

A Study of SCTP Services in a Mobile-IP Network

James Noonan, Philip Perry & John Murphy
Performance Engineering Laboratory
Dublin City University
{jnoonan,perry,murphy}@eeng.dcu.ie

Abstract

A recently standardized transport protocol - SCTP and the Mobile IP standard for IPv4 are examined. Both these protocols have been suggested as offering some additional functionality to mobile users over and above that which is currently available from TCP/IP. Here, the interaction of these two new protocols is investigated through the examination of traffic flows through a number of network topologies. The results seem to suggest that although these protocols do work together, they do not currently provide all the functionality that a mobile user might desire. In particular, some limitations caused by implementations are highlighted that could be remedied by small modifications without the need to change either standard specification.

1 Introduction

Transmission Control Protocol over Internet Protocol (TCP/IP) has been a phenomenally successful family of protocols. It has survived exponential growth in the Internet without significant modification, which is a testament to its robustness and ability to scale. However, a number of different groups have found it inappropriate for their needs. Multimedia communities find the congestion control mechanisms TCP employs are too restrictive. Wireless and mobile users also find that TCP reacts badly to losses and delay variations due to wireless links or handovers. The signalling community has found that TCP service is too restrictive for their needs. This has led to the introduction of new protocols and standards to the Internet and this paper studies how two of these interact.

As well as TCP, a second transport protocol, UDP (User Datagram Protocol) provides an unreliable datagram delivery service. TCP and UDP existed as the only IETF (Internet Engineering Task Force) standardised transport protocols up until late 2001. At this time the requirements of the signalling community resulted in a third protocol being standardised by the

IETF. This is known as Stream Control Transmission Protocol (SCTP) [1] and, although it shares many characteristics with TCP, it has many significant and interesting differences.

An entirely different group of researchers realised that wireless technology was going to play an increasingly important role for users accessing the Internet, which led to an examination of the problem of Internet Mobility. Without modification, it was clear that TCP/IP could not allow a user to roam between different networks while maintaining connectivity to the Internet. As a response, changes to the IP layer were developed that would enable TCP to operate unmodified, resulting in the standardisation of Mobile-IP (M-IP) [3].

M-IP was designed specifically with TCP in mind. This paper focuses on the effect of using SCTP instead of TCP over an M-IP network. It introduces some unusual configurations, which suggest that SCTP is worthy of further research in the mobile/wireless domain. Indeed, though the objective of the paper is to show that SCTP will operate in a M-IP network, initial experiences suggest that SCTP could potentially perform a greater role in Internet mobility. As a relatively new technology, SCTP will be discussed in some detail, followed by a more concise description of M-IP. Following this, 3 different experiments are described and the results are analysed in the context of using SCTP in the mobile arena.

2 Stream Control Transmission Protocol

2.1 Introduction to SCTP

SCTP was designed to be a general-purpose transport protocol for message-orientated applications [2]. It is envisaged to be a means of replacing SS7 networks using an IP based network. The reason for this is the popularity of IP networks, and a desire to converge IP with the current signalling networks. SCTP is only the third Transport Protocol to be standardised by the Internet Engineering Task Force (IETF).

SCTP, being a Connection-Orientated Protocol, shares many characteristics with TCP. Two end-points must establish a 'connection', and this connection is known as an *Association*. It is formed using a 4-way handshake, which is resistant to the "Denial-of-Service" attacks that have troubled TCP. Like TCP, SCTP is reliable, so that any data that is transferred must be acknowledged. If the data is not acknowledged, it is retransmitted. There are two fundamental differences between TCP and SCTP.

1. SCTP uses *Streams* to transmit data, which are appropriate for message-based applications. This differs from TCP's byte-stream method, which delivers a stream of bytes in the same order as it was presented by the application. SCTP is more sophisticated and the data can be divided up into different streams. Each stream can then be delivered with its own characteristics, and largely independent from other streams. Streams can be defined as 'Strictly-Ordered and Reliable', like TCP, or just 'Reliable', so that data will be delivered to the application as soon as it arrives. Newer versions of SCTP have also introduced a third variation called 'Partially Reliable', which offers a service resembling UDP [4]. The Head-of-Queue Blocking of TCP, which prevents it delivering subsequent data if data is lost, is avoided as each stream operates independently. SCTP can deliver data to the application while waiting for the retransmitted Protocol Data Unit (PDU) to be delivered.

