TOR VERGATA SIMULATION/EMULATION FACILITIES

M. Luglio, C. Roseti, F. Zampognaro

0.1 Reference Documents ........................................................................................................................................... 3
0.2 Glossary of Terms, Acronyms and Abbreviations ............................................................................................... 3

1. Introduction ............................................................................................................................................................. 5
   1.1 Testbed General Description ................................................................................................................................. 6
       1.1.1 Simulation Platform ........................................................................................................................................ 6
       1.1.2 Emulation Platform ......................................................................................................................................... 7

2 TCP over DVB-RCS .................................................................................................................................................... 9
   2.1 Overview ............................................................................................................................................................... 9
   2.2 Testbed Description ............................................................................................................................................. 10
   2.3 Trials and Demonstrations .................................................................................................................................... 10
   2.4 Results .................................................................................................................................................................. 10

3 TOP project ............................................................................................................................................................. 11
   3.1 Overview ............................................................................................................................................................... 11
   3.2 Testbed Description ............................................................................................................................................. 12
   3.3 Trials and Demonstrations .................................................................................................................................... 13
   3.4 Results .................................................................................................................................................................. 13

4 Satellite-WiFi Handover Simulation ..................................................................................................................... 14
   4.1 Overview ............................................................................................................................................................... 14
   4.2 Testbed Description ............................................................................................................................................. 15
   4.3 Trials and Demonstrations .................................................................................................................................... 15
   4.4 Results .................................................................................................................................................................. 16
       4.4.1 Reordering and generation of segment bursts ........................................................................................... 16
       4.4.2 Bandwidth-Delay product variation .......................................................................................................... 16
       4.4.3 Time-out expiration ....................................................................................................................................... 16

5 SatNEx “Cross-Layer IPsec” activity ....................................................................................................................... 17
   5.1 Overview ............................................................................................................................................................... 17
   5.2 Testbed Description ............................................................................................................................................. 20
   5.3 Trials and Demonstrations .................................................................................................................................... 21
   5.4 Results .................................................................................................................................................................. 21

6 EMERSAT Project ....................................................................................................................................................... 23
   6.1 Overview ............................................................................................................................................................... 23
   6.2 Testbed Description ............................................................................................................................................. 24
   6.3 Trials and Demonstrations .................................................................................................................................... 25
   6.4 Results .................................................................................................................................................................. 25

7 INTERSECTION Project .............................................................................................................................................. 26
   7.1 Overview ............................................................................................................................................................... 26
   7.2 Testbed Description ............................................................................................................................................. 26
   7.3 Trials and Demonstrations .................................................................................................................................... 27
   7.4 Results .................................................................................................................................................................. 28

8 SENSIBLE Project ...................................................................................................................................................... 29
   8.1 Overview ............................................................................................................................................................... 29
   8.2 Testbed Description ............................................................................................................................................. 29
   8.3 Trials and Demonstrations .................................................................................................................................... 30
   8.4 Results .................................................................................................................................................................. 30
0.1 Reference Documents

[RD-1]  Ns-2 web site, URL: isi.edu/nsnam/ns
[RD-7]  ETSI, Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, new gathering and other broadband satellite applications, EN 302 307 v1.1.2, 2006-06.

DVB
DVB RCS
IETF RFC
Standard TCP
IP Sec
.....
.....

23.1 Glossary of Terms, Acronyms and Abbreviations

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>DVB RCS</td>
<td>Digital Video Broadcasting Return Channel on Satellite</td>
</tr>
<tr>
<td>DVB</td>
<td>Digital Video Broadcasting</td>
</tr>
<tr>
<td>DAMA</td>
<td>Demand Assignment Multiple Access</td>
</tr>
<tr>
<td>PEP</td>
<td>Performance Enhancing Proxy</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>TCP</td>
<td>Transmission Control Protocol</td>
</tr>
<tr>
<td>TDMA</td>
<td>Time Division Multiple Access</td>
</tr>
<tr>
<td>RTT</td>
<td>Round Trip Time</td>
</tr>
<tr>
<td>UDP</td>
<td>User Datagram Protocol</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>STK</td>
<td>Satellite Tool Kit</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>CAC</td>
<td>Call Admission Control</td>
</tr>
<tr>
<td>ASI</td>
<td>Agenzia Spaziale Italiana</td>
</tr>
<tr>
<td>NCC</td>
<td>Network Control Center</td>
</tr>
<tr>
<td>ESA</td>
<td>European Space Agency</td>
</tr>
<tr>
<td>EEG</td>
<td>Electro-Encephalo-Graphy</td>
</tr>
<tr>
<td>MAC</td>
<td>Medium Access Control</td>
</tr>
<tr>
<td>NS2</td>
<td>Network Simulator 2</td>
</tr>
<tr>
<td>TCL</td>
<td>Tool Command Language</td>
</tr>
<tr>
<td>CWND</td>
<td>Congestion Window</td>
</tr>
<tr>
<td>SSTHRESH</td>
<td>Slow Start Threshold</td>
</tr>
<tr>
<td>ST</td>
<td>Satellite Terminal</td>
</tr>
<tr>
<td>UT</td>
<td>User Terminal</td>
</tr>
<tr>
<td>PSTN</td>
<td>Public Switched Telephone Network</td>
</tr>
<tr>
<td>CRA</td>
<td>Constant Rate Allocation</td>
</tr>
<tr>
<td>RBDC</td>
<td>Rate Based Dynamic Capacity</td>
</tr>
<tr>
<td>VBDC</td>
<td>Volume Based Dynamic Capacity</td>
</tr>
<tr>
<td>FIFO</td>
<td>First In First Out</td>
</tr>
<tr>
<td>AVBDC</td>
<td>Absolute Volume Based Dynamic Capacity</td>
</tr>
<tr>
<td>MF TDMA</td>
<td>Multi Frequency Time Division Multiple Access</td>
</tr>
<tr>
<td>ATM</td>
<td>Asynchronous Transfer Mode</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hyper Text Transfer Protocol</td>
</tr>
<tr>
<td>TCPN</td>
<td>TCP-Noordwijk</td>
</tr>
<tr>
<td>SNACK</td>
<td>Selective Negative Acknowledgement</td>
</tr>
<tr>
<td>I-PEP</td>
<td>Interoperable Performance Enhancing Proxy</td>
</tr>
<tr>
<td>BER</td>
<td>Bit Error Rate</td>
</tr>
<tr>
<td>LAN</td>
<td>Local Area Network</td>
</tr>
<tr>
<td>CoA</td>
<td>Care-of-Address</td>
</tr>
<tr>
<td>MN</td>
<td>Mobile Node</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
</tr>
<tr>
<td>CN</td>
<td>Correspondent Node</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point</td>
</tr>
<tr>
<td>MIP</td>
<td>Mobile IP</td>
</tr>
<tr>
<td>RTO</td>
<td>Retransmission Time Out</td>
</tr>
<tr>
<td>BDP</td>
<td>Bandwidth Delay Product</td>
</tr>
<tr>
<td>HO</td>
<td>Handover</td>
</tr>
<tr>
<td>MPEG</td>
<td>Moving Picture Expert Group</td>
</tr>
<tr>
<td>RMT</td>
<td>Reliable Multicast Transport</td>
</tr>
<tr>
<td>FLUTE</td>
<td>File Delivery Over Unidirectional Transport</td>
</tr>
<tr>
<td>ALC</td>
<td>Asynchronous Layered Coding</td>
</tr>
<tr>
<td>FEC</td>
<td>Forward Error Correction</td>
</tr>
<tr>
<td>IETF</td>
<td>Internet Engineering Task Force</td>
</tr>
<tr>
<td>ESP</td>
<td>Encapsulating Security Protocol</td>
</tr>
<tr>
<td>ISAKMP</td>
<td>Internet Security Association Establishment and Key Management Protocol</td>
</tr>
<tr>
<td>AH</td>
<td>Authentication Header</td>
</tr>
<tr>
<td>CRC</td>
<td>Cyclic Redundancy Code</td>
</tr>
<tr>
<td>SA</td>
<td>Security Association</td>
</tr>
<tr>
<td>ICV</td>
<td>Integrity Check Value</td>
</tr>
<tr>
<td>VLC</td>
<td>VideoLAN Client</td>
</tr>
<tr>
<td>PLR</td>
<td>Packet Loss Rate</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

Satellite systems well cope with a large set of service requirements as concerns broadband and untethered connectivity. This is the case in particular of information dissemination and gathering, emergency support and disaster response, long range mobility, digital divide. Moreover, satellite systems can be integrated to terrestrial wireless systems (e.g. WiFi, WiMax, etc.) to enhance capillarity and spread network access to a large set of users.

DVB-RCS standard, which can be considered as widespread European standard for broadband satellite connectivity, defines DAMA Bandwidth on Demand schemes to allow an efficient access to satellite resources in the return link, but introduces further contributions and variability to the overall delay due to processing and negotiation of the bandwidth requests. With a specific focus on transport layer, performance of the main protocols, TCP and UDP, is particularly affected in a DVB-RCS communication scenario and requires specific enhancements. TCP/IP applications must face some challenges due to some intrinsic characteristics (large latency, possible harsh propagation environment – link disruptions/packet losses – high link asymmetry), typical of satellite systems. To this extent, proposed solutions are based on architectural changes, which basically introduce intermediate agents (i.e. PEP) with the aim of converting/adapting standard protocols and the use of optimized protocols and procedures. In this framework, some security issues arise especially when satellite segment is interconnected with other terrestrial networks. Therefore, performance analysis must be always associated to possible impacts on security aspects.

