

Optimising Differentiated Services Strategies under IMS

David K. Osei-Aboagye¹ and Peter S. Excell^{2,*}

¹O2-Telefonica PLC, Slough, United Kingdom

david.osei007@gmail.com

²Wrexham Glyndŵr University, Wrexham, United Kingdom

p.excell@glyndwr.ac.uk

*Correspondence: p.excell@glyndwr.ac.uk

Received: 17th April 2018; Accepted: 20th May 2018; Published: 1st July 2018

Abstract: The evolving standards of mobile communications, the wide variety of services they offer and the rapid growth of the Internet have made a merger of the two network technologies inevitable. One of the most prominent platforms that has been developed to facilitate this is the IP Multimedia Subsystem (IMS) concept. Many mobile communications standards integrate IMS as the main core network architecture and Quality of Service (QoS) is the main concern for customer satisfaction. A major approach to optimisation of QoS is the Differentiated Services scheme, and a simulation study of implementations of this is presented. The study covered an IMS core network architecture modelled with discrete-event network simulator software, with a Differentiated Services QoS scheme run over it with differing bearer traffic scenarios. Implications for core network architectures are discussed.

Keywords: *IMS; Mobile communications; Quality of Service; Differentiated Services; Network simulation*

Abbreviations

CBS	Committed Burst Size	PIR	Peak Information Rate
CIR	Committed Information Rate	QoS	Quality of Service
EBS	Excess Burst Size	RED	Random Early Detection
FTP	File Transfer Protocol	SIP	Session Initiation Protocol
IMS	IP Multimedia Subsystem	srTCM	Single Rate 3-Colour Marker Policer
MN	Mobile Node	trTCM	Two Rate 3-Colour Marker Policer
NAM	Network Animator	TSW2CM	Time Sliding Window with 2-Colour Marking Policer
NS	Network Simulator	TSW3CM:	ditto, 3-colour
PBS	Peak Burst Size	TCP	Transmission Control Protocol
PHB	Per-Hop Behaviour	UDP	User Datagram Protocol

1. Introduction

Mobile communications has evolved towards an IP (Internet Protocol) core network. This has led to much research into the IP Multimedia Subsystem (IMS) concept, which is an IP-based network that allows integration of voice and multimedia, as well as permitting new environments with new purposes [1]. Quoting [1]: “IMS carries signalling and bearer traffic over the IP layer, functioning as an intelligent ‘routing engine’ that matches a user profile with an appropriate call handling server

and switches the call control over to the designated handler.” Most of the applications expected under IMS have different Quality of Service (QoS) requirements: to handle these with the same QoS preferences as all other packet data routed through the Internet would result in unsatisfactory performance and customer dissatisfaction.

Most of the services offered by IMS were already being offered by existing circuit and packet-switched mobile networks, but IMS aimed at providing these services with a guaranteed QoS as well as guaranteed bandwidth efficiency. Thus certain services like real-time multimedia applications can be given preference in terms of bandwidth since they require more bandwidth than non-real-time services. Circuit and packet-switched multimedia services give no preference to real-time or non-real-time sessions.

As a tool to investigate IMS performance under QoS constraints, a simulation of the IMS core network architecture was developed using discrete-event network simulator software, the network architecture being designed to carry both session signalling traffic as well as bearer traffic. The model was then used to implement a Differentiated Services QoS scheme on top of the designed IMS architecture and a performance evaluation was made for different bearer traffic scenarios.

2. Software Configuration for Simulation

The IMS core network is managed mainly by SIP signalling for session set-up and tear down. Due to this, the first objective was to create a SIP signalling architecture for the network topology. The next objective after creating the signalling architecture was to create IP traffic to run over the signalling architecture to help facilitate the implementation of the Differentiated Services (DiffServ) Quality of Service scheme.

Network Simulator allinone version 2.27 was the only NS-2 version with SIP functionality built into its modules but it did not support multi-interface architecture. The later NS allinone-2.31 had an extension for multi-interface layers but no SIP support. To enable the simulation the NS allinone-2.27 SIP signalling module patch was thus manually altered in order to make it function in NS allinone-2.31.

3. Network Design and Simulation

The network topology chosen to be simulated for the IMS core network’s SIP signalling consisted of five routers, two proxies and four Mobile Nodes (MN). The traffic over the air interface to the Mobile Node was ignored, hence its propagation delays were not taken into account.

Figure 1 shows the network architecture to be simulated. P1 represents proxy server 1 which is the proxy for “David.com” and P2 represents proxy server 2 which is the proxy for “Demola.com”. R0 represents the core router while R1, R2, R3 and R4 are the other outer routers within the network. MN1 represents Mobile Node 1 “Peter”. MN 2 represents Mobile Node 2 “John”. MN 3 represents Mobile Node 3 “Victoria” and MN 4 represents Mobile Node 4 “Wendy”.

