Discussion of MIF API

draft-liu-mif-api-extension-04

IETF 80

March 28, 2011

Dapeng Liu

MIF API Concept

```
+-------------------------------------+
|             Application             |
+-------------------------------------+
       ^           ^              ^
~~~~~~~|Socket Interface~~~~MIF |API~~~~~
       |           |              |
+------|-----------|--------------v-----+
|      V          TCP/UDP              |
+-----------------|-------------------+
                  |
+-------------------------------------+
|                  IP                 |
+-------------------------------------+
|         Physical Interface          |
|  +----+  +-----+  +-----+  +----+   |
|  |IF#1|  |IF#2 |  |IF#3 |  |IF#4|   |
|  +----+  +-----+  +-----+  +----+   |
+-------------------------------------+
```

Abstract MIF API:

- Advanced MIF API for app developer.
- Describe the abstract function of MIF API.

MIF API definition

- Address selection API
- Routing selection API
- Interface selection API
- Interface information API
- DNS selection API
- Notification API

1. Address Selection API

- **int setsockopt (int socket, int level, int option_name, const void *option_value, socklen_t option_len);**
  - The setsockopt() function shall set the option specified by the **option_name** argument, at the protocol level specified by the **level** argument, to the value pointed to by the **option_value** argument for the socket associated with the file descriptor specified by the **socket** argument.

    Examples of option: SO_DEBUG, SO_DONTROUTE, SO_SNDBUF…

  - Extend new option type used for address selection: **SO_ADDR.**
  - When setsockopt() set this parameter, the application will use the IP address specified by option_value as the source IP address.

2. Route selection API

- Extend new option type parameter for setsockopt() function:

- **SO_FSTHOP:** When this option is set, the packet will send via the gateway that specified by option_value.

3. Interface selection API

- Extend new option type parameter for setsockopt() function:

- **SO_INF:** When this option is set, the interface that specified by option_value is used.

- Discussion: Do we need to include the interface selection API in the draft?

4. Interface Information API

- Get Interface status/configuration API
- getInterfaceInfo()
  - get all interface's information and configuration
  - Need to define new structure, e.g.:

```
Struct InterfaceInfo{
char * name;          /*Interface name*/
struct parameter;     /*IP, MAC, addr Scope, Type…*/
    unit32_t status;}
```

- Discussion:
- Do we need the setInterfaceInfo() API?

5. DNS selection API

- Current version of gethostbyname()/gethostbyaddr() has the following definition:

  struct hostent *gethostbyname(const char *name)

- Propose to extend:

  struct hostent *gethostbyname(const char *name**, struct sockaddr *DNS_addr**)

  DNS_addr is used to specify which DNS server the application want to use for the DNS query.

- Discussion: Do we need the DNS selection API?

6. Notification API

- Sometimes, application need to know physical interfaces changing event.
  - For example: Apple platform provide mechanism that application can re-connect using new interface when a new interface is up and obtaining an IP address.

- Standardize those implementation in MIF API.
  - Example: new notification API to notify the event of interfaces changing.

- Discussion: Do we need to include this notification API in the draft?

Q & A