The analysis and the design of satellite systems, either stand-alone or integrated with other terrestrial systems, has been supported by simulation and emulation activities, associated to specific projects and activities. In this document eight study cases have been addressed:

1. R&D internal project on “TCP behavior over DVB-RCS”, involving both simulations and emulations to outline the TCP dynamics in the DVB-RCS environment with the inclusion of bandwidth on demand and TDMA based transmission;
2. TOP Project (ESA/Artes1) aimed to design and develop a specific TCP protocol for DVB-RCS links, tested on linux PC with a reference C++ implementation; when TCP operates over a DAMA-based satellite link, two nested control loops govern transfer process (TCP flow control loop and the DAMA bandwidth allocation control loop); the exploitation of each loop requires time duration of the same order, which is proportional to the propagation Round-Trip Time (RTT) and, depending on circumstances, can cooperate to give good performance or can interact in a way that degrades performance; a completely new TCP, called TCP-Noordwijk, has been conceived, designed and developed to counteract these issues;
3. The R&D project “Satellite-WiFi Handover Simulation” undertaken in-house by the Univ. of Rome, Tor Vergata aims at the development of an NS-2 based simulator to simulate the handovers between DVB-RCS satellite and WiFi networks; when a vertical handover is performed through the MIP protocol, there is a time period, known as “Handover Latency”, in which all the packets/acknowledgements get lost; this leads to a TCP timeout expiration with a consequent performance degradation; in addition, handover between links with different delay and bandwidth-delay product can cause a number of further problems such as the generation of segment bursts and deliver of out-of-order packets;
4. The “Cross-Layer IPsec” activity, performed as part of the EC FP6 IST project SATNEX, deals with network security issues and particularly focuses on the definition of a Cross Layer-IPsec Authentication Header protocol; UDP is largely used by various delay sensitive as well as streaming applications which can tolerate bit errors in the data payload better than the loss of a full packet; a cross-layer IPsec-based security architecture, using on UDP-Lite can allow forwarding corrupted packets to applications;
5. The EMERSAT project, funded by the Italian Space Agency (ASI), deals with satellite solutions for applications and communications services to national institutional agencies dedicated to security and emergency management; the main contribution to this project is the emulation of heterogeneous satellite-terrestrial networks to support emergency based on the use of the Satellite Tool Kit (STK) and the integration of other simulation/emulation tools; another aspect faced concerned the correct handling of satellite-dependent
technology to guarantee network-level Quality of Service (QoS), i.e. Control Access - CAC, and then guaranteeing target performance;
6. The INTERSECTION project is a Collaborative Project co-funded by the European Commission in the context of FP7 ICT area and, particularly, under the “Secure, Dependable and Trusted Infrastructures” subprogramme area; the main contribution to this project is the design of an intrusion system based on a satellite network architecture dealing with attacks exploiting PEP vulnerability; the presence of Performance Enhancing Proxies (PEPs) at the edges of satellite links leads to the twofold effect of improving TCP performance but also increasing a PEP-related vulnerability due to violation of the end-to-end semantic of TCP;
7. The SENSIBLE project (ESA Artes 3/4) aims to provide traffic shaping (at IP level) to ships sharing a common return link (DVB-RCS) in order to let the ship owner to flexibly assign capacity to its fleet without the need of explicit interaction with the Satellite Operator and its NCC;
8. TELESAL project (ASI), focused on the integration of multi-segment networks to a satellite central platform to support and provide telemedicine; in particular tests were performed over the emulated DVB-RCS platform using real equipment for an EEG (Electro-Encephalo-Graphy).

The analysis of the systems and related protocols involved in the previous projects, as well as tests on the proposed enhancements, have been carried out through two different test platforms: a satellite network simulator based on the Network Simulator Ns-2 and a Linux-based emulator platform. Both the platforms support DVB-RCS DAMA algorithms and are alternatively used depending on the objective of the study case and are described in the next section.

1.1 Testbed General Description
1.1.1 Simulation Platform
A DVB-RCS simulation platform has been built on the satellite network extensions of the Network Simulator NS-2. The reproduced scenario models a star-based architecture where multiple Satellite Terminals (STs) are connected to a satellite gateway (GW), including the Network Control Centre (NCC) functionalities, through a geostationary “bent-pipe” satellite node. The configuration scripts also allow for intra-segment mobility simulations, where a mobile node detaches from a satellite node and attaches to a wireless node and vice versa, simulating either soft and hard handover techniques. A Performance Enhancing Proxy (PEP) module can be attached to every configured node (i.e. on board the satellite) allowing to “split” TCP connections and to select a different transport protocol over each sub-connection.

In addition, a new module for DVB-RCS MAC layer has been implemented in the satellite stack. All the DAMA schemes envisaged by the standard are supported (i.e., CRA, RBDC, VBDC), as well as their possible combinations (i.e., mixed CRA-VBDC). The structure of the implemented DAMA model for Ns-2 is represented in Figure 1, mentioning the names of the new agents/classes introduced. In Figure 1 it is also detailed a possible path for data IP packets (plain arrows) and messages exchanged at MAC layer (dashed arrows). MAC signaling is not crossing the satellite link, but it is part of the same MAC NS2 instance. Nevertheless satellite delays are accounted internally to simulate the real signaling latency.

Encapsulation of packets in MPEG is not performed but can be included as overhead.
A vast gamut of statistics (graphs or data files) can be obtained as simulation output through TCL scripts. In particular, such outputs can be divided into three categories:
1) Lower layer statistics – these concern allocation dynamics at the MAC layer
   a. “System Response Time” measurements – time elapsing between a capacity request and the corresponding allotment;
   b. Traces of the allocated slots per superframe in the different DAMA disciplines;
   c. Calculation of the “saturation time” – time at which the maximum allowed transmission rate is achieved (significant for TCP long transfers).
1 http://www.tlcsat.uniroma2.it/DAMA/
2) *End-to-end statistics*
   a. Round Trip Time (RTT);
   b. Throughput

3) *TCP statistics* – monitoring of the evolution of TCP internal variables:
   a. Congestion Window (cwnd) and Slow Start threshold (ssthresh) trends;
   b. Packet sequence Numbers;
   c. Trace of the retransmissions.

The simulator framework was developed and adopted mainly for the study of Handover involving satellite and wireless terrestrial links, protocol tuning for the TOP project and the Sensible project for the system dimensioning.

### 1.1.2 Emulation Platform

A Satellite emulator has been conceived and developed to reproduce in real-time the services and quality of service of a DVB-RCS satellite network, with some network impairments (e.g., end-to-end delay) reproduced by ad-hoc software modules. The Emulator core is composed of 5 units. Every unit-PC emulates a single component of a typical VSAT satellite environment, with DVB-RCS or DVB IP and terrestrial return channel. Both star and mesh architectures can be alternatively configured. In particular, PCs represent:

- Gateway (Access Router)
- NCC/Hub (SatGW);
- Satellite (SAT);
- Satellite Terminal (ST) which hosts several virtual STs;
- User Terminals (UT) which hosts several virtual UTs.

Figure 2 shows the PC’s logical interconnections, and the interfaces provided, such as connections with real hardware and external networks (PSTN, Wi-Fi). The emulator provides the interface for a wireless extension in order to integrate satellite segment with other real or emulated wireless systems. With reference to Figure 2, a WiFi access point, behind which a real WiFi network is configured, can be attached to a virtual ST. The platform can be accessed and used remotely through a web interface, since is connected to the Internet and has a public address. In this way, a remote user can access to one of the WEB interfaces available (platform configuration, test execution, real-time plots, etc) in order to configure a target satellite scenario, run tests and collect statistics, or even access to use the entire platform with command shell (ssh) or use some of its components as data gateway/tunnel.to test remote applications. Finally, the emulator can be interconnected with other platforms, both locally and remotely, or be a part of a bigger emulation environment and exchange data with the other components in real time.

![Figure 2: Emulator Architecture](image)

The hardware equipment of the emulator is shown in Figure 3, deployed into a small-sized rack.
Software modules have been developed over Linux OS in order to take advantage from the Open Source strategy. The kernel used is version 2.6.20, which offers a large set of functions for classifying and scheduling network traffic, IPsec support and a large set of transport protocols including UDP-Lite and new variants of TCP (i.e., TCP Hybla, TCP Westwood+, TCP Reno, TCP Low-Priority).

Finally, DAMA functionalities rely on the same algorithms implemented on the NS-2 simulator and work intercepting traffic from ST (or multiple virtual STs), managing a queue at application layer (ip_queue) and delivering requests on a dedicated channel via an UDP socket. DAMA allocation decision is performed centrally in the NCC by a dedicated software module, and the allocation response is broadcasted back to all STs to allow, on IP packet basis, the data output with a strategy close to TDMA adopted in real DVB-RCS systems. With this approach the emulator proved good performance comparable to real systems.

Since real packets and applications are involved in the emulations, performance analysis can be done using classic network analyzer tools (e.g. Wireshark/Ethereal).

In order to guarantee the reliability of the measurements for the tests performed over the platform, a first validation activity has been carried out, to tune the emulator platform with a reference real DVB-RCS system. In Table 1 it is shown the comparison of execution of ping with different DAMA policies over a real scenario (Alcatel A9780 Release 2.0 Gateway with EMS RCS Terminals) and over the emulation platform. Ping packets have default size and a frequency of 1 Hz.

<table>
<thead>
<tr>
<th>DAMA Policy</th>
<th>RTT (ms)</th>
<th>Real Scenario</th>
<th>Emulated Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRA</td>
<td>AVG</td>
<td>633.690</td>
<td>640.014</td>
</tr>
<tr>
<td></td>
<td>Std.dev</td>
<td>9.693</td>
<td>15.06</td>
</tr>
<tr>
<td></td>
<td>Min</td>
<td>610.092</td>
<td>601.914</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>654.131</td>
<td>672.513</td>
</tr>
<tr>
<td>RBDC</td>
<td>AVG</td>
<td>794.999</td>
<td>766.597</td>
</tr>
<tr>
<td></td>
<td>Std.dev</td>
<td>58.872</td>
<td>54.606</td>
</tr>
<tr>
<td></td>
<td>Min</td>
<td>664.192</td>
<td>668.591</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>1625.529</td>
<td>1696.361</td>
</tr>
<tr>
<td>VBDC</td>
<td>AVG</td>
<td>1257.503</td>
<td>1293.430</td>
</tr>
<tr>
<td></td>
<td>Std.dev</td>
<td>411.453</td>
<td>388.663</td>
</tr>
<tr>
<td></td>
<td>Min</td>
<td>614.319</td>
<td>612.427</td>
</tr>
<tr>
<td></td>
<td>Max</td>
<td>2115.071</td>
<td>2183.794</td>
</tr>
</tbody>
</table>

Table 1: Ping results (Round Trip Time)

The achieved results are very similar. Some differences are present, especially in standard deviation and range for CRA ping time, mainly due to the DAMA framing structure, which do not impact on overall protocols performance.

The Emulator platform has been used and/or extended mainly in the frame of these research project:
1. TOP
2. EMERSAT
3. INTERSECTION
4. SENSIBLE
5. TELESAL.
2 TCP OVER DVB-RCS

2.1 Overview

The reference model for a DVB-RCS broadband network is shown in Figure 4. The network topology is a star network with the GW station as hub.

