ISSN: 2321-2152

IJMECE

International Journal of modern electronics and communication engineering

E-Mail
editor.ijmece@gmail.com
editor@ijmece.com

www.ijmece.com

ISSN 2321-2152
www.ijmece.com
Vol 7, Issue 3, 2019

Switch Port Mapper Network Discovery Tool for Networking Applications

Tata Jagannadha Swamy
Professor
Department of Electronics and Communication Engineering
Gokaraju Rangaraju Institute of Engineering and Technology (GRIET),
Bachupally, Hyderabad, India
Jagan.tata@griet.ac.in

***Abstract*—**The Switch Port Mapper can discover the ports on any manageable Cisco Catalyst Switch and detail the devices connected to those ports by MAC address, IP address and hostname. This tool eliminates tracing of LAN cables to determine the physical location of devices and makes the network documentation easy. The port mapping is done by discovering and correlating Port/MAC/Interface information from the switch. The MAC/IP address information is discovered from a Router that is directly connected to the same subnet as the switch. SNMP (Simple Network Management Protocol), Perl language and HTML/JavaScript are used to develop this project. SNMP protocol collects the required Port/MAC/Interface information from the switch; Perl is used to run these SNMP queries and HTML/Javascript to display the results. Since this is a real-time discovery you can view the operational status and port speed of each port. The Output can be exported to Microsoft Excel® or mySQL database.

***Index Terms*—Switch Port Mapper, Cisco Catalyst Switch, SNMP, mySQL, MAC/IP address.**

I. INTRODUCTION

In a switched network environment where hundreds of PCs get connected to tens of Layer-2 switches, Network Administrator often finds it difficult to trace a PC using its IP or MAC address. To get the info about which port the PC is on, he has to login to each and every Switch and run few commands to determine the port. This will be a time-consuming process if the number of switches increases beyond five. Switch Port Mapper does the required mapping within seconds. It maintains the database of the Switched Network by querying the information from the Switches and Routers at periodic intervals and provides a user-friendly interface to fetch and display the data at the click of the mouse. The Cisco Catalyst 3550 Series Switch is a stackable, multilayer switch that provides high availability, quality of service (QoS), and security to enhance network operations. With a range of Fast Ethernet and Gigabit Ethernet configurations, the Cisco Catalyst 3550 Series is a powerful option for enterprise and metro access applications. The Catalyst 3550 switch provides 48 10/100Mbps Fast Ethernet ports and 2 GBIC-based Gigabit Ethernet ports. The 10/100Mbps Fast Ethernet ports are used to connect PCs and Gigabit ports for interconnecting switches. These switches are SNMP enabled, meaning they understand SNMP protocol. “Switch Port Mapper” uses Perl’s Net-SNMP module and its related methods to query the data from these switches[1], [2].

ISSN 2321-2152
www.ijmece.com
Vol 7, Issue 3, 2019

Understanding SNMP: SNMP is based on the manager/agent model consisting of a manager, an agent, a database of management information, managed objects and the network protocol. The manager provides the interface between the human network manager and the management system. The agent provides the interface between the manager and the physical device(s) being managed (see the illustration below).

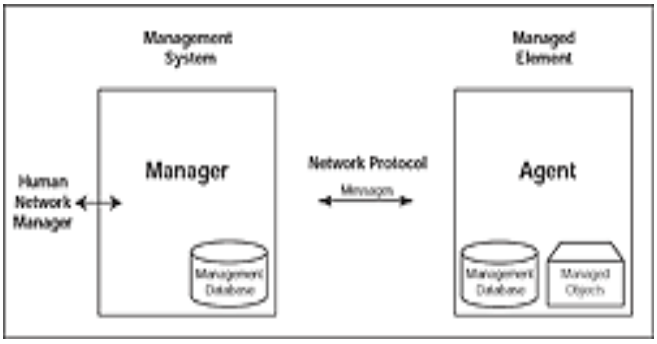


Figure 1.1: SNMP protocol

II. System Model for Design

Perl and its Modules:

Perl is an interpreted programming language known for its power and flexibility. It combines the familiar syntax of C, C++, grep, sh, and csh into a tool that is more powerful than the separate pieces used together. Among the best things with Perl is the huge number of freely available Perl modules. These modules contain pre-written Perl code that helps you complete your Perl scripts in a lot less time. A *module* provides a way to package Perl code for reuse[3]. Available modules include support for access to Oracle and other databases; networking protocols such as HTTP (Web), POP3 (email), SNMP and FTP (file transfers); and special Win32 modules for access to the Windows 95 or NT operating systems. Many modules support object-oriented concepts. *Switch Port Mapper* uses Net-SNMP module to fetch the information from the network devices and DBI module to store the data in MySQL database server[4].

Net-SNMP Module:

The Net-SNMP module implements an object-oriented interface to the Simple Network Management Protocol. Perl applications use the module to retrieve or update information on a remote host using the SNMP protocol. The module supports SNMP version-1, SNMP version-2c (Community-Based SNMPv2), and SNMP version-3. The Net-SNMP module abstracts the intricate details of the Simple Network Management Protocol by providing a high level-programming interface to the protocol. Each Net-SNMP object provides a one-to-one mapping between a Perl object and a remote SNMP agent or manager. Once an object is created, it can be used to perform the basic protocol exchange actions defined by SNMP. This module has certain methods, which the tool uses to achieve desired results.

III. Design Considerations

Apache and PHP:

As mentioned earlier, *Switch Port Mapper* uses DBI module to store the data in the Mysql DB server and that ends the first half of our project. The second half deals with using client software to select the data from the Database server and display it in a client readable format. Client being used depends on how

ISSN 2321-2152
www.ijmece.com
Vol 7, Issue 3, 2019

the front-end is designed. The front-end is a HTML file with some Java-script embedded in it. HTML stands for *HyperText Markup Language* and web pages are written in this language. Web pages are files stored on computers called ***Web servers***.

Computers reading these Web pages use client software called a ***Web browser***. Most popular browsers are ***Internet Explorer and Netscape Navigator***. In this project, ***Apache*** is used as a Web Server to store the HTML pages and ***Internet Explorer*** to view the pages. Our project requires the HTML files to do more than just display static code on a browser. They should display the dynamic content selected from the database server. So, we add extra strength to HTML by using PHP. Let's learn more about Apache and PHP.

PHP (recursive acronym for "PHP: Hypertext Preprocessor") is a widely used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. PHP is mainly focused on server-side scripting, so you can do anything such as collect data from database server, generate dynamic page content, or send and receive cookies. For many people, the main reason for using a scripting language like PHP is because of the interaction with databases it can offer. In this project we will show you how to use PHP and the MySQL database to store information on the web and include it into your website. You only really need three things to run PHP scripts, which access MySQL databases. Firstly, you will, of course, need a webserver. This can either be on a computer of your own or on a web host. Any web server software should work with PHP and MySQL but the best to use is Apache, which is free[5].

Apache is a well-known web server. It listens on TCP port 80. Apache and PHP are separate modules and hence we configure Apache to be compatible with PHP. This requires to edit *httpd.conf* file present in /Apache/conf directory and restart *apache*. This basically loads the PHP module and gives the web server an ability to run PHP scripts invoked by the Client, i.e., web browser. PHP also needs to be installed on the server. It can be downloaded from PHP.net and is also free[6]. In short, *web browser* at the Client machine connects to the *web server*, which loads the *index page* with links to other pages referencing PHP scripts. By clicking on these links the client invokes PHP scripts at the server-side, which contains required methods and functions to connect and retrieve data from the Mysql server. PHP fetches the data and displays it on the browser.

IV. IMPLEMENTATION ISSUES

Front-end:

In the front-end process, the Client program sends requests to Apache web server and invokes the PHP scripts, which initiates connection to MySQL server to run SQL commands to collect data. The requested data obtained from the database server is passed back to the client in a predetermined format.

