International Journal of Control and Automation
Vol. 9, No. 5 (2016), pp. 185-198
http://dx.doi.org/10.14257/ijca.2016.9.5.18

A Policy Based Management of a Smart Adaptive QoS for the Dynamic and Multipoint Virtual Private Network

Ayoub Bahnasse, Najib El Kamoun

Dept. Physics, Lab. STIC, Faculty of Sciences,
University Chouaïb Doukkali El Jadida, Morocco
bahnasse.a@ucd.ac.ma, elkamoun@ucd.ac.ma

Abstract

The IP Virtual Private Network technology is increasingly being used due to its efficiency in terms of cost, its availability and its high security level, companies deploy multiple secure tunnels and send through them different traffics, in order to guaranty a reliable level of communication and the availability of resources, many qualities of service's policies must be defined. According to our research, most of works treat only the problematic of the Quality of Service management in a site to site Virtual Private Network, this was an incentive for us to: design and implement a new model of a smart adaptive quality of service management for the dynamic and multipoint Virtual Private Network, and create a new friendly-user web application to facilitate the quality of service management for such complex networks.

This paper discusses a new model for the policy-based management of a smart adaptive quality of service for the Dynamic and Multipoint Virtual Private Network using a new WEB interface.

Keywords: *DMVPN, VPN, QoS, Policy-based, adaptive, Smart*

1. Introduction

Virtual Private Network known as VPN, is a technology that allows extending the security zone of the internal network through a public or shared infrastructure like the Internet. VPNs are increasingly used due to their efficiency in terms of cost, high availability and their very high levels of security compared to solutions such as Frame Relay, ATM [1, 2].

1.1. Dynamic Multipoint Virtual Private Network

Medium and large companies currently tend toward using the Dynamic Multipoint VPN network (DMVPN) [3] as a solution, due to its ability to deploy a very high number of secure tunnels with a minimum of cost and configuration, the DMVPN technology is based on NHRP, mGRE, IPsec and routing protocols:

- The NHRP[4] protocol allows to dynamically determine the public IP address of the destination "SPOKE", using the cache of the master router of the cloud "HUB";
- The mGRE protocol allows the interface to support multiple destination, Using mGRE as using Single GRE tunnels , provides support for IP and non-IP traffic to pass through a tunnel, the traffic can be Unicast, Multicast or Broadcast;
- The IPsec protocol [5] is an extending of Encapsulation Security Payload [6] protocol and Authentication Header Protocol[7], the last two protocols provide integrity and authentication of data contrariwise, the first provides more privacy exchanges. IPsec operates in two modes; the tunnel mode and transport mode, tunnel mode replaces the original IP header and encapsulates the entire packet, the

second does not modify the initial header, it is inserted between the network layer and the transport layer of the OSI model. But with the presence of the NAT, integrity issues causes packets rejection[8];

- internal routing protocols ensure optimal routing of data between the routers of the same cloud[9, 10], many studies has been made to study the impact of these protocols on DMVN networks[11] or on Non Broadcast Multi Access networks in general[12, 13].

Figure 1 shows an example of a DMVPN architecture.

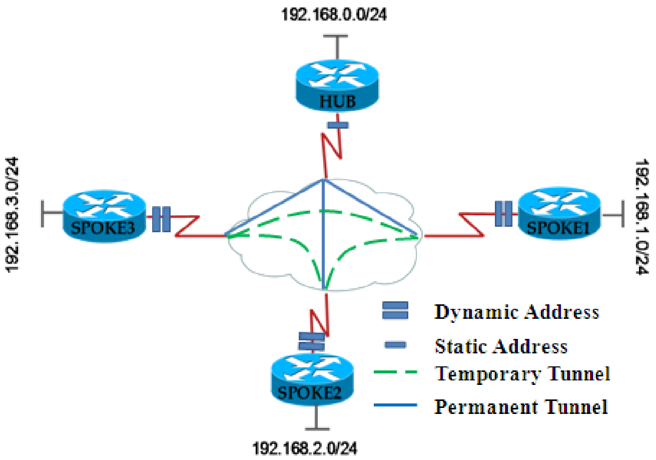


Figure 1. An Example of DMVPN Architecture

1.2. Differentiated Service

Differentiated Service "DiffServ" [14] is a mechanism to classify data into Behavior Aggregate "BA" based on the field DS "DiffServ" of the IPv4 packet. Classification, control operations and flow marking are carried out by "Edge Routers", intermediary routers treat packets only depending on DS field’s value according to a specific behavior known as Per Hop Behavior "PHB" ; two PHB behavior were defined;

- Expedited Forwarding "EF" [15]: has as purpose to guarantee a bandwidth with a low loss rate, delay and jitter,
- Assured Forwarding "AF" [16] this family is divided into four classes providing a bandwidth and a minimum delay, each class contains three priority levels (Drop Precedence).

