

SCTP Update

Randall Stewart (rstewart@huawei.com)

Michael Tüxen (tuexen@fh-muenster.de)

Robin

- Happy Birthday!

draft-ietf-tsvwg-sctpsocket

- Issues brought up in discussion between the authors are resolved.
- A substantial amount of discussion happened between the authors regarding send()/recv() functions: sctp_sendmsg(), sctp_sendx, sctp_send(), sctp_recvmmsg().
- Resolution, agreed between the authors, but not included in current revision: sctp_sendv() and sctp_recvv().
- To Do:
 - Replace sctp_sendxxx()/sctp_recvxxx() by sctp_sendv()/sctp_recvv().
 - Review between the authors (another one).

draft-tuexen-tsvwg-sctp-sack-immediately

- Does not require to update RFC 4960 anymore, it just registers a bit in the the flags field of the DATA chunk.
- No other changes.
- Other criteria for setting the I-bit will be included in the next revision.

draft-tuexen-sctp-udp-encaps

- Usage scenarios:
 - Implement SCTP without modifying the kernel and without using raw socket.
 - Get SCTP through legacy NATs.
- Needed mechanisms:
 - Insert/Remove a UDP header between the IPv4/IPv6 header and the SCTP header.
 - Avoid IP-addresses inside the SCTP packets by using the same mechanism as the SCTP aware NAT. This procedure will be referenced.

draft-stewart-tsvwg-sctp-nonce

- Specifies ECN-Nonce for SCTP
- Should it proceed?

draft-stewart-natsupp-tsvwg

- A discussion was added on the impacts of this mechanism to interoperability.
- If one host does not support RFC 5061, associations are limited to be single homed.
- If both hosts support RFC 5061 and the internal host supports the NAT extension, communication will work fine. However, port number collisions can't will be avoided by the NAT box.
- If both hosts support RFC 5061 and both hosts support the NAT extension, communication will work fine. Even port number collisions can be handled.
- All major Unix implementations (FreeBSD, Linux, Mac OS X, Solaris) do support RFC 5061.