hidden or unknown.

Firewalls

Firewalls are widely used to protect networks against attacks, especially those coming from the Internet. Usually, firewalls control access based on source IP address, destination IP address, protocol type, source port number, and destination port number. For example, we can configure a firewall to deny any access to a WWW server except for WWW access using HTTP (destination port number 80). If an attacker attempts to exploit the WWW server using HTTP, however, the firewall cannot prevent it.

Intrusion Detection

An intrusion detection system (IDS) detects network attacks to a computer system. One major method currently implemented in IDS products is misuse detection. In this method, the IDS compares the *attack signatures*, which are features of known attacks, with the contents of packets on the network or log data on the host computer. When the packet content or log data matches an attack signature, the system recognizes that an attack has occurred. IDSs still pose accuracy problems for site managers, however. In practice, IDSs detect possible attacks, which site managers must examine to determine whether it is a real attack.

Intrusion Source Identification

Using IDSs, we can detect certain attacks and find the attack packets' source IP addresses. Because the IP address is not enough to identify the attack source, however, we typically run a DNS inverse query to check the fully qualified domain name (FQDN), or look up the database in a WHOIS server to find the source identity (for example, organization name and e-mail address).

If the attack's purpose is penetration or reconnaissance, most attackers will hardly disguise the source IP address because they must receive a response from the target. An attacker who aims for denial of service (DoS), however, does not need to receive packets from the target and can therefore forge its source IP address.

Ingress filtering deals with forged addresses.[1] In this method, a router compares an incoming packet's source IP address with a router's routing table and discards packets with inconsistent source addresses as having been forged. This method is effective for many spoofed DoS attacks, but it fails if an attacker changes its source IP address to one that belongs to the same network as the attacker's host.

Reference

1. P. Ferguson and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing," Internet Engineering Task Force RFC 2267, Jan. 1998; available at http://www.ietf.org/rfc/rfc2267.txt.

easy to forge a packet's source IP address, it is difficult for a sender to forge the datalink-level identifiers of these intermediate forwarding nodes. By referring to the datalink-level identifier corresponding to a particular packet, we can identify for each forwarding node the adjacent node through which the packet has passed.

In our approach, forwarding nodes, or *tracers*, keep data about an incoming packet and its datalink-level identifier (such as source MAC address) in a buffer memory (the *packet information area*) and identify each adjacent node by searching for the datalink-level identifier of the forwarded packet that matches the attack packet. If the traffic is very high, old data can quickly be overwritten by new data, making tracing impossible. To save memory space, the tracer stores only some IP header fields and some IP data for all packets. As Figure 4 illustrates, we start tracing from the tracer closest to the target to identify each adjacent node along the attack path and reach the attack packet's source even if its IP address has been forged.

System Configuration

The major components of our traceback system are the sensor, monitoring manager, and tracer.

- The sensor, which is deployed at a target site, monitors packets on the network. When it detects an attack, the sensor sends a tracing request to the monitoring manager.
- In response to a sensor request, the monitoring manager controls tracers and manages the entire tracing process.
- The tracer, which is implemented in forwarding nodes such as routers, maintains log information about forwarded IP packets. The tracer also compares the log data with information about the tracing packet and finds a trace path.

The Internet's size makes it impossible to centrally manage the entire tracing process and the necessary tracing information. Moreover, it is difficult for networks configured with different access policies to trace packets coming from other networks without imposing any limitations. We therefore adopt a distributed management approach that controls the tracing process and information within a particular group of networks. We call this trace control sec-

