INTRODUCTION

International Telecommunication Union — Telecommunication Standardization Sector (ITU-T) Study Group 13 has recently begun activities to develop standards for future telecommunication networks [1]. Study Group 13 has framed Question 21 to explicitly study the requirements of future networks and gradually standardize their functional architectures. This initiative is aided by many ongoing research activities on future networks such as FIND [2] and GENI [3] in the United States, several projects in Europe [4], AKARI in Japan [5, 6], and the Future Internet Forum in Korea (http://fif.kr).

Current packet-based networks, both the Internet and next-generation network (NGN), are based on two kinds of namespaces: domain names and IP addresses. Internet applications resolve the domain name into an IP address during a communication initialization phase via a Domain Name System (DNS) lookup. However, DNS has some inherent limitations. It is not suitable for the fast update and retrieval of hosts' dynamic information because of the existence of multiple cached copies in the global system of DNS servers. There are also problems in the use of IP addresses in the protocol stack. In particular, an IP address that is used in the network layer as a locator to locate the destination host and forward packets to it is also used as an identifier in the transport and application layers to identify the host or communication sessions. This dual use of IP addresses as IDs and locators makes it difficult to design efficient solutions for mobility, multihoming, renumbering, and security because such solutions require the capability of changing locators without changing IDs. To eliminate this problem, different approaches to introducing the ID/locator split concept in network architectures have been discussed recently [6–10]. An ID/locator split architecture uses distinct sets of values for IDs and locators, and allows the network layer to change locators without requiring the upper layers to change IDs to ensure that the communication sessions associated with the IDs are not interrupted.

ITU-T Study Group 13 has been discussing the ID/locator split concept and has recently approved a Recommendation (ITU-T Y.2015) that outlines the general requirements for introducing the concept in the NGN functional architecture. To contribute to ITU-T's effort, we first develop a new naming system, called Host Name and Identifier System (HNIS), for configuring hostnames and identifiers and map them to locators. We then propose a network architecture that is based on the ID/locator split concept and the naming system. The proposed architecture allows the network layer to change protocols and locators without disturbing the upper-layer communication sessions. This capability is helpful for designing efficient solutions for mobility, multihoming, routing, and security as well as for integrating heterogeneous network layer protocols.

RELATED WORK

Recently, the ID/locator split concept has been extensively discussed not only in ITU-T but also in the Internet Engineering Task Force (IETF) and Internet Research Task Force (IRTF). Several working groups (WGs) of the IETF and the Routing Research Group of the IRTF are actively involved in finding solutions for one or two of the problems related to the dual use of IP addresses as IDs and locators.
addresses. For example, to solve the multihoming problem, the Site Multihoming by IPv6 Intermediation (Shim6) WG has developed the Shim6 protocol [7]. In this protocol a host consistently uses one of the available IPv addresses as the upper layer ID (ULID) and changes the IP addresses (i.e., locators) used in the network layer when interfaces are switched. Six/One routers [8] extend the Shim6 concept to allow hosts as well as edge network operators to choose a provider’s network by ensuring that the edge routers have address translation capability. Similarly, for improving the security of the Internet, the Host Identity Protocol (HIP) WG has developed HIP [9] that uses public keys (and their hash values) and IP addresses as IDs and locators, respectively.

To make routing scalable, the Locator ID Separation Protocol (LISP) [10] has been considered by the LISP WG. LISP uses prefix-aggregatable endpoint identifiers (EIDs), which can be mapped to routing locators (RLOCs) in ingress tunnel routers by consulting a mapping system. LISP does not consider hostnames. It assumes hierarchical EIDs, it does not contain information on how such EIDs are generated and mapped to hostnames. On the other hand, HIP assumes that DNS servers store and provide information for hostname IDs and IDs (and locators). However, we suspect that DNS may not be able to accommodate mapping information for a huge number of hosts such as sensors and mobile devices that would be connected to future networks. HIP assumes that every host has a public key as its ID. While this assumption helps secure communications, it may not be applicable in a general case where many small hosts support only lightweight communications. To overcome these limitations, we propose the use of HNIS as a naming system in the ID/locator split architecture.

HOSTNAME AND IDENTIFIER SYSTEM

In this section we discuss the configuration of hostnames and IDs. A hostname and a host ID play similar roles in that both identify the host. The differences between them lie in their structure and usage. Hostnames are usually denoted by variable-length character sets that can be read and remembered by humans, while host IDs are denoted by fixed-length bit strings that cannot be memorized by humans. Hostnames are used during a communication initialization process to find locators, and authenticate and authorize hosts. On the other hand, host IDs are used as control information in communication protocols and packet headers to identify sessions, packets, or communication endpoints [6].

