MJRET
Open Access
Multidisciplinary Journal of Research in Engineering and Technology
Journal homepage: www.mjret.in
ISSN:2348 - 6953

DISCOVERY OF NETWORK ELEMENTS AND RECONCILIATION (DNER)

Narendra Shewale[1], Sharad Hadke[2], Mayur Kharote[3]
Department of Computer Engineering, University of Pune
Imperial College Of Engineering and Research
Pune, India
[1]narendra.shewale4@gmail.com, [2]sharadhadke5@gmail.com,
[3]mayurkharote369@gmail.com

Abstract: *Designing of an efficient discovery module is a key challenge in a spread out network. It is necessary to keep records of each and every device in network for the company by which it can keep the details about the devices recorded in the network as per the status of presence or failure of that device. Discovery of Network Elements in the network is achieved through two modes of discovery that are through IP Range Specification i.e. Range Based Discovery, Device Based Discovery and Individual Discovery. In Range Based Discovery we have to give the range of IP addresses which are already present in the database, this will discover existing network elements and maintained their information. In Device Based Discovery manufacturer of network element is specified and IP addresses are fetched from the master database depending on the manufacturer and a SNMP request is send to those network elements. In Individual Discovery a single IP address is specified and the discovery of that network element is carried out. Reconciliation module contains two types of discrepancies which are depending upon the types of Discovery carried out as Range Based and Device Based. Discrepancies are nothing but the differences found by comparing the master database and discovered database.*

Keywords*- SNMP, NMS, Network Elements, Node Discovery, Reconciliation*

1. INTRODUCTION

Discovery of Network Elements (NEs) in the network is achieved through two modes of discovery that are through IP Range Specification i.e. Range Based Discovery. In Range Based Discovery we have to give the range of IP addresses which are already present in the database, this will discover existing network elements and maintained

their information. If we specify whole IP range of company network then it will scan all the devices present in the network as well as subnets in the networks. "Discovery of NEs & Reconciliation using SNMP" is used to scan the devices connected in the network. It is important to keep track of all the devices in the network because functioning of every device & to check whether they are alive is very important purpose. In case of existing system it is possible to run discovery but Reconciliation was not their & it was given to third party. This software gives facility to discover the nodes [2] as well as after completing discovery we can reconcile it. This will become very helpful for company & easier because no need of third party. Authorized person can perform the Discovery & Reconciliation very easily.

1.1 Existing system

In case of existing system the organization is using a tool called IBM Tivoli Net cool Configuration Manager which is used in the conjunction with the Bulk Loader utility which discovers the network elements in bulky manner and the results produce are not that efficient as per the organization standards. When new device get added to network then they manual entry it into their data & then next time when discovery will run that device will be identified. Existing system does not allow end user to reconcile the discovered information.

1.2 Problem statement

"Discovery of NEs using SNMP & Reconciliation" is based on concept of discovering nodes in network which can include switches, routers, hubs etc. Aim to get information about all those devices in network & keep records of each & every element. This is helpful for company to identify if any fault occurs in the process afterwards. Using SNMP i.e. Simple Network Management Protocol we will perform discovery of Network Elements.

2. PROPOSED SYSTEM

In this system we are providing Range based discovery, Individual discovery & Device based discovery of network elements from which Device Based Discovery is absent in existing system. This will be advantageous if any new device has been added will get discovered at the time of Range Based Discovery and gets added in the main inventory. This will save lot of time & will improve the efficiency of the system by reducing the efforts of manual entry.

Our system also has options by which the user can perform the selection for updating the database either through automatic update functions or it can be done by user manually after the discovery of network elements. Our system also provides the Individual Device Discovery by which the user can search any one specific standalone device by its IP address.

3. SYSTEM REQUIREMENT

3.1 Hardware Interfaces

For successful working of the discovery module the following elements are the important requirements:

1. Routers
2. Firewall
3. Hubs
4. Switches
5. Unstructured Network

3.2 Software Interfaces

Operating System: Windows, UNIX Environment
Language : JDK 6
Data Bases : MySQL
Front End : JSP, CSS, HTML
Browser : Internet Explorer 5& above

3.3 Java (JDK 6)

Java is a general purpose programming language with a number of features that make the language well suited for use on the World Wide Web. Small Java applications are called Java applets and can be downloaded from a Web server and run on your computer by a Java-compatible Web browser, such as Netscape Navigator or Microsoft Internet Explorer

3.4 Java Server Pages

Java Server Pages (JSP) technology enables Web developers and designers to rapidly develop and easily maintain, information-rich, dynamic Web pages that leverage existing business systems. As part of the Java technology family, JSP technology enables rapid development of Web-based applications that are platform independent.JSP technology separates the user interface from content generation, enabling designers to change the overall page layout without altering the underlying dynamic content.

