Abstract

The usage of multiple interfaces in a host causes other problems due to the multiplicity of available interfaces. If interface is changed during communication, the communication is disrupted. Multiple interfaces in a host do not support the simultaneous usage of multiple interfaces. In this document, we discuss how to solve the problems of interface change and simultaneous usage of multiple interfaces and propose a virtual interface which hides the multiple interfaces from the IP stack of host. And we describe the applicability of virtual interface into the problem of multiple interfaces in multiple interfaces PS document. This document provides the implementation guidelines of virtual interface for a multiple interface host.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 28, 2011.

Copyright Notice
1. Introduction

In network environment of conventional TCP/IP stack suite, a communication entity usually has a wire connection with a single network interface and it is fixed. As an introduction of wireless technologies and heterogeneous access technologies, a communication entity is able to move around between different access technologies networks and have multiple network interfaces [I-D.ietf-monami6-multihoming-motivation-scenario].

Because conventional network applications and TCP/IP stack suite are developed for a communication entity which has a single network interface, the adoption of multiple network interfaces into a general communication entity makes some problems. Because of the usage of multiple interfaces during communication, there may be many considerations to support multiple interfaces in a host [I-D.ietf-mif-problem-statement].

If we use multiple interfaces, the change of interface may happen. In current OS/platforms, if an interface which is used to communicate is changed during communication, then the communication is disrupted. Interface change happens as various reasons; the host move to different networks, the using network interface is no longer available to use (e.g., H/W failure, unexpected wireless signal situation), by some policy, and by user experience.

Multiple interfaces in a host may not support the simultaneous usage of multiple interfaces. Even though a host can utilize multiple access technologies through multiple interfaces simultaneously, only one interface is chosen to communicate at a time in a conventional TCP/IP stack suite. The host wants to use multiple interfaces simultaneously to increase data bandwidth, reliability, and etc.

In order to solve the problems mentioned above, we propose a virtual interface for a host with multiple network interfaces. We currently use a virtual interface to provide the duplication of network connections through multiple network interface cards on an important network node such as a server. With a virtual interface scheme, the host with multiple interfaces can operate as it has a single interface irrespective to the number of network interfaces [I-D.ietf-netext-logical-interface-support-00].
2. Requirements Language

In this document, the key words "MAY", "MUST", "MUST NOT", "OPTIONAL", "RECOMMENDED", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [RFC2119].
3. Virtual interface for multiple interfaces

In some operating systems such as Linux (or Unix), most network interfaces, such as eth0, wlan, and ppp, are associated to a physical and/or logical devices that are in charge of transmitting and receiving data packets. However, there are exceptions to this rule, and some logical network interfaces do not feature any physical packet transmission.

The virtual interface is not a real physical device and it is a logical network interface. It has connections with physical network devices within a network entity and the path between the virtual interface and real physical network devices is determined dynamically according to some policy.

The virtual interface is registered to the network layer and it is regarded as a general network interface. Then real physical interfaces are connected to the virtual interface. The network layer does not know the existence of these physical interfaces.

The virtual interface can be used for the duplication of network connections (the duplication of network interface cards) for fault tolerance or load sharing. If an important server has multiple physical interface cards, it can survive even though one interface card is down. It can keep a communication session with other living interface cards. In this case, the presence of multiple interfaces can be hidden to network layer and network layer regards the virtual interface as a general network interface. The conventional network applications and network modules such as TCP/IP stack suite do not need to be modified to support multiple interfaces in a host if a virtual interface is used.
4. The usage to use a virtual interface in a host

4.1. Configuration of a virtual interface in a host

In the following figure, network interfaces if_1, if_2 are real physical interfaces. The network interface VI is a virtual interface. The virtual interface is connected to the physical interfaces and it is shown to the network layer. In this figure, the network layer uses the virtual interface VI instead of physical interfaces if_1, if_2. The host needs a specific module (e.g., connection manager) to manage the virtual interface and select the path between the virtual interface and physical interfaces.

![Architecture of a virtual interface](image)

Figure 1: Architecture of a virtual interface in a host with two physical interfaces

4.2. Operations of a host with a virtual interface

When a network module in a host starts, the virtual interface is configured to send and receive packets. In Figure 1, if the host uses a physical interface if_1, the path between the virtual interface VI and the physical interface if_1 is configured.

When sending packets to another node, packets are delivered to the virtual interface and these packets are also forwarded into physical interface if_1 according to the path configuration. When receiving packets from another node, packets are delivered to if_1 and these packets are also forwarded into VI according to the path.
configuration.

If the host changes another interface due to movement of a host or the failure of network interface, the host chooses a physical interface if_2 and then makes the path between the virtual interface VI and the physical interface if_2. At this time, the connection manager updates the relation between a destination address and a interface. When the host is sending packets to another node, packets are delivered to VI and these packets are forwarded into if_2 according to the path configuration. When the host is receiving packets from another node, packets are delivered to if_2 and these packets are also forwarded into VI according to the path configuration.
5. Functions and properties of virtual interface

In this section, we will describe the function of MIF virtual interface such as interface selection function and policy routing function, etc.
6. Use cases of virtual interface

In this section, we will describe how to use MIF virtual interface to address the issues that list in MIF PS draft such as source address selection and routing issues, etc.
7. Considerations for implementing virtual interface

7.1. All physical interfaces and virtual interface share the same link-layer identifier

The simplest way to implement virtual interface is to set the same link-layer identifier such as MAC address for both virtual interface and physical interfaces. To do that the devices should allow source address spoofing or allow changing link-layer identifier and the link-layer identifier format of access type should be similar.