![Figure 4: DVB-DCS reference scenario](image)

The GW transmits a forward carrier towards all STs. The STs, in turn, share a channel towards the GW according to a Demand Assignment Multiple Access (DAMA) protocol. The GW typically provides access to the Internet and/or to a corporate network. STs cannot communicate directly with each other, but some systems allow double hop connection via the GW. DVB-RCS bandwidth management is asymmetric. In the forward link a TDM scheme is used. This is managed by the GW, and the way the available bandwidth is shared between concurrent flows is an implementation issue. Typically, the GW maintains several FIFO queues with different priority. In the return link, the bandwidth is shared between ST’s according to a DAMA scheme. Three schemes of bandwidth allocation are considered:

- **CRA (Constant Rate Allocation)** permanently assigns a fixed amount of bandwidth to an ST irrespective of whether it is used;
- **VBDC (Volume Based Dynamic Capacity)** dynamically allocates bandwidth on request of the ST; the request is for a given volume of data; each request is a one-off request for a given volume, and requests are cumulative; the time between issuing the request and getting the allocated bandwidth is typically in the order of 1.5 s, but may vary from one system to another depending on the details of the frame structure and the precise allocation algorithm; the standard supports also an Absolute VBDC (AVBDC) scheme in which requests are absolute rather than cumulative, typically used to cope with potential loss of requests; in the following analysis only to VBDC will be considered;
- **RBDC (Rate Based Dynamic Capacity)** dynamically allocates bandwidth on request of the ST; the request is for a given data rate and is typically valid for some seconds, being updated by subsequent requests; the extra delay of the request/allocation cycle is only felt on the initial request for RBDC, and on requests that modify the requested rate; the impact of dynamic allocation on RTT is therefore less severe than for VBDC.

Some systems support a fourth type of bandwidth allocation: FCA (Free Capacity Assignment). Any bandwidth not used for fulfilling requests is split between STs in some way. This feature is not included. It requires knowledge of, or assumptions about the total traffic in the channel, and its effect is therefore unpredictable from the viewpoint of a single or a few STs.

From a performance point of view, the three types of allocation can be described as follows:

- CRA has a constant RTT of around 500 ms for a geostationary satellite channel; bandwidth is reserved whether or not it is used; any unused bandwidth is wasted;
- VBDC has a much longer RTT, typically around 1.5 s, due to the request/allocation cycle, normally, capacity will only be requested for data that is already in the buffer, so the bandwidth utilization is 100%;
- RBDC has an RTT similar to CRA, except some extra delay on first request and on adjustment of requested rate; the bandwidth provided will rarely match precisely the bandwidth needed, so some bandwidth will be wasted most of the time, but less than for CRA.

Most implementations offer combinations of two, or all three of the allocation schemes.

The precise algorithm for requesting and granting VBDC and RBDC is not part of the DVB-RCS standard, and it is difficult to get information of the details of implementations. So we had to make some assumptions:

- VBDC is rather straightforward, and there are not many choices available;
- RBDC, on the other hand, is much more open to different algorithms; we chose a relatively simple model based on a sliding average of recent traffic demand.

In a DVB-RCS environment, the use of DAMA mechanisms implies that TCP can experience an RTT well above the two-way propagation delay and the available capacity may vary strongly and abruptly. Nevertheless, packet loss is not
meaningful, since DVB-RCS applies a strong forward error correction at the lower layers, providing to upper layers a quasi-error-free channel. When TCP operates over a DAMA-based satellite link, two nested control loops govern the data transfer process, namely the TCP flow control loop and the DAMA bandwidth allocation control loop. In these conditions TCP connections may underperform and some protocol adjustment will be proposed.

2.2 Testbed Description

In order to verify TCP based connection behavior, both simulation and emulation platform were used. The same allocation and request algorithms compliant with the DVB-RCS DAMA specification were used. Simulation represents a powerful opportunity to evaluate performance of transport layer protocols over the target network. For this purpose Ns-2 has been identified and enhanced with DVB-RCS custom extensions. Nevertheless, not all aspects, especially at higher layers, can be included into an event driven not-interactive environment.

Emulation is instead able to reproduce transport layer working closer to real case, including the capability to use real applications such as web browsing and ftp.

With regard to already deployed DAMA enabled DVB-RCS systems, the following assumptions were made on both simulation and emulation platforms:

- Superframe duration is set to 96 ms;
- Return channel maximum capacity is 2.048 Mbit/s;
- Time elapsing from a bandwidth request until the response, called access delay, is around 1 s (including elaboration time, synchronization with superframe structure, insertion of the BTBP in the DVB-S framing structure, etc.).

For both simulation and emulation, the lower layers and signaling of a complete DVB-RCS systems are not fully implemented, but the main characterization of a real system is guaranteed thanks to either simplified approach (e.g. simple TDMA instead of the general MF-TDMA), additional ad-hoc contributes to access delay (in order to take into account lower layer delay contributions e.g. interleaving) and simplified implicit signaling.

The platforms are described in more details in Section 2.1 and Section 2.2.

2.3 Trials and Demonstrations

DAMA exploitation introduces a significant delay in addition to the already high propagation delay of the satellite link, and affects significantly the behavior of TCP data transfer. This effect is present on both data traffic directions, but it is less evident on the forward link data transfer, because in this case the return link (where the DAMA mechanism is implemented) handles small packets corresponding to TCP acknowledges (ACKs) and has small rate variations. A common slot size in the DVB-RCS return channel is a multiple of an ATM cell, so that each ACK just requires at maximum a single slot. Data packets for TCP transmission are usually bigger and require a greater amount of slots, implying greater allocation variations since the DAMA algorithms work under competition and resource constraints.

The first test concerns the measure of a single TCP connection sequence number evolution in both emulated and simulated approach for TCP Reno and TCP Vegas over DAMA RBDC (Rate Based Dynamic capacity) access scheme. Then RTT and throughput as tested as well.

2.4 Results

Figure 5 shows that with the two approaches the same results are carried out and that performance are significantly different for the two TCP flavors. In fact, TCP Vegas clearly suffers from bad estimation of available bandwidth due to the high variability of perceived RTT in a DAMA system.

Figure 6 shows RTT variations as a function of time for a TCP Reno connection over DAMA with VBDC (Volume Based Dynamic Access) both in the emulated and in the simulated environment. Although the match is not perfect, the overall behavior is similar, with a common average RTT of about 1.5 s.

Figure 7 shows the throughput for TCP Reno connection evaluated both through simulation and emulation. The results from the two approaches are close in average and represent quite well the real system behavior. Oscillations peaks on the Linux line are above the system capacity because of buffering at MAC layer not included in the TCP dump performed at the source, but the average throughput is compliant with the system capacity set at the beginning.

With the aim to optimize TCP performance in such environments, the examined tests are important also to identify the criticalities of the DVB-RCS system. In particular, for classic TCP congestion control algorithms the initial growth of the transmission window is inefficient in the startup phase, even with high ssthresh as it is actually in Linux PC (and consequently set in NS2 too). Furthermore, Vegas estimation of optimal transmission rate at steady state is fooled by the DVB-RCS asymmetric and RTT-variant links. The problem of defining a more efficient transport protocol must cope with all these issues, but still keeping the typical fairness of concurrent TCP flows.
3.1 Overview

The point-to-point satellite link between I-PEPs can be considered as a controlled environment with characteristics fairly well known and managed (e.g. MAC buffer size). To improve on the above inefficiencies over DVB-RCS links, the design of a new TCP-based protocol has been developed with the aim of satisfying three particular requirements:

1. to optimize the transmission of short object (web traffic);
2. to guarantee a fair behavior with other competing flows and efficient resource sharing in case of larger transfers;
3. to efficiently operate over all the DVB-RCS DAMA schemes.
In addition, it must also respect the condition to be friendly with flows adopting standard TCP.

TCP Noordwijk (TCPN) is designed as an I-PEP-compatible and TCP-friendly transport protocol to efficiently perform in DVB-RCS environment. Compatibility with transport protocols of other vendors I-PEPs is ensured because sender-side only modifications are introduced, with SCPS-TP receiver kept unchanged, while friendliness with standard TCP is achieved forcing TCPN flows to use only the bandwidth not used by flows.

Mainly, TCPN replaces the “window-based” transmission with a “burst-based” transmission. The transmission of bursts of packets is characterized by two variables: the burst size (BURST), which is the number of packets to send in one shot, and burst transmission interval (TX_TIMER), which is the time interval between two consecutive transmissions. BURST and TX_TIMER are updated according to ACK-based measurements, but not using the ACK reception timing of classic TCPS.

At the start of connection, TCPN transmits by using default burst setting, BURST\(_{\text{INIT}}\) and TX\_TIMER\(_{\text{INIT}}\). These values are accurately chosen in order to be compliant with typical HTTP traffic characteristics. BURST\(_{\text{INIT}}\) is chosen of the order of magnitude of the dimension of most of the typical web objects and at the same time not too big to affect TCP performance of concurrent flows, because a too big BURST might keep the transmission buffer occupied too long affecting fairness and friendliness. TX\_TIMER\(_{\text{INIT}}\) is chosen such that the initial transmission rate (BURST\(_{\text{INIT}}\)/TX\_TIMER\(_{\text{INIT}}\)) satisfies some pre-determined rate requirements. This initial “blind” phase lasts at least 1 RTT (typically more than one second) that is the time necessary until one or more ACKs are received. After the initial blind phase, TX\_TIMER is updated to optimize system capacity utilization with a reference maximum burst value BURST\(_{\text{INIT}}\). On the other hand, actual BURST used for transmission is tuned according to congestion detection principles but always kept smaller or equal than BURST\(_{\text{INIT}}\). Congestion of the I-PEP \(\leftrightarrow\) I-PEP link can be controlled thanks to two factors:

- the communication environment is assumed to be controlled, buffers can be tailored according to the expected traffic patterns in the point-to-point links and then able to accommodate spikes in the incoming traffic (e.g. maximum number of TCPN connections starting simultaneously);
- TCPN implements a burst rate control scheme aiming to privilege short transfers; in other words, to compensate the “blind” transmission of new connections, longer connections reduce their rate behaving as “low-priority” connections.

