Internet Routing Operation Technology (ENCORE)

Toshimitsu Oshima†, *Mitsuho Tahara, Kazuo Koike, Yoshihiro Otsuka, and Souhei Majima*

Abstract

ENCORE is a multi-agent-based system for automatically analyzing failures on the Internet that affect several autonomous systems at once. Its two monitoring functions, hijack and missing-route monitoring, are currently based on one-to-one cooperation among ENCORE agents. To improve the detection precision of these monitoring functions, we are extending the agent-cooperation method. This paper explains the need to perform monitoring by using one-to-many-agent cooperation which enables an agent to coordinate with multiple agents at the same time. Problems associated with achieving one-to-many-agent cooperation are described and their solutions are presented.

1. Background

The Internet is a huge aggregate of countless networks known as autonomous systems (ASs), which are operated by Internet service providers (ISPs), universities, companies, or other organizations. An IP (Internet protocol) packet sent from one AS must pass through several other ASs before finally reaching the targeted AS. During this process, routing information is exchanged between ASs to set the IP packet route. But for some time an anomaly has been recognized: because each AS has its own routing information management policy, inconsistencies can very easily occur among the policies of ASs. Furthermore, because policies are set manually, routes can suffer from instability due to setting errors and other anomalies. As a result, there have been several incidents of large-scale loss of connectivity.

In the past, however, routing anomalies between ASs could only be analyzed manually by network operators with specialized skills, and it was extremely difficult for network operators to constantly monitor huge volumes of routing information that changed from moment to moment in order to discover such anomalies at an early stage.

NTT Laboratories therefore initiated research targeting automatic routing anomaly diagnosis technologies as part of its efforts to counteract the instability and vulnerabilities of the Internet. In 2001, NTT Network Innovation Laboratories developed ENCORE, the world's first system of its kind [1]. Later, NTT verified the effectiveness of the system in evaluation tests on a global scale, connecting four locations inside and outside Japan. NTT Network Service Systems Laboratories, in collaboration with NTT Communications, confirmed the system's feasibility on actual networks and added "hijack monitoring" and "missing route monitoring" functions based on the results of new research [2]. This led to the start of full-scale operations on NTT's commercial OCN service network.

2. Monitoring functions of ENCORE

2.1 Hijack monitoring function

ENCORE detects cases in which a given router's routing table has been rewritten due to an erroneous routing information advertisement from another AS, and it identifies whether the source of the error is being operated as a proper punching hole*. In this way, when an AS's route has been taken over as a result of a deliberate illegal setting by a malicious

† NTT Network Service Systems Laboratories
Musashino-shi, 180-8585 Japan
E-mail: ooshima.toshimitsu@lab.ntt.co.jp

third party or by an erroneous setting performed accidentally by a network operator, the situation can be quickly and automatically detected, and the AS causing the anomaly can be identified [3].

2.2 Missing route monitoring function

When a route is omitted due to an erroneous filter setting by the network operator or by a difference in policies among ASs, the agents at the two ASs in question exchange and analyze observation information to automatically detect routes that are missing from a given AS's routing table. This improves the stability of IP packet transmission [4].

3. Agent cooperation

In the conventional ENCORE system, cooperation between agents is performed on a one-to-one basis (i.e., a one-to-one cooperation model). In this paper, aiming to improve detection precision for the hijack and missing-route monitoring functions, as the next step in agent cooperation, we describe the need for a method of cooperation between one agent and many other agents. To achieve this one-to-many monitoring cooperation, however, we must address the issues described below.

3.1 One-to-many cooperation among agents in hijack monitoring

Hijack monitoring has been based on the one-to-one cooperation until now (**Fig. 1**). However, a hijacked route cannot be detected by certain ASs. Two representative cases are described below.

