Abstract: - The OSPF (Open Shortest Path First) protocol is interior routing protocol and uses Link State Routing to update the routing tables inside an AS (Autonomous System). Also, OSPF protocol supports a new class of link-state advertisement (LSA) called Opaque LSA. Opaque LSA provides a generalized mechanism to allow for the future extensibility of OSPF. This document proposes a method that active nodes can do a routing more dynamically in network situation that active nodes and IP nodes are mixed together in routing domain through extension of OSPF protocol.

Key-Words: - Active network, Active packet, Active node, OSPF, Opaque LSA

1 Introduction

Active network is designed to offer greater networking flexibility by providing the means to execute packet-carried programs in the network infrastructure[1]. This ability would mean that networks might be configured dynamically rather than requiring extended downtime, as is currently the case.

Ironically, the original released implementation of active network had a very simple routing algorithm based on static routing tables that had to be maintained at each router. Therefore, a change in the active network topology would require tearing down the whole network, altering the routing table files, and then restarting the network.

Also, at present network development process, all routers of network by technological and commerce reason cannot be consisted of active nodes.

Therefore, this document proposes a method that active nodes can do a routing more dynamically in network situation that active nodes and IP nodes are mixed together in routing domain through extension of OSPF protocol.

2 Routing Protocol Design

2.1 Opaque LSA

The OSPF (Open Shortest Path First) protocol is interior routing protocol and uses Link State Routing to update the routing tables inside an AS (Autonomous System)[2]. Also, OSPF protocol supports a new class of link-state advertisement (LSA) called Opaque LSA. Opaque LSA provides a generalized mechanism to allow for the future extensibility of OSPF[3].

Opaque LSA consists of a standard LSA header followed by application-specific information. This information contained in Opaque LSA may be used directly by OSPF or indirectly by some application wishing to distribute information throughout the OSPF domain. Like any other LSA, the Opaque LSA uses the link-state database distribution mechanism for flooding this information throughout the topology.

The link-state ID of the Opaque LSA is divided into an Opaque type field (the first 8 bits) and a type-specific ID (the remaining 24 bits). This document proposes the method of flooding the active node topology using Opaque type, Opaque ID and Opaque information field. Fig.1 shows Opaque LSA format.

![Fig.1 Opaque LSA](image-url)
Opaque types are maintained by the IANA. Extension to OSPF protocol should require and be allocated a new Opaque type. Opaque type values in the range of 0-127 are allocated through an IETF Consensus action and Opaque type values in the range of 128-255 are reserved for private and experimental use. Opaque types following the policies outlined in IANA are shown in Table 1.

<table>
<thead>
<tr>
<th>Value</th>
<th>Opaque Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Traffice Engineering LSA</td>
</tr>
<tr>
<td>2</td>
<td>Syncamore Optical Topology Descriptions</td>
</tr>
<tr>
<td>3</td>
<td>Grace-LSA</td>
</tr>
<tr>
<td>4-127</td>
<td>Unassigned</td>
</tr>
<tr>
<td>128-255</td>
<td>Reserved for private and experimental use</td>
</tr>
</tbody>
</table>

Table 1  Opaque Types

2.2 Flooding of Opaque LSA

If we are allocated an Opaque type for active network, active node can generate Opaque LSA and distribute the active node topology information to other active nodes throughout the OSPF domain. At this time, active node topology information can be stored in Opaque ID and Opaque Information field of Opaque LSA.

Fig.2 shows that active nodes flood Opaque LSA that has active node topology to other active nodes.

![Fig.2  flooding of Opaque LSA](image)

2.3 Construction of Routing Table

Active nodes can construct routing tables for transferring active packet by flooding Opaque LSAs that have active node topology.

When active nodes receive Opaque LSA that has active node topology, they calculate routing tables for transferring active packet using Dijkstra algorithm. The Dijkstra algorithm calculates the shortest path between two active nodes on network using a graph made up of nodes and edges.

The routing table for transferring active packet consists of Destination host and Neighbor host. Destination host becomes an ultimate destination host of active packet, and Neighbor host is next active node toward this destination.

Fig.3 shows active network topology and routing table at Active Node A.

![Fig.3  Active network topology and Routing table at Active node A](image)

2.4 An Example: Routing of Active Packet

Active packet consists of IP header, Active specific header and payload. Active specific header can be consisted of various forms according to execution environment of active node. In this paper, we suppose that active specific header has the field that describes address of an ultimate destination node of active packet. Payload contains the program to be executed in execution environment of active node. Fig.4 shows the Active Packet format.

![Fig.4  Active Packer format](image)

When active packet is transferred form Active Node A to Active Node D, Fig.5 shows the routing table that is referred at each active node and Fig.6 shows
the packet format that is generated at each active node.

(1) When Active Node A generates the active packet for transferring to Active Node D, it refers its routing table for transferring active packet and confirms that Neighbor host is Active Node B. Using this routing table information, Active Node A generates the active packet that destination address of IP header is Active Node B and transmits this packet to Active Node B. At this time, the destination address of active specific header is Active Node D.

(2) After this active packet arrives at Active Node B and is executed in execution environment of this node, Active Node B searches destination address of active specific header within this packet. Because an ultimate destination of this packet is Active Node D, Active Node B refers its routing table for transferring active packet and confirms that Neighbor host is Active Node D. Using this routing table information, Active Node B generate the active packet that destination address of IP header is Active Node D and transmits this packet to Active Node D. When this packet arrives at Active Node D, it is executed in execution environment in Active Node D. Because the destination address of active specific header within this packet is Active Node D, this packet becomes arrived at ultimate destination and the transmission of the packet is completed.

3 Conclusions and Future Work

This document proposed a method that active nodes can do a routing more dynamically in network situation that active nodes and IP nodes are mixed together in routing domain through extension of OSPF protocol.

For security, an OSPF router can authenticate received OSPF packets by requiring the sender to append keyed message digests to the OSPF packet. Stronger security has been proposed for OSPF by attaching digital signatures to OSPF LSAs. Also, to prevent of denial of service attacks resulting from repetitive origination of the same router advertisement or origination a large number of distinct advertisements resulting in database overflow, the frequency at which new LSA instances may be originated is set.

But, OSPF protocol is used by intra-domain routing protocol because of scalability. Therefore, we need to define a new protocol that can be used as inter-domain routing protocol such as BGP. Also, original released implementation of active node closely tied basic routing protocol to interpreter that exists in execution environment of active node. And active node reads its routing tables from a file upon startup. Therefore, the interpreter that exists in execution environment of active node must be separable from routing table to use routing method to present in this document.

References: