QoS Issues in a Large-scale IPv6 Network

Christos Bouras[1,2] Dimitris Primpas[1,2] Kostas Stamos[1,2]
[1]Research Academic Computer Technology Institute, PO Box 1122, Patras, Greece and
[2]Computer Engineering and Informatics Dept., Univ. of Patras, GR-26500 Patras, Greece
Tel:+30-2610-{960375, 960316, 960316 }
Fax:+30-2610-{969016, 960358, 960358}
e-mail: {bouras, primpas, stamos}@cti.gr

Keywords: Quality of Service, IPv6, 6NET, IP Premium, real time applications

Abstract
This paper presents our work in the framework of the 6NET project, regarding the design, implementation and testing of a QoS service in a large-scale IPv6 network. The DiffServ mechanism for providing QoS guarantees, is generally preferred over IntServ. However, the current support for QoS mechanisms in IPv6 implementations still lags behind QoS support in IPv4. Furthermore, the new Internet Protocol introduces a different network environment in many aspects and therefore the QoS services should be specifically designed and evaluated for IPv6. In order to evaluate this different behaviour, large-scale experiments have been carried out within the scope of the 6NET project. This paper describes the results from the experimentation with the DiffServ mechanism on a large-scale native IPv6 network that aims to service aggregates of real time traffic with minimum delay, jitter and packet loss.

1 INTRODUCTION

Most networks today can only provide best-effort service for all kinds of traffic. Best-effort treatment causes many problems especially to real time applications (for example videoconferencing applications), because they are sensitive on parameters such as delay, packet loss or jitter. Therefore, these applications are better suited in networks that make use of QoS mechanisms, that can guarantee a level of acceptable service that has been mutually agreed between the network and the user using a Service Level Agreement (SLA). The QoS guarantees can be measured using a number of specific metrics such as the bandwidth that a traffic class uses, the delay that the packets of each class experience, the packet loss and jitter. During the last years several architectures have been proposed in order to provide QoS and some services have already been deployed. Another aspect of the future of IP networks seems to be the IPv6 protocol [1] that updates the existing IPv4 protocol. Its main benefit is that it solved the problem of the limited IPv4 address space, but it is also designed in order to offer a series of additional improvements over IPv4 in various areas such as autoconfiguration, network management, security, mobility and QoS. with its increasing usage. A major force behind the adoption of the IPv6 protocol in the European continent has been the 6NET project [2]. Our work focuses on the combination of these two promising concepts, QoS and IPv6, as it examines how well they operate together.

The DiffServ architecture [3] minimizes the number of actions to be performed on every packet at each node and builds a configuration that does not use a signaling protocol. Individual DiffServ mechanisms are applied on traffic aggregates rather than individual flows. The operation of the DiffServ architecture is based on several mechanisms. The first mechanism is the classifier that tries to classify the whole traffic into aggregates of flows (traffic classes), mainly using the field DSCP (Differentiated Service CodePoint [4]). This field exists in both the IPv4 and IPv6 packet headers. In IPv4 it was part of the field Type of Service (ToS) and in IPv6 that is our focus in this paper, it is part of the field Traffic Class. In addition, the IPv6 packet header also has the field Flow label (20 bits) but it is still experimental and its use has only been recently standardized [5].

The operation of services based on DiffServ architecture uses also several additional mechanisms that act on every aggregate of flows. These mechanisms are packet marking, metering and shaping. In addition, in order to provide QoS guarantees it is necessary to properly configure the queue management and the time routing/scheduling mechanism. The most common queue management approaches use the Priority Queue, Weighted Fair Queue or Modified Deficit Round Robin mechanisms.

Generally, the main problem that has been noticed is that not all IPv4 QoS related mechanisms have been fully implemented to work for IPv6 domains yet. Actually, some IPv4 mechanisms are not planned to be implemented for IPv6 at all, as different techniques are going to replace them. As the usage of IPv6 increases, the support for IPv6 is expected to reach the level of IPv4 support for most router vendors.

The area of Quality of Service has many open issues and is currently studied by researchers intensively. In IPv4

environments, there are many experimental studies that have been implemented and many network providers and educational networks (such as a lot of NRENs (National Research and Education Networks) and GEANT [6]) offer QoS services (IP Premium and Less than Best effort) to their customers [7]. The implementation of these services in IPv6 environment is the next challenge, as well as the use of the new feature of IPv6 protocol, the flow label field [5]. The 6NET project aims to investigate this issue and in this framework several studies and experiments about QoS in IPv6 have been presented. These studies referred to tests in local testbeds that aimed to investigate the behavior of the supported mechanisms ([8], [9], [10]). Consequently, the next step is to move all the tests in a native large scale IPv6 network and try to implement the services and evaluate their performance.

The rest of the paper is organized as follows. Section 2 presents more details about the 6NET project, which has been the framework of our work. Section 3 describes the experimenting environment, the applications used, the structure of the network and the QoS techniques and traffic patterns that have been used for the experiments. Section 4 presents the experiments and the results from each one. Finally, section 5 describes the conclusions from those experiments and the future work that we intend to do on this area.