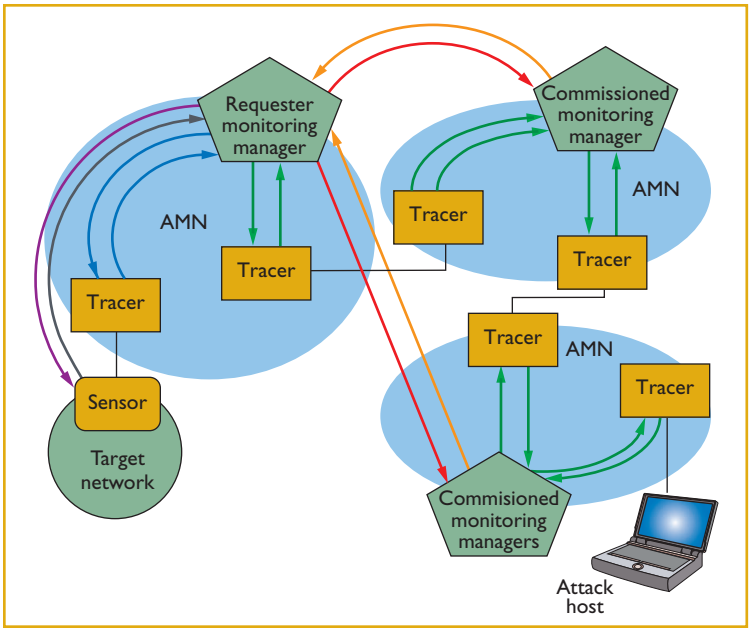

Figure 5. Proposed architecture's tracing process. In response to a request from a sensor, the monitoring manager sends tracers to trace an attack packet to its source.

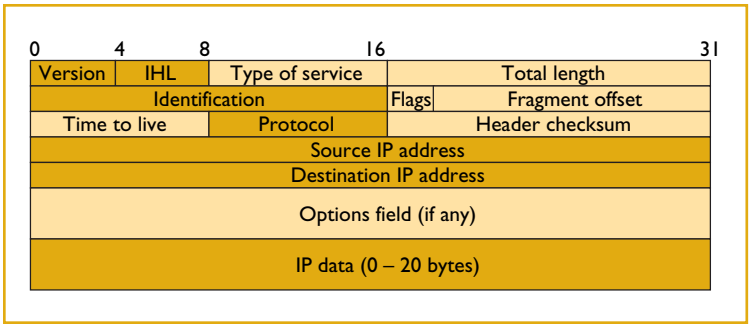

Figure 6. Packet feature structure (darkened fields). Tracers store packets' packet features in their buffer memories, or packet information areas, for use in packet tracing.

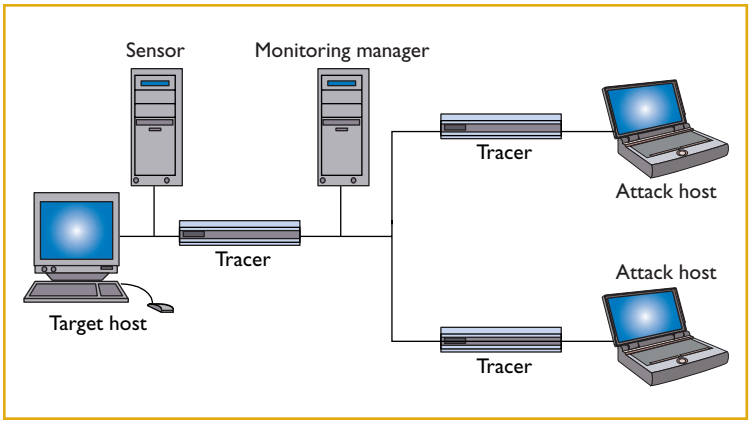

Figure 7. Experimental network. The sensor monitors the target network, and attackers send attack packets with forged source IP addresses to the target host.

tion the autonomous management network (AMN).

The monitoring manager, which is deployed in each AMN, executes a tracing process within its own AMN and manages tracing information. If a tracing process goes beyond the AMN's boundary, the monitoring manager of the AMN that initiated the tracing process asks the monitoring manager in the adjacent AMN to trace the packet.

Process Flow

Our traceback approach involves several steps, from attack detection to source identification, which are illustrated in Figure 5.

1. Sensors are deployed at each target network. When a sensor detects an attack, it creates data containing features of the attack packet and sends a tracing request to the monitoring manager deployed in its AMN.
2. The monitoring manager orders the AMN's tracer to trace the attack packet. The tracer identifies the adjacent node and returns the result to the monitoring manager.
3. Based on the result returned, the process described above continues until the tracer identifies the attack packet's source.
4. If a tracing process goes beyond the AMN's boundary, processing is handed over to the relevant monitoring manager (*the commissioned monitoring manager*) that controls that AMN.
5. The monitoring managers in each AMN trace the packet in their AMN and send the tracing results to the monitoring manager that initiated the traceback request (the *requester monitoring manager*).
6. The requester monitoring manager sends the final results to the sensor that requested the trace.