1.3. Policy-Based Smart Adaptive QoS

Policy-based Smart Adaptive QoS (BP SAQOS) [24], provides a separation and a distribution of: data plane, control plane and management plane, in order to achieve a fast and effective treatment, as well as automatically and dynamically reallocate bandwidth, reduce the loss rate, minimize the delay and jitter in a converged network, this model has been incorporated in our framework for several reasons:

- BP SAQOS is a scalable and flexible model, because it uses distributed data plane, control plane and management plane,
- BP SAQOS proved its efficiency and flexibility compared to the CBWFQ and FIFO solution.
- PB SAQOS support networks converge.

1.4. Related Works

The management of the quality of service in VPN networks, is an active area of research until today, several contributions were made in the centralized management of the quality of service in VPNs [17-20], but most of works treat only the static VPN (site to site), Cisco's approach for Dynamic Multipoint VPN networks [21] operates the HUB router as a QoS policy server, i.e., the process of creation of the QoS policies is made at the HUB side, policies are then mapped to a specific NHRP group, this group must be configured on each Spoke router on the GRE or mGRE interface, The name of NHRP group is communicated to the HUB in each of the periodic NHRP updates, after receiving request, the HUB checks the policy associated to NHRP group and delivers it to SPOKE. This solution was implemented for specific versions of routers and IOS.

The proposed model provides a dynamic and automatic detection of the architecture’s equipment, and their capacity of the links, also it detects active flows and allows the customization of the company’s private applications. Quality of Service policies are defined graphically through a user-friendly interface and stored on the database server for further modifications. The framework uses BP SAQOS model to guaranty a low loss-rate, latency and jitter.

This paper is organized as follows: in Section 2 we discuss the framework Policy-Based Smart Adaptive QoS for DMVPN “PB-SAQOS for DMVPN” and define its modules, section 3 will be reserved for an overview and guided tour of the tool “WEB SAQOS for DMVPN”, in section 4 we will evaluate the model and conclude in Section 5.

2. The PB-SAQOS for DMVPN Framework

The PB-SAQOS for DMVPN framework [Figure 2] is composed of three different Agents: User Agent, Server Agent, Decision and Delivery Agent.

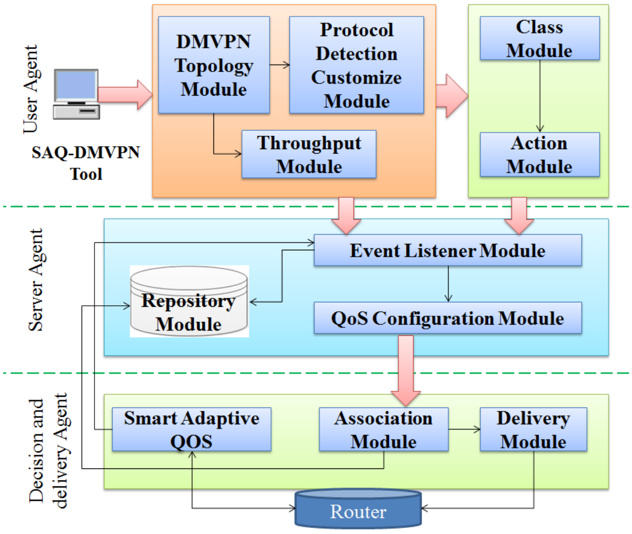


Figure 2. The PB-SAQOS for DMVPN Model

2.1. User Agent

Provides an oriented WEB interface for the automatic and dynamic detection of DMVPN equipment, and necessary information for the proper implementation of the QoS policy such as: active traffic and Uplink/Downlink throughput of each detected equipment. User Agent is composed of five modules:

2.2. Server Agent

Allows converting the data of the "User Agent" to a quality of service policies adapted to destination equipment, this agent is composed essentially by three modules:

Event Listener Module: This module detects changes occurred on the User Agent modules, and updates the entries of the database "Repository Module", this module allows defining the process: create, modify or delete, that the “QoS Configuration Module” must execute.

QoS Configuration Module: This module automates generation of final policies by converting the data of the "User Agent" in command lines adapted to versions of detected equipment by the "DMVPN Topology” Module, the generation of policies is performed only when a create, modify or delete notification is received by the "Event Listener Module".

Repository Module: module is a database containing User Agent data’s and triggered events of the Event Listener Module.

2.3. Decision and Delivery Agent

This agent has two main roles, firstly, to ensure mapping and secure delivering of policies for specific equipment automatically, secondly, to readapt automatically and dynamically delivered policies using PB SAQOS model. This agent is composed of three modules:

Association Module: This module performs a mapping between the policy already generated by the "QoS Configuration Module" and detected equipment; all associations will be saved in the "Repository Module".

Delivery Module: This module ensures a secure delivery of QoS policies created on QoS Configuration Module, to their mapped equipment, specified on Association Module.

Smart Adaptive QoS Module: This module performs an active monitoring of the effectiveness of the Quality of Service policy delivered to the routers, by detecting for each equipment its bandwidth usage, Packet Losses, Jitter and latency, and readjust policies on each equipment simultaneously, all adaptation or liberation actions are stored on Repository Module.

