An Introduction to Virtual Private Networks: Towards D-VPNs

Luís Henrique M. K. Costa[1,2] — Serge Fdida[1] — Otto Carlos M. B. Duarte[2]

[1] LIP6 – Université Pierre et Marie Curie
4, place Jussieu
75252 Paris Cedex 05 - France

{Luis.Costa, Serge.Fdida}@lip6.fr

[2] Grupo de Teleinformática e Automação – GTA
Universidade Federal do Rio de Janeiro – COPPE/EE
P.O. Box 68504 - 21945-970 - Rio de Janeiro - RJ - Brasil

otto@gta.ufrj.br

ABSTRACT. *Virtual Private Networks (VPN) have many different implementations being deployed and numerous definitions are consequently found in the literature. Nevertheless, the VPN concept has two important characteristics: the ability to construct different "virtual" networks sharing the same physical infrastructure and to provide closed group communication. Starting from these basic properties, this paper first makes a survey of the main VPN technologies to afterwards extend the VPN concept giving the key ideas of the Dynamic VPNs (D-VPN). To define this new concept we analyze the similarities between the multicast/group communication model and the VPN concept, showing that attaching some properties, such as Quality of Service guarantees, to a multicast group is conceptually the same as providing these guarantees inside a VPN.*

RÉSUMÉ. *Les Réseaux Privés Virtuels (Virtual Private Networks - VPN) ont de nombreuses implémentations en déploiement et par conséquent la littérature en présente des différentes définitions. Néanmoins, les VPNs ont deux caractéristiques spécialement importantes : l'habilité de construire des différents réseaux "virtuels" qui partagent la même infrastructure physique et de permettre la communication de groupe contrôlée. Avec ces propriétés comme point de départ, nous présentons un panorama des principaux technologies des VPN pour ensuite proposer une extension naturelle de son concept, les VPN Dynamiques (Dynamic VPN - D-VPN). Pour bâtir ce nouveau concept nous analysons les similarités trouvées entre les modèles de communication de groupe/multicast et le VPN. Nous montrons qu'associer certaines propriétés, comme des garanties de Qualité de Service, à un groupe multicast revient conceptuellement à la même approche utilisée lorsque ces garanties sont ajoutées à un VPN.*

KEYWORDS: *virtual private networks, group communication, multicast, quality of service*

MOTS-CLÉS : *réseaux privés virtuels, communication de groupe, multicast, qualité de service*

1. Introduction

Virtual Private Network (VPN) is an evolving technology which is currently the subject of several academic and commercial projects. VPNs are mainly used for closed group communication within a controlled and secure network environment, without needing to construct a private network. Thus a major argument for the deployment of VPNs is the economical advantages in relation to dedicated networks.

Unfortunately, multiple definitions of a Virtual Private Network proliferate in the market place. It is difficult to give a closed definition of a VPN. Nevertheless, we find that a quite general and complete definition is provided in [FER 98]:

“A VPN is a communications environment in which access is controlled to allow peer connections only within a defined community of interest, and is constructed through some form of partitioning of a common underlying communications medium, where this underlying communications medium provides services to the network on a non-exclusive basis”.

A simpler definition is also presented:

“A VPN is a private network constructed within a public network infrastructure, such as the global Internet”.

We believe that the simplest definition of a VPN derives from the main concept of VPNs, the ability to provide closed group communication. Other VPN characteristics can be derived from the *properties* associated to that communication, such as the security level and Quality of Service (QoS) guarantees. In this paper we present the VPN motivations, implementations, and emerging solutions to address QoS and security. We introduce the envelope concept which is found at the crossroads of multicast communication and VPN.

1.1. *VPN motivations*

Maybe the strongest motivation of VPNs relies in the economics of communications. A collection of virtual networks implemented on top of a common physical communications infrastructure is cheaper than the construction of the equivalent collection of discrete networks, each for a single client.

Another interest of VPNs is, of course, the ability to turn some parts of an organization’s network virtually invisible to external observers. This aspect of communications privacy can play an important role for organizations. Depending on the level of security demanded by the organization, the simple abstraction of discretion and network obscurity of a VPN may suffice. However, for higher levels of user security a corresponding requirement for strong security of access and data transmission within the network is required, and additional mechanisms should be implemented.

The VPN abstraction is also interesting in providing closed group communication. VPNs can also be thought as a support of the same functionality of multicast communication, but within a controlled and secure environment. One may also think of the use of multicast communication within a VPN, which looks like a two-level group communication. The applicability of VPNs in this area is promising.

Another important motivation of VPNs is the intrinsic ability to partition network resources. By means of resource partitioning, it is possible to provide some kind of QoS guarantees to applications. Network resources may be reserved, or provided, in a per-VPN basis. Network resources may include buffers in routers, bandwidth on links, etc., and even address space might be thought as a shared resource (depending on how VPNs are implemented, address spaces can overlap).

1.2. *VPN requirements*

From the definition and motivations of VPNs, one can enumerate several requirements for VPN construction [GLE 00]. Any VPN has a minimal set of capabilities and may have optional features according to the requirements of the user. The main point is that for a VPN to work, it must be possible to map the user services to corresponding capabilities in the network infrastructure.

To define a VPN, an administrative mechanism is needed for designating membership in the VPN. A host may belong to one or more VPNs, as well as the global Internet. The ability to dynamically change the host membership is desirable. The addition of new hosts to the VPN (e.g., a dial user) should be signaled to the provider in the case of an enterprise VPN. In general this should be done transparently to the VPN members. Nevertheless, the VPN may be informed of user additions and deletions because of security or group management issues. The set of members of a VPN may be identified, [FOX 99] proposes one approach.