Packet Identification Mechanism

A part of the packet is temporarily stored in tracers' packet information areas. This information, the *packet feature*, consists of several IP header fields that are unchanged in transit and a part of the IP data field (from the first byte up to 20 bytes). Fields that can change in transit, such as time to live (TTL), header checksum, and options, are not included. The darkened fields in Figure 6 represent the packet feature.

When a tracer receives an attack packet's packet feature, it searches data in its packet information area for the corresponding datalink-level identifier of the adjacent node. This matching process is subject to several conditions:

Table 1. Prototype specifications.

Component	Processor	CPU	Memory	Network interface	Operating system
Sensor	Dell Power Edge 1300	Pentium III 600 MHz	384 Mbytes	10 Mbps	FreeBSD 4.2-Release
Monitoring manager	Sun Enterprise 250 Server	UltraSparc-II 400 MHz	512 Mbytes	10 Mbps	Solaris 2.7
Tracer	Kawasaki Steel A2DIS SV-1000	MC68360 25 MHz	16 Mbytes (packet information area: 8 Mbytes)	10 Mbps	INFOS/INCS

- The packet feature in the forwarded packet's packet information area is identical to the attack packet's packet feature.
- If the lengths of the IP data portions of the packet features differ, the tracer uses the shorter length to compare.
- If IP fragmentation has occurred in transit, the bottom part of the original packet's IP data portion might be missing from the first fragment packet. Therefore, we use the shorter length when comparing the IP data portion in the packet feature generated by the sensor with the IP data portion kept by the tracers.

Implementation and Experimental Results

We have implemented our proposed architecture in a prototype system. Table 1 shows the system specifications, and Figure 7 shows our experimental network.

As Figure 8 shows, traceback provides an attack path. Table 2 shows the trace time of our experiment.

In our experimental environment, the prototype system can trace packets to their sources. For tracers in real-world environments, however, especially on backbone networks where network traffic is very high, more memory will likely be needed. Moreover, it is difficult to deploy tracers and monitoring managers all over the Internet at the same time. One way to enable traceback in real-world environments is to introduce these components into one administrative domain (such as a corporate intranet) and enable traceback within that domain first. If the adjacent domain also introduces the tracing function, the domains can trace beyond their network boundaries by exchanging trace information between monitoring managers.

Limitations and Open Issues

IP traceback has several limitations, such as the problem with tracing beyond corporate firewalls. To accomplish IP traceback, we need to reach the host where the attack originated. It is difficult, however, to trace packets through firewalls into corporate intranets — the last-traced IP address might be the firewall's address. Knowing the IP address of the organization's network entry point, however, allows us to obtain information about the organization where the attacker's host is located, such as the organization's name and the network administrator's e-mail address. If we can identify the organization from which the attack originated, the organization can often identify the user who launched the attack.

Another limitation relates to the deployment of

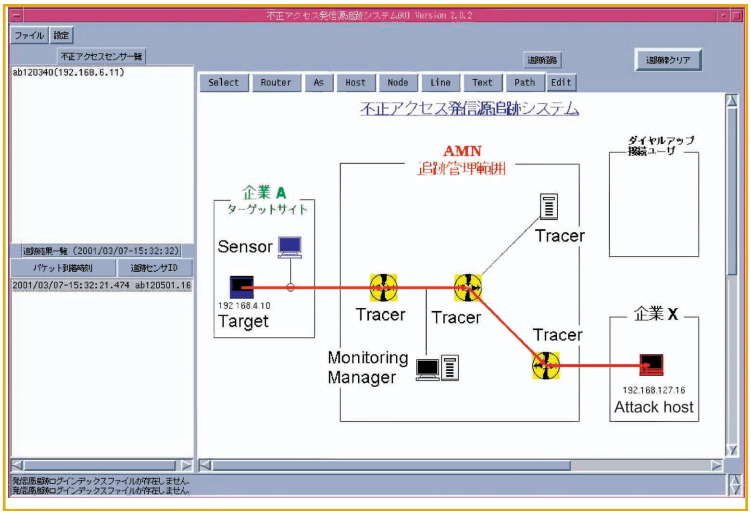

Figure 8. Experimental result of attack path. After completing the trace, the trace results and attack time are displayed in a browser window.