Figure 1 shows the configuration of global hostnames and IDs. A global hostname is formed by concatenating a local hostname and domain name using the concatenation symbol "#." The local hostname is generated from a combination of the host’s attributes such as usage, owner, location, serial number, and installation date and time. Examples of a local hostname are my-pc-20090731 or sensor-temp-room-5-202. The local hostname is unique in the administrative domain with which the host is logically associated. The administrative domain is represented by its domain name, which is globally unique. The global hostname is in a format similar to my-pc-20090915#mydomain.com or sensor-temp-room-5-202#my-domain.com.

A host ID is generated by concatenating prefix, scope, and version fields with the cryptographic hash value of the global hostname and an additional parameter. The prefix is used to aggregate host IDs that refer to a specific context and simplify their resolution. A host can derive different versions of IDs from a single global hostname by using different values of the parameter in the hash function, such as SHA-1 or MD5. The scope field indicates the validity and scope of the ID. It may indicate whether the ID is private, public, local, or global. The public and global IDs are stored in public registries, whereas private IDs that are used for private communications are kept private by the host.

As shown in Fig. 1, the host ID is dynamically mapped to different locators, depending on the host’s location and network association. In the case of mobility, a host ID is mapped to two different locators at different instances. In contrast, in the case of multihoming, the host ID is simultaneously mapped to two or more locators.

PROPOSED NETWORK ARCHITECTURE

ARCHITECTURAL LAYOUT

Figure 2 shows the layout of the proposed network architecture. It mainly comprises edge (or access) networks, a global transit network, and a unified logical control network [6].

The edge networks provide network access to various end systems or hosts. They can be as dynamic as a wireless sensor network, ad hoc network, vehicular network, or moving network whose topology changes frequently due to mobility or intermittent connectivity. The edge networks are connected through one or more gateways to the global transit network.

The global transit network, which is a collec-
The global transit network, which is a collection of backbone networks of Internet Service Providers, has a relatively stable configuration. The transit network includes backbone routers to forward packets from the source edge network to the destination edge network.

We call the locators used in the global transit network global locators, whereas those used in an edge network are termed local locators. The global transit network has only one network layer protocol and one namespace for global locators (e.g., it may use IPv6 addresses as locators). However, the edge networks may have many different network layer protocols and local locator spaces; for example, they may use IPv4 or IPv6 addresses, geographical coordinates, or administrative divisions as locators. However, the edge networks may have many different network layer protocols and local locator spaces; for example, they may use IPv4 or IPv6 addresses, geographical coordinates, or administrative divisions as locators. When the local locator spaces used in the edge networks are different from the global locator space, the gateways have to translate local locators into global locators (and vice versa) for packets passing through them. For this purpose, the gateway maintains information such as the host IDs and locators of all the local hosts as well as the host IDs, locators, and gateways of all the remote hosts — also called correspondent hosts — with whom the local hosts are communicating.

The third component of the architecture is the unified logical control network. It contains the domain name registries (DNRs) and ID registries (IDRs) for resolving domain names and distributing ID/locator mapping or binding information, respectively. In addition, it may also contain additional logical networks (which are outside the scope of this article) for maintaining and distributing information on authentication, authorization, and accounting (AAA), policy profiles, network configuration, QoS control, and so on.

The DNR stores information on the binding between domain names and the IDs and locators of host name registries (HNRs) that manage the domain names. The binding information about the HNRs stored in the DNR (also called DNR record) does not change frequently because the HNRs are generally fixed nodes that are not changing their locators. The DNRs can be organized in a hierarchical structure, similar to that of DNS, for storing and retrieving static mapping information. Note that DNRs store only the domain names assigned to HNRs, not the global hostnames of all nodes. Thus, the DNR record size does not grow as fast as the number of hosts. The smaller the size of the DNR mapping table, the faster the search and retrieval process for DNR records.

The IDR stores and distributes information on the binding between host IDs and locators of all the active hosts (i.e., hosts involved in communication sessions through the gateway). The binding information is provided by the gateway, which collects the information by actively interacting with hosts or passively observing host behavior (i.e., when a host initiates a session or enters from another network). The gateway also stores a copy of the binding information and uses it to translate protocols or locators. Whenever the gateway needs to translate a protocol or locator, it looks up the information in the IDR.
uploads ID/locator binding updates to the IDR, the updated information passes through the logical control network of IDRs to the gateways and hosts located in remote edge networks. The IDRs thus help the network to adapt to changes in network conditions and mobility. Moreover, they also support fault tolerance. In other words, when a gateway crashes, a substitute gateway can download the binding information from the IDR to resume communications.