2 6NET PROJECT

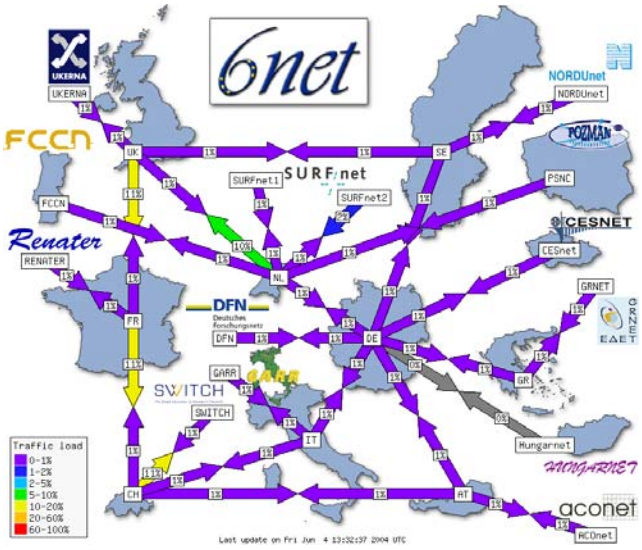

Figure 1. The pan-European 6NET network

The 6NET project (IST-2001-32603) is a European project that intends to demonstrate that the emerging IPv6 technology can meet the requirements of the continued Internet growth [2]. For this purpose, 6NET has built a native IPv6 network that covers most European countries and is extensively used in order to test the IPv6 services and new or legacy applications. The topology of the 6NET network is shown in Figure 1.

6NET partners come from both the academic and the industry world, with a large number of participants being Universities and NRENs. The 6NET project studies the transition from IPv4 to IPv6 and uses the native IPv6 network in order to test basic network services over IPv6 such as routing, DNS and multicast, advanced services like Quality of Service and mobility, IPv6 applications and IPv6 network management. The work we present in this paper is focused on the Quality of Service and applications interaction with IPv6, while the wide spectrum of the 6NET project areas of interest allowed us to view the subject we study with a broad approach to the emerging IPv6 networks.

3 DESCRIPTION OF TESTBED

3.1 Traffic Metrics

There are multiple performance metrics proposed to measure the services provided in a QoS-enabled networks. Most of them are defined by the IETF IP Performance Metric working group [11].

During our tests, the following parameters were used to qualify the QoS services provided:

- One-way or round trip delay. It is defined as the time needed by a packet to be transmitted and fully received by the destination. The overall time consist of the propagation delay, e.g. the time to transmit a bit over long-distance circuits, and the transmission time, e.g. the time to transmit a bit over a specific-speed circuit. As one-way delay measurements require strict synchronization among the monitoring systems. In order to be able to reliably measure delay, clocks at the testing stations were synchronized using stratum 1 NTP server at ntps1-0.cs.tu-berlin.de
- Inter-packet delay variation (jitter) . Inter-packet delay variation is measured for packets belonging to the same packet stream and shows the difference in the one-way delay that packets experience in the network. Large values for jitter usually reveal queuing delays in the network.
- Packet loss. Packet loss is measured as the portion of packets transmitted but not received in the destination compared to the total number or packets transmitted. Large values of packet loss usually shows highly congested networks or frequent sharp increases of the traffic load. Packet loss is measured as the portion of packets lost in the network (or delayed more that a specific time threshold) compared to the number of packets successfully delivered their destination. Packet loss usually reveal congestion in the output queues of the routers.
- Packet reordering. Packet reordering is measured as the portion of packets that are delivered to the destination in wrong order compared to the total number of packets. There are multiple reasons that lead to packet reordering; parallel forwarding engines in high performance routers, per packet

load balancing on parallel physical links, routing path changes, etc. There is a significant impact to the TCP (application) performance even for small packet reordering values.

3.2 Monitoring and Measurement Requirements

Subscription to a premium class of service implies a Service Level Agreement (SLA) is signed between the customer and the service provider. On 6NET, provided SLAs are clearly not commercial agreements but only define performance guarantees that are experimentally provided to portions of traffic.

Monitoring and measuring activities should demonstrate the network infrastructure is able to provide services guarantees to portion of traffic. For example, traffic marked as EF receives preferential treatment compared to BE and LBE, under congestion conditions in the core or the access networks. Also, LBE traffic should also be reduced to a minimum level when other traffic is present.

QoS measurements can be performed with software tools that are able to generate traffic with pre-defined characteristics and measure the performance of the network. A tool that is already extensively used in measurements is iperf [12], which is able to provide accurate throughput and jitter measurements for flows under test, and for this reason was our choice for generating the artificial traffic in most experiments.