3.5 MySql Server 5.1

MySQL is a popular choice of database for use in web applications. Many programming languages with language- specific APIs include libraries for accessing MySQL databases. MySQL is primarily an RDBMS and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools,[citation needed] or download MySQL front-ends from various parties that have developed desktop software and web applications to manage MySQL databases, build database structures, and work with data records.

3.6 Communication Interface

1. Keyboard
2. Mouse
3. Monitor
4. NIC

4. SYSTEM ARCHITECTURE

Our system mainly consist of two modules first is Discovery, it includes sub-modules as (Range Based Discovery, Device Based Discovery and Individual Discovery) and second one is Reconciliation

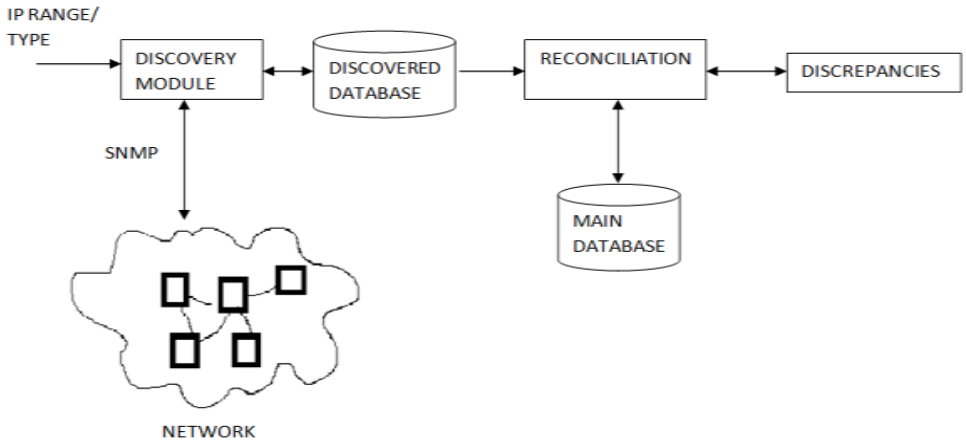

Fig. 4.1.System Architecture

In Range Based Discovery we specify the range of IP addresses and the IP addresses included in the range are scanned and SNMP request is send on which the network elements are present and the response is collected in discovered database which is used to find further discrepancies. In Device Based Discovery we specify the manufacturer of network element and the IP addresses are fetched from the master database depending on the manufacturer and a SNMP request is send to those network elements. In Individual Discovery a single IP address is specified by the user and the discovery of that network element is carried out for further processing.

Reconciliation module contents two types of discrepancies which are depending upon the types of Discovery carried out as Range Based and Device Based. Discrepancies are nothing but the differences found out by comparing the master database and discovered database.

4.1 Importance Discovery

This will be used by the company for management of the networks as well as keeping the track of the devices used in network. As our system also allows dynamic discovery of NEs, this will be helpful in error detection & recovery. The reconciliation module implement in our system is for the handling of the database perfectly. System gives the facility to view newly updated, added or old deleted nodes before the changes are reflected to the database. User can also check for the information after the discovery gets completed and he can either do the updating automatically or manually.