Regarding security, different capabilities may be required from the VPN depending on the user service. The minimal ability asked from a VPN is to guarantee some kind of isolation between the virtual network and the exterior world, as derived from the definition of a VPN. Depending on the level of security required, user connectivity may be defined including a variety of security mechanisms as IPSec [KEN 98]. Security may be requested discretely by end-user hosts or the VPN may enforce a mandatory security policy.

The VPN may provide QoS support. This can be done in two different ways: users may signal its QoS requirements with RSVP [BRA 97] (or other signaling method), IP precedence bits or what ever, or the VPN may internally assign QoS requirements to be mapped to the transmission infrastructure. For QoS to be guaranteed, the infrastructure may explicitly support QoS requests, or it may be projected in such a way that it consistently provides adequate QoS (with a certain level of confidence).

An enterprise may demand a specific level of availability from the service provider for the VPN service. This can also be understood as a QoS requirement and may be achieved by redundant links that are visible to, or hidden from, the user enterprise.

Other optional capabilities may also be defined, relative to addressing, load sharing, frame sequencing, MTU support, etc. [GLE 00].

1.3. *Administrative model*

There are two basic models of administration and management of VPNs. In the first category, the end user organization designs and operates the VPN, for example with end-user access through the public dial network. In the second model, a service provider has contractual responsibility for designing and operating the VPN in response to specified user requirements. Nevertheless, VPNs are not limited to these two models. Since the concept of VPN is tied to private communication but also to resource reservation and partitioning, one can imagine constructing VPNs within a single organization network, to ensure a certain QoS or data security to a group of users [GLE 00].

This work presents an overview of the VPN research area. Section 2 briefly reviews the main technologies of constructing VPNs. Section 3 discusses the QoS issues in VPN implementations. Section 4 presents the main VPN research projects. Section 6 presents some final remarks.

2. VPN Implementations

There are several kinds of VPN. Depending on their functional requirements, there may be also multiple methods of constructing VPNs. The selection of a specific VPN implementation includes issues like the security level needed, scalability relative to growing the VPN, as well as the complexity of implementing and maintaining the VPN (how automated is VPN creation, and how dynamic is the structure of the VPN?).

We can divide the several VPN types according to the layer in the TCP/IP protocol stack where the VPN is implemented. We can have network-layer, link-layer and transport/application-layer VPNs [FER 98].

The VPN types can also be divided into "peer" and "overlay" VPNs, according to the implementation model. In the peer model, the network layer path computation is done on a hop-by-hop basis. Each node in the data forwarding path is a peer node with the next-hop node. Traditional routed networks are examples of the peer model. Alternatively, in the overlay model the network layer forwarding is not done hop-by-hop, but via "cut-throughs" in the link-layer, communicating an entry node "directly" with an edge node on the other side of a large cloud. ATM and Frame Relay implementations are examples of the overlay model. In many cases this model can provide better performance, but scaling issues arise when a huge number of edge nodes is connected in a mesh.

We briefly describe in the following sections the characteristics of the main VPN types, beginning by network-layer VPNs (route filtering and tunneling) followed by link-layer VPNs.

2.1. *Route filtering*

Route filtering, or "controlled route leaking", consists of controlling route propagation in such a way that only certain networks receive routes for other networks which are within their own community of interest. This is a peer model, since a router within a VPN site establishes relationship with a router in the VPN provider's network, instead of edge-to-edge communication with other sites in the VPN.

While the common underlying Internet generally carries the routes for all networks connected to it, this architecture assumes that only a subset of such networks form a VPN. Given the lack of explicit knowledge of reachability to locations other than the VPN members, privacy of services is directly achieved by the inability of VPN members to respond to messages originating from outside the VPN.

Route-filtering VPNs are conceptually simple, but present some problems and are sometimes considered as a primitive VPN construction method. The use of partial routing information is very subtle to misconfiguration problems. It supposes a common routing core, which imposes problems to VPN addressing, since the different VPN address spaces can not overlap.

There is, however, a more scalable and less suitable to human misconfiguration approach to route filtering, the use of BGP Communities [CHA 96]. The use of the BGP communities attribute allows a VPN provider to mark BGP NLRIs (Network Layer Reachability Information) with a community attribute. Configuration control allows route information to be propagated in accordance with a community profile.

2.2. *Tunneling*

Sending specific parts of the network traffic through a tunnel is another method of constructing VPNs. The most common mechanisms are GRE (General Route Encapsulation) [HAN 94], L2TP (Layer Two Tunneling Protocol) [TOW 99] and PPTP (Point-to-Point Tunneling Protocol) [HAM 99]. DVMRP [WAI 88] tunnels can also be used: the MBone may be viewed as a kind of worldwide VPN.

Tunneling is an overlay model, thus scaling issues arise and are dependent on the utilization of point-to-point or point-to-multipoint tunnels and how these tunnels are used to construct the VPN.

2.2.1. *GRE tunnels*

GRE tunnels [HAN 94] are generally configured between an ingress and an egress router in the network. The packets forwarded through the tunnel are encapsulated with a GRE header and placed in the tunnel with the address of the tunnel endpoint as destination. In the endpoint the GRE header is popped and the packet continue to be forwarded in the original IP fashion. GRE tunnels are generally point-to-point but there are some vendor implementations that support point-to-multipoint configurations.

The architectural concept of tunneling is to build a VPN as a collection of tunnels across a common host network. Each tunnel connects the point of attachment to this common network to remote points from the same VPN. Routing in a specific VPN is isolated from routing in the common network. The several VPNs within the common host network may use the same address space without problems of overlapping. The tunnel can also encapsulate different protocol families.

Nevertheless, there are problems with GRE tunnels regarding the administrative overhead and scaling to large number of tunnels, since GRE tunnels must be manually configured. A complete mesh of tunnels with a large number of peering adjacencies can result in a negative effect on routing efficiency.

