NETWORK TOPOLOGY DISCOVERY USING SNMP AND SWITCH CAM TABLE

[1]Nishant R B, [2]Mourya B S, [3]Sandesh S,[4]Sajjan L
Department of Computer Science and Engineering,
BMS College of Engineering, Bangalore, India

Abstract—large organizations usually contain huge LANs which consists of many remotely manageable network switches. It is a mammoth task to manage, maintain and troubleshoot nodes in such large networks. One of the key functions of network management is to identify and locate nodes in the network. So therefore, it is essential sometimes to isolate switch ports connected to trouble causing nodes for ensuring security in the network.

Index Terms—topology, network, MIB, nodes, MAC, switch, routing, SNMP, LAN, CAM

I. INTRODUCTION

A computer network is a telecommunication tool which enables computers to exchange data. Networked computing devices exchange data with each other using a data link. The connections between nodes can be established using either cable or wireless-media. Network computer devices that originate, route and terminate the data are called network nodes.

Computer networks differ in the transmission medium used to carry their signals, communications protocols to organize network traffic, the network's size, topology and organizational intent. Computer networks are used for enormous real world applications and services such as access to the World Wide Web, digital video, digital audio, shared use of computer peripherals.

Network topology is the arrangement of the various elements such as links and nodes in a computer network. It is the basic structure of a network which is depicted physically or logically. Physical topology is the placement of the various components, whereas logical topology illustrates how data flows within the network, regardless of its physical design. There are various types of network topology such point to point, bus, star, ring and mesh.

Topology discovery is the process of discovering and mapping network devices and links which is vital for managing network's efficiency. With the advent of virtualization and mobile computing, current networks often alter dynamically, and automatic topology discovery is essential for monitoring network state and also identifying bottlenecks and failures

II. OVERVIEW

A network consists of a multitude of devices, such as computers, mobile devices, printers and even switches and routers depending on the network level. A network may have multiple levels, in which each node maybe an entire network on its own. This is abundantly found in today's internet landscape, every household with a Wi-Fi router has a local area network (LAN). The Internet Service Provider (ISP) sees each house as a single node. However, the node is actually a combination of routers and end devices. This creates a two-level network and so on [2]. Such a scenario may take place in an Intranet too, like the ones used in large organizations such as Indian Space Research Organization (ISRO). In such an organization manually keeping track of each device and its location can be a daunting task for any network administrator. The number of devices can quickly overwhelm any experienced administrator, let alone a new network administrator. Hence, to monitor the network and maintain it, it is useful to know the arrangement of the various devices in the Intranet. This arrangement is what we call Network Topology. The topology of the network can be one of the fundamental ones mentioned earlier in the introduction or a suitable combination of some of them. When such many devices are part of the same network, the topology is often complex and visualizing it without any aid is extremely difficult. A suitable software solution can be implemented to discover this arrangement and a visual representation can be shown to the network admin. This will provide the network admin with a real time view of the network. In today's world we often use laptops and mobiles and connect via Wi-Fi. This makes networks very dynamic and the need for a software tool is even greater. The software solution can provide the network admin with various forms of information such as a visual representation of the nodes, the IP, Media Access Control (MAC) Address, availability of ports, etc. Thus, this will enable the network admin to have better control of the network, respond faster to failures, shut down rogue nodes and reduce his daily burden.

III. EXISTING SYSTEMS

Traceroute is a computer network diagnostic tool for displaying the route (path) and measuring transit delays of packets across an Internet Protocol (IP) network. The history of the route is recorded as the round-trip times of the packets received from each successive host (remote node) in the route (path); the sum of the mean times in each hop indicates the total time spent to establish the connection.

Ping or Packet Internet Gopher is a computer network administration software utility used to test the reachability of a host on an Internet Protocol (IP) network and to measure the round-trip time for messages sent from the originating host to a destination computer. Ping operates by sending Internet Control Message Protocol (ICMP) echo request packets to the target host and waiting for an ICMP response. In the process it measures the time from transmission to reception (round-trip time) and records any packet loss.

The Address Resolution Protocol (ARP) is a telecommunication protocol used for resolution of network layer addresses into link layer addresses, a critical function in multiple-access networks. ARP was defined by RFC 826 in 1982. It is Internet Standard STD 37. It is also the name of the program for manipulating these addresses in most operating systems. ARP is used to convert a network address (e.g. an IPv4 address) to a physical address such as an Ethernet address (also known as a MAC address).

