# Active Network –Based Services for Resource Discovery and Packet Filtering Policy Configuration

Xin Yao
(Submitted as part of the MRes requirements, supervisor Prof. C J Todd)
Department of Electronic & Electrical Engineering, University College London

**Abstract:** Active networks represent a new approach to the network architecture. This paper aims to offer a resource discovery and packet filtering setting services based on an active network. It mainly describes the design that helps understand and achieves the major purpose that a request packet from the trusted user could discover and configure the routers automatically. A test bed was built for the demonstration of the procedures.

## 1. Introduction

With the rapid development of both wired and wireless networks over the last decade, IP connectivity has become more flexible and convenient. Users can easily access the network through either cable or wireless connection [1]. Meanwhile, due to the security issues raised by the increasing number of users, there are always some intruders who want to hack the network system and turn the network computer to their own use. A packet filtering policy has been employed in the IP network management to protect the whole network from being attacked [2]. This allows the data from the privileged or trusted users to go through the network and permits those users to reach the network nodes while others including intruders cannot discover the resource of the network information at all. This work aims to make the packet filtering policies configuration more flexible based on the active network so that these policies on the intermediate routers can be automatically set up on demand for those trusted users whenever and wherever they connect to the network.

The word “trusted” which was mentioned means that such users can be recognized by the network and, during the connection, what they will do within the network is not only permitted but also anticipated by the network administrator. Therefore, if trusted users have the abilities to configure or re-configure the intermediate routers for specific purpose by sending a request packet, this will be much more convenient than the administrators setting up the routers individually. Furthermore, due to the trusted actions, the entire performance and configuration specified by that packet can be considered as acceptable ones that are safe for the whole network.

The recognition of the users should be based on a unique identity. Because of the current limitation of the IPv4 address mechanism, not everyone can hold a public IP address as their unique identity. Thus, for the cable users, the MAC address could be referred as the identity while GPRS technology could be involved with the wireless connection, which, in turn, makes the mobile phone number as a global unique identity. However, this paper mainly demonstrates the former rather than the latter.

## 2. Scenario Description

Suppose that a trusted user is going to plug his laptop into the network to talk to the remote server while, due to the security reason, all the intermediate routers have not been set up properly to accept the traffic from that user. As soon as the routers confirm the user identity, the entire configuration needed within them will be carried out automatically in response to the request packet. After that, the real traffic could run.

When that user has to move to a new place, obviously, the laptop will connect to the network through a new access point whose IP address is different from the one in the previous location. However, the configuration of the routers is for the previous IP address so that the traffic from this new address probably could not reach the server. By sending the request packet again, all the intermediate routers will, again, recognise this trusted user and re-configure themselves. Hence, that user could keep the link with the proper setting.

## 3. Design and Test

The section discusses the whole project design and the latest test result. It starts with the two working environments.

### 3.1 Environments

The working environments consist of the DINA (shown in Figure 1), which is an active platform [3] giving the router the ability to compute and execute programmes [4], and the AdventNet Simple Network Management Protocol (SNMP) environment [5] (shown in Figure2) for resource discovery purpose. The former contains three major components (Diverter, Session Broker and Information Broker) and the latter has two level APIs (High-Level API and Low-Level API).

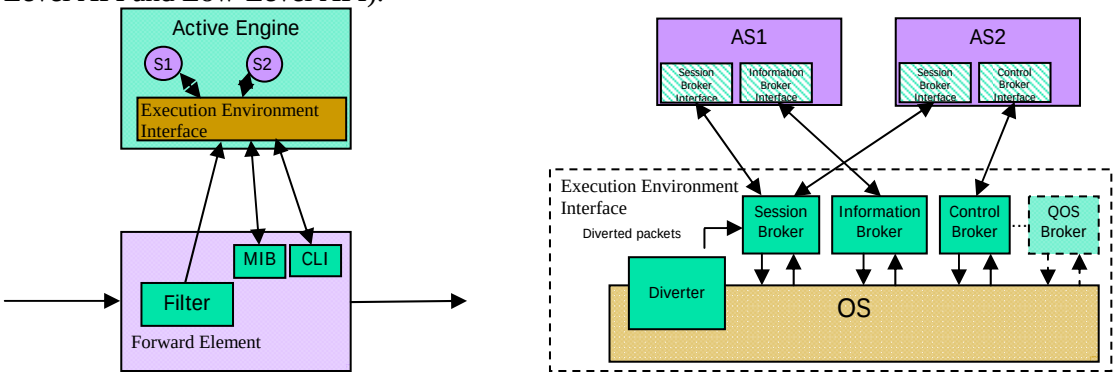


**Figure 1: Active Platform Environment [3]**

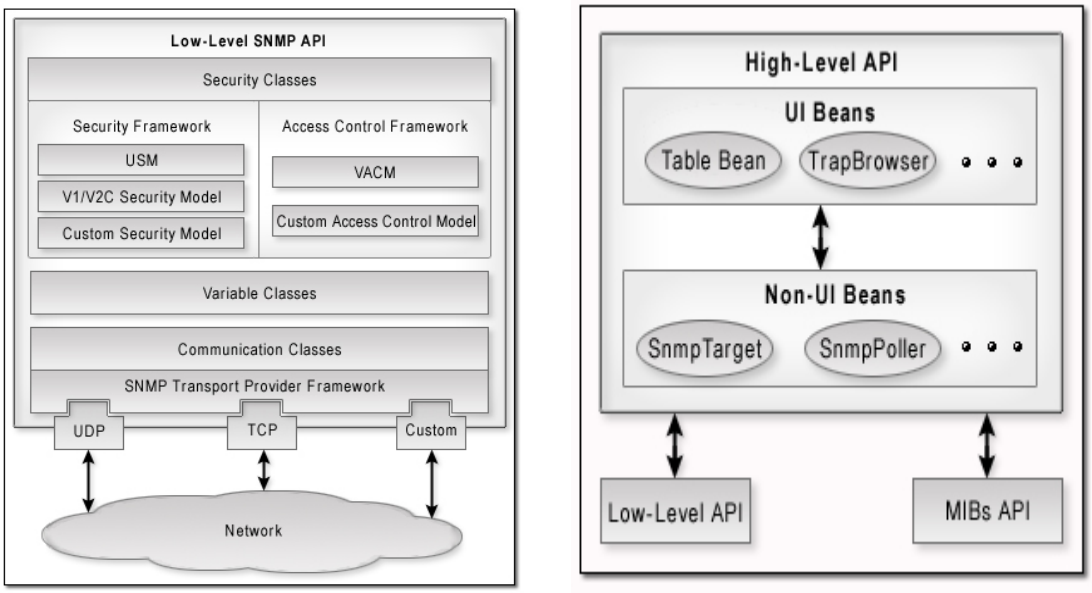


**Figure 2: AdventNet SNMP API Architecture [5]**

### 3.2 Test Bed

Figure 3 shows the test bed of the implementation. It contains five hosts. Three of them are the intermediate routers which are active-enabled. Correspondingly, they connect the

networks with address 10.1.1.0/28, 10.1.2.0/28, 10.1.3.0/28 and 10.1.2.0/28. On the rightmost, the laptop (called “shanghai”) is linked to the 10.1.1.0/28, who is trying to reach the server that is on the leftmost (named “leukada”). The request packet will launch from the laptop. On its way to the server, each active router will intercept it and execute the tasks specified before forwarding it on. That packet contains programme inside to achieve those tasks.

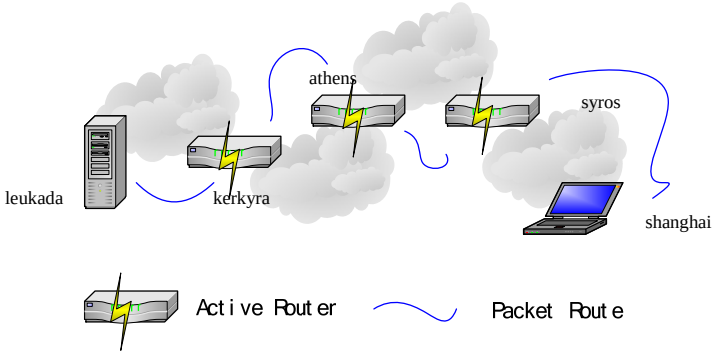


**Figure 3: the Test Bed of the implementation**

## 3.3 Tasks

There are two major tasks. One is the packet filtering policy setting; the other is the Resource Discovery which is normally done before setting the policies.

### 3.3.1 Resource Discovery

Since the policy setup or other router configuration needs some necessary information, the resource discovery plays a very important role in this paper. The necessary information is especially about the interface and IP routing, which can be obtained from the corresponding table content of a Management Information Base (MIB) [6] based on the SNMP. AdventNet SNMP API can be employed in the programme to discover the MIB.