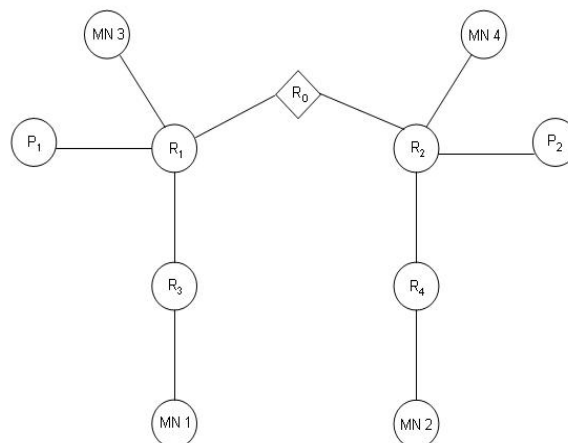


Figure 1. Diagram of the network architecture to be simulated.

In the design, “Peter” would connect to proxy David.com by sending a SIP signalling registration request and an ACK (acknowledgement) response would be sent back, confirming registration procedure. “John” would also do a similar SIP registration request with proxy Demola.com. After the registration procedure, “Peter” would then send a SIP signalling invitation request to “John” which accepts the invitation to initiate the session. “Victoria” and “Wendy” would also go through a similar registration and invitation procedure.

3.1 First Design

The design was simulated by first creating an instance for the simulator object. This is achieved in NS-2 by executing the command “set ns_ [new Simulator]” [8,9]. The next objective was setting the addressing format, chosen as hierarchical and set at a domain number of 3. This was achieved by executing the command “ns set-address-format hierarchical” [7]. The addressing parameters were set at the domain number of 3 with the NS command “set domain_num_ 3”. In order to have output files with data on the simulation (trace files) and visualisation files (NAM files), a trace file and NAM file was created [8,9]. The commands were “set tf [open_sip_test.tr w]” and “set nf [open out_sip_senario.nam w].”

After the trace file and NAM file were created, the routers were set and the addressing parameters and domain numbers were set. The proxies were also given addressing parameters and domain numbering. The positions of all the Mobile Nodes, proxies and routers were defined and set. The signalling links between the routers and the proxy were also set: to define a bi-directional link a “duplex-link” command is used. The signalling links between the Mobile Nodes were also defined and set at 1MB capacity and 10 ms of propagation delay, as was the case for the routers and proxies.

The next design goal was labelling the proxies and mobile nodes and then setting the server addresses as well as attaching agents to them. The procedure of attaching agents is used to define a signalling flow between two points in the topology. A finish procedure was defined and configuration of “John” and “Wendy’s” terminals was done for complex responses. Finally, the scheduler for the events of the simulation was set up, used to set the beginning and end of a session or application being run on the simulation platform.

3.2 Second Design

The second design goal was to design an IP traffic flow over the SIP signalling architecture. A Transmission Control Protocol (TCP) connection was set up between Mobile Node “Peter” and Mobile Node “John”. This was achieved by setting up a TCP agent as well as defining the Class of Service to be implemented for the traffic flow. A sink agent was also set up to indicate where the TCP connection was to terminate. After the TCP connection was set up, a File Transfer Protocol (FTP) application was set up to run over the TCP connection. This was done because “Peter” would be sending a file over TCP to “John”. A User Datagram Protocol (UDP) connection was then set up between Mobile Node “Victoria” and Mobile Node “Wendy”. Again a Null agent was set up to show the termination point for the UDP connection. Since “Victoria” would be making a Voice over Internet Protocol (VoIP) call to “Wendy”, a UDP connection was used to avoid TCP acknowledgments and a constant bit rate was run over the UDP connection.

3.3 Third Design

The final design goal is an implementation of a Quality of Service scheme on the IMS architecture. The Differentiated Services module, originally implemented by the Advance IP Network group in Nortel Networks [8,10], was chosen as the Quality of Service scheme for implementation. The Differentiated Services architecture design uses different Random Early Detection (RED) parameter configurations for virtual queues, causing packets from one virtual queue to be dropped more frequently than packets from another [8]; thus code points are given to different classes of traffic.

The Network Simulator module developed by Nortel has three major components [8]:

- Policy: specified by the network administrator defining the level of service a class of traffic should receive in the network
- Edge router: marking the packets with a code point according to the policy specified.
- Core router: examines the packets' code point marking and forwards them accordingly.