Iperf's statistics were produced at the server instance of the Iperf traffic generator and included the average throughput and the average jitter of the UDP traffic and the average throughput of the TCP traffic. Iperf calculates jitter using the RFC 3550 definition that defines jitter as:

$$\mathbf{Ji = Ji\text{-}1 + (| D(i\text{-}1,i) | - Ji\text{-}1) / 16}$$

where D(i,j) is the difference of the interval between two successive packets at the receiver from the interval between two successive packets at the sender, defined as

$$\mathbf{D(i,j) = (Rj – Ri) – (Sj – Si)}$$

For some of the tests the mgen tool was also used, because of its advanced capabilities in producing variations in the artificial traffic according to predetermined scenarios. Also, the Ethereal network protocol analyzer [13] was used in order to capture the packets and then be able to extract metrics like packet reordering, delay, and throughput.

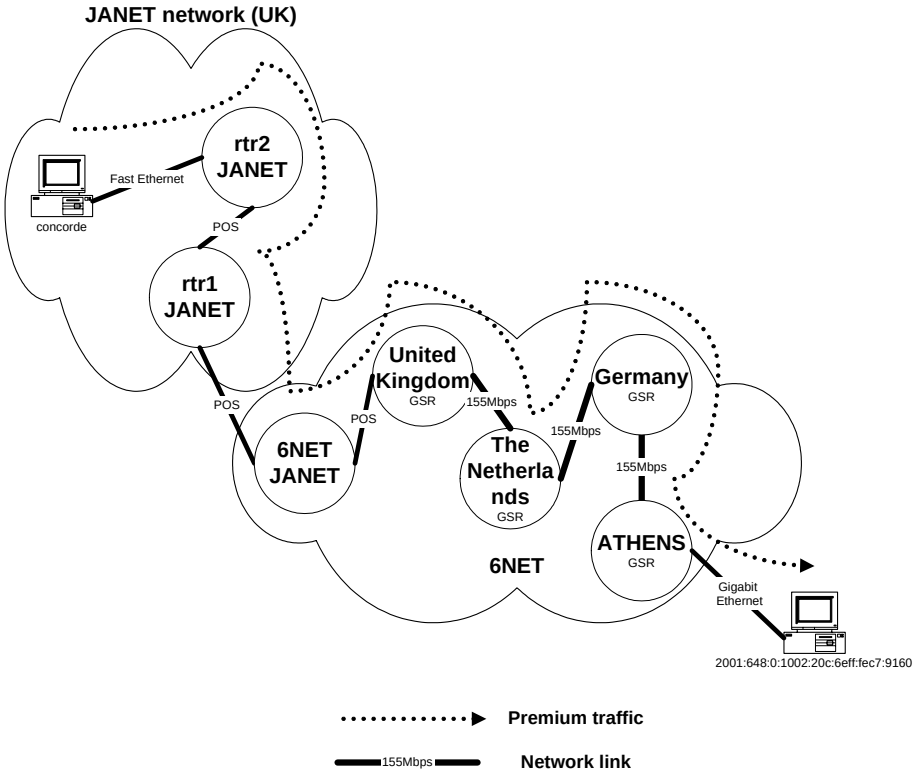

Figure 2. 6NET QoS testbed

3.3 Network Configuration

For the purposes of the experiments we used 3 PCs located at various points in the network. Premium traffic in the foreground was sourced from the United Kingdom (JANET network), was routed through the Netherlands and Germany, and was received at Greece. Furthermore, in the experiments that we wanted to artificially create background traffic, a generating PC at the Netherlands was used which sent traffic through Germany to Greece.

The infrastructure displayed in Figure 2 has been deployed for the IST Project 6NET and is part of the pan-European IPv6 network (the whole 6NET network is presented in Figure 1). The software version that this testbed uses is the CISCO IOS 12.2(13) T [14].

3.4 QoS configuration

The experiments with the QoS mechanisms have been performed throughout the 6NET network. With the examined QoS architecture, three classes of service were supported:

- Premium IP service, based on Expedited Forwarding (EF) [15]
- Best Effort service (BE)
- Less than Best Effort service (LBE)

Packets were identified by a unique DSCP value and mapped to a dedicated queue at each output interface in the backbone links. Setting the DSCP values was a task handled by the authority managing a domain outside the core of the network. In our case this authority was the NREN institutes, each managing a national part of the network. The backbone trusts them, expecting that they will appropriately mark real-time traffic to be handled with the Premium IP service.

To prevent problems caused by improper configuration, the policing and DSCP marking/re-marking were configured first, on the 6NET access routers. If the preferential queueing were to be enabled in the core before the edge policing, all packets marked with the EF DSCP value would start to be forwarded in the priority queue. If the EF queueing were improperly configured in some way, this would have the potential to cause disruption on the network.

In particular, some access points have been defined as trusted to use the QoS services. These access points (NRENs) are responsible to mark the packets with the appropriate DSCP values (46 and 8 for Premium IP and LBE service respectively). The QoS configuration enables policy in all access interfaces for both Premium IP and LBE service. For Premium IP service, the policy action defines that the acceptable rate for the traffic is at most 5% of the capacity of the access link and in case that this limitation is violated, then the packets are dropped. On the other hand, in LBE service, the acceptable rate is 1% of the capacity of the access link. In all the trusted access points, there is a check if there are packets with invalid DSCP value that means packets with DSCP value other than 46, 8 or 0. In this case, the router resets the DSCP value to 0 (best effort service). All the incoming interfaces of the edge routers of the backbone network, that belong to connected sites that are not allowed to use the QoS service, have been configured to remark all the packets in DSCP value 0.