### 3.2 Testbed Description

The tests on TCP-Noordwijk were performed first on the simulation platform, implementing from scratch the protocol in C++. The simulation allowed the fine tuning of the protocol parameters, identifying all the margin of improvements and fixed inefficiencies due to the nature of the DVB-RCS link. TCP-Noordwijk protocol architecture and functionalities have been introduced into NS2, as a new agent called TCP/Burst and modifying the TCP Sink module to support the SNACK option (Sink/Snack). TCP/Burst agent has been based on standard TCP NS2 implementation, with a main modification consisting in removing the dependence between the process managing packet transmission (send\_much method) and the ACK reception (recv method). A new procedure is in charge to perform burst-based transmission regulated by two global state variables, burst size (BURST\(_{\text{INIT}}\)) and transmission timer (TX\_TIMER\(_{\text{INIT}}\)). Other new procedures implement Burst Rate Control and Flow Control algorithms, which update BURST\(_{\text{INIT}}\) and TX\_TIMER\(_{\text{INIT}}\) on the basis of statistics coming from the ACK-based Capacity and Congestion Estimation component. Finally, other changes concern error handling and retransmissions according to the SNACK method.

Simulation scenario includes a satellite link between two I-PEPs, one at the ST side and the other at the HUB/Gateway/NCC block side, which constitute the two end points of TCP-based connections. The target satellite link is based on a DVB-RCS star-based architecture with capacity of 10 Mbit/s in the forward link and 2 Mbit/s in the return link. A simplified but realistic DVB-S link configuration is already available in NS2 as an Aloha channel including the physical delay of a geostationary communication link, with a typical physical RTT of 500 ms. This NS2 link has been adjusted to match the transmission rate of 10 Mbit/s. As claimed by the DVB standards, the link is dimensioned to work in Quasi Error Free conditions (Bit Error Rate –BER– less than 10\(^{-10}\)), so no error model has been added to the simulated link. Such a low BER in fact is leading to a packet loss every several hours, and is easily recovered by the SNACK mechanism without major impacts to the protocol.

A DVB-RCS DAMA access scheme for the return link has been realized from scratch in NS2 and developed on purpose. Each ST can have a granted number of slots in Constant Rate Assignment (CRA) and a maximum allowed request in terms of slots for both Rate-based and Volume-Based Dynamic Capacity (RBDC and VBDC respectively). The maximum bandwidth for the return link has been set to 2 Mbit/s and divided into 32 slots per frame. With these constraints, ST issues periodical capacity requests compliant with the already described DVB-RCS DAMA standard. All buffers involved in the simulated node have been tailored to avoid losses but still of reasonable size (not infinite).

The same protocol and architecture were implemented into the lunch environment of the Emulation platform, to perform last validation tests.
3.3 Trials and Demonstrations

All the tests presented herein envisage TCP connections flowing from ST to HUB (return link). First of all, TCPN sender behavior reproduced by the TCP/Burst NS2 module has been validated through a preliminary simulation test, also to prove correct implementation. The scope of the other tests is to evaluate performance of the new TCP algorithm highlighting the effects of DAMA on transferred data packets and the benefit of the burst based transport protocol approach.

3.4 Results

At the transport layer, TCPN, TCP Reno and TCP Vegas are alternatively utilized. Results in Figure 8, Figure 9 and Figure 10 highlight TCPN capability to quickly reach the maximum transmission rate independently on the running DAMA scheme. In fact, after the initial blind phase, where rate has been pre-configured at about 450 kbit/s\(^2\), TCPN runs the Tracking Rate algorithm that quickly reaches the maximum allowable rate (2048 kbit/s). Instead, both TCP Reno and TCP Vegas gradually increase the transmission rate on a RTT basis. The initial underutilization of the available bandwidth confirms the inefficiency of TCP congestion control over large bandwidth-delay links, especially for short transfers. In case of a static resource allocation (CRA), both TCP Reno and TCP Vegas need nearly 40 s to reach the maximum rate (see Figure 8). In addition, TCP Vegas attempt to estimate the optimal \(cwnd\) fails when RTT is variable over the time for the DAMA mechanism. Basically, TCP Vegas interprets RTT variations directly related to the congestion level and reduces \(cwnd\) accordingly.

![Figure 8: CRA throughput comparison for transmission over return link](image)

In case of the RBDC (Figure 9), TCP Reno needs 80 s to achieve the maximum rate, while TCP Vegas steadies its rate to about 200 kbit/s (a tenth of the available bandwidth).

![Figure 9: RBDC throughput comparison for transmission over return link](image)

Figure 10 shows instantaneous throughput trends of TCPN connections when CRA is adopted. Results confirm TCPN fairness in sharing bandwidth. In addition, the thick line represents the aggregate throughput over the whole simulation time. Other than fairness, TCPN guarantees also optimal channel utilization. In fact, the overall throughput is always close to the total return link capacity. The slight oscillations are due to TCPN dynamics aiming to track the “fair” rate. In more details, TCPN continuously tries to approach the maximum channel rate (Tracking Rate algorithm) but, if other connections are running, this leads to RTT increase with a consequent reduction of the rate (Adjustment

\(^2\) Note that initial settings are arbitrarily tuned to match a target initial rate and hosting network, usually tailored to optimize http transfers.
Rate algorithm) that allows the RTT decrease. In turn, RTT decrease triggers back Tracking Rate algorithm, and so on. Therefore, throughput oscillation cannot be avoided, but they are usually below the 10% of the overall bandwidth.

Figure 10: Throughput for multiple connections adopting CRA for transmission over return link

In case either RBDC or VBDC is run as access scheme, TCPN continues to provide a good fairness in the bandwidth sharing independently on the access delay variability. As a drawback, throughput oscillations are bigger. The rationale relies on the RTT increase due to the access delay, which slows down TCPN dynamics described above for the CRA case.

RBDC leads to large and variable RTT only when there are significant ST rate changes. Otherwise, RTT tends to be similar to that experienced with CRA. In the considered scenario, this happens only when a TCPN connection stops. In line with this, the aggregate throughput (Figure 11) presents the bigger oscillations close to the connection 1 stop (50 s) and the connection 2 stop (180 s).

Figure 11: Throughput for multiple connections adopting RBDC for transmission over return link

4 SATELLITE-WIFI HANDOVER SIMULATION

4.1 Overview

The reference scenario to provide connectivity to a Mobile Node (MN), shown in Figure 12, is composed of:

- one satellite segment (SAT), which through the wide coverage of a GEO satellite reaches the MN in all its service area;
- \( n \) independent wireless LANs (WiFi) scattered (even partially overlapping) into the above defined satellite coverage area and identified by access points (APs).
As a term of reference, the SAT technology can adopt the aforementioned DVB-RCS mobile standard, WiFi adopts 802.11b standard, MIPv6 (Mobile IPv6) standard is considered and TCP is adopted as transport layer protocol.

Mainly, MIP envisages the definition of a location management agent, called Home Agent (HA), which keeps the association between a home permanent IP address assigned to the Mobile Node (MN) in the home network and several care-of-address (CoA) assigned when visiting other networks. In this way, connection between MN and the Correspondent Node (CN) is kept alive irrespective of the access network serving the MN.

During the TCP connection life, one or more handovers could be performed as a consequence of user mobility. Since WiFi networks usually provide higher bandwidth and are characterized by lower propagation delay with respect to satellite networks, switching from SAT to a WiFi access point is preferable when possible (overlapping of both WiFi and satellite coverage).

4.2 Testbed Description

The baseline simulation script has been setup to simulate a data transfer according to the described scenario, assuming a SAT channel of 1 Mbit/s with RTT fixed to 500 ms and a WiFi “Hot Spot” area with a bandwidth of 2 Mbit/s and a RTT delay of 20 ms. Data packet dimension is fixed to 1460 bytes. The amount of bytes transferred and the instantaneous throughput are measured, as a function of the handover latency variations, for a long data transfer application, such as FTP, in both the communication directions, that is from Mobile Node (MN) to Correspondent Node (CN) and vice versa.

At the beginning of the FTP session, the MN is connected to a WiFi AP. During the transfer, three handover events occur:

a) the MN moves from WiFi to Satellite,
b) the MN switches to a new WiFi AP,
c) MN connects again to the satellite link until the end of the simulation.

The disconnection events occur at 10 s, 30 s and 43 s for a), b) and c) handover event respectively.

In the case of Cross-Layer modified stack, at 9.98 s, 29.5 s and 42.98 s (which are the disconnection time of a) b) and c) minus 1*RTT of the destination network) the L2 trigger is generated, followed by the proper cross-layer procedures.

The data transfer simulation has been repeated varying the handover latency in a range from 0.1 s up to 5 s for satellite originated HO. In the case of WiFi originated handover latency starts at 2.5 s since disconnection lasts at least 5 RTT of the destination network.

FTP data transfer uses TCP standard New Reno with and without Cross-Layer actions optimized according whether MN-to-CN or CN-to-MN direction is considered.

4.3 Trials and Demonstrations

To ensure continuous access to the Internet for mobile and nomadic users, it is necessary to consider an integration among different kinds of telecommunications networks. In particular, switching between satellite and terrestrial networks, such as WiFi, can represent an optimum solution because high capacity and low delays can be achieved under the WiFi “Hot Spot” coverage and long range mobility and service continuity can be ensured thanks to satellite links where WiFi access is not available. In this scenario a Handover (HO) must be performed when a Mobile Node (MN) switches from a point of access to another one of the same system, or between points of access belonging to different systems (even adopting different standards), keeping communication alive. If HO is performed in a network adopting the internet protocol stack and offering multimedia broadband IP services, also exploitation of upper layers functionality (network, transport and application) is affected and proper actions must be undertaken at each layer to keep the relative protocol performance unaltered.

Mobile IP (MIP) protocol provides mobility support at network layer, allowing mobile nodes to change point of
access to an IP network transparently. MIP presents a problem called “HO Latency”, or out of service time, and it is not able to mask the impairments of such latency to higher layers. In fact, a MIP handover results in an interruption of the communication at layer 3 with consequent packet losses. The duration of such interruption is usually composed of the time to connect MN (Mobile Node) to the new link (HO performed at layer 2) and the time needed to update routing (HO performed layer 3).

4.4 Results

After having performed all the simulations, it was observed that at transport layer, some problems occur after a HO event, because TCP protocol is not designed to handle mobility. In particular, when HO is performed between links with different delay and bandwidth-delay product (BDP), HO causes reordering and generation of segment bursts, Bandwidth-Delay product sudden variation and Expiration of TCP's time-out.

4.4.1 Reordering and generation of segment bursts

The packet reordering can occur after a HO, if consecutive packets are routed through different paths in case the delay of the new path is lower than the old one. This occurs only in the case of soft-HO, where the MN exchanges data with both satellite and WiFi link, before physically switching of the mobile device. Burst generation can be due either to TCP cumulative ACK scheme, where the acknowledgment for the first packet received after the switch will anticipate the acknowledgments still in transit over the old link, or due to the loss of several ACKs over the old link.