A policy defines how traffic is aggregated within the network. The Network Simulator has six different policy models [8]:

- Time Sliding Window with 2 Colour Marking (TSW2CM) Policer: uses a CIR and two drop precedence. The lower precedence is used probabilistically when the CIR is exceeded.
- Time Sliding Window with 3 Colour Marking (TSW3CM) Policer: uses a CIR, PIR, and three drop precedence. The medium drop precedence is used probabilistically when the CIR is exceeded and the lowest drop precedence is used probabilistically when the PIR is exceeded.
- Token Bucket Policer: uses a CIR and a CBS and two drop precedence. An arriving packet is marked with the lower precedence if and only if it is larger than the token bucket.
- Single Rate Three Colour Marker (srTCM) Policer: uses a CIR, CBS, and an EBS to choose from three drop precedence.
- Two Rate Three Colour Marker (trTCM) Policer: uses a CIR, CBS, PIR, and a PBS to choose from three drop precedence.
- Null Policer: does not downgrade any packets

CIR: Committed Information Rate; PIR: Peak Information Rate; CBS: Committed Burst Size; EBS: Excess Burst Size; PBS: Peak Burst Size [8,9].

The Nortel Differentiated Services network topology design for NS-2 uses a bottleneck architecture (Fig. 2).

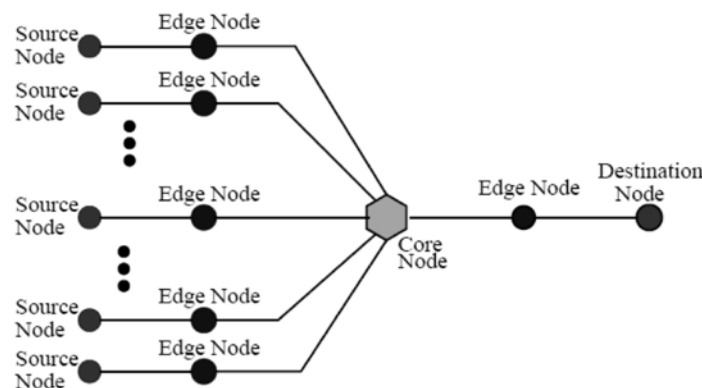


Figure 2. Bottleneck network topology. [8]

In the bottleneck topology each source node is connected to a corresponding edge router and then to the core router and via another edge router to the destination. The bottleneck architecture implements a multiple source to single destination routing. Since the IMS architecture core network design has a multiple source and multiple destination, a double bottleneck network topology would be used to achieve the design goal. The Differentiated Services policy chosen for the design was a token bucket policy. Figure 3 shows the edge and core routers within the IMS core network architecture, with two possible paths for classes of traffic.

The token policy design was achieved by setting and defining the CIR1, CBS1, CIR0, CBS0, rate0 and rate1. The rate0 and rate1 command is used to define the rate of packet flow within the defined class of traffic. The link between the edge routers and the core router were re-defined with the Random Early Detection queuing policy. The link between the proxies and the Mobile Nodes (terminals) remained the same except for a change in the data capacity.

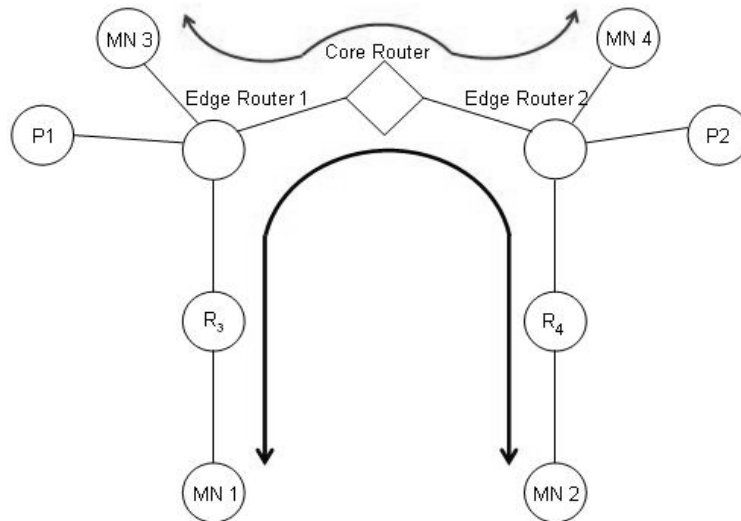


Figure 3. Differentiated Services implementation

The maximum packet size was set to 1Mb. The queuing parameters were also set for all routes between edge routers and the core router. A double policy entry was introduced for the routing path thus double source – double destination as compared to the original multiple source – single destination implementation of the bottleneck architecture designed by Nortel Networks.

The Per-Hop Behaviour (PHB) entries were defined and the queues configured for the physical and virtual queues. After the configuration was done for the different classes of traffic, the scheduler was again reconfigured to enforce the events introduced due to the Differentiated Services implementation.