3. Overview and Guided Tour for the PB-SAQOS For DMVPN Tool

3.1. Overview of the Tool

To simplify the use of the SAQ DMVPN model, its implementation is necessary, the SAQ-DMVPN tool is a WEB oriented application and it can run on any operating system with different browsers.

The developed SAQ-DMVPN Tool gives its users the following functionalities;

- Easy and quick creation and modification of the Quality of Service policies,
- An automatic conversion of user’s data in adequate command lines for the destination equipment,
- A dynamic QoS policies adaptation.

Figure 3, illustrates different steps of the operation of the application.

http://dx.doi.org/10.14257/ijca.2016.9.5.18

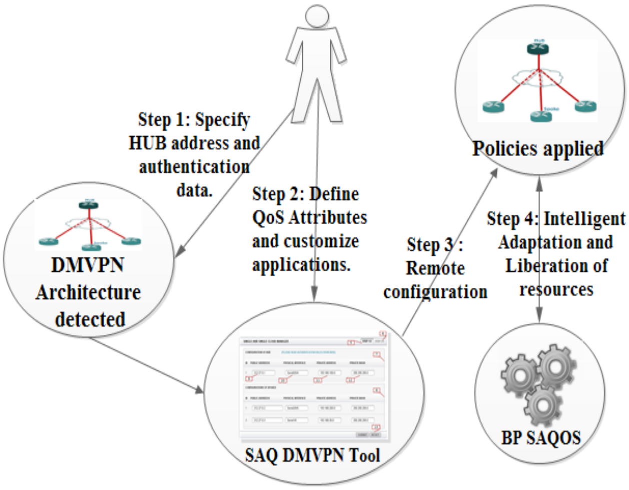

Step 1: Specify HUB address and authentication data.
Step 2: Define QoS Attributes and customize applications.
Policies applied
Step 3: Remote configuration
Step 4: Intelligent Adaptation and Liberation of resources
DMVPN Architecture detected
SAQ DMVPN Tool
BP SAQOS

Figure 3. Use Case Diagram of the Tool

Step 1: In client side, user specifies the address of the HUB and SSH authentication data, then the application consult the NHRP cache of the HUB in order to detect the architecture's equipment. As the initial setup of the DMVPN network isn't easy, a Framework was proposed in order to automate its configuration [23].

Step 2: The user through a friendly-user graphical interface: can customize the applications used by his company, configure QoS policies, and map policies to the specific equipment of the architecture.

Step 3: The Application performs: an automatic conversion of the user data into a specific command lines, and then ensures a secure remote delivery of policies.

Step 4: In order to automatically ensure an efficient management of the QoS policies and avoid any possible packets drops due to the insufficient bandwidth or high latency and jitter values, the BP SAQOS model perform automatically, dynamically an adaptation of resources then readjust policies.

3.2. Guided Tour of the Tool

The application menu is organized as follows [Figure 4]:

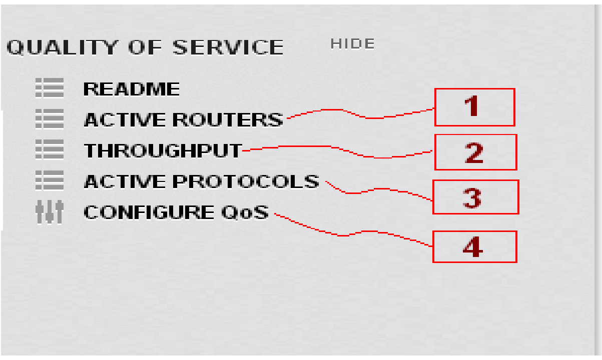
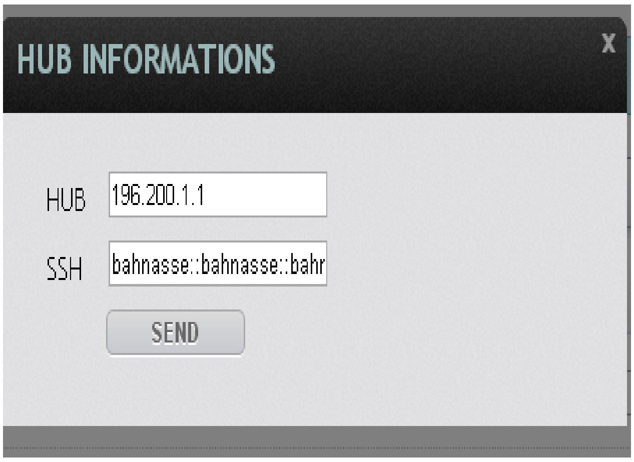


Figure 4. Application Menu - Quality of Service

- Option 1 : Allows the dynamic detection of DMVPN architecture equipment,
- Option 2: Allows the automatic dynamic detection of real Uplink and Downlink throughput of each architecture equipment,

- Option 3: Allows the customization of application and the detection of active flows,
- Option 4: Allows the management of QoS policies through a graphical friendly-user WEB interface, and the display of policies already configured and the reservation made.