The figure 2 shows an example scenario in which the virtual interface and physical interfaces use the same link-layer identifier ZZZ. When the physical interface 1 (if_1) and physical interface 2 (if_2) associated with Access Point 1 (AP1) and Access Point 2 (AP2) respectively, both AP1 and AP2 recognize the same link-layer identifier ZZZ from the MN’s attachments. The AP1 and AP2 can forward any packet sending from CN to the virtual interface or vice versa without any problem.

```
+-----+
| CN  |
+-----+
  /\  
+--------+------------------+
| WiMAX |     Virtual      |
|       |   Interface      |
|   if_1 |        MN        |
+--------+------------------+
          | (LL-ID: ZZZ)    |
+--------+------------------+
| AP1    |                  |
| (LL-ID: ZZZ) | (WiMAX) |
+--------+-------+
| AP2    |                  |
| (LL-ID: ZZZ) | (WLAN) |
+--------+-------+
          | (WLAN)  |
+--------+-------+
```

Figure 2: Virtual interface uses the same link-layer identifier with physical interfaces
In this scenario, to find a nearby router the MN send Router Solicitation (RS) message via both if_1 and if_2. The MN can receive Router Acknowledgement (RA) message from both of these two interfaces. In case that the MN wants to send a unicast packet but does not know the neighbor’s link-layer addresses, it will perform address resolution by sending Neighbor Solicitation (NS) message through both if_1 and if_2. On receiving NS message, the CN will send back Neighbor Acknowledgement (NA) message back to the MN through the path that it receives NS from. Thus the virtual interface at the MN may receive NA message through both if_1 and if_2. The same for the inverse direction, the CN node may sends NS message to the MN through both direction. On receiving the NS message the virtual interface at the MN will send NA reply message through the interface that it receives NS from. Some time the virtual interface may also send unsolicited Neighbor Advertisement message via all enabled physical interfaces in order to propagate new information quickly.

7.2. Virtual interface uses different link-layer identifier with physical interfaces

Many wireless devices do not allow source address spoofing or do not allow changing link-layer identifier. In addition some time different access devices use different link-layer identifier format. In these cases the physical interface and virtual interface cannot use the same link-layer identifier. In this case, the virtual interface can use totally different link-layer identifier from those of physical interfaces or it can only use the same link-layer identifier with one of the physical interfaces.
Figure 3: Virtual interface uses the same link-layer identifier with some physical interfaces

The figure 3 shows an example that the virtual interface uses the same link-layer identifier ZZZ with physical interface if_1. Physical interface if_2 uses a different link-layer identifier YYY. This implementation scenario can cause several problems:

The virtual interface cannot send packet to CN node via AP2 since AP2 can recognize the associated link-layer identifier YYY of if_2 only. It can send packet to the CN node only if the AP2 is placed in promiscuous mode. For example, if AP2 is WLAN Access Point, AP2 only recognizes the link-layer identifier YYY of if_2. If layer 2 frames come from the virtual interface uses link-layer identifier ZZZ, the WLAN AP2 cannot process these frame and discard them because only link-layer identifier YYY of if_2 is associated with WLAN AP2.

The CN node cannot send the packet to the MN via AP2 because the global IP address is assigned to virtual interface only. When the CN node sends NS message, the NA message will be sent by virtual interface with link-layer identifier ZZZ thus the NA message cannot be delivered through AP2, which understands only link-layer
identifier of if_2. However there is no problem if the virtual interface receives/sends NS/NA message via the AP1.

In this scenario, to find a nearby router the MN also send RS message via both if_1 and if_2. The MN can receive RA messages from both of these two interfaces. In case that the MN wants to send a unicast packet but does not know the neighbor’s link-layer addresses, it will perform address resolution by sending NS messages through both if_1 and if_2. On receiving NS message, the CN will send back NA message back to the MN through the path that it receives NS from. However, it is different with the first implementation method that the CN node cannot send NA message to MN through AP2 since the AP2 may understand and accept only the NA message that carries the link-layer identifier of the if_2. Thus the virtual interface at the MN may receive NA message only through if_1. In case of the inverse direction, the CN node may send NS message to the MN through both direction. On receiving the NS message the virtual interface at the MN will send NA reply message through the interface that it receives NS from. However the NA reply message sent via AP2 may not be forwarded to the CN node.

If we want to implement the virtual interface which is transparent from some specific access technologies (e.g., WLAN Access Point processes only layer 2 frame that the link-layer identifier is associated with itself), one of the possible approach is swapping the link-layer identifier of layer 2 frame between virtual interface and physical interfaces. Some specific access technologies may need the real(original) link-layer identifier of physical interfaces and it is difficult to modify layer 2 Access Point to process the layer 2 frame that the link-layer identifier is not associated with itself.
8. Security Considerations

TBD
9. IANA Considerations

This document has no actions for IANA.
10. References

10.1. Normative References


10.2. Informative References


Authors’ Addresses

Yong-Geun Hong
ETRI
161 Gajeong-Dong Yuseung-Gu
Daejeon, 305-700
Korea

Phone: +82 42 860 6557
Email: yonggeun.hong@gmail.com

Joo-Sang Youn
DONG-EUI Univ.
Busan,
Korea

Phone: +82 51 890 1993
Email: joosang.youn@gmail.com

Tran Minh Trung
ETRI
161 Gajeong-Dong Yuseung-Gu
Daejeon, 305-700
Korea

Phone: +82 42 860 1132
Email: trungtm2909@gmail.com

Dapeng Liu
China Mobile
Unit2, 28 Xuanwumenxi Ave,Xuanwu District
Beijing, 100053
China

Email: liudapeng@chinamobile.com