Web Cache Location and Network Design in VPNs

Pasquale Avella[1], Maurizio Boccia[2], Roberto Canonico[3], Donato Emma[3], Antonio Sforza[3], Giorgio Ventre[3]

[1]Università del Sannio, Dipartimento di Ingegneria
Address: Corso Garibaldi 107 – I- 82100 Benevento (Italy)
Email : avella@unisannio.it

[2]Università di Salerno, Dipartimento di Ingegneria dell'Informazione e Matematica Applicata
Address : Via Ponte don Melillo – I-84084 Salerno (Italy)
Email : boccia@crmpa.unisa.it

[3] Università di Napoli "Federico II", Dipartimento di Informatica e Sistemistica
Address : Via Claudio, 21 – I-80125 – Napoli (Italy)
Phone: +39 081 7683831, fax : +39 081 7683816
Email : roberto.canonico@unina.it, demma@napoli.consorzio-cini.it, sforza@unina.it, giorgio@unina.it

Abstract

In the new marketplace of Provider-Provisioned VPNs, the VPN service is often provided in combination with other services, like Internet access and web caching. Deploying web caches in an organization's private network provides two benefits: reduction of incoming traffic and faster access to popular web resources. In this work we refer to Provider-Provisioned Virtual Private Networks whose virtual links may be properly dimensioned, according to a Service Level Agreement that provides bandwidth guarantees for each virtual link established through the Network Service Provider infrastructure. In such a scenario, locating a predefined number of web caches at the edge of the VPN impacts the problem of properly dimensioning the VPN virtual links, due to the necessity of taking into account the effect of web traffic redirection to the web caches. In this paper we present a two stage solution approach for the web cache location problem combined with proper VPN design.

Keywords : Location; Design, Integer Programming; VPN.

1 Introduction

Virtual Private Networks (VPNs) are private Wide Area Network facilities implemented over a public network infrastructure [1]. In the past, private WANs were traditionally implemented by interconnecting the various sites of an organization by means of leased lines. In recent years, the massive diffusion of the Internet has leveraged the market of IP-based Virtual Private Networks, realized by using the Internet or an IP-based backbone as supporting infrastructure [2, 3]. Most important Network Service Providers offer to their customers the possibility of realizing their own VPN by interconnecting an access device located at the customer premises (Customer Edge, CE) to a corresponding device located at the edge of the provider network (Provider Edge, PE) (Fig. 1).

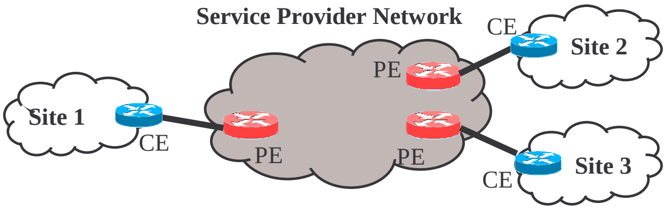

Figure 1 : IP-based Virtual Private Network

VPNs over Internet are becoming popular, thanks to their flexibility, scalability and cost-efficiency. However, VPNs over Internet also present some disadvantages, such as lesser reliability and the lack of Quality of Service (QoS) guarantees. With more ad more sensible applications being carried over VPNs, those disadvantages will become an important issue to be solved. Voice-over-IP (VoIP) is just an example of a new kind of application that requires communication services with QoS guarantees [4]. These requirements are accelerating the development of new networking technologies for the next generation VPNs. At present, Optical Wavelength Division Multiplexing (WDM) and Multi Protocol Label Switching (MPLS) appear as the most promising technologies for realizing VPNs with QoS guarantees [5,6,7].