4. Results and Analysis

4.1 First Design

After running the code for the SIP signalling architecture on NS-2, the results and visualisation files generated by Network AniMator (Fig. 4) showed that registration and session connection performed reliably: this is illustrated in Fig. 5, where the black arrow between nodes 1 and 0 represents an example of a SIP registration and signalling packet to the proxies. After the SIP signal design was run, the results obtained showed a successful registration of the Mobile Nodes and initiation as well as termination of sessions: this indicates that the SIP signalling architecture designed was successful.

```
[root@localhost codes]# ns sip_scenario.tcl
Default value for cluster_num set to 4

Default value for nodes_num set to 5

Node 1.1.2: peter@David.com successfully registered contact peter@1.1.2
Node 1.2.1: victoria@David.com successfully registered contact victoria@1.2.1
Node 2.1.2: john@Demola.com successfully registered contact john@2.1.2
Node 2.2.1: wendy@Demola.com successfully registered contact wendy@2.2.1
Node 1.1.2: initiating session 5 from peter@David.com to john@Demola.com
Waiting 1.00136 seconds before picking up the phone (CallID=5)
TODO: fill SIPUA::processPrackResponse()
Node 1.1.2: session 5 from peter@David.com to john@Demola.com succeeded
Node 1.2.1: initiating session 6 from victoria@David.com to wendy@Demola.com
Node 1.2.1: session 6 from victoria@David.com to wendy@Demola.com succeeded
Node 2.1.2: ending session 5 from john@Demola.com to peter@David.com
Node 2.1.2: session 5 from john@Demola.com to peter@David.com terminated
Node 1.2.1: ending session 6 from victoria@David.com to wendy@Demola.com
Node 1.2.1: session 6 from victoria@David.com to wendy@Demola.com terminated
NS-simulation finished!
```

Figure 4. Textbox showing Mobile Node registration

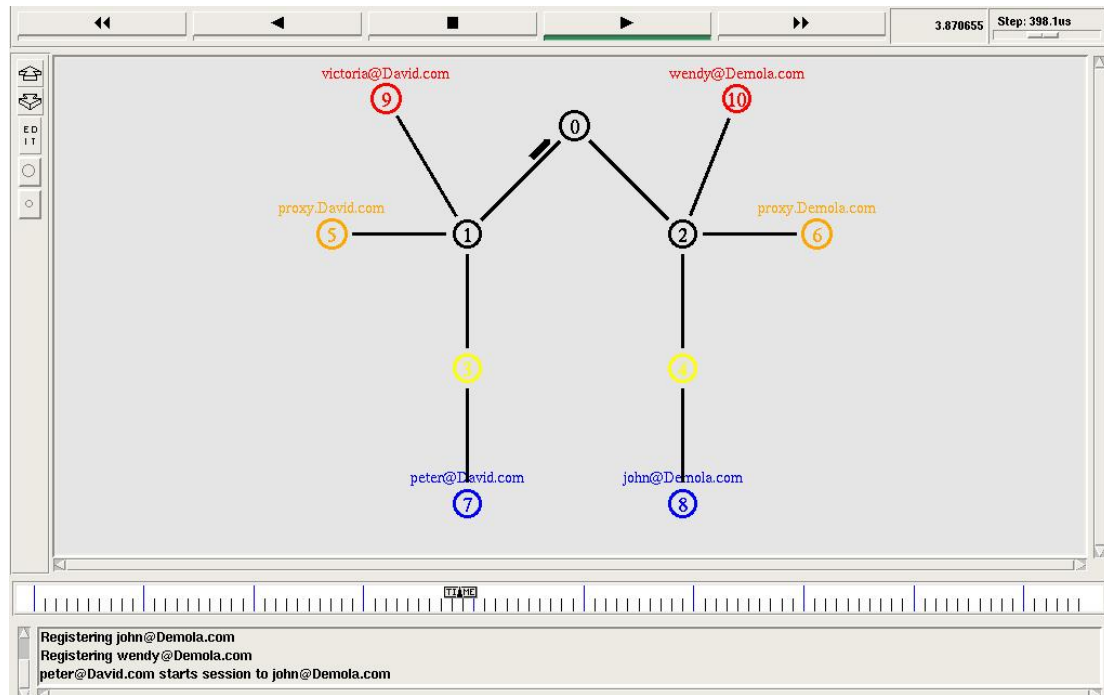


Figure 5. Mobile Node positions in NAM (Network Animator).

4.2 Second Design

The second design goal involved the implementation of IP traffic on top of the SIP signalling architecture. After the SIP signalling code was edited to introduce IP traffic and then run, the following results were obtained.

