Design and Implementation of a Web-Based Local Area Network (LAN) Monitoring System with SMS notifications

OLUWARANTI, A. I.1 and TALABI, Y.O.
Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria.

ABSTRACT: The paper presents the design and implementation of a web-based LAN monitoring system. The objectives of this work include the detection of the reachability of network hosts, the provision of a web application interface for easy monitoring and an alert mechanism to signal network failures. The model network architecture consist of a proxy server, a monitoring server, a 3-host representation for Ethernet subnets and a mobile phone interfaced via a Short Message Service (SMS) server with the monitoring server. The host on the network is detected using a ping-like tool. A polling application on the proxy server provides the monitoring server with the list of machines connected to the Internet through gateway proxy. The troubleshooting detects which network hosts are up or down, while the web application provides a graphic interface for viewing dynamic network information on all host machines on the network. A message stream containing the location of critical network disconnection is sent through SMS to a network administrator when such occurs.

Keywords: Proxy server; polling engine; OAUNET; mobile phone; SMS

Accepted for Publication, December 10, 2010

INTRODUCTION

This research is focused on making network monitoring activities easier through the use of a web interface, thereby improving an aspect of network management namely, fault management. The monitoring system was designed and implemented on Obafemi Awolowo University Campus Network (OAUNET). The Obafemi Awolowo University campus is located at Ile-Ife, Nigeria. The distributed client – server network architecture comprises several Ethernet local area networks (or subnets). Internet Protocol (IP) addresses are assigned to host dynamically through a Dynamic Host Configuration Protocol. Access to the Internet is possible through proxy servers (Abiona et al., 2006). This network access property of the Ethernet network was found useful in the means of acquisition of reachability information for host machines on the network. The network connection status is first determined from the proxy server, thereby reducing the ping and ping - reply packets generation to the barest minimum. This paper aims at developing an easy-to-manage, cost-effective independent system for monitoring a Local Area Network (LAN). The objectives of the LAN monitoring system are to:

a. detect and distinguish between hosts that are down due to power failure and hosts that are unreachable as a result of network connection.

b. develop a web-based application with color-coded display feature for viewing network status, notification and problem history.

c. create SMS notifications to specific contacts when such network problems occur and specify the location of affected host.

1 Correspondence to A.I. Oluranti, aranti@oauife.edu.ng
LITERATURE REVIEW

The network management model as designed by the International Standard Organization (ISO) consists of five functional areas, which are: Fault management, Configuration management, Accounting management, Performance management, and Security management, popularly known as FCAPS (Cisco, 2006). A common network management protocol is the Simple Network Management Protocol (SNMP). SNMP is an internet protocol designed to facilitate the management of information between network devices. It allows for remote and local management of items on the network including servers, workstations, routers, switches and other managed devices. SNMP is part of a larger architecture called the Internet Network Management Framework (NMF), which is defined in Internet documents called requests for comments (RFCs 1065, 1066, and 1067).

SNMP is a standardized lightweight protocol that is universally supported, and allows distributed management access (Ballew, 1997; Al-Shaer, 2005).

SNMP uses User Datagram Protocol (UDP) as the transport mechanism for SNMP messages. By using SNMP-transported data (such as packets per second and network error rates), network administrators can more easily manage network performance, find and solve network problems, and plan for network growth (Francios, 1998)

Existing tools
Numerous application software (open source and commercial) have been developed to focus on one or more elements from the ISO Network Management Reference Model. They include amongst others, the Multi-Router Traffic Grapher (MRTG): a common tool for bandwidth monitoring on networks - an aspect of performance management (Oetiker, 2006); Network Management Information System (NMIS) that employ tools such as round – robin database, Perl, SNMP Session, Apache, and Net-SNMP to implement performance, security, configuration and fault management and Network Top (NTOP): a network performance manager as well as a protocol analyzer which displays a list of hosts that are currently using the network through a web browser and reports information concerning the (IP and non-IP) traffic generated and received by each host Lopez (2000) and Hunt (1998).

Related works on campus network
A study of related works was carried out in order to understand what has been done in the area of open-source, campus-wide network management. The study of Obafemi Awolowo University Campus network (OAUNet) shows that the existing network management software in use is the Multi-Router Traffic Grapher (MRTG), which is applied in the area of performance management to monitor the bandwidth utilization of the network. This assists the network administrator to determine what part of the network is mostly used, and when. With the MTRG in place, graphical reports can be generated periodically in order to know which part of the network is always busy, and at what time of the day.