- The second difference relates to the way SCTP interacts with the IP layer. TCP assumes that each host has only one IP address, while SCTP introduces the possibility that many different IP addresses are possible. For any transport protocol, it is important to be able to identify the source of incoming information, and the application it is destined for. TCP uses a 4-tuple in order to do this; a source address and port number pair, and a destination address and port number pair are used to uniquely identify each connection. SCTP allows an association to use a range of available IP addresses, so that it is possible to have $n \times m$ pairs of valid IP addresses, where n and m are the number of available IP addresses at each end-point. The main reason for doing this is to make an Association more resilient to network failures, since the signalling community expects a higher level of reliability than is generally available from the Internet.

This paper is concerned with the multi-homing feature of SCTP, but before this is examined, more information is presented about the SCTP streams.

2.2 Streams in SCTP

TCP delivers data in the same order as the data is presented to it. Therefore if a PDU is not delivered in sequence, or gets lost, TCP will not deliver subsequent PDU until the lost one is successfully delivered. If a link is lossy, this leads to head-of-queue blocking, where the performance of the connection suffers while waiting for the lost PDU. If the application is not concerned about in-order delivery then this additional delay is unnecessary. For instance, if a device is a router, it might be in the process of synchronising stored statistics with a database. During this, if an interface fails, the Network Administrator needs to be informed of this as soon as possible. It is not sufficient that this warning is queued behind the synchronisation data. SCTP overcomes this by assigning it a separate stream, so that the warning is delivered as soon as it arrives, without unnecessary delays caused by sharing the association with the synchronisation.

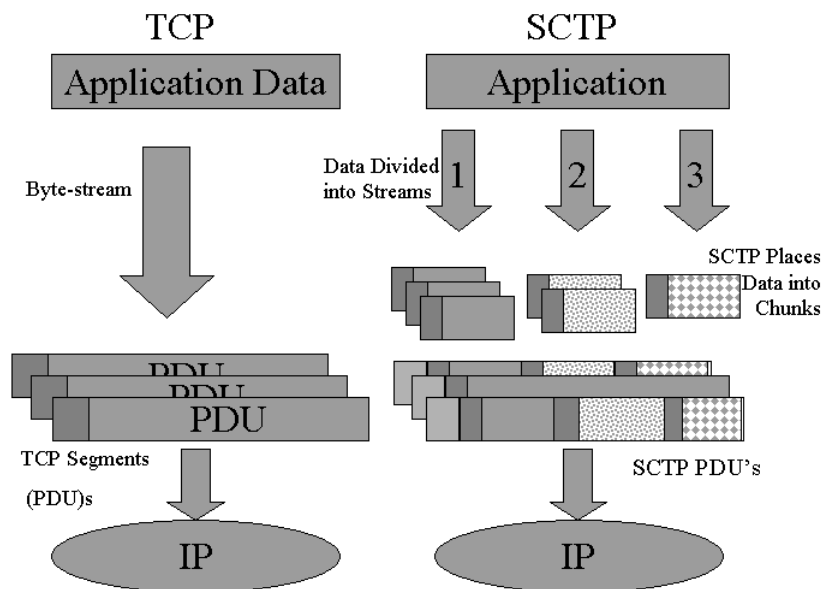


Figure 1 – TCP and SCTP Data Delivery

In order to facilitate streams and some other features of SCTP, the notion of chunks is introduced. As opposed to the TCP byte-stream, information in SCTP is segmented into chunks, which are then carried inside an SCTP PDU. It is the chunks that are acknowledged using a scheme based on the TCP option of *Selective Acknowledgements (SACKs)*. Figure 1 shows how SCTP streams compare to TCP byte-stream delivery. It is interesting to note that the current set of SCTP chunks is not definitive, and already a ‘Partially Reliable’ stream type has been introduced [4].