2.2.2. *Virtual Private Dial Networks*

There are several technologies, proprietary mechanisms as well as standard-based mechanisms, that are able to construct Virtual Private Dial Networks (VPDN). Nevertheless, there are two technologies that seem to grow in popularity: L2TP [TOW 99] and PPTP [HAM 99] tunnels. L2TP is in fact a mechanism originated from the combination of the earlier L2F protocol specification and the PPTP protocol [FER 98].

L2TP has a “compulsory” tunneling model, or NAS-initiated tunnel (Network Access Server or dial access server). It means that when a user dials to a NAS, this server creates a tunnel to a predetermined endpoint in the network, possibly based on a local configuration profile.

PPTP has a “voluntary” tunneling model, or client-initiated. It permits clients to configure and establish point-to-point tunnels to arbitrary located PPTP servers, without the participation of the NAS. The tunnel may terminate in any PPTP server in the network, given that this server is reachable through classical routing.

In fact, PPTP and L2TP have many similarities, the main difference between them residing in the tunneling model adopted. This tunneling model may be the decision factor when choosing one of the two technologies according to the needs of the application. The main issue is whether the network provider has the control of the PPP session to be established or not.

2.3. *ATM and Frame Relay VPNs*

To construct a conventional private data network, dedicated circuits from a public carrier are combined with additional private communications infrastructure. In general the large communications (public) infrastructure uses some kind of time-division and/or frequency-division multiplexing to create a dedicated circuit, where there is synchronization of the data clock of the sender and receiver. This clocking rate is fixed by the capacity of the dedicated circuit.

A link-layer VPN tries to provide the same functionality of these conventional private data networks, while achieving economics of scale by the use of a common switched public network infrastructure. A collection of VPNs share the same infrastructure. In general these networks operate at Layer 3 or higher, while the infrastructure is represented by Frame Relay or ATM. The main difference from conventional private data networks is that there is no synchronized data clock between the sender and the receiver, nor is there a dedicated transmission path. In general, the sender has no knowledge of the available capacity of the virtual circuit, so sender and receiver may use adaptive clocking of data. The sender can adapt its transmission rate according to the requirements of the application and signaling received from the network and the receiver.

A public switched WAN (ATM or Frame Relay-based) is very flexible. The client is unaware whether the provider is actually able to guarantee the contracted service at all times under all possible conditions. It is a task of the service provider the provision of sufficient network resources and the admission control of connections in the network to guarantee a certain QoS level. On the other hand, the client is able to transmit data at a rate higher than the contracted. The excess information units (cells) are marked (as DE - Discard Eligible in Frame Relay or by setting the CLP (Cell Loss Priority) bit in ATM). In times of congestion, these marked cells are firstly discarded. The service provider may project the network to provide a specific QoS level to the VPNs carried.

There are scaling issues with ATM and Frame Relay VPNs, related to the configuration and provisioning of new virtual circuits. A complete mesh of virtual circuits between all pair of endpoints may not scale, while on the other hand a partial mesh can lead to suboptimal routing. These scaling issues are common to the overlay model.

2.3.1. *Multi-Protocol Over ATM*

VPN implementation through the use of MPOA (Multi-Protocol Over ATM) [HEI 93] is similar to other “cut-through” methods. The idea is to use a switched link-layer to enable all “Layer-3 nodes” to be a single hop away from one another.

In this model, edge routers determine the path in the ATM network since they know which egress router packets must be forwarded to. Therefore packets are forwarded on a Virtual Connection (VC) specific to an egress router. Nevertheless, since egress routers cannot use ARP on the ATM cloud, an IP to ATM address resolution server is needed.

One advantage of MPOA is the use of dynamic circuits, while some other VPN implementations rely on statically configured structures. MPOA may be used to control the creation of edge-to-edge VCs dynamically. There are nevertheless two problems with this model. The first is the dependence on the ATM technology which poses problems with hybrid networks. Second, there are scaling issues as with other "cut-through" (and overlay) models, for example, routing sub-optimality when there is no full-mesh connectivity.

2.4. *Multi-Protocol Label Switching*

Most of the network architectures can be divided into two models: the use of network-layer routing and per-packet switching or the use of link-layer circuits and per-flow switching. MPLS (Multi-Protocol Label Switching) [ROS 00b] is an hybrid architecture that mixes these two approaches. MPLS VPNs [MUT 00] address the scaling issues presented by other models by using a single routing environment with VPN labels, as packet labels are necessary to activate per-VPN routing tables.

MPLS makes the Layer 2 switching environment visible to routing by adding Layer 3 (IP routing) functionality to switches. MPLS is intended to support a variety of link-layer technologies and heterogeneous Layer 2 switching environments. MPLS uses a label swapping forwarding structure. Packets entering the MPLS cloud are assigned a label based on local forwarding decision. Packets are then switched in the MPLS cloud by checking the incoming label that determines the next hop and produces the outgoing label, in a local label-indexed table. These tables are constructed by the use of the IP routing protocol in conjunction with a Label Distribution Protocol [AND 00].

This lightweight encapsulation based on the use of labels together with the notion of boundary-determined transit paths provides some of the mechanisms needed to support VPNs. MPLS VPNs are formed by the constrained distribution of routing information (a way to form VPNs and control connectivity between them), the use of VPN Identifiers (that concatenated with IP addresses provides unique addresses) and the use of MPLS to forward packets along the constructed routes [MUT 00].