In our paper we refer to VPNs of this latter type, i.e. we assume that the connectivity service is provided by a Network Service Provider according to a pre-negotiated *Service Level Agreement* (SLA), which guarantees a minimal bandwidth among each pair of VPN sites. In particular, we refer to *Provider-Provisioned VPNs* [8], i.e. VPNs for which the Service Provider participates in management and provisioning of the VPN. For Provider-Provisioned VPNs, tunneling and routing are carried out by edge devices provided by the Service Provider, whose functionality is shared by several concurrent VPNs. In such a scenario, the VPN service is often provided in combination with other services, like Internet access and web caching [9]. In most cases, in fact, organizations that use a Virtual Private Network to interconnect their geographically dispersed sites, also use their networking infrastructure to access the public Internet. Hence, the access routers used as Customer Edges of the organization's VPN, are crossed by two types of traffic:

- *intra-VPN traffic*, i.e. traffic exchanged with other VPN sites;
- *extra-VPN traffic*, i.e. traffic exchanged with the rest of the Internet.

For several organizations, a significant fraction of extra-VPN traffic is represented by World Wide Web traffic, i.e. traffic directed to web servers located outside the organization VPN. A common practice for large organizations is to deploy one or more *web proxies* within their network, acting as caches for the most popular web content. A proxy cache intercepts web requests from a population of several web clients and keeps a copy of most popular contents. This results in a faster access to popular web contents, which can be retrieved from local web proxies rather than from the original content servers. In a VPN scenario, this approach also leads to cost savings, due to a reduction of extra-VPN traffic.

In this paper, we study the problem of optimally locating a limited number of web caches at the edges of a Virtual Private Network. Deploying web proxy caches at some of the edges of a VPN impacts the problem of properly dimensioning the VPN virtual links. In fact, once web requests are redirected towards the web caches, part of the extra-VPN traffic is transformed into intra-VPN traffic, and this must be taken into account to properly design the VPN. This leads to a combined cache location and network design problem, for which we propose a two stage solving strategy and present some preliminary results. To validate the results obtained from the analytical model we intend to use a simulation environment based on the well known *ns-2* network simulator [10].

2 Cache location and network design in a VPN

The aim of our work is to find the best location for a limited number of web caches in a Virtual Private Network. To simplify the terms of the problem, we ignore cache misses. In other terms, when a document is transferred from the cache located at site *i* to a client located at site *j*, we only consider the cost of transferring the document from site *i* to site *j*, and we neglect the cost associated to a cache update (i.e. the cost associated to the transfer of the document from its original web server to the cache). Our assumption makes our problem somewhat similar to the problem investigated in [11], where the authors study the problem of locating mirrors of popular web sites at given places of the Internet. In this context, a mirror can be seen as a cache with a 100% hit ratio.

In the literature, the web cache location problem has been approached with classical location models, typically p-median or p-center models [12, 13], that do not take into account the traffic demand. Other approaches show that the cache location problem can be modeled as a dynamic programming problem [14]. On the other side, the VPN provisioning problem has been approached by other works, either as a classical network design problem, or more recently, according to the "hose" model [15,16].

Our work investigates both the cache location and the VPN design problems, in an integrated way [17]. A global model for the location and design problem contains a huge number of constraints and variables even for a small problem size. Such a model cannot be efficiently solved by a commercial Mixed Integer Programming solver (MIP) but requires more sophisticate combinatorial optimization algorithms like column generation and branch and cut and price techniques [18]. These approaches will be the object of our future investigations, while in this paper we are mainly interested to validate a new solution approach for the problem. Hence, we partition the solving strategy according to the double nature of traffic in the VPN: *intra-VPN traffic* (site-site demand) and e*xtra-VPN traffic* (site-web demand). Accordingly, we propose a two stage solving method. The first stage is aimed at determining the best location of caches at the VPN sites. This first stage also assigns each VPN site to one web cache, and provides the correct dimensioning for the virtual links that need to be established to carry the internal web traffic from each VPN site to the corresponding web cache. The second stage is aimed at properly dimensioning all the other VPN links according to the traffic matrix describing the site-site demand.

The first stage can be reduced to either a *plant location* model, if the number of caches is to be determined from the model itself on the basis of the tradeoff between location costs and design costs, or a *p-median* model, if the number of web caches has been decided in advance. In the following we will assume this latter case.