The classification and policing configuration has been applied on all the access (incoming) interfaces of the edge backbone routers that connect the NRENs. In the core routers (packet transmission in the core network), the MDRR scheduling mechanism has been configured properly to treat the packets from the 3 supported QoS services. In particular, 2 queues have been defined, one full priority queue and one normal. The packets that belong to EF class are enqueued in the priority queue instead of all the other packets that are enqueued in the normal queue. This configuration has been applied on all POS output interfaces in the core routers.

4 EXPERIMENTS

4.1 The need for QoS guarantees

Before evaluating the QoS over IPv6 behaviour at the large-scale network, we performed some tests within the 6NET network at a smaller scale, without initially activating any QoS mechanism for the real-time traffic. The point-to-point and multipoint conferences using IPv6-enabled H.323 software took place between the local CTI network and endpoints at Thessaloniki, so they were constrained to the Greek part of the 6NET network. In this part of the network we had greater control so that we could evaluate our procedures and identify possible problems at a more easily managed environment before scaling the experiments throughout the 6NET network in Europe.

Table 1. Quality statistics for IPv4 and IPv6 experiments

	Packets lost	**Average jitter**	**Maximum jitter**	**Average receive time**
IPv4	32%	101	137	220
IPv6	13%	16	103	30

We repeated the connections over both IPv4 and IPv6. Because our traffic was routed through links that are also being used in actual production networks, and since almost all applications and users today are still connected to the Internet over IPv4, the IPv4 network was much more congested than the equivalent IPv6 connections. So naturally the IPv6 conferences took place much more smoothly and with far fewer packet losses, as can be seen in Table 1, which displays the average measurements over a number of conferences. We have to note that the IPv4 network was especially congested, which caused our

measurements to illustrate the very low quality we received from the IPv4 connection. The IPv6 network on the other hand carried lighter traffic, and the quality for the IPv6 participants at the conferences was significantly higher.
As can be seen in Figure 3 and Figure 4, IPv6 also maintains a more stable bandwidth consumption, which also is slightly higher than the IPv4 bandwidth consumption, due to the larger standard header for IPv6. We expect that under the same network conditions the behaviour for both protocols would be more similar to one another.

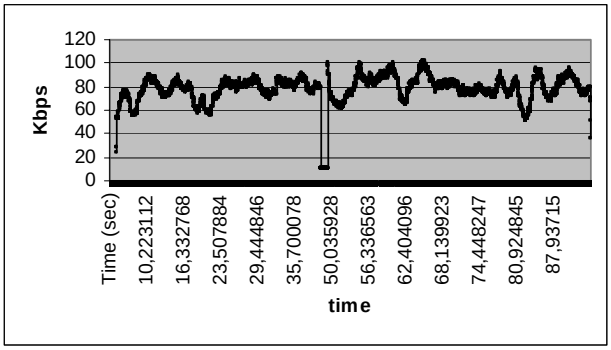

Figure 3. IPv4 bandwidth consumption

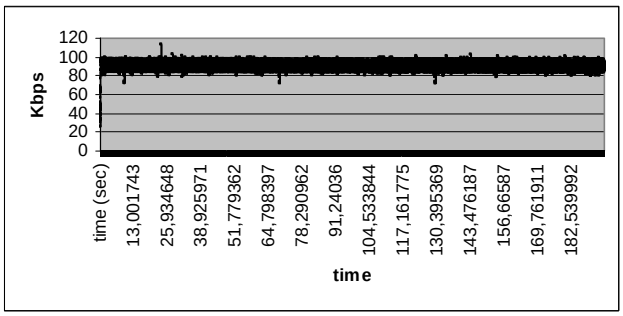

Figure 4. IPv6 bandwidth consumption

The main purpose of this first set of experiments was to identify the necessity for QoS guarantees for the operation of real-time applications over both IPv4 and IPv6 networks across a wide area network. The applications were stable, but they suffered (especially in the case of IPv4) from the lack of any QoS guarantees because of their sensitivity to unfavorable network conditions.

4.2 Investigating the QoS service on a large-scale network

The first stage of the large-scale tests was to verify the correct operation of the marking, policing and shaping mechanisms. After that, we conducted a number of scenarios that included simultaneous UDP and TCP background traffic, while the foreground traffic was alternated between the 2 transport protocols.
The setup for this part of the experiments was simple and aimed to verify that traffic marked with the DSCP decimal value of 46 (IP Premium traffic) was indeed handled preferentially compared to the rest of the traffic. The verification of the configuration of the 6NET core was achieved by a series of small tests. These tests were the following:

4.3 Marking test

The objective was to validate that the marking mechanisms were correctly implemented at the access and core routers. Using the iperf traffic generator, we created a flow with BE packets from the testing PC at Athens, Greece (GRNET) to the one at London, UK (University of Lancaster's concorde). Upon reception, we were able to verify that packets were marked with the proper IPP DSCP value (decimal 46). On the contrary, packets that were sent by the Netherlands testing PC arrived with a DSCP value of 0, verifying that they were treated as best-effort traffic and we could therefore use that flow for simulating background traffic.
The correct re-marking configuration of the core routers was also verified, since packets that were sent by the GRNET access router with a different DSCP value (0), were received at the UK testing PC with the proper DSCP value, which means that they were correctly re-marked by 6NET core routers.
We then artificially congested the network by generating 200 Mbps of background UDP traffic, while simultaneously sending IP Premium traffic. As Table 2 demonstrates, while in general traffic had very large losses (about half of the transmitted background traffic packets were lost), premium marked traffic was able to comfortably traverse the congested links almost without any losses.