In [ROS 00a], a method for implementing MPLS VPNs is proposed. In this approach BGP distributes routing information. The MPLS cloud (or backbone) is viewed as a set of P and PE routers (service Provider and service Provider Edge routers). PE routers are connected to CE devices (Customer Edge devices - often a switch or router). CE devices connect the MPLS backbone to the site network. Each PE router maintains a forwarding routing table per site; packets are forwarded only to sites that have at least one VPN in common with the originating site. It is desirable that the address spaces of (non-intersecting) different VPNs can overlap, thus the routing protocol has to distribute multiple routes to the same destination (address prefix in the case of BGP). MBGP (Multiprotocol extensions to BGP) [BAT 98] allows BGP to carry routes from multiple address families. A new address family, named VPN-IPv4, is defined to carry per-VPN routes. Figure 1 gives an overview of the BGP/MPLS VPN framework.

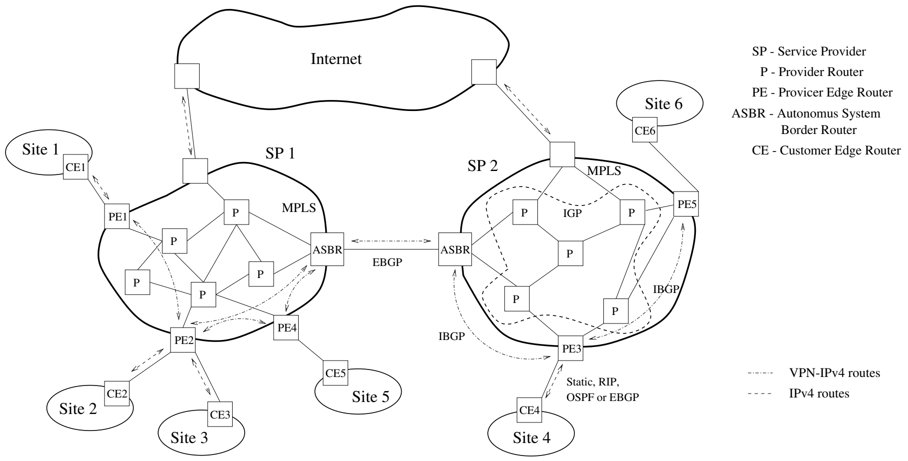

Figure 1. *BGP/MPLS VPN architecture.*

Scalability issues are taken into account by the architecture. P routers do not store any VPN routes, by means of a two-level labeling scheme in MPLS [ROS 00a]. PE routers maintain one routing table per site and store routes just for VPNs that are present in the sites directly connected to it. In Figure 1, router $PE1$ stores only one routing table for Site 1 while router $PE2$ stores two routing tables, one associated with the Site 2 the other for Site 3. Routing table for Site 1 stores routes just for sites which have VPNs in common with Site 1. In this way the communication between two sites that have no VPN in common is prevented and on the other hand VPNs that have no site in common may have overlapping address spaces. Each PE routing table is marked with the associated VPN (or VPNs), and the PE router distributes these routes only for other PE and $ASBR$ (Autonomous System Border Routers) routers that have sites within common VPNs connected to it. Thus VPN-IPv4 routes are exchanged just between PE routers, they are not propagated to the customer sites neither to the Internet backbone.

BGP route reflectors, if used, can be partitioned among VPNs so that each partition carries routes for only a subset of the VPNs supported by the service provider. The same applies to $ASBRs$, present in the case where VPNs span more than one service provider. No component has to store routes for all VPNs, improving scalability.

2.5. *Virtual Local Area Networks*

Virtual Local Area Networks (VLAN) [VAR 97] are a particular VPN type. VLANs were proposed as an alternative approach to the use of routers to contain broadcast traffic. The VLAN technology was recently standardized by the IEEE (standard IEEE 802.1q).

A LAN is considered a broadcast domain whereas a LAN segment is a collision domain. LAN segments are connected through bridges or switches, elements that do not forward collisions (as opposite to hubs and repeaters). LANs are connected through routers, elements that do not propagate broadcasts. VLANs allow a network manager to logically segment a LAN into different broadcast domains without routers. Bridging software defines which workstations belong to the broadcast domain and routers are only used to communicate between two VLANs.

VLANs are constructed by means of explicit tagging or implicit tagging. In explicit tagging a VLAN identifier is added to the data (in [VAR 97], tag headers for Ethernet, Token Ring and FDDI are presented). In implicit tagging the data is not marked, instead the VLAN from which the data came is identified based on information like the port on which the data arrived. This approach is less flexible, especially relating to host mobility.

Membership may be defined by the port (e.g. of a bridge), by MAC address, protocol type, IP subnet address, or even by the application type. Any combination of these methods supposes that the bridges have a coherent database with the VLAN membership.

The main arguments for the deployment of VLANs relate to performance gains especially when the network traffic consists of a high percentage of broadcasts and multicasts. There is also an economical argument for VLANs: using bridges instead of routers to implement broadcast domains is likely to be a cheaper approach.

2.6. *Transport and Application Layer VPNs*

Although there is no theoretical impeachment to the deployment of transport and application-layer VPNs, this is not very common. The most common implementation method is to use encryption services at either layer. For example, encrypted e-mail transactions and authenticated DNS (Domain Name System) are cited in [FER 98].

The Transport Layer Security protocol (TLS) [DIE 99] is worth mentioning. The main idea of TLS is to provide privacy and data integrity between two communicating applications. The protocol is composed of two layers: the TLS Record Protocol and the TLS Handshake Protocol. The TLS Record Protocol lays on top of some reliable transport protocol (e.g., TCP). The TLS Record Protocol provides private and reliable connections. Symmetric cryptography is used for data encryption. The TLS Record Protocol is used for encapsulation of various higher level protocols. One such encapsulated protocol, the TLS Handshake Protocol, allows the server and client to authenticate each other and to negotiate an encryption algorithm and cryptographic keys before the data transmission.