The second stage can be reduced to a Network Design problem, with some of the links already established in the first stage. The underlying flow model is a multicommodity flow model, in order to take into account the traffic origin/destination demand [19]. In case the demand could be splitted over several alternative paths, a continuous flow model should be adopted. Otherwise, i.e. if we assume that the traffic between each pair of VPN sites needs to be routed along just one path, an integer multicommodity flow model must be adopted. In this second stage, we could also assume that the number of virtual links departing from each VPN site can be bounded to a given maximum.

In the following we report definitions and notations used throughout the rest of this paper.

- $V = N \cup M$ is the set of vertices of the $G(V,A)$ graph, where:
- $N = \{1, 2, \ldots, n\} \subseteq V$ is the set of nodes representing the VPN sites, hence they can be both origin and destination of traffic demand, and potential candidates to host a web cache;
- $M = \{1, 2, \ldots, m\} \subseteq V$ is the set of the m most popular web servers for the VPN users, hence they are origin of traffic demand;
- $A = L \cup Q$ is the set of arcs of the $G(V,A)$ graph;
- $L \subseteq A$ is the set of potential virtual links to be established for the VPN (site-site links). $\forall i \in N$ and $\forall j \in N$ it must be decided if the arc of vertices (i,j) belongs to the optimal set of arcs to be established; in particular, $L = L_e \cup L_{ne}$, where:
  - L_e is the set of site-site arcs to be established as resulting from the cache location model,
  - L_{ne} is the potential set of site-site arcs to be established as resulting from the network design model;
- $Q \subseteq A$ is the full set of site/server arcs;
- O is the set of graph vertices which are origin of flow;
- F is the set of graph vertices which are destination of flow;
- $D_{ij} \geq 0$ is the demand generated from node $i \in N$ with destination in node $j \in N$ (this is the a priori traffic matrix between the VPN sites);
- $S_{kj} \geq 0$ is the demand generated from node $k \in M$ with destination in node $j \in N$ (this is the traffic demand from the web server k to the VPN site j, and coincides with the demand for the caches);
- S_j represents the total demand from site $j \in N$ to all the web servers, i.e. the extra-VPN traffic entering site j:
$$\forall j \in N \quad S_j = \sum_{k \in M} S_{kj} \ ;$$
- H is the cost to establish the (i,j) virtual link; for each arc $(i,j) \in A$:
  - $h_{ij}^{fixed} \geq 0$ is the fixed cost to establish the arc $(i,j) \in L$;
  - $h_{ij}^{unitary} \geq 0$ is the cost per megabit/s of the arc $(i,j) \in L$;
- B_1 is the available budget to establish and properly dimension the site-cache virtual links;
- B_2 is the available budget to establish and properly dimension the site-site virtual links;
- $l_i \geq 0$ is the cost of deploying a web cache at node $i \in N$;
- $C = \{c_{ij}\}_{\substack{\forall i \in N \\ \forall j \in N}}$ is the set of site/site moving costs; in particular, $c_{ij} \geq 0$ is the cost per flow unit associated to the arc $(i,j) \in L$, which may be estimated as proportional to the number of routers crossed in the path from i to j;
- w_{ij}^1 is the maximum capacity of the arc $(i,j) \in L$ with respect to f_{ij};
- w_{ij}^2 is the maximum capacity of the arc $(i,j) \in L$ with respect to g_{ij};
- $f_{ij} \geq 0$ is the continuous variable that represents the total flow over the arc $(i,j) \in L$ determined by the server/site demand;
- $f_{ij}^o \geq 0$ is the flow component over the arc $(i,j) \in L$ originated from the vertex $o \in O$, determined by the server/site demand;
- y_{ij} is a binary design variable defined as follows:
  $\forall (i, j) \in L, \quad y_{ij} = 1$ if the arc (i,j) is created, 0 otherwise
- y_i is a binary localization variable defined as follows:
  $\forall i \in N, \quad y_i = 1$ if a cache is located in node i, 0 otherwise

- x_{ij} is a binary variable, defined as follows:
  $\forall i \in N, \forall j \in N,\ \ x_{ij} = 1$ if site *i* is assigned to the cache located in *j*, 0 otherwise
- x_{ij}^{od} is a binary variable defined as follows:
  $\forall i \in N, \forall j \in N, \forall od \in O \times F,$
  $x_{ij}^{od} = 1$ if the arc *(i,j)* belongs to the path from the origin *o* to the destination *d*, 0 otherwise.