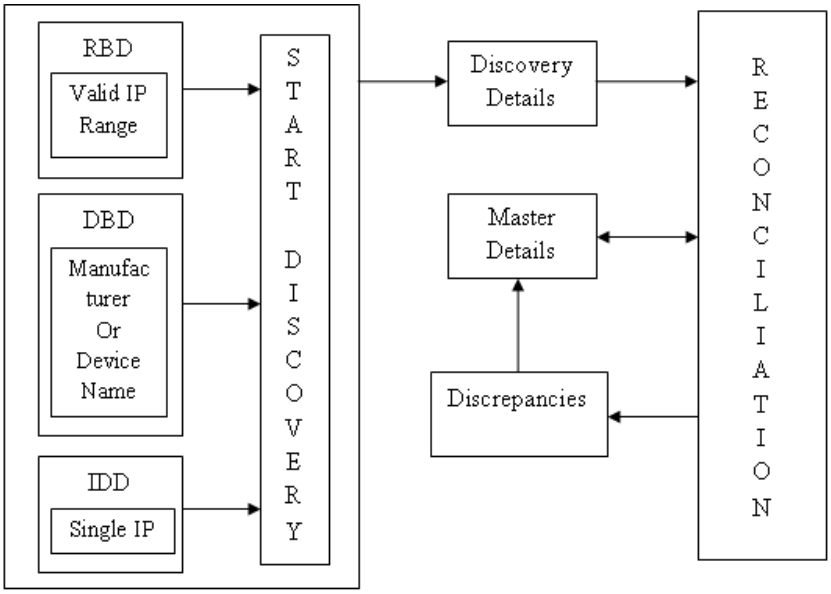

Fig.4.2.System Process

4.2 Workflow of System

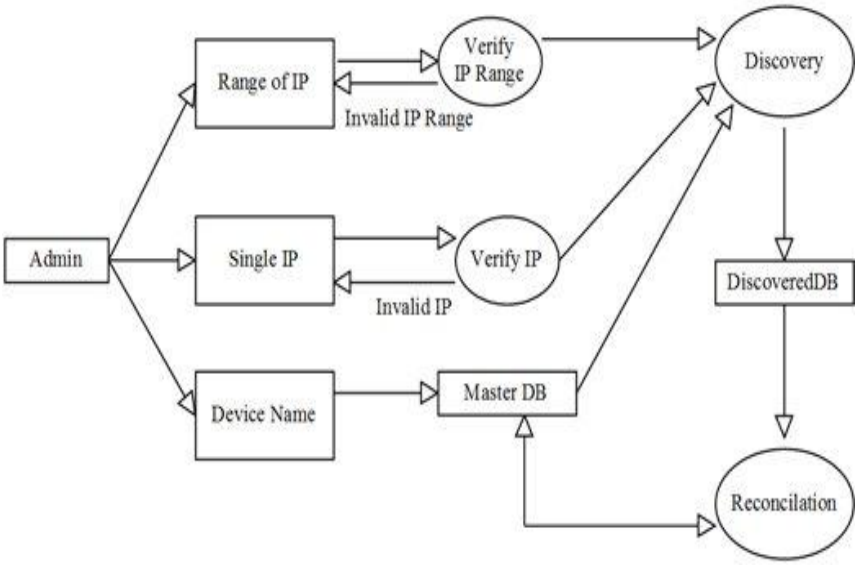

Fig. 4.3.Workflow of System

5. CONCLUSION & FUTURE SCOPE

In this document, Discovery algorithm is proposed, which overcomes the limitation of previous static method. It will significantly improve the performance. Discovery also helps in keeping track of the entire device in network. When combined with knowledge based search algorithms, its performance could be further improved. Our proposed algorithm relies on standard SNMP MIB information that is widely supported in modern IP Networks and is the SNMP-based discovery tool to offer strong completeness guarantees for recovering the true network information from the given MIB data.

ACKNOWLEGEMENT

We would sincerely like to thank Mrs. Sonali Tikde, our guide, for her guidelines and support.

REFERENCES

[1]. "Network Management – Principles and Practices" by Mani Subramanian
[2]. Kishori M. Konwara, DariuszKowalski , Alexander A. Shvartsman , "Node discovery in networks", 2008
[3]. Y. Bejerano, Y. Breitbart, M. Garofalakis, and R. Rastogi, "Physical topology discovery for large multi-subnet networks," in Proc. IEEE INFOCOM, 2003, pp. 342–352.
[4]. M. ShoaibYousaf ,AhmedMattin ,AhsanRazaSattar , "Real Time Network Monitoring System in LAN Environment", IJRIME Volume1Issue5 ISSN-2249- 1619,2011
[5]. SumanPandey, Mi-Jung Choi, Sung-Joo Lee, James W. Hong "IP Network Topology Discovery Using SNMP"
[6]. C. Perkins and E. Royer, "Ad-Hoc On-Demand Distance VectorRouting," Proc. IEEE Second Workshop Mobile Systems and Applications,1999.
[7]. S. Podlipnig and L. Boszormenyi, "A Survey of Web CacheReplacement Strategies," ACM Computing Surveys, vol. 35, pp. 374-398, 2003.