The **HNR** stores information on the binding between hostnames and IDs, locators, and security keys of the hosts that belong to the administrative domain managed by the HNR (i.e., hosts whose global hostnames contain the domain name managed by the HNR). The hosts register their information with the HNR when they first connect to an edge network (i.e., when the user of the host subscribes to the communication service). The hosts also send update requests to the HNR when they change their IDs, locators, or other information. The HNR thus stores dynamic host information that changes often due to mobility or the activation of different interfaces.

**IDENTITY SUBLAYER IN PROTOCOL STACK**

In the new architecture we introduce a shim layer, called identity sublayer, between the transport and network layers of the host protocol stack. The function of the identity sublayer is similar to that of the HIP identity layer [9] in the sense that the identity sublayer maps host IDs to locators. The identity sublayer separates the transport and upper layers, in which host IDs are used for host or session identification, from the network layer, in which locators are used for finding host location and forwarding packets through the network. The difference from HIP lies in the fact that we attach an identity header to both data and control packets, whereas HIP attaches the HIP header only to control packets and the IPsec header to data packets for packet identification. IPsec is mandatory for HIP implementation. We use IPsec or cryptographic security mechanisms only when the applications require them.

**GATEWAY OPERATION**

In the proposed architecture, gateways connecting the edge networks to the global transit network mainly perform two tasks: translating network layer protocols or locators, and updating the ID/locator mapping records of IDRs. The gateways also perform the identity sublayer functions. They use the host IDs present in the identity header of packets as reference values to translate the network layer protocols or locators. Similarly, when the gateway detects a change in the ID-to-locator mapping information of the hosts that are connected to the edge network, it uploads the mapping updates to the IDR.

**COMMUNICATION PROCESS**

The communication process involves two phases: hostname resolution and data communication. Figure 3 shows the hostname resolution procedure when a correspondent host (CH) with the global hostname ch#domain-2.com wants to communicate with a mobile host (MH) with the global hostname mh#domain-1.com. For the hostname resolution to be successful, the HNR of the MH (HNR_MH) must have registered its domain name, ID, and locator in the DNR, and the MH must have registered its hostname, ID, and locator in the HNR (shown by the solid arrows 1 and 2, respectively).

The CH gets the hostname to ID and locator mapping of the MH by querying the DNR and HNR_MH. The CH first acquires HNR_MH's IDs and locators from the DNR by resolving the domain name part of the MH's global hostname. The CH then sends a hostname lookup query to HNR_MH. HNR_MH searches for the hostname in its record to find the MH’s ID and locator, and subsequently relays the query to the MH. Upon receiving the hostname resolution query through the HNR, the MH selects its ID and locator (in case the MH is multihomed) on the basis of the type of communication service required between the MH and CH. The MH then replies to the CH by sending its ID and locator.

The hostname resolution process provides both the CH and MH with each other’s hostnames, IDs, and locators. In the above example the CH obtains the MH’s hostname, ID, and locator from the hostname lookup query message, while the CH acquires the MH’s hostname, ID, and locator from the response message. After obtaining the MH’s hostname, ID, and locator, the CH can either directly start data communication, as shown by the thick arrow (6) in Fig. 3, or start communication after exchanging additional signaling messages for more secure communications. We call the former and latter modes of communication lightweight communication and secure-mode communication, respectively. The CH indicates its choice of the mode of communication in the hostname resolution request message. If secure-mode communication is requested, the MH responds with its public key and certificate obtained from a certifying authority, along with its hostname, ID, and locator. After having exchanged a few additional messages, the MH and CH establish security contexts that are required for protecting the integrity or privacy of the communication session.

---

**Figure 3. Procedure for hostname registration and resolution. Broken arrows represent steps for domain name and hostname registration, while solid arrows represent steps for hostname resolution.**
The proposed architecture has two features that aid mobility management. First, it allows network layer protocols to change locators without necessitating a change in the IDs of the upper layer protocols. Second, the logical control network of IDRs facilitates fast location updates.

The components involved in mobility management are shown in Fig. 4. Let us assume that the MH is located in its home network when it starts communication with the CH. At this time, the home network gateway (GW_HN) stores the following binding information: MH[ID, LOC], CH[ID, LOC], and GW_CN[ID, LOC], where Z[ID, LOC] represents the ID and locator of node Z. GW_HN uploads this information along with GW_HN[ID, LOC] to the home network IDR (IDR_HN). IDR_HN uses GW_HN[ID, LOC] to clearly recognize the gateway through which the MH is connected to the global transit network (in case multiple gateways connect the edge network to the global transit network). The correspondent network gateway (GW_CN) also stores similar information, i.e., CH[ID, LOC], MH[ID, LOC], and GW_HN[ID, LOC], and the correspondent network IDR (IDR_CN) stores GW_CN[ID, LOC] in addition to this information.