3.2.1. Step 1-Option 1: The user indicates the public IP address of the HUB and specifies its authentication data's [Figure 5]



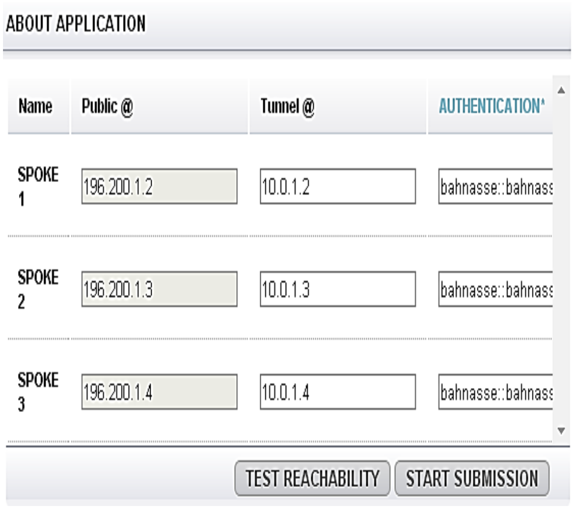
HUB INFORMATIONS

HUB 196.200.1.1

SSH bahnasse::bahnasse::bahr

Figure 5. Definition of the HUB IP Address and Authentication Data's

Application check authentication status then consults the NHRP cache of the HUB to detect architecture equipment [Figure 6].

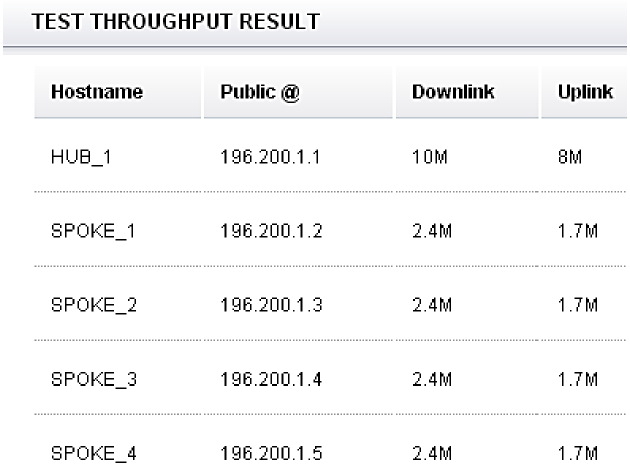


ABOUT APPLICATION

Name	Public @	Tunnel @	AUTHENTICATION*
SPOKE 1	196.200.1.2	10.0.1.2	bahnasse::bahnass
SPOKE 2	196.200.1.3	10.0.1.3	bahnasse::bahnass
SPOKE 3	196.200.1.4	10.0.1.4	bahnasse::bahnass

Figure 6. Detected Architecture

3.2.2. Step 2- Option 2: In order to guide the user to the implementation of quality of service policy, the application offers a dynamic and automatic detection of Uplink and Downlink throughputs of equipment detected on step 1, figure 7 shows the result of the Uplink/Downlink throughputs detection.



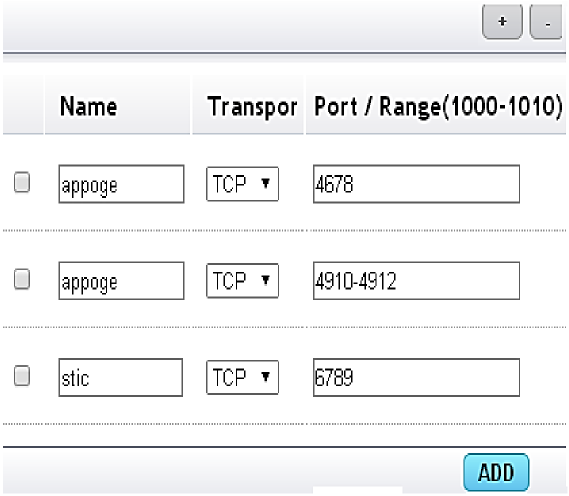
TEST THROUGHPUT RESULT

Hostname	Public @	Downlink	Uplink
HUB_1	196.200.1.1	10M	8M
SPOKE_1	196.200.1.2	2.4M	1.7M
SPOKE_2	196.200.1.3	2.4M	1.7M
SPOKE_3	196.200.1.4	2.4M	1.7M
SPOKE_4	196.200.1.5	2.4M	1.7M

Figure 7. Uplink and Downlink Throughputs Detected Automatically

3.2.3. Step 2- Option 3: Network metrology is also offered by the developed tool, the user can check active flows and customize applications, an application is characterized by; a transport protocol, port number or pool of ports numbers, figure 8 illustrates an example of applications personalization, in this demonstration we customized two TCP applications:

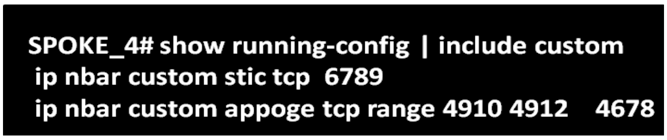
- Appoge: uses a pool of port number, 4679 and from 4910 to 4912,
- Stic: uses a single port number 6789.