Table 2. Comparing premium to best-effort traffic under congestion

	Achieved bandwidth (Mbps)	Jitter (ms)	Packet loss (%)
UDP foreground	5.11	4.1780	0.082
UDP background	105.00	0.1430	49.000

The above results verify that the prioritization mechanism was properly treating marked packets beneficially compared to unmarked or improperly marked packets. The packet loss in foreground traffic is significantly low, but is non zero as

was ideally expected. This was probably caused by the PC generators themselves (the measurements came through the iperf traffic generator), but in any case the packet loss is too low to cause any measurable problem.

4.4 Traffic Policing test

The objective of the traffic policing test was to validate that policing mechanisms are performing as expected at input interfaces. In general, input policing takes place at the upstream providers edge routers towards the customers direction. The applied configuration specified that excessive packets were to be dropped (and not demoted to best-effort traffic). The rate limit was configured at 5% of the total capacity of the backbone lines, which corresponds to around 7.5Mbps of traffic at the physical level. The traffic policing test did in fact verify that when trying to send 10Mbps of UDP foreground traffic from GRNET testing PC to UK testing PC, only 7.10 Mbps of actual traffic got through. Packet loss was at 29%, and average jitter at 0.273ms. The operation of the policing mechanism was further verified during the scenarios which are described in section 4.6.

4.5 Shaping test

The objective of the shaping test was to validate that the shaping mechanism is functioning as expected at output interfaces of partner's access routers and at the input interfaces of the 6NET core routers. In general, output shaping takes place at the customer's side, preferable as much as close to the source. Input shaping in performed by the upstream provider as an additional service to his/her customers, usually when customers are not able to perform output shaping in their routers. We therefore applied shaping at the access router and run this test to verify that the traffic forwarded to the core domain remained within the bandwidth limit.

For this test we used the mgen traffic generator to generate bursty UDP traffic at the GRNET testing PC destined for a receiver at the UK testing PC. The mgen generator was configured so that it was starting and stopping sending traffic every 5 seconds. During the 5-second sending intervals, mgen was sending 5000 packets of 1Kbyte in a period of 5 seconds for an average rate of about 8Mbps.

The shaping mechanism smoothed the transmission rate and the traffic was not policed by the core routers. As a result, the sink point of the experiment received traffic that always remained under the rate limit.

4.6 Scenarios

The setup for each scenario are displayed in Table 3. They have been designed so that we could investigate the effectiveness and characteristics of the implemented QoS mechanisms in order to provide a measurable improvement to various types of traffic that has been marked as premium traffic. In order to more closely simulate actual network conditions, scenarios include combinations of either UDP, TCP or both types of traffic at the background (best-effort traffic). Each of the first six scenarios is a single experiment for some initial evaluation of the network characteristics under various combinations of foreground and background loads, while scenarios 7, 8, 9 and 10 are actually multiple experiments each, in order to more thoroughly examine the network's behaviour and in order to identify in detail the thresholds in its behaviour.

Table 3. Testing scenarios

Scenario	Background	Foreground	Notes
1	50Mbps (30% TCP- 70% UDP)	UDP traffic (1.5Mbps)	
2	50Mbps (30% TCP- 70% UDP)	TCP traffic (1.5Mbps)	
3	80Mbps (30% TCP- 70% UDP)	UDP traffic (1.5Mbps)	
4	80Mbps (30% TCP- 70% UDP)	TCP traffic (1.5Mbps)	
5	120Mbps (30% TCP- 70% UDP)	UDP traffic (1.5Mbps)	
6	120Mbps (30% TCP- 70% UDP)	TCP traffic (1.5Mbps)	
7	80Mbps (30% TCP- 70% UDP)	UDP traffic (1.5Mbps)	And increase it in steps of 0.5Mbps
8	80Mbps (30% TCP- 70% UDP)	TCP traffic (1.5Mbps)	And increase it in steps of 0.5Mbps
9	120Mbps (30% TCP- 70% UDP)	UDP traffic (1.5Mbps)	And increase it in steps of 0.5Mbps
10	120Mbps (30% TCP- 70% UDP)	TCP traffic (1.5Mbps)	And increase it in steps of 0.5Mbps

Table 4 summarizes the results from the above scenarios for the experiments with UDP foreground traffic. The results from Table 4 for scenarios 7 and 9, which are comprised, of multiple tests, are also visualized in Figure 5 and Figure 6

respectively. In these figures the horizontal axis represents the achieved throughput for each repetition of the corresponding scenario measured in Mbps, while the vertical axis represents both the value of jitter is milliseconds (ms) and the number of packets lost for every 100 packets transmitted (% percentage). They are useful in visualizing the effect of increasing network traffic on the quality that connections using the IP Premium service would receive.

