行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※※※
※　　　　　　　　　　　　　　　　　　　　　　　　※
※　　　　　　虛擬私有網路效能的研究　　　　　　　※
※　　　　　　　　　　　　　　　　　　　　　　　　※
※※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：■個別型計畫　　□整合型計畫
計畫編號：NSC 90-2213-E-009-090-
執行期間：　90 年　8 月　1 日至　91 年　7 月　31 日

計畫主持人：廖維國
共同主持人：金仲達
計畫參與人員：施邦欣，王聰瑞，葉家宏，黃永瑋

本成果報告包括以下應繳交之附件：
□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
■出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

執行單位：國立交通大學電信工程系

中　華　民　國　　91 年　　10 月　　30 日

行政院國家科學委員會專題研究計畫成果報告

計畫編號：NSC 90-2213-E-009-090-

執行期限：90 年 8 月 1 日至 91 年 7 月 31 日

主持人：廖維國　　國立交通大學電信工程系

共同主持人：金仲達　　國立清華大學資訊工程系

計畫參與人員：施邦欣，王聰瑞，葉家宏，黃永瑋

國立交通大學電信工程系

I. 中文摘要

虛擬私有網路（ VPN ）提供這個商業公司(社團）一種無縫通信的方法。 在全球網頂部使用虛擬私有網路，合併多媒體訊息流和資料流是可能的。由於全球網之間不能保證強制性的計時約束，虛擬私有網路的應用不可能忽視這些網路狀態而不會調整。 取而代之，如果目前虛擬私有網路狀態不太壞，例如 網際網路話務(Voice over IP)，動畫隨取服務(Video on demand)或者 WWW 的應用就可作某種程度的彈性調整。 因此，如何判定目前虛擬私有網路狀態是否好到足以支援彈性應用，以及調整這些彈性應用，是一個重要課題。 在這個計劃中，我們將焦點放在研習以網路的仿傚器(network emulator)環境為基礎的虛擬私有網路效能。我們建構一個能夠容易設置和設計的一個虛擬私有網路的仿傚器。 另外，內容分配網路是為提供動畫隨取服務的虛擬私有網路，本計畫並發展一套代理暫存器管理來增進虛擬私有網路效能。

關鍵詞：虛擬私有網路，網路仿傚，動畫隨取服務，代理暫存器

Abstract

Virtual Private Network (VPN) offers the enterprise, possibly an institute or an Internet service provider, a way of seamless communication. Using VPN overlaying the shared IP network, it is possible to consolidate multimedia streams and data streams together. However, Due to the fact that the IP network cannot guarantee hard timing constraints, applications on VPN cannot be oblivious to the network conditions. Instead, applications such as video on demand or WWW are now elastic to a certain degree if the current VPN condition is not too bad. Therefore, it is a critical problem to decide whether the current VPN condition is good enough so that the elastic applications can still be supported and how to shape these elastic applications. In this project, we have built a high-throughput emulator for VPN that can be easily configured and programmed. Besides, we also developed a cache management scheme tailoring to the content distribution network, a VPN dedicated to the provisioning of the video on demand service.

Keywords: virtual private network, network emulation, video on demand, proxy cache.

II. Motivations and Objectives

Virtual private network (VPN) provides enterprise that have many sites a secure and private way to interconnecting their sites through the public network infrastructure. In such a network, the communication is restricted to taking place only among the sites of the corporation. VPN is private since the access to the VPN is restricted to a set of defined entities, and third parties cannot gain access. The private network is virtual in the sense that the transmission lines may overlay with the shared IP network. All users in the VPN can communicate with each other seamlessly through the shared network as if they were in the same private network.

One major application of VPN is to carry the voice and video stream of the corporation. *Voice over IP* (VoIP) and *Video on demand* are two of the fastest growing applications in the world today. For example, VoIP traffic grew by almost 900 percent from 1998 to 1999, and is projected to grow another 5000 percent by 2004 (135 billion minutes in year 2004). The fast growth stems from many inherent benefits, one of which is saving money. Basically, when VoIP is deployed, the long-distance charges – including charges for international calls – can be transformed into a flat monthly fee.

One consequence of transmitting voice over the shared IP network is that the transmission may be highly-variably delayed or even lost. To keep variations in the delay – *jitters* – from degrading the voice quality, a *jitter buffer* is used to store the incoming voice packets in the recipient and then play them out at a constant rate. When a voice packet is lost due to heavy traffic on the network, speech decoders generally play the previous packet again to maintain the pace of the playback. How to allocate enough buffer space and adjust the space properly becomes a very important issue in VoIP over VPN. On the other hand, the high-volume characteristics of video on demand service also impacts the VPN performance seriously. To address these issues, our research will provide an evaluation system with measurement and estimation methodologies for characterizing the performance of VPN.