IV. LITERATURE SURVEY

In the modern enterprise network, not only do we have to deal with physical network topology, but also logical network topology such as subnets and Virtual Local Area Networks (VLANs).Knowledge of the underlying network helps a network administrator perform root

cause analysis during failures, check for traffic bottlenecks, find failed components and also perform basic resource planning [1]. Network topology discovery can come under either internet topology, overlay topology or link layer topology as proposed by Donner and Friedman [2]. Our work comes under link level or router level topology which is an intra-domain topology discovery task. Drawing a network topology map involves finding the physical topology, i.e. finding the peers which are connected to ports via a physical link and logical network topology which involves finding subnets and VLANS. This allows network visualization at different levels of abstraction [3, 4].Different techniques exist for discovering network topologies such as ICMP trace router [5], DNS [6],ARP [7] and Simple Network Management Protocol (SNMP) [8]. Using the SNMP approach, we query different management information base (MIB) modules using suitable object identifiers to retrieve multiple roperties of various network devices. This provides us enough data to help construct a simple, efficient method to draw the network map. The SNMP method can be combined with the trace route, ARP and ICMP methods to find the network topology [9]. Finding the L2 and L3topologies is not easy. L2 connections were previously found using ICMP spoofing. This method is not allowed on most networks today and cannot be used anymore [10]. Instead, switch port mapping can be used to discover L2 level topologies. Using SNMP queries we can easily query SNMP enabled devices such as network managed switches, network printers and end devices using relevant MIBs. RFC 1213 contains the sysServices, sysDescrand ip tables. The ip table holds IP addresses and MAC addresses stored in the ARP table of the device. The table also contains ipAddrTable and ipRouteTable which together help identify multiple IP addresses the same device may hold and next hops that can be reached via this device respectively. sysServices returns an integer, this when converted to its 7bit binary representation can be used to find the level in which it provides services. For example, if 78 is returned (1001110), the highest one bit is in the seventh position, hence is a L7 device [10, 11]. This RFC along with the Bridge MIB and Printer MIB helps in recursive device and device type discovery. Connectivity discovery be classified into four different categories: switch-switch, switch-router, router-router and router-end host.

I. Switch-switch connectivity can be discovered using Bridge MIB using the dot1dTp group of the MIB. The interface group if mandatory and can be found on the bridge MIB, it uniquely identifies each port and its associated interface.

II. Switch-router connectivity can also be discovered using the Bridge MIB. If the router supports it, we can find the interface to which it is connected. First a list of L2 and L3devices are collected and then a mapping between them is found using dot1dTpFdbPortand ifIndex.

III. Router-router connectivity is discovered using the routing table entries using the ipRouteNextHop field in RFC 1213.

IV. Router-end host connectivity is discovered using subnet information and spanning tree information of L2 devices. The subnet information collected is then used to group devices belonging to the same subnet using ipNetToMediaNetAddress[9].Spanning tree algorithm is used along with the dot1dStp group to find the bridge's state. Some devices may not respond to SNMP requests and the ICMP address-mask should be used for such devices. The spanning tree algorithm is described as follows. Each bridge in the network elect one of the members as their root bridge. Then, each bridge determines its distance to the root bridge, it selects the closest port on the root bridge as its root port. The bridge now elects one port on each subnet, this is called the designated port. This is connected to the designated bridge of each switch and can reach the edge node of the network. The dot1dStp MIB helps with the discovery of the designated bridge. Once these edges are found, the MAC addresses of the hosts in the AFT table of each interface allows us to map the interface to the edge node.

Logical connectivity discovery is an essential part of topology discovery as mentioned previously. Logical networks deal subnets and VLAN and their relationship with the physical topology. IpAdEntNetMask MIB helps in the discovery of subnet devices. The subnets of all the devices are first obtained and then the devices are grouped together according to their subnet. Inter-subnet connectivity can be established by finding devices which are part of more than one subnet. These devices are usually routers. VLANs or Virtual Local Area Networks are major components of modern networks. Each VLAN is identified using a unique identifier. Every packet forwarded to a VLAN has the identifier appended to it. Some bridges can identify VLAN tagged frames and use this while forwarding packets. Each VLAN can have its own spanning tree and have nested VLANs also. Most component manufacturers provide their own protocols for finding such information. Such as Cisco providing the CISCO-VTP-MIB for its devices and uses community string indexing [12].A router has many ports and each port has a corresponding IP address. During topology discovery the different addresses and ports maybe seen as different router and this is called the router multi address problem [13]. One of the solutions to this problem is to choose the largest IP address as the router ID.
A router queue is created and each router found is added to this queue. Each time a new IP address is encountered, the router queue is checked. If it is found it means the router has already been found, otherwise it is added to the list of newly found routers

V. METHODOLOGY

During the development of our application, we have used switches to discover the network. Using RFC 1213, we have queried specific OIDs as responses from various agents present on the network. The ipToMediaNetTable table maintains ipToMediaNetPhysAddress to obtain MAC addresses and ipToMediaNetAddress to obtain IP addresses connected to the switch.

