Broadband Satellite Multimedia (BSM) Interworking with Performance Enhancing Proxies

H. Cruickshank¹, R. Mort², M. Berioli³

1 CCSR, University of Surrey, Guildford, UK

2 SysTek, Havant, UK

3 German aerospace center (DLR), Munich, Germany

Keywords: PEP, SI-SAP, TCP, BSM.

Abstract

Satellites had been successful in the past due to their wide area coverage and speedy deployment of new services especially in remote regions of Europe and the rest of the world. The future development of broadband satellite systems providing services based on the Internet Protocol (IP) needs to be stimulated by means of common standards. This paper presents the ETSI BSM PEP architecture which includes the satellite terminal and gateway protocol stacks and security configurations for successful PEP implementations

1 Introduction

The development of broadband satellite systems providing services based on the Internet Protocol (IP) needs to be stimulated by means of common standards. These standards will allow building blocks and services for such satellite systems to become more readily available. The ETSI Broadband Satellite Multimedia (BSM) working group [1] ensures that this work can be carried out in a timescale that will allow development of universal access to everyone, everywhere via a combination of private and public internet access points.

The BSM work is focussed on the efficient transport of IP data streams and on how to interoperate resulting satellite networks with terrestrial IP networks. The BSM standards are being designed to use existing standards (such as DVB-RCS [2]) while remaining open to emerging standards and other available technologies (the ultimate choice is left to the market). This paper presents the current work in defining the PEP architecture for BSM satellite networks.

In general, the Internet transport protocol (namely TCP) exhibits suboptimal performance due to the following satellite characteristics:

- Long feedback loops: Propagation delay from a sender to a receiver in a geosynchronous satellite network can range from 240 to 280 milliseconds. This will cause slow

connection setup, slow to respond to loss and slow discovery of available bandwidth.

- Large bandwidth*delay products: TCP needs to keep a large number of packets "in flight" in order to fully utilize the satellite link.
- Asymmetric capacity: The return link capacity for carrying ACKs can have a significant impact on TCP performance.

End-to-end improvements to TCP (without using PEPs) can be by using Maximum segment size (MSS), maximum transmission window, Selective Acknowledgements (SACKs) and using Timestamps Option. HTTP end-to-end improvement can be using compression plus HTTP 1.1 with Persistent connections Persistent connections and pipelining [3] and [4]. Such mechanisms are advantageous because they are based on standard options and maintain end-to-end semantics.

However, end-to-end techniques have the following drawbacks:

- Certain parameters cannot be optimized at the same time for different access technologies. For example, the Bandwidth Delay Product (BDP) in satellite networks is much larger than terrestrial wireless network. Moreover, servers are by default unaware of the access technology used by a client. Additional intelligence could be provided to a server for selecting adequate protocol settings in each case, but this approach is not used for practical reasons.
- At least one TCP Slow Start phase will still take place during a web page download.
- Should multiple objects be hosted under different domain names, DNS lookup overhead cannot be avoided or reduced using end-to-end options.
- End-to-end mechanisms cannot cope with connectivity gaps suffered when link outages occur. The reason for this is that a server is unable to distinguish between congestion and radio link losses. This gives rise to the unnecessary activation of TCP congestion control mechanisms.

Considering the issues discussed above, it can be concluded that end-to-end optimization mechanisms provides some improvements, but cannot provide optimal performance. An alternative solution is by splitting the end-to-end connection

by using Performance Enhancing Proxies (PEPs). The PEP is an entity placed somewhere between the endpoints of a communication link. We focus on TCP PEPs (T-PEP) and Application PEPs (A-PEP). This approach can further be divided into two categories: Distributed PEPs where the PEP client and server are located at each end of the satellite link. The other category is Integrated PEPs with only one PEP entity residing with the satellite gateway. Typical TCP PEP improvements are:

- TCP Spoofing: Eliminates effects of satellite delay on TCPs slow start and window sizing.
- ACK Reduction: Reduces unnecessary acknowledgements to improve bandwidth efficiency.
- Flow Control: Employs network feedback to intelligently control traffic flow.
- Error Recovery: Works closely with Flow Control to recover damaged or lost packets.
- Traffic Prioritization: Classifies traffic by IP address and port and prioritizes accordingly.
- Connection Establishment Spoofing: Intelligently spoofs the TCP three-way handshake to speed up establishment of a connection.

In addition to TCP PEPs (T-PEP), there are other complementary solutions such as application layer PEPs (A-PEP), where web browsing is the major target for application PEPs. Typical application layer PEPs improvements are:

- HTTP pre-fetching: Intercepting requested Web pages, identifying Web objects referred to by the Web pages, downloading these objects in anticipation of the next user requests.
- Browser Cache Leveraging: Cache's some web pages not residing in browser cache, improving efficiency.
- Bulk Transfer Prioritization: Prioritizes bulk transfers to prevent adverse effect on other Web traffic.
- Cookie Handling: Ensures accurate painting of Web pages with the proper cookies.
- Compression: Payload compression provides increased transmission speeds. In addition, header compression for TCP, UDP, and RTP protocols results in additional bandwidth savings.
- DNS caching techniques, to further improve bandwidth utilization.

