IITB Network Monitoring and Development of an Archival Tool

Project Dissertation

Submitted in partial fulfillment of the requirements

of the degree of

Master of Technology

by

Abhishek Kumar

Roll No. :153050043

Supervisor:

Prof. Varsha Apte

Department of Computer Science and Engineering
Indian Institute of Technology, Bombay

22 July,2017

Dissertation Approval

This dissertation entitled **A Long Term Archival and Reporting Application for Network Monitoring Data** by **Abhishek Kumar** is approved for the degree of Master of Technology in Computer Science and Engineering from **Indian Institute of Technology Bombay.**

Examiners

PSKulkarni

(Prof. Purushottam Kulkarni)

(Prof. Kameswari Chebrolu)

Supervisor

Prof. Varsha Apte

Chairman

PSKulkarni

Date: 28/6/2017

Place: IITB

Declaration

I declare that this written submission represents my ideas in my own words and where others ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

ABHISHEK KUMAR

(Name of the student)

153050043

(Roll No.)

Date: 28/6/2017

Acknowledgement

I would like to express gratitude and heartily thanks towards my guide **Prof. Varsha Apte**, for providing her excellent guidance, encouragement and inspiration throughout the project work. Without her invaluable guidance, this work could never have been possible.

I would also like to thanks synerG lab group, professors specially **Prof. Kameswari Chebrolu** and my friend as well.

Abhishek Kumar
153050043
M.tech, CSE

Abstract

IITB has huge networking infrastructure which comprises a lot of networking devices, thousands of users with different privileges. Maintaining such a huge network is a challenge for network administrator. To maintain such a network and full-fill requirement of a robust, secure internet a lot of firewalls, proxy devices with sophisticated rules are implemented. Besides this, a number of monitoring and alerting devices are also used. All of these devices do a very handsome job in their own way but there are some cases where these devices are not that much helpful. One such scenario is measurement of traffic intensity and storage of traffic intensity based statistics. Although IITB has a lot of current information related to traffic intensity but this current information is not processed and go in vain after some time as new traffic data comes. To maintain such a network, it is important to analyze traffic data and store past trends. Past trends of traffic are very important because past trends not only tells about network behavior in past it also help in predicting future behavior of network. A lot of prediction and questions can be answered by just analyzing past traffic.

Purpose of this thesis is to propose and implement one such tool which can store raw traffic data for a shorter period of time and later replace this raw data with statistics over a defined granularity. The granularity at which these statistics are stored would change as data will become old. Idea was to come up with an architecture and algorithm which is totally configurable in term of defining the archiving level and later to use all these archival level to serve as reporting functionality. These trends are later used for analyzing or measuring traffic intensity.

Contents

List of Figures

Chapter 1

Introduction

IITB has huge networking infrastructure comprising of thousands of switches, Wireless access points and thousands of kilometers of cable, which is used by a lot over tens of thousands users in a variety of different ways. Complicated devices, like proxy servers Internal and external firewalls VPN all with sophisticated rules, are used, implemented and maintained. Within users, there are a variety of users like faculty, staff members, students or residential people. When we talk about using IITB network all of these users don't have equal privileges. IITB also need to take care of user's authorization and authentication before providing any kind of network service. Continues usage of Internet, variety of traffic, authorization and authentication challenges , attacks attempts from inside and outside make it really challenging for computer center staff to full-fill demand of fast, efficient, robust and secure network to IITB.

Figure 2.1 in Section 2.2 of chapter2 describes set of network components which come into picture when user access Internet from inside. This diagram can give us a fair Idea of the complexity of IITB network because of the fact that we are only talking about Internet access from inside and not talking about Intranet access, any kind of internal activity, Access from outside and other aspects of the network.

IITB network is mainly divided into four following parts.

- Hostels network
- Academic area network
- Residential area network
- Wireless network

All of these areas are connected with a backbone switch through number of layer2 switches and distribution switches. This backbone switch receives traffic from the user of one of the area and on basis of user, location of user and destination type it will transfer traffic to one out of Internal Firewall, netmon proxy server or transparent proxy servers via layer2 switch. These devices on receiving packets take care of authentication and authorization of service with help of LDAP server and internet.iitb.ac.in server. On basis of rule implemented in devices, packets can be forwarded further or can be blocked. Once process is completed, packets are hand-overed to external firewall which again does filtering and natting on basis of rule implemented in it. At last packets are hand-overed to BGP which send traffic to one of ISP of IITB. Some of devices also maintains logs of packets going through it and finally all logs are moved to a central log server. This whole structure is described and shown in section xx and figure yy.

For providing fast,secure and robust network it is very important to have every device-related detail at every moment. Traffic coming and going at each interface of devices, Performance of devices, free disk space at devices, number of processes at device, maintaining of logs at devices and to study log collected at devices are few examples which are very useful for maintaining and diagnosis purpose in network. For example, if some user is continuously bombarding the traffic and causing network congestion, It is important for CC staff to not only identify the problem but also culprit. One way of doing this is to monitor logs. Whenever a particular IP is appearing in logs for more than a threshold value alert should be triggered. Another example is if at BGP router no traffic is coming for some time then a notification should be sent to concern authority. If for some reason IITB network goes down, cause and location of the problem should be recognizable and that also in minimal time.

All of these issues can be handled by continuously monitoring network. For monitoring purpose, IITB is using many monitoring tools like Zabbix, nms tool for monitoring switches, Cacti and observium. One of such tool which I studied in detail is ZABBIX monitoring tool. It does a very good job by letting user define events, triggers and action corresponding to triggers. For example, monitoring traffic at some interface of BGP router is an event. If traffic at this interface of BGP is zero for more than 5 minutes then this is a trigger and if this trigger happens send a mail to head of CC is an action. According to need we can define events, triggers and action. Zabbix does a very beautiful job in terms of monitoring and reporting events like number of processes, traffic intensity at each interface of each device, free disk space, number of users logged-in in device. It generates real-time graph for all these events.

Despite having too many features it has some limitations when it comes to generating statistics and archiving long term trends out of raw data. There are many questions which can not be answered by zabbix. Any kind of questions in discrete period of time like "report maximum traffic in month of December from 2010 to 2017" or any kind of queries related to "Zero traffic ". These are some questions which can not be answered by Zabbix. We would see many of such questions in chapter 2. Other Zabbix limitation is it can not parse or process a log file and can not generate reports or statistics out of those logs file. Logs file specially proxy log file are very useful in providing answer to number of questions like number of unique user per day, number of different devices at IITB, network usage for academics vs residential vs hostel, number of wireless users vs wired user and many more similar kind of question.

To summarize above paragraph, there has been an ample amount of work done for installing, maintaining and diagnosis of IITB's network by monitoring it but not much has been done in area of log analysis, statistical analysis and measurement of network traffic. Long-term trends need to be archived. These long-term trends are not only useful in diagnosis and maintaining but also very useful in predicting future behaviour of network and are very helpful in taking future decision. For this purpose, it is very important to have a tool which will collect data from Zabbix and will generate statistics out of it and these statistics will be stored instead of raw data. Tool must be able to archive data at different level of archiving granularity and as data will become old it will be archived at greater granularity level. Similarly, it will parse log file at proxy servers and will pull important information out of it and generate statistics from this information. These statistics will be archived and later will help in studying trend and predicting future behaviour of network in advance. Few of graphs or tables which we would like to see using tool is as follow.

- Max Traffic Statistics for December from 2011 to 2016

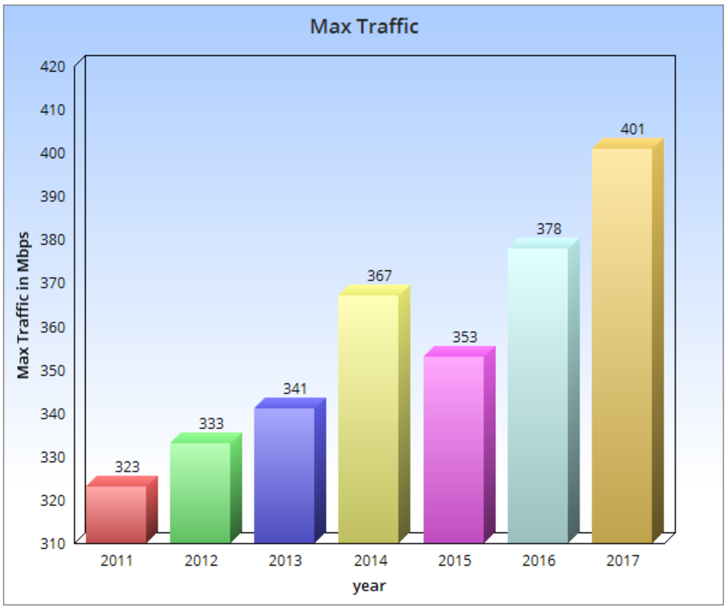


Figure 1.1: A hypothetical graph showing December Month Statistics

Now after looking at this graph expected traffic for December 2018 can be guessed easily. We will see more of such graph in chapter 4 where we will explain expected functionality of tool in detail.

So at final, My thesis work mainly revolve around this aspect of IITB network. Specially:

- Develop a web interface tool for traffic intensity based data analysis that will statistically characterize and archive the statistics on an ongoing basis so that long term trends can be studied over many years.

- To analyze traces collected at firewalls, proxy servers and other devices to extract quality information like type of traffic, number of user, peak bandwidth. A prerequisite for this is optimal level of anonymization so that users remain anonymous, but usage trends remain extractable.
- Work on Zabbix framework for network monitoring and determining the usage of network resources like memory, bandwidth for different application in IITB network.
- Extend zabbix tool as per IITB network requirement.
- As a prerequisite, it was very important for me to understand architecture of IITB and thus study and documentation of IITB network is important part of my thesis work.

Chapter 2

Background Study & Motivation

First and foremost task for me was to revise my concepts related to network devices, protocols, network security related issues and solution etc. Since my thesis topic is related to IITB network, It was also important for me to understand IITB network. I needed to understand how traffic flows in IITB from different different areas w.r.t to user type and connection type like wired or wireless connection. For that I spent initial phase in understanding IITB network and formally documenting it.

Another important part for me was to study and understanding working of Zabbix Monitoring tool. Since IITB uses this tool for monitoring purpose and I need to develop a tool which will extract statistics out of raw data collected by zabbix, It was very important to understand it's architecture and working. Besides these, IP anonymization was also one more area in which I did study, studied different tool and implemented those tools. Before tracing any log file It was very important to anonymize all IP addresses in that file. For that a best suitable tool and best approach was required. In upcoming sections of this chapter I will discuss my background study in three section - IITB Network Architecture, IP anonymization and Zabbix Monitoring tool respectively. At last section of this chapter formal definition of archival tool and motivation behind having an archival tool is explained.

2.1 IITB Network Flow Diagrams

Important Note

- We will use following labels throughout documentation for switches.

Switch	**label Used**
X8/BackbobeSwitch	switch_1
201 Switch	switch_2
209 switch	switch_3
200 switch	switch_4

- This documentation is to describe architecture and packet flow when user is inside IIT & and is accessing internet(not intranet). Flow diagrams presented in document are only to describe this flow. In actual it might happen that there are some more devices attached with a particular device but not shown in diagram because that device has no business when describing flow of accessing internet. For example in actual there are more than two external firewall but we will show only two firewall because other firewalls are of no use in internet access from inside IITB.

2.2 Network Overview -

Network in IITB is mainly divided into four parts:

1. Hostels network
2. Academic area network
3. Residential area network
4. Wireless network

Main components of IITB network are as follow:

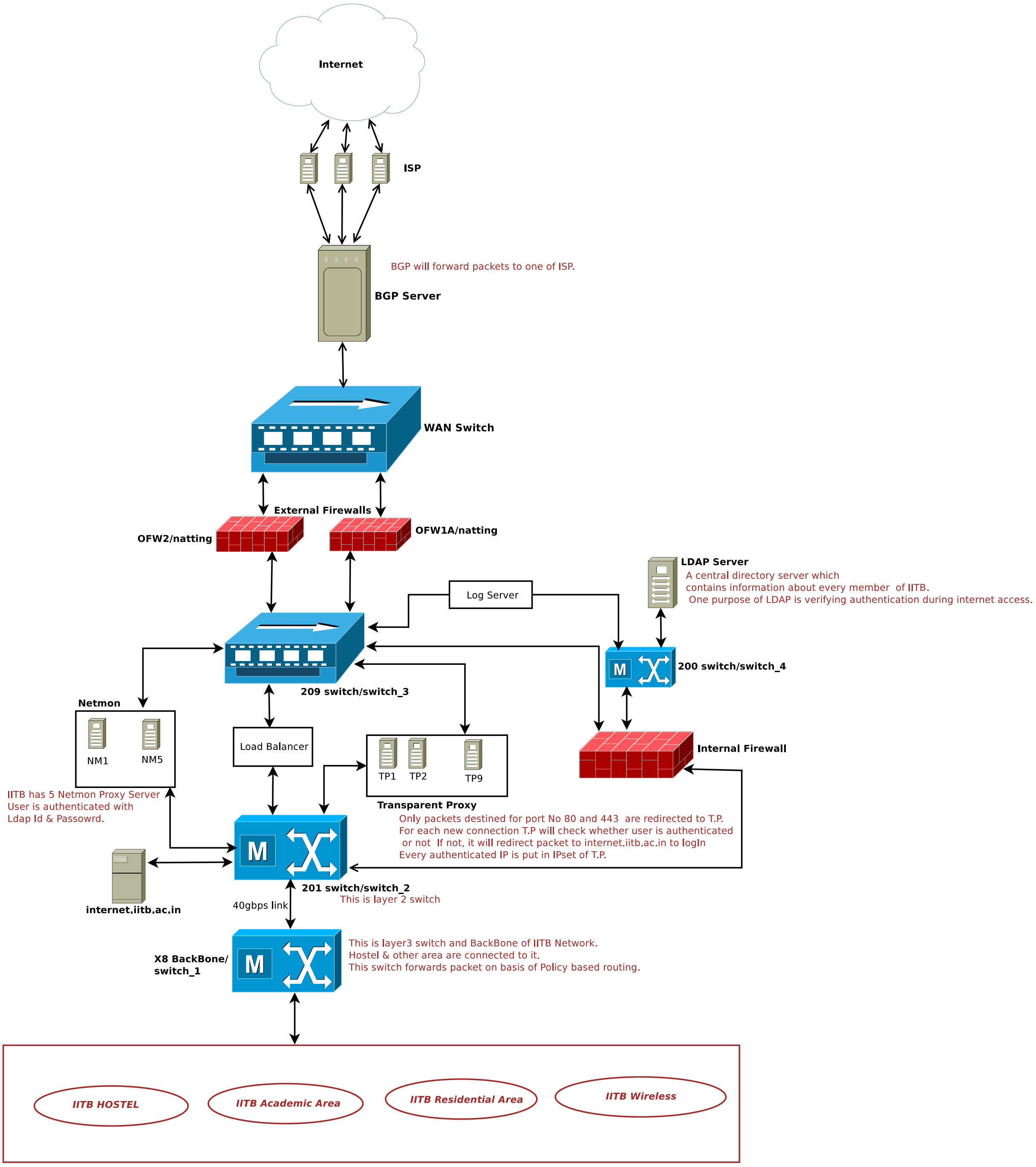
2.2.1 BGP Server

- BGP server is face of IITB to external world. Every packet destined for IITB will enter IIT through BGP server and Similarly every packet going outside will go through BGP server.
- **Connection & Flow**
  - It is connected with WAN switch at one side & with 3 ISP's of IITB from other side.
  - It receives packet from WAN switch and forwards packet to one of three ISP's of IITB for outgoing packet.
- **Configurational detail**
  - add configurational detail here.

2.2.2 ISP

Following table shows internet service providers of IITB:

ISP Name	**Bandwith**	**AS Number**
TATACOMM-AS TATA	1.25 Gbps	4755
Vodafone-Net-AS-AP	1.0 Gbps	55410
NKN Core Network	10 Gbps	55824



BGP will forward packets to one of ISP.

A central directory server which
contains information about every member of IITB.
One purpose of LDAP is verifying authentication during internet access.

IITB has 5 Netmon Proxy Server
User is authenticated with
Ldap Id & Passowrd.

Only packets destined for port No 80 and 443 are redirected to T.P.
For each new connection T.P will check whether user is authenticated
or not If not, it will redirect packet to internet.iitb.ac.in to logIn
Every authenticated IP is put in IPset of T.P.

This is layer 2 switch

This is layer3 switch and BackBone of IITB Network.
Hostel & other area are connected to it.
This switch forwards packet on basis of Policy based routing.

Figure 2.1: General Overview of IITB Network

2.2.3 WAN Switch

- This switch is layer3 switch.
- It forwards packets at speed of 10GBps.

