

Socket Application Program Interface (API) for Multihoming Shim

`draft-ietf-shim6-shim-api-01.txt`

Miika Komu
Marcelo Bagnulo
Kristian Slavov
Shinta Sugimoto

What is this document about?

- The document specifies a socket API which lets application have better control of:
 - Locator management
 - Failure Detection and Path Exploration (REAP)
- History:
 - Individual submission (June 2006)
 - Submitted as a WG item version 00, 01 (October 2006)
- Status:
 - Still in an early stage. Need more work.

Changes from previous versions (1)

Technical changes

- Determined to define the socket options at the level SOL_SHIM
- Defined a data structure for locator information
 - shim_locator{ }
- Defined a data structure for path exploration parameters
 - shim_pathexplore{ }
- Defined both read and write operation for each socket option (except for SHIM_ASSOCIATED socket option)
- Removed the descriptions about ‘stickiness’ of the socket options
 - Moved SHIM_FEEDBACK_POSITIVE and SHIM_FEEDBACK_NEGATIVE to Section 6
- Deprecated SHIM_IF_RECV and SHIM_IF_SEND (already covered by IPV6_PKTINFO, IPV6_PKTINFO)
- Gave a default value and an instruction how to disable each socket option

Changes from previous versions (2)

Editorial changes

- Rewrote Section 1 (Introduction) in a way that it gives clear statement what kind of application would need this API for what reason
- Added an usage (sample codes) of the socket options in Section 5

Things left & Next Steps

- Finalize the placeholder for locator information
- Specify negative/positive feedbacks
 - What exact information should be provided?
- Examine the impact to the existing socket APIs
- Open issues:
 - What if applications sharing the same shim context have different preference?
 - Error handling for invalid locator specified by an application
- Ask for more reviews by socket API experts and implementers

Thank you &
Any questions/comments?