2.3 Multi-homing in SCTP

As mentioned previously, a significant difference between TCP and SCTP is that SCTP does not limit an association to the same two IP addresses for the duration of the association. This raises many interesting issues, some of which are discussed here. During association initialisation, each end-point of the potential SCTP association advertises any IP addresses that are available to it. This allows the end-points to create a list of addresses. They must then accept any PDU’s with a valid address pair for the duration of the association. In order to ensure validity, port number and verification tag are also included with each PDU.

SCTP does not use the multiple links for load sharing, though there is nothing in the standard that specifically rules this option out. It operates by designating one of the addresses of the corresponding end-points as the *Primary Address* and will attempt to communicate with this address, while all other addresses are *Secondary Addresses*. It does not specify any particular IP address as a source address, but instead allows the operating system to decide. In the case of an error, it will retransmit the chunk to a secondary address. This is different from TCP, as TCP requires a connection to maintain the same addresses for the duration of the connection.

IP routing at an end-point can affect SCTP redundancy. If each end of the association has two interfaces, each with an IP address, then the operating system might decide that the best way to reach either peer address is via the same local interface. If that interface develops a fault, then the entire association may collapse. Despite this, and other potential routing issues, there is still some interest in investigating the use of SCTP in the mobile environment [5].

3 Mobile IP

Mobile IP (M-IP) [3] is a response to the need for mobile users to access typical Internet Services while on the move. Unlike a dial-in type scenario, where a user logs into some local access technology (a phone-line in a hotel) and establishes a tunnel to their corporate head office, M-IP allows a Mobile Node (MN) to change networks in a dynamic fashion. Communication is not lost as a user moves between networks. It provides a method to locate a user, so that Internet-initiated services are possible. For example, if the mobile computer hosts a web page, M-IP can be used to access this web page. This paper deals only with the version of M-IP for IPv4. There is also a version of M-IP for IPv6, which takes advantage of some features that allow it to be optimised. Essentially, M-IP consists of a tunnel between a *home network* and a MN and this tunnel tracks the MN to maintain the connection.

Any traffic directed to the IP address of the MN will be delivered to the home network. At this point, if the MN is not in the home network, a Home Agent (HA) will intercept it. When a MN

enters a *foreign network*, it listens for Agent Advertisements. When it hears an “Advert” for a Foreign Agent (FA), it replies with a “Registration Request”, so that the FA can establish a tunnel to the HA. Traffic intercepted by the HA is then tunnelled to the FA, which then delivers it to the MN. The MN can reply to such traffic by responding to the HA (via the tunnel) and then allowing the HA to route the response onwards.

There are a number of variations to the above scheme. In some cases, when a FA is not available, a MN may obtain an address on the foreign network by some means (e.g. DHCP). It can then send the “Registration Request” directly to the HA, and the tunnel will be directly between the HA and the MN. Another common optimisation is to allow the MN to send packets directly to the destination, leading to *triangular routing*. However, many routers attempt to stop traffic with unusual IP addresses, which means that this method can lead to problems with firewalls.

Despite its inefficiencies, M-IP does solve many of the problems associated with user mobility in the Internet. It allows the user to be located easily, it allows the user to change network without reconfiguring, and it allows applications to continue operation during a network change. Also, by assigning a permanent address to each MN, it allows TCP to work despite changing networks on a regular basis. This is important due to the large number of applications that are dependent on TCP.

Since TCP was not designed to inter-operate with M-IP, some effects of changing networks can interact with TCP and cause poor performance. For instance, TCP assumes that all losses are due to congestion and that the correct response is to reduce the sending rate. If the losses are due to the transition of the MN between two networks (*handover*), then this may lead to performance that is below that which would be expected. There is a lot of research in this area, for example in [6] and [7].