- **Flow & Connection**
  - This switch is directly connected to BGP router from one side & with external firewall of IITB form other side.
  - For outgoing packet it receives packet from one of firewall & forward its to BGP router and vice versa for packets coming from outside(more on it in section 5)
- **Configurational Detail**
  - add configurational detail, like IP address of switch. Number of input output ports. what are open ports?, here .

2.2.4 External Firwalls

- External firewalls are labelled as **OFW_2** & **OFW_1A** in diagram.
- Both firewall servers same purpose but for different kind of user(discussed later in same section)
- These are connected with WAN switch from one side and from other side are connected to switch_209.
- Main purpose of external firewall is to protect IITB network from outside attack.
- **Logic Implemented**
  - It does filtering for packet coming from WAN switch. After filtering packets are forwarded to switch_3.
  - Besides filtering it also does natting for packets going outside.
  - Add filtering rule here. what kind of traffic is allowed to go outside.
  - Similiarly what kind of outside traffic is allowed to pass through this firewall.

These External firewall are as follow:-

1. **OFW_2**
   - It is a pathway to WAN switch for packets which are coming from Transparent proxy or netmon proxy when service accessed by them is at port 80 and port 443.
   - It means traffic generated by a student or a staff member will always pass through this firewall, if service they are demanding is at port 80 or port 443.
   - Traffic generated from wireless network of IITB will pass through this firewall irrespective of type of user. It means not only student and staff but faculty's traffic also will go through this firewall if connected through wireless connection.

      - **Configuration**
        - Add config detail here?

2. **OFW_1A**
   - It is a pathway to WAN switch for packets which are coming via Internal firewall.
   - It is pathway mainly meant for faculty traffic.
   - A student or staff traffic will never go through this firewall if he is accessing http or https web server.
     - **Configuration**
       - what is config?
       - what are filteration rule here?

2.2.5 Switch_209/Switch_3

- **Flow & Connection**
  - This switch is connected with external firewalls from one side and from other side it is connected with Trasparent Proxy, Internal Firewall and netmon proxy server.
  - It is layer2 switch and forwards packet on basis of destination MAC address.
  - For each incoming packet, it will forward packet to one of two external firewalls.
- **Logic implemented**
  - It is layer2 packet and will forward packets on basis of destination MAC address.
- **Configurational Detail**
  - Add configuration here.

2.2.6 Internal Firwall (IFWA)

- It is linux box. It means firewall implemeted here is a software firewall.
- This linux box servers as firewall for packets coming from internal LAN and destined for DMZ or outside IITB.
- This firewall is basically to protect DMZ from attack attempted from students or IITB LAN.
- **Flow & Connection**
  - All packet having source IP allocated to residential or academic area first comes at this firewall

  - All packets coming from hostels, irrespective of user, having destination port other than 80 and 443 will come here
  - It is connected with switch_3 at one side
  - From other side it is connected with switch_2 which is connected with backbone of IITB
  - It is also connected with switch_4 which is further connected to LDAP server
  - It will receive packet from switch_2 and will forward it to either transparent proxy or switch_3 on basis of source IP.(discussed more in logic implemented section)

- **Authentication**
  - Firewall will allow internet access only if user is authenticated.
  - In IITB for authentication purpose, each user is provided with a LDAP id and a chosen password. internet.iitb.ac.in is webpage used for authentication purpose with help of LDAP server.
  - LDAP server is central directory server which contains all information about every member of IITB in hierarchial and structured way. It contains user's personal information as well services authorised to them as internet user. (more on it later in section XXX)
  - For authentication purpose, firewall maintains two IPset tables. These IPset tables maintains list of authenticated users
    * One IPtable contains IP address which are related to faculties
    * Other IPtables contains IP address of users other than faculty of IITB.
  - If source IP address of some packet is not present in any of IPset, firewall will redirect packet to internet.iitb.ac.in
  - internet.iitb.ac.in will do authentication and will put source IP in one of IPset table of firewall according to user of that IP.
  - internet.iitb.ac.in will do authentication with help of LDAP server. LDAP server will give info whether user is faculty or staff or student.

- **Logic implemented**
  - If packet reaching here is not authenticated, it will redirect packet to internet.iitb.ac.in for authentication as described above.
  - Once authentication is done source IP will be present in one of IPSet table of firewall.
  - If packets belongs to faculty, it will be forwarded to Switch_3 without any log formation with destination MAC address as MAC address of extenal firewall OFW_1A.

  - If packet does not belong to faculty and destination port is 80 or 443 it will be redirect packet to transparent proxy.(more on above two points in section 2).
  - if packet does not belong to faculty and destination port is other than 80 or 443 it will forward packet to switch_3

- **Configuration**
  - Add configuration like IP address of firewall
  - what kind of filteration rules are being applied here.

2.2.7 Transparent Proxy

- Transparent proxy is for students, staff when service being accessed is http or https(destination port number 80 or 443)
- If packet is related to student or staff and destination port number is 80 or 443, it will reach here at some point irrespective of the fact that it was generated from hostels or from academic area.(more on it in section 2, section 3 and section 4)
- It is web proxy and only supports services for HTTP and HTTPs. This is also one of the reason why this proxy is not meant for faculties's traffic.
- IITB has 10 transparent proxies. It makes network more fault tolerant.
- **Flow & Connection**
  - From one side it is connected to switch_3 and at other side it connected to switch_2.
  - It will receive packet from switch_2 and after reading IPset table(same concept as firewall) it can reject connection or can Initiate a new connection with destination web server.
  - Unlike firewall, transparent proxy only maintains one IPset table.
- **Authentication**
  - For each incoming, packet Transparent proxy checks IPset table. If source IP is not present in IPset table, it will redirect packet to internet.iitb.ac.in
  - internet.iitb.ac.in does authentication with help of LDAP server in same way as described in section 1.6.
  - Once authentication is done, internet.iitb.ac.in will put IP in IPset table of transparent proxy.
- **Logic implemented**
  - Once authentication is done Transparent proxy has detail of user logged in.

  - If packet destination port is not 80 or 443 reject the connection.
  - If packet is from hostel & user is student. Initiate a new connection with web server.
  - If packet is from hostel & user is teacher block it.
  - If packet is from academic area & user is student. Initiate a new connection with destination URL.
  - If packet is from wireless connection, Initiate new connection with web server mentioned as destination in packet irrespective of user logged in

- **Configurational detail**
  IITB has 10 transparent proxy

S. Numer	**IP address**	**Used for**
TP1		
TP2		
TP3		
TP4		
TP5		
TP6		
TP7		
TP8		
TP9		
TP10		

2.2.8 netmon proxy server

- IITB has 5 netmon proxy servers.

- All proxy servers are squid proxy server

- It is explicit proxy server of IITB.

- If a user wants to use netmon proxy server, he has to configure it in browser explicitly unlike transparent proxy.

- **Flow & Connection**
  - It is connected with switch_2 from one side and from other side it is connected with external firewall with help of switch_3
  - Every packet with destination of netmon first comes at load balancer box. Load balancer will forward packet to one of 5 netmon on basis of availability and load.
  - When a user configures netmon proxy there is no need to login at internet.iitb.ac.in as netmon has it's own way of authentication and netmon is directly connected with LDAP server.

- It will pop up a form which will ask for LDAP id and password when user will access it first time.

- **Configurational detail**
  - IITB has 5 netmon servers.
  - IP address for netmon proxy is 10.201.13.50.
  - Add other configurational detail.

2.2.9 internet.iitb.ac.in

- It is a web page used for authentication purpose & hosted at IP address 10.201.250.201 .
- Authentication for each user is done by LDAP ID & LDAP password.
- For verification of user's credentials it access LDAP server.
- LDAP is a directory server which contains information related to users in IITB.

2.2.10 Log Server {LOG PROC}

IITB has a dedicated Log server. In IITB, Logs are stored at two places.

1. Netmon proxy server : Here user's LDAP ID is also stored.
2. Transparent proxy server. No user LDAP ID is stored

log stored at both place are in form of squid proxy log format.
All logs are finally moved and stored at LOG PROC.

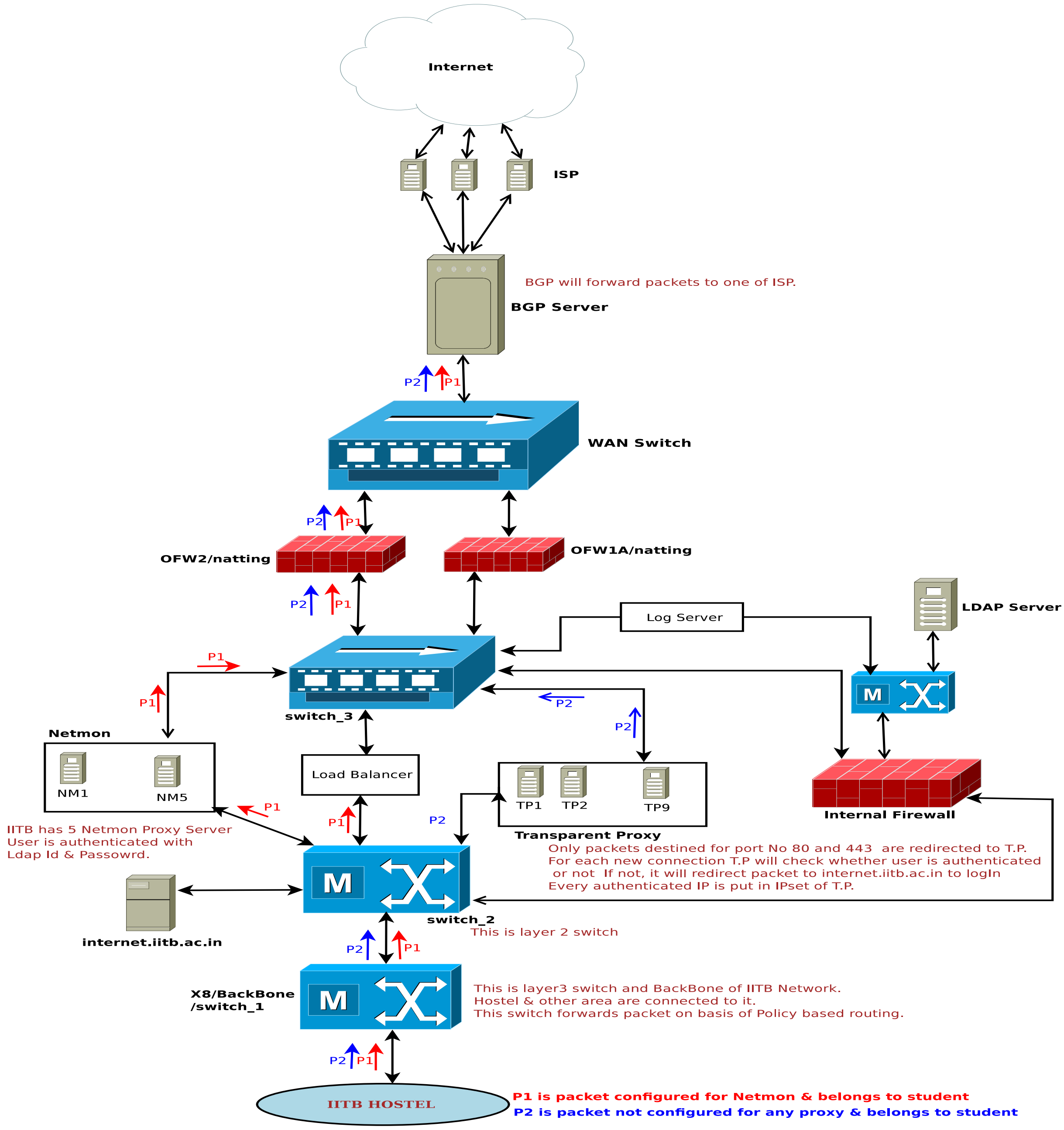
2.2.11 X8 Backbone/Switch_1

- **Flow & Connection**
  - All the four parts of IITB network are connected to this switch. It works as backbone switch for IITB.
  - It forwards packets on basis of policy based routing.
  - It forms ring topology with core switches to make network fault tolerant.
  - Core switches are connected with hostels via distribution switches. Link between Core switch and distribution switches has capacity of 10gbps.

- **Configurational Detail**
  - It is connected with Switch_2 with link of 40 GBPS.

- SDN is used at this switch.
- Add detail like open port, forwarding table etc.
- what feature of SDN are being used here.

2.3 Hostels Network



BGP will forward packets to one of ISP.

IITB has 5 Netmon Proxy Server
User is authenticated with
Ldap Id & Passowrd.

Only packets destined for port No 80 and 443 are redirected to T.P.
For each new connection T.P will check whether user is authenticated
or not If not, it will redirect packet to internet.iitb.ac.in to logIn
Every authenticated IP is put in IPset of T.P.

This is layer 2 switch

This is layer3 switch and BackBone of IITB Network.
Hostel & other area are connected to it.
This switch forwards packet on basis of Policy based routing.

P1 is packet configured for Netmon & belongs to student
P2 is packet not configured for any proxy & belongs to student

Figure 2.2: IITB Hostel Network Architecture & packet Flow

- IITB hostels are connected through switch_1. This layer3 switch acts as backbone of IITB Network.

- Each hostel is connected with number of distribution switches.
- Distribution switches are connected to one core switch. Link capacity between core switch and distribution switch is 10gbps
- These core switches and Switch_1 makes ring topology to make network more fault tolerant
- Every ethernet port in hostel rooms are provided with DHCP IP or static IP address as follow:

10.hostel number.	**wing number floor number.**	**last 2 digit of room number**

- Once packet is received at switch_1, it is then forwarded on basis of its source address and destination address to switch_2.

- Switch_2 will forward packet according to destination MAC address

- **Flow of packets not configuered for Netmon is as follow:**
  - Packet with destination other than destination port 80 and 443 will be redirected to Internal Firewall.
  - Internal firewall will do filtering and will forward packet to switch_3(as discussed in section 1.6)
  - Packets with port number 80 and 443 will be forwarded to Transparent proxy server.
  - Transparent proxy will check whether it is authenticated or not through its IPSet table.(as described in section 1.7)
  - If user is not authenticated, packet will be redirected to internet.iitb.ac.in via switch_2. Internet.iitb.ac.in will access LDAP server to authenticate a user.
  - Every authenticated IP address is stored in IPset of Transparent Proxy.
  - If authentication belong to faculty but source IP is Hostel, packet will be dropped.
  - Once authentication is done, new connection with destination URL is initiated through external firewall OFW_2 via switch_3.

- **Flow of packets configuered for Netmon is as follow:**
  - Packet will be forwarded to LoadBalancer by switch_2.
  - LoadBalancer will forward packet to one out of 5 Netmon servers according to load and availability of Netmon server.

  - Netmon will do authentication by poping up a window on user's screen asking for user's login.
  - Once verified, Netmon proxy will initiate a new connection with destination URL through OFW_2 firewall on behalf of user.

- Outer firewall will provide Natting functionality and packet will be hand overed to BGP router via WAN switch.