- Internal routes
  Among routes listed in the routing table of an AS, there are some routes that are used only within an AS and are not advertised on the Internet. When an AS (AS5) hijacks a route of another AS for its own internal route, AS2 cannot detect this. Only the AS committing the hijacking (AS5) is aware of it. It is therefore necessary to pre-assign ENCORE agents to as many ASs as possible. As shown on the right of Fig. 1, even though AS2 cannot detect the hijacked route, other agents can, which enables AS1 to detect the hijacked route through one-to-many cooperation.
- Route filtering
  In some ASs, long route lengths are filtered out by a route filter. For example, if the prefix length of a hijacked route is long (say, "10.0.0.0/29"), the

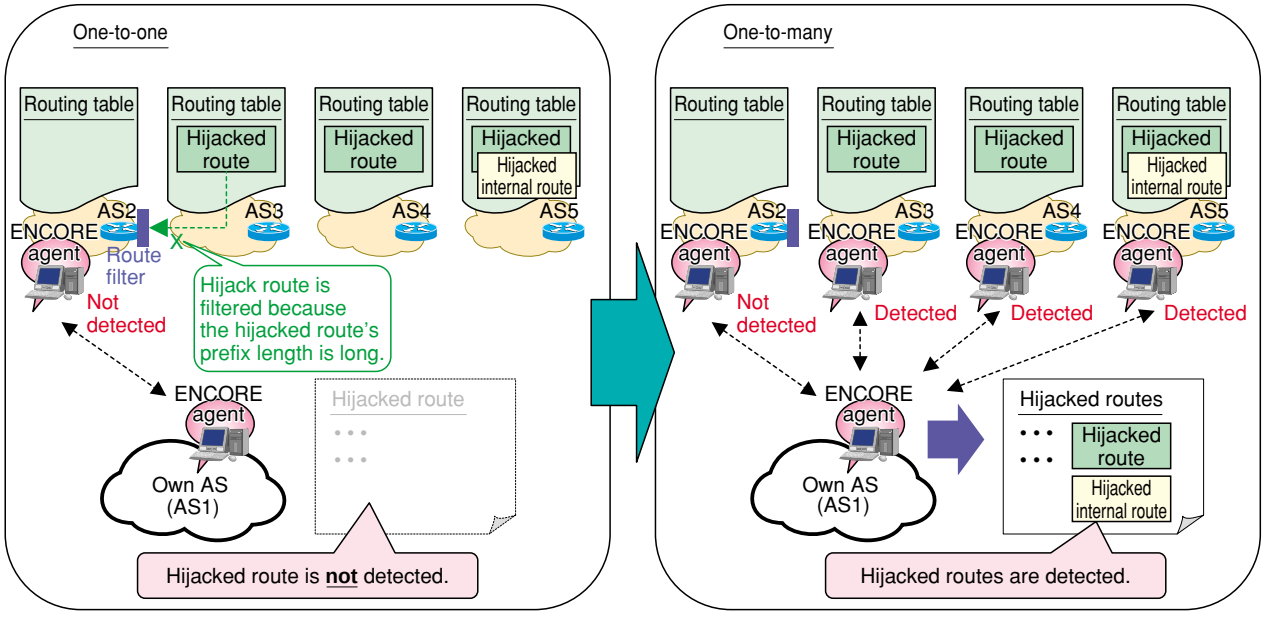

Fig. 1. Cooperation models for hijack monitoring.

* Punching hole: A multihomed site connected to multiple ISPs may obtain a small prefix from an ISP and announce it from multiple ISPs to achieve better reachability. This technique is called a punching hole.

hijacked route will not be written in the routing table in that AS (AS2), so the ENCORE agent cannot detect the hijack. In other words, an AS with such an operation policy is inappropriate as an AS for monitoring hijacked routes. To improve the precision of detecting hijacked routes in this case, it is desirable to perform as much monitoring as possible at ASs (such as AS3–AS5) that are not filtering routes by route length.

3.2 One-to-many cooperation among agents in missing-route monitoring

The current monitoring function detects routes missing from the routing table of one's own AS by utilizing cooperation among ENCORE agents on a one-to-one basis and by comparing the routing table of one's own AS with that of another AS (**Fig. 2**). The two current problems concerning this function are described below.

- Absence of routes in the routing table of the comparison AS

  If the comparison AS is missing the same routes from its routing table, then the absence of these routes in one's own routing table cannot be detected, resulting in a detection failure. In Fig. 2, in one-to-one cooperation, the missing route "prefix_B" cannot be detected through a comparison with AS2 because AS2 is also missing this route. On the other hand, it can be detected in the one-to-many cooperation-monitoring model. Although AS2 is also missing "prefix_B", other ASs (AS3–AS5) have it, so AS1 can detect that it is missing from its own routing table.
- Influence of routes on reachability