Table 4. Results for UDP foreground traffic

Scenario	Achieved foreground bandwidth (Mbps)	Foreground jitter (ms)	Foreground packet loss (%)
1	1.54	7.813	0.0000
3	1.54	7.810	0.0000
5	1.53	7.808	0.0770
7	1.54	7.814	0.0000
	2.05	7.217	0.0000
	2.56	6.634	0.0000
	3.07	5.927	0.0000
	3.58	5.382	0.0000
	4.10	4.727	0.0000
	4.61	4.348	0.0000
	5.12	3.785	0.0110
	5.63	3.459	0.0000
	6.15	3.629	0.0000
	6.65	3.513	0.0088
	7.17	3.240	0.0082
	7.08	2.717	7.9000
9	1.53	7.799	0.1100
	2.04	7.317	0.1400
	2.56	6.321	0.2300
	3.07	5.579	0.1500
	3.57	5.035	0.3000
	4.08	4.639	0.3600
	4.59	4.368	1.2000
	5.09	4.103	0.5500
	5.60	3.315	0.5400
	6.12	3.340	0.4900
	6.62	3.346	0.6300
	7.11	3.320	0.7900
	7.01	3.221	8.8000

Scenarios 1, 3 and 5 prove that the foreground traffic receives superior quality regardless of the rate of background traffic (50, 80 or 120Mbps), which means that the network effectively isolates the important IPv6 traffic in order to provide the guaranteed service.

Figure 5 and Figure 6 clearly demonstrate the effectiveness of the policing mechanism that was applied at the input interface for the premium traffic. The configuration of the policing mechanism was done using the option of completely dropping excess packets (instead of simply treating them as best-effort traffic, which is also a viable solution depending on the requirements and the policies of each organization). Therefore, as soon as foreground traffic exceeded the allocated bandwidth (5% of the total available bandwidth or about 7.5 Mbps at the physical level), packet losses increase dramatically. Our choice for dropping exceeding packets instead of simply handling them as best-effort is more suitable for real-time applications, since for that type of application timing in the reception of the packets matters more than late delivery. In such case, late delivery of packets can be useless if the data should already have been presented to the user.

Another interesting observation is that the jitter for the foreground traffic steadily decreases as we are increasing the transmission rate. This observation is explained by taking into account the way jitter is calculated. A higher

transmission rate leads to packets arriving closer together at the destination, and therefore variations in the inter-arrival time are smaller (although in reality they can be steady when weighted against the inter-arrival times.

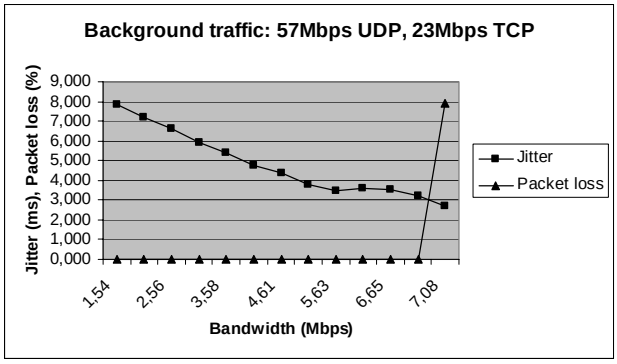

Figure 5. UDP foreground traffic with 80Mbps background traffic

Table 5 summarizes the results for TCP foreground traffic and the corresponding characteristics (jitter, packet loss) for the background traffic that was artificially created for each experiment. It is interesting to note that for scenarios 8 and 10, we were gradually increasing the TCP foreground transmission rate by adding ever more TCP streams. As soon as the transmission rate approached the allocated threshold, the transmission rate could no longer be increased, since the policing mechanism was dropping excessive packets and the TCP congestion avoidance mechanism was using this information to reduce the transmission rate.

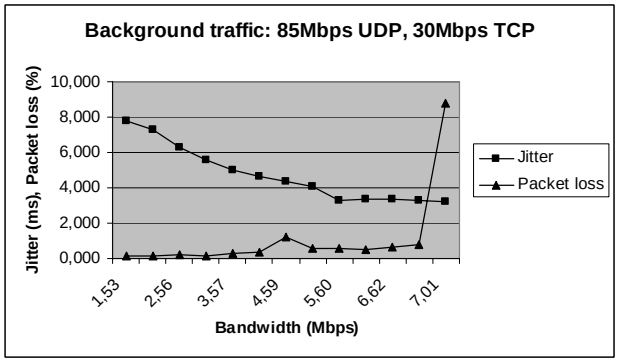

Figure 6. UDP foreground traffic with 120Mbps background traffic

Table 5. Results for TCP foreground traffic

Scenario	Achieved foreground bandwidth (Mbps)	UDP background jitter (ms)	UDP background packet loss (%)
2	1.95	0.439	0.1300
4	1.57	0.246	0.0160
6	1.52	0.230	0.1100
8	1.54	0.320	0.0490
	1.97	0.714	0.0450
	2.01	0.465	0.1600
	2.36	0.234	0.3100
	2.47	0.265	0.2200
	2.68	0.239	0.2000
	3.49	0.221	0.1500
	3.85	0.268	0.1400
	4.57	0.379	0.0980
	5.44	0.219	0.0670
	6.07	0.300	0.1200
	5.76	0.313	0.2900
10	1.54	0.195	1.5000
	1.59	0.209	1.1000
	4.62	0.148	1.1000
	4.90	0.156	1.5000
	5.09	0.206	1.4000
	5.29	0.170	1.5000
	5.03	0.149	1.3000