Interface program:

ISSN 2321-2152
www.ijmece.com
Vol 7, Issue 3, 2019

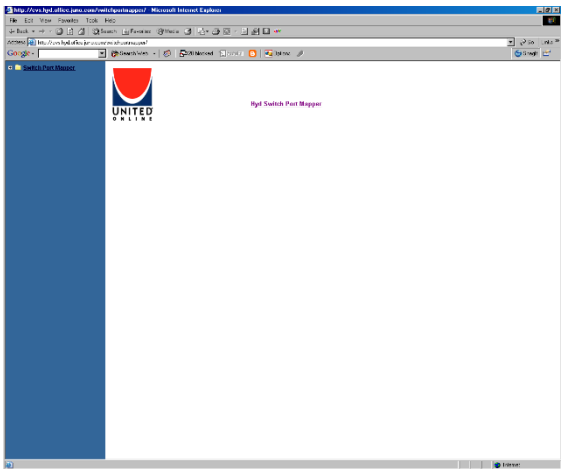
Once the data is in the database, we need a program that acts as an interface between the client and the server to retrieve the data as and when required. The interface should be simple and user-friendly.

Switch Port Mapper has used Javascript program to build this interface. This program is a freeware downloaded from Internet. We made some changes to the code to use it in the project.

Basically, we include this Javascript code in a HTML file that resides on web server's **/root/switchportmapper** directory[7].

Whenever we connect to **http:/webservername/switchportmapper**, this file gets loaded onto web browser (used as client program).

V. Results and Discussions



Main Page

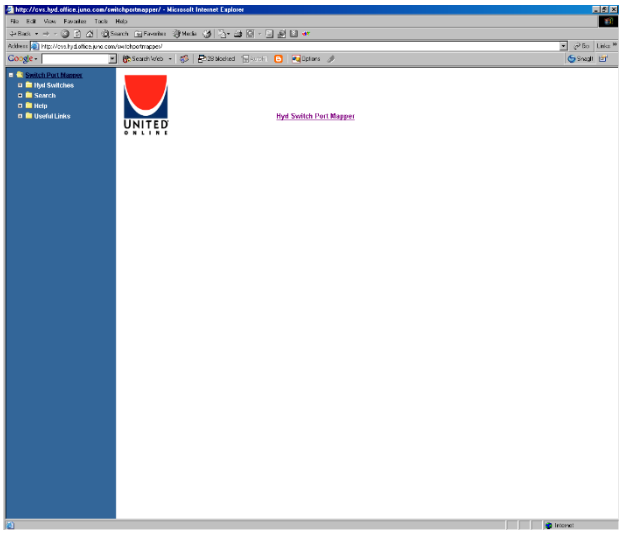
This is the page that gets loaded into the browser when we connect to Switch Port Mapper main page.

The frame on the left displays the folder tree view built using Javascript, which is a freeware available on Internet. On the left frame, click on the '+' sign to display the folders under the Root folder 'Switch Port Mapper'.

The Root folder 'Switch Port Mapper" has the following folders:

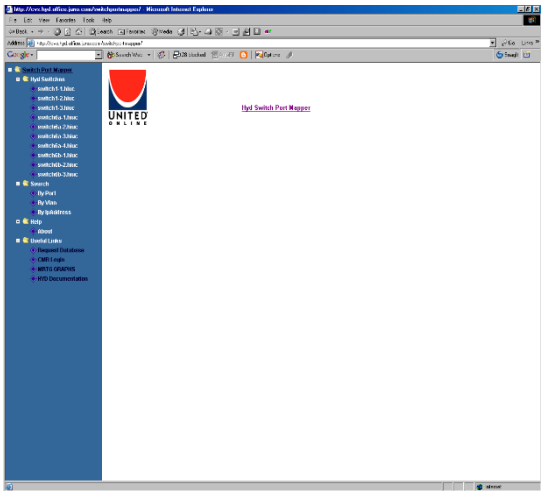
1) Hyd Switches -> Individual Switch info is displayed
2) Search -> Provides options to display data
3) Help -> Documentation
4) Useful Links -> Links useful to our organization.