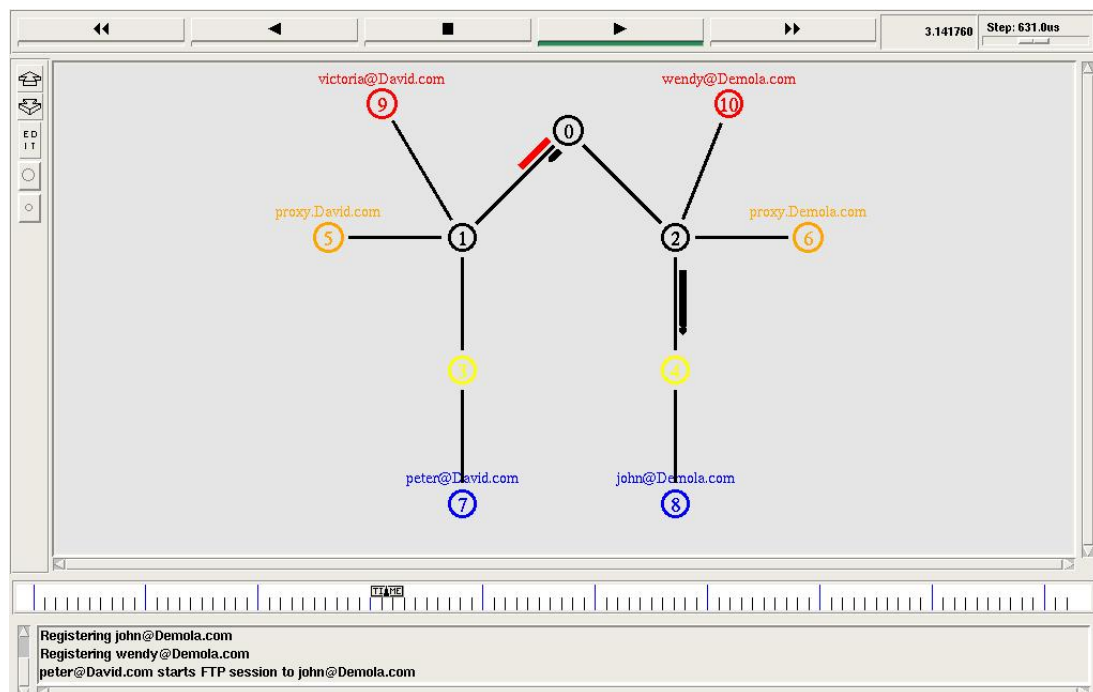


Figure 6. Diagram showing UDP traffic

Figures 6 to 9 show IP traffic initiation and termination after the SIP signalling between Mobile Nodes and Proxies. In Fig. 6, red arrows are UDP packets for voice calls: UDP is used since retransmissions are not needed when having a conversation and UDP is faster and can make use of larger packets than TCP. Also a wide bandwidth was considered to be available for this purpose. In

the complete animation, red arrows are travelling from 9 to 10 and vice versa, to represent a two-way communication. The black arrows go to the respective proxies for registration and to the opposite proxies for signalling to initiate sessions. In the model, UDP packets were always given preference over all other types of traffic in all the policies that were chosen for the project, due to their low-latency requirement.

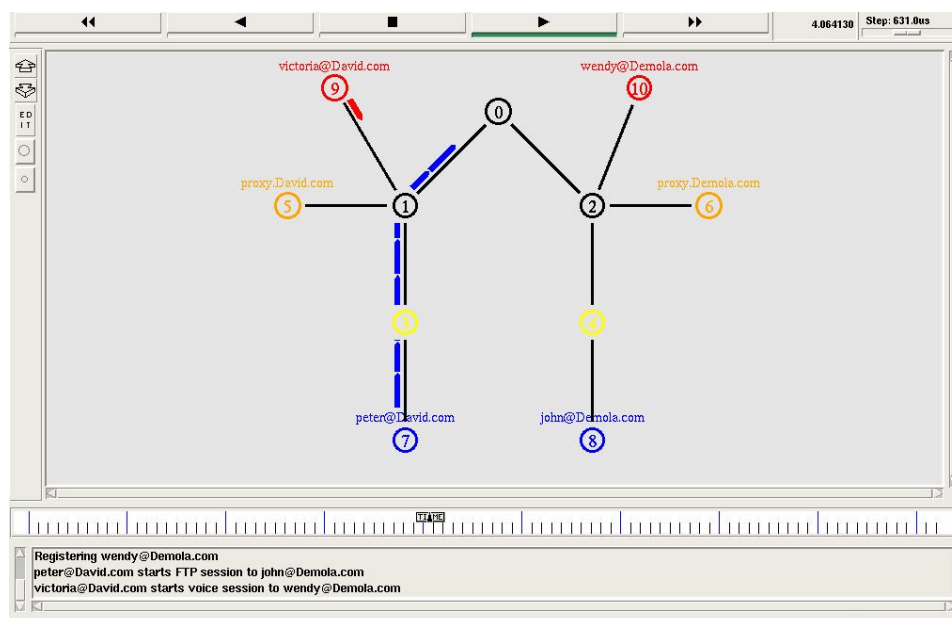


Figure 7. Initial FTP transfer.

In Figures 7 and 8, the blue arrows are FTP packets from 'Peter' (Node 7) to 'John' (Node 8): this is a one-way traffic but there is a return traffic of acknowledgements (small blue dashes - visible in Fig. 8); if these are not detected a retransmission is initiated.