- BGP router will send packet to one out of three ISPs of IITB

Summary

User	netmon configuered	netmon not configuered
Student	packets will get service through netmon proxy	packet will get internet service through transparent proxy only for http & https. For rest, packet will be redirected to internal firewall
Faculty	packets will get service through netmon proxy	packets will be blocked at transparent proxy after seeing user credentials
Staff	packets will get service through netmon proxy	packet will get internet service through transparent proxy

2.4 Academic Areas network

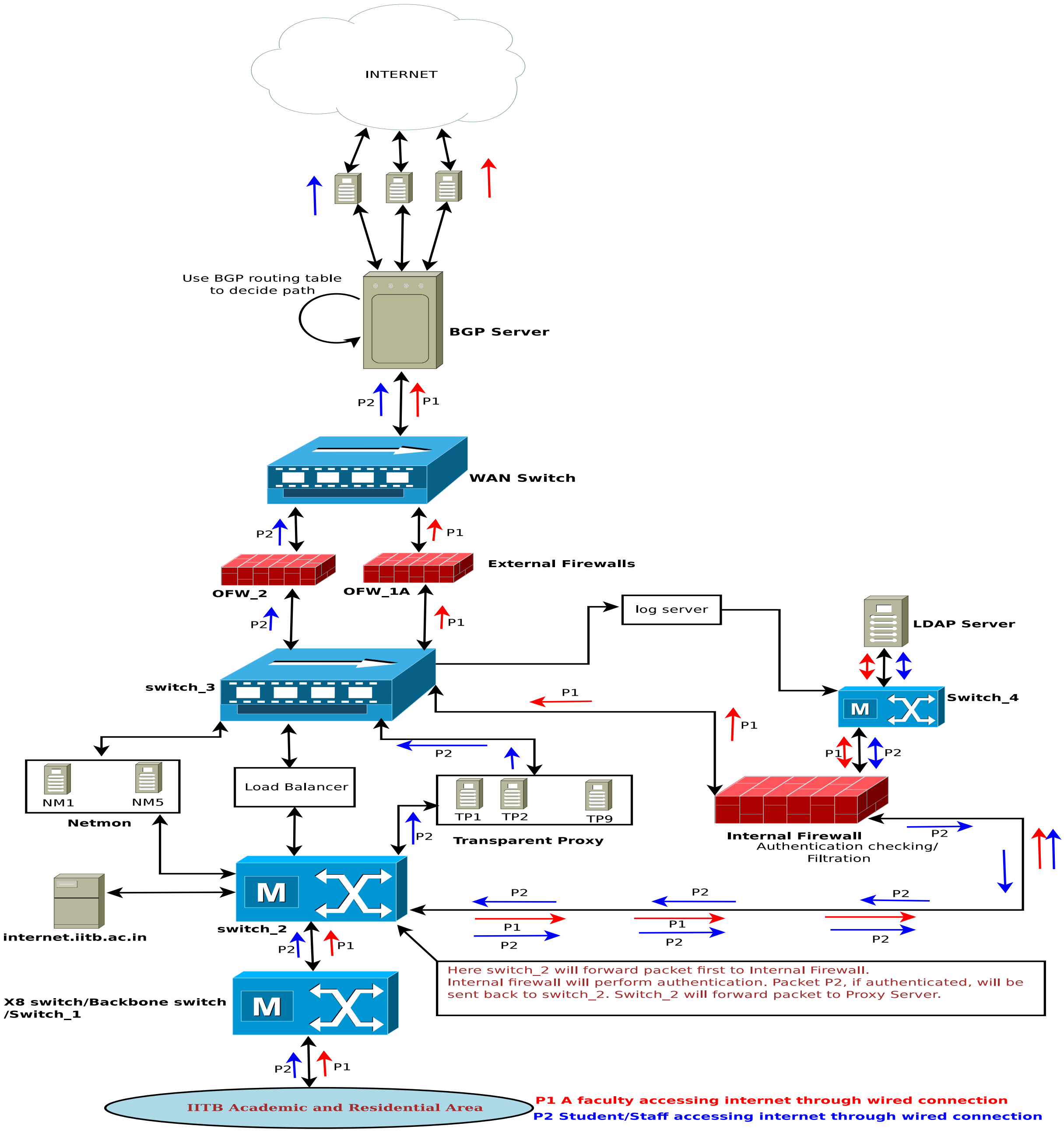

INTERNET
Use BGP routing table to decide path
BGP Server
P2
P1
WAN Switch
P2
P1
External Firewalls
OFW_2
OFW_1A
P2
P1
log server
LDAP Server
switch_3
Switch_4
P1
P1
P2
P1
P2
NM1
NM5
Netmon
Load Balancer
TP1
TP2
TP9
P2
Transparent Proxy
Internal Firewall
Authentication checking/ Filtration
P2
P2
P2
P2
P1
P1
P2
P2
P2
internet.iitb.ac.in
switch_2
P2
P1
Here switch_2 will forward packet first to Internal Firewall. Internal firewall will perform authentication. Packet P2, if authenticated, will be sent back to switch_2. Switch_2 will forward packet to Proxy Server.
X8 switch/Backbone switch /Switch_1
P2
P1
IITB Academic and Residential Area
P1 A faculty accessing internet through wired connection
P2 Student/Staff accessing internet through wired connection

Figure 2.3: IITB Academic and Residential Area Network Configuration Setup

- Any packets which are generated from academic or residential area will firstly go to switch_1.

- Switch_1 will process the packets by looking at source and destination IP, from which it will identify that this packet is comming from academics or residential area, and determines should it go to firewall or not.
- Switch_1 will forward this packet to switch_2 .
- Switch_2 then will forward this packet to firewall.
- Firewall will check for authentication, here there are two case:
    - *Not authenicated*, here firewall will authenticate pacekt IP through internet.iitb.ac.in; which authenticats it with help of LDAP server.
    - *Already authenticated*
- After authentication, depending on user type packet will follow different way to switch_3 as follow:
    - For *student or staff and destination port 80 or 443 :-* packet will be sent to Transparent Proxy by Firewall through switch_2. Transparent proxy will forward packet to switch_3
    - For *faculty & service other than 80 or 443 :-* Some filtration will be done and packet is transferred to switch_3.

- At one point switch_3 will receive packet from either transparent proxy or internal firewall.
- Switch_3 will forward packet on basis of destination MAC address.

- External Firewall will do natting and will forward packet to WAN switch.
- WAN switch will forward this packet to BGP Server.
- BGP Server will forward this packet to one of the ISP based on routing table.

Summary

User	**wired**	**wireless**
Student	packets will first go to firewall then will be forwarded to transparent proxy	packet will get internet service through transparent proxy
Faculty	packet will first go to firewall & will get service from firewall only	packet will get internet service through transparent proxy
Staff	packets will first go to firewall then will be forwarded to transparent proxy	packet will get internet service through transparent proxy

2.5 Wireless Network

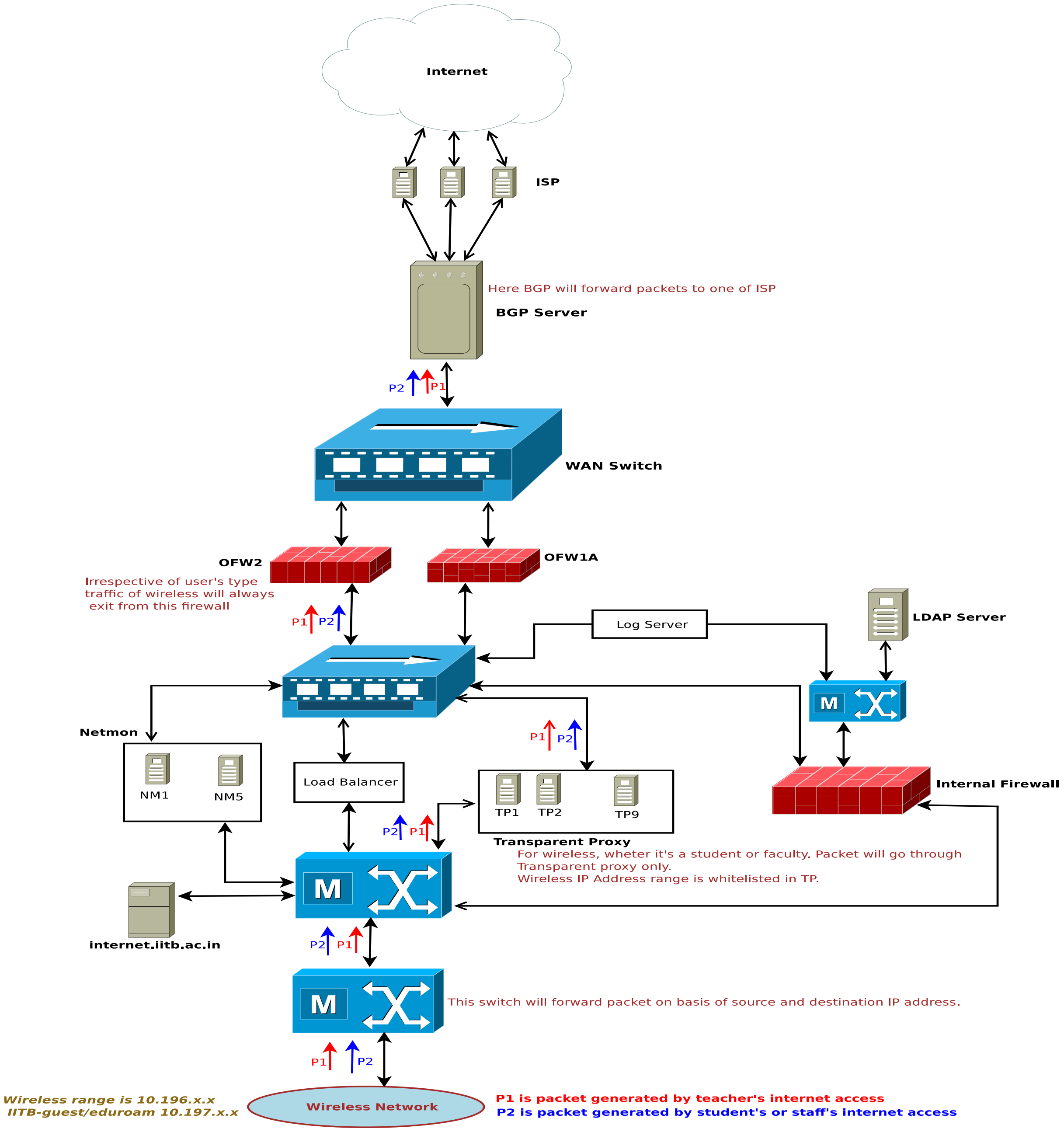

Internet
ISP
Here BGP will forward packets to one of ISP
BGP Server
P2
P1
WAN Switch
OFW2
Irrespective of user's type
traffic of wireless will always
exit from this firewall
OFW1A
P1
P2
Log Server
LDAP Server
M
Netmon
NM1
NM5
Load Balancer
P1
P2
TP1
TP2
TP9
Transparent Proxy
For wireless, wheter it's a student or faculty. Packet will go through
Transparent proxy only.
Wireless IP Address range is whitelisted in TP.
Internal Firewall
P2
P1
M
internet.iitb.ac.in
P2
P1
M
This switch will forward packet on basis of source and destination IP address.
P1
P2
Wireless range is 10.196.x.x
IITB-guest/eduroam 10.197.x.x
Wireless Network
P1 is packet generated by teacher's internet access
P2 is packet generated by student's or staff's internet access

Figure 2.4: IITB Wireless Configuration Setup

– IITB network has reserved one subnet for user connected through IITB Wireless. User connected with wireless will get IP address in range of *10.196.x.x or 10.197.x.x*

– Packet generated by IITB Wireless connection will first come at Switch_3.

- This switch will see that source IP lies in range of *10.196.x.x or 10.197.x.x* it will put physical address of transparent proxy as destination MAC address

- Switch_2 will forward packet on basis of MAC address

- Important point here is packet coming through wireless connection will always go through Transparent proxy whether it is faculty or student.

- If connected through wireless faculty can not access service other HTTP and HTTPs web pages.

- *10.196.x.x* and *10.197.x.x* is white-listed in Transparent Proxy. Any packet with source address in this range will be forwarded without any authentication.

- Transparent proxy will initiate a new connection with destination web server on behalf of user via switch_3 and will return response to user back.

- Switch_3 will forward packet to OFW_2

Summary

Location	**Faculty**	**Student**	**Staff**
Hostels	user will get internet service through transparent proxy. Transparent proxy will not block traffic for faculty like wired connection	user will get internet service through transparent proxy	user will get internet service through transparent proxy
Academic Area	user will get internet service through transparent proxy	user will get internet service through transparent proxy	user will get internet service through transparent proxy
Residential Area	user will get internet service through transparent proxy	user will get internet service through transparent proxy	user will get internet service through transparent proxy

2.6 Outside Access of cse.iitb.ac.in

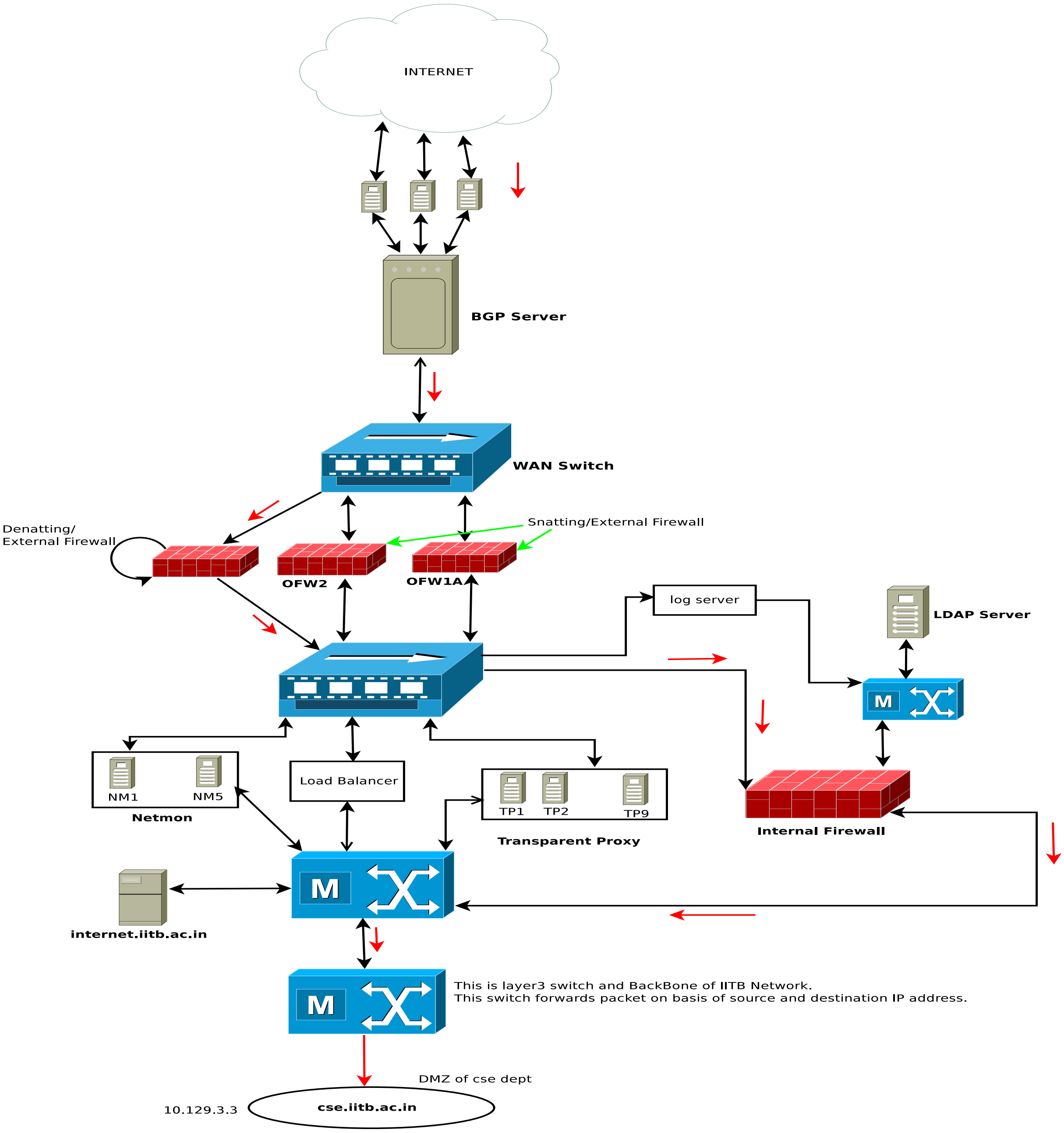


Figure 2.5: IITB Wireless Configuration Setup

– If any user from outside of IITB network try to connect cse.iitb.ac.in then first packet will come at BGP server.

– BGP server forwards this packet to External Firwall OFW_2.

- After processing at OFW_2, it forwards packet to the internal firewall through WAN switch and switch209.
- Internal firewall forwads this packet to the cse.iitb.ac.in through switch_2 and switch_200.

2.7 IP Anonymization

IP anonymization is considered more in the sense of a privacy feature that is important for protection of the privacy of visitors from analytic. Before handling logs to analytic for any kind of analysis It is very important to hide actual IP address of visitor. By no means, person who is doing analysis should be able to reach source of IP by looking at logs. IP anonymization is a way or technique to change all IP addresses in a log in a way so that although users remain anonymous, but usage trends remain extractable.So Whenever logs are sent for any kind of research or parsing it is advisable to anonymize all IPs at earliest possible stage.