Let us assume that the MH moves to a foreign network. As soon as GW_HN detects the movement of the MH to the foreign network, it requests IDR_HN to transfer a copy of the MH’s binding record stored in IDR_HN to the foreign network gateway (GW_FN) via the logical control network of IDRs. Alternatively, if GW_FN detects the MH in its network before receiving the MH’s binding record from IDR_HN, it requests the foreign network IDR (IDR_FN) to obtain the MH’s ID/locator bindings from IDR_HN. Simultaneously, GW_HN also starts buffering data packets to be sent to the MH. When the MH configures/obtains a new locator in the foreign network, it informs GW_FN of the new locator. GW_FN updates its record relating to the MH’s binding and forwards a binding update to IDR_FN to register the bindings in the latter’s records. IDR_FN sends the binding update to IDR_HN and IDR_CN in order to update their records. IDR_HN also forwards the binding update to GW_HN, which then sends the buffered packets to the MH via GW_FN. Similarly, IDR_CN also forwards the binding update to GW_CN, which updates its record and relays the binding update to the CH. The CH then uses the MH’s new locator as the destination locator in packets. The binding information of the MH stored in the MH’s HNR (HNR_MH) can be updated either by GW_HN, which receives the updated binding from GW_FN via the IDR logical control network, or by the MH itself via explicit signaling.

MULTIHOMING SUPPORT

The new architecture supports both host and site multihoming. In host multihoming a host has two or more interfaces connected to the same or different edge networks, while in site multihoming a site has two or more interfaces connected to the same or different edge networks.
ing an edge network is connected to multiple ISPs through one or more gateways.

The architecture enables a host to switch interfaces (or locators) without interrupting communication services. For this purpose, multihomed hosts exchange lists of their available locators during the hostname resolution phase; the preferred locator is that used for default communication. The identity sublayer thus maintains information on the binding of host IDs with the list of available locators. Whenever the host wants to change its preferred locator, it sends a locator update request message to the peer host. Upon receiving the message, the corresponding host or gateway acknowledges the sender by sending a locator update response and uses the new default locator as the destination locator in the subsequent outgoing packets.

Multihomed gateways exchange their available locators via the IDR logical control network. The gateways can change the destination locators of outgoing packets for forwarding the packets through different ISP networks. At the receiving gateway, the packets are identified by the IDs present in the identity header and are forwarded to the destination host’s local locator. Similarly, the gateway can change the source locator of outgoing packets in order to receive the response packets from the global transit network through a preferred link.

SCALABLE ROUTING

The new architecture makes routing scalable by allowing the use of different locator spaces in the edge networks and the global transit network. Consequently, the growth of the global routing table can be controlled because fewer global locators need to be stored in the routing table, even for supporting multihoming and provider-independent addressing in the edge networks. The gateways hide the effect of changing the local locator space of the edge networks from the global routing system.

The proposed architecture offers mobility, multihoming, and scalable routing by introducing protocol or locator translation functions in gateways and using the IDR logical control network for distributing the ID/locator binding information. Future challenges include implementing the protocol or locator translation function and ID-to-locator mapping function system scalable. Since the gateways are located at the edge of the Internet, we presume that the translation function scales in a manner similar to the currently used network address translators (NAT). Some protocols proposed in previous studies, such as Six/One routers [8] and LISP [10], are also based on a similar assumption. To make the ID-to-locator mapping system scalable, we intend investigating various hierarchical and flat architectures of the IDR logical control network in a future study.

IMPLEMENTATION

We have implemented the basic components of the proposed ID/locator split architecture on Linux. We have created new application program interfaces (APIs) that identify sockets by using IDs, instead of IP addresses. Figure 5 shows a block diagram of the implemented software components. The state manager stores information on session states, such as IDs, locators, and hash functions associated with a session. The signaling block carries out the functions required to resolve global hostnames into IDs and locators, initiate communication with peer hosts, and manage mobility and multihoming. The network manager monitors the status of interfaces (i.e., locators) and signals the state manager to update information on the currently active locators. The session manager carries out the ID/locator mapping functions that is, it creates packets by attaching the identity header to the application data and the signaling message packets. Additionally, it also attaches the network layer (i.e., IP) header to the packets and forwards them directly to the network interface drivers through raw sockets. By forwarding or receiving packets through the raw socket, the session manager conceals the existence of the IP or IPv6 addresses in the network from the transport and application layers. In this way, the implementation supports both IPv6 and IPv4 addresses (i.e., heterogeneous protocols and locators) in the network layer without requiring the transport and application layers to contain network layer protocol-specific information.