III. Results and Discussions

Our results can be separated into two categories: network emulation and cache management for video on demand.

A. Network emulator design

Network simulation helps evaluating various design variations. Usually, the network simulation is carried out within a single computer. As an extension of network simulation, in a general network emulation settings, the network emulator models or simulates the network behavior but links to the operational nodes via real networks. Real packets produced by an operational node destined to the other host, which could be another operational node, will be led into the network emulator first. Upon the receipt of these packets, the network emulator discards, modifies, or delays the packets according to the modeled or simulated network behavior, and accordingly they are sent to the destined host. Thus, the network emulation offers an evaluation that is closer to its real-world equivalent.

When the volume of input/output emulated traffic to and from the emulator goes high, the usual situation in VPN, the network emulator, even on a dedicated host, will bias the instant of scheduled event, and even worse, degrade the emulator throughput. There are two reasons accounted for these performance problems. First, burst of network interrupts will tie up the system kernel. This prevents the network emulator from accessing the CPU resources and thus leads to the throughput degradation of network emulator. Second, the network emulator might not be able to dispatch events in time, which causes the scheduled event delay.

To solve the problem of interrupt burst, we propose an enhanced version for ns2 emulator called RTns2 to incorporate ns2 emulator into the Real-time Linux (RTLinux). The RTLinux allows us to control the signal of interrupts to Linux. It also helps to record more precisely the time when a network interrupt occurs. Two techniques: interrupt control and traffic pattern maintaining, are developed on top of RTLinux to solve the burst interrupt problem. To solve the overrun problem, we develop an overrun control mechanism. Our experiment results in the following table show that the developed techniques do improve overrun situation and the problem of interrupt burst.

	Original NS	NS with Hard Real Time Scheduler
Throughput (pkts/s)	584	10295
Event Delay	38 seconds	646 μ s

B. Proxy Cache for Content Distribution Network

Not until recently, HTTP is recognized as an efficient method to support interactive video; however, many fundamental issues are still required to be solved. To this end, we propose an approach of segment-based video streams retrieval and caching via HTTP byte-range requests to further mask the possible start-up delay, enhance the interactive function "pause", and simplify the web proxy cache design. Proxy partial caching introduces the second architectural challenge called dirty-initial-segments problem, i.e. cached portion in the proxy cache is inconsistent with the original in the server. If this happens, the cached portion already delivered to the client cannot be concatenated with the latter portion from the server. To relieve this problem, we propose a novel algorithm called validating upon partial replacement, which synchronizes the operations of proactive consistency validation, partition and replacement performed upon an object. We show that our proposed algorithm bears the merit of controllable chance to disturb users, easily embedding into the existing web proxy cache software with minimum modification, and improving the cache performance.

IV. Self-Assessment

We have achieved the first-year goal of this project, i.e., building up a network emulator. The network emulator allows us to diverse follow-up researches. We are searching for publishing our recent results. By the way, we are now enhancing the current version of network emulator to the version enabling cluster computing.

From our study during this project year, we also find many fundamental problems associated with the VPN, e.g., efficient provisioning of the content distribution network using VPN. We have started our research from developing a proxy cache management, which is published in IEEE international conference on computer and communication network. Further analysis is underway now.

References

［1］ G. Thomsen and Y. Jani, "Internet Telephony: Going Like Crazy," *IEEE Spectrum*, May 2000, pp. 52-58.

［2］ E. Rosen and Y. Rekhter – "BGP/MPLS VPNs", RFC 2547, March 1999.

[3] B. Li, M. Hamdi, D. Jiang, and X.-R. Cao, "QoS-Enabled Voice Support in the Next-Generation Internet: Issues, Existing Approaches and Challenges," *IEEE Communication Magazine*, April 2000, pp. 54-60.

[4] Berkeley NS research group, USC/ISI and Xerox PARC. *"The NS Manual"*, 2001.

[5] Michael Barabanov and Victor Yodaiken. *"Real-Time Linux"*, 1996.

［6］ Husni Fahmi et al. "Proxy servers for scalable interactive video support," IEEE Computer, pages 54-60, September 2001.

[7] Derek L. Eager, Michael C. Ferris, and MaryK. Vernon. "Optimized regional caching for on-demand data delivery," In Proceedings of Multimedia Computing and Networking, January 1999.

［8］ W.K. Liao and P.S. Shih, "Architecture of Proxy Partial Caching Using HTTP for Supporting Interactive Video and

Cache Consistency", IEEE ICCCN 2002.