There are many IP anonymization techniques. Each technique has it's own benefit and loopholes. Some common IP anonymization techniques are described below by help of diagrams. In each diagram, a central box is an anonymization tool which will get an input IP address as input and will produce an anonymous IP address at output. Now it is implementation of this central box which varies from one technique to other technique

- Black Market Effect
  - Changes all IP to single output IP
  - It is very easy to implement.
  - Leave no scope for IP based analysis

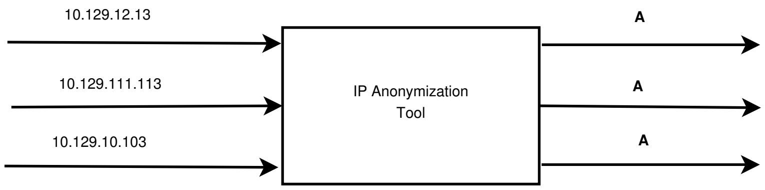


Figure 2.6: Black Market IP anonymization

- Truncation
  - Truncates last two bytes of IP address as shown in figure
  - Since prefix is maintained it is good for research and analysis
- Random Permutaion of IP
  - Does random one to one mapping of IP address

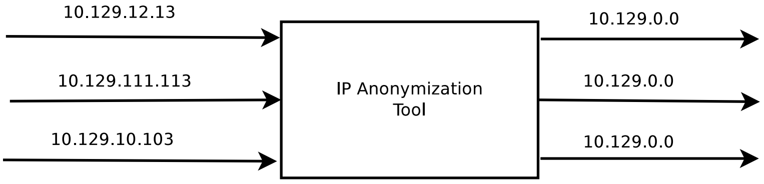


Figure 2.7: Truncation way of IP anonymization

  - Allows correlation between IP unlike black market technique but destroys structure
  - This technique also leave no scope for analyzer to analyze IP based

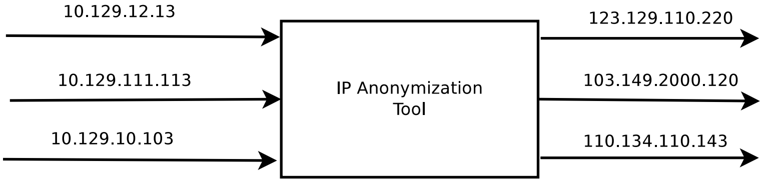


Figure 2.8: IP anonymization by Random Permutaion

- Prefix Preserving anonymization
  - Best anonymization technique for analysis as well as optimal level of IP anonymization
  - P is a prefix-preserving anonymization function if and only if: for all IP addresses x and y: If x and y have first k bits are same than P(x) and P(y) also share first k bits.

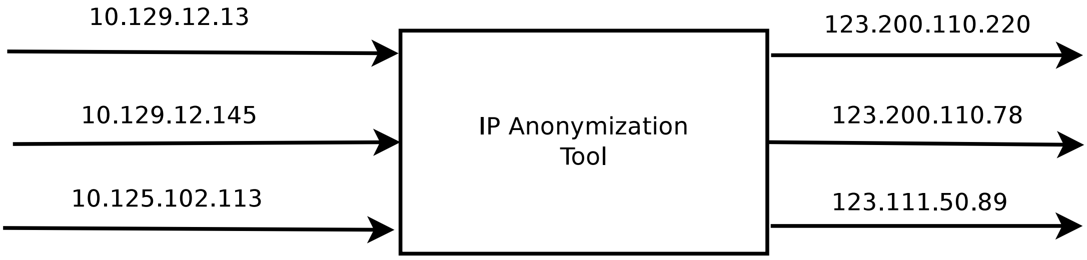


Figure 2.9: Prefix Preserving IP anonymization

2.8 Zabbix

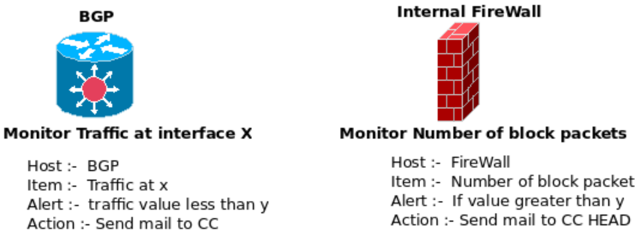
2.8.1 Introduction

What is Zabbix? The straightforward answer to this question would be Zabbix is a monitoring tool which is highly extensible, efficient and reliable. Using Zabbix server, and by installing Zabbix agents on servers or any other network devices you can monitor devices remotely and whenever something bad will happen admin will get an alert. Zabbix lets you define what device you want to monitor, what feature on that device you want to monitor. It also let you define excellent, good or bad behavior of a device and whenever a device will behave badly according to defined value of bad, Zabbix will report to admin. For example, you want to monitor traffic on BGP router and you want that whenever traffic on BGP router exceeds a threshold, an alert should be triggered because you suspect that this is a situation where some kind of bombarding or BGP overloading is there. For this kind of situation, Zabbix is an ideal tool. You install a zabbix agent on BGP server and define incoming traffic as an item to monitor and also define a value above which you want an alert. Now, whenever Incoming traffic will exceed that value an alert will be sent to concern authority.

2.8.2 Zabbix terminology

- **Host:** A Host is device which is being monitored by Zabbix Server. Zabbix agent is installed on host device. A unique hostid is associated with each host.
- **Items:** items are the feature of host device which is monitored by Zabbix Server. For example, an item can be number of user logged in , Traffic rate at BGP, number of processes at host etc. Each item is recognized by an itemid in Zabbix Database Server.
- **Alert:** Zabbix not only monitors network devices but it can also act like an alarming device. Zabbix lets network administrator define condition or values under which an item can be considered in situation of good, bad or worse. Now as value of item will fall under range of good or bad or worse ,it would be displayed at web interface.
- **Action:** Network administrator can also define action for any particular or threshold value of item. As value of item will cross this threshold this action will be taken. Action can be anything like shutting off device, killing process or send mail to admin.

Following diagram explains meaning of all four terms in a real example


BGP
Internal FireWall
Monitor Traffic at interface X
Monitor Number of block packets

Host :- BGP
Item :- Traffic at x
Alert :- traffic value less than y
Action :- Send mail to CC

Host :- FireWall
Item :- Number of block packet
Alert :- If value greater than y
Action :- Send mail to CC HEAD

Figure 2.10: Zabbix terminology

2.8.3 Zabbix Architecture

Figure 2.11 describes architecture of Zabbix framework. Zabbix Architecture is mainly composed of following entities:

- Zabbix Server
- Zabbix Database Server
- Zabbix Proxy Server
- Hosts

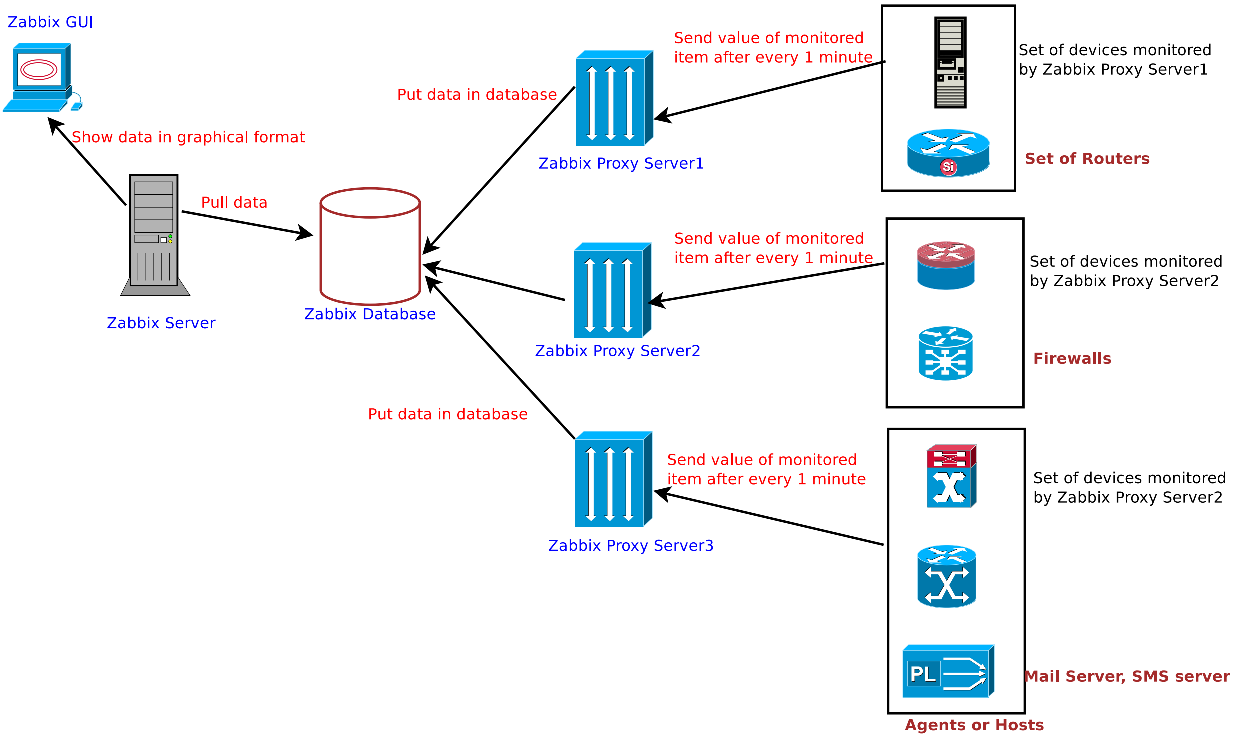

Zabbix GUI
Show data in graphical format
Pull data
Zabbix Server
Zabbix Database
Put data in database
Zabbix Proxy Server1
Send value of monitored item after every 1 minute
Set of devices monitored by Zabbix Proxy Server1
Set of Routers
Zabbix Proxy Server2
Send value of monitored item after every 1 minute
Set of devices monitored by Zabbix Proxy Server2
Firewalls
Put data in database
Zabbix Proxy Server3
Send value of monitored item after every 1 minute
Set of devices monitored by Zabbix Proxy Server2
PL
Mail Server, SMS server
Agents or Hosts

Figure 2.11: Zabbix Architecture

- **Hosts:** These are the devices which are being monitored by Zabbix server. These are called hosts or agents. Information is gathered at hosts and polled by the server.Within a device admin can also define what he wants to monitor. For example, for a BGP router he might be interested in bandwidth at particular interface, for proxy server he might be interested in number of connection at any time, for some other particular device he might be interested in disk free space. The thing which you want to monitor under a host are called items. Each host has a hostid and each item has a itemid.

  In zabbix there are multiple ways of monitoring a hosts. Most popular is by installing a zabbix agent at host. Another ways are SMTP monitoring, ping, IPMI checks.

- **Zabbix Server:** This is central node of whole Zabbix architecture. It controls all other component. Every host's config file which contains it's IP addresses stored at this server

- **Zabbix Proxy Server:**These are optional. If network or organization is really big then multiple proxies can be installed. Each proxy can monitors a specific set of devices. Proxy server will receive value from hosts and will put in database.

- **Zabbix Database Server:** It is MySQL or PostgreSQL as backend database. It can be on same machine as Zabbix server or can be at different machine. This database holds various number of tables each serving a purpose. Like host table contains host related detail, item table contains items on each host, similarly, history table contains item value received from hosts.

2.8.4 Limitation of Zabbix

Although Zabbix does a very handsome job when its come to monitor network devices. Besides monitoring it also let admin define alert and action feature but still there are some of limitations of zabbix when it comes to generating statistics and saving long term trends. Some of few limitations are:-

- Although it shows traffic data upto last 1 year, but it is not meant for this purpose. It's main purpose is monitoring and alerting if something goes wrong.

- Besides basic statistics like min, max or average traffic, it does not show other important stats like down time trends of network devices

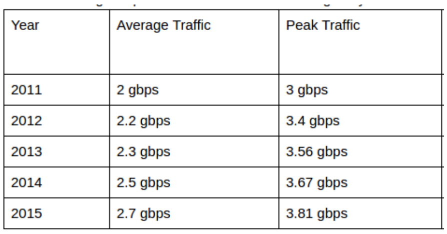
- Only graph representation is of no use if tool can not analyze data.

- There is no scope for any kind of log and text based analysis.

Questions which can't be answered by Zabbix

- Any kind of statistics in discrete period of time?

  E.g :- From 2011 to 2015 Show avg traffic at BGP in month of December. Following kind of table is not possible through zabbix



Year	Average Traffic	Peak Traffic
2011	2 gbps	3 gbps
2012	2.2 gbps	3.4 gbps
2013	2.3 gbps	3.56 gbps
2014	2.5 gbps	3.67 gbps
2015	2.7 gbps	3.81 gbps

Figure 2.12: Statistics in Dirscrete Period

- Even for a selected period, It can't tell any statistics other than min,max,avg.
- No support for Down- Time trend.

  In following picture, it can be easily seen from graph that minimum traffic is zero but according to statistics part minimum traffic is 114.55kbps.

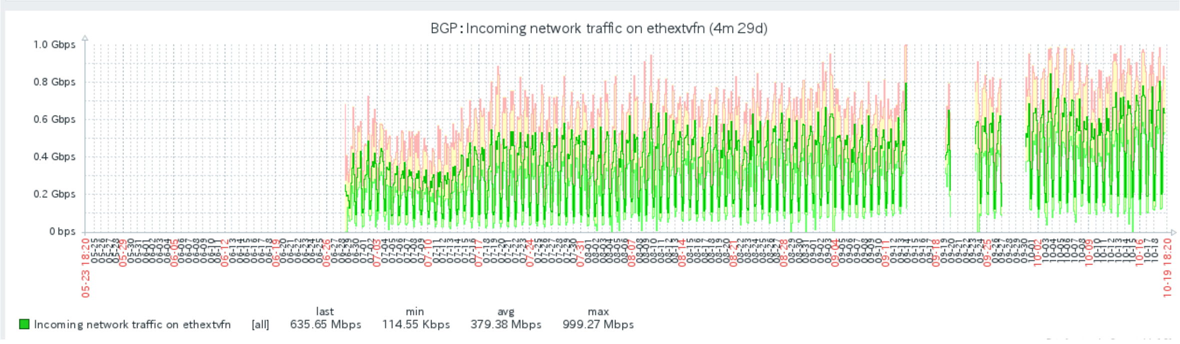


Figure 2.13: No support for Down Time Trends

2.9 Introduction to Archival Tool

2.9.1 Archival Tool Definition

What is an Archival Tool? A tool which can be used to remove raw data while maintaining reporting functionality by extracting long-term trends from raw data.

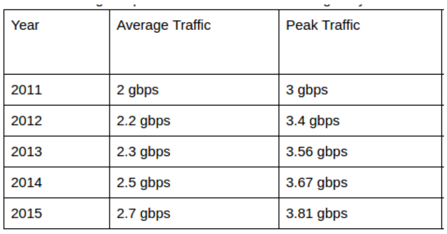
It is not feasible to keep raw data for long terms as raw data comes in huge size and takes a lot of space and secondly, raw data is not of that much use over a long period of time. It is trends or statistics that matter over a long period of time. For example, If we talk about traffic we would like to have minute level detail of traffic for any host for a shorter duration because raw data is very useful to go in detail and solve any kind of traffic crisis,if happens but for 15 years earlier minute level traffic detail become useless. At that time, it is important to have trends or statistics which we would like to see. So basically an archival tool , extracts long term trends from raw data and removes raw data. Granularity at which long term trends are stored may vary depending on time. For example, admin might like to have raw data for one month then before one month trends on daily basis and before that trends at yearly basis. As data become older its granularity will change.

2.9.2 Scenario where an Archival Tool is required

Following are few of scenario in which it is very important to have an archival tool

- Helpful in taking decision by looking at past trends

For example, Lets assume today CC-head wants to increase IITB network bandwidth by some amount. How do you think how he/she can take decision ? First question that comes into mind how much increase would be sufficient? At that time CC-Head would like to see previous years usage of network bandwidth and according to past usage it would become easy for him/her to take decision. For example if he/she has following kind of statistics it is easy to predict future usage of network bandwidth.



Year	Average Traffic	Peak Traffic
2011	2 gbps	3 gbps
2012	2.2 gbps	3.4 gbps
2013	2.3 gbps	3.56 gbps
2014	2.5 gbps	3.67 gbps
2015	2.7 gbps	3.81 gbps

Figure 2.14: Last 5 years statistics

- Helpful in predicting future network behaviour

  Archival tool also helps in predicting future network behaviour. For example, If down time trends are stored by tool and admin gets to know most of down time happens in month of monsoon and number of switches that corrupts are this much admin can prepare for all that scenario in advance.

Chapter 3

Proposal of an Archival Tool

In last chapter we discussed zabbix. It's working, it's purpose and it's limitation. We also saw despite having so many features there are some questions which can't be answered by Zabbix. We also discussed what is an Archival tool and need of an archival tool in context of IITB. In this chapter we will propose an archival tool - **IITBLTTA - IITB LONG TERM TREND ARCHIVAL** tool. We will see purpose of archival tool in detail. What kind of functionality it can provide . We will also see concept of periodic time interval query for statistics and how this tool can fulfill that purpose and how it can answer all those questions which were unanswered by Zabbix.

3.1 Purpose of Archival Tool

As discussed earlier, Zabbix is meant for alerting and monitoring not for storage of data. We also see how raw data become useless after a particular time point and how it is important to have statistics out of raw data. Main purpose of archival tool is to eliminate raw data after a time period and save long term trends out of raw data. Later, these long term trends can be used to get useful information.

3.2 Tool Functionality

Main purpose of tool is to store and display **Traffic Intensity Based Statistics**. This tool can be used to save traffic related statistics and later these statistics can be used to get information, take decision or find any kind of abnormality in network behaviour. These kind of data can also be used for future prediction related to network. Following are some functionality which this tool can provide for network traffic

3.2.1 Statistics over a continuous period of time

In these kind of statistics, trends would be displayed between a given range of time period. A time- period would be specified by user and desired traffic statistics would be displayed in that time-range. Following are few examples of user queries which lie in this area.