	Name	Transpor	Port / Range(1000-1010)
□	appoge	TCP	4678
□	appoge	TCP	4910-4912
□	stic	TCP	6789

Figure 8. Customization of Applications

After the delivery process, the “show running-configuration | include custom” command on a SPOKE router, show the converted and delivered commands [Figure 9].

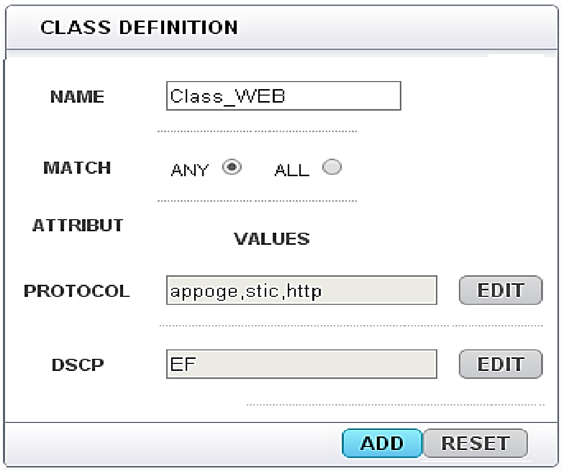


```
SPOKE_4# show running-config | include custom
 ip nbar custom stic tcp  6789
 ip nbar custom appoge tcp range 4910 4912    4678
```

Figure 9. Show Running-Configuration | Include Custom

3.2.4. Step 3- Option 4: The final step is the configuration of quality of service policies, the configuration will be set up in three phases:

Phase 1-The creation of class's [Figure 10]. The phase consists of the definition of class. A class can match to a protocol or a DSCP code. Protocols are selected from a list that contains customized and standard protocols. DSCP field can be EF, variety of AF or Precedence codes.



CLASS DEFINITION

NAME: Class_WEB

MATCH: ANY ● ALL ○

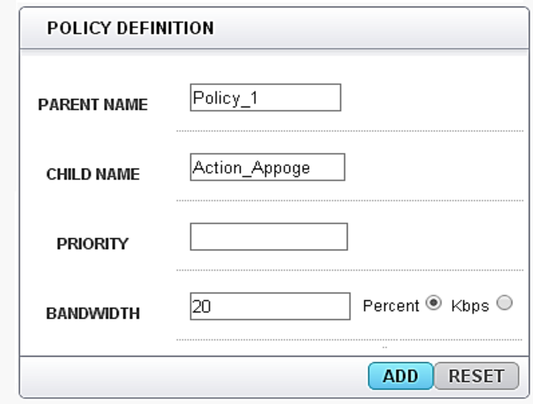
ATTRIBUT — VALUES

PROTOCOL: appoge,stic,http

DSCP: EF

Figure 10. Class Definition

Phase 2-The creation of actions [Figure 11] and the application of an action on a specific class [Figure 12].



POLICY DEFINITION

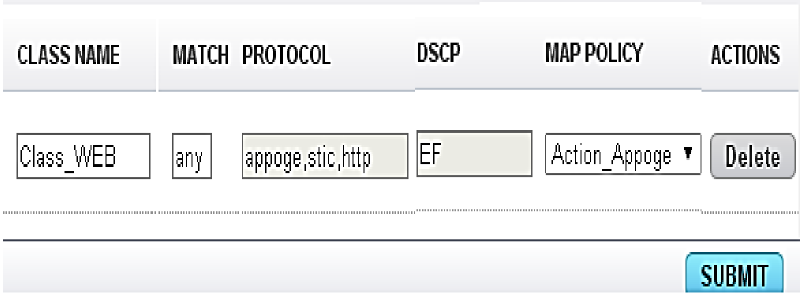
PARENT NAME: Policy_1

CHILD NAME: Action_Appoge

PRIORITY:

BANDWIDTH: 20 Percent ● Kbps ○

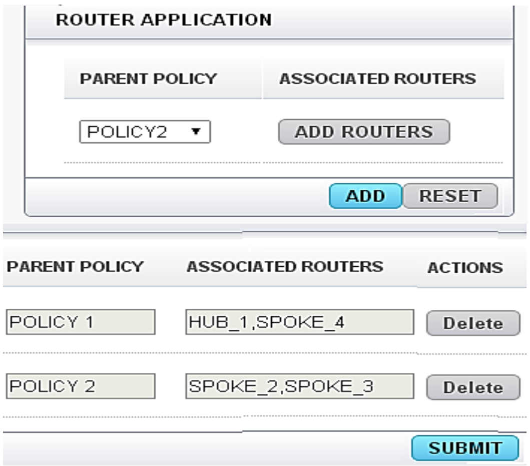
Figure 11. Action Definition



CLASS NAME	MATCH	PROTOCOL	DSCP	MAP POLICY	ACTIONS
Class_WEB	any	appoge,stic,http	EF	Action_Appoge ▾	Delete

Figure 12. Apply Action on Class

Phase 3- Map policies to specific equipment and deliver configurations remotely, Figure 13 illustrate an example of policy to router mapping, the application will convert user data's to a specific commands and deliver them to selected routers automatically.