### 3.3.2 Packet Filtering Policy

After fetching all the necessary information, the packet filtering policy will be set according to the user’s request, which allows the particular packet to go through the network; ideally, this process can be completed within the MIB. However, currently, there is no MIB for the packet filtering use. Therefore, in this paper, commands, such as “iptables”, are employed to achieve the same purpose.

## 3.4 Implementation

The procedure is shown in the following sequence diagram.

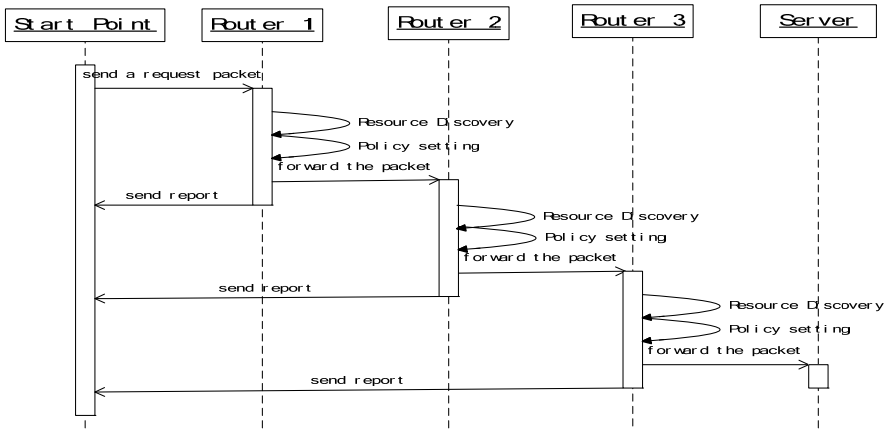


**Figure 4: Sequence Diagram of the packet programme**

After the host launching the request packet destined for the server, the packet will be caught by the first active router and then executed. After finishing the jobs specified by the active packet, the router will forward the packet. The next router will intercept and execute it again. This procedure will be repeated until the active packet reaches the destination.

The request packet (active packet) obtained the information shown in the following table. It checked the interface table and IP address table of the RFC1213-MIB to obtain some base and necessary information. After it compared the last hop egress interface address with the current interface addresses, the ingress and egress interfaces of the current router were identified. Based on the information, a packet-filtering rule was set within these intermediate routers by turns so that the traffic from the trusted user could arrive at the server.

| Router Name | Ingress | | | | | Egress | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Name | Num | IP Address | Mask | Net Address | Name | Num | IP address | Mask | Net address |
| syros | Eth1 | 3 | 10.1.1.1 | 255.255.255.240 | 10.1.1.0 | Eth0 | 2 | 10.1.3.2 | 255.255.255.240 | 10.1.3.0 |
| athens | Eth1 | 3 | 10.1.3.1 | 255.255.255.240 | 10.1.3.0 | Eth0 | 2 | 10.1.4.1 | 255.255.255.240 | 10.1.4.0 |
| kerkyra | Eth0 | 2 | 10.1.4.2 | 255.255.255.240 | 10.1.4.0 | Eth1 | 3 | 10.1.2.1 | 255.255.255.240 | 10.1.2.0 |

**Figure 5: Content obtained by the active packet**

## 4. Conclusion and Future Work

In conclusion, this paper provides a service which is based on an active network to make the router configuration automatically, according to the trusted user's request. It also demonstrated that most of the router information could be discovered by sending a request active packet. To get other specific contents, changes can be made in the programme correspondingly based on different SNMP object ID.

Ongoing work to establish the performance of this scheme in a test bed environment will be including in the conference presentation.

## References


[1] Sikora, A. (2003) *Wireless Personal and Local Area Network*, pp 1-14 West Sussex: John Wiley & Sons Ltd.

[2] Chapman, D.B., (1992) Network (In) Security of IP Packet Filtering, *Proceedings of the Third USENIX UNIX Security Symposium*; Baltimore, MD

[3] Cohen, R. & Raz, D. (ed.) (2003) *Active Platform Capabilities for Provisioning of Context Aware Services,* IST–2001–38142–CONTEXT

[4] Cheng, L., Galis, A., Eaves, W. & Gabrijelcic, D., (2003) *Strong Authentication for Active Network,* FAIN-10561 project

[5] AdventNet (2004) *Developing SNMPv3 Management Applications*, online at: http://www.adventnet.com/products/snmpapi4/help/snmpapi/snmpv3/

[6] Feit, S. (1995) *SNMP: A Guide to Network Management*, pp. 85-110 US: McGraw-Hill