Since the last experiment in scenarios 8 and 10 are the ones that present the most interest (as the foreground traffic approaches the upper bound of the policing profile), Figure 7 through Figure 10 display the variation of the throughput rate from the beginning until the end of each experiment.
In particular, Figure 7 and Figure 8 present the corresponding throughput rates from the last two experiments for scenario 8, while Figure 9 and Figure 10 present the throughput rates from the last two experiments for scenario 10.

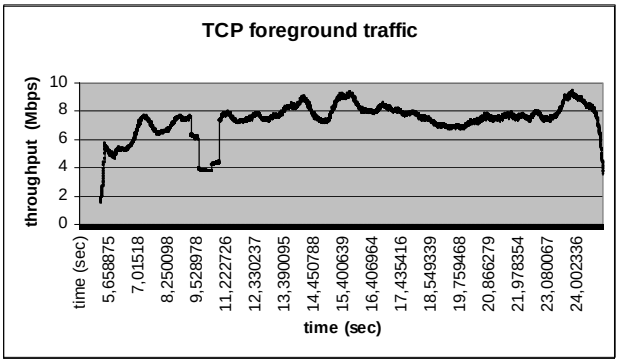

Figure 7. TCP foreground throughput for scenario 8, average throughput 6.07 Mbps

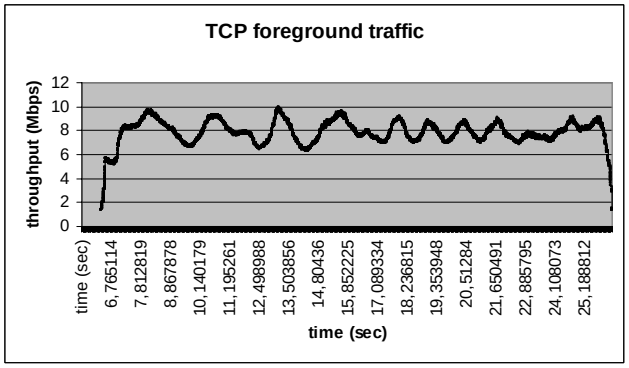

Figure 8. TCP foreground throughput for scenario 8, average throughput 5.76 Mbps

The most interesting observation when comparing Figure 7 and Figure 8 is that in the latter case the throughput seems to variate more regularly, while in Figure 7 the throughput is more stable. The reason is probably the fact that during the second experiment the artificially generated traffic was setup in order to transmit more traffic than the policing limit (through the usage of multiple TCP streams). The result was that the congestion control algorithm of TCP was backing off when it was sensing the packet losses due to the QoS policing mechanism.

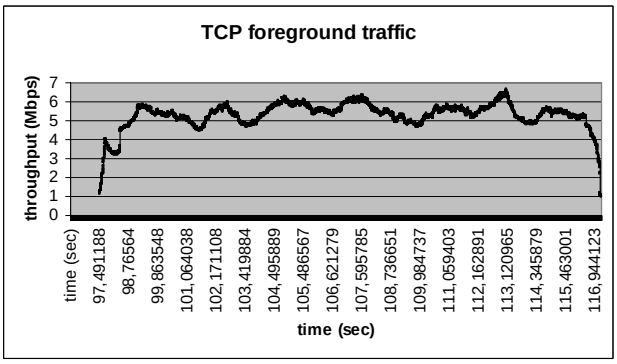

Figure 9. TCP foreground throughput for scenario 10, average throughput 5.29 Mbps

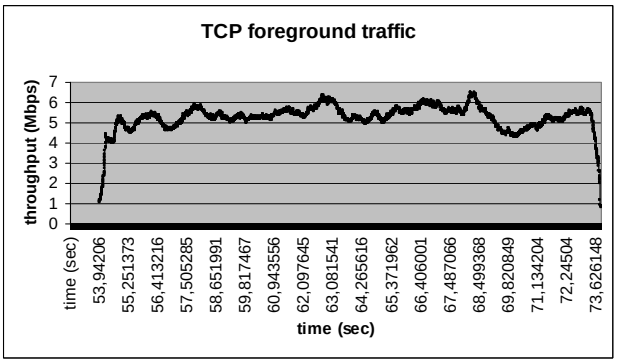

Figure 10. TCP foreground throughput for scenario 10, average throughput 5.03 Mbps