Commercial PEPs normally combine some or all of the T-PEP and A-PEP techniques together such the Hughes (see [5]), XipLink ([6]) and , FastSat ([7]).

This paper is organized as follows: Section 2 provides an overview of current ETSI BSM architecture. Section 3 presents the Satellite Terminal (ST) and Gateway (GW) PEP architecture and overview of a usage scenario. Section 4 describes the security implications of using PEPs. Finally section 5 concludes this work.

2 Overview of BSM architecture

The BSM architecture, [1], [8] and [9] is presented in Figure 1 with the general BSM protocol stack for IP services in the

Satellite Terminals (ST) and the Gateways (GW). An important feature is the Satellite Independent Service Access Point interface or SI-SAP interface. This interface provides the BSM with a layer of abstraction for the lower layer functions. It allows the BSM protocols developed in the satellite independent layer to perform over any BSM family (specific satellite technologies). Moreover, the SI-SAP also enables the use of standard Internet protocols for example address resolution, QoS, security and network management, directly over the BSM or with minimal adaptation to BSM physical characteristics. Finally the SI-SAP even makes it possible to envisage switching from one satellite system to another and to even a non-satellite technology while preserving the BSM operator's investment in layer 3 software development.

In addition, Figure 1 shows that there are only a small number of generic functions that need to cross the SI-SAP and those are related to connection/session management, resource management or security. The BSM protocols are based on the OSI layered protocol stack. For the IP services most of the work has concentrated on the network layers with links to the underlying data link and Media Access Control (MAC) layers. The reason for this is simple: the developed protocols for IP over BSM should primarily be located in the satellite independent part of the BSM stack to be applicable to a range of different satellite dependent lower layers such as DVB-RCS [1].

Although not shown in Figure 1, this BSM architecture is directly applicable to PEP designs. If the PEP design adopts a satellite-independent approach it can be used with different lower layers without requirement significant redevelopment. This has benefits for both the PEP manufacturers (by reducing the new costs and time of new developments) and also for the end-user who can migrate to a new satellite system while retaining the same or similar "known" PEP properties.

Figure 1: BSM Protocol Stack

Figure 1 also show that the SI-SAP is further divided into user (SI-U-SAP), control (SI-C-SAP) and management (SI-M-SAP) interfaces. These are used to for user data, connection control and network management respectively. The PEP data will use the SI-U-SAP, while related QoS signalling and management functions will use the SI-C-SAP and SI-M-SAP interfaces respectively.

3 PEP terminal architecture and components

Figure 2 and Figure 3 show the combined PEP protocol stack with the BSM ST and Gateway terminal architectures respectively. The PEP residing on the BSM ST side is called ST PEP (PEP client) and the one on the BSM gateway side is called and Gateway PEP (PEP server). Both PEPs have a similar architecture with two interfaces, one to the BSM satellite network and one to terrestrial networks. On the satellite side, the ST/Gateway PEP are connected to BSM ST/Gateway through an Ethernet LAN. On the terrestrial network side, normally, the PEP terminal connects to hosts on the same LAN, while the gateway PEP connects to a content server through the general Internet. However, the Gateway PEP can be located remotely from the BSM Gateway terminal (such as Gateway PEP run by a service provider).

The transport protocol in the PEP is divided between standard TCP/UDP and PEP specific transport protocols. As shown in Figure 2 and Figure 3, the PEP specific transport protocol can be:

- A modified TCP (TCP+) such as the Hybla protocols [10], which is used in integrated PEP configurations, where only Gateway PEP will be used (no ST PEP).
- Standard I-PEP Transport Protocol (I-PEP TP), recommended by Satlabs [4] and used in the distributed PEP configurations. The I-PEP TP is based on an extension set to TCP termed SCPS-TP, which was produced by the Consultative Committee for Space Data Systems (CCSDS).
- Proprietary distributed Transport Protocol (TP+), where other company specific (non-standard) protocols are used.

Figure 2 BSM ST PEP

Figure 3 BSM Gateway PEP

The ST/Gateway PEPs can be managed either locally or remotely. For remote management, either SNMP or HTTP protocols can be used to communicate with the BSM management system. In both cases the PEP monitoring and configuration controls can be based on the standard MIB II and enterprise specific PEP MIBs.