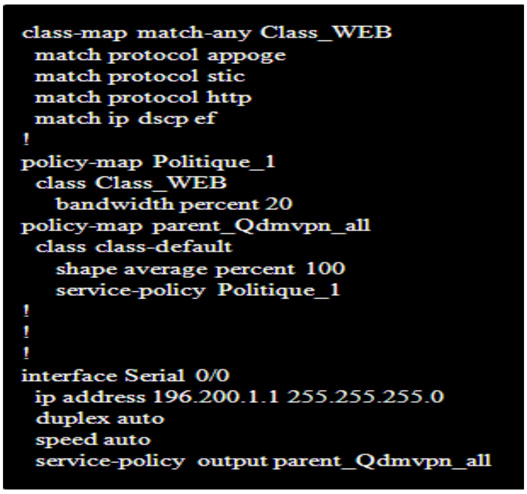
ROUTER APPLICATION

PARENT POLICY	ASSOCIATED ROUTERS
POLICY2 ▾	ADD ROUTERS

PARENT POLICY	ASSOCIATED ROUTERS	ACTIONS
POLICY 1	HUB_1,SPOKE_4	Delete
POLICY 2	SPOKE_2,SPOKE_3	Delete

Figure 13. Policy to Equipment Mapping

Figure 14 shows final configuration delivered to HUB_1 router.



```
class-map match-any Class_WEB
 match protocol appoge
 match protocol stic
 match protocol http
 match ip dscp ef
!
policy-map Politique_1
 class Class_WEB
   bandwidth percent 20
policy-map parent_Qdmvpn_all
 class class-default
   shape average percent 100
   service-policy Politique_1
!
!
!
interface Serial 0/0
 ip address 196.200.1.1 255.255.255.0
 duplex auto
 speed auto
 service-policy output parent_Qdmvpn_all
```

Figure 14. Policy Configuration Result

4. Simulation and Evaluation of the Framework

4.1. Experimental Topology

In order to evaluate the framework, we set the DMVPN architecture as shown on [Figure15], we simulated various applications:

- Bulk applications (FTP and TFTP),
- Real time applications (VOIP),
- Signaling protocol (SCCP),
- Best Effort protocol (HTTP).

VOIP traffic was generated by setting up Call Manager Express (CME) on Cisco routers [25], Cisco phones downloads their Firmware from the TFTP server using TFTP protocol, the firmware used in the simulation is 7921, its size is about 10.3MB.

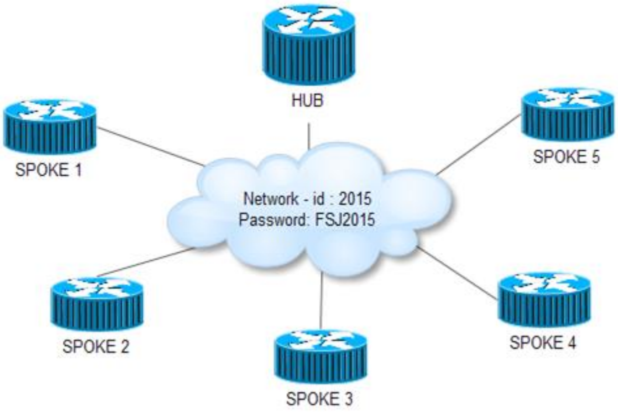


Figure 15. Evaluation Dmvpn Topology

We configured ten clients per site. Each site performs two simultaneous connections to other sites, for VOIP calls we used G729 codec, the minimum throughput required for ten simultaneous calls per site is 1.6Mbps [26] , assignments of resources and DSCP codes[Table 1] were made based upon a real example of a corporate network [27,28].

Table 1. Quality of Service Requirements

Traffic	Reservation	DSCP	Tool
Voice Signaling	Priority 20%	AF31	_
Voice Traffic	Priority 40%	EF	CME
FTP	Bandwidth 25%	AF11	FileZilla Server
TFTP	Bandwidth 10%	AF11	TFTPD 32
HTTP	Bandwidth 5%	BE	WAMP Server

In order to validate the PB-SAQOS model, we created 3 scenarios:

- Without Quality of Service,
- Using per-tunnel QoS for DMVPN,
- Using SAQOS for DMVPN.

The duration of each simulation scenario is fixed to one hour: Throughput of each spoke is chosen of E1= 2048 Kbit/s, the throughput of the HUB is chosen of 6144 Kbit/s, because the HUB will assume additional charges, like creating temporary tunnels, fulfill NHRP cache of Spokes, ensures the convergence of the network and delivers QoS policies to Spokes in case of Per-tunnel QoS for DMVPN scenario.

4.2. Obtained Results

Figure 16, illustrates the delay required for both solutions, Per-tunnel QoS for DMVPN and PB-SAQOS for DMVPN. Using the Policy Based technique, the graphical friendly-user interface and the remote centralized delivery of policies we achieved a low delay of configuration and delivery of policies compared to Per-tunnel QoS for DMVPN.