  Depending on the AS chosen for comparison, from around ten to a few hundred routes missing from one's own routing table are detected. These routes include those that do not need to be kept in one's own routing table, that is, irrelevant routes. More specifically, some internal routes used only within the comparison AS and not advertised on the Internet will be discovered. Even though such routes are missing from one's own routing table, this is not a problem. In determining which of the detected routes have the biggest influence on user needs, the communication destination included in the route must be considered. However, determining the communication destination requires a huge amount of data, so it is difficult. Like majority-rule decisions, the most important routes among those missing from one's own routing table are those held by many ASs. On the other hand, when routes are held by only a few ASs, their reachability is considered to be of little importance. Accordingly, we

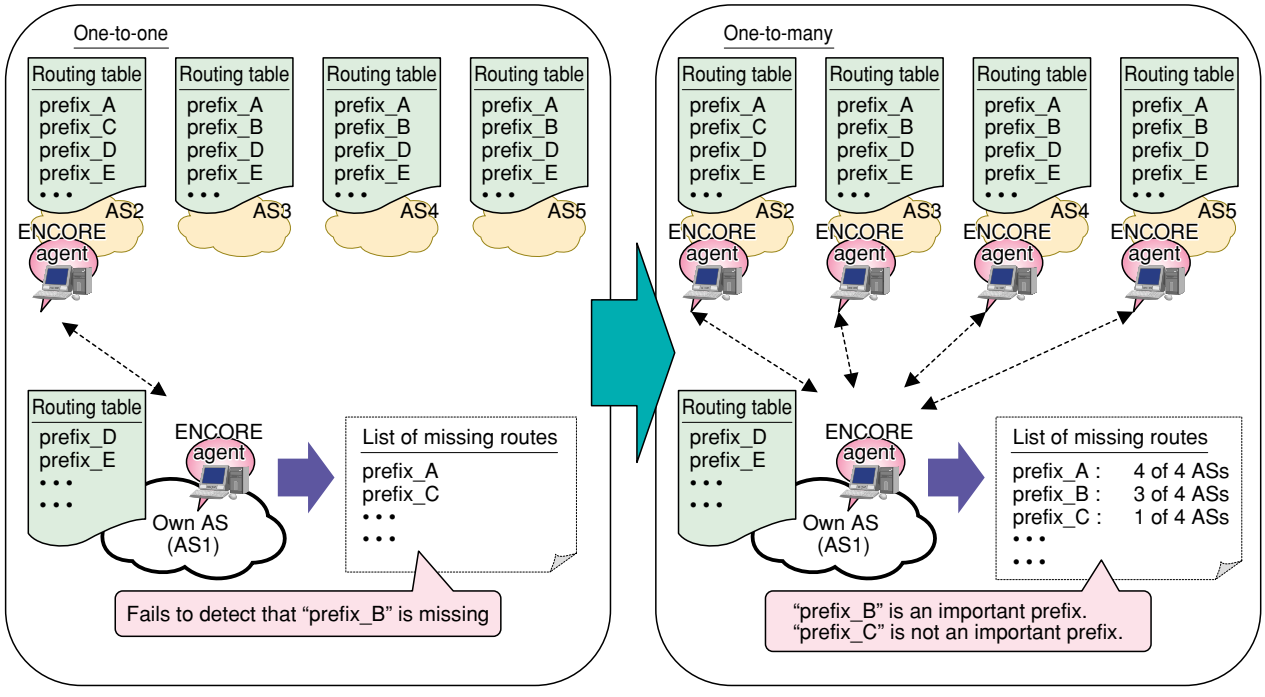

Fig. 2. Cooperation models for "missing route" monitoring.

are investigating a method for determining the importance of missing routes for the benefit of ENCORE operators.

Figure 2 shows that one-to-one cooperation detects route "prefix_C" in only one AS (AS2) out of four (AS2–AS5), so it is assumed that the reachability of "prefix_C" is of little importance. On the other hand, since all four ASs have route "prefix_A", it is assumed that route "prefix_A" is important. This shows that one-to-one cooperation cannot fully grasp this situation.

Considering the above problems, to improve the precision of missing route detection, we should design a system that monitors many ASs simultaneously.

3.3 Problems in achieving one-to-many cooperation

There are the four key problems in achieving one-to-many cooperation.