Also both figures show the QoS signalling between the PEP and the BSM QoS managers in the ST and Gateway. Such signalling is necessary for QoS monitoring of the ST/Gateway queues and adjusting rate control parameters accordingly to maximize the use of the satellite capacity. The optimum PEP performance is expected to require a close matching between the PEP configuration and the QoS of the associated lower layer bearer services. This signalling can be based on IntServ or DiffServ architectures [11] and [12].

Figure 4 shows a typical distributed PEP scenario with a single user behind the ST PEP (PEP client). This reflects the typical home user or home office configuration. The PEP client may be integrated with the BSM ST, or it may be a stand-alone entity separate from both the end user's device and the ST. There can be several variations to this scenario. One variation is a multi-user scenario where the same ST PEP serves multiple users in a LAN configuration. Another variation to Figure 4 is where the Gateway PEP (PEP server) is external to the BSM Gateway (satellite terminal), motivating two different set-ups:

- PEP server may be run by a separate Internet Service Provider (ISP) on behalf of many users or
- PEP server may be operated by an enterprise on its own behalf.

Figure 4 Typical BSM PEP scenario

A third variation is the use of multiple Gateway PEPs. The motivation here can be the presence of multiple ISPs or because performance enhancement is managed directly

between user sites (VPN configuration). Here the ST PEP needs to interoperate with multiple Gateway PEPs from different vendors. This is an ideal setup for using the I-PEP protocol [11] mentioned earlier.

In the Integrated PEP scenario, there will be only one PEP entity at the BSM Gateway. In this case I-PEP type protocol is not used.

4 Security impact on Performance Enhancing Proxies (PEPs)

Interworking between PEPs and security system has been researched in the past [13]. For example, many researchers had addressed the issue of interworking between IPsec and PEPs. One solution was the use of an intelligent switch at the PEP. As such, the PEP provides acceleration for the unencrypted packets, while the encrypted packets are allowed to bypass the PEP. With this approach, the applications can choose between security and performance, but both are not obtainable together.

Transport Friendly ESP (TF-ESP) or Modified ESP (M-ESP) [13] proposes a modification to ESP header to accommodate the necessary TCP header information in the ESP header outside the scope of encryption. The mechanism proposes that the unencrypted TCP header information in ESP should be authenticated for integrity. Although this method addresses the performance issues, it exposes enough information to make the connection vulnerable to security threats [14].

The Multilayer IPSEC Protocol (ML-IPSEC) proposes a multi-layer encryption scheme. The IP datagram payload is divided into zones; each zone has its own security associations and protection mechanisms. For instance, the TCP data part can be a zone, using end-to-end encryption with the keys only shared between end-hosts. The TCP header could be another zone with security associations between the source, destination and a few trusted nodes (such as PEPs). The trusted nodes can decrypt the transport layer headers to provide the performance enhancements. This mechanism ensures security and can accommodate existing performance solutions. Though the requirements are satisfied, the complexity involved is tremendous. Also, the assumption that intermediate nodes are trustworthy may not be acceptable for users preferring end-to-end security.

Some other solutions explore the use of transport layer security. Secure Socket Layer (SSL) as proposed by Netscape and later been standardized by IETF as Transport Layer Security (TLS) [15] is a transport layer mechanism that provides data security. It encrypts the user data, but not the transport layer headers, such as TCP headers. Since the transport layer headers are in plaintext, the intermediate nodes (such as PEPs) can access or modify them; thereby the performance related issues can be resolved. However, it is not recommended to have TCP headers in plaintext due to security concerns [14]. Suggestions were also made to use SSL/TLS with IPsec in order to protect the header

information. The use of SSL/ TLS with IPsec is not a good solution because PEP cannot function as IPSEC encrypts the TCP headers.

In summary, there is a requirement that security must be implemented in such way that allows ST and Gateway PEPs to access the transport protocol headers for T-PEPs and HTTP content for A-PEPs.

Transport/application layer security will work seamlessly with T-PEPs because the TCP header is not encrypted by the security system (see in Figure 5). However, transport/application layer security will not function with A-PEPs. The reason is that application layer data will be encrypted by the security system. Hence, it will not be possible to perform techniques such as HTTP prefetching, caching and header and payload compressions.

End-to-end network layer security (such as IPsec) will encrypt the TCP header and user data therefore both T-PEPs and A-PEPs will not work. As such, T-PEPs will not be able to perform techniques such as TCP spoofing, ACK reduction and flow control. In addition, A-PEPs will not be able to perform HTTP prefetching, caching and compression. Thus a user or network administrator must choose between using PEPs or using end-to-end IPsec.

As shown in Figure 5, PEPs can be used successfully with IPsec in tunnel mode between the BSM ST/Gateway. Here the encryption is performed on incoming traffic after the PEP operations and decryption is performed on outgoing traffic before the PEP operations. Here the IPsec operations are under the control of the satellite network.