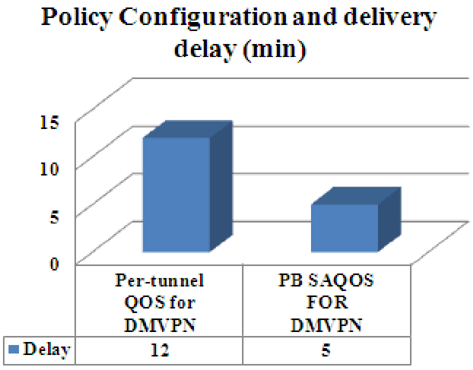


Figure 16. Policy Configuration and Delivery Delay in Minutes

Figure 17, shows a loss rate comparison between 3 scenarios: Without Quality of service, using per-tunnel QoS for DMVPN and PB-SAQOS for DMVPN, it turned out that PB-SAQOS for DMVPN framework is the best thanks to the BP SAQOS model used, that perform smart, dynamic and distributed allocation and release of resources between classes.

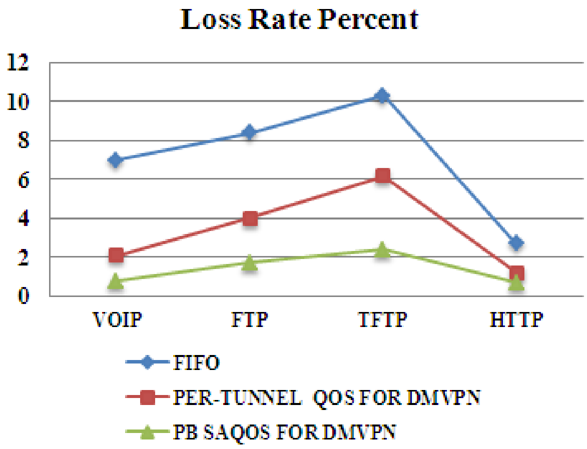


Figure 17. Loss Rate Percent

Figure 18, illustrates the jitter and latency values, with FIFO mechanism, Per-tunnel QoS for DMVPN and PB SAQOS for DMVPN, as show the two latter solutions offers acceptable delays for VOIP traffic.

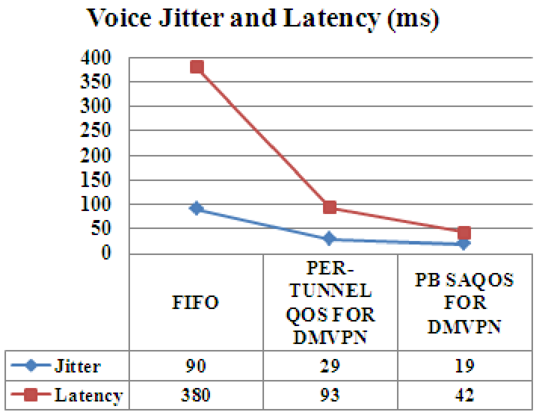


Figure 18. VOIP Jitter and Latency (Msec)

As shown from [Figure19] FIFO mechanism doesn't take too much CPU cycles because it has no additional task, unlike two others solutions, PB-SAQOS for DMVPN consume more CPU than Per-Tunnel QoS for DMVPN because it opens additional secure connections to routers.

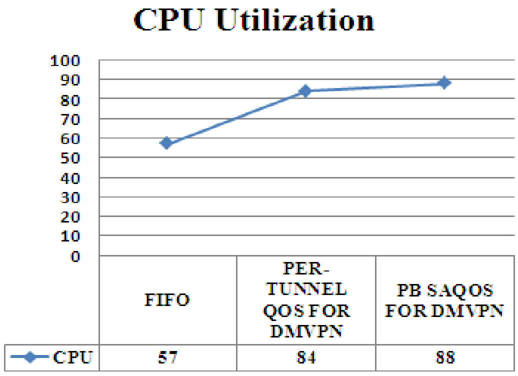


Figure 19.CPU Utilization

5. Conclusion

The proposed PB-SAQOS for DMVPN model, allows to automate the management of QoS policies in a very large and complex network like DMVPN, using a new web application, and readapt intelligently QoS policies based on the BP SAQOS model.

In order to evaluate the model, we created three scenarios, the first one without Qos, the second with Per-tunnel QoS for DMVPN and the last one with our model, flows used for each scenario were: VOIP, HTTP, FTP and TFTP. The simulation results proved that the PB-SAQOS for DMVPN is more efficient and flexible due to its simplified, automatic and smart resources management.

References

[1] S. Bhaskaran, S. Desai, L. Jou and A. R. Matthews, U.S. Patent No. 7,263,106. Washington, DC: U.S. Patent and Trademark Office, **(2007)**.

[2] C. J. Chase, S. L. Holmgren, J. B. Medamana and V. R. Saksena, U.S. Patent No. 6,188,671. Washington, DC: U.S. Patent and Trademark Office, **(2001)**.

[3] Dynamic Multipoint VPN (DMVPN) Design Guide, Corporate Headquarters Cisco Systems, Inc. **(2006)**, pp. 104.