3. QoS Provision in VPNs

The definition of a VPN as a "private network" leads directly to performance issues, in other words, how one can associate some QoS characteristics to a specific VPN. There is a lot of work in progress that address the VPN QoS requirements [DUF 98, SYS , LIM , DEL 97].

A QoS framework for IP-based VPNs is presented in [DUF 98]. Hereafter we review its main concepts. As VPNs are supposed to replace networks constructed with private lines, the same kind of service offered by these networks would be expected from VPNs. Therefore, in addition to closed group communication and secure data transmission, VPNs have to offer some performance guarantees in terms of delay and bandwidth for example.

Two connectivity models are introduced. The first is the Virtual Private Link, where the VPN is constructed from a set of point-to-point virtual links. The tunneling techniques presented earlier fit into this model. The second is the Virtually Routed Network. Customer routers connect to provider routers and traffic isolation between VPNs is achieved by some kind of encapsulation within the provider network. Techniques like MPLS VPNs fit into this model.

A performance oriented service description is defined by means of two service abstractions. In the first one, a customer asks the service provider a pipe between two VPN sites with certain QoS requirements. To construct a VPN, a customer has to know the traffic matrix of the network and to ask for the pipes between every pair of sites in the network.

The "hose" performance abstraction is introduced [DUF 98]. The idea is to reduce the information that the customer has to pass to the provider. In this abstraction, a VPN endpoint informs the "outgoing" and "incoming" maximum data rates that this site is expected to send and receive from **all** other VPN endpoints (optionally a set of them). A hose may be viewed as a pipe where one endpoint is known and the other endpoint may be any other hose in the VPN. The customer does not have to provide a complete traffic matrix nor the spread of this traffic to other endpoints in the VPN. Figure 2 illustrates the hose model. The user site only informs its outgoing and incoming data rates, and the set of hose endpoints. The contract specifies that $x + y + z = 5Mb/s$, but it does not specify the values of x, y and z, as would be the case if the pipe model was used.

In the hose model supposes the connectivity between all endpoints (or a specified set). The hose abstraction gives the customer freedom to send traffic between every pair of endpoints provided that the traffic in and out of the endpoints does not exceed

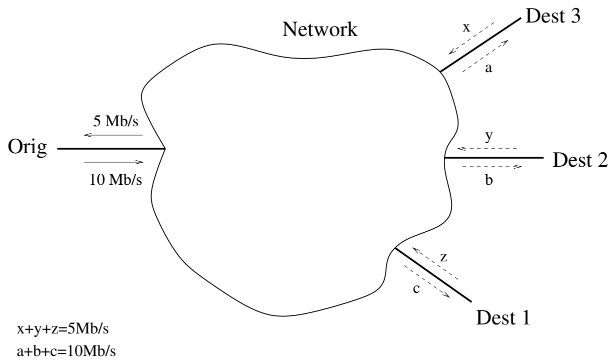

Figure 2. *An example of the hose model.*

the agreed capacities. In these conditions the service provider is expected to meet certain Service Level Agreements (SLA), such as bounded delay and packet loss. Therefore the problem is to the service provider to guarantee these SLAs under all possible conditions, all the time. The Virtually Routed Network model might fit better to this abstraction than the Virtual Private Link model.

QoS support

Regarding QoS provision there are several possibilities. One may define a single set of QoS requirements for the VPN. The VPN construction reflects these requirements, possibly with bandwidth reservation by the service provider. However we may also think of different QoS levels within the same VPN. There are at least two ways to achieve this more general model:

– Model A - To manage the resources on a VPN-specific basis. Resources for different flows, with different QoS requirements, are allocated from the resources reserved to the specific VPN. Here there are two possibilities:

1. to do scheduling only at the edges, with different scheduling services for different QoS flows (it is possibly done by the customer), or

2. to mark packets at the network entry, in this case the network serves differently different packet types. In any case resources are managed and reserved for a specific VPN.

– Model B - To manage resources on a QoS-basis, across all VPNs. The traffic specification is thus made for each of the QoS classes (a VPN with three classes of QoS has to provide three traffic specifications). The network treats differently each of the classes for a specific VPN, or in other words, the network manages the traffic across all tuples (QoS class, VPNs). In this technique, each origin of the VPN has a set of hoses with different QoS characteristics.

3.1. *Implementing QoS*

3.1.1. *Integrated Services*

The Integrated Services (IntServ) [BRA 94] framework is one of the possibilities to implement QoS within a VPN. Either the Guaranteed Service or the Controlled Load Service can be used to support the VPN resource management. In the IntServ framework, there is a signaling protocol (Resource reSerVation Protocol - RSVP) [BRA 97] that is used to make reservations between two endpoints.

In the hose model, the dynamic reservation mechanism provided by RSVP can be used to implement reservations between any two endpoints in a dynamic fashion. It remains to the service provider to guarantee the SLAs given that the hose specification is respected. The capacity negotiated for the hose plays a role when there are several pipes reserved for a customer.

One incompatibility between the hose model and the IntServ framework is that in the first the reservation is made for a pipe, that is, for all traffic between a pair of nodes; in IntServ the reservation is made on a per-flow basis. A pipe can potentially carry several flows of the same VPN. A resource aggregation might be used in this case.

3.1.2. *Differentiated Services*

In the Differentiated Services (DiffServ) [BLA 98] model scheduling decisions at routers are based in the DS byte of the IP header. The values of DS define different per-hop behaviors (PHB). These DiffServ PHBs may be used with additional signaling and/or provisioning to support QoS in VPNs.

