Development of an End-to-End Communication Adapter and Implementation

Fumiya Ogyu*, Hidekazu Suzuki*, Katsuhiro Naito† and Akira Watanabe*
*Graduate School of Science and Technology, Meijo University,Aichi 468-8502, Japan
†Faculty of Information Science, Aichi Institute of Technology, Aichi 470-0392, Japan
Email: 173426006@ccalumni.meijo-u.ac.jp, hsuzuki@meijo-u.ac.jp,
naito@pluslab.org, wtnbakr@meijo-u.ac.jp

I. Introduction

With the development of high-speed wireless technologies and the rapid spread of mobile terminals, opportunities for users to use the Internet have been drastically increasing. The present network base including the Internet is mostly realized through IP networks, and it is considered that the communication infrastructure in future will also be developed on the premise of using IP networks.

However, we have the following problem associated with the IP networks. Namely, in the case of IPv4 networks, it is not possible to initiate communication from the side of global address area to the side of IPv4 private address area (i.e. the so-called NAT traversal problem). In addition, There is a demand for "mobility" that allows communication to continue even if the network is switched to another network.

In this paper, we call the type of communication that is not affected by any of the above-said restriction "end-to-end communication". "NTMobile" has been proposed as a protocol to realize the end-to-end communication [1]. For this type of communication, it is necessary to incorporate a communication library called "NTMfw" in the end node in order to give functions of NTMobile to the end node [2]. NTMfw offers API for NTMobile communication. For applications to be newly developed, end-to-end communication can be realized by using this library. However, as regards built-in type home appliances and application servers, it is not possible to rewrite programs after the manufacturing at factories.

Thus, we propose a special type of adapter that transforms normal communication to NTMobile communication (this adapter is hereinafter referred to as "NTMA"). We hereby report that we implemented our proposed system on Linux, and verify its behavior and evaluated its performance.

II. NTMobile

NTMobile is composed of NTM nodes in which NTMfw is installed, and Direction Coordinator (DC)which manages real IP addresses and virtual IP addresses and directs communication routes. NTM nodes can be existed in any network area, whereas DC needs to be placed in the global network area.

NTMobile allocates a virtual IP address (which does not depend on its location) to each NTM node, and the application performs communication based on the virtual IP address. DC has not only the function of allocating virtual IP addresses among NTM nodes but also directs the most appropriate communication route towards NTM nodes. Packets based on the virtual IP address created by the application are encapsulated by the real IP address and delivered to the corresponding node. Even if the end node switches its network to another network during communication, the application which is making communication based on the virtual IP address can continue its communication because the virtual IP address does not change, while the real IP address does.

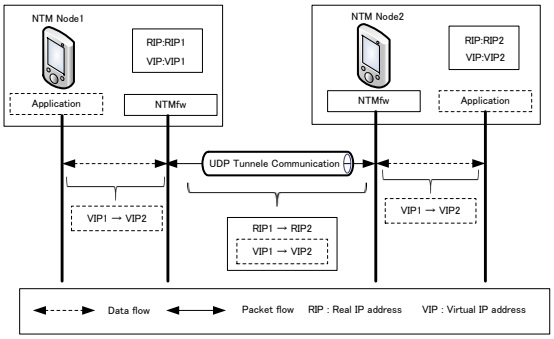


Fig. 1. Sequence of address changes in NTMobile communication.

III. NTMfw

NTMfw is a communication library operating in the application layer, and it provides API for NTMobile communication to higher level applications. Fig. 1 shows the sequence of changes in the addresses for NTMobile communication. The application for NTM node 1 creates a packet by using virtual IP addresses. This packet is encapsulated by the real IP addresses and transmitted to NTM node 2. NTM node 2 decapsulates the received packet by NTMfw and takes out the packet based on the virtual IP addresses, which is then delivered to the application of NTM node 2.

IV. Issues associated with NTMobile

NTMobile communication is realized by installing NTMfw in end nodes. However, in the case of built-in type home appliances, there are cases where NTWfw cannot be additionally installed after their manufacturing at factories. And, while it is necessary to modify the existing application program in order

to incorporate NTMfw in general end nodes, there are certain communication devices for which stability is quite important, and any modification of the program is inhibited, like the case of application servers.