In addition and in terms of overheads, IPsec tunnel mode requires an extra IP header, where basic IPv4 header is 20 bytes and IPv6 header is 40 bytes. Also IP multicasting over satellites can exploit the broadcast nature of satellites. However, secure multicasting with IPsec (in tunnel mode) has two more added implications: First, IP multicast becomes effectively point-to-point connections between the IPsec tunnel ends; second manual keying only is used. Therefore, the recently published RFC 5374 (multicast extension to IPsec) provides an optional extension to IPsec to resolve these issues. However, the multicast extensions to IPsec might not be available on all BSM ST, Gateway or router equipment.

Also Figure 5 shows the link layer security mechanism can be used such as DVB-RCS [1] security or Unidirectional Link Encapsulation (ULE) security [16]. Here T-PEPs and A-PEPs will work seamlessly over the secure satellite link. The reason is TCP header and user data are handled in clear text (no encryption) in the Gateway PEP. Then, the satellite link layer security is only applied between the BSM ST and GW (satellite terminals). Finally, the TCP header and user data are handled in clear text (no encryption) in the ST PEP.

Although link layer security does not provide the desired end-to-end security, it is more efficient than using IPsec (in tunnel

mode). It also can provide extra security functions that are not possible IPsec or upper layer security such user identity hiding (such as IP and MAC addresses). This allows providing strong privacy service over the satellite broadcasting link.

Figure 5 Security solutions with PEPs

5 Conclusions

The ETSI BSM standardisation work is focussed on the efficient transport of IP data streams and on how to interoperate resulting satellite networks with terrestrial IP networks. The paper presented the current work in ETSI BSM group in defining the PEP architecture for BSM networks. The ST and Gateway PEP protocol stack has been shown.

In addition, the paper analyses the security implications for using PEPs. The paper presents the three security solutions: application/transport, IPsec and link layer security. The analysis show that link layer security well work seamlessly with T-PEPs and A-PEPs.

Acknowledgements

This work was sponsored by the European Telecommunications Standards Institute (ETSI) [17] and supported by the EU Information Society Technologies SATNEX project [18].

References

- [1] ETSI TS 102 292, "Satellite Equipment and Systems (SES); Broadband Satellite Multimedia; IP Interworking over Satellite; BSM Functional Architecture".
- [2] ETSI. Digital Video Broadcasting (DVB); DVB specification for data broadcasting. ETSI EN 301 790 V1.4.1 (2005-04). Interaction channel for satellite distribution systems", 2005-04.
- [3] IETF RFC 3135: "Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations".
- [4] C. Gomez, et al "Web browsing optimization over 2.5G and 3G: end-to-end mechanisms vs. usage of performance enhancing proxies". WIRELESS COMMUNICATIONS AND MOBILE COMPUTING. 2008; 8:213–230. Wiley InterScience.

- [5] http://www.direcway.com/HUGHES/Doc/0/SIKPBJS69O6KP42VCE4K4ER2BF/Hughes%20PEP_H35661_A4_LR_091206.pdf IETF document: www.ietf.org
- [6] http://www.xiplink.com/IMG/pdf/XipLink_Internet_over_Satellite_Optimization-R2.pdf
- [7] <http://www.spacebel.be/FR/Space/FastSatDataSheet.pdf>
- [8] ETSI TR 101 984: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; Services and Architectures".
- [9] ETSI TR 101 985: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia; IP over Satellite".
- [10] Carlo Caini, Rosario Firrincieli, Daniele Lacamera, "PEPs: A Performance Enhancing Proxy for TCP Satellite Connections". IEEE A&E SYSTEMS MAGAZINE, AUGUST 2007
- [11] ETSI TS 102 463: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Interworking with IntServ QoS".
- [12] ETSI TS 102 464: "Satellite Earth Stations and Systems (SES); Broadband Satellite Multimedia (BSM); Interworking with DiffServ QoS". I-PEP specifications, Issue 1a. Satlabs group recommendations (<http://www.satlabs.org>). October 2005.
- [13] V. Obanaik Secure performance enhancing proxy: To ensure end-to-end security and enhance TCP performance over IPv6 wireless networks. Elsevier Computer Networks 50 (2006) 2225–2238. 2006
- [14] S. Bellovin, Probable plaintext cryptanalysis of the IPSecurity protocols, Proceedings of the Symposium on Network and Distributed System Security, February 1997
- [15] T. Dierks, et al, The TLS Protocol Version 1.2, RFC 5246, Aug. 2008
- [16] H. Cruickshank, P. Pillai and M. Noisternig, "Security requirements for the Unidirectional Lightweight Encapsulation (ULE) protocol", Internet Draft (draft-ipdvb-sec-req-09.txt), August 2008.
- [17] ETSI home page: http://portal.etsi.org/Portal_Common/home.asp
- [18] Satnexus project home page: <http://www.satnexus.de>