SNMP Based Discovery Tool on Raspberry pi 2

Ashwini R. Patil[1], Snehal A. Mandlik[2], Priyanka D. Khedkar[3], A. M. Pujari[4]

[1 2 3] *Student, Computer Department, Jaywantrao Sawant College of Engineering, Pune, Maharashtra, India*

[4] *Asst. Prof, Computer Department, Jaywantrao Sawant College of Engineering, Pune, Maharashtra, India*

ABSTRACT

In commercial industries, large scale computer networks are deployed. It is not feasible to manage these networks manually. Solution to this issue is Network Management System (NMS). However, the current NMS are too costly, complex to understand and needs to be configured by an operator level person. This paper proposes a Simple Network Management Protocol (SNMP)-based web application that handles automatic device discovery and devices monitoring for local network. This system is implemented on Single Board Computer (SBC). SBC used in this system is Raspberry Pi 2. Key contribution of this system is that it can be deployed at server side without making any major changes in existing server system. Main Purpose of using Raspberry Pi is that it has low investment and maintenance cost, affordable replacement and fulfills single operation requirement. It is Server based system, so it does not need to configure devices manually. Thus any number of hosts can be traced. In short, this web software along with hardware gives effective solution for managing and monitoring local networks.

Keyword: - *Network, SNMP, MIB, NMS, Raspberry pi*

1. INTRODUCTION:

Today network monitoring has become an essential tool for all organizations. Understanding network traffic can allow an organization to better utilize their resources. A newly joined inexperienced network administrator faces many difficulties due to the unavailability of a network tools for discovery and monitoring. Even for an experienced administrator, keeping track of devices and their connectivity details becomes a tedious task. Simple network management protocol (SNMP) is a protocol that widely used for network monitoring. SNMP protocol is application–layer protocol outlined by the net design Board (IAB) in RFC1157 for exchanging management information between network devices. It is a part of Transmission management Protocol ⁄ Internet Protocol (TCP⁄IP) protocol suite. Most of the professional–grade network parts comes with bundled SNMP agent. These agents need to be enabled and configured to communicate with the network management system (NMS) [2]. Due to the employment of SNMP protocol generates watching information within the type information, it’s necessary to use intermediate process therefore so as to form method watching become more efficient.[3]. The network management platform deployed in the network, management the infrastructure consisting the network elements. The platform receives the events from the network elements. The most common functions in the standard management platform are:

- Automatic Device Discovery
- Monitoring Devices in network
- Fault Detection
- Notification system

In this proposed system we will focus on network discovery and monitoring of devices in local network. Automatic device discovery finds out every device that is connected to a particular local network. It searches along the length and breadth of the network and finds out all the different types of connected devices like switches, routers, hubs and PCs. This will also take care of automatic configuration of new device added to the network. In addition to this it lists the basic characteristics of devices like uptime, operating system installed etc. to identify them on network. A single-board computer i.e. SBC is a complete computer in which a single circuit board comprises memory, input/output, a microprocessor and all other necessary features for functional computer. However, it does not rely on expansions for other functions. A single-board computer reduces the System’s overall cost as the number of circuit boards, connectors and driver circuits are all reduced [1]. Here SBC used is Raspberry pi 2. With the introduction of the Raspberry Pi, building a custom network monitor has become really inexpensive and accessible. Computer enthusiast and novices can now easily deploy network monitors to their networks. Thus, this project researches how to develop a very simple network monitor using a Raspberry Pi and Simple Network Management Protocol (SNMP) to query and monitor the health of all nodes on a private network.

2. SYSTEM DESCRIPTION:

2.1 Interface:

- Hardware Requirements: Raspberry Pi 2
- Software requirements: PHP 7, AngularJS 1.7

2.2 System Architecture :

In order to effectively monitor network activity, System relies on an architecture consisting the following elements:

- **Managed devices (Hosts):** Managed device is any device with SNMP Configuration within an organization’s network that have to be managed and monitored. SNMP Configuration allows them to communicate with management system.
- **SNMP Agent:** Overall SNMP management relies on a system of collected and transmitted local device information. This is carried out via agents. Agents are the programs that are tied to local devices with the purpose of collecting, storing, and signaling the presence of data from these environments.
- **SNMP Manager:** SNMP manager consists of management console through which queries are sent. Also it provides required the memory and processing functionality for network management.