METHODOLOGY

Data for the work (e.g. host’s properties, host’s unique Media Access Control (MAC) identification, IP address and the subnet structure) were sourced by physical observation of the machines running on selected subnets of the campus network. A random sampling of the computers on each subnet indicated that most of the machines function as hosts. These hosts are interconnected through a central network device (i.e. a switch) in a star-star network topology. The implementation involves the development of:

i. a network monitoring application for detecting connection status of network machines.
ii. a monitoring software in form of a web application, which is compatible with LINUX operating system (it is possible to run the client on other operating system, but LINUX based system was used in this work). This web interface shall display information regarding the status of network hosts using color – coded display.

iii. a SMS gateway application which interfaces a mobile phone with the monitoring system to generate short messages when critical network disconnection problem occur. The program notifies of the occurrence and location of unavailability.

Design Architecture
The design architecture is presented in figure 1. This is a test network setup to model the network topology of the OAUNet taken as case study. It comprises a proxy server, a monitoring server and a representation for each subnet using three machines connected through a switch.

Attached to the network backbone is the monitoring system, such that it is on the same Ethernet network as the network servers. The system communicates with the proxy server which functions as the gateway to the Internet for all network hosts through its external network interface. From the proxy server, the connection status for each host machine is obtained. Connected to the monitoring system, via a data cable, is a Global System for Mobile communications (GSM) phone which is used to send SMS notification alerts to the network administrator.

The proxy server is the secondary source of network information for the monitoring server. Normally, in order for all the computers on campus to have Internet access they would all have to be assigned routable (or public) IP addresses that could pass through the Internet. For a conservative use of the Internet address space however, all the host computers on a local area network are assigned private or unique addresses and they all share a single “public” address to access resources on the Internet. This sharing is accomplished by configuring the privately-addressed hosts to use a special server, referred to as a proxy server to access the
Internet. A proxy server has two Network Interface Cards (NIC) because it is connected to two different networks. One NIC is connected to the Internet and is assigned a single public IP address (This NIC is referred to as the external interface). The other NIC is connected to the internal local area network. It is assigned a private IP address so that it can communicate with all of the other privately-addressed computers on Campus. The proxy server acts as a gateway to the Internet and also acts as a firewall that protects the internal network. However, in addition to acting as a gateway, the proxy server acts as an address translator. When a computer on the internal network with a private address wants to request information from a Web site, it actually sends the request to the internal interface of the proxy server. The proxy server, with its public routable address on the external NIC, is the one that actually sends the request to the Internet Web server. The Web server sends the response back to the proxy server’s external NIC, and the proxy server then forwards the response on to the computer on the internal network that made the initial request. The proxy server keeps track of which internal computers make which requests.

**Design model**

The implementation comprises a set of applications running at the back-end, (i.e., modules which run silently at the background) and a web application user interface (Web browser) at the front-end to capture the essence of the monitoring system. Figure 2 represents the design model, comprising the various aspects of the system and the flow pattern. The first back-end application is a polling agent that provides current connection information of host machines on the network. It is used to detect the sockets opened on the proxy server for the purpose of Internet communications. The socket information include the local and remote IP addresses as well as the local and remote port numbers used by the network connection, the connection status and the associated process identification. The information collected by the polling agent is stored in a MySQL database as input data to be processed by the next back-end application, the troubleshooting engine. The function of this module is to identify the network hosts that are not reachable by cross-checking the list of hosts connected to the internet against a table of recognized network hosts also in the same database, and then determining whether such hosts are reachable on the local network. In the event of critical disconnection problem, such as the disconnection of a backbone switch leading to the unavailability of network service for an entire subnet, control is transferred to a SMS server that will generate SMS alert to the network administrator via a GSM phone. The network administrator’s contact is specified and a specific message stream indicating the physical location affected is sent. The next phase in the design involves accessing the database for information retrieval, then generating a color – coded graphic display of host availability.

**IMPLEMENTATION AND RESULTS**

The implementation of the design model groups the monitoring application into four separate modules which are integrated to give a complete solution. These include: a polling agent, a troubleshooting application, a web application server and a SMS gateway server.