ISSN 2321-2152
www.ijmece.com
Vol 7, Issue 3, 2019



Switch Port Mapper root folder

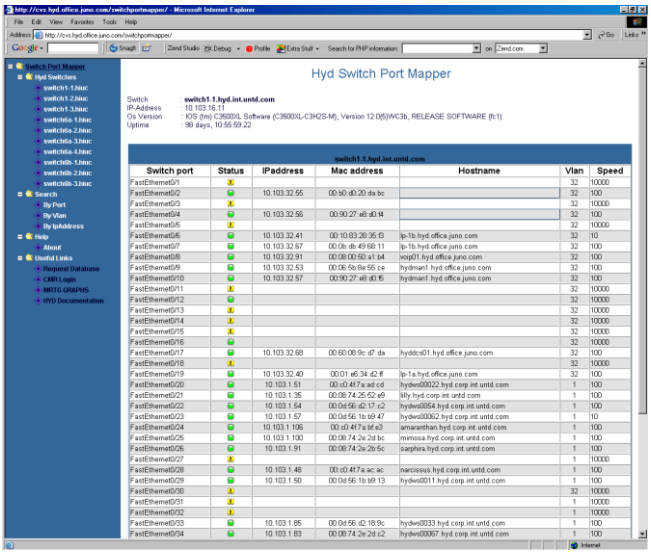
Click on the '+' sign beside these folders to display sub-folders under them.



Hyd Switches

Under Hyd Switches, we have the sub-folders each representing a switch. Clicking on the names invokes a PHP script on apache server that connects to the database server and fetches the information specific to that switch. The screenshot below shows the output displayed when the user selects the first sub-branch under 'Hyd Switches'.

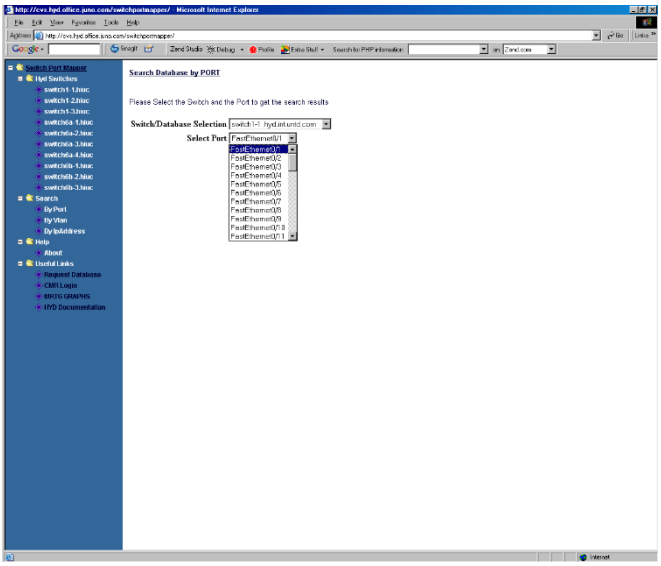
ISSN 2321-2152
www.ijmece.com
Vol 7, Issue 3, 2019



Switch Configuration Page

It shows that the information specific to each port, its status info, IP address of the device connected to it, its mac address, the hostname and the vlan info. This gives the complete picture of the switch and its busy ports. Similar pages get displayed, but with some changes specific to each switch configuration, by clicking on the other switch names. So far we have seen that the tool can be used to get the complete configuration details of a switch just by clicking on its name. Sometimes it is required to search for specific info like to what switch-port a device is connected given the IP address. Focusing on such requirements we have come up with some options, which are included under Search folder. Search has the following sub-folders:

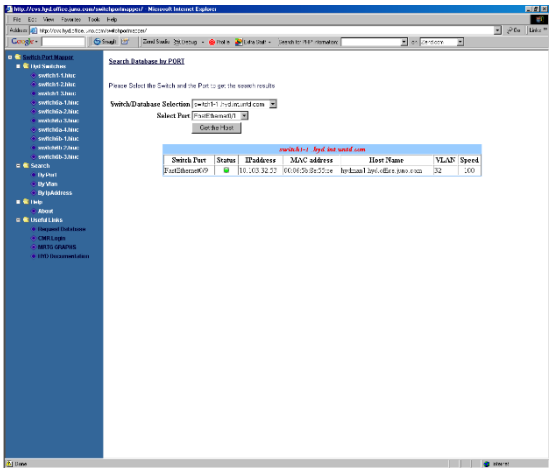
1) By Port
2) By VLAN
3) By IP-address



Search by Port Page

ISSN 2321-2152
www.ijmece.com
Vol 7, Issue 3, 2019

The screenshot displays the output when user intends to search the database based on port.The page displays two drop-down boxes, one with switch names and the other with port numbers. User selects the switch name and then any port to get the details of it. This operation is shown in the next screenshot.