V. Our proposed method

NTMA is placed adjacent to the communication device which cannot install NTMfw, and performs the function of NTMobile on its behalf. NTMA is expected to be used by both sides of the initiator that starts communication and the responder that waits for receiving communication. This time, we verified the behavior of the system for the case of placing NTMA on the side of the initiator.

A. Network configuration and preconditions

As NTMA, two pieces of physical NIC need to be prepared. One NIC is bridge-connected with a general end node (GN) and the other one is bridge-connected with the side of the Internet. The NTM node (which becomes the responder) and DC are assumed to be placed on the Internet.

The NTM node and NTMA together perform registry processing against DC at the time of starting up communication, and obtain allocation of a virtual IP address. As the IP address of GN, the virtual IP address allocated by DC to NTMA is used.

B. NTMA's communication sequence

Fig. 2 shows the communication sequence of our proposed system. At the time of starting up, GN designates FQDNcn as its corresponding node and sends DNS query to the network. Upon receiving the query, NTMA creates a tunnel between NTMA and the NTM node by way of signaling processing, using the function of NTMfw. After the completion of tunnel creation, NTMA receives the virtual IP address of the NTM node and sends it to GN as a part of DNS response. Then, GN recognizes the corresponding node as the virtual IP address of the NTM node. As the next step, GN sends a packet to the virtual IP address of the NTM node. NTMA, upon receipt of the packet, encapsulates it using the real IP address of the NTM node and NTMA and sends it to the NTM node. In the case when a packet is sent back from the NTM node, NTMA decapsulates the packet, transforms it to a normal communication packet, and sends it to GN.

VI. Implementation

NTMA receives a packet sent from GN, using its RAW socket. "RAW socket" is a socket that can send/receive packets, including MAC header. NTMA encapsulates it by use of Remodeled NTMfw (R-NTMfw) and sends it to the NTM node. R-NTMfw is a new library in which a part of NTMfw is remodeled for NTMA. NTMA also decapsulates the packet from the NTM node, by use of R-NTMfw and sends it to GN, using its RAW socket.

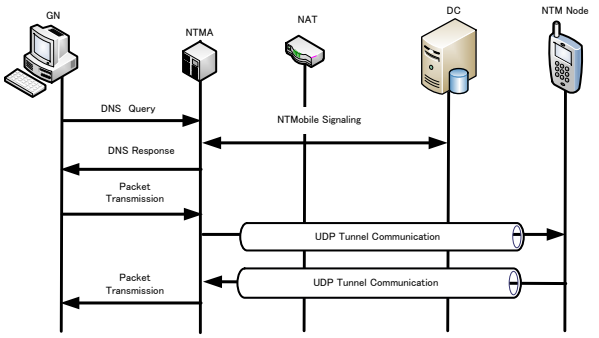


Fig. 2. Communication sequence of our proposed system.

TABLE I
Results of NTMA's relaying time measurement.

Relaying Time	Time(us)
Packet relaying time	218.5
Response packet relaying time	324.0

VII. Evaluation

We sent/received packets between GN and the NTM node, and measured the processing time at NTMA. Table 1 shows the results of our time measurement. The "packet relaying time" means the time required for the processing of NTMA's receiving a packet from GN and relaying it to the NTM node, whereas the "response packet relaying time" means the time required for the relaying of the response packet from the NTM end node to GN.

The time required for the passing through NTMA was less than 400us for both cases, and accordingly, we confirmed that the required time to pass through NTMA did not cause any problem from the practical point of view.

VIII. Conclution

We developed a device, which we named "NTMA", to be placed adjacent to a general node when NTMobile system cannot be installed in the end node, and which performs the function of NTMobile on its behalf. By using our proposed method, we confirmed that it is possible to give NTMobile function to general nodes without making any modification to them. We verified that NTMA performs our expected behavior, by implementing it on Linux as a trial. In addition, we evaluated the performance of our proposed method and confirmed that the packet relaying time was quite short.