**The polling engine**

The test at this phase provides a feedback of the application’s ability to filter relevant monitoring information, which includes the IP addresses of the hosts that are currently using the Internet services. This application uses the netstat command with optional attributes to gather opened sockets from the virtual proc file...
system which gives access to the kernel (i.e. the core of the operating system) status information on Linux operating system via the network files. The information on open sockets include connection status, process identification and process name of the program that has the socket opened, the IP address and port number of both the remote and local ends of the socket and the type of protocol used by the socket. Table 1 shows a sample result obtained by running the program statement:

```
netstat -a -A inet -n -p
```

<table>
<thead>
<tr>
<th>Proto</th>
<th>LocalAddress</th>
<th>ForeignAddress</th>
<th>State</th>
<th>PID/Program Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP</td>
<td>10.105.21.121:32829</td>
<td>66.249.85.104:80</td>
<td>SYN STOP</td>
<td>2200/mozilla</td>
</tr>
<tr>
<td>UDP</td>
<td>10.105.21.121:138</td>
<td>0.0.0.0:0</td>
<td>CLOSE</td>
<td>2003/nmbd</td>
</tr>
<tr>
<td>TCP</td>
<td>192.168.0.2:32816</td>
<td>192.168.0.1:139</td>
<td>LISTEN</td>
<td>2110/nautilus</td>
</tr>
</tbody>
</table>

The Proto column specifies whether the connection uses a transmission control protocol (TCP), a user datagram protocol (UDP) or a Raw Ethernet Communication (RAW). Also included in the result are the destination and source IP addresses and port combinations (separated by a colon) used for the connection. The PID/Program Name column describes the application that is being used for communication, for example, the Mozilla Web Browser indicated in column 8 of the first row of the table. The program is identified by a numeric program identity (PID) 2200. The state
of the connection refers to the status of the opened socket. This could take any one of these values:
- **??**, which implies that the state is unknown
- **ESTAB**, which indicates that the socket has an established connection
- **SYNSNT**, which implies that the socket is actively attempting to establish a connection
- **SYNRCV**, which denotes request for connection has been received
- **CLOSE** state, which indicates that the connection socket is not in use
- **LISTEN**, which implies that the connection is established.
- **FIN_WAIT1**, **FIN_WAIT2**, which describe the shutting down of the connection

The polling agent then retrieves distinct host IP addresses in the LISTEN state from the netstat result column and stores in an array variable. The selected result is passed on by means of client-server socket communication from the proxy server to the monitoring system and the HostStatus field of the reference table in the MySQL database (running on the monitoring system) is automatically updated.

The troubleshooting application
The database server keeps the record of machines to be monitored in a database table H_REF. The table fields include IPaddress, MACIdentifier, Hostname, Machine Type, SubnetID and HostStatus. While all other fields have almost static values, the values in the HostStatus field changes with a change in the status of a monitored host. A periodic update of this field is performed by the monitoring system on receiving the information from the proxy server.

The HostStatus field is scanned for hosts that are in the OFFLINE (0) state. The IP addresses of the hosts are again collected in an array, and then the list is passed on to the troubleshooting application. Such hosts are sent ping packets to confirm unavailability. Any host that does not reply the ping is considered unavailable. However, the ping is not adequate where a large number of host machines may be affected in a subnet. With a large a number of IP addresses for each subnet, its becomes more and more time consuming to check which of the IP addresses are actively in use are reachable using a ping command. A modification to the ping is used for achieving the desired purpose. This modification (known as fping) entails the use of the Internet Control Message Protocol (ICMP) echo request to determine if a host is up. A list of hosts to ping is specified, and instead of trying one host until it timeouts or replies, the program sends out a ping packet and moves on to the next host in the list. If a host replies, it is noted and removed from the list of hosts to check. If a host does not respond within a certain time limit and/or retry limit it is considered unreachable.

In situations whereby the number of confirmed unreachable hosts exceeds a specified maximum for a subnet, the time of day in which this occurs is considered and if it is during a period when there should be network activity, a critical network failure is assumed. Table 2 shows the test results for the database table H_REF. A value ‘1’ represents an ONLINE status; while a zero (0) value indicates the machine is OFFLINE.