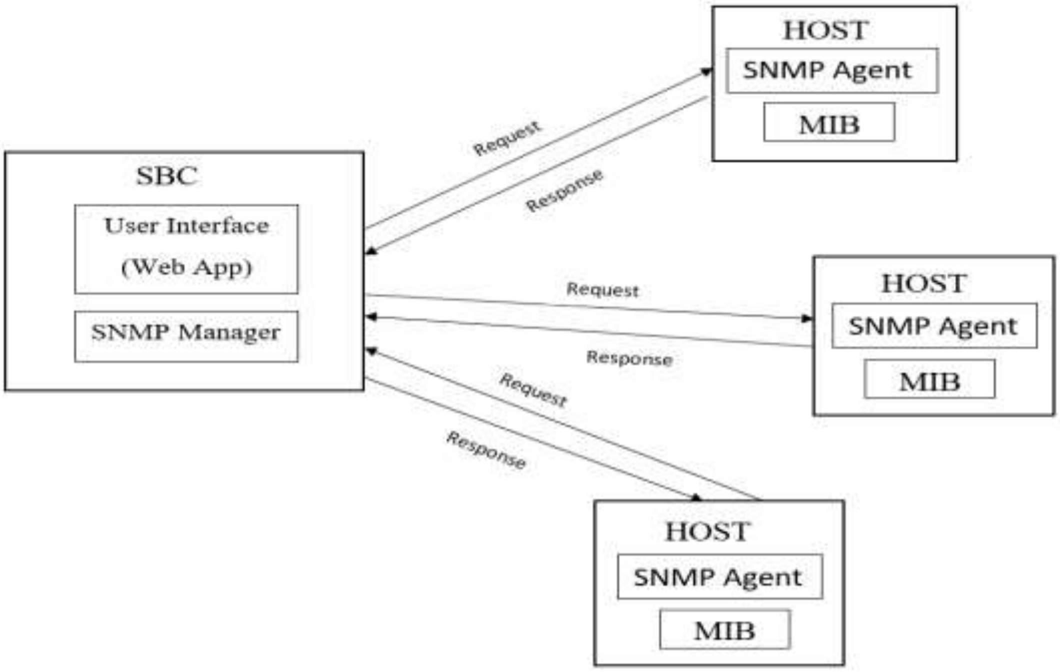


Fig - 1: System Architecture

2.3 System Working:

System working is carried out by sending messages which is called protocol data units (PDUs) between SNMP manager and agents. By the use of SNMP queries, the manager can identify and retrieve information of the devices by receiving the responses sent by the agent. First manager will send query to intended host. This query consist of SNMP Command, SNMP community string, OID. In response to the query, agent sends information regarding requested OID. Then SNMP manager will send the data to the monitoring tool i.e. web interface using HTTP. Upon receiving data from, web interface will display the data in appropriate format. In this way, the administrator can manage the devices in local network through SNMP control commands.

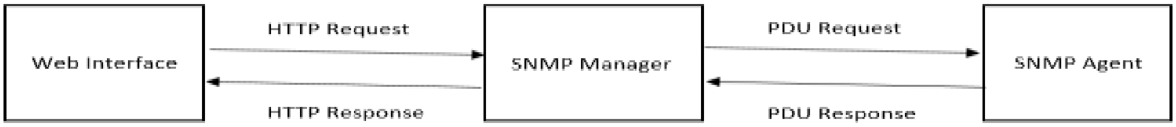


Fig -2: Communication between system elements.

2.4 System Implementation:

To be able to complete this project we used a few items. We had a Raspberry Pi 2. We installed Raspbian OS as the operating system for my Raspberry Pi. This OS was installed on a 16GB SD card as the Pi did not come with an OS pre-installed. The additional space on the SD card was used to store excess data or programs. Programmatically we used PHP 7, MySQL for the backend and HTML, AngulrJS for frontend. MySQL allowed for storage of all captured data from any SNMP call. This then allowed us to retrieve and manipulate this data in a frontend application. Because the data was securely store in MySQL We were able to keep track of host activity over a period of time. For the frontend we used PHP. The network was made up of Windows OS based hosts. The SNMP server was running on Raspberry Pi. To be able to query the Pi we also installed SNMP. Our network included several Windows machines. These machines also had to have SNMP enabled and configured. SNMP runs on port 161 across all agents. Thus our code scanned port 161 of each machine. The MIB hierarchy is really quite large[11]. Thus for this project we decided to focus on some items we thought would be of some importance to any network administrator. We decided to focus on host processor usage, bandwidth usage and memory usage. To begin the reason why we chose to focus on memory usage was we wanted to be able to ensure that each host was not overusing the RAM available to it. Our thinking was that an administrator would like to know this information as a way to figure out what may be causing a system to be slow. For example an agent may return very low values for free memory; thus indicating high memory usage, at boot-up. Thus they may take this as an indication that there are a lot of start-up processes consuming a lot of the available RAM. We focused on bandwidth usage as we thought an administrator would want to know what hosts are using the most bandwidth. Thus, in the future the monitor may allow for a shifting or limiting of bandwidth usage by host. Finally we wanted to focus on processor usage because a host that is using a lot of its processor would not run efficiently. Thus we thought this would be an important piece of information for a network administrator to know.