4 Test Network

Our experiments were performed in a using two physically separate locations, in Dublin Universities, **A** and **B**. It consisted of three tunnels across the Internet that connected four networks, the home network in location **B** and the others in location **A**, as is shown in Figure 2. M-IP was used across this network. A Linux workstation in network **B** was designated the HA, while one in network **A** was designated the FA. The MN was placed in a foreign network and registered with the HA. It was connected to the foreign network using an 802.11 Wireless LAN. The third and fourth networks were connected via a Linux router and a Corresponding Node (CN) was connected to both these networks. Depending on the configuration under test at any time, some of the links were not made. For example, the CN is only connected to one network for the first two configurations. The bandwidth was limited to 1.6 Mb/s to reduce the effects of Internet congestion.

The M-IP program used was M-IP HUT, from the Helsinki University of Technology [8]. The SCTP reference implementation from Randall Stewart of Cisco Systems [9] was used during the tests. Ethereal [10] was used throughout the experiments for measurements and troubleshooting. Bandwidth measurements were carried out using the ‘sctp_test_app’, which is part of

the reference implementation. It has a feature named ‘bulk’, which measures the time taken to transmit ‘X’ amount of chunks of size ‘Y’ over a particular stream. The size of these chunks was chosen to optimise performance. A bandwidth measurement was estimated by the amount of time required to transmit 12 MB of data. This was compared to a large file transfer using standard FTP.

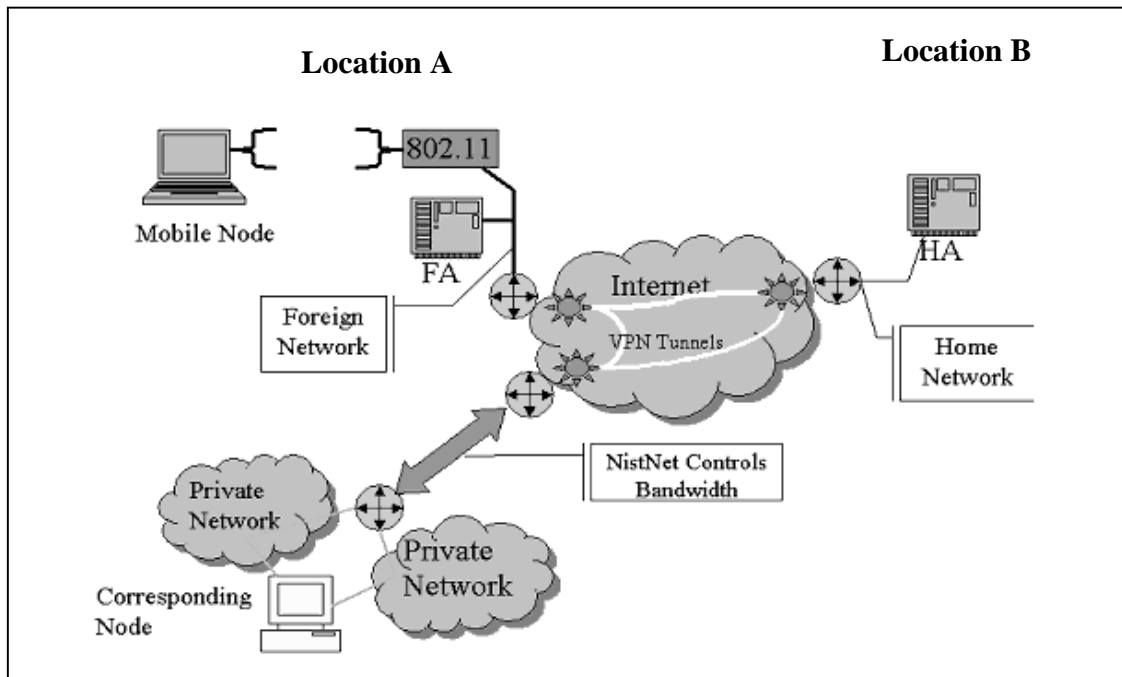


Figure 2 – Test Network

In order to introduce losses and control bandwidth, a program called NISTnet [11] was used. NISTnet allows a single Linux computer set up as a router to emulate a wide variety of network conditions. In some experiments, it added random loss. As a real network was used, real congestion occasionally affected the results. Some of the results presented may reflect this, though all efforts were taken to exclude such effects from our results. However, it should be remembered that this paper is concentrating on how M-IP and SCTP inter-operate and that the performance results are not definitive measures. Three different configurations were implemented and are described with results in the following section.