2.1 I stage: A combined p-median – network design model

In this stage we study the problem of optimally locating *p* web caches at the *n* sites of a given Virtual Private Network. We intend to deploy at most one web cache per VPN site, hence $p \le n$. For a given $p < n$, the case $p = n$ being trivial, we intend to select the *p* sites that will host a web cache, and assign each of the remaining $(n - p)$ sites to one of the *p* caches. For the $(n - p)$ sites not equipped with a web cache, we establish a direct virtual link between each site and the site hosting the cache of pertinence. This leads to a “clustering” of the VPN sites.

The proposed model is a capacitated p-median network design model:

$$Min \quad \sum_{i \in N, j \in N} c_{ij} f_{ij} \tag{2.1}$$

$$s.t. \quad \sum_{i \in N} y_j = p \tag{2.2}$$

$$\sum_{i \in N} y_{ij} = 1 \qquad \forall j \in N \tag{2.3}$$

$$y_{ij} \le y_i \qquad \forall i \in N,\ \forall j \in N \tag{2.4}$$

$$\sum_{i \in N} f_{ij} = S_j \qquad \forall j \in N \tag{2.5}$$

$$f_{ij} \le w_{ij}^1 y_{ij} \qquad \forall i \in N,\ \forall j \in N \tag{2.6}$$

$$\sum_{\substack{i \in N, j \in N \\ i \ne j}} (h_{ij}^{fixed} y_{ij} + h_{ij}^{unitary} f_{ij}) \le B_1 \tag{2.7}$$

$$f_{ij} \ge 0 \qquad \forall i \in N,\ \forall j \in N \tag{2.8}$$

$$y_{ij} \in \{0,1\} \qquad \forall i \in N,\ \forall j \in N \tag{2.9}$$

$$y_{ij} \in \{0,1\} \qquad \forall i \in N \tag{2.10}$$

The objective function expresses the minimization of the total moving cost. Constraints (2.2-2.4) are the usual constraints of the p-median model. Constraints (2.5) express the traffic generation of each node *j*. Constraints (2.6) express the upper bound for the flow on each arc *(i,j)*. Constraint (2.7) expresses the B_1 budget limit.

2.2 II stage: A network design model

In this stage we study the problem of optimally dimensioning the remaining links of the Virtual Private Network. We suppose that the traffic between each pair of VPN sites needs to be routed along just one path. The proposed network design model is the following:

$$Min \quad z = \sum_{od \in O \times F} \sum_{(i,j) \in L} D_{od} c_{ij} x_{ij}^{od} \tag{2.11}$$

$$s.t. \quad y_{ij} = 1 \qquad \forall (i, j) \in L_e \tag{2.12}$$

$$\sum_{i \in N} y_{ij} + \sum_{k \in N} y_{jk} \leq q \qquad \forall i \neq j, k \in N \tag{2.13}$$

$$\sum_{k \in N} x_{jk}^{od} - \sum_{i \in N} x_{ij}^{od} = \begin{bmatrix} -1 & j \equiv d \\ \sum_{d \in F} D_{jd} & j \equiv o \\ 0 & \text{otherwise} \end{bmatrix} \qquad \forall o, d \in O \times F, \forall j \in N \tag{2.14}$$

$$\sum_{od \in O \times D} D_{od} x_{ij}^{od} \leq w_{ij}^2 y_{ij} \qquad \forall (i, j) \in L \tag{2.15}$$

$$\sum_{\substack{(i,j) \in L_{ne} \\ i \neq j}} h_{ij}^{fixed} y_{ij} + \sum_{od \in O \times F} \sum_{\substack{(i,j) \in L \\ i \neq j}} D_{od} h_{ij}^{unitary} g_{ij} \leq B_2 \tag{2.16}$$

$$x_{ij}^{od} \geq 0 \qquad \forall (i, j) \in L, \forall o \in O \tag{2.17}$$

$$y_{ij} \in \{0,1\} \qquad \forall (i, j) \in L_{ne}, i \neq j \tag{2.18}$$

The objective function expresses the minimization of the total moving cost. Constraints (2.12) fix the arcs deriving from step I. Constraints (2.13) fix the maximum number of links per edge. Constraints (2.14) are the path continuity constraints. Constraints (2.15) express the upper bound for the flow on each arc (*i,j*). Constraint (2.16) expresses the B_2 budget limit.