4.4.2 Bandwidth-Delay product variation

The bandwidth-delay product (BDP) is a very important parameter in a window based protocol such as TCP because it determines the amount of data that can be in transit in the network. After a HO occurrence some problem can be experienced:

1) a large number of packets can drop when moving from satellite to WiFi, because BDP_{sat} > BDP_{WiFi} and then the most of the packets injected over the new network leads to buffers overflow increasing the experienced latency;

2) moving from WiFi to satellite, since TCP enters in Congestion Avoidance phase (linear cwnd increase on a RTT basis) when cwnd is far away from the BDP_{sat}, the increase of the cwnd up to the optimum value requires long time, resulting in a temporary significant underutilization of the satellite resources.

4.4.3 Time-out expiration

When a HO from WiFi network to satellite occurs, due to high delay of the new link, RTO expiration can be experienced because the last packets received through the old link can be acknowledged after that RTO, evaluated on the basis of the old link delay, expires.

Due to standard TCP dynamics, shown in Figure 13, after a link disconnection, a certain number of sent packets/ACKs will be lost and RTO is initialized with reference to the first unacknowledged packet.

![Figure 13: TCP standard behavior](image)

Depending on the “handover latency”, one or more RTO expiration and consequent retransmissions can occur. When MN is reconnected to the new link, TCP sender might be not yet able to re-start the transmission since RTO is not yet expired. In this way, an additional contribution T* is generally added to the “handover latency” causing a further performance degradation in terms of throughput. This added time is a function of HO latency and varies from 0 to RTO_k. At the end, transmission restarts with \( cwnd \) set to 1 and \( ssthresh \) halved as many times as the number of RTO expirations.

To overcome inefficiency introduced by TCP standard, it is possible to introduce a Cross-Layer interaction between link, network and transport layers. The scope is to make TCP aware of the incoming HO in advance. In this way, TCP can perform proper actions aiming to both avoid losses during HO and efficiently restart the transfer over the new link.
To evaluate performance of such a method, NS 2 simulation platform has been improved with ad hoc modules that reproduce such cross layer interactions.

Figure 14 and Figure 15 show the instantaneous throughput achieved over time during the data transfer by standard and cross-layer enabled stack from MN to CN (Correspondent Node) between WiFi to SAT link and vice versa respectively, assuming a SAT channel of 1 Mbit/s with RTT fixed to 500 ms and a Wi-Fi “Hot Spot” area with a bandwidth of 2 Mbit/s and a RTT delay of 20 ms.

In these simulations, the disconnection event occur at 10 s where the MN moves from WiFi to SAT link and at 30 s where MN moves from SAT to WiFi link, considering for both a HO latency $T_{HO} = 5$ s. Both figures show a significant improvement of performance, after the disconnect event due to the cross-layer action. Performance degradation of TCP standard is due to the HO latency and $T^*$. 

![Figure 14: Instantaneous throughput from WiFi to SAT link](image1)

![Figure 15: Instantaneous throughput from SAT to WiFi link](image2)

5 SATNEX “CROSS-LAYER IPSEC” ACTIVITY

5.1 Overview

The reference protocol stack herein considered is shown in Figure 16.
Modern multimedia codecs are designed to be error resilient. For instance, MPEG video coding sends data using three different frame types: I-, P- and B-frames. I-frames hold information about an entire video frame, while P- and B-frames only include the differences with respect to other frames. Usually, it is better to deliver damaged P- and B-frames than discarding them. MPEG-4 provides higher compression with greater error robustness at a large range of bit rates. MPEG-4 video standard includes new features such as object-based coding, error resilience and improved compression. Error resilient tools in MPEG-4 video do not reduce errors like FEC or ARQ, but reduce quality degradation caused by errors (i.e. use error concealment).

Reliable Multicast Transport (RMT) protocols use various error/erasure correction codes to protect against channel impairments. Mass file delivery consists of one-to-many data communication using UDP transport over IP. File Delivery Over Unidirectional Transport (FLUTE) defines a specific file delivery application of Asynchronous Layered Coding (ALC), adding the following specifications: definition of a file delivery session built on top of ALC, including transport details and timing constraints, in-band signaling of the transport parameters of the ALC session, in-band signaling of the properties of delivered files and details associated with the multiplexing of multiple files within a session.

Transport layer – UDP-Lite is used instead of UDP. The main difference between UDP and UDP-Lite is the partition of each packet into sensitive and insensitive parts. An application uses the Checksum Coverage field, to indicate the number of bytes from the start of the UDP-Lite header (including the pseudo header) that are to be considered sensitive to bit errors. As shown in Figure 17 the difference between UDP and UDP-Lite is that the 16-bit Length field in the UDP is replaced by a 16-bit Checksum Coverage field. Since the receiver only calculates checksum over the sensitive part, any bit errors in the sensitive part results in a packet drop, while packets with errors in the insensitive part are forwarded to the application.

Network layer – Either IPv4 or IPv6 can be alternatively selected. IP checksum is computed only on the IP header field to verify that the IP header was not damaged. In IPv4, such a checksum is mandatory, while IPv6 relies on both the link CRC and transport checksum to assure IP header integrity.

In addition, CL-IPsec is used to provide security services at the network layer. Herein, AH protocol in the transport mode is implemented. As a benchmark standard IPsec AH protocol is considered.

Link-layer – Link-layer must calculate the CRC according to the payload type, i.e. in case of UDP-Lite a partial checksum must concern only the link layer header and UDP-Lite sensitive part. Header protection techniques can help to avoid bit errors in the sensitive part (possibly used in combination with link header compression).

In the framework of Internet security, IETF has standardized the IP security protocol (IPsec) with the aim to offer inter-operable cryptographically-based security services (confidentiality, authentication, integrity and non-repudiation).
while continuing to use the existing infrastructures.

Such services are provided through an authentication protocol, named Authentication Header (AH), a confidential protocol named Encapsulating Security Protocol (ESP), and an Internet Security Association Establishment and Key Management Protocol (ISAKMP). These protocols have been designed as an IPv4 upgrade and as predefined security for IPv6.

The used cryptographic/authentication algorithm and keys of the IPsec services are defined through Security Associations (SAs). A single SA can support the use of AH or ESP, but not both. IPsec operates in two modes: transport and tunnel mode. The former is used between end-systems and adds a new header (AH or ESP) to the IP guaranteeing the protection of the IP payload. In tunnel mode the end-system delegates the security service to the gateway. In this mode, AH or ESP header encapsulates the entire IP packet and a new IP encapsulation is formed, whose destination and source addresses can be different from those of the encompassing IP packet.

IPsec provides various security services at the cost of increased overhead. Especially in the tunnel model, the IPsec overhead implies inefficient bandwidth utilization as well as a higher probability to have bit errors within the header. Header Compression (HC) protocols can mitigate these drawbacks. In fact, Header compression over IPsec (HCoIPsec) aims to reduce overhead, without compromising the security services provided by IPsec. HCoIPsec framework relies on two assumptions:

1. Existing HC protocols are considered;
2. HC protocols operate at the IPsec SA endpoints (HC applied in a SA basis).

Since existing HC protocols compress packets on a hop-by-hop basis, HCoIPsec requires the extension of the HC functionalities in order to operate at IPsec SA endpoints. Furthermore, HCoIPsec framework proposes that the configuration of the HC parameters is accomplished by the SA management protocol (i.e. IKEv2) while compressed packet can be identified through the Next Header field of the security protocol (AH or ESP).

Performing HCoIPsec, outbound IP traffic is first properly compressed and then encrypted/authenticated. Similarly, inbound IP traffic is first decrypted/authenticated and then decompressed. An example concerning AH in tunnel mode is shown in Figure 18.

![Figure 18: HCoIPSec Example: AH in tunnel mode](image)

With or without the application of HC techniques, in the frame of the considered cross-layer architecture, UDP-Lite and the IPsec protocol suite are intrinsically incompatible. In practice, IPsec performs its security tasks on the entire IP payload, irrespective of the UDP-Lite protocol that identifies a sensitive and an insensitive part. Therefore, IPsec discards all packets with one or more bits corrupted. The UDP-Lite capability to manage corrupted bits in the insensitive part is then prevented resulting in the net performance of UDP-Lite to be similar to UDP.

To avoid the intrinsic incompatibility between IPsec and the partial payload coverage of UDP-Lite, IPsec must identify the UDP-Lite sensitive part within the IP packet (only for UDP-Lite packets), which requires access to the checksum coverage field in the UDP-Lite header. This task can be performed through a Cross-Layer interaction between IPsec and UDP-Lite. Furthermore, the integrity check must be performed in accordance with UDP-Lite checksum coverage: the UDP-Lite insensitive part should be excluded from the Integrity Check Value (ICV) calculation.

Therefore, the design of the cross-layer IPsec (CL-IPsec) protocol has been divided into two different phases:

1. Identification of an appropriated cross-layer method;
2. Design of a new algorithm for the ICV calculation.

A. Design of the cross-layer signaling

The first step is to identify the checksum coverage value in the UDP-Lite header.

Cross-Layer methods are mainly classified on the basis of the presence or absence of signaling between the involved protocol layers. Specifically, in an implicit cross-layer design, cross-layer interactions are defined in the design phase without any exchange of control information during protocol operations. On the contrary, explicit cross-layer requires exchange of control information between participating protocol layers. For instance, application explicitly informs
transport layer on which bytes are to be considered sensitive; implicit CL signaling can be used to modify link layer to provide partial error check.

From the analysis of IP, IPsec (AH in transport mode) and the UDP-Lite protocol, two particular aspects can be highlighted:
- UDP-Lite header size is fixed (8 bytes);
- IP and AH header sizes are not fixed, but known to IPsec.

On this basis, the AH protocol can know the sensitive bytes through the definition of a new “cross-layer” pointer that is dynamically associated to the Coverage field in the UDP-Lite header. This makes effective an implicit cross-layer interaction between transport and network layer. Such a modified IPsec (AH) protocol is called CL-IPsec.

B. A new algorithm for the ICV calculation

Once cross-layer signaling is defined, the algorithm for the Integrity Check Value (ICV) calculation must be modified to take into account only sensitive bytes. In particular:
- The insensitive part of the UDP-Lite packet must be excluded from ICV calculation;
- The new ICV algorithm must run only for UDP-Lite packets; standard procedures are considered for all the other packets (i.e. UDP, TCP) making, CL-IPsec compatible with any other transport protocol.