Figure 6 shows the implemented components and an experimental setup. The IDR functions and gateway functions are implemented in the same physical node, whereas the HNR functions are implemented in separate nodes (i.e., one in each edge network). Using the setup, we measured the time taken by the architectural functions. For example, the hostname resolution took approximately 280 ms, and the round-trip time (RTT) for data communication between host 1 and host 3 was 2.8 ms. The change in the RTT was insignificant (<0.001 ms) when the gateways translated only locators and when they translated the network layer protocols. Similarly, for the mobility of host 1 from edge network 1 to 2, it took around 8 ms to prepare and transmit an ID/locator binding update request and around 4 ms for the request to reach host 3 via the IDRs. That is, the network layer handover delay was around 12 ms.
We intend to evaluate the scalability of the mapping functions at gateways as well as the scalability of the logical control network for the secure distribution of the ID/locator mapping information in the network.

CONCLUSION

In this article we propose a new naming system that facilitates efficient hostname resolution in future networks. We demonstrate the use of the naming system in the ID/locator split architecture for supporting mobility, multihoming, and scalable routing. In a future study we intend to focus on how the architecture scales. In particular, we intend to evaluate the scalability of the mapping functions at gateways as well as the scalability of the logical control network for the secure distribution of the ID/locator mapping information in the network.

ACKNOWLEDGMENT

The authors thank the AKARI Architecture Design Project members for their valuable suggestions on the research content presented here. They are also grateful to Dr. Yoichi Maeda and Dr. Mostafa H. Sherif, General Chair and Program Committee Chair, respectively, of the 2nd ITU-T Kaleidoscope event on Innovations for Digital Inclusion, 2009, for their contribution to the publication of this article.

REFERENCES


Figure 6. Experimental layout.

BIographies

VED P. KAFLE [M’04] (kafle@nict.go.jp) received his B.E. degree in electronics and electrical communications from Punjab Engineering College (now PEC University of Technology), Chandigarh, India. He received his M.S. degree in computer science and engineering from Seoul National University, South Korea, and his Ph.D. degree in informatics from the Graduate University for Advanced Studies, Japan. Since 2006 he has been working as a researcher at the National Institute of Information and Communications Technology (NICT), Tokyo. He is a member of the AKARI Architecture Design Project for New Generation Networks. He is involved in the designing of new-generation network architectures and protocols. In particular, his current research interests lie in node identification and location architectures, scalable routing, integration of heterogeneous network layer protocols, and mobility management in new-generation networks and the future Internet. He received the ITU Association of Japan Award in 2009 and the best paper award (second prize) at the ITU-T Kaleidoscope event on Innovations for Digital Inclusion, 2009.

HIDEKI OTSUJI (ejj@nict.go.jp) received his B.E. and M.E. degrees from the Musashi Institute of Technology, Tokyo, Japan, in 1989 and 1991, respectively. He received his D.E. degree from the Tokyo Institute of Technology in 2002. He joined the Communication Research Laboratory of the Ministry of Posts and Telecommunication (now NICT) in 1991. He is currently a senior researcher and is involved in designing new-generation network architectures. In the past he has been involved in research on intelligent network and in the development of the Intelligent Network Service Simulator for B-ISDN. He has also contributed to the development of a testbed with ATM switches. He has carried out several experiments on ATM satellite communications. His research interests lie in active networks, GMPLS protocols, standardization of new technologies, and interoperability testing of different optical technologies.

MASUDI INOUE [M’91] (inoue@nict.go.jp) received his B.E. degree from Kyoto University in 1992, and his M.E. and D.E. degrees from the University of Tokyo in 1994 and 1997, respectively, all in electrical engineering. He is currently a research manager in the Network Architecture Group at NICT. He is also a member of the AKARI Architecture Design Project for New Generation Networks. He has been engaged in R&D of ultra-high-speed WLANs, future mobile and sensor access networks, and ID/locator split architectures. He was a visiting researcher at the Polytechnic Institute of New York University, Brooklyn, in 2000. He is a member of IEICE, the New Generation Network Promotion Forum, the Ubiquitous Networking Forum, and the Next Generation IP Network Promotion Forum. He received the Best Paper Award from the Information Processing Society of Japan (IPSJ) in 2006 and 2007, and the Young Scientists’ Prize of the Commendation for Science and Technology by the Ministry of Education, Culture, Sports, Science and Technology in 2007.