3 Model solution and some preliminary results

Our two stage solving approach has been tested so far over both synthetic and real network topologies of small size. The preliminary results, obtained solving the model by the CPLEX Solver 8.0 are encouraging both in terms of quality of the solution and in terms of computation time.

In the following we report the results obtained for a small instance. Particularly we consider a network with 8 VPN sites and 3 Web Servers. In table 1 we report objective function values of the first stage for different values of the budget B_1 and for different values of p (number of caches). The values preceded by * indicate the minimal budget needed. In this table we do not report computation time because they are very small due to the size of the network. In table 2 we report the objective function values of the second stage and computation times for different values of the budget B_2 and for different values of q. Computation times are very small on average, except for three cases which present a peak. In conclusion, the two stage solution approach appears promising to solve the problem in scenarios of realistic dimensions. Nevertheless, solving greater instances could require more sophisticated combinatorial techniques to support the use of commercial MIP solvers.

h^{fixed}(€)	$h^{unitayo}$ (€/Mbps)	p	B_1(€)	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	z_1
3360	7360	2	*73152	0	1	0	1	0	0	0	0	41.4
3360	7360	2	79040	0	0	0	1	1	0	0	0	38.5
3360	7360	2	84192	0	1	0	0	1	0	0	0	35.4
3360	7360	2	88608	0	1	0	0	0	1	0	0	34.6
3360	7360	2	104064	0	0	1	0	0	1	0	0	34.4
3360	7360	2	∞	0	0	1	0	0	1	0	0	34.4
3360	7360	3	*55808	0	1	0	1	1	0	0	0	18.4
3360	7360	3	60224	0	1	0	1	0	1	0	0	17.6
3360	7360	3	∞	0	1	0	1	0	1	0	0	17.6
3360	7360	4	*42880	0	1	0	1	1	1	0	0	13.8
3360	7360	4	43616	0	1	0	1	1	0	1	0	11.2
3360	7360	4	48032	0	1	0	1	0	1	1	0	10.4
3360	7360	4	∞	0	1	0	1	0	1	1	0	10.4

Table 1: *p-median & design problem*

h^{fixed}(€)	$h^{unitary}$ (€/Mbps)	p	B_1(€)	z_1	q	B_2(€)	**n°arcs**	z_2	t_{cpu}(sec)
3360	7360	2	73152	41.4	10	860960	-	-	354.35
					10	*876320	33/50	648	4.03
					10	880320	32/50	642	499.90
					10	895040	32/50	640	2.27
					10	902400	32/50	638	1.92
					10	917120	32/50	636	241.84
					10	928480	31/50	634	8.27
					10	943200	31/50	632	5.04
					10	976640	30/50	630	7.47
3360	7360	2	73152	41.4	11	856960	-	-	2.43
					11	*860960	35/50	636	2.23
					11	864960	34/50	632	2.64
					11	891040	33/50	630	5.57
3360	7360	2	73152	41.4	12	852960	-	-	3.65
					12	*856960	36/50	630	1.02
3360	7360	2	73152	41.4	13	*852960	37/50	630	0.74

Table2*: network design problem*

In order to validate the results obtained from our model, we have also developed an extension of the *nam* Network Animator, a design tool used in combination with the *ns-2* Network Simulator [10]. Our tool allows to create realistic network topologies and to create VPNs over them [20]. This tool can be used to evaluate some of the simplifications made in the definition of the analytical model.

Acknowledgements

This work has been partially carried out under the financial support of the Ministero dell’Istruzione, dell’Università e della Ricerca (MIUR) in the framework of the FIRB project “Middleware for advanced services over large-scale, wired-wireless distributed systems” (WEB-MINDS), and of the project “Scalability and Quality of Service in Web Systems”.