When a node is discovered, using the unique values of IP addresses on the ipToMediaNetAddress, which leads to the discovery of new nodes. This process can be recursively done to discover the entire network. Some devices on the network may not be SNMP enabled. These devices do not respond to the queries. ICMP echo can be used to check if the device is alive and then display basic information about the device. A device may have multiple IP addresses. These can be obtained using ipAdEntAddr object of ipAddrTable.

During the development of our application, we have used switches to discover the network. Using RFC 1213, we have queried specific OIDs as responses from various agents present on the network.

The ipToMediaNetTable table maintains ipToMediaNetPhysAddress to obtain MAC addresses and ipToMediaNetAddress to obtain IP addresses connected to the switch.

When a node is discovered, using the unique values of IP addresses on the ipToMediaNetAddress, which leads to the discovery of new nodes. This process can be recursively done to discover the entire network.

Some devices on the network may not be SNMP enabled. These devices do not respond to the queries. ICMP echo can be used to check if the device is alive and then display basic information about the device. A device may have multiple IP addresses. These can be obtained using ipAdEntAddr object of ipAddrTable.

1. Set of switch IP address, discovering devices through each switch.
2. ipNetToMediaNetAddress (1.3.6.1.2.1.4.22.1.3)
    a) for each switch, get all the IP addresses of devices from ipNetToMediaNetAddress
    b) if there is no ipNetToMediaNetAddress, then return.
    c) Call ipNetToMediaNetAddress recursively for all the switch IP addresses.

3. ipNetToMediaPhysAddress (1.3.6.1.2.1.4.22.1.2)
    d) for each switch, get all MAC addresses from

Figure 1. Device Discovery Algorithm

Once all the nodes are discovered using the above algorithm, it is required to know the type of all the devices discovered. This can be done using the sysServices object and convert the value into a seven-bit string. Each bit corresponds to the 7 layers of the OSI model. If a device has the value as 6(0000110), it provides services to layer 2 and layer 3. Hence it is a L3 switch. The algorithm for device discovery is given below.

I. Discovering input switch type.
    1. For each switch given as input sysServices (1.3.6.1.2.1.1.7) object is used.
    2. Convert sysServices object into a 7Bit string.
    3. Type of switch cab be obtained based on the bits enabled.
    4. Repeat
II. Discovered Device type.
    1. Check if device supports printer MIB(1.3.6.1.2.1.43.5.1.1.1.1) then return printer.
    2. Else check if device supports bridge MIB, then return switch.
    3. Else return workstation.
    4. Repeat for all devices.

Figure 2. Device Type Discovery Algorithm

Once the types of the devices are discovered, interface to interface connectivity discovery is done which helps in obtaining the topology of the network. This helps us establish the connectivity between devices in the network. Using bridge MIB objects dot1dTbFdbAddress to find MAC addresses of all the devices connected to the switch. The values obtained using ipNetToMediaPhysAddress are mapped to the aforementioned MAC addresses. The IP address corresponding to the MAC addresses are obtained using ipNetToMediaNetAddress, and the port numbers they are connected to are obtained using dot1dFdbport bridge MIB object. Therefore, we can obtain the switches connected, if any, to the ports of switches in the layers above. The following algorithm describes how to obtain the information required to establish the connectivity between devices.

For each switch, to discover the interface to which the devices are connected

e) Get the set of MAC addresses from the switch from dot1dTpFdbAddress.
f) Get the set of port numbers from the switch from dot1dTbFdbport.
g) Get the set of MAC addresses of the switch from ipNetToMediaPhysAddress.
h) Get the set of IP addresses of the switch from ipNetToMediaNetAddress.
i) Repeat.

Figure 3. Mapping Algorithm

High level system architecture can help better understand the structure and flow of the application. The following diagram depicts the system architecture in an abstract form.

