2 Question B: Paper Review

Title: An Adaptable Network Architecture for Multimedia Traffic Management and Control
Authors: Sheng-Yih Wang and Bharat Bhargava
In Proceedings of IEEE International Conference on Multimedia and Expo (ICME), New York, NY, 2000.

2.1 Summary

Quality of Service (QoS) provision for emerging applications is an important research topic. This paper presents the design of an adaptable network architecture, called ADNET, which provide mechanisms to allow the application adapt to the resource constraints to achieve improved QoS. ADNET is designed to satisfy the requirements of adaptability, safety, security, efficiency, scalability and inter-operatability. A new virtual machine is proposed as the execution environment for active traffic flows. ADNET is IP-based. On top of IP, a transport layer protocol is introduced for active traffic, which is called ACTP. A part of the functionality of ADNET is simulated using ns-2 network simulator. Three schemes (IP fragmentation, ACTP fragmentation with or without active program) of video transmissions are compared. The authors show that QoS is improved in ACTP scheme with active programs. The ultimate goal of this research aims to unify different QoS control mechanisms together to provide a wide range of network services to all users and meet their specific needs.

2.2 Major Contributions

The major contributions of this paper are as follows.

- The design of an adaptable network architecture, called ADNET, which allows all of the active traffic, Intserv traffic, and Diffserv traffic to co-exist.
- A new virtual machine and CPU scheduler for the execution of active traffic flows.
- A new transport protocol called ACTP to isolate the interactions among active traffic flows and other traffic flows based on other transport protocols such as TCP or UDP.

2.2.1 ADNET

A unique feature of the design is the explicit consideration of resource management. The concept of *default resource reservation* is introduced to provide better service the active traffic.

ADNET explicitly includes the CPU scheduler as one of the resource management component. Without proper management of CPU time, a router can not provide guaranteed services to the applications.

ADNET addresses simultaneously the issues of the overheads associated with per-flow accounting and customized services for individual traffic flow. ADNET seeks to provide a broad spectrum of services ranging from traditional stateless best-effort to Intserv-style per-flow accounting. In one extreme, per-flow customized service such as Intserv or active traffic can be provided. On the other extreme, traditional best-effort service can be provided. In the middle, aggregated QoS schemes such as Diffserv can be provided.

The resource reservation mechanism for traffic flows in ADNET architecture is not active. The current design employs the RSVP-style resource reservation mechanism. To provide safe and secure execution of active capsules a common resource reservation mechanism has to be built into the infrastructure. It is apparent that the current RSVP will not be suitable for this task because it is heavy-weight and no CPU scheduling capability is included.

ADNET routers can play different roles, depending on where they are located: (1) As *leaf (first hop) routers*: The sender can simply send any traditional traffic as usual. The sender can send active traffic with or without reservation. If a Premium service (as in Diffserv) is required, the sender can send an active program to the router to request resources. After validation, the request is granted by the router and the traffic from the sender will be tagged as Diffserv traffic with specific Diffserv class.(2) As *boundary (border) routers*: The egress router will issue a setup request to the ingress router of ISP. (3) As *core routers*: The AEE only process the active flow that are authorized to go across the domain and enter the current domain. The router will not accept Non-Active Explicit Reservation to ensure safety.

2.2.2 Execution Environment of Active Traffic Flows

Virtual Machine. Although Java Virtual Machine is popular, a new customized virtual machine for running programs inside the router is necessary because of the following reasons:

(1) *Compactness.* The object code (or bytecode) format for currently available general-purpose virtual machines such as JVM is not very compact. The original design of these VMs is for execution in the end system or device, which usually don't need to handle a huge amount of VM programs. In contrast, the router may need to process thousands or even millions of active programs in a very short period of time. Even the existing VM specifically designed for active networks such as Spanner is not able to encode any interesting algorithms in few bytes. (2) *Resource management*. Currently available general-purpose virtual machines such as JVM dose not include fine-grain resource management mechanisms. In particular, the VM is not tightly coupled with the OS, therefore resource management can not be done effectively. In the design of ADNET, the VM includes operations dedicated to resource management tasks.

The VM plays several roles in the overall architecture. Specifically, the VM can (1) work as a Bandwidth Broker (BB), (2) work as a signalling mechanism, (3) change DS Codepoint definitions and update policy database, (4) constrain resource consumptions of active flows.

The VM includes some instructions that are unique: (1) Resource management (Request and Setup). (2) Specialized multimedia data processing (MPEG and H261 encoding/decoding). (3) User installable operation codes. The program can ask the virtual machine to install some particular part of the code into a single opcode. There are two modes of installable opcodes: the *secure mode* and the *light-weight mode*. In secure mode, the security hash algorithm MD5 is used to generate the footprint of the user code. In light-weight mode, the opcode is determined by the user as a 16-bit number.

CPU Scheduler. For each active traffic flow, a thread (light-weight process) of the designated execution environment (virtual machine) is created to handle the flow. The execution of the thread will be under the control of the CPU scheduler to compete for CPU time with other active traffic flows. Since all the threads are scheduled together, the CPU scheduler will ensure that each flow receive a fair share of the CPU time. After the active capsule is processed, all the active capsules generated from the processing will be put in an output queue. For each active traffic flow, there is a corresponding output queue. These output queues, together with all the output queues of the traffic shapers for non-active guaranteed service traffic flows and non-active best-effort traffic, are linked to the output scheduler.

2.2.3 Active Transport Protocol (ACTP)

A new transport protocol called ACTP is introduced to isolate the interactions (possibly very complex) among active traffic flows and other traffic flows based on other transport protocols such as TCP or UDP. ACTP not only reduces the complexity of traffic and resource management tasks but also simplifies the programming interface for active network programming by providing APIs similar to BSD-sockets.

Active Transport Protocol is a transport protocol on top of the IP protocol. ACTP provides datagram delivery similar to UDP, but with some extensions such as built-in fragmentation scheme and security mechanism to better support the active network traffic. ACTP supports the concept of source/destination port similar to those in TCP or UDP to differentiate different sessions. Since the behavior of a transport protocol can greatly influence the network performance, care must be taken when new functionality are added.

ACTP supports built-in security mechanisms. For a secure ACTP session, the key management protocol for IP layer security is performed at ACTP layer. The authentication, encryption, and other security functions are performed at IP layer using IPSEC.

2.3 Suggestions for Improvement and Future Work

This paper presents an architecture to unify all three paradigms (active networks, Intserv, and Diffserv) together. However, the following issues must be addressed.

- The *correctness and efficiency* of the design should be validated through either analytic study or simulation experiments.

- *Admission control* is a fundamental problem in QoS. Research questions such as What is the proper admission control for ADNET? How does it cooperate with ADNET? etc., must be answered.

- In the experiments presented in this paper, the constant rate traffic without traffic control is used to compared with ACTP schemes. The impact of *end-to-end flow control* techniques such as TCP should be investigated to provide a more comprehensive evaluation of ADNET.

- The execution of mobile code introduces *overheads* to routers. The overhead should be quantified with the number of active traffic flows, the load of the active traffic, etc., so that the cost/benifit ratio of the deployment of ADNET can be estimated.