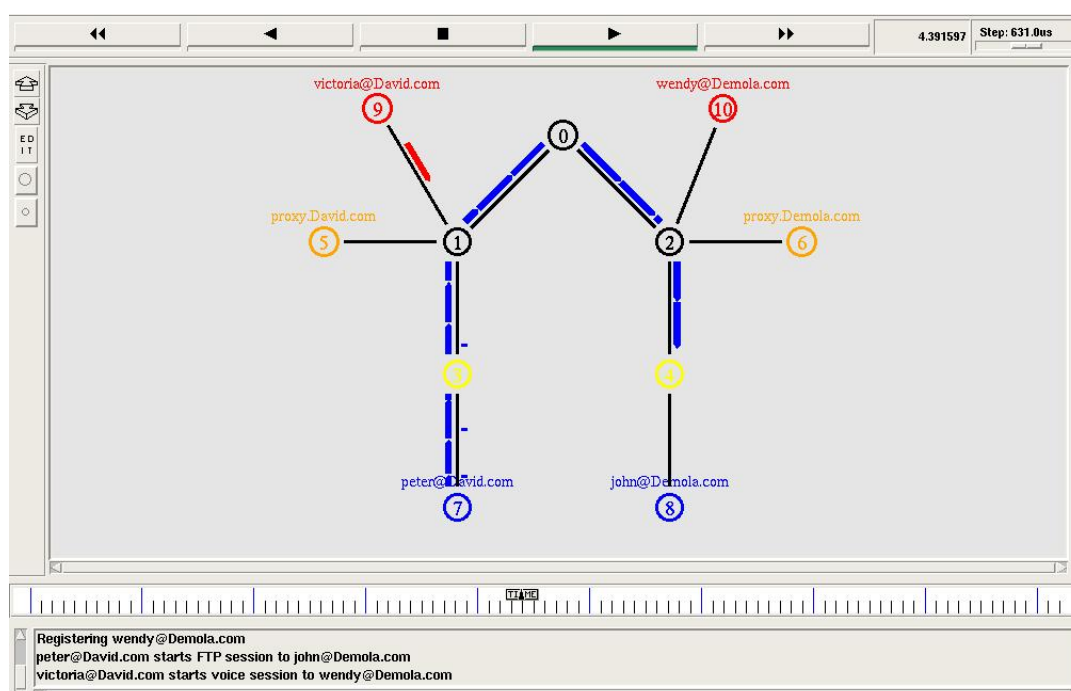


Figure 8. Diagram showing FTP over TCP with acknowledgement.

When IP traffic was introduced into the SIP signalling architecture the results obtained showed a successful SIP signalling registration. The IP traffic architecture introduced into the network ran concurrently without any errors or conflicts. The IP traffic session set for both communication paths were observed to flow from end to end successfully. However, if any FTP packets are lost in

transmission they are retransmitted, thus resulting in congestion if the retransmission rate is high. Since voice (UDP) packets were given priority over all other packets, UDP packets will go through irrespective of what policy is being used to manage congestion. This situation is illustrated in Fig. 9.

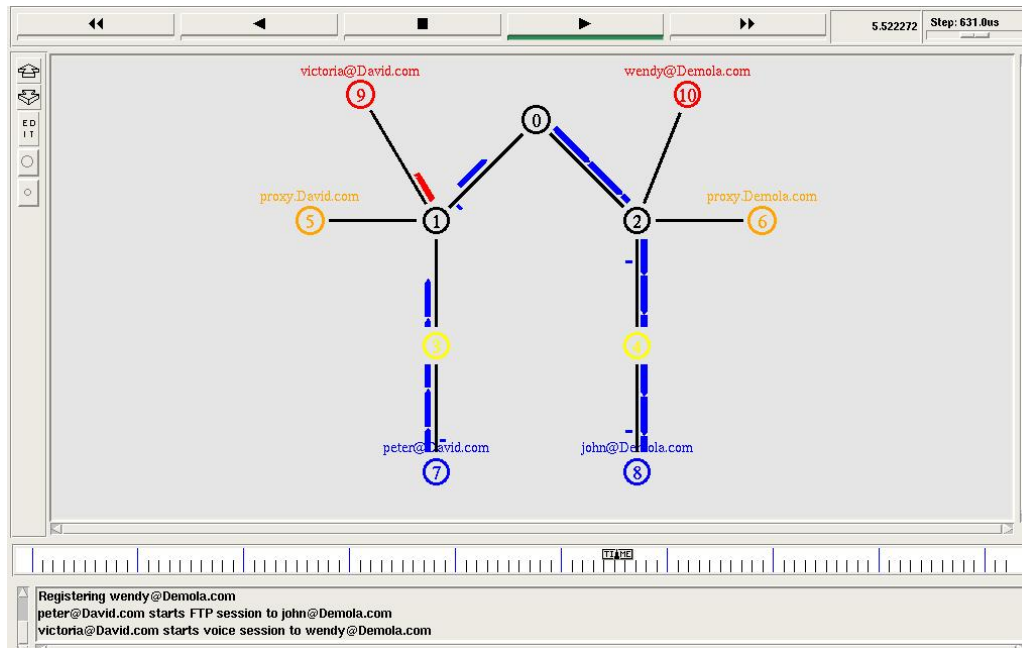


Figure 9. Voice (UDP) session running concurrently with FTP in the topology.