Although in the last two figures (Figure 9 and Figure 10) the background traffic was significantly increased (120 Mbps that almost saturated the core links) the figures demonstrate that the implemented QoS mechanism over IPv6 properly prioritizes the Premium traffic and therefore the sensitive to congestion TCP traffic is unaffected and displays the same behavior as before.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we presented our work within the 6NET project in order to evaluate real-time applications and the behaviour of a QoS service that was designed and implemented in the pan-European 6NET IPv6 network. The QoS mechanisms that were used were tested widely, both in a small-scale local testbed and in a large-scale international testbed, in order to make sure that they work properly and in

order to thoroughly investigate their performance. The QoS service was tested using actual traffic combined with simulated traffic (artificially generated but trying to simulate the important characteristics of real traffic as much as possible), while providing QoS to real-time application traffic and/or artificially generated traffic that represented real-time traffic. The main observation was that the network operated and provided the IP Premium QoS service, without any performance degradation or conflict with any other service. All the QoS mechanisms that were tested, showed that they are mature enough to be used for a production deployment.

Our future work includes investigating and evaluating later versions of the CISCO IOS for IPv6 QoS or different router platforms, since QoS mechanisms over IPv6 still remain a novel issue with unexplored aspects. Furthermore, the usage of the IPv6 flow label is expected in the near future and this will give the opportunity to extend the classic QoS service from an aggregated basis to a more per flow basis. Finally, the development, deployment and evaluation of the usefulness of a management tool for the whole IPv6 QoS service is in our future plans.

6 ACKNOWLEDGMENTS

The authors would like to thank the 6NET project partners for their valuable cooperation and contribution to the experiments. In particular, we would like to thank Lancaster University, Cisco Systems, Greek Research & Technology Network (GRNET), United Kingdom Education & Research Networking Association (UKERNA) and the 6NET project as a whole, which is funded by the IST program of the European Commission (IST Contract No: 2001-32603).

7 REFERENCES

[1] Deering S. and Hinden R., "Internet Protocol, Version 6 (IPv6) Specification" IETF RFC 2460, December 1998

[2] 6NET Project homepage, http://www.6net.org

[3] Blake S., Black D., Carlson M., Davies E., Wang Z., Weiss W., "An Architecture for Differentiated Services" IETF RFC 2475, December 1998

[4] Nichols K. and Carpenter B., "Definition of Differentiated Services Per Domain Behaviors and Rules for their Specification" IETF RFC 3086, April 2001

[5] Rajahalme J., Conta A., Carpenter B. and Deering S., "IPv6 Flow Label Specification" IETF RFC 3697, March 2004

[6] Geant's Network Home Page, http://www.dante.net

[7] Vegesna S., "*IP Quality of Service: the Complete Resource for Understanding and Deploying IP Quality of Service for Cisco Networks*", Cisco Press, 2001

[8] Bouras C., Gkamas A., Primpas D., Stamos K., "Quality of Service aspects in an IPv6 domain", 2004 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS' 04), San Jose, California, USA, July 25 - 29 2004, pp. 238 - 245

[9] Bouras C., Gkamas A., Primpas D., Stamos K., "Performance Evaluation of the Impact of Quality of Service mechanisms in an IPv6 network for IPv6-capable real time applications" Journal of Network and Systems Management, Kluwer Academic Publishers, Volume 12, Issue 4, December 2004, pp. 463 - 483

[10] Siris V. and Fotiadis G., "A test-bed investigation of QoS mechanisms for supporting SLAs in IPv6" 6NET Workshop II, Terena Networking Conference, Rhodes, Greece, June 7-10 2004

[11] IP Performance Metrics (IPPM) Working Group, http://www.ietf.org/html.charters/ippm-charter.html, IETF

[12] Iperf homepage, http://dast.nlanr.net/Projects/Iperf/

[13] Ethereal homepage, http://www.ethereal.com

[14] Cisco Systems, Inc. home page, http://www.cisco.com

[15] Jacobson V., Nichols K. and Poduri K., "An Expedited Forwarding PHB", RFC 2598, June 1999

Christos Bouras obtained his Diploma and PhD from the Computer Science and Engineering Department of Patras University (Greece). He is currently an Associate Professor in the above department. Also he is a scientific advisor of Research Unit 6 in Research Academic Computer Technology Institute (CTI), Patras, Greece. His research interests include Analysis of Performance of Networking and Computer Systems, Computer Networks and Protocols, Telematics and New Services, QoS and Pricing for Networks and Services, e-Learning Networked Virtual Environments and WWW Issues.

Dimitris Primpas obtained his Diploma and Master Degree from the Computer Engineering and Informatics Department of Patras University (Greece). He works in the Research Unit 6 of Research Academic Computer Technology Institute (CTI). His research interests include Computer Networks, Telematics, Distributed Systems and Quality of Service.

Kostas Stamos obtained his Diploma and Master Degree from the Computer Engineering and Informatics Department of Patras University. He has worked for the Networking Technologies Sector of Research Academic Computer Technology Institute (CTI) , Patras, Greece from the end of 1999 until December 2000. Since July 2001 he works with Research Unit 6 of CTI.