- **User Query:** From 1 Jan,2017 to 31 Jan,2017 tell Max, min and average traffic for BGP at eth0 interface .
  Tool Response:

From	To	Max Traffic	Min Traffic	Avg Traffic
1 jan,2017	31 jan,2017	312.8 Mb	7.8 Mb	120Mb

Table 3.1: Statistics over a period of Time

- **Time of Max and Min Traffic**
  User Query: From 1 Jan, 2017 to 31 Jan,2017 tell me max and min traffic and when did both occured?
  Tool Response

From	To	Max Traffic	Min Traffic	Time of Max Traffic	Time of Min Traffic
1 jan,2017	31 jan,2017	312.8 Mb	7.8 Mb	7/1/2017 12:33	22/1/2017 15:12

Table 3.2: Time of occurrence of max and min traffic

3.2.2 Down Time Trends

These statistics are corresponding to zero traffic at particular devices.

User Query: From 1 Jan, 2017 to 31 Jan,2017 Tell what % of time traffic was zero, also tell longest down time and when did it occurred ?
Tool Response

From	To	Zero traffic %	Max Down Time	Time
1 jan,2017	31 jan,2017	10% of time	7 Hours	3 jan 17:30

Table 3.3: Down Time Trends

3.2.3 Statistics over Discrete Time Period

Unlike first subsection, here time range for which user wants statistics is not continuous. Here user won't get a single result but instead would get number of results corresponding to different different chunk of time-range. For example "show statistics for 1st of every month". Here time range i.e Date 1st of every month is discrete and user will get 12 results, each result corresponding to 1st of every month. Few examples of such query are as follow

- **User Query:** Tell Stats on monthly basis for year 2015
  Tool Response:

Month	Max Traffic	Min Traffic	Avg Traffic
Jan	312.8 Mb	0 Mb	167Mb
Feb	354.8 Mb	0 Mb	187Mb
March	388.4 Mb	0 Mb	183Mb
April	371.1 Mb	0 Mb	191Mb
March	377.6 Mb	0 Mb	171Mb
May	312.0 Mb	0 Mb	162Mb
June	222.8 Mb	0 Mb	111Mb

Table 3.4: Traffic Statistics on Monthly basis

Significance of these statistics
These kind of statistics clearly gives an overall idea about traffic on monthly basis. It can be easily seen on an average consumption is nearly 200 Mb. Besides this it can also be seen very easily that in month of June network consumption is less may be due to fact that student leave for home in this month. So, during this duration many server can be shut down.

- **User Query:** Tell Max Traffic Statistics for December from 2011 to 2016
  Tool Response

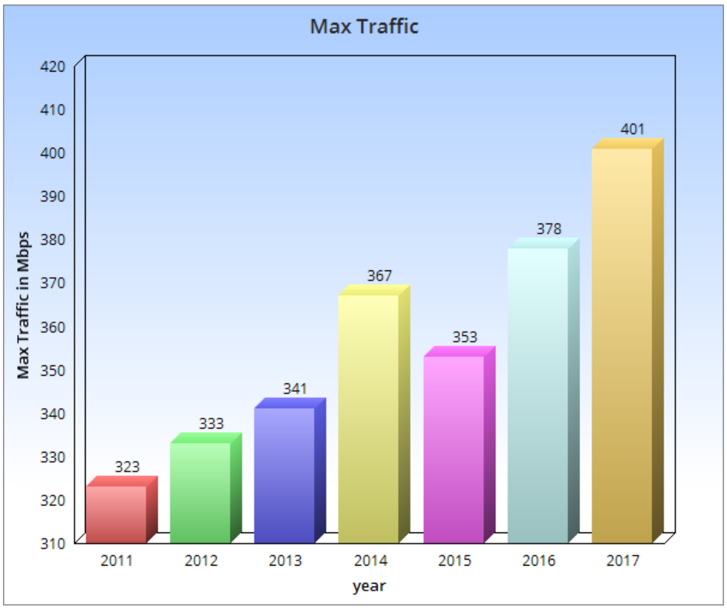


Figure 3.1: Hypothetical graph showing December Month Statistics

Significance: At start of December 2017 if admin can get following kind of statistics, he can have fair amount of idea about network load for this month and he can be better prepared.

3.2.4 Trends showing traffic comparison among devices/Interfaces

- Traffic Comparison among interfaces of same device

  A host is configured to have traffic on multiple interfaces. For BGP along there are more than 10 interfaces on which traffic is monitored. So any kind of statics that can let us compare traffic among these interface would be very useful in deciding load balancing between interfaces for that devices.
  User Query: Tell Maximum traffic for this week per day for all interface of BGP
  Tool Response

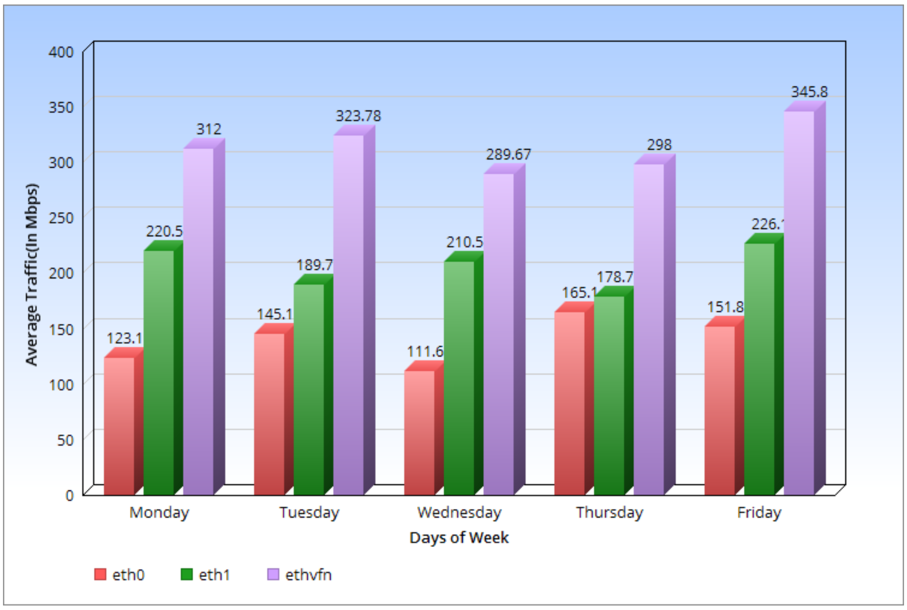


Figure 3.2: Comparison of Traffic among interfaces of BGP(Hypothetical Graph)

- **Traffic statistics Comparison among devices**

  In IITB, many devices are divided into virtual machines. For example, Transparent Proxy have 10 instances Similarly, Netmon has 6 instances and it is very important to do proper load balance among all these instances. So any kind of statistics which can compare two or more devices would be very useful to see load balance status among all devices.

  User Query: Tell total traffic for Tp0 to tp5 for today.

  Tool Response

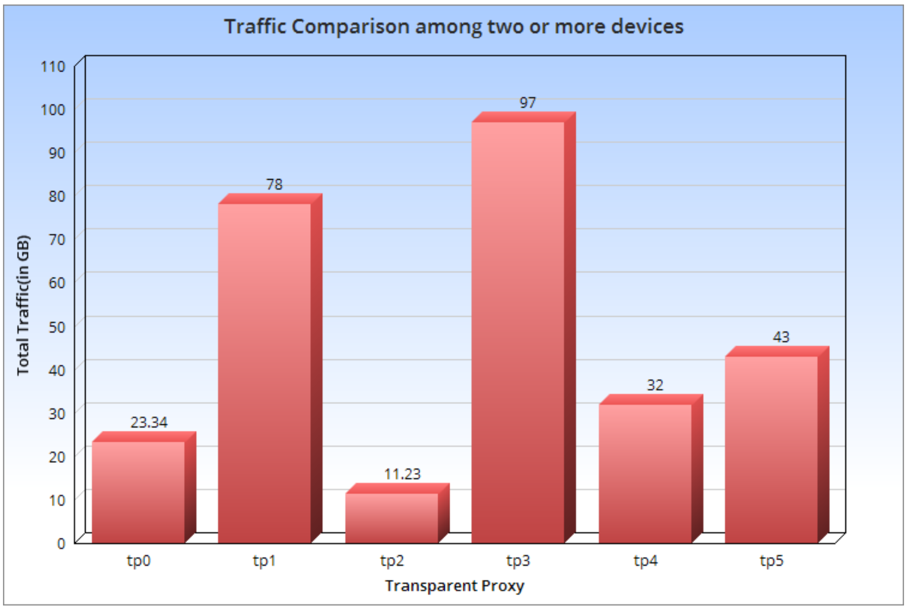


Figure 3.3: Traffic Comparison among Transparent Proxies(Hypothetical Graph)

  Significance : Now this figure clearly tells that for today tp1 and tp3 were highly loaded while other had very less traffic. If similar results appear for 3 4 more days, CC can balance load among tps either manually or through some other means.

These were few purposes which this tool can offer. Over a longer period of time when good amount of data is calculated even advanced machine learning algorithm can also be applied to get future behaviour of network. Queries like expected traffic in next day or expected down time trend for upcoming month can also be served.

Chapter 4

Tool Design, Architecture and working

4.1 Tool Design and Architecture

IITBLTTA has two main components. One component is back-end component whose task is to generate and maintain database at back-end. Other is front-end part whose task is to take input from user, parsing the query, retrieving result from back-end database and display result at back-end. For better understanding of tool - design we will discuss both of these two component in two separate subsections. In each subsection we will describe design of both parts and finally we will combine both parts and show complete architecture. We will also discuss working of tool in an entire different section where we will discuss tool operation by taking an example of data archival at multiple granularities.

4.1.1 LTTA-PE

Here PE stands for Periodic Events. This component is responsible for all kind of periodic events taking place at back-end. This component generates and maintains all kind of database tables at back-end. This component also have multiple schedulers which help in maintaining database. These schedulers acts like kind of cron jobs which automatically gets execute after a specified period of time. Following diagram describes design of this part.

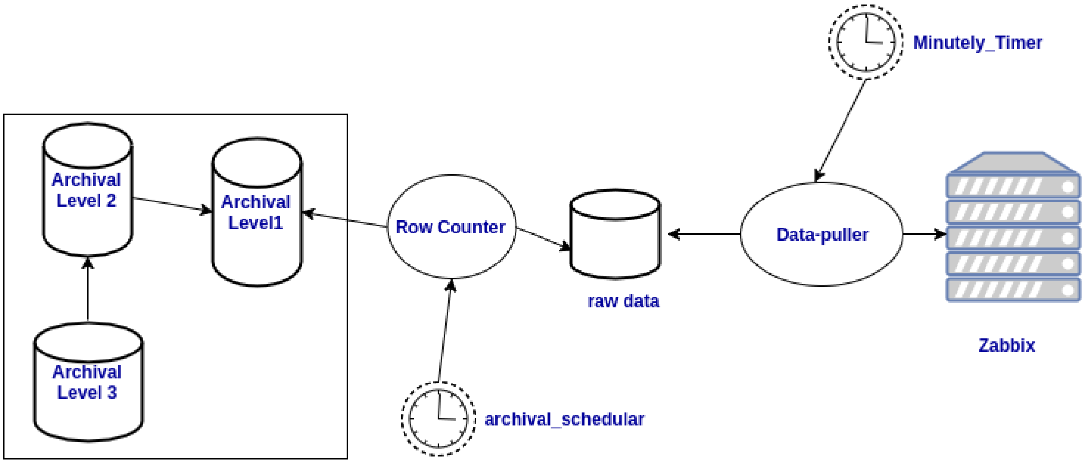


Figure 4.1: Design Architecture of back-end part

Components of LTTA-PE

- **Minutely-Timer** This component is kind of cron-job. It will get execute after a specified period of time. When first time this component will run it will create an instance of Data puller and then it will use this instance to execute data puller every time. In other word, this component will run after a fix period of time and in each run it will execute Data puller component.

- **Data Puller** This component will get executed by a trigger generated by Minutely-Timer. This componet has two job one is to get raw data from zabbix server and then store raw data in database. Minutely timer will bring this component in execution state after every one minute and on execution it will pull raw data from zabbix for each host and items. Once data is retrieved it will store data in raw data table of IITBLTTA.

- **archival_schedular** Just like Minutely-Timer it is also a cron - job. Time after which this job will get executed depends on granularity of first archival table. JOb of this timer is to archive the statistics.

- **Row Counter** Row Counter is responsible for generation of all archival tables. Once it trigger, it will count number of rows in each table and if any table contains more number of rows than it is supposed to keep then this component will aggregate these extra rows and will put in next archival level. For example if raw data table is supposed to keep data for last 4 hour. Then if at any time raw data table has more number of entries than 240 then all extra entries will get aggregated and would be placed in next level.

  Although in figure Row Counter is placed in between raw_data and first archival table but in actual it is also present between all archival tables and it works same as described above.

- **Archival Levels** These are multiple archival tables in database. These archival tables stores data at different different granularity. By granularity we mean time - period over which trend are aggregated before storing in that table. hourly_archival table would have traffic statistics aggregated at hourly level. Table with lowest granularity will get aggregated data from raw-data table. Other archival tables will get data from archival table with just higher granularity.

4.1.2 LTTA FE

Here FE stands for Front End. This is second component of IITBLTTA. It deals with user interface and result display. Front-end is based on MVC i.e. Modal View Controller where View part is composed of user interface and result display part while modal part deal with database part. We can also say that modal part of this component is actually final result of IITB-PE component. Controller part consist of Query parser , stats fetcher and result calculator part. Whole of this architecture is shown pictorial in following diagram.

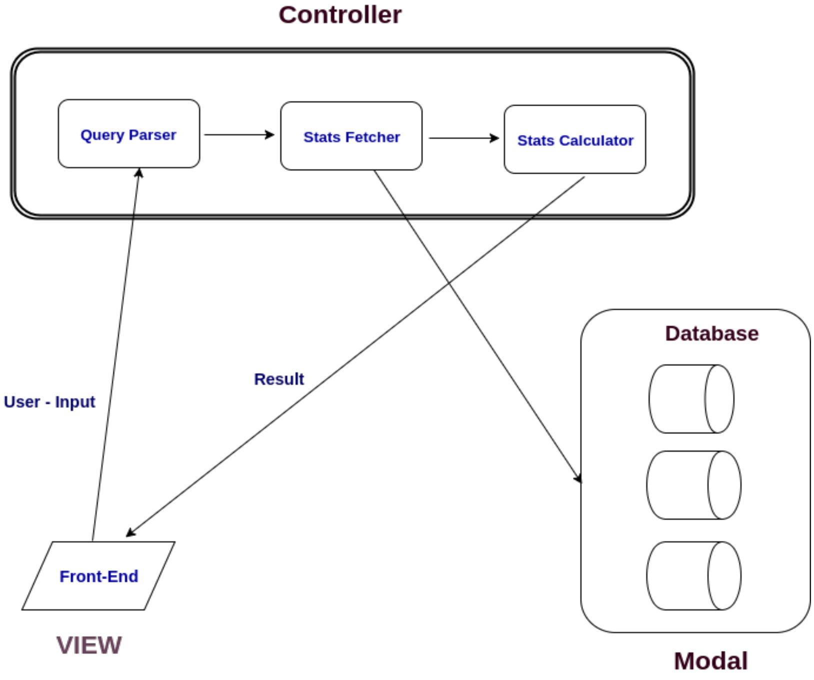


Figure 4.2: Front-end Design for IITBLTTA

Components of IITBLLTA FE

– **UI** This is a part of view section. This part takes input from user in form of dates, desired statistics , host and all other parameter. This part on receiving input, sends input-form to Query Parser after validating all inputs. It has two part one for user interaction and other for Admin.Admin has a privileged section where he can see all archival tables related detail and can make changes to table. There are many other tasks that can be performed. We will discuss both part in greater detail in upcoming chapter with screen-shots.

- **Query Parser** This component receive user input at back-end. It parses all those inputs and on basis of selected input it forms valid mysql queries and passes these query to stats fetcher component.
- **Stats Fetcher** After receiving mysql queries from Query Parser this component receives statistics from IITBLTTA database. To get desired statistics from database first it need to know is among so many archival tables in which table desired statistics lie. First it gets table name and on basis of table name it fetches desired data.
- **Stats Calculator** Now since stats are fetched from multiple tables. For example, user might have asked for statistics for a period which lies among two or more than two tables. In that case after fetching the data it also has to do further calculation to get final statistics. Once result is calculated it forward statistics to front-end to display.