When a customer constructs a VPN, it passes to the network a set of endpoints to be connected along with the traffic characteristics and QoS profile of the specific hose/pipe. The network has to check if there are available resources to meet the QoS requirements of this hose/pipe. In the case of a pipe, resources have to be available across a path between two endpoints. In the case of a hose, it is necessary to make such a check between all VPN access points. In this case one should assume a particular traffic matrix. Defining such a traffic matrix is a problem of the service provider, considering the dynamic characteristic of a hose. The bandwidth check may be made by a bandwidth broker (aware of the network topology) or by a hop-by-hop signaling mechanism, that in this case is used only for admission control. This approach differs from the IntServ model: there is no creation of packet classifiers or scheduling state at intermediate routers.

If resources are managed in a VPN-basis with scheduling made by the customer (model A - option 1), the QoS required from the network to a specific VPN would have to be the more stringent QoS requirement of that VPN. Model A option 2 would be difficult to implement since the DiffServ routers do not know how to distinguish traffic from different VPNs by default.

Model B, where resources are partitioned on a per-QoS basis, can be implemented by marking packets at the access point based on its traffic class. The admission control would be made in a per-class basis. The construction of the resource-management models proposed in [DUF 98] must be better studied since one may imagine many variations of them.

3.1.3. *MPLS*

MPLS seems to be a technology well suited to support QoS in VPNs, especially considering the hose abstraction and a virtual routed connectivity. A full mesh can be realized in MPLS by means of a sink tree (in fact, a Label Switched Path (LSP) tree) to each hose endpoint from all other endpoints. The performance model can be realized by dynamically changing the resources allocated to each LSP tree.

Using a sink tree makes it easy to the provider to check if the committed resources for a hose are respected, since the reservation requests flowing towards the exit point will be merged and only succeed if the total falls in the range reserved at the hose exit.

Another possibility is to use the IntServ model at the customer network. In this case the RSVP requests from the customer site are merged and translated to reservation requests in the MPLS network, in a way similar to the pure IntServ realization. The traffic management and service categories that MPLS will support are still in discussion [ROS 00b].

4. VPN Projects

There are some projects currently in progress that treat the problem of VPN QoS provisioning. We make a brief review of the most important ones.

Cisco Systems proposes a QoS-VPN architecture [SYS] that is based on the features of a proprietary QoS-provision product. The proposal includes packet classification (using a committed access rate), bandwidth management (including policing, shaping and bandwidth allocation), congestion avoidance and continuity of packet priority over Layers 2 and 3 by means of MPLS. Packet classification is made (by now) using the ToS (Type of Service) field of the IP header and packets can be marked following different criteria such as IP addresses, TCP/UDP port numbers, etc.. Bandwidth allocation is achieved by WFQ, in one of two ways, class-based or flow-based. Congestion avoidance is implemented by WRED. These components are combined to provide specific QoS guarantees. A VPN-specific signaling facility is absent.

Other projects are also concerned with VPN construction by the user. These projects try to automate the VPN generation and some of them address QoS issues. It is the case of VNS (Quality of Service Virtual Network Service) [LIM].

VNS proposes an architecture where users are provided an application that enables the construction of virtual networks. VNS is responsible for partitioning the network resources (bandwidth, router buffers, etc.) and allocating them to individual virtual networks. Routers are able to identify flows and virtual networks. A packet scheduling algorithm (HFSC - Hierarchical Fair Service Curve) at routers is responsible for delivering link sharing and supporting real-time service. The link layer is supposed to be ATM, and each virtual-link of the network is associated with a virtual circuit. In this way the bandwidth of virtual links can be specified.

The Supranet concept is introduced in [DEL 97]. The approach employed to provide QoS is to construct VPNs on top of a Integrated Services network. A Supranet is a network-layer VPN constructed by means of tunneling. The proposed architecture has a double-stack layered structure, with one stack for real-time services and the other for non real-time ones.

There are also other projects that are mainly concerned with automating the creation of virtual networks, such as the Genesis Kernel [CAM 99] and the X-Bone [TOU 98]. In the Genesis architecture, programmable networks are used to create "child" virtual networks. This is a new paradigm for automating the creation, deployment and management of the network architectures. One of the main motivations of the project is the emergence of open programmable networks.

The X-Bone is a system for the automated deployment of overlay networks, virtual networks constructed by means of encapsulation tunnels. The idea is to automate the creation of overlay backbones such as the 6-Bone and to facilitate testing of new protocols by creating an isolated environment. The X-Bone uses a graphical interface and multipoint control channel to manage overlay deployment at the IP layer, much like teleconference sessions are controlled via the session directory (sd) tool. The X-Bone has three basic components. The Overlay Manager is responsible for running the algorithms that acquire resources, configure them and manage them as a coherent overlay. Resource Daemons are processes running at or near resources such as routers and links, they are responsible for keeping information about the state of that resource and coordinating its sharing by multiple overlays. A Multicast Control Protocol uses a two-layer IP system similar to that used by sd/sdr. A common announcement channel (the first layer) is used for resource requests. Once the resources for an overlay are acquired, a control channel (second layer) is created for this specific overlay. This control channel reaches only that set of resources.

5. Towards Dynamic VPNs and the "Envelope" Concept

Advances in networking research and technology will enable sophisticated multimedia group communications. Therefore, besides increasing the speed of the local-loop as well as the core network, new services will be required to support the application needs. VPNs as well as VLANs share the objective of being able to shape the network resources to address some critical issues. Today, a VPN is mainly used for closed-group communication, secured and controlled. It avoids to construct a private network, allows to organize the network resources according to the customer needs and therefore might have an economic advantage. It is usually constructed through some form of partitioning of a common underlying communication medium. Assume that group communication becomes a major traffic source, we will face applications where the group of users varies in time and space. Unfortunately, today's VPNs are statically managed and configured and will suffer facing these new application dynamics.