5 Results

5.1 Configuration 1 - SCTP over MIP

Configuration 1 used a single-homed CN communicating with a MN, with only the M-IP address being used. The SCTP PDU’s were successfully transferred and the results are shown in Table 1. It can be seen that SCTP provides a performance enhancement over TCP and as SCTP uses selective acknowledgements, this was to be expected [12] and [13]. This improved performance continues as errors are added to the link as is seen in Table 1. In this SCTP configuration, there is no use of multi-homing to add redundancy. Indeed it could be argued that M-IP is not very resilient to errors, as there are a number of different single point of

failures in the network, including both the FA and HA. Although SCTP performs well over this M-IP configuration, the multi-homing feature of SCTP has been suppressed.

| Errors | Throughput in Mb/s | | % of 1.6 Mb/s | | % Change SCTP to TCP |
|--------|--------------------|-------|---------------|--------|-------------------------|
| | TCP | SCTP | TCP | SCTP | |
| 0 % | 1.32 | 1.525 | 82.5% | 95.3% | 15.5 |
| 1 % | 1.08 | 1.225 | 67.5% | 76.6 % | 13.4 |
| 2 % | 0.72 | 0.855 | 45.1% | 53.4% | 18.8 |
| 5% | 0.35 | 0.481 | 21.9% | 30.0% | 37.4 |

Table 1: Performance of SCTP and TCP: Configuration 1

5.2 Configuration 2 – SCTP using MIP as a Locator

Unlike TCP, SCTP allows the use of multiple addresses. This means that it may be possible to use the M-IP address and an address belonging to the foreign network (referred to as a local address from now on). In such a scenario, it is envisaged that the M-IP address can be used to locate a mobile terminal. It then becomes possible to use the local address for the rest of the transmission. Since traffic to the local address does not have to pass through the HA the distance travelled is likely to be shorter, so that the overall network load can be expected to decrease in almost all cases. In many cases, the overall end-to-end performance may also improve dramatically.

In this configuration, the routing table had to be carefully provisioned. The Wireless LAN interface of the MN had 2 addresses, the M-IP address and the local address. The MN was forced to use the M-IP address as its source address and the CN selected the M-IP address as the primary address. The effect of this configuration is that the CN has two possible destinations, while the MN has only one. A similar configuration would allow the use of more than one interface for secondary addresses at the MN. This would not affect the presented results as path properties are artificially controlled in this experiment. The secondary link from the CN to the MN’s local address is only used for retransmissions. It should, therefore, carry considerably less traffic than the primary route and in the case of a lossless link the transfer time results match those found in Configuration 1. Most wireless systems guarantee the delivery of layer 2 PDUs, so that loss is primarily caused by congestion at the radio link. That is, if the CN pushes data towards the MN at a rate that is greater than that which the radio link can accommodate, then the radio link buffer will overflow. To emulate such a link on the secondary route, it is assumed that no such losses will occur and this path has therefore had no additional losses added by NISTnet.

| Errors | Throughput in Mb/s | | % of 1.6 Mb/s | | % Change 2 Links to 1 |
|--------|--------------------|---------|---------------|---------|--------------------------|
| | 1 Link | 2 Links | 1 Link | 2 Links | |
| 0 % | 1.525 | 1.525 | 95.3% | 95.3% | 0 |
| 1 % | 1.225 | 1.375 | 76.6 % | 85.6% | 12.2 |
| 5% | 0.481 | 0.68 | 30.0% | 42.5% | 41.4 |

Table 2: Performance of 1 and 2 link SCTP links.

This configuration shows a significant improvement over a single link configuration as is seen in Table 2. However, there is little added resilience since the MN has a single point of failure. Also, SCTP does not select the “best” primary address, this is simply set by the application. The results presented only reflect transmission from the CN to the MN as this is expected to be the primary direction of data flow.