4.3 Third Design

The Differentiated Services QoS scheme was implemented after the successful implementation of different classes of IP traffic over the IP Multimedia Subsystem core network. However, the results obtained after the final code was run displayed an error: "The policy object does not exist" (Fig. 10). This was because the introduction of a double policy for a single edge router – core router – edge router path caused a conflict.

```
[root@localhost ns-allinone-2.31]# ns sip_scenario_3.tcl
Default value for cluster_num set to 4

Default value for nodes_num set to 5

Node 1.1.2: peter@David.com successfully registered contact peter@1.1.2
Node 1.2.1: victoria@David.com successfully registered contact victoria@1.2.1
Node 2.1.2: john@Demola.com successfully registered contact john@2.1.2
Node 2.2.1: wendy@Demola.com successfully registered contact wendy@2.2.1
Node 1.1.2: initiating session 5 from peter@David.com to john@Demola.com
ERROR: No Policy Table entry found for Source 4194305-Destination 8388609.
Policy Table(2):
Flow (7 to 8): Token Bucket policer, initial code point 10, CIR 1000000.0 bps, CBS 3000.0 bytes.
Flow (9 to 10): Token Bucket policer, initial code point 10, CIR 1000000.0 bps, CBS 10000.0 bytes.

The policy object doesn't exist, ERROR!!!
[root@localhost ns-allinone-2.31]#
```

Figure 20. Textbox showing the error encountered after running the Code.

The six different categories of policies developed by Nortel for Differentiated Services were separately implemented on the designed architecture but they all gave the same conflict error. The results obtained from the introduction of the six different categories indicates that the conflict observed was due to the fact that the bottleneck architecture works on a 'multiple-sources – single-destination' node routing policy. In the Differentiated Services design implemented, a double bottleneck architecture was improvised to avoid any possible conflicts but the desired result was not obtained.

6. Conclusions

The implementation of an all-IP mobile communications network is a major requirement for the latest mobile networks. This means that former circuit-switched functionalities are replaced by a packet-switched functionality which works as efficiently for voice data as in a circuit-switched network. This need has driven the development of the IP Multimedia Subsystem (IMS).

The goal of the present work was to develop a model of the core network of IMS on Network Simulator 2 (NS-2), implementing a suitable Quality of Service scheme, and then to evaluate performance of the scheme in its handling of different classes of IP traffic. In coming up with the design, bearer traffic over the air interface was ignored: the core network considered was the wired network consisting of SIP proxy servers and routers.

The design also presented the implementation of differentiated classes of service of IP traffic within the network, focussed on different IP traffic classes routed through one edge router – core router – edge router path. The simulation techniques were used to perform an evaluation of Nortel Networks' Differentiated Services Architecture on the designed IMS core network. The simulation results demonstrated a fully functional IMS core network and investigated performance of the Differentiated Services design for IP traffic from single or multiple sources to multiple destinations when using one routing path: this constituted a bottleneck and caused a failure.

It may be concluded from this study that, whichever policy is chosen, if preference is given to UDP over all other packets, the Nortel bottleneck design defines its policies to the source edge router and not the destination edge router. In a case where there are several destination edge routers there is difficulty resulting from the range of destinations for which the packets are intended. Hence the design of a 'butterfly' architecture was devised which defines policies to the source and destination edge routers as well as the core routers to help facilitate the use of multiple source and destination routers and send traffic to and from any of the routers, including two way communication.

The implementation of a double Differentiated Services policy on the first edge router of a bottleneck topology architecture resulted in a failure of traffic flow due to a lack of a defined policy on packet distribution for the second edge router. As a more appropriate architecture, the "butterfly" architecture can be designed by editing the NS-2 source code for differentiated services such that policies can be defined for both edge router 1 and edge router 2: an outline diagram is given in Figure 11. A successful implementation of this would facilitate the design of multiple source – multiple destination routing. This implies that the butterfly Differentiated Services design could handle multiple sources – multiple destinations, single source – multiple destinations, single source – single destination and multiple sources – single destination bearer traffic paths reliably.