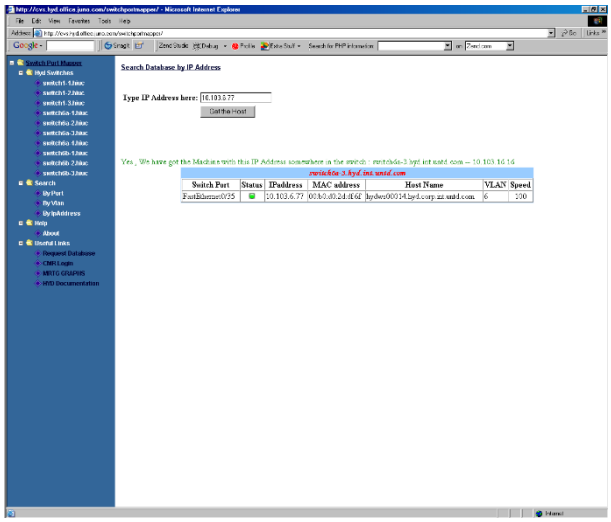
Search by Port results

On selecting switch ‘switch1-1.hyd.int.untd.com’ and port ‘FastEthernet0/1’ and clicking on the button ‘Get the Host’ will actually send the request to a PHP script, passing the switch name and port number as arguments. The Script connects to database server and queries the information and displays the data as shown above.

Next option search by IP address displays the below page that prompts the user to enter the IP address of the device whose switch port information needs to be queried.

User enters the IP address in the text box and clicks on the button ‘Get the Host’. This action invokes the PHP script on the server and passes the IP address as the argument to the script. Using this argument PHP connects to the database server and searches the tables for switch port details and displays the information regarding the name of the switch and the port.

ISSN 2321-2152
www.ijmece.com
Vol 7, Issue 3, 2019



Search by IP results Page

VI. CONCLUSION

In corporate offices where hundreds of computers are interconnected using switches, Switch Port Mapper greatly helps in tracing the computers affected by virus within seconds thereby reducing the damage to almost zero.

[1] Hajjaj, Sami Salama Hussen, and Khairul Saleh Mohamed Sahari. "Establishing remote networks for ROS applications via Port Forwarding: A detailed tutorial." *International Journal of Advanced Robotic Systems* 14, no. 3 (2017): 1729881417703355.

[2] Tarnaras, G.; Haleplidis, E.; Denazis, S. SDN and ForCES based optimal network topology discovery. In Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft), London, UK, 13–17 April 2015; pp. 1–6.

[3] Rajaratnam, A.; Kadikar, R.; Prince, S.; Valarmathi, M. Software defined networks: Comparative analysis of topologies with ONOS. In Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking, Chennai, India, 22–24 March 2017; pp. 1377–1381.

[4] Bah, M.T.; Del-Piccolo, V.; Bourguiba, M.; Haddadou, K. A centralized controller to improve fault tolerance in TRILL-based fabric networks. In Proceedings of the 2016 3rd Smart Cloud Networks & Systems (SCNS), Dubai, United Arab Emirates, 19–21 December 2016; pp. 1–6.

[5] Alharbi, T.; Portmann, M.; Pakzad, F. The (In)Security of Topology Discovery in Software Defined Networks. In Proceedings of the 2015 IEEE 40th Conference on Local Computer Networks, Clearwater Beach, FL, USA, 26–29 October 2015; pp. 502–505.

[6] Araniti, G.; Cosmas, J.; Iera, A.; Molinaro, A.; Morabito, R.; Orsino, A. OpenFlow over wireless networks: Performance analysis. *IEEE Int. Symp. Broadband Multimed. Syst. Broadcast. BMSB* **2014**.

[7] Aslan, M.; Matrawy, A. On the Impact of Network State Collection on the Performance of SDN Applications. *IEEE Commun. Lett.* **2016**, *20*, 5–8.