4.2 Working of Tool

This tool works like a **sliding window**. To explain concept of sliding window let us take an example. Let us assume that network administrator specify that for last four hour he wants to maintain traffic data at minute level i.e raw data. Before 4 hour there is no need of raw data , Instead of raw data, aggregation of data on hourly basis should be maintained and before 3 days there is no need of keeping data at hourly basis, just aggregate data at at daily basis and stores those daily trends. Following figure illustrates admin demand Now assume that p represents current time then admin wants previous four hour data i.e (p-4H) should be stored in raw data table. Previous 3 days data i.e (p-4H-3D) is stored at hourly_archival . 15 days before that i.e data between p-4H-18D and p-4H-3D is stored in daily_archival and data older than this period is in weekly_archival table. Following table sums up whole architect discussed so far

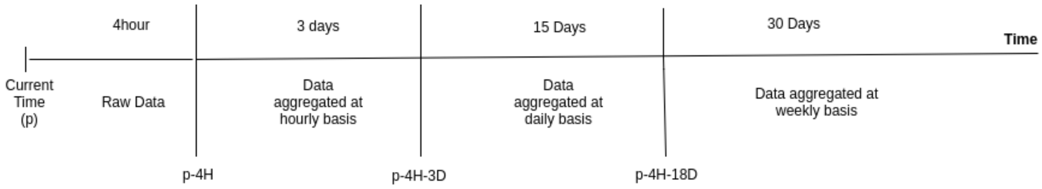

4hour
3 days
15 Days
30 Days
Time
Current Time (p)
Raw Data
Data aggregated at hourly basis
Data aggregated at daily basis
Data aggregated at weekly basis
p-4H
p-4H-3D
p-4H-18D

Figure 4.3: Concept of Archiving explained

Granularity	Time-Period	Table Name
1 Minute	4 Hour	raw_data
1 Hour	3 Days	hourly_archival
1 Day	15 Days	daily_archival
1 Week	30 days	weekly_archival

Table 4.1: Complete Table describing Archiving

In this table raw_data is supposed to keep data at minute level for last 4 hour. It means number of entries in this table should be 4*60 = 240 entries. Now as this table will possess 60 more entries than 240 entries, then oldest 60 entries i.e 1 hour data will get aggregated and will be stored in hourly_archival table. Similarly, hourly_archival table is supposed to keep 3* 24 entries in it. Now as it will contain 24 extra entries , oldest 24 entries i.e 1 day will get aggregated at would be stored at daily_archival. Now this is very much like a **sliding window** as data is getting older it is getting shifted from one table to other table and granularity is getting increased with time-period.

Following diagram explain working of tool in pictorial way

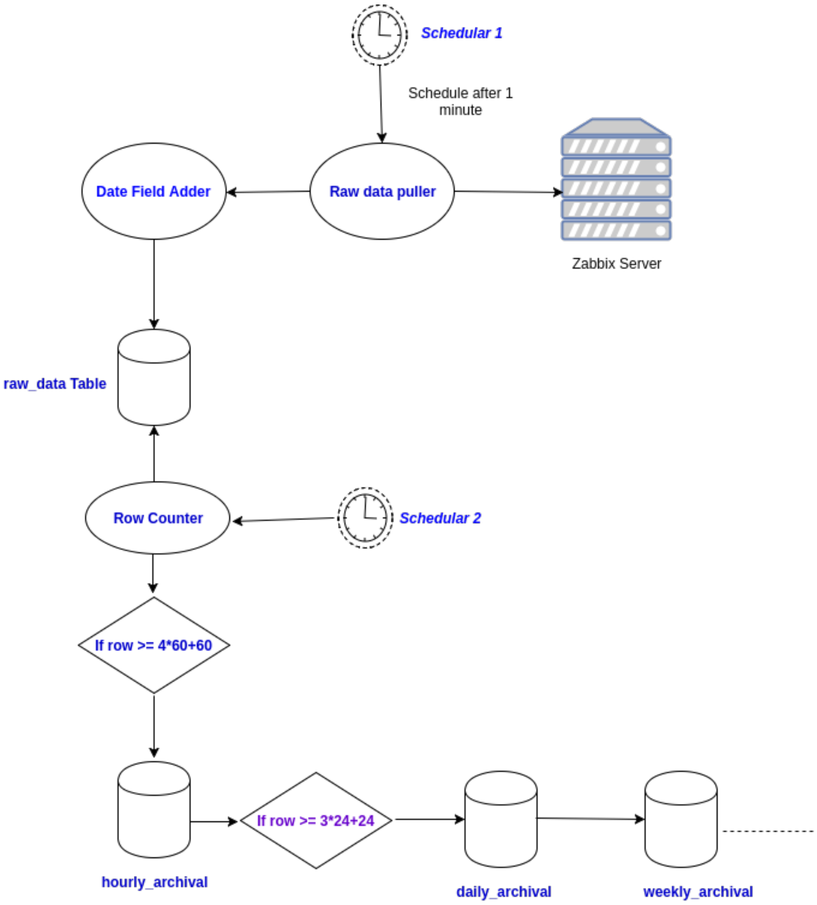


Figure 4.4: Working of IITBLTTA

4.3 Database Architecture

IITBLTTA database contains following databases and tables in respective databases

1. **IITBLTTA**

   This database contains raw_data table and all other archival tables. Number of archival tables are dynamic. According to convience and need admin can increase and decrease number of archvial tables in database. List of tables present in this database are following

   (a) **raw_data:** This table holds raw data for traffic at each host. This table receives data from zabbix for each minute and store it without any processing. Table schema for this table is as follow:-

hostid	itemid	clock	Time	value
10125	10119	1496561685	04/06/2017 20:04	3456891

Table 4.2: Schema for raw data

(b) **Hourly_archival:** Data in this table is aggregation of raw data at granularity of an hour. Data older than time-period, for which raw_data is supposed to keep data , is aggregated and put in this table. For example, if admin has specified that for last four hour data is kept at minute level then data older than four hour will be aggregated and be put in this table. Name of columns in table;

- itemid
- clock
- Time
- Max Traffic
- Min Traffic
- Avg. Traffic
- Total Traffic
- zero Traffic: Number of times, value of traffic was zero between clock and clock + 3600

itemid	clock	Time	Max	Min	Average	Total	Zero Traffic
10119	1496561685	04/06/2017 20:04	12658901	6789	68904.45	690532789	23

Table 4.3: Schema for an archival table

(c) **Daily_archival:** Structure of this table is same as that of Hourly_archival table. This table is aggregation of data at daily basis. It takes oldest 24 entries from Hourly_archival: table and stores statistics out of it.

2. **hosts**

This database stores information related to hosts i.e devices on which traffic is being captured and items i.e interfaces on which traffic is being measured. List of tables in this database are following

(a) **hosts:** This table contains information related to devices on which traffic is being measured. A sample of this table is as follow:

host	hostid
BGP	10125
dwar	10123
ifwa	10140

Table 4.4: Sample of hosts table

(b) **items :** This table contains information related to interface and direction of traffic on that device. A unique number known as itemid is associated with each set of device, interface and direction of traffic. A sample of this table is as follow.

itemid	hostid	name	key_
25279	10125	ethextnkn	Incoming traffic
25288	10125	ethextnkn	Outgoing traffic
25282	10125	ethexttcl	Incoming traffic
24199	10109	virbr0	Outgoing traffic
24225	10110	eth0	Incoming traffic

Table 4.5: Sample of items table

3. **admin**

This Database contain only one table named admin. This table contains information related to admin. Only privileged people can access portion of web interface where one can make changes to archival tables and can add new hosts and items.

4.4 Memory requirement Analysis for database

4.4.1 Need of Memory Requirement Analysis:

Since IITB LTTA is going to save statistics over a long period of time one thing we were sure about that it's going to be a database sensitive tool. For example, let us assume we want to save statistics for traffic at different devices of IITB and granularity at which we want to save statistics is one hour. IITB monitors traffic at following five devices and within each device their is number of interface at which traffic is monitored.

BGP router	26 items
transparent proxy	6 items
netmon proxy	6 items
ifwa internal firewall	10 items
dwar VPN	10 items

For BGP alone, there are 26 different items to monitor and that also when we talk about traffic monitoring. IF we take transparent proxy, there are 6 transparent proxy and each transparent proxy has 6 items. So in this way Total number of entries in database per hour will be =26+9*6+6*6+10+10 = 136 entries
So there will 136 new entry after every one hour in this table. but we also have another table which is storing data at minute level i.e after every one minute there will 136 new entries in that table. Now It is very important to Analyzing the database size. How much space database is going to take. Once we have an idea how much space particular table is

going to take at particular granularity level only then we can take a decision of granularity level of storing statistics.

4.4.2 Memory requirement Analysis:

[3]

Note : Calculations and experiment done are on mysql database and innodb engine

Queries to calculate space :-

- Show table status like ‘table name’
- SELECT
  table_name AS ’Table‘,
  round(((data_length + index_length) / 1024), 2)
  ‘KB‘ , round(((data_length + index_length)), 2)
  ‘B‘
  FROM information_schema.TABLES
  WHERE table_schema = ”mem_analysis”
  AND table_name = ”traffic”;

Expected space by database :

Space taken by table = (number of of rows) * ((space_per_row)+ 35)

- space_per_row in above equation can be calculated by looking at datatype of each column.
- For a table with 10 columns each of integer type space_per_row will be 4 * 10 bytes.
- 35 is for extra overhead per row which was described above.
  Extra overhead per row :-
  1. 5 byte row header
  2. 6 byte transaction ID
  3. 7 byte roll pointer
  4. A variable-length bit flag vector of size CEIL(nullable_column _count/8) which indicates whether each nullable column for that row is null .
  5. DB also tends to leave gaps at the end of each data page, reducing the need for page splits and for possible grow of variable length data type.
  6. Size of the gap varies by whether rows are inserted in primary key me order, or not.
  7. With an AUTO_INCREMENT primary key, pages should fill to somewhere around 15/16ths.

Note : Above formula will be an an upper limit on space as DB engine also does some kind of compression and optimization as size of database grows but this compression will be nullified by the fact that DB also maintains some free space per table . As number of rows will increase we can decrease value of extra overhead to 25.

Now, Let x be the space consumed when trends are saved on per hour basis

x = Number of of rows*((space_per_row)+35)
where
no. of rows = (Number of entries per hour)*24 *30 *12;
Let N1 is number of years for which we are maintaining trends on per hour basis:
then.
Total space = N1 * x;
For same entity if we want to store trends on weekly basis for N2 years
y = No.of rows *((space_per_row)+35)
where
Number of rows = (Number of entries per hour) * 13;
Total space = N1 *x +N2 * y

Experiment & Verification:- Following table tells my expected calculated space vs actual space taken

There are two tables

Table 1 : table with 10 int type column where each int type takes 4 bytes

Table 2 : table with 1 int type column

Number of entries per hour = 136 Number of entries per month = 136 * 720 = 97920

Month	Row Count	Table 1		Table 2	
		Expected space	Actual space	Expected space	Actual space
1	97920	7344 KB	6672 KB	3818 KB	3600 KB
2	195840	14688 KB	13840 KB	7637 KB	6672 KB
3	293760	22032 KB	20016 kB	11456 KB	8720 KB
4	391680	29376 KB	27200 KB	15272 KB	11792 KB
5	489600	36720 KB	34368 KB	19094 KB	14864 KB
6	587520	44064 KB	41552 KB	22913 KB	17968 KB
7	685440	51408 KB	46672 KB	26732 KB	22064 KB
8	783360	58752 KB	51792 KB	30551 KB	24112 KB
12	1175040	90321 KB	83024 KB	41234 KB	36720 KB

- In 7th month average length per row was 72 for table1 so extra overhead was 32 only.
- In 8th month average length per row was 67 for table1 so extra overhead was 27 only.
- For table 2 5 month onward extra overhead is either 25 or 26 .

Here we can see as size of table increases, extra-overhead will decrease down to 25.

Chapter 5

Algorithm & Implementation

5.1 Raw Data Puller Algorithm

This algorithm is responsible for maintaining raw data. It pulls data from zabbix server and put it in LTTA raw_data table. After every one minute this algorithm execute itself and get data from zabbix for that minute. Since data is pulled every time both databases zabbix and LTTA remains in sync but sometimes due to disconnection or other network problem both databases go out of sync, in that case it is responsibility of this algorithm to bring both back in sync and then again start pulling data minute after minute.

5.1.1 Algorithm Input

hosts database: This database is described earlier in previous chapter. It contains two table host and items. Before start execution all itemid from items table are loaded in a list and then that list is used to fetch data from zabbix server. To maintain this database a front-end is provided at front-end under admin section. New items or hosts can be added,deleted or modified from this section of front-end.
Clock: This variable tells next unix time for which traffic value will be pulled from zabbix. For example if local LTTA server has traffic value upto 11:30 than clock variable will contain unix time corresponging to 11:31.

5.1.2 Algorithm

Algorithm 1 Raw Data Puller Algorithm

```
Input: mysql database 'hosts'
Output: traffic value for Clock is inserted in raw_data table

items = read unique(itemid) from table items in hosts database
for every 1 minutes do
    call DATA_PULLER(items)
end for

clock, Global variable {CLock = Time value for which traffic data would be pulled next}
procedure DATA_PULLER(list items)
    long localClock = getlocalMaxClock();
    long zabbixClock = getZabbixMaxClock();
    if $|localClock - zabbixClock| > 60$ then
        bringSync(localClock,zabbixClock)
        return
    end if
    for for each itemid in items do
        long value = getValue(itemid,clock)
        insert(itemid,clock,value)
    end for
    clock = clock + 60
end procedure
procedure BRINGSYNC(long zabbix,long local)
    while clock < zabbix do
        for every itemid in items do
            long value = getValue(itemid,clock)
            insert(itemid,clock,value)
        end for
        clock =clock+ 60;
    end while
    end procedure =0
```

5.2 Data Archival Algorithm

This algorithm is responsible for archival of data at multiple levels. This algorithm extracts statistics from raw_data and store it at first level of archival and then from this level other archival level are maintained. As an input it takes **archival_info** file as an input. Let's discuss this file first before going into further details.

5.2.1 Algorithm Input

archival_info File

As said earlier this file serves as an input for Data archival algorithm. This file basically contains information for all archival tables. It contains name of table, granularity at which statistics are stored and number of days for which statistics at preserved at this granularity. Following snippet gives a glimpse of this file

Granularity	stats Preservation Time Period	Table Name
1 Minute	4 Hour	raw_data
60 Minutes	72 Hour	hourly_archival
1440 Minutes	360 Hour	daily_archival
10080 Minutes	360 Hours	weekly_archival

Table 5.1: Working of IITBLTTA

In this file each line describe an archival level. First level table whose name is raw_data stores data at 1 minute level for last 4 hours. Similarly second level table name is hourly_archival and it stores archival at 60 minutes i.e 1 hour basis for 72 hours i.e 3 days.

To maintain this file front end is provided under admin section. Admin can see all these level at front-end. Admin can add , delete or modify a level. Although in file, 1st column and 2nd column always remain in minute and hour but at front-end admin can use any unit to define these level. While writing into file data will be converted to minutes and hours.

hosts database: This input is same as described in Data Puller alogorithm. Before execution all itemid are loaded in items list. This load process mainly takes place in construcotr part.

5.2.2 Algorithm

Algorithm 2 Data Archival Algorithm

```
Input: archival_info File
Output:
$fileMatrix$ = (matrix)archival_info
$g \leftarrow$ fileMatrix[1][0] {i.e Granularity of first level}
while True do
   for every g minutes do
      call stats_generator(2)
   end for
end while

procedure STATS_GENERATOR(level) {level: archival-info line number}
   if $read(level) == null$ then
      return
   end if
   G(level) = granularity at this level
   H(level) = hoursPreservedAtThisLevel
   T (level) = DB table at this level
   T' = T(level-1)
   D' = D (level-1);
   G' = G(level-1);
   for itemid in items do
      rc' = Row count(T')
      if $rc' > (H' * 60/G') + G/G'$ then
         while rc' >= (H' * 60 / G' ) + G / G' do
            Collect oldest G/G' entries
            Aggregate G/G' entries
            Store stats in T
            delete oldest G/G' entries from T'
            rc' = rc'-G/G'
         end while
         end if
      end for
      call stats_generator(level+1)
   end procedure =0
```

Chapter 6

Implementation and Result

In this chapter, we will see implementation detail of tool. We will see technology used for implementation for both front - end and back-end. First section of this chapter would be dedicated to implementation only. In the second section, we will see user interface and how it can be used for posting different kind of queries. Next to this, we will discuss admin section. In next section, we will see all these sections and features in action through screen-shots.

6.1 Implementation Detail

As said earlier IITBLTTA is divided into two parts . One is LTTA-PE i.e Periodic Event which is working at back-end and other is LTTA-FE which is front-end. LTTA-PE is fully developed using **java7** language. Implementation of cron-job is done via **TimerTask class** of Java.