References

[1] H. Suzuki, et al., "NTMobile: new end-to-end communication architecture in ipv4 and ipv6 networks", in Proc. ACM MobiCom 2013, pp. 171–174, 2013.

[2] H.Nodo, et al., "Practical Realization of NTMobile Framework that makes Flat Networks", IPSJ UBI Technical Report, Vol. 54, No. 46, pp. 1-8, 2017

Development of End-to-End Communication Adaptor and Implemantation

† Fumiya Ogyu, † Hidekazu Suzuki, † Katsuhiro Naito, † Akira Watanabe Email : fumiya.ogyu@wata-lab.meijo-u.ac.jp (fumiya ogyu)
† Graduate School of Science and Technology, Meijo University, Japan † Information Science, Aichi Institute of Technology, Japan

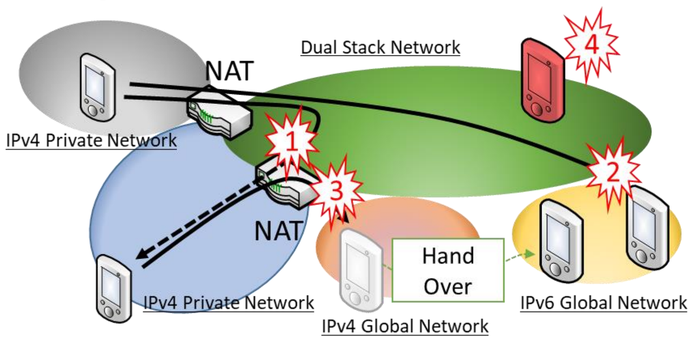
Introduction

NTMobile (Network Traversal with Mobility) has been proposed as the technology that can solve every problems of the current network.
Users can develop applications using the communication library so-called NTMobile framework (NTMfw).
However, NTMobile is not available for communication devices which can not implement NTMfw.
In this research, We have developed NTMA (NTMobile Adapter) and enabled general terminals to use NTMobile functions.

The Real Network

☐ The Real Network has following issues

1. Communication can not be started from the global address area to private address areas. (NAT traversal problem).
2. IPv4 terminal and IPv6 terminal can not communicate directly.
3. Communication stops when the network changes.
4. There are such fears like eavesdropping , tampering , spoofing attacks etc...



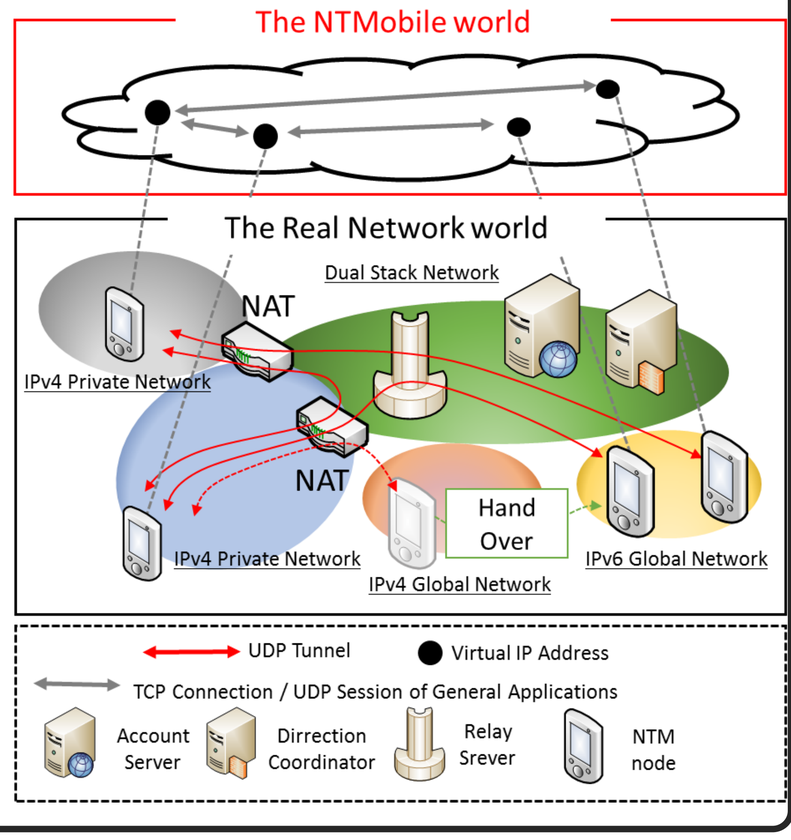
NTMobile (Network traversal with Mobility)