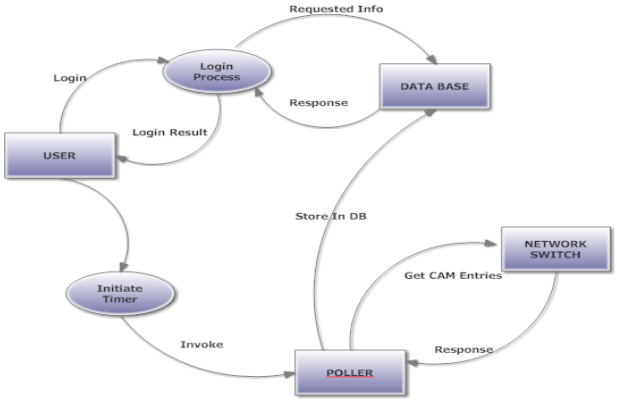
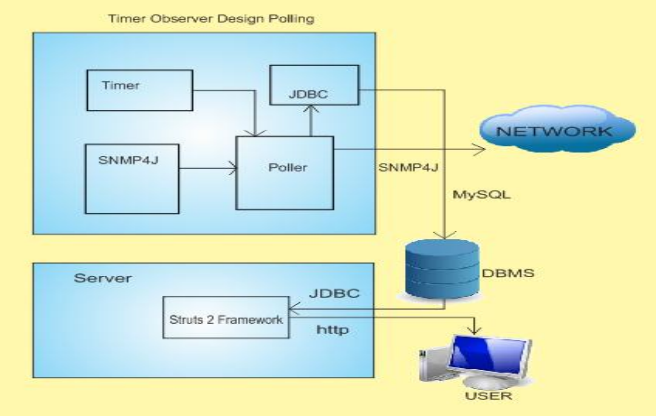


Figure 4. Dataflow Diagram Figure



5. System Architecture

The components mentioned in the abstract can be understood with a brief explanation given below.

- USER INTERFACE**:** The user interacts with the system through the browser. The required information is retrieved from the database using JSP and displayed on the browser window.
- TIMER**:** The timer is a program which keeps track of the intervals at which the polling program should be executed. Hence after a given time interval, the program is executed, which monitors the status of the LAN.
- PROGRAM**:** This program interacts with the SNMP manager, which intern interacts with the SNMP agents of the devices in the network and delivers a report on the status of all/required node(s) in the LAN. This information is updated onto the database. The program is known as poller.
- DATABASE: We have chosen MySQL relational database. The database is used to store the information provided by the MiB objects. This information can be data such as port addresses, MAC addresses sand IP addresses (CAM table values). The algorithm used in the program for the application queries and updates the information as new devices are added or removed.
- SERVER: We have chosen the Apache Tomcat container. It is an open source Java Servlet container which implements several Java EE specifications such as JSP, Java Servlet, and more in a "pure Java" Http web server environment where java code can run.

VI. ACKNOWLEDGMENT

We would like to thank Indian Space Research Organization for providing us the resources and also thank BMS College of Engineering for providing us the valuable opportunity to undertake the project. We would also like to thank our guides Dr Kayarvizhy and Mr Guromoorthy for their valuable guidance.

VII. REFERENCES

[1] Carofalakis M, Rastogi R. Data mining meets network management: The Nemesis project. ACM SIGMOD International Workshop on Research Issues in Data Mining and Knowledge Discovery, May 2001.

[2] Donnet B, Friedman T. Internet topology discovery: a survey. IEEE Communications Surveys and Tutorials 2007; 9(4):56–69.

[3] Passmore D, Freeman J. The virtual LAN technology report.

[4] IEEE 802.1Q. IEEE Standard for Local and Metropolitan Area Networks: Virtual Bridge Local Area Networks, 1998.

[5] Torren M. tcptraceroute: a Traceroute implementation using TCP packets. UNIX man page (2001). http://michael.toren.net/code/tcptraceroute/ [12 June 2010].

[6] Mockapetris P. Domain names: concepts and facilities. IETF RFC 1034, November 1987.

[7] Plummer DC. An Ethernet address resolution protocol or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware. IETF RFC 826, November 1982.

[8] Case J, Fedor M, Schoffstall M, Davin J. A Simple Network Management Protocol (SNMP). IETF RFC-1157, May 1990.

[9] Siamwalla R, Sharma R, Keshav S. Discovering internet topology. Technical report, Cornell University, May 1999.

[10] Breitbart Y, Garofalakis M, Jai B, Martin C, Rastogi R, Silberschatz A. Topology discovery in heterogeneous IP networks:theNetInventory system. IEEE/ACM Transactions on Networking 2004; 12(3): 401–414.

[11] Nazir F, Tarar TH, Javed F, Suguri H, Ahmad HF, Ali A. Constella: a complete IP network topology discovery solution. In APNOMS 2007, Sapporo, Hokkaido, Japan, October 2007; 425–436.

[12] Cisco. How to get dynamic CAM entries (CAM table) for catalyst switches using SNMP. http://www.cisco.com/en/US/tech/tk648/tk362/technologies_tech_note09186a0080094a9b.shtml [12 June 2010].

[13] Lantao You, Qiaoming Zhu, Peifeng Li. Design and Implementation of a Fast Network Topology Discovery Algorithm, Computer Applications and Software, 2007, 24(9): pp181-183

[14] Xiaoping Li, Yinxing Li, Jinghui Chen, Qiong Xu. The Study of Network Layer Topology Discovery Algorithm for Optimization Problem Based on SNMP.