The routers can modify certain fields in the IP header. These fields are called mutable fields, for example Type of Service, Flags, Fragment Offset, Time to Live, Header Checksum, Explicit Congestion Notification. All mutable fields are set to zero before computing the ICV at both sender and receiver ends.

In CL-IPsec, this approach is also used to manage bits comprising the insensitive part. Upon receiving UDP-Lite packets, CL-IPsec acquires the checksum coverage value from the UDP-Lite header (implicit cross-layer), and accordingly sets to zero all the bits of the insensitive part before computing the ICV at both sender and receiver sides. In this way, packets with corrupted bits in the insensitive part are not discarded.

C. Security services

IPsec offers a flexible set of security services. These services are:
- Data origin authentication/Connectionless data integrity. Assurance that in an IP packet, the source address, destination address, and packet payload cannot be maliciously or accidentally modified in transit without detection by the receiver.
- Replay protection. A replay sequence number is used to avoid replay attacks. Furthermore a replay sequence number window is defined and only packets whose sequence numbers were within such a window are accepted.
- Confidentiality. This ensures that only the intended receiver is able to decrypt the received data.

The IPsec uses AH and ESP to provide various combination of security services. The AH protocol provides authentication for connectionless integrity, data origin authentication and (optionally) replay protection. Instead, ESP ensures confidentiality, data origin authentication and data integrity, anti-replay service.

Since CL-IPsec has been tailored towards the AH protocol, alongside security services provided by standard AH, CL-IPsec ensures partial integrity (only the UDP-Lite sensitive part) and data origin authentication.

5.2 Testbed Description

CL-IPsec envisages modifications in the ah4 module of the Linux kernel and is implemented in the kernel release 2.6.20.1. In order to both validate the implementation and to evaluate benefits coming from the proposed CL architecture a satellite emulation platform has been set up. Specifically, 3 PCs have been interconnected, as shown in Figure 19, with the following configuration:
- ST1 and ST2 represent the end-systems of a satellite link: e.g. satellite terminal and a satellite gateway;
- SAT represents a transparent GEO satellite and introduces physical delays in both the communication directions and bandwidth constraints.
5.3 Trials and Demonstrations

Either a CL-IPsec or an IPsec SA can be established between ST1 and ST2: the security protocol is AH and SPD/SAD databases are manually configured (no rekeying, infinite SA lifetime). UDP-Lite is run as transport protocol (a standard feature since 2.6.11), while both IPERF and VLC, patched to run over UDP-Lite, are used as applications.

To emulate an error-tolerant link layer, possible bit errors are generated at the network layer (just before entering CL-IPsec/IPsec routines) of the receiving application end-systems (usually ST2) using a random error generation with a uniform distribution.

5.4 Results

IPerf tests consist in the transfer of dummy packets from an iperf server (ST1) to an iperf client (ST2) over UDP-Lite. Test parameters are summarized in Table 2.

<table>
<thead>
<tr>
<th>Duration</th>
<th>Packets Length</th>
<th>Bandwidth</th>
<th>Checksum Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>180 sec</td>
<td>1460 byte</td>
<td>1.05 Mb/s</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 2: Transmission parameters

Both IPsec and CL-IPsec SA were alternatively configured for a set of BER values ranging from $10^{-6}$ to $10^{-2}$. In case of CL-IPsec, IPERF application explicitly set checksum coverage field through the option “-u” followed the desired sensitive part size.

The packet loss rate (PLR) of the two techniques was compared also mathematically under uniformly distributed errors using the aforementioned parameters. A mathematical model is also considered as a benchmark. The results are shown in Table 3, the subscript “num” are results using mathematical model and “sim” are results obtained from the testbed.

<table>
<thead>
<tr>
<th>BER</th>
<th>CL-IPsec_{num}</th>
<th>CL-IPsec_{sim}</th>
<th>IPsec_{num}</th>
<th>IPsec_{sim}</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00E-02</td>
<td>98.47%</td>
<td>98.79%</td>
<td>100.00%</td>
<td>100%</td>
</tr>
<tr>
<td>1.00E-03</td>
<td>34.05%</td>
<td>35.50%</td>
<td>100.00%</td>
<td>100%</td>
</tr>
<tr>
<td>1.00E-04</td>
<td>4.07%</td>
<td>4.30%</td>
<td>70.17%</td>
<td>70.87%</td>
</tr>
<tr>
<td>1.00E-05</td>
<td>0.42%</td>
<td>0.36%</td>
<td>11.39%</td>
<td>11.20%</td>
</tr>
<tr>
<td>1.00E-06</td>
<td>0.04%</td>
<td>0.02%</td>
<td>1.20%</td>
<td>1.17%</td>
</tr>
</tbody>
</table>

Table 3: Mathematical Analysis of CL-IPsec and IPsec

It can be inferred from Table 3 that the CL-IPsec outperforms IPsec, since packets with corrupted insensitive parts are not dropped. This property of CL-IPsec is also evident from Figure 20. The metric for comparison is packet loss rate perceived versus the BER.
The improvement in packet loss rate offered by CL-IPsec consequently results in an enhancement in performance for error-tolerant applications, while continuing to provide security services. In Figure 21 the packet loss rate is provided as a function of the Checksum Coverage value ranging from 8 to 45 bytes. Two BER values were considered: $10^{-4}$ and $10^{-5}$.

Tests were conducted to evaluate the ability of MPEG-2 and MPEG-4 video codec over either UDP-Lite/CL-IPsec or UDP-Lite/IPsec protocol stack to support high quality video streaming although bit errors affect received packets. Video streaming was set up between two VLC applications (a sender and a receiver) running over ST1 and ST2 respectively. The UDP-Lite checksum coverage value is defined in the VLC network configuration file. To change it, configuration file must be modified and VLC must be recompiled. In all the considered cases, checksum coverage involves both application and transport header. Test parameters are summarized in Table 4. Specifically, the video streaming duration is 2 minutes and video format is 720x576. BER varies from $10^{-6}$ to $10^{-3}$ and the transmission bit rate is set to 1.02 Mbit/s. Both UDP-Lite/CL-IPsec and UDP-Lite/IPsec protocol stacks were alternatively configured.

<table>
<thead>
<tr>
<th>Streaming Duration</th>
<th>BER</th>
<th>Protocol stack</th>
<th>Codec</th>
<th>Bit rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>120 s</td>
<td>$[10^{-6}-10^{-3}]$</td>
<td>UDP-Lite/CL-IPSec</td>
<td>MPEG-2</td>
<td>1.02 Mbit/s</td>
</tr>
<tr>
<td></td>
<td></td>
<td>UDP-Lite/IPSec</td>
<td>MPEG-2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>MPEG-4</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Video streaming: test parameters

Results are shown in the Figure 22 and display the number of dropped frames by application vs BER. Dropped frames are less than 100 overall the BER. However, MPEG-4 outperforms MPEG-2 by reducing the dropped frames by about 20% when using UDP-Lite/IPsec. Finally, CL-IPsec allows a further performance improvement, thanks to an
increased number of bytes passed to the application codecs. As a consequence also the performance gap between MPEG-2 and MPEG-4 is reduced.

Figure 22: Video streaming with CL-IPsec and IPsec

Figure 23 and Figure 24 show two frames relative to the video streaming tests respectively with MPEG-2 and MPEG-4 codec, when BER=$10^{-5}$. It is evident that UDP-Lite/CL-IPsec allows a better video quality compared with UDP-Lite/IPsec. Of course, the rationale is the major number packets that UDP-Lite/CL-IPsec forward to the application, although corrupted in the insensitive part.

Figure 23: Frames of video streaming test with MPEG-2

Figure 24: Frames of video streaming test with MPEG-4

These results demonstrate the correct operation of CL-IPsec AH with UDP-Lite and affirm the advantage of using UDP-Lite for error-tolerant multimedia.

6 EMERSAT PROJECT

6.1 Overview

The emulation platform has been integrated in a more extended platform based on the use of the Satellite ToolKit (STK) for the EMERSAT project. Specifically, a federation of simulation/emulation tools has been integrated to provide a flexible architecture well suited to support analysis of heterogeneous terrestrial-satellite scenarios. STK acts as coordinator of the whole platform, performing the following actions:

- Configurations of the application scenario;
- Scheduling of the simulated events (terminal motion, transfer start, parameter changes);
- Triggering of distributed tasks on the various federated tools
- Collections and visualization of results.

STK interacts with the satellite emulation platform to activate services of a telecommunication-advanced system: mechanisms at the network layer, transport layer, satellite resource management and implicit or explicit interaction among protocol layers. In this way, satellite emulator allows to investigate on most relevant issues of actual broadband satellite systems while interfacing with the largely deployed terrestrial systems.
In addition, STK exploits three further tools focused on the control and management of the communications:

1. **Call Admission Control (CAC) tool** - this tool enhances the platform with CAC procedures needed to guarantee target quality of service at the flow level in terms of guaranteed bandwidth and delays and at the call level in terms of blocking probability and call breaking down probability;

2. **Dynamic resource assignment tool** – this tool manages a dynamic assignment of resources available over the satellite link, based on both the control theory and methodologies of Operational Research; algorithms to compute bandwidth requests can be properly integrated to scheduling algorithms that Earth Station runs at the IP level;

3. **Multidimensional link budget tool**: this tool performs multi-dimensional link budgets for transponder in Ku/K/Ka bands, both bend-pipe and regenerative, which can be applied for different coverage areas and various propagation models; such a tool is able to interface with external databases to retrieve data concerning large set of areas; in particular, it envisages an interface with a database concerning propagation models of satellite signal through the atmosphere.

6.2 **Testbed Description**

The Satellite Network Emulation Platform (SNEP) is composed of five Linux-based computers. With reference to Figure 25, each computer plays the role of one of the macroelements of a real star-based satellite network:

- A computer acts as one or more user terminals, meant as end-systems behind satellite terminals, through a set of virtual machines (VMs) based on the VmWare software; each virtual user terminal generates heterogeneous IP traffic running real applications;

- A set of Return Channel on Satellite Terminals (RCSTs) is reproduced through VMs on the same physical host; each virtual RCST is mainly in charge to run Demand Assignment Multiple Access (DAMA) algorithms to access the return link, the application of Quality of Service (QoS) policies for the incoming traffic and optionally the enabling of a Performance Enhancing Proxy (PEP);

- A Satellite channel machine introduces propagation delays, bandwidth constraints and errors in the two communication directions transparently;

- Both the satellite Hub and the Network Control Centre functionalities are located in a single machine; the hub manages transmission and QoS policies on the satellite forward link; the NCC mainly runs a DAMA controller in charge to evaluate all the capacity requests coming from RCSTs and to assign all or part of the requested capacity according to both the overall traffic conditions and the Service Level Agreement (SLA) between customer and Service Provider;

- The last machine is a gateway to Internet services, hosting locally some reference applications too (ftp, HTTP and VoIP).