For LTTA-FE, Technology being used is **JSP & Servlets**. I am using **Model View Controller (MVC)** framework for implementation where as View classes **(Mainly JSP pages along with Java script, Jquery and Bootstrap)** are responsible for getting user request and at the end providing response to user at front-end, While model classes **(Mainly Java classes)** are responsible for database interaction. It makes connection with database server, executes a query and get results. On other hand, Controllers (Java servlets) are responsible for controlling View and Model part. In short, JSP will get input data from user and call controller, controller will call appropriate modal class according to user request. Modal class will fetch data out of database by forming a right mysql query and handover result to controller and finally Controller will call another or same view class to show results at front-end.

Within JSP & Servlet I have used **struts framework** for implementation and also used **Java Beans** feature for JSP form implementation.

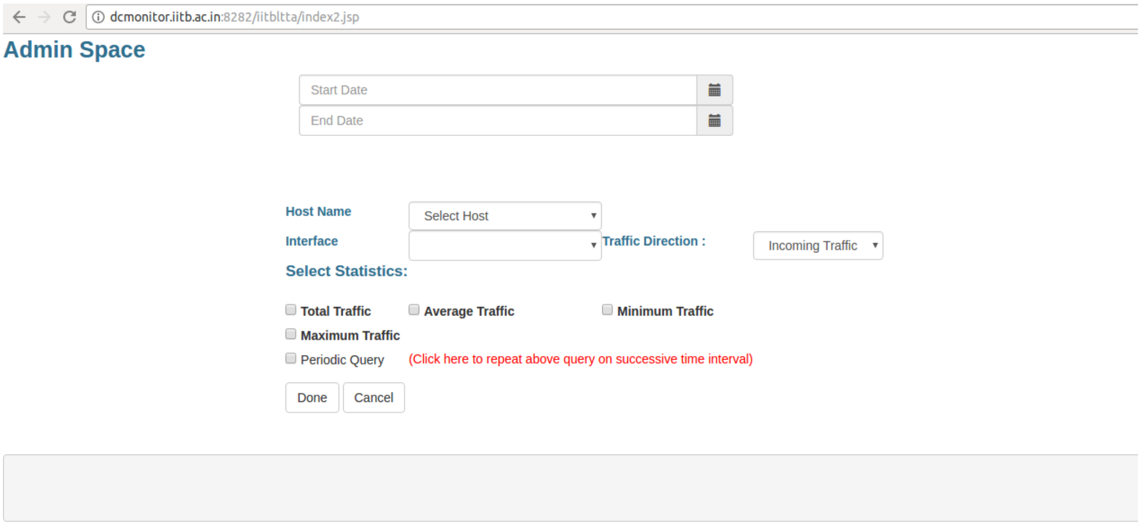
IITBLTTA is currently running at one of CC server and can be accessed via link **http://dcmonitor.iitb.ac.in:8282/iitbltta/index2.jsp** . Specification for whole tool and server is as follow.

Specification:

- **Technology Used** :- JSP and Servlet, Java7, Struts
- **Frontend** :- JSP, Javascript, Jquery , Bootstrap, Struts, Java-Beans
- **Databse** :- mysql server, mysql Version 14.14 Distrib 5.7.15, Innodb engine
- **Machine Info** :- Linux (x86 64) with i5 processor

6.2 Front-End Implementaion

6.2.1 User-Interface



dcmonitor.iitb.ac.in:8282/iitbltta/index2.jsp

Admin Space

Start Date

End Date

Host Name: Select Host

Interface:

Traffic Direction : Incoming Traffic

Select Statistics:

□ Total Traffic □ Average Traffic □ Minimum Traffic

□ Maximum Traffic

□ Periodic Query (Click here to repeat above query on successive time interval)

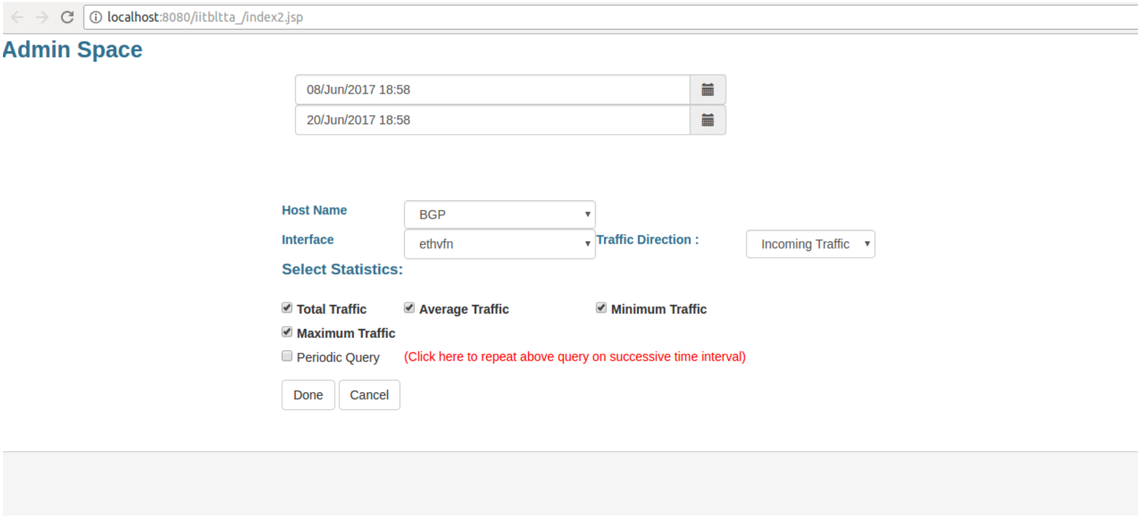
Figure 6.1: Screen-Shot of Front-end

6.2.2 Framing a query in continuous Time Period.

These kind of queries display selected statistics between selected time-period for selected host and other parameters.

- **Posting a query:**

  With help of form shown user can provide his/her query.



localhost:8080/iitbltta_/index2.jsp

Admin Space

08/Jun/2017 18:58

20/Jun/2017 18:58

Host Name: BGP

Interface: ethvfn

Traffic Direction : Incoming Traffic

Select Statistics:

☑ Total Traffic ☑ Average Traffic ☑ Minimum Traffic

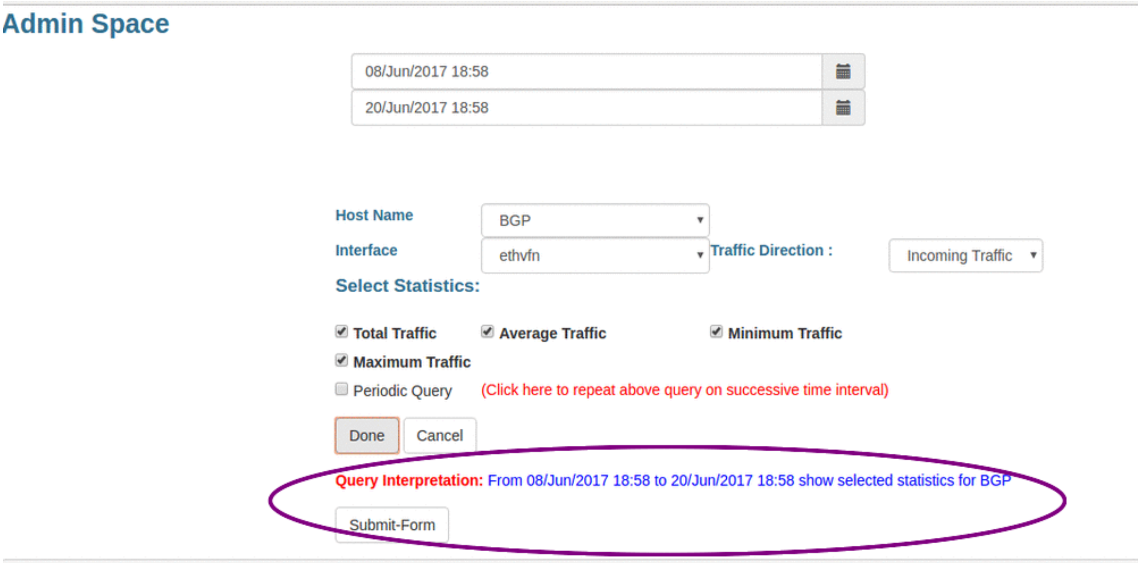
☑ Maximum Traffic

□ Periodic Query (Click here to repeat above query on successive time interval)

Figure 6.2: User posting a query

- **Query Interpretation:**

  On basis of user Input, tool itself tries to interpret what kind of statistics it is going to give to user. It is displayed when user is done with input selection and click on done button. Once done is clicked,before submit this interpretation is shown.



Admin Space

08/Jun/2017 18:58

20/Jun/2017 18:58

Host Name: BGP

Interface: ethvfn

Traffic Direction : Incoming Traffic

Select Statistics:

☑ Total Traffic ☑ Average Traffic ☑ Minimum Traffic

☑ Maximum Traffic

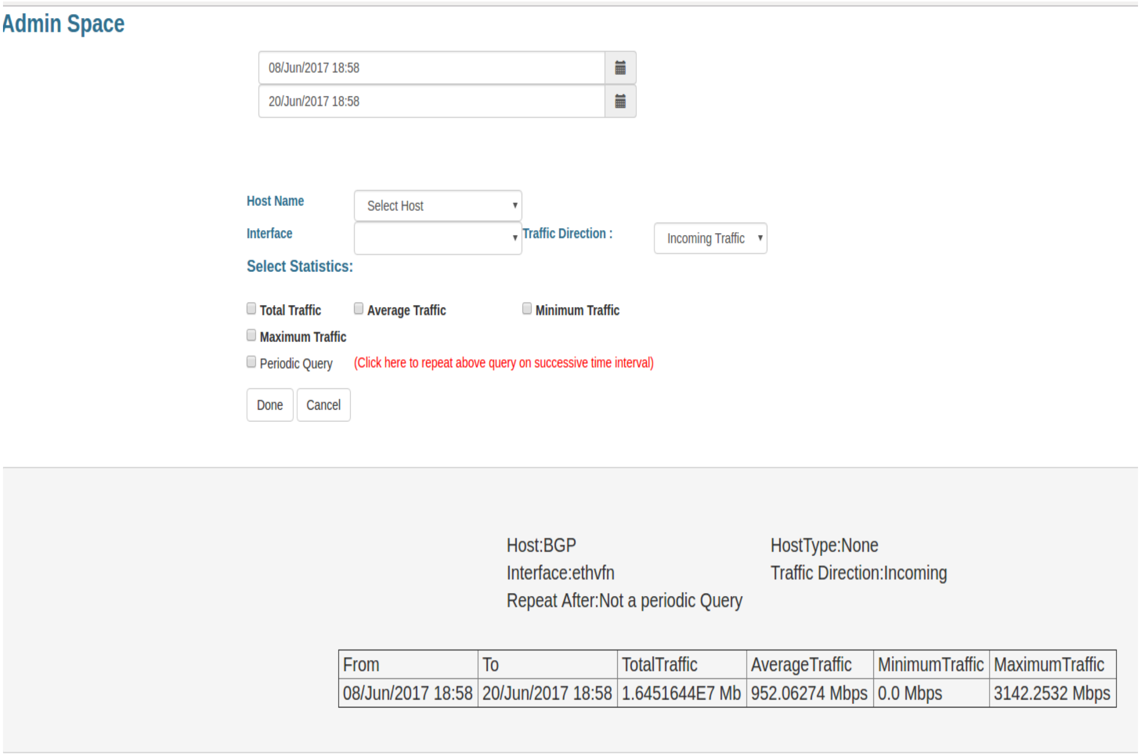
□ Periodic Query (Click here to repeat above query on successive time interval)

Query Interpretation: From 08/Jun/2017 18:58 to 20/Jun/2017 18:58 show selected statistics for BGP

Figure 6.3: Query Interpretation

- **Result Display**

  Result is displayed in a table as show in below figure



08/Jun/2017 18:58

20/Jun/2017 18:58

Host Name: Select Host

Interface:

Traffic Direction : Incoming Traffic

Select Statistics:

□ Total Traffic □ Average Traffic □ Minimum Traffic

□ Maximum Traffic

□ Periodic Query (Click here to repeat above query on successive time interval)

Host:BGP HostType:None

Interface:ethvfn Traffic Direction:Incoming

Repeat After:Not a periodic Query

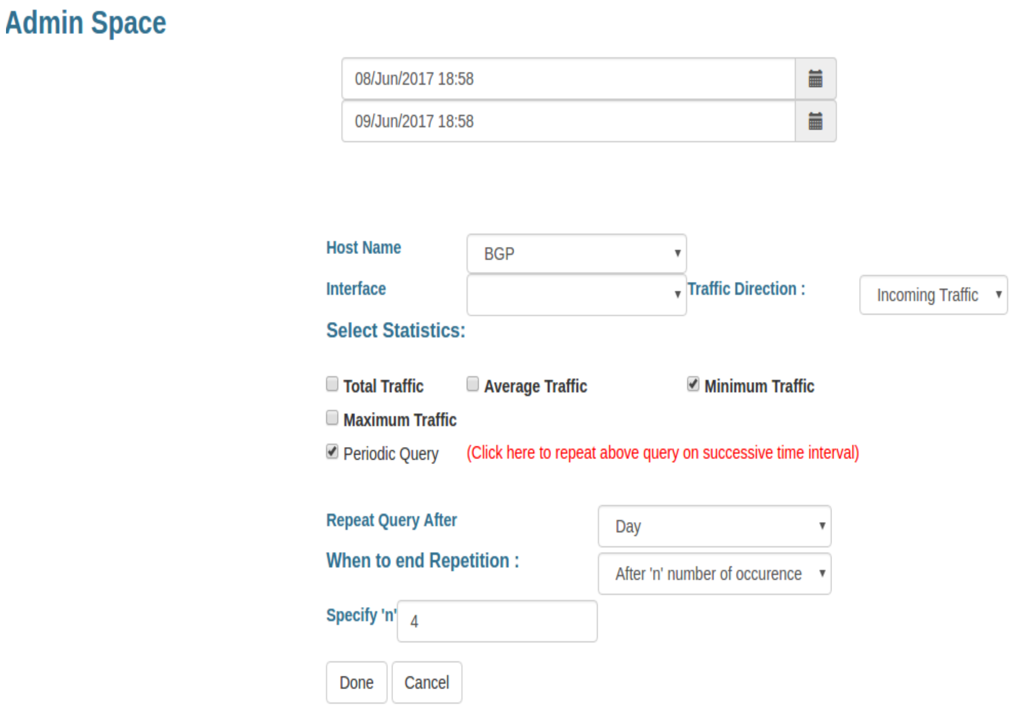
From	To	TotalTraffic	AverageTraffic	MinimumTraffic	MaximumTraffic
08/Jun/2017 18:58	20/Jun/2017 18:58	1.6451644E7 Mb	952.06274 Mbps	0.0 Mbps	3142.2532 Mbps

Figure 6.4: Result Display

6.2.3 Framing a query in Periodic Time Period

- **Posting a query**

  To execute same query in periodic mode, user has to click on Check-box with label Periodic Query. After clicking this option further option will be displayed for framing query.