5.3 Configuration 3 – SCTP using two Distinct Paths

In this configuration, the CN is connected to a second network. A slight variation of this configuration is that two addresses are placed on the same network and interface. Again, the MN has a M-IP address and a local address so that there are four possible combinations of valid addresses. However, the only way to force selection of a particular source address is to tie it to a destination address in the routing table. For this reason the routing tables were constructed so that only two combinations could be chosen [5] and [13].

Again, the primary link is selected by the application and the performance is similar to that of Configuration 2. In this case, however, the effect is bi-directional, as the retransmission benefit is now also available to traffic flowing from the MN to the CN. It would also appear to be possible to use this configuration to swap from a poor primary link to the exclusive use of a better secondary link. This would enable a layer four handover for use in mobile systems [5].

Testing for this handover capability revealed that a layer-4 handover was unlikely with the existing SCTP implementations. If we consider a file download to the MN, then a failure of the primary link will cause the CN to retransmit data on the secondary link. The MN, however, will detect duplicate PDUs and will assume that there is a fault on the link from which the duplicate PDU was received. It appears that it then sends its acknowledgements to the CN via the other link, that is, the primary link. This sequence of events will usually result in the collapse of the association. This is an implementation issue rather than something implicit in the standard.

6 Conclusions

The experiments presented here demonstrate some possible topologies using current technology and protocols. They show a progression from a straightforward replacement of TCP with SCTP to some more complex uses of M-IP in a foreign network. The results presented above show a moderate benefit from using SCTP and show that in most circumstances it will operate well with M-IP. The current implementations of SCTP do not appear to be able to support layer-4 handover, but the standard does appear to be able to provide this functionality. In fact, during handover, the current implementations could cause the association to fail due to the policy of sending acknowledgements. It seems likely that some minor modifications will be required to enable SCTP to fully function with M-IP and manage layer-4 handover, but care must also be taken to ensure that these changes do not cause unwanted side effects.

Acknowledgements

This research was supported by Enterprise Ireland's Applied Research Scheme in conjunction with Ericsson Ireland, to whom we are grateful. The authors further wish to acknowledge Cyril Roger, Thomas Ravier, Fouad Chmainy and Rob Brennan for their valuable assistance in the experiments and expertise described in this paper.

References

- [1] R. Stewart, "Stream Control Transport Protocol" RFC 2960, October 2001
- [2] A. Jungmaier, "SCTP for Beginners" http://tdrwww.exp-math.uniessen.de/pages/forschung/sctp_fb/
- [3] C. Perkins, "IP Mobility for IP v4, revised" RFC 3220, January 2002
- [4] R. Stewart, "SCTP Partial Reliability Extension" Internet-Draft, draft-stewart-tsvwg-prsctp-01.txt, July 2002
- [5] M Riguel, M Tuexan, "Mobile SCTP", draft-riegel-tuexen-mobile-sctp-01.txt
- [6] H. Balakrishnan et al "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links" Proc. ACM SIGCOMM '96
- [7] K Brown, S. Singh, "M-TCP: TCP for Mobile Cellular Networks", ACM Computer Communication Review, 1997
- [8] "Dynamics HUT Mobile-IP" (Software) <http://www.cs.hut.fi/Research/Dynamics/news/2000-04-17.html>
- [9] Author R. Stewart , "Reference Implementation of SCTP", (Software) www.sctp.org
- [10] "Ethereal", www.ethereal.com (Software)
- [11] "NISTnet" (Software) <http://snad.ncsl.nist.gov/itg/nistnet/>
- [12] K. Fall, S. Floyd "Simulation-based Comparisons of Tahoe, Reno, and SACK TCP", Computer Communication Review, vol. 26, pp. 5--21, July 1996
- [13] L. Coene , "Multihoming issues in the Stream Control Transmission Protocol" Internet-Draft, draft-coene-sctp-multihoming-04.txt Author
- [14] Rob Brennan, T. Curran, "SCTP Congestion Control: Initial Simulation Studies" Proceedings of the International Teletraffic Congress,