There exist some similarities between VPN in the general sense and multicast/group communication. In fact, both concepts share common features, needs, and objectives. Namely, a group is a unique virtual object attached with a name and an address. A group is a set of entities that satisfy, at a given point in time, a well-defined membership rule. This leads to the decomposition of the framework into two different sets of systems. The social group carries an application semantics and the networking group is the abstraction that enables the social group entities to communicate. The social group holds some constraints so that the application can be established and maintained. This represents the basis of the group management functions. Active Group Integrity (AGI) conditions are a way to express the constraints on the group composition. It could be defined as a condition on the size of the group, the presence of a quorum, etc. Also, other aspects such as floor control and session management have to be considered.

A group can be either static or dynamic according to the fact that its composition can change over time. It is likely that most groups will be dynamic. Therefore, an entity needs to find a way to express its wish to join a group. This is achieved through a join operation that enables this entity to become a member of a given group (if the AGI is satisfied or without control in an open model such as the Deering one for IP multicast). The join is directed to a given group whose name has been advertised by another protocol. Being attached to a group address means for the group members that the routing protocol will be in charge to deliver the protocol data units to every single active user.

In order to be successful, a group communication will also require the availability of network resources to satisfy quantitative as well as qualitative requirements. Information on the QoS (Quality of Service) as well as the security needs will be attached to the group.

Similarities with the VPN concept clearly appear because of the common features summarized in Table 1.

A complex situation will appear when applications and users will become more and more dynamic in space and time. Todays' statically managed and configured systems will suffer facing this new capabilities. It will then become mandatory to extend the multicast concept to a Q-VPN one that is a VPN attached with QoS properties but also to a D-VPN solution able to attach QoS as well as dynamicity.

Assume that a set of communicating entities share the same space defined by a list of properties. An "envelope" (ξ) is defined as the set of properties attached to a group of active entities involved in a given communication in the wide sense (application, subnetwork, flow, etc.). The envelope defined properties such as AGI, QoS,

Table 1. *Similarities between the Multicast and VPN models.*

Multicast	**VPN**
Multicast address	Globally unique VPN Identifier
Multicast Membership	VPN Membership
Group Management (IGMP), Session Management, and Floor Control	VPN Management (static or dynamic) & Advertisement
Multicast properties (QoS)	VPN properties (QoS, security)
Traffic Engineering	Traffic Engineering
Multicast routing	VPN reachability

security and so on. These properties should be maintained by the system whatever the configuration of the environment and its evolution.

An envelope is therefore uniquely defined by an envelope ID (ξ_{ID}). The issue of being able to provide a unique (ξ_{ID}) in a distributed way is out of the scope of this paper. An envelope ID is attached to an envelope and used by the underlying system to perform the routing and distribution functions. Once an (ξ_{ID}) is chosen, it is distributed over the network using an advertisement protocol (ξ_{AP} : Envelope Advertisement Protocol). A new entity can join a given envelope. As such, and if granted by the envelope AGI, this entity will inherit from all the properties of that envelope and the system will adapt to cover this new entity with these attached properties. This operation is complex and requires sophisticated network engineering functions to extend the envelope to the new entity. Therefore, the basic transfer mechanism of the network can be tag based. A tag is a marker or eventually a piece of code that, when processed in a router will explicitly provide the envelope properties, namely: QoS, security, forwarding, and homing (mobility). We believe that the forwarding function is a sophisticated mechanism able to efficiently derive the output port from the tag information, as well as the required resources to forward the packet according to the envelope needs.

The envelope concept (ξ) is very similar to the group concept with QoS and security or the dynamic concept of a VPN. Therefore, it is likely that many mechanisms and protocols of these elements can be re-used in this framework. Nevertheless, we expect that looking at the problem from a different perspective will allow us to achieve a more elegant and efficient solution. This work is in progress.

6. Final Considerations

The VPN concept is very general, thus giving an exact definition of a VPN is a difficult task. Nevertheless, we are able to identify some characteristics that any "VPN-service" should provide: closed user group communication and data security. Economics of scale appear as a strong argument for the deployment of most of the

VPN solutions, provided by the VPN ability to share a common underlying infrastructure in a transparent manner.

VPNs may be implemented by a large variety of technologies, ranging from dial-lines to ATM or MPLS links, using multiple mechanisms to support data security as well as some level of QoS guarantees. Some of the protocols and technologies that are likely to be used to implement VPNs are in the area of current research, this is the case of the IPSec protocol and the MPLS technology. Thus we may expect several new implementation approaches in the near future.

Especially regarding QoS provision in VPNs, there is a lot of work to be done as some of the possible support-technologies of VPNs are still under discussion, like MPLS, and the Integrated and Differentiated Services architectures.

We have introduced the new envelope concept which is found at the crossroads of multicast communication and VPN. A work is on-going to build the architecture required to enable this system.

7. References

[AND 00] ANDERSSON L., DOOLAN P., FELDMAN N., FREDETTE A., THOMAS B., "LDP Specification", 2000, Work in progress: draft-ietf-mpls-ldp-11.txt.

[BAT 98] BATES T., CHANDRA R., KATZ D., REKHTER Y., "Multiprotocol Extensions for BGP-4", RFC 2283, 1998.

[BLA 98] BLAKE S., BLACK D., CARLSON M., DAVIES E., WANG Z., WEISS W., "An Architecture for Differentiated Services", RFC 2475, 1998.

[BRA 94] BRADEN R., CLARK D., SHENKER S., "Integrated Services in the Internet Architecture: An Overview", RFC 1633, 1994.

[BRA 97] BRADEN R., ZHANG L., BERSON S., HERZOG S., JAMIN S., "Resource Reservation Protocol (RSVP) - Version 1 Functional Specification", RFC 2205, 1997.