Table 2. Trace time results.

No. of hops	1	2	3	4	6	9	10
Elapsed time (in seconds)	0.300	0.419	0.609	0.910	1.723	2.464	2.598

traceback systems. Most traceback techniques require altering the network, including adding router functions and changing packets. To promote traceback approaches, we need to remove any drawbacks to implementing them.

Moreover, even if IP traceback reveals an attack's source, the source itself might have been used as a stepping-stone in the attack. IP traceback methods cannot identify the ultimate source behind the stepping-stone; however, techniques to trace attacks exploiting stepping-stones are under study.[9]

Some operational issues must also be solved before IP traceback can be widely deployed. To trace an attack packet through different networks, for example, there must be a common policy for traceback. We also need guidelines for dealing with traceback results to avoid infringing on privacy. Furthermore, we need to consider how to use information about an attack source identified by IP traceback. In the future, we will likely need to focus on the authenticity of results from IDSs and IP traceback systems.

Acknowledgments

This work is funded by the Telecommunications Advancement Organization of Japan (TAO).

References

1. Computer Emergency Response Team, "TCP SYN Flooding and IP Spoofing Attacks," CERT Advisory CA-1996-21, Sept. 1996; available online at http://www.cert.org/advisories/CA-1996-21.html.
2. Computer Emergency Response Team, "IP Denial-of-Service Attacks," CERT Advisory CA-1997-28, Dec. 1997; available online at http://www.cert.org/advisories/CA-1997-28.html.
3. S. Taketsume et al., "A Study of Architecture for Unauthorized Access Tracing System," *Proc. 60th Nat'l Convention of IPSJ*, vol. 3, Information Processing Soc. of Japan (IPSJ), Tokyo, Mar. 2000, pp. 287-288 (Japanese only).
4. S. Savage et al., "Practical Network Support for IP Traceback," *Proc. 2000 ACM SIGCOMM*, vol. 30, no. 4, ACM Press, New York, Aug. 2000, pp. 295-306; available online at http://www.cs.washington.edu/homes/savage/traceback.html.
5. S. Bellovin, M. Leech, and T. Taylor, "ICMP Traceback Messages," Internet draft, work in progress, Oct. 2001; available online at http://www.ietf.org/internet-drafts/draft-ietf-itrace-01.txt (expires Apr. 2002, last accessed 28 Jan. 2002).
6. R. Stone, "CenterTrack: An IP Overlay Network for Tracking DoS Floods," *Proc. 9th Usenix Security Symp.*, Usenix Assoc., Berkeley, Calif., Aug. 2000; available online at http://www.usenix.org/publications/library/proceedings/sec2000/stone.html.
7. H.Y. Chang et al., "DecIdUouS: Decentralized Source Identification for Network-Based Intrusions," *Proc. 6th IFIP/IEEE Int'l Symp. Integrated Network Management*, IEEE Comm. Soc. Press, New York, May 1999, pp. 701-714; available online at http://shang.csc.ncsu.edu/deciduous/.
8. K. Ohta et al., "Detection, Defense, and Tracking of Internet-Wide Illegal Access in a Distributed Manner," *Proc. INET 2000*, Internet Soc., Reston, Va., July 2000; available online at http://www.isoc.org/inet2000/cdproceedings/1f/1f_2.htm.
9. M. Asaka et al., "A Method of Tracing Intruders by Use of Mobile Agents," *Proc. INET 99*, Internet Soc., Reston, Va., June 1999; available online at http://www.isoc.org/inet99/4k/4k_2.htm.

Tatsuya Baba is a researcher in NTT Data's research and development division. He graduated in electrical engineering at Keio University, Japan, in 1995. His research interests include intrusion detection, security protocols, and privacy protection. He is a member of the IEEE, the Internet Society, and the Information Processing Society of Japan (IPSJ).

Shigeyuki Matsuda is a senior manager of NTT Data's research and development division. He received BA and MS degrees in information processing from the University of Electro-Communications, Japan. His research interests include network technology and network security. He is a member of the IEEE, the IPSJ, and the Institute of Electronics, Information, and Communication Engineers, Japan (IEICE).

Readers may contact the authors at {baba, matu}@rd.nttdata.co.jp.