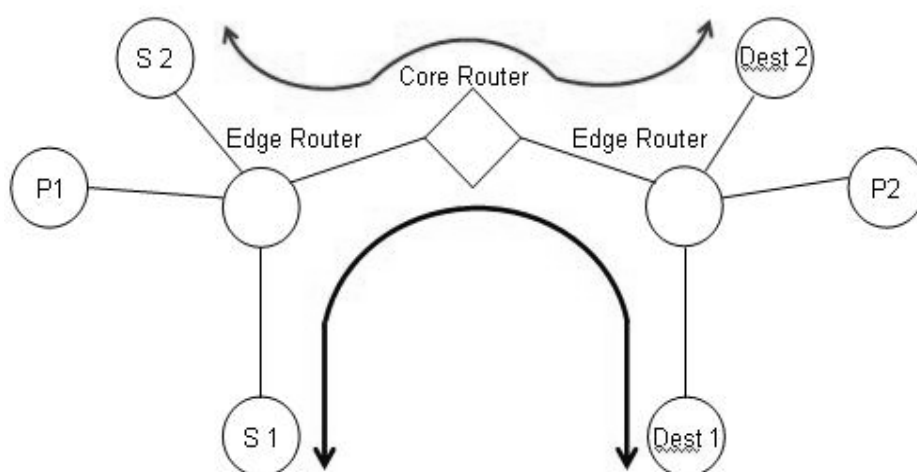


Figure 31. Proposed butterfly network topology for Differentiated Services.

S1 = Source 1; S2 = Source 2; Dest 1 = Destination 1; Dest 2 = Destination 2

The paths with arrowheads show key traffic flow paths.

Looking at the diagram in Figure 11, UDP packets give bigger packet sizes with no QoS guarantee while TCP gives a guarantee due to acknowledgements, but sizes are restricted. Hence to give a guarantee for UDP packets a tunnel is created once the SIP signalling establishes that a UDP session is going to be in place. If we label the side of the core router facing the source nodes as R_r and that of the destination nodes as T_r , then what happens is that, for packets from any source to R_r through whatever number of edge routers along the way, multiple tunnels are established for UDP sessions and, if there are some TCP packets along the way as well, preference is given to the UDP packets. When there is nothing going through the tunnel it is dropped for other packets to go through or another tunnel is created within the tunnel for the other packets with preference still given to the UDP packets. However, on the T_r side of the network the tunnel is dropped and data is sent to other destinations: as soon as a UDP packet comes through, a tunnel is created again and the TCP packets are dropped hence causing a large number of retransmissions until the tunnel is dropped again. The only time both UDP and TCP are sent through the same tunnel is if they are all going to the same recipient. The butterfly architecture mirrors the R_r side of the network thus creating tunnels within the tunnel from the core to the edge routers so that the bandwidth is not fully utilized by one connection but preference is just given to UDP over the rest.

It is suggested that this insight will assist with future research in IP-Multicast and broadcast scenarios within the IP Multimedia Subsystem network.

References

- [1] Motorola Inc., Chicago, "Motorola IP Multimedia Subsystem", August 2007.
Available: <http://www.motorola.com/networkoperators/pdfs/IMS-WhitePaper.pdf>.
- [2] Polikselka M., Mayer G., Khartbill H. and Niemi A, The IMS IP Multimedia Concepts and Services in the Mobile Domain, John Wiley & Sons, 2004.
- [3] Camarillo, G. and Garcia-Martin, M.A. The 3G IP Multimedia Subsystem (IMS), John Wiley & Sons, 2004.
- [4] Johnston, A.B. SIP: Understanding the Session Initiation Protocol, 2nd Edition, Artech House, 2004.
- [5] Network Simulator NS-2, "The Network Simulator ns-2", August 2007. Available: <http://www.isi.edu/nsnam/ns/>.
- [6] University of Padua, Italy, "NS-MIRACLE: Multi-InterfAce Cross-Layer Extension library for the Network Simulator", August 2007. Available: <http://www.dei.unipd.it/~baldo/nsmiracle-dei80211mr-howto.html>.
- [7] University of Oporto, Portugal, "NS-2 Network Simulator Extensions", SIP Module, August 2007. Available: <http://www.ncc.up.pt/~rprior/ns/index-en.html>.
- [8] Altman E. and Jimenez T. "NS Simulator For Beginners", Lecture Notes, Sept 2003-2004 Univ. de Los Andes, Merida, Venezuela, and ESSI, Sophia Antipolis, France. <http://www-sop.inria.fr/maestro/personnel/Eitan.Altman/COURS-NS/n3.pdf>.
- [9] The VINT Project, "The NS Manual", <http://www.isi.edu/nsnam/ns/doc/index.html>.
- [10] Noureddine W. and Tobagi F. "Improving the Performance of Interactive TCP Applications Using Service Differentiation", IEEE Proc., June 1998.



© 2018 by the author(s). Published by Annals of Emerging Technologies in Computing (AETiC), under the terms and conditions of the Creative Commons Attribution (CC BY) license which can be accessed at <http://creativecommons.org/licenses/by/4.0/>.