[CAM 99] CAMPBELL A. T., MEER H. G. D., KOUNAVIS M. E., MIKI K., VICENTE J., VILLELA D. A., "The Genesis Kernel: A Virtual Network Operating System for Spawning Network Architectures", *IEEE International Conference on Open Architectures and Network Programming - OPENARCH99*, 1999.

[CHA 96] CHANDRA R., TRAINA P., LI T., "BGP Communities Attribute", RFC 1997, 1996.

[DEL 97] DELGROSSI L., FERRARI D., "A Virtual Network Service for Integrated-Services Internetworks", *7th International Workshop on Network and Operating Systems Support for Digital Audio and Video*, 1997.

[DIE 99] DIERKS T., ALLEN C., "The TLS Protocol Version 1.0", RFC 2246, 1999.

[DUF 98] DUFFIELD N., GOYAL P., GREENBERG A., MISHRA P., RAMAKRISHNAN K. K., VAN DER MERWE J. E., DORASWAMY N., JAGANNATH S., "A Performance Oriented Service Interface for Virtual Private Networks", 1998, Internet-draft: draft-duffield-vpn-qos-framework-00.txt.

[FER 98] FERGUSON P., HUSTON G., "What is a VPN", Cisco Systems and Telstra Internet, 1998, White paper available at http://www.employees.org/˜ ferguson/.

[FOX 99] FOX B. A., GLEESON B., "Virtual Private Networks Identifier", RFC 2685, 1999.

[GLE 00] GLEESON B., LIN A., HEINANEN J., ARMITAGE G., MALIS A., "A Framework for IP Based Virtual Private Networks", RFC 2764, 2000.

[HAM 99] HAMZEH K., PALL G. S., VERTHEIN W., TAARUD J., LITTLE W. A., ZORN G., "Point-to-Point Tunneling Protocol (PPTP)", RFC 2637, 1999.

[HAN 94] HANKS S., LI T., FARINACCI D., TRAINA P., "Generic Routing Encapsulation (GRE)", RFC 1701, 1994.

[HEI 93] HEINANEN J., "Multiprotocol Encapsulation over ATM Adaptation Layer 5", RFC 1483, 1993.

[KEN 98] KENT S., ATKINSON R., "Security Architecture for the Internet Protocol", RFC 2401, 1998.

[LIM] LIM L. K., ZHANG H., MANKIN A., "Implementing Quality of Service Virtual Network Service (VNS) on CAIRN", project site: http://www.cs.cmu.edu/ hzhang/VNS.

[MUT 00] MUTHUKRISHNAN K., MALIS A., "A Core MPLS IP VPN Architecture", RFC 2917, 2000.

[ROS 00a] ROSEN E. E. A., "BGP/MPLS VPNs", 2000, Work in progress: draft-rosen-rfc2547bis-02.txt.

[ROS 00b] ROSEN E. C., VISWANATHAN A., CALLON R., "Multiprotocol Label Switching Architecture", 2000, Work in progress: draft-ietf-mpls-arch-07.txt.

[SYS] SYSTEMS C., "Quality of Service for Virtual Private Networks", White Paper available at http://www.cisco.com/warp/public/779/largeent/vpne/qsvpn_wp.htm.

[TOU 98] TOUCH J., HOTZ S., "The X-Bone", *IEEE GLOBECOM'98 - Internet Mini-Conference*, 1998.

[TOW 99] TOWNSLEY W. M., VALENCIA A., RUBENS A., PALL G. S., ZORN G., PALTER B., "Layer Two Tunneling Protocol (L2TP)", RFC 2661, 1999.

[VAR 97] VARADARAJAN S., "Virtual Local Area Networks", Ohio State University, 1997, Tech. report available at http://www.cis.ohio-state.edu/˜ jain/cis788-97/virtual_lans/index.htm.

[WAI 88] WAITZMAN D., PARTRIDGE C., DEERING S., "Distance Vector Multicast Routing Protocol", RFC 1075, 1988.

Acknowledgements

This work was sponsored by FUJB, CNPq, CAPES/COFECUB, and IST Project GCAP N^0 1999-10504.

Luís Henrique M. K. Costa *was born in Rio de Janeiro, Brazil, on October 4, 1973. He received the Electronic Engineer degree and the M. Sc. degree in Electrical Engineering from Universidade Federal do Rio de Janeiro, Brazil, in 1997 and 1998, respectively. He is currently doing a D. Sc. thesis at the University Pierre et Marie Curie (Paris 6), France. His major research interests are group communication, quality of service routing, and multicast routing.*

Serge Fdida *is a professor at the University Pierre et Marie Curie (Paris) since 1991. He received the Doctorat de 3eme Cycle in 1984, and the Habilitation a Diriger des Recherches specializing in Modelling of computer networks in 1989 from the University Pierre et Marie Curie. From 1989 to 1995, he was a Full Professor to the University Rene Descartes (Paris). His research interests are in the area of high speed networking, multicast communication, resource management and performance analysis. He is heading the Network and Performance group of the LIP6 Laboratory (CNRS-University of Paris 6).*

Otto Carlos M. B. Duarte *was born in Rio de Janeiro, Brazil, on October 23, 1953. He received the Electronic Engineer degree and the M. Sc. degree in Electrical Engineering from Universidade Federal do Rio de Janeiro, Brazil, in 1976 and 1981, respectively, and the Dr. Ing. degree from ENST/Paris, France, in 1985. Since 1978 he is Associate Professor at Universidade Federal do Rio de Janeiro. Presently, he is heading the computer network group (Grupo de Teleinformática e Automação - GTA). His major research interests are high speed communications, multimedia protocols, QoS guarantees, mobile agents, and active networks.*