A BGP/MPLS PPVPN Management Information Model and a J2EE-based Implementation Architecture for Policy and Web-Based Configuration Management Systems

Marcos A. Siqueira, Marcel C. Castro, Maurício F. Magalhães, Lavoisier J. L. Farias

1CPqD Telecom & IT Solutions, Telecommunications Research and Development Center
Rod. Campinas – Mogi-Mirim (SP-340) km 118,5 CEP 13088-902 Campinas-SP Brazil
2DCA-FEEC-Unicamp PO Box 6101
CEP 13083-970 Campinas-SP, Brazil
{siqueira, mcastro, lfarias}@cpqd.com.br, mauricio@dca.fee.unicamp.br

Abstract

IETF's Policy Framework Workgroup is developing a policy model that covers the utilization of signaled and provisioned QoS. This model links high-level QoS business requirements to low-level device specific parameters through a policy architecture named PCIM (Policy Core Information Model). The policy architecture was first proposed for DiffServ and its policy components are distributed in a three-tier model formed by a policy repository, the policy decision points (PDP) and the policy enforcement points (PEP). On the other hand, the DMTF (Distributed Management Task Force) is proposing the Web-Based Enterprise Management (WBEM) Initiative, which takes the CIM (Core Information Model) management information models from DMTF and maps into XML Document Type Definitions (DTD).

This work presents the information model and implementation architecture of a policy management system for BGP/MPLS PPVPN configuration. The first goal is the proposal of BGP/MPLS PPVPN configuration policies to be mapped into LDAP through a given LDAP schema, so that the policies can be stored at the repository, retrieved and interpreted by the PDP. The second goal is the abstraction of the device management information base through the implementation of an XML layer, based on the WBEM standards. This layer allows the policy manager to communicate with different devices using HTTP on XML/HTTP aware devices or traditional mechanisms such as HTTP, SNMP, RMON and CLI through an XML-policy proxy translator. Finally, a software architecture based on the J2EE standards is proposed for the implementation of the whole system.

Keywords: BGP/MPLS PPVPN, MPLS, Policy Management, XML

I. INTRODUCTION

Traditionally, VPNs (Virtual Private Networks) can be set up over leased lines between customer sites, or by using tunneling/security software running in customer devices (such as Firewalls). The first approach is not a cost-effective solution for small enterprises, and both solutions do not scale for large enterprises. Therefore, the VPN outsourcing (using PPVPN - Provider Provisioned VPN) is becoming an attractive solution for enterprises. The enterprise does not need to acquire network elements for providing authentication, encryption and tunneling, as these functions are delegated to the Service Provider. A network-based VPN or PPVPN is built over the SP shared backbone.

Currently, IP Virtual Private Network (IP VPN) is widely used by the SPs for providing VPN services. IP VPNs can be defined as the collection of switching and transmission resources that will be used by a dedicated set of authorized users to exchange information over a shared IP infrastructure [1].

Some of the PPVPN architecture requirements are to: enable customer sites to be flexibly connected to allow connectivity to extranets, intranets or another access networks; support arbitrary topology such as hub and spoke and full mesh; provide network and data privacy and enable opaque transport of VPN traffic across the SP network; provide the Quality of Service (QoS), and most importantly to expand to several hundreds and thousands of sites and users. Therefore, the PPVPN technology must satisfy the requirements of privacy, flexibility, scalability and QoS.

The BGP/MPLS architecture [2] proposes solutions for all the issues listed above using a complex set of protocols and mechanisms. Therefore, the need for PPVPN management mechanisms becomes apparent, allowing Service Providers to manage the different components of
the PPVPN architecture. This paper describes the PPVPN management requirements, and proposes some solutions for MPLS VPs web and policy-based management. Firstly, a Policy-based information model for BGP/MPLS PPVPN management is proposed, followed by a J2EE-based architecture for implementing the policy management system.

II. BGP/MPLS PPVPN ARCHITECTURE
A BGP/MPLS VPN [1] is a VPN architecture described as a set of Customer Edge (CE) routers, each attached to one or more Provider Edge (PE) routers that are members of the Service Provider network, as shown in Figure 1. The network also includes Provider Core (P) routers. In this model, only the PE routers will maintain VPN membership and topology information. The CE routers take no active part in the MPLS domain and use standard IP routing protocols and mechanisms to exchange data and control information with other subscribers of their respective VPNs. PE routers specify reachability information between VPN members as BGP peer relationships with other PE routers attached as members of the same VPNs. LSPs (MPLS Label Switched Paths) are established between BGP peers and may carry traffic associated with multiple VPNs. In this case a BGP/MPLS VPN is able to deal with overlapping address spaces for different customers.

III. PPVPN MANAGEMENT REQUIREMENTS
Reference [6] proposes some requirements for PPVPN management, such as troubleshooting and problem resolution; query, configuration and receiving of alarms regarding the infrastructure supporting the VPN Service. The reference also proposes a management model based on the TMN model as shown in Figure 2, and is composed of the following three layers: service management layer network management layer and network element layer.

a) The Service Management Layer (SML)
The SML is related to the service point of view. The Customer Manager (CM) deals with client parameters such as SLS (Service Level Specification), billing, reporting of statistics and operational data to the Customer Agent (CA). The CM may, for instance, provide the capability of dynamic bandwidth allocation to a given customer through requests coming from the Customer Agent.