- Problem 1: Cost of assigning cooperative agents
  Conventionally, ENCORE operators designate the ENCORE agents for cooperation. Therefore, in monitoring based on one-to-many-agent cooperation, the cost of setting up ENCORE agents is high.
- Problem 2: Cost of replacing certification information of agents
  During communication within ENCORE, certification between ENCORE agents is executed in the application layer. At that time, certification information about one-to-one cooperation is exchanged between ENCORE operators in advance and configured in ENCORE. In one-to-many-agent cooperation, the conventional method of replacing certification information of agents forces an ENCORE operator to exchange ENCORE certification information with many other operators. Therefore, the cost for these exchanges is high.
- Problem 3: Difficulty in understanding the most suitable AS for each type of monitoring
  For better detection precision, appropriate ENCORE agents must perform their hijack and missing-route monitoring in harmony. For example, in missing-route monitoring, since routing tables with about 150,000 lines are analyzed, a huge amount of machine resources is expended. Since the AS to be compared is an upstream AS and different from one's own AS, not only must all the routes of many ASs be analyzed but the optimum route for an upstream AS must be chosen from among multiple dependent ASs. However, though choosing the optimum AS for such coordinating requires that the ENCORE operator must ascertain the optimum AS, ascertaining the characteristics of multiple ASs is actually troublesome.
- Problem 4: Need for resource management
  In contrast to the situation with a single ENCORE agent, when multiple ENCORE agents are designated for coordination, the manageable level of machine resources for operating the ENCORE agents is exceeded, so resource management is necessary.

In the next section, to try and solve the problems outlined above, we propose a scheme for securely connecting the appropriate ENCORE agents for coordination.

4. One-to-many cooperation support system

A system for supporting one-to-many cooperation in ENCORE is shown in **Fig. 3**. Its operation consists of four steps.

- Step 1: An ENCORE operator configures the system as the connection address of the ENCORE agent of its own AS. At that time, attribute information of one's own AS (AS number, country, an upstream AS, machine resources, etc.) and authentication information (passwords, etc.) for connecting the system are set up.
- Step 2: After authentication by the server, an ENCORE agent is connected to that system and transmits the information about one's own AS to the system.
- Step 3: The server sends back the destination of every appropriate ENCORE agent to be included in the cooperative monitoring and the authentication information between those agents to the ENCORE agent of one's own AS.
- Step 4: The ENCORE agent that receives that authentication information is connected to a designated ENCORE agent. After authentication, cooperative monitoring begins.

Considering problem 1, the designation of the connection destination for ENCORE agents involves only one place, namely, the ENCORE one-to-many cooperation support server. This solves the problem of the large setup burden for designating many ENCORE agents for cooperation.

To solve problem 2, the ENCORE one-to-many cooperation support system issues authentication information to every pairing of ENCORE agents for cooperation, and it executes authentication between the pairs of ENCORE agents. As a result, an ENCORE operator need not be concerned with