References

[1] D. Fowler. "Virtual Private Networks". Morgan Kaufmann Publishers, 1999.
[2] B. Gleeson, A. Lin, J. Heinanen, G. Armitage, A. Malis. "A Framework for IP Based Virtual Private Networks". IETF Request for Comments, RFC 2764, Feb. 2000.
[3] J. De Clercq and O. Paridaens. "Scalability Implications of Virtual Private Networks". IEEE Communications Magazine, Vol.40 No.5, May 2002.
[4] "VocalTec Voice Virtual Private Network Solution". (2001) [Online].
Available: http://www.vocaltec.com/images/pdf/voip_vpn.pdf
[5] Y. Qin, K. M. Sivalingam and B. Li. "QoS for Virtual Private Networks (VPN) over Optical WDM Networks". In Proceedings of OPTICOMM 2000, SPIE/IEEE/ACM Optical Communications and Networking Conference, Dallas (TX), Oct. 2000.
[6] D. Mitra, J. A. Morrison and K. G. Ramakrishnan. "Virtual Private Networks: Joint Resource Allocation And Routing Design". In Proceedings of IEEE Infocom ‘99, Mar. 1999
[7] N. G. Duffield, P. Goyal, A. Greenberg, P. Mishra, K. K. Ramakrishnan and Jacobus E. van der Merwe. “Resource Management With Hoses: Point-to-Cloud Services for Virtual Private Networks”. IEEE/ACM Transactions on Networking, vol. 10 n. 5, pp 679-692, Oct. 2002
[8] R. Callon, M. Suzuki. "A Framework for Layer 3 Provider Provisioned Virtual Private Networks". IETF Internet Draft (work in progress, Mar. 2003). [Online]
Available: http://www.ietf.org/internet-drafts/draft-ietf-ppvpn-framework-08.txt
[9] Nortel Networks. “Universal IP VPN Services - Any Access, Any Core, Using Any IP VPN Model” (Whitepaper, 2003) [Online]. Available at http://www.nortelnetworks.com
[10] UCB, LBLN, VINT. "The Network Simulator - ns-2". http://www.isi.edu/nsnam/ns, 2003.
[11] S. Jamin, J. Cheng, A.R. Kurc, D. Raz, Y. Shavitt. “Constrained mirror placement on the Internet”. In Proceedings of IEEE Infocom 2001.
[12] P. Krishnan, D. Raz , Y. Shavitt. "The Cache Location Problem". IEEE/ACM Transactions on Networking, vol.8 n.5, pp. 568-582, Oct. 2000.
[13] R.L. Francis, P.B. Mirchandani. “Discrete Location Theory”, Wiley, 1990.
[14] B. Li, M. Golin, G. Italiano, X. Deng, and K. Sohraby. "On the Optimal Placement of Web Proxies in the Internet". In Proceedings of IEEE Infocom ‘99, Mar. 1999.
[15] A. Gupta, J. Kleinberg, A. Kumar, R. Rastogi, and B. Yener. "Provisioning a Virtual Private Network: A Network Design Problem for Multicommodity Flow". In Proceedings of the 33rd Annual ACM Symposium on Theory of Computing STOC'01, pp. 389-398, Jul. 2001.
[16] A. Kumar, R. Rastogi, A. Silberschatz, and B. Yener. "Algorithms for Provisioning Virtual Private Networks in the Hose Model". IEEE/ACM Trans. on Networking, vol.10 n.4, pp. 565-578, Aug. 2002.
[17] P. Avella, M. Boccia, R. Canonico, A. Sforza, G. Ventre. "A Cache Location Model for Virtual Private Networks ", EURO/INFORMS 2003, Istanbul, July 2003.
[18] Wolsey L.A. “Integer Programming”, Wiley, 1998.
[19] R.K. Ahuja, T.L. Magnanti, J.B. Orlin “Network Flows”, Prentice Hall, 1998.
[20] D. Emma, R. Canonico, G. Ventre. "Extended Nam Editor". [Online]
Available: http://www.grid.unina.it/grid/ExtendedNamEditor/index.html