[4] J. Luciani, D. Katz, D. Piscitello, B. Cole and N. Doraswamy, "NBMA next hop resolution protocol (NHRP)", Work in Progress, **(1997)**.

[5] S. Kent and R. Atkinson, “RFC 2401: Security architecture for the Internet Protocol”, Obsoletes RFC1825 [Atk95a]. Status: Proposed Standard, November **(1998)**.

[6] S. Ken and R. Atkinson, “RFC 2406”. Encapsulating Security Protocol, **(1998)**.

[7] Kent, S., Atkinson, R., & Header, I. A. RFC 2402. IP Authentication Header (1998).

[8] B. Adoba and W. Dixon, “RFC 3715–IPSec-network address translation (NAT) compatibility requirements”, **(2004)**.

[9] R. Asati, M. Khalid, A. E. Retana, D. Van Savage and P. P. Sethi, “U.S. Patent No. 8,346,961. Washington”, DC: U.S. Patent and Trademark Office, **(2013)**.

[10] H. Chen, “Design and implementation of secure enterprise network based on DMVPN”, 2011 International Conference on Business Management and Electronic Information (BMEI), IEEE, May, vol. 1, **(2011)**, pp. 506-511.

[11] A. Bahnasse and N. Elkamoun, “Study and evaluation of the high availability of a Dynamic Multipoint Virtual Private Network”, Revue MéDiterranéEnne Des TéLéCommunications, vol. 5, no. 2, **(2015)**.

[12] R. Jankuniene and I. Jankunaite, “Route creation influence on DMVPN QoS. In Information Technology Interfaces”, Proceedings of the ITI 2009 31st International Conference on IEEE. **(2009)** June, pp. 609-614.

[13] S. G. Thorenoor, “Dynamic routing protocol implementation decision between EIGRP, OSPF and RIP based on technical background using OPNET modeler”, In Computer and Network Technology (ICCNT), 2010 Second International Conference on IEEE. **(2010)**, April, pp. 191-195.

[14] S. Blake, D. Black, D. Carlson, E. Davies, Z. Wang and W. Weiss, “An architecture for differentiated services”, **(1998)**.

[15] V. Jacobson, K. Nichols and K. Poduri, “An expedited forwarding PHB”, **(1999)**.

[16] J. Heinanen, F. Baker, W. Weiss and J. Wroclawski, “Assured forwarding PHB group”,. RFC 2597, June, vol. 470, **(1999)**, pp. 471-472.

[17] R. T. Gibson, J. H. Buchanan, L. MacFadyen, R. MacCharles and M. Jamensky, “U.S. Patent No. 8,014,283”, Washington, DC: U.S. Patent and Trademark Office. **(2011)**.

[18] Y. Yu, H. Wang, Z. Zhou and D. Zhou, “Quality of service policy control in virtual private networks”, In Asia-Pacific Optical and Wireless Communications. International Society for Optics and Photonics. April, **(2004)**, pp. 1055-1060.

[19] M. Gunter, T. Braun and I. Khalil, “An architecture for managing QoS-enabled VPNs over the Internet”, In Local Computer Networks, 1999. LCN'99. Conference on. IEEE. **(1999)**, October, pp. 122-131.

[20] T. Braun, M. Guenter and I. Khalil, “Management of quality of service enabled VPNs. Communications Magazine, IEEE, vol.39, no. 5, **(2001)**, 90-98.

[21] Per-Tunnel QoS for DMVPN (ONLINE) http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec_conn_dmvpn/configuration/15-mt/sec-conn-dmvpn-15-mt-book/sec-conn-dmvpn-per-tunnel-qos.html

[22] Test, T. C. P. **(2005)**. Benchmarking tool for Measuring TCP and UDP Performance. (Online) WWW: http://www. pcausa.com/Utilities/pcattcp.htm.

[23] A. Bahnasse, N. El Kamoun, “Policy-Based Automation of Dynamique and Multipoint Virtual Private Network Simulation on OPNET Modeler” International Journal of Advanced Computer Science and Applications (IJACSA), vol. 5, no. 12, **(2014)**. http://dx.doi.org/10.14569/IJACSA.2014.051201.

[24] A. Bahnasse, N. El Kamoun, “Policy-Based Smart Adaptive Quality of Service for Network Convergence. International Journal of Computer Science and Information Security, vol. 13, no. 3, **(2015)**, pp. 21-27.

[25] Cisco IP Communications Express: CallManager Express with Cisco Unity Express. Cisco Press, **(2005)**.

[26] Press, C. Voice over IP-Per Call Bandwidth Consumption, **(2005)**.

[27] J. Babiarz, K. Chan, and F. Baker, “Configuration guidelines for DiffServ service classes”, IETF-Request for Comments (RFC 4594), **(2006)**, pp. 4594.

[28] S. Floyd and V. Jacobson, “Link-sharing and resource management models for packet networks. Networking”, IEEE/ACM Transactions on, vol. 3, no. 4, **(1995)**, pp. 365-386.