On the other hand, the Customer Agent should provide capabilities, such as VPN topology visualization, operational state, order status, and the configuration of dynamic requests, for example, on-demand bandwidth allocation.

b) The Network Management Layer (NML)
The NML should provide the FCAPS (Fault, Configuration, Accounting, Performance and Security) functions to the provider's network operator. The configuration tasks deal with the provision of the following
PPVPN components: Provider Edge Router, Customer Edge Router, tunnels, access connections, routing, QoS, etc. The parameters to be configured are: intranet and extranet membership, CE routing protocol, routing metrics, MPLS tunnels, identifiers for Service Providers, PPVPNs, PEs, CE devices, tunnels and access connections.

The architecture proposed in reference [6] does not specify either the Network Manager information model or the architecture, so the next section this paper proposes architecture and implementation of architecture solutions for these components and for the interfaces CA/CM, CM/PNM and PNM/PE and CE devices.

c) The Network Element Layer (NEL)

The policies configured at the upper layers should be enforced at the Network Element Layer devices. For BGP/MPLS VPNs, the provider devices are PE (Provider Edge) and P (Provider) routers, and the customer devices are CE (Customer Edge) routers. Most of the configuration tasks have to be performed for the PE devices.

![Figure 2: PPVPN Management Model](image)

IV. PROPOSAL OF A PPVPN MANAGEMENT ARCHITECTURE

a) Network Management Information Model

The PPVPN Management model aims at providing a set of classes, attributes and relationships, such as Object-oriented models for representing policies and configurations for the different physical and logical components on the network.

PPVPN configuration policies can be classified according to the following categories of information: topology information, addressing information, routing information, security information, and Quality of service (QoS) information. This set of information should be mapped into policy rules (conditions and actions) for allowing network provision decisions to be taken automatically by Policy Decision Points.

The device configuration information results in the creation of a virtual topology over the physical topology. Policy targets for IP VPN deployment are identified at the physical topology, while the virtual topology is used by the provisioning system to track the current status of the network resource allocation, due to previous VPN related configuration [1].

Figure 4 shows the BGP/MPLS PPVPN information model proposed in this paper. This model allows the configuration of the basic mechanisms necessary for setting up a PPVPN over a Provider network. The main class in the model is mpVFR, representing a Virtual Routing and Forwarding - VRF instance at a given Provide Edge Router. This class is associated with VPN route import and export policies, with a MP-BGP address family responsible for translating VPNv4 addresses into IPv4 addresses and vice versa, and with a set of PE interfaces which may be connected to client devices. The model also has classes for representing the Interior Gateway Protocol (IGP) running at the SP network and the IGP running between the CE and PE routers.

The MPLS information model should be implemented, according to previous work - reference [7].

Reference [1] proposes the policies shown in Figure 3 for PPVPN configuration tasks. The policy actions shown should trigger configurations at the information model objects. After the validation of these actions at the “virtual topology” the configuration must be conveyed to the network devices. As described in the next section, this paper proposes the use of HTTP/XML for enforcing the policy configurations at the network elements.

The policy actions for VPN configuration, shown in Figure 3, are:

- The action mpVpnPolicyVFICreationAction specifies a VFI (Virtual Forwarding Instance) or VRF for BGP/MPLS VPNs, to be created in a Provider Edge node.
- The action mpVpnPolicyRouteDistributionAction establishes connectivity between Provider Edge Routers performing the Label and route distribution (PE-PE), besides it provides the establishment of route distribution between Provider Edge and Customer Edge Routers (PE-CE).
- The mpVpnPolicyVPNTopologyDescriptionAction provides a description of the IP VPN topology according to the connectivity requirements of the MPLS VPN service.

Reference [1] proposes additional action classes for configuring VPN NAT (Network Address Translation) for extranets, VPN Firewall, Encryption, etc. These issues are not discussed in this work.

![Figure 3: BGP/MPLS VPNs Policies](image)
Figure 4: BGP/MPLS PPVPN Information Model

Figure 5: Implementation Architecture
b) Network Manager Architecture

The J2EE (Java 2 Enterprise Edition) framework [9] proposes the implementation of enterprise applications based on a three-tier architecture formed by the client-side presentation layer, the server-side presentation and business logic sub-layers, and the enterprise information system.

This paper proposes the application of the J2EE framework as a guideline for developing PBNM systems. The client-side presentation layer should be the management GUI, allowing the network operator to input, visualize and edit policies. The server side presentation layer should be implemented using Java Servlets on a Web Server with the Server-side business logic implemented using Enterprise Java Beans for each component, such as the PDP, the adaptation layer components and the DAO (Data Access Object). The DAO allows the data repository to be transparent for the rest of the system, permitting the transparent interchange of SQL databases and LDAP directory without any impacts that may be apparent in the implementation as a whole.

Conversely, the Java XML APIs could be used for parsing, transforming and conveying the policy actions throughout the management system, following the web-based management approach proposed in the DMTF WBEM standards [10]. Figure 5 shows the policy components mapped into the J2EE architecture.

IV. CONCLUSION AND FUTURE WORK

This work proposed the use of the Policy Based Network Management (PBNM) architecture for managing BGP/MPLS PPVPN mechanisms at Service Provider Networks. The VPN architecture was described and the management components and configuration tasks were identified. Given these inputs, firstly an information model and the policy rules were suggested and finally a J2EE implementation architecture was proposed.

The information model proposed allows the implementation of policy-based management systems for BGP/MPLS VPNs. With the implementation architecture proposed, the policy translation, transportation and interpretation is facilitated through the use of the different XML facilities, the options of middleware, and APIs available with the J2EE framework.

Ongoing and future work will include: The implementation of a module for translating the BGP/MPLS information model from LDAP objects into XML, based on a pre-established DTD (Document Type Definition); The implementation of a Policy HTTP/XML Proxy for translating Web-based management messages into device specific configuration commands such as CLI and SNMP; Implement and validate the top to down architecture based on J2EE standards as proposed in this paper.

V. REFERENCES