Testing the web application server
Information given on the web page includes links to hosts on each subnet, as shown in Figure 3. The map of the campus network is the major feature on this home page. It shows at a glance the subnets monitored.
Table 2: Example inputs for database table H_REF

<table>
<thead>
<tr>
<th>HostIP</th>
<th>HostMAC</th>
<th>HostName</th>
<th>Type</th>
<th>Subnet ID</th>
<th>Location</th>
<th>Host Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.105.31.26</td>
<td>00-14-85-04-3A-C6</td>
<td>Aadewuyi</td>
<td>Desktop</td>
<td>1</td>
<td>Rm 223 Demography</td>
<td>1</td>
</tr>
<tr>
<td>10.105.31.158</td>
<td>00-08-0D-02-FB-18</td>
<td>Aakinlo</td>
<td>Laptop</td>
<td>1</td>
<td>Rm 116 Economics</td>
<td>1</td>
</tr>
<tr>
<td>10.105.31.165</td>
<td>00-30-4F-2B-6C-A3</td>
<td>Oribosep</td>
<td>Desktop</td>
<td>1</td>
<td>Rm 127 Administration</td>
<td>0</td>
</tr>
<tr>
<td>10.105.31.246</td>
<td>00-11-5B-6D-B4-2D</td>
<td>Polomola</td>
<td>Desktop</td>
<td>1</td>
<td>Rm 120 Economics</td>
<td>1</td>
</tr>
<tr>
<td>10.105.1.240</td>
<td>00-E0-B8-A7-71-3B</td>
<td>Tripleo</td>
<td>Laptop</td>
<td>2</td>
<td>Rm 005 Computer</td>
<td>0</td>
</tr>
<tr>
<td>10.105.47.246</td>
<td>00-20-ED-5C-A7-17</td>
<td>Ajayi</td>
<td>Desktop</td>
<td>3</td>
<td>Rm 227 Paediatrics</td>
<td>0</td>
</tr>
<tr>
<td>10.105.50.12</td>
<td>00-34-7A-54-23-12</td>
<td>Topjee</td>
<td>Desktop</td>
<td>4</td>
<td>Rm 12 Hum Block</td>
<td>0</td>
</tr>
<tr>
<td>10.105.30.110</td>
<td>00-30-4F-2B-6C-A3</td>
<td>Jegede</td>
<td>Laptop</td>
<td>5</td>
<td>Rm 002 Dentistry</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 3: Links to hosts monitored per subnet
Figure 4 is an instance of the status of all machines on a selected subnet – the Computer Building subnet. A color-coded view of the host status is given alongside the host identification, i.e. the host IP address and the hostname. The status indicator shows GREEN for an online host and RED for any host in the offline state. The network address portion of the IP address is identical for all machines on the same local network, e.g. ‘10.105.1’ for the computer subnet.

![Color-coded display of host status](image)

**Figure 4: Color – coded display of host status**

The machines are distinguished by the last octet of their IP address. All other hosts on the campus network are also grouped according to their subnets on this dynamic page known as the **hosts** page. Each host’s IP identity serves as link to the **details** page shown in Figure 5. The web page displays the physical location (building and room) of the machine, the machine type (desktop or laptop), the machine status, connection speed, and its MAC address. The **details** page summarizes all information stored in the H_REF table about each host.

Availability statistics is a measure of the network usage by each host on a monthly basis. The content of the HISTORY database table is accessed through the Notification History web page shown in Figure 6. The page displays the alerts generated when critical problems occur and the present status of the problem, whether resolved or otherwise. The date of occurrence and the exact time of fault log are also displayed. The network administrator is given the option of deciding the order of information presentation. Ordering by the problem status helps to distinguish between resolved and unresolved network issues.

A view of notification history ordered by the date of occurrence determines how many issues are detected per day. The alert message order option specifies how many times a particular subnet has experienced critical fault over a period of time. The number of notifications received by distinct administrators can be inferred when the information is ordered by the network administrator.
SMS notification
Short Message Service (SMS) alert is generated by running codes that triggers a SMS server application. The trigger is raised on specific need of notifying a critical network disconnection affecting an entire subnet. The mechanism notifies the administrator in a few minutes after the occurrence depending on the level of traffic from the GSM service provider. The implementation uses a Nokia 3310 phone.
CONCLUSION

The web-based monitoring system was designed for use on a campus-wide network. This is able to determine whether a host machine is up or down. However, capabilities for detecting the power state of a machine are yet to be included. The implementation requires the development of agents that may reside on host machines and alert the monitoring system when the host is about to be turned off. This is an area which still requires much development. Instead, emphasis was laid on detecting network hosts without generating considerable network traffic, and providing a web interface that would eliminate the need for tasking configurations before a monitoring tool can be utilized.

REFERENCES