Admin Space

08/Jun/2017 18:58

09/Jun/2017 18:58

Host Name: BGP

Interface:

Traffic Direction : Incoming Traffic

Select Statistics:

☐ Total Traffic ☐ Average Traffic ☑ Minimum Traffic

☐ Maximum Traffic

☑ Periodic Query (Click here to repeat above query on successive time interval)

Repeat Query After: Day

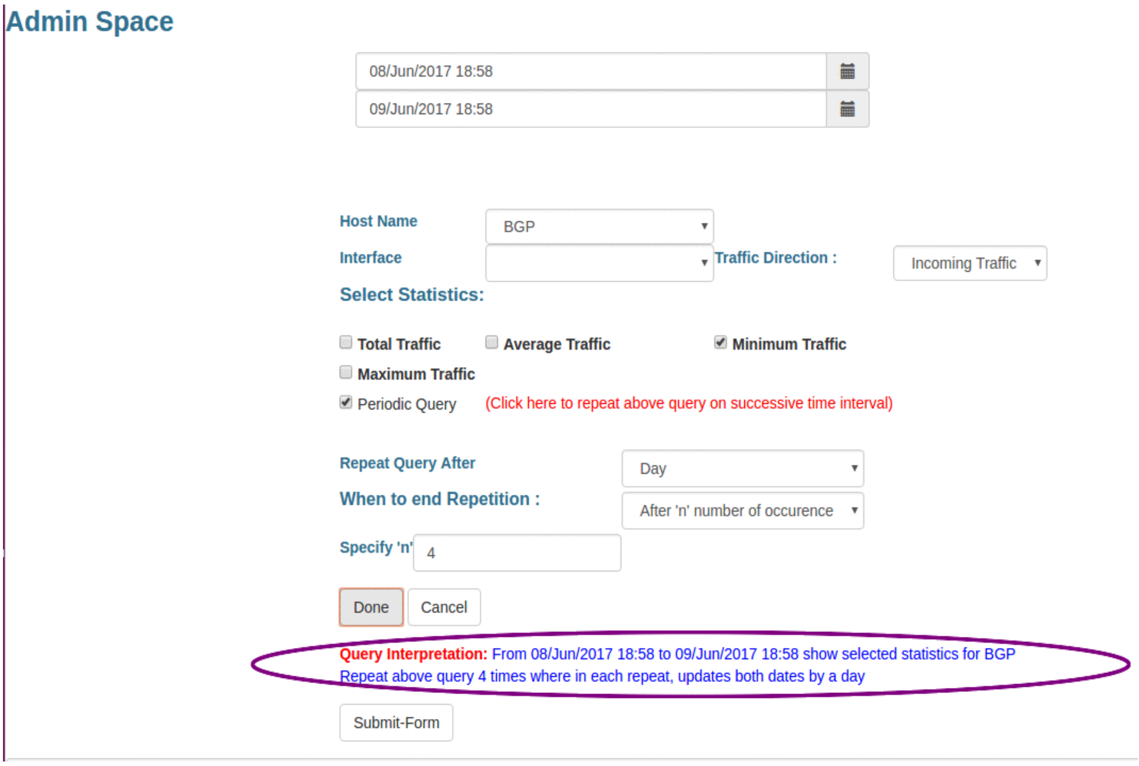
When to end Repetition : After 'n' number of occurence

Specify 'n': 4

Figure 6.5: Frame a query in periodic form

- **Query Interpretation**

  same as previous, Query Interpretation would be shown.



Admin Space

08/Jun/2017 18:58

09/Jun/2017 18:58

Host Name: BGP

Interface:

Traffic Direction : Incoming Traffic

Select Statistics:

□ Total Traffic □ Average Traffic ☑ Minimum Traffic

□ Maximum Traffic

☑ Periodic Query (Click here to repeat above query on successive time interval)

Repeat Query After: Day

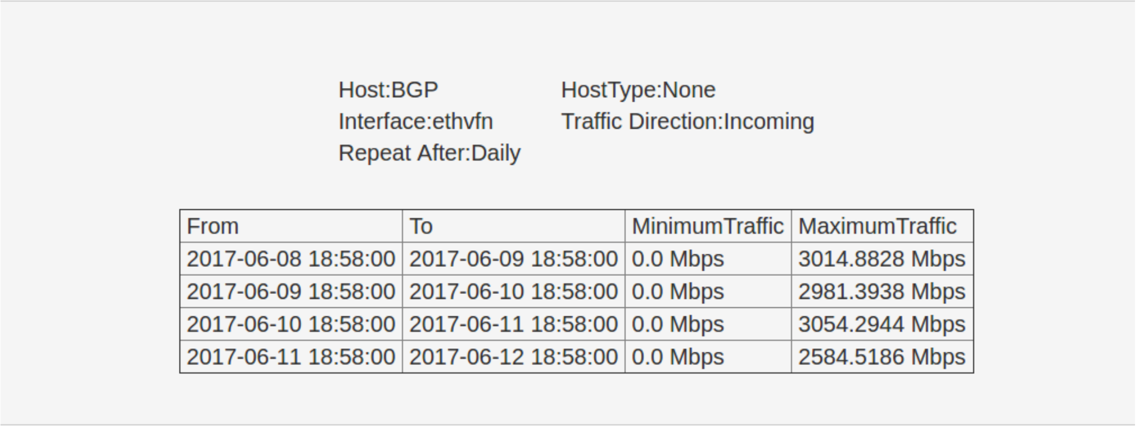
When to end Repetition : After 'n' number of occurence

Specify 'n': 4

Query Interpretation: From 08/Jun/2017 18:58 to 09/Jun/2017 18:58 show selected statistics for BGP
Repeat above query 4 times where in each repeat, updates both dates by a day

Figure 6.6: Query Interpretation for Periodic Query

- **Result**



Host:BGP HostType:None
Interface:ethvfn Traffic Direction:Incoming
Repeat After:Daily

From	To	MinimumTraffic	MaximumTraffic
2017-06-08 18:58:00	2017-06-09 18:58:00	0.0 Mbps	3014.8828 Mbps
2017-06-09 18:58:00	2017-06-10 18:58:00	0.0 Mbps	2981.3938 Mbps
2017-06-10 18:58:00	2017-06-11 18:58:00	0.0 Mbps	3054.2944 Mbps
2017-06-11 18:58:00	2017-06-12 18:58:00	0.0 Mbps	2584.5186 Mbps

Figure 6.7: Query Interpretation for Periodic Query

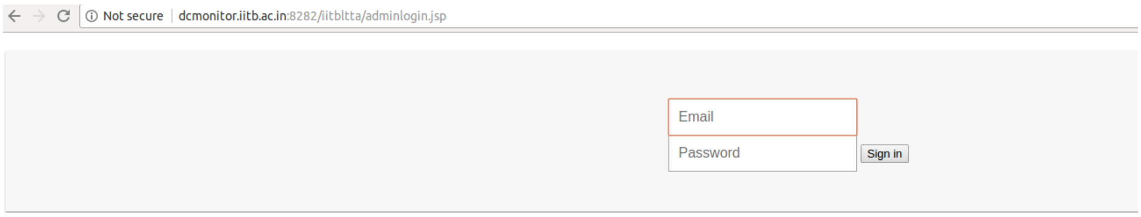
6.3 Admin Interface

Tool front-end has a separate section for admin tasks. This section is privileged and admin can login through a given username or email and password. Admin section can be accessed by click on **Admin Section** link on home page.

6.3.1 Login Page

- **Login Page**

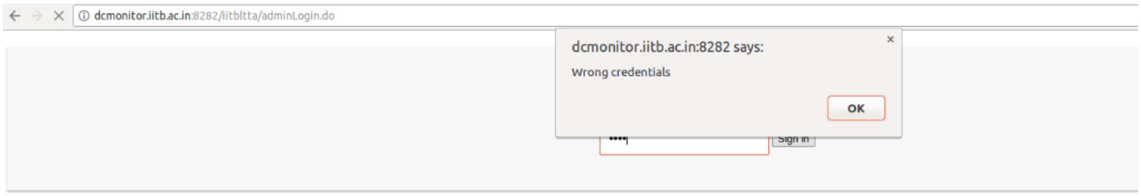
  Following page appears after clicking on AdminSpace link on home page



Not secure | dcmonitor.iitb.ac.in:8282/iitbltta/adminlogin.jsp

Figure 6.8: Login Page

- **Unsuccessful Login**



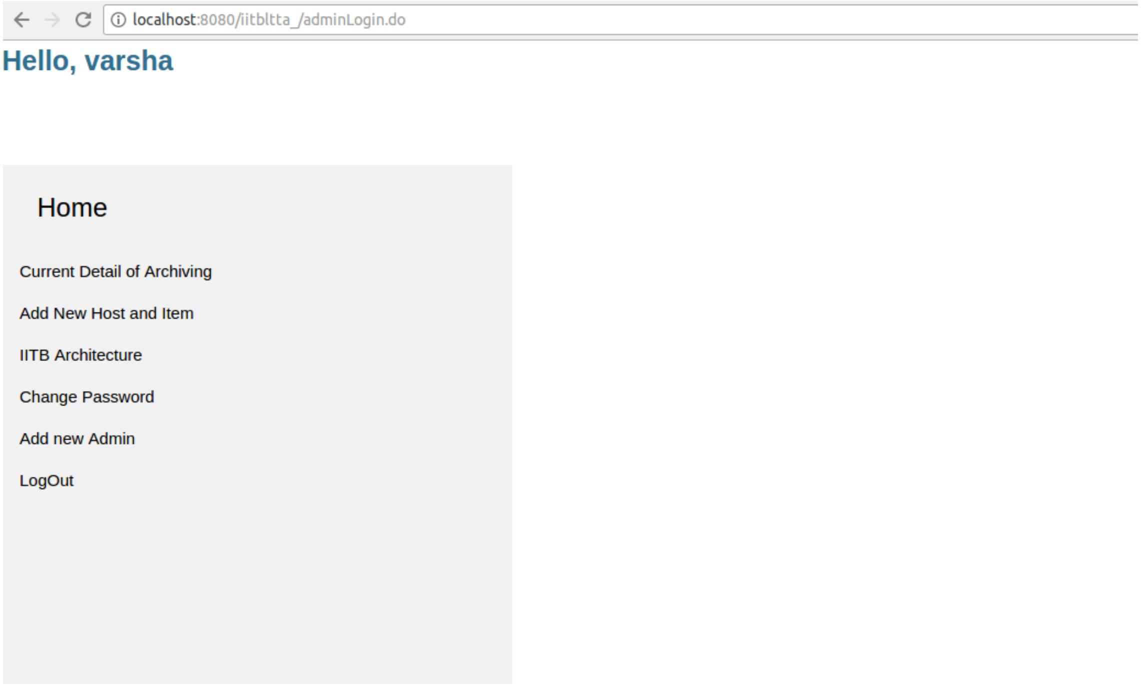
dcmonitor.iitb.ac.in:8282/iitbltta/adminLogin.do

dcmonitor.iitb.ac.in:8282 says:

Wrong credentials

Figure 6.9: Unsuccessful Login

- **Successful Login**



localhost:8080/iitbltta_/adminLogin.do

Hello, varsha

Home

Current Detail of Archiving

Add New Host and Item

IITB Architecture

Change Password

Add new Admin

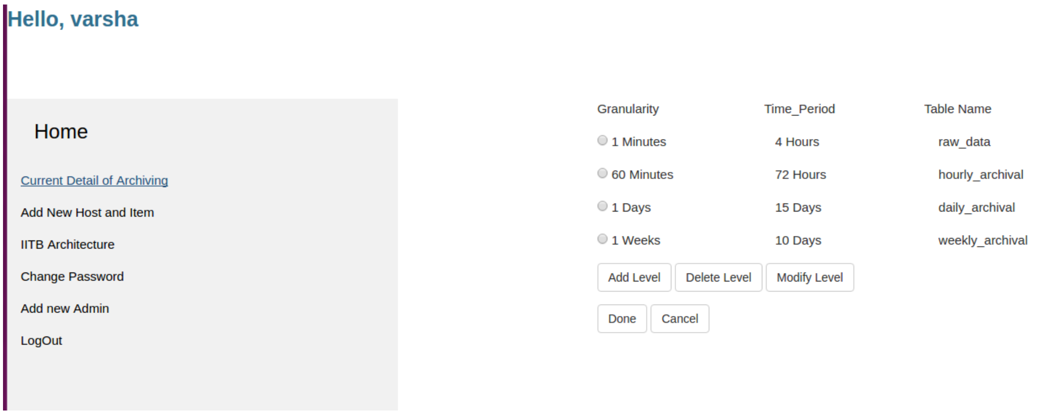
LogOut

Figure 6.10: Successful Page

6.3.2 Going deep in "Current Detail of Archiving" option

- **Seeing all current level of archivaing**

  Simply click on "Current Detail of Archiving" shows all level of archiving , its granularity and time period for which data is preserved at this granularity is also shown .This option also give you other option like adding new level, modifying an existing level and deleting a level. We will see all option one by one.



Hello, varsha

Home

Current Detail of Archiving

Add New Host and Item

IITB Architecture

Change Password

Add new Admin

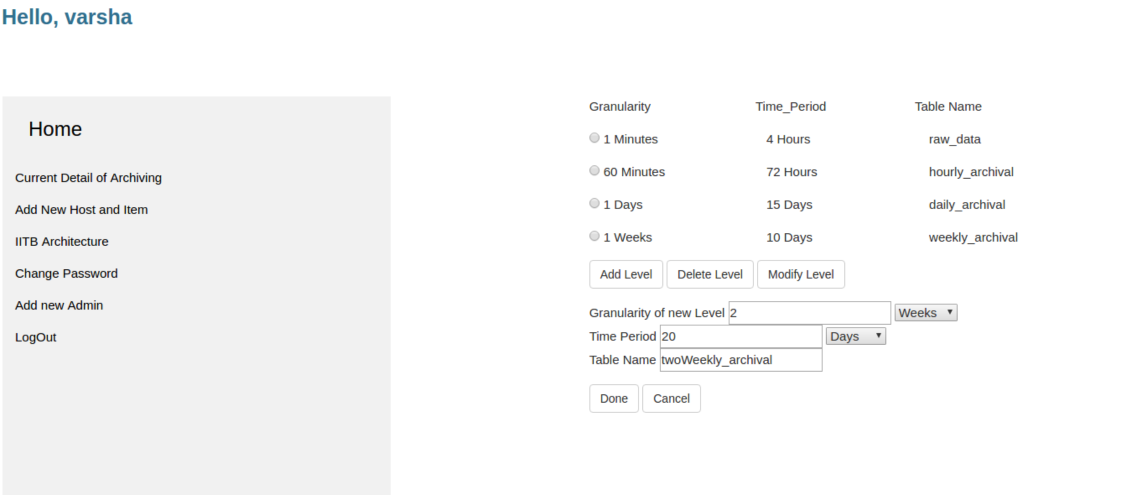
LogOut

Granularity	Time_Period	Table Name
○ 1 Minutes	4 Hours	raw_data
○ 60 Minutes	72 Hours	hourly_archival
○ 1 Days	15 Days	daily_archival
○ 1 Weeks	10 Days	weekly_archival

Figure 6.11: Checking current details of all archiving tables

- **Adding new Level of Archival**

Under this section admin can add new level of archival. Admin can tell granularity at which he/she wants to store trends, number of days for which data would be preserved for this granularity and name of table.



Hello, varsha

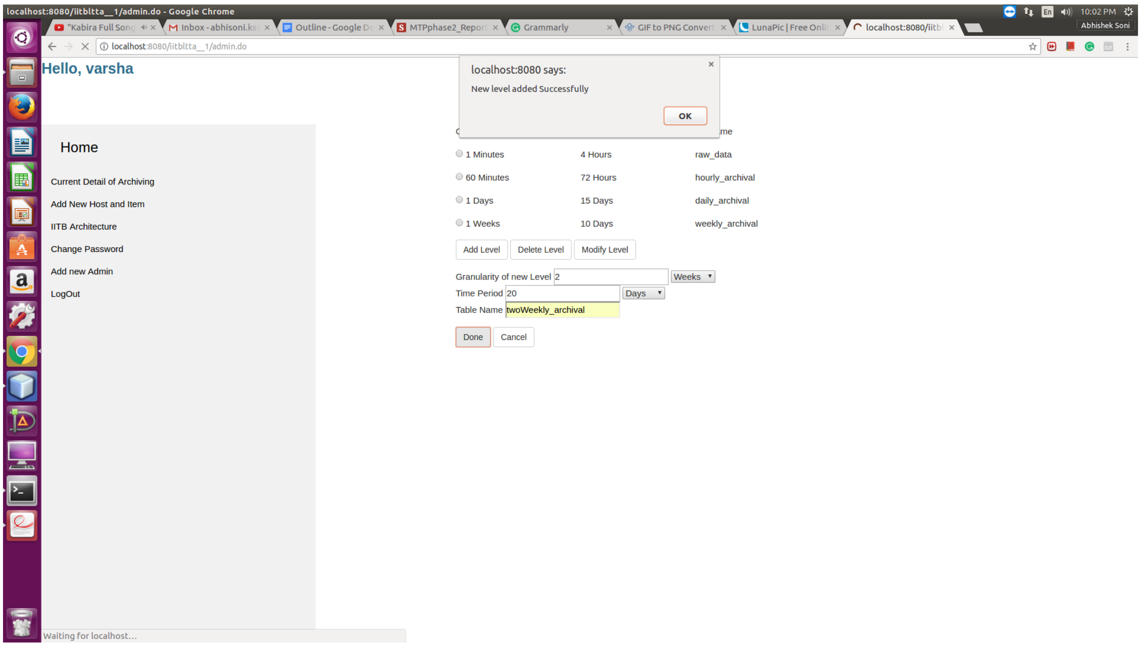
Granularity	Time_Period	Table Name
○ 1 Minutes	4 Hours	raw_data
○ 60 Minutes	72 Hours	hourly_archival
○ 1 Days	15 Days	daily_archival
○ 1 Weeks	10 Days	weekly_archival

Granularity of new Level 2 Weeks

Time Period 20 Days

Table Name twoWeekly_archival

Figure 6.12: Adding new Level Details



localhost:8080/iitbltta__1/admin.do - Google Chrome

localhost:8080/iitbltta__1/admin.do

Hello, varsha

localhost:8080 says:

New level added Successfully

