

RFC 3286 : An Introduction to the Stream Control Transmission Protocol (SCTP)

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 22 novembre 2006. Dernière mise à jour le 23 septembre 2010

Date de publication du RFC : Mai 2002

<https://www.bortzmeyer.org/3286.html>

Une des particularités du protocole IP est que vous avez plusieurs protocoles de transport disponibles. TCP et UDP sont les plus connus mais SCTP, présenté dans notre RFC, est également intéressant.

SCTP ressemble plutôt à TCP, notamment par le fait qu'il fournit un transport fiable. Mais il a plusieurs différences :

- il ne gère pas un flot d'octets continu mais une série de messages, bien séparés,
- il gère plusieurs flux de données séparés, qui partagent le même contrôle de congestion mais gèrent à part les pertes et retransmissions,
- il gère le cas où la machine a plusieurs adresses IP, ce qui lui fournit normalement plus de redondance, si on est connecté à plusieurs réseaux.

C'est le RFC 4960¹ qui normalise SCTP. Notre RFC est une introduction, donnant une vue générale du protocole.

SCTP est depuis longtemps mis en œuvre <<http://lksctp.sourceforge.net/>> dans Linux et, depuis peu également <<http://lists.freebsd.org/pipermail/freebsd-current/2006-November/067218.html>> dans FreeBSD. Sur Linux, si vous voulez tester SCTP, le plus simple est d'installer les outils du paquetage lksctp cité plus haut (sur Debian, `aptitude install lksctp-tools`, sur Gentoo, `emerge lksctp-tools`). Testez d'abord si votre noyau gère SCTP :

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc4960.txt>

```
# Bon
% checksctp
SCTP supported
```

```
# Mauvais
% checksctp
checksctp: Protocol not supported
```

Ensuite, désignez deux machines pour tester, mettons 192.168.2.1:6666 et 192.168.2.25:8888. L'une, 192.168.2.25 va écouter avec SCTP :

```
% sctp_darn -h 192.168.2.1 -p 6666 -H 192.168.2.25 -P 8888 -l
sctp_darn listening...
```

L'autre, 192.168.2.1, va envoyer le message toto :

```
% sctp_darn -H 192.168.2.1 -P 6666 -h 192.168.2.25 -p 8888 -s
sctp_darn ready to send...
192.168.2.1:6666-192.168.2.25:8888> toto
Recieved SCTP_COMM_UP
New connection, peer addresses
192.168.2.25:8888
192.168.2.1:6666-192.168.2.25:8888>
```

Le serveur a bien reçu le message :

```
Recieved SCTP_COMM_UP
NOTIFICATION: ASSOC_CHANGE - COMM_UP
SNDRCV
  sinfo_stream 0
  sinfo_ssn 0
  sinfo_flags 0x0
  sinfo_ppid 0
  sinfo_context 0
  sinfo_tsn 2215576728
  sinfo_cumtsn 0
  sinfo_assoc_id 1
DATA(6): toto.
```