![Figure 25: SNEP platform](image)

The whole emulation platform is based on IP interfaces, with dedicated signaling channels for emulated DVB-RCS messages. The star topology of the satellite network is realized using IP-IP tunnels and ad-hoc routing to make the packets traveling over satellite twice (double hop) avoiding a direct connection. In the case of mesh configuration (i.e. Skyplex or RCS-RCS), although the communication with the NCC is similar to the DVB-RCS star network case, the data communication between two Satellite Terminals is direct (single hop). On the NCC and on every RCST, there is a DAMA software component for handling of the resources of the satellite return link. With this tool it is possible to approximate a MF-TDMA access scheme with multiple carriers and multiple terminals, reflecting the operational emergency scenario identified. DAMA plays a key role in the current satellite systems, also outside of the DVB-RCS
standard, since it guarantees an efficient sharing of the limited satellite resources. Furthermore, a manager for QoS is added to the DAMA software component, to enable a DiffServ-like packet processing. The SNEP network can be considered as a DiffServ cloud, where Access Router connected to the Hub and RCSTs are edge routers that perform the marking of incoming packets. In all the internal nodes, five class of service are provided: Assured Forward, Best Effort and three dropping class of Expedite Forward. In practice, a two-level QoS scheme is applied to classify data and forward it to a DiffServ queue and to MAC layer respectively. The emulation platform is integrated with the other emulation and simulation tools, according to the logical scheme shown in Figure 26:

6.3 Trials and Demonstrations

Emersat project foresees trials related a network architecture able to simultaneously manage different sub-networks, composed of a large number of satellite terminals, either fixed or mobile/nomadic, operated by different Institutional Bodies and jointly involved in the emergency management.

A particular emphasis will be devoted to mobile/nomadic satellite terminals. Application scenario envisages a sudden deployment of such terminals in the emergency area with the aim to restore the connectivity within a given time threshold defined by the operative requirements and provide the required telecommunication services with the respect of a pre-defined prioritization of the services.

Satellite segment reproduced by the emulator platform will manage communication services by respecting the following requirements:

- Implementation of a flexible network topology able to establish satellite links among different terminals deployed over the emergency area and operated by different Bodies, and between terminals and a remote control center.
- Provision of broadband for applications requiring an high data rate (i.e. up to 2 Mbit/s).
- Performance compliant to a pre-determined prioritization of various services/applications in terms of latency, round-trip delay, QoS management.
- Implementation of an IP interface on satellite terminals to make easier and faster the integration with terrestrial systems.
- Guarantee security requirements in terms of restricted satellite channel access, authentication and protection of data contents.

6.4 Results

Several results were obtained during the testing campaign. A couple of tests are reported in Figure 27 Figure 28, showing the network load under typical usage conditions, for both the forward and return link. These plots are obtained using the real-time monitoring function of the emulation platform, where the data exchanged is accounted during the emulation process and updated accordingly.
7.1 Overview

From a security perspective, the utilization of a satellite segment in an integrated architecture leads to vulnerabilities that can be classified as follows:

- **“Satellite-specific” vulnerabilities**, which completely rely on satellite characteristics and technologies (usually involving lower protocol layers);

- **“Non satellite-specific” vulnerabilities**, which are common to all the IP-based networks (i.e. IP data corruption, interception, etc.);

- **“Satellite intersection-based” vulnerabilities**, which rely on satellite technology needed to support interconnection with terrestrial networks.

The latter class involves technology solutions aimed to guarantee good performance over the satellite link. In fact, the use of traditional TCP/IP stack and related paradigms implies poor performance since the basic features of the protocol design don’t match with the actual environment. A common baseline to mitigate the problem envisages the use of Performance Enhancing Proxies (PEPs) at the edge of satellite links with the main scope to “accelerate” TCP over satellite. PEP is mainly based on a splitting architecture, which divides end-to-end connections into multiple sub-connections, each one adopting a transport protocol suited to the link characteristics (i.e. delay, bandwidth, BER). Over the satellite link, ad-hoc transport protocols usually replace standard TCP.

7.2 Testbed Description

Although performance improvement is significant, PEP-based architecture is intrinsically vulnerable due to the violation of the TCP end-to-end semantic. A PEP intentional or unintentional failure (i.e. dropping of locally acknowledged packet before reaching actual destination) leads to an irreversible break of the end-to-end reliability. Implications on security are straightforward since TCP is used to provide transport services also to comply with severe requirements in terms of reliability.

Figure 29 shows a possible attack scenario.
Specifically, a malicious user in the same network of a target legitimate TCP source can access PEP in front of satellite gateway with the aim of installing a malware application that performs the task to drop either a part or all TCP packets flying towards Internet through the satellite link. Different harmful effects can be caused depending on both malware implementation and considered application protocol:

- loss of several packets within a connection, which causes continuous retransmissions, for instance after a retransmission timeout expiration, increasing transfer time;
- packet dropping, which does not allow the application to successfully conclude its operations and end-system is aware of the negative transfer outcome;
- packet dropping, which is transparent to applications that trusts on a successfully transfer.

To face with the above described attack, an Intrusion Detection System (IDS) has been developed. It integrates different functions and components (monitoring, detection, reaction, remediation, visualization and topology discovery), which cooperate to increase network protection and security in a real-time and automated fashion. The goal is to gather data from probes and network elements, to analyze it, to detect intrusions, and to select the most suitable remediation action against the detected attack. In addition, off-line functions aim at using data coming from the network or provided by the real-time part of the framework in order to help the human operator in analyzing the network state and make the network more secure.

7.3 Trials and Demonstrations

In the frame of the INTERSECTION project an intrusion system based on the above architecture dealing with attacks exploiting PEP vulnerability was designed.

Satellite IDS exploits two types of probes: the SYN detector and the TCP traffic analyzer. The “SYN detector” is installed in the satellite gateway and is in charge to monitor all the TCP traffic in order to create an IPFIX record for every SYN/FIN exchange through the satellite link. Each record includes the parameters identifying the specific connection and it is enhanced with the time information. “TCP traffic analyzer” probes run on the access router of all the networks interfaced to the satellite network. Such probes aim to collect statistics about TCP traffic coming from and going to the satellite network. Specifically, a TCP traffic analyzer grabs the number of transferred bytes over any active TCP connection.

All the probe records are collected by a data broker, which represents the first functional element of the IDS. Data broker is in charge to forward the received records to the Detection Engine, which implements the following detection logic:

1. **Step 1** - Compare the number of SYN and FIN records coming from the SYN detector probe; if \#SYN ≠ \#FIN, regular operations are assumed; otherwise (i.e. \#SYN - \#FIN > threshold = 90) detection process moves to step 2;
2. **Step 2** - Delete <SYN ; FIN> pairs related to a same connection; residual SYN records will provide information on networks involved in possible attack.
3. **Step 3** - For each connection under investigation, compare the amount of bytes crossing corresponding source-destination networks; results are computed in the form of “byte differences”.

The achieved results are then forwarded to the decision maker that determines if what has been detected as an anomaly can be interpreted as an attack from an external entity and thus forward the relative commands.

The Reaction and Remediation (ReaRem) subsystem involves a Reaction Engine, a repository of scripts used for the attack remediation, and a Remediation Point, which represents the network element where such scripts will be run. Specifically, when an attack is detected (decision maker issues a trigger), the Reaction Engine downloads the proper script from its repository to stop PEP misbehavior. This script, executed in the Remediation Points/PEPs, performs a procedure aimed to temporary disable PEP involved in the attack in order to recover correct settings: delete of the...
malware and reset PEP routing tables. Deployment of the overall IDS system over the considered scenario is shown in Figure 30.

![Figure 30: IDS for PEP-related vulnerability](image)

### 7.4 Results

Target vulnerability has been reproduced in a communication scenario where a satellite geostationary link, operated by Telespazio (TSP) is used to interconnect a remote terrestrial LAN, belonging to Polska Telefonia Cyfrowa (PTC), with an Internet Service Provider (ISP) represented by Telefonica (TID).

Test bed core relies on hardware in the loop configuration where a real satellite link (Intelsat IS906 @ 64° E) operating at Ku band with a channel of 300 kHz is accessed by two satellite modems. At the edges of such a satellite link, both a Satellite Gateway/Network Control Centre (satGW/NCC) and Return Channel Satellite Terminal (RCST) functionalities are emulated through Linux-based machines. PEP agents are installed in both satGW/NCC and RCST machines.

The application implemented for test purposes is based on a traffic generator, running on a Telefonica (TID) machine. Basically TCP connections with a machine installed in PTC are generated over different ports on the basis of the following input parameters: average rate of the overall data flow, the amount of bytes to transfer over any connection. In the tests presented below, the average rate is set to 200 kbit/s and the amount of byte in a single connection is taken equal to 100 kbytes.

Results are summarized in Figure 31 and Figure 32. Over the first 210 s of simulation, ordinary operations are detected on the network: the number of active connections oscillates around the expected average (50) and throughput measured at both sides of the satellite link is similar. The selected IDS configuration envisages an active connection threshold of 90 s and a threshold for the throughput asymmetry equal to 75 kbit/s. In general, these values have to be tuned on the basis of a training period where characteristics of the ordinary traffic profile are observed.

At 210 s, malware starts running. Malicious packet dropping does not allow correct connection terminations. As a consequence the number of active connections grows more and more (see Figure 31). At about 250 s, the number of the active connections overcomes the threshold value. Then, IDS performs the second step of the detection process comparing measured throughput at the edges of the PEP-PEP satellite link. From Figure 32, it is evident the high experienced asymmetry. During the attack, TCP senders (Figure 32-a) continue to transmit without suspecting any packet dropping. On the other side, TCP receiver receives only a minimum part of packets (attack is designed to allows transmission of sporadic packets in order to keep connections on) and experienced throughput in much lower than that measured at the sender side.

The joint compliance of the above conditions with characteristics assumed for the attack profile leads to the issue of an alert message that, in turn, triggers the remediation procedure. Then, after a short time needed to disable PEPs instances involved in the attack, a reliable connectivity is restored and connections starting afterwards are successfully performed. With a particular reference to the Figure 32-b, it is possible to observe a decrease of the overall throughput due to the stop of TCP acceleration provided by PEPs.
8 SENSIBLE PROJECT

8.1 Overview
The idea is to supply the SATCOM market with a service capable to distribute dynamically the bandwidth and managing the quality of service for a set of users within the same administrative domain, to which a fixed amount of bandwidth is assigned from the satellite operator. The system will perform bandwidth management regardless the management policy at data link layer. In fact, it will be able to ensure appropriate QoS only on the network level (third OSI layer).