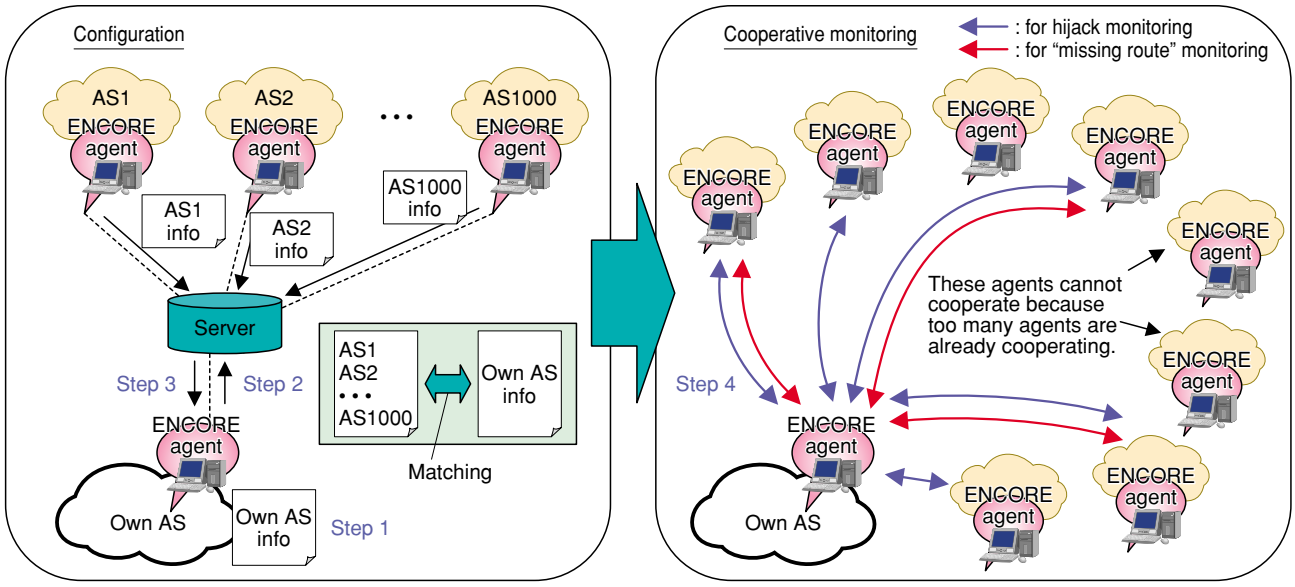

Fig. 3. Support system for one-to-many cooperation.

authentication between the ENCORE agents.

To solve problem 3, each ENCORE operator registers information about its own AS in the server through its ENCORE agent. As a result, the appropriate connection destination designated by the cooperation support system is sent to the ENCORE agent.

To solve problem 4, the cooperation support system ascertains the machine resources and the number of cooperating agents and distributes the cooperation destinations to ASs with similar characteristics. This makes it possible to avoid concentrations of cooperation destinations.

5. Future plans

After making this ENCORE system, we plan to continue studies on the required parameters for transmitting information about one's own AS to the one-to-many cooperation support system, an agent matching method, and clarification of security, and so on, and then continue implementation aimed at achieving one-to-many cooperation. NTT Laboratories will continue to promote research and development of autonomous network management environments based on multiple agents as an extension of the ENCORE system.

References

[1] http://www.ntt.co.jp/news/news01e/0108/010830.html
[2] "OCN Introduces ENCORE Inter-AS Diagnostic System Targeting the Construction of an Internet Environment with Outstanding Stability," NTT Technical Review, Vol. 2, No. 5, pp. 44, 2004.
[3] T. Oshima, M. Tahara, and K. Koike, "A study on the method to observe hijack route," Technical Report IEICE, TM2004-37, pp. 7-12, 2004.
[4] T. Oshima, M. Tahara, and T. Asano, "The method to observe the Internet fullroute for reachability improvement," Technical Report of IEICE, TM2003-72, pp. 125-130, 2003.

Toshimitsu Oshima
Research Engineer, Network Software Service Project, NTT Network Service Systems Laboratories.

He received the B.E. and M.E. degrees in system engineering from Hokkaido University, Hokkaido in 1999 and 2002, respectively. Since joining NTT Network Service Systems Laboratories in 2002, he has worked on operation support systems.

Mitsuho Tahara
Research Engineer, Network Software Service Project, NTT Network Service Systems Laboratories.

He received the B.E. and M.E. degrees in electronic engineering from the University of Tokyo, Tokyo in 1995 and 1997, respectively. Since joining NTT Network Service Systems Laboratories in 1997, he has worked on operation support systems. His research interests include IP routing, MPLS, and VPN.

Kazuo Koike
Senior Research Engineer, Network Software Service Project, NTT Network Service Systems Laboratories.

He received the B.S. and M.S. degrees in applied mathematics engineering from Tokyo University of Science, Tokyo in 1986 and 1988, respectively. Since joining NTT Network Service Systems Laboratories in 1988, he has been engaged in R&D of operation support systems. He is a member of the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan.

Yoshihiro Otsuka
Senior Research Engineer, Supervisor, Network Software Service Project, NTT Network Service Systems Laboratories.

He received the B.E degree in electrical and electronic engineering from Kanazawa University, Ishikawa in 1983 and the M.E. degree in physical electronics from Tokyo Institute of Technology, Tokyo in 1985. Since joining NTT in 1985, he has mainly been engaged in research on broadband switching systems and network management systems. He is currently working on the modeling of network management functions and multi-layered network management systems. He is a member of IEICE.

Souhei Majima
Senior Research Engineer, Supervisor, Network Software Service Project, NTT Network Service Systems Laboratories.

He received the B.E. and M.E. degrees in information science from Fukui University, Fukui in 1982 and 1984, respectively. He joined the Electrical Communication Laboratories, Nippon Telegraph and Telephone Public Corporation (now NTT), Musashino in 1984. He has been involved in the development of a computer-aided software development environment (CASE) system and a switching equipment management system for large-scale networks. He is currently engaged in research on an NGN operation support system. He is a member of IEICE and the Information Processing Society of Japan.