☐ NTMobile is mobility protocol in the IP layer

- Applications establish connections based on Virtual IP address.
- Virtual IP packets are encapsulated by Real IP address and sent to the real network.

☐ These issues are solved by NTMobile

- Bidirerectional end-to-end communication.
- Mobility transparency during the communication.
- Secure cipher communication.

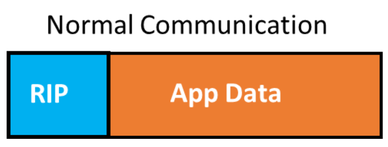


NTMobile communication packet

☐ In the case of general communication

Role of RIP (Real IP Address)

- Identifier of the location of the terminal.
- Identifier of the terminal.



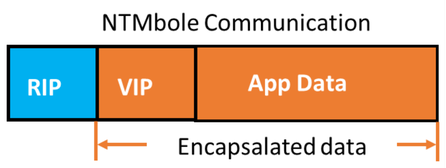
☐ In the case of NTMobile communication

Role of RIP

- Identifier of the location of the terminal.

Role of VIP (Virtual IP Address)

- Identifier of the terminal.

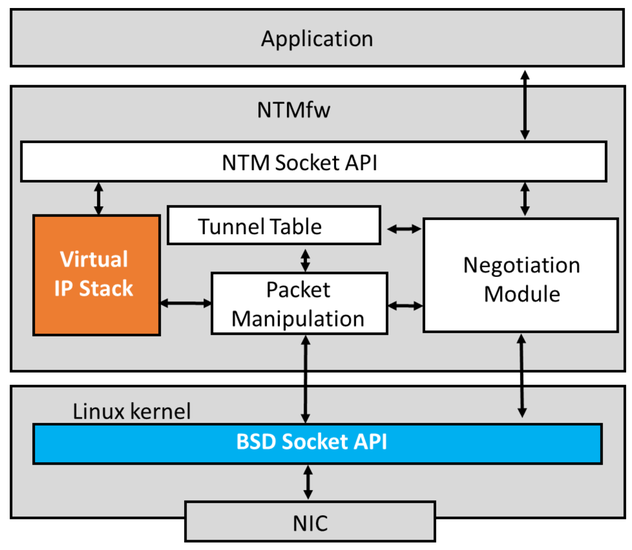


NTMobile framework library (NTMfw)

☐ What is NTMfw?

- **NTMfw is the library that provides NTMobile functions to applications.**

- Application uses NTMobile socket API.
- Virtual IP header is made by Virtual IP Stack.
- Real IP header is made by Linux kernel.



Issues of NTMfw

☐ NTMfw can not be used on OS other than Linux

ex) Windows

- the socket APIs are different from Linux.

☐ There are terminals that can not implement NTMfw

ex) Built-in appliances

- Program can not be changed after factory shipment.

Application servers

- Adding new functions are not permitted from the viewpoint of stability.

Watanabe Lab.

Proposed method (NTMA)

☐ What is NTMA?

- **NTMA is the communication device that executes NTMobile on behalf of a general terminal.**

- It is set next to a general Node (GN)
- There is no need to change any existing programs.

☐ Implementation of NTMA

- **NTMfw is partly remodeled to realize the NTMA application.**

- Virtual IP address is set on GN.
- GN generates virtual IP packets.
- NTMA receives packets from GN with a RAW socket.
- R-NTMfw (Remodled NTMfw) generates NTMobile communication packets skipping Virtual IP Stack..
- Linux kernel encapsulates the above packets and sends them to the corresponding node.