8.2 Testbed Description
The traffic-oriented simulations are executed using the Network Simulator 2 (NS2) framework. It is a well known and consolidated simulator, particularly suitable for the study of packet-based communications, including MAC layer simulation, satellite physical impairments, TCP/IP stack, etc.

NS2 is an open source project, which leaves the possibility to change its core functionalities in the source code (C++ language) available to the community. The interface to the user is through TCL scripts, allowing the configuration of tests to run, with the topology definition, the parameters adjustments and the right timing of events. NS2 in this way has the flexibility to have a pre-compiled fixed and fast set of core functionalities, directed by a scripting language which can easily modify the simulation (duration, number of nodes, etc.) without touching the core functions. Nevertheless several functionalities necessary for the testing of Sensible system are missing from the original NS2 suite or need a specific setup, so they have been added as core functions in C++ or as TCL routines.

NS2 already includes DiffServ architecture blocks to handle different traffic priorities. It is necessary to define an edge router, responsible to mark packets and apply a policy according to pre-defined rules. In NS2, the rules can be applied with reference to source and destination node identifier, with a pre-defined Committed Information Rate (CIR). This means that only packets of a given flow, within an upper bound for the resulting rate, are marked with a DSCP (Differentiated Service Code Point) value. The DSCP value is then used to route the packet to the right QoS queue in a set of Random Early Drop (RED) queues. The RED queues are emptied according to pre-defined policy rules and have a probability of random early drop increasing in inverse proportion of priorities. Following nodes in a DiffServ domain are only required to have the same set of queues to perform what is called Per Hop Behaviour.

In the Sensible testing scenario, the DiffServ node is assumed in the sensible box, before the effective bottleneck of the end-to-end-link. In this way it is possible to make higher priority flows (in Figure 33 EF) suffer less losses than lower priority ones before the variable output bottleneck enforced by the Sensible Box.
Specific modifications were added to the NS2 source code in order to make the CIR marking rate adjustable in real-time. The sensible message can in this way contain both information on the overall allowed output rate, and the single queue settings, which can be adjusted during the execution of the simulation. The new command for the adjustment of the policer profile is the following:

```
updatePolicyCIR [source id] [dest id] new_bw ;# kbit/s
```

![Figure 33 – NS2 DiffServ architecture](image)

8.3 Trials and Demonstrations

The general reference scenario, configured for simulations, is represented in the Figure 34.

![Figure 34 – NS2 Simulations scenario](image)

The preliminary test performed include, in first instance, the Application Messages overhead, the message delivery times due to the satellite environment, and interaction of the shaping performed at layer 3 with the DAMA of DVB-RCS and the PEPs for performance enhancement.

8.4 Results

Message overhead was measured and impact for less than 5% of the satellite bandwidth assigned to each terminal. The sequence diagram obtained for the bandwidth allocation, using the COPS (Common Open Policy Service) protocol, is shown in Figure 35.
The average results in milliseconds (ms) for the message exchange times are presented in Table 5.

<table>
<thead>
<tr>
<th></th>
<th>CRA Unloaded network</th>
<th>CRA Traffic on forward link and on return link</th>
<th>RBDC Unloaded network</th>
<th>RBDC Traffic on forward link and on return link</th>
<th>VBDC Unloaded network</th>
<th>VBDC Traffic on forward link and on return link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensible Delivery Time</td>
<td>320 ms</td>
<td>340 ms</td>
<td>320 ms</td>
<td>330 ms</td>
<td>320 ms</td>
<td>330 ms</td>
</tr>
<tr>
<td>Sensible ACK Time (unsolicited COPS message)</td>
<td>600 ms</td>
<td>620 ms</td>
<td>1420 ms</td>
<td>700 ms</td>
<td>1420 ms</td>
<td>1550 ms</td>
</tr>
<tr>
<td>Handshake complete/explicit COPS message exchange</td>
<td>770 ms</td>
<td>780 ms</td>
<td>1700 ms</td>
<td>810 ms</td>
<td>1700 ms</td>
<td>1810 ms</td>
</tr>
</tbody>
</table>

Table 5 – IPT-1.2 results

Concerning the bandwidth allocated, it was verified that the layer 3 allocation is affecting the DAMA resources assigned at layer 2, as shown in Figure 36 (theoretical in red line vs effective in green).

9 TELESAL PROJECT

9.1 Overview

Telemedicine concept can be applied to several medical activities: monitoring of remote equipment, home care, first aid (also on board of planes, ships and off-shore oil platforms), support to disaster management, consultation, access to medical files, share of medical information and records, etc. To provide telemedicine services is considered one of the
most efficient solutions to improve quality of health care optimizing costs and to care patients located in remote sites. In fact, through the utilization of an efficient telecommunication infrastructure, a lot of information can be shared among a number of medical personnel and entities, allowing faster, cost effective and high quality diagnosis, quicker and more efficient medical support in remote sites, limiting travelling and/or patient transportation in a significant subset of situations.

A key aspect of telemedicine is to reduce costs and offer flexibility in use cases, still maintaining a high degree of quality in the medical service. The use of a satellite network can be of great advantage in a number of scenarios. The intrinsic characteristics of a satellite network, such as availability, coverage and ease in relocation of satellite terminals, make it very suitable for this study-case. It is also possible to define an architecture composed of a core network realized with a satellite infrastructure and terrestrial wireless links, to extend coverage (for example indoor) and ensure flexibility.

9.2 Testbed Description

A DVB-RCS emulation platform allows to test both real applications and TCP/IP protocols in a communication scenario similar to the real one, but greatly increasing flexibility and avoiding costs of expensive satellite capacity and terminals/hardware leases. It is however necessary that the emulated environment matches the real system as much as possible. The emulation tool considered herein is called Satellite Network Emulation Platform (SNEP), already validated to match a real system behavior. It is realized with a network of five computers, each one performing the functionalities of one or more elements of a real DVB-RCS network.

For the execution of the telemedicine test, the emulation platform has been modified to handle the same subnet mask at both the Gateway and RCST side. In that way it was possible to remove the direct Ethernet cable connecting the EEG (Electro Encephalography) head with the data collecting Server and by-pass transparently the communication through the satellite link (thus the emulated satellite network is completely transparent to the existing Software/hardware already in use at the hospital).

In addition, the SNEP has been enriched with a preliminary support for Quality of Service, through services classification at TCP and IP layer and multiple queues at MAC layer in the RCST. In this way, it is possible to discriminate packets related to higher priority traffic and send them to the right MAC queue, to resemble a DiffServ approach. This addition was needed because of the multimedia nature of the application and the different priorities of its component flows.

In the case of CRA-only for the return link transmission (fixed allocation), the use of QoS does not give major benefit to the application, but in case of using RBDC and/or VBDC dynamic capacity requests its contribution is fundamental, as will be discussed later.

To realize a realistic scenario for a typical EEG exam, the architecture presented in Figure 37 has been set up. On the left bottom there is the visual representation of the devices needed to run the exam in a remote location: EEG machine, microphone, video camera for video streaming and a loudspeaker. In the bottom right it is represented the communication box composed of the SNEP infrastructure (to emulate data transmission over a DVB-RCS satellite link). Finally, in the upper right corner it is represented the specialized doctor in charge of checking the EEG hardware setup (electrodes displacement) and determining the patient health conditions.

![Figure 37: Telemedicine application architecture](image)

9.3 Trials and Demonstrations

Usually 15 minutes of recordings are sufficient for completing the exam, and some preliminary effort is necessary to prepare the setup: for example electrodes must be put on specific location of the patient’s head and must be filled with
conductive gel while their impedances must be checked and be kept low (e.g. less than 10 kΩ). This important preparatory phase must be performed or supervised by specialized technicians (neurophysiopathology technicians) that directly interact with the neurologists responsible of the medical reporting relative to the EEG exam. In this context, it is clear that a communication channel between the patient and the neurologist is necessary, either to visually inspect the patient head (e.g. traumas, sensors positioning, etc.) or to signal abnormalities in the EEG recordings due to the environment (noise, etc.) that must be corrected. In the first case the communication must be monodirectional, from the patient to the neurologist, in the second it must be bidirectional.

In our laboratory, the hardware has been applied on a volunteer patient and a real doctor performed the visit in the "remote" location, as shown in Figure 38.

![Figure 38: Experiment execution](image)

### 9.4 Results

A first set of tests, running the exam with QoS enabled and testing different RCST profiles, allowed to find optimal balance of capacity request strategies, with the aim of limiting the amount of CRA capacity needed (thus the cost). With a proper setup of capacity requests and QoS support, it has been possible to run with a satisfactory degree of quality (evaluated by a doctor) the integrated Telemedicine application, using in particular a mixed allocation of capacity with CRA = 56 kbit/s, RBDC 192 kbit/s and 8 kbit/s in VBDC. With the QoS support, the EEG data and voice call are routed utilizing the available CRA slots, the video uses the bandwidth left from the CRA capacity and requests part (or all, if needed) of the RBDC capacity. Finally, the rest of traffic can use cheaper VBDC allocation and spare capacity from other services. The quality of video has been judged acceptable, although the RBDC allocation introduces delays in the visualization, in addition to the coding/decoding times needed.

This compromise in the requests strategy allowed to better exploit the satellite resources and to leave unused capacity to other RCSTs. In fact, for a complete telemedicine session of 39 minutes, the overall amount of traffic collected (outgoing from the RCST and recorded using tdpump for post-processing matters) was about 55.7 Mbytes, resulting in a net rate of 190 kbit/s (lower than the typical RCST capacity of 256 kbit/s and even lower than the theoretical sum of 242 kbit/s). The measured value differs from the theoretical one due to the variable-rate coding of video which take advantage of static images (the patient moves slowly), together with silence suppression for voice. The use of an adaptive request mechanism (rather than CRA-only) allowed to request only the resources necessary. In the event of only CRA strategy had been adopted, the whole 256 kbit/s channel would have been reserved for the whole duration of the exam, introducing a significant resource waste.

The overall exam quality was evaluated satisfactory for the doctor, who was able to write the diagnosis only on the basis of the multimedia information delivered over the emulated satellite link (brainwave tracks, video, audio).