3. EVALUATION:

Contrary to the information provided by the Linux MIB to SNMP, Windows machines do not seem to provide as much information about processor usage. To be even able to query the MIB for processor usage data one needs to find the OID associated with the processor. This can be done by doing a SNMPWALK on the HOST-

RESOURCES-MIB subtree [4]. Under this subtree you can find system information such as time and date, storage information, device information and much more. Under the devices subtree you'll be able to find the processor description for the machine as well as descriptions of other devices associated with the machine. Windows for some reason chooses to have the MIB store information for each core of the processor thus it is important to note each instance of 'hrDevice Processor' in the device subtree. Particular attention must be paid to the last digit of each OID associated with 'hrDevice Processor'. This is due to the fact that this digit can be used to query for processor load in the 'hrProceessorEntity' subtree. For example upon completing an SNMPWALK on the HOST-RESOURCES-MIB there is a 'hrDeviceProcessor' at .1.3.6.1.2.1.2.2.1.11 and .1.3.6.1.2.1.2.2.1.12, then you can retrieve details from the 'hrProceessorEntity' using a SNMPGET on hrProcessorLoad.11 and hrProcessorLoad.12 respectively[5].

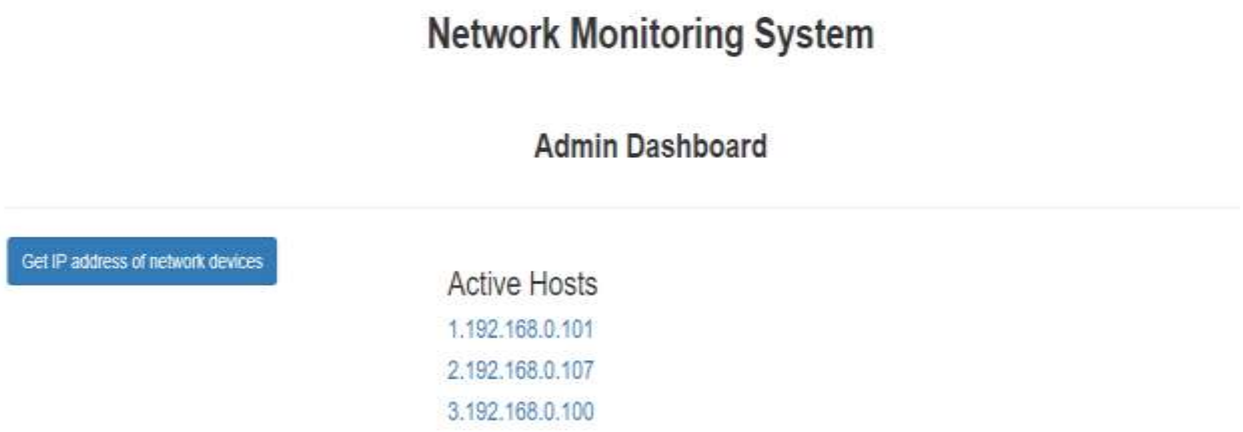
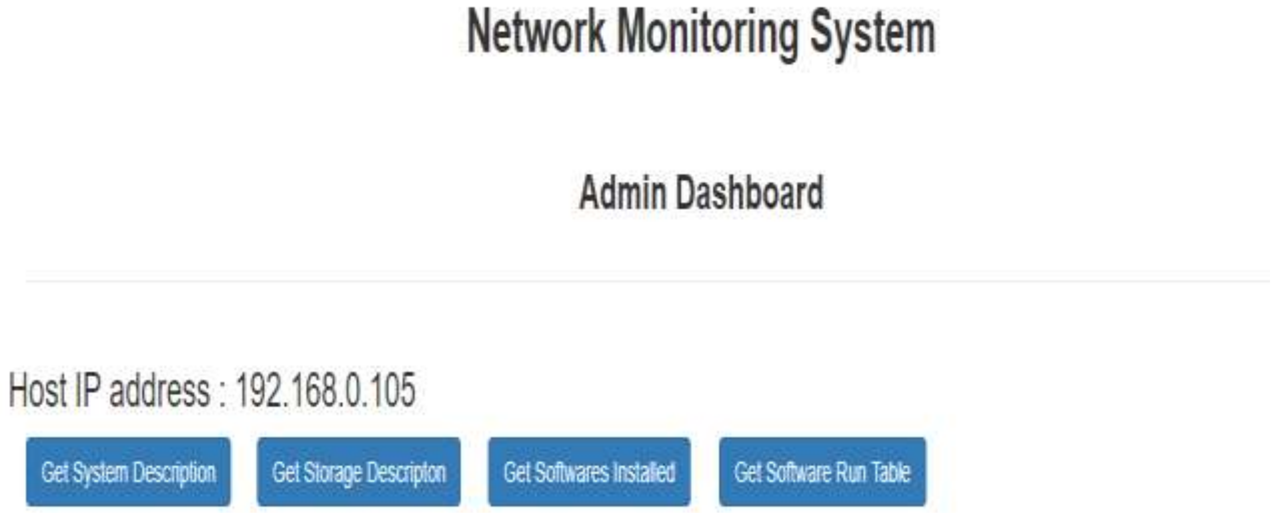
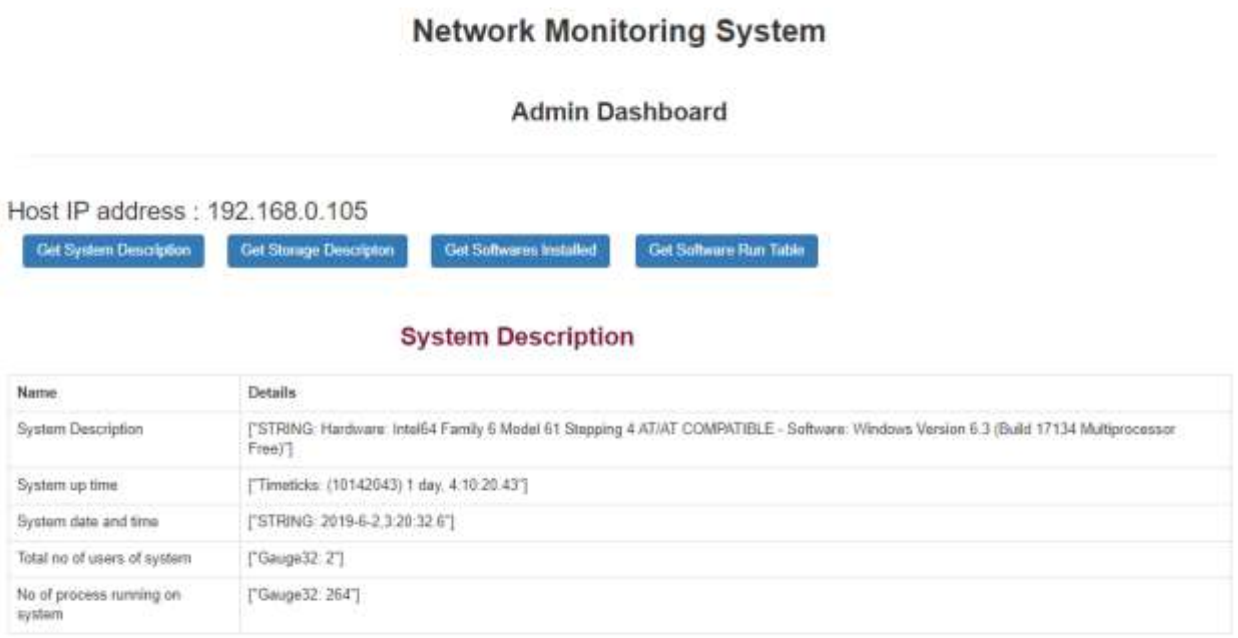


Figure 3.1: Landing page after a successful login into the web application portion of the project. Users can select view details for an active host



Host IP address : 192.168.0.105

Figure 3.2: Getting Details of particular host



Network Monitoring System

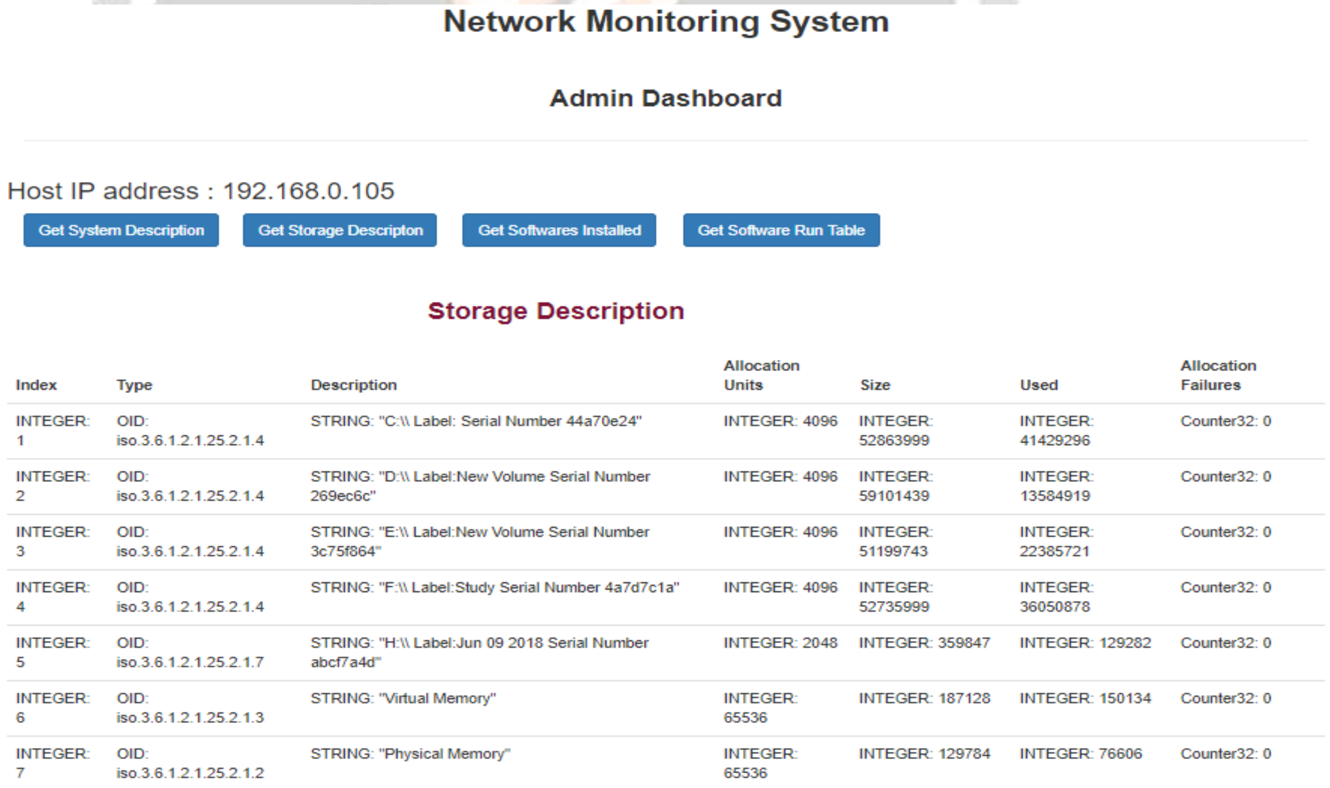
Admin Dashboard

Host IP address : 192.168.0.105

System Description

Name	Details
System Description	["STRING: Hardware: Intel64 Family 6 Model 61 Stepping 4 AT/AT COMPATIBLE - Software: Windows Version 6.3 (Build 17134 Multiprocessor Free)"]
System up time	["Timeticks: (10142043) 1 day, 4:10:20.43"]
System date and time	["STRING: 2019-6-2,3:20:32.6"]
Total no of users of system	["Gauge32: 2"]
No of process running on system	["Gauge32: 264"]

Figure 3.3: A sample of data returned from successive SNMPGET commands for system description on a Windows host



Network Monitoring System

Admin Dashboard

Host IP address : 192.168.0.105

Storage Description

Index	Type	Description	Allocation Units	Size	Used	Allocation Failures
INTEGER: 1	OID: iso.3.6.1.2.1.25.2.1.4	STRING: "C:\\ Label: Serial Number 44a70e24"	INTEGER: 4096	INTEGER: 52863999	INTEGER: 41429296	Counter32: 0
INTEGER: 2	OID: iso.3.6.1.2.1.25.2.1.4	STRING: "D:\\ Label:New Volume Serial Number 269ec6c"	INTEGER: 4096	INTEGER: 59101439	INTEGER: 13584919	Counter32: 0
INTEGER: 3	OID: iso.3.6.1.2.1.25.2.1.4	STRING: "E:\\ Label:New Volume Serial Number 3c75f864"	INTEGER: 4096	INTEGER: 51199743	INTEGER: 22385721	Counter32: 0
INTEGER: 4	OID: iso.3.6.1.2.1.25.2.1.4	STRING: "F:\\ Label:Study Serial Number 4a7d7c1a"	INTEGER: 4096	INTEGER: 52735999	INTEGER: 36050878	Counter32: 0
INTEGER: 5	OID: iso.3.6.1.2.1.25.2.1.7	STRING: "H:\\ Label:Jun 09 2018 Serial Number abcf7a4d"	INTEGER: 2048	INTEGER: 359847	INTEGER: 129282	Counter32: 0
INTEGER: 6	OID: iso.3.6.1.2.1.25.2.1.3	STRING: "Virtual Memory"	INTEGER: 65536	INTEGER: 187128	INTEGER: 150134	Counter32: 0
INTEGER: 7	OID: iso.3.6.1.2.1.25.2.1.2	STRING: "Physical Memory"	INTEGER: 65536	INTEGER: 129784	INTEGER: 76606	Counter32: 0

Figure 3.4: A sample of data returned from successive SNMPWALK commands for Storage Description on a Windows host

4. CONCLUSIONS

We was able to successfully achieve all the goals we set out to achieve. We set out to build an SNMP base monitor on a Raspberry Pi. The stated goal was successfully develop this monitor and be able to capture data associated with hosts on the network. This goal was achieved. We were able to retrieve system description, storage description, processor and memory usage. As stated in the Problem Statement, the aim was illustrated that an effective and cheap network monitor could be developed for the Raspberry Pi. This is relevant because a lot of commercially available network monitors are costly. Thus, in order to provide the highly required protection of network monitoring it should be made cheap.

5. REFERENCES

[1]. Achmad Affandi, Dhany Riyanto, Istas Pratomo, Gatot Kusrahardjo “Design and Implementation Fast Response System Monitoring Server Using Simple Network Management Protocol (SNMP)” International Seminar on Intelligent Technology and Its Applications, 2015

[2]. J.D. Case, M. Fedor, M.L. Schoffstall, J. Davin, “Simple Network Management Protocol (SNMP)”, IETF RFC 1157, May 1990

[3]. Manage Engine- Enterprise IT Management, https://www.manageengine.com/network-monitoring/what-is-snmp.html.

[4]. SNMPWALK. (n.d.). Retrieved April 10, 2016, from http://net-snmp.sourceforge.net/docs/man/snmpwalk.

[5]. Net-SNMP Tutorial --snmpget. (n.d.). Retrieved April 10, 2016, from http://net-snmp.sourceforge.net/tutorial/tutorial-5/commands/snmpget.html

[6]. SNMP Overview. (n.d.). Retrieved March 29, 2016, from https://www.webnms.com/cagent/help/technology_used/c_snmp_overview.html

[7].SNMPGETNEXT. (n.d.). Retrieved April 10, 2016, from http://net-snmp.sourceforge.net/docs/man/snmpgetnext.html

[8].What is network management system? - Definition from WhatIs.com. (n.d.). Retrieved March 29, 2016, from http://whatis.techtarget.com/definition/network-management- system

[9]. Mauro, D. R., & Schmidt, K. J. (2005). *Essential SNMP*. Beijing: O'Reilly.

[10]. What's with the different SNMP versions? v1, v2c, v3? - LogicMonitor. (2012). Retrieved April 10, 2016, from http://www.logicmonitor.com/blog/2012/10/05/whats-with-the- different-snmp-versions-s1-v2c-v3

[11].SNMP MIB for windows- http://www.mibdepot.com/cgi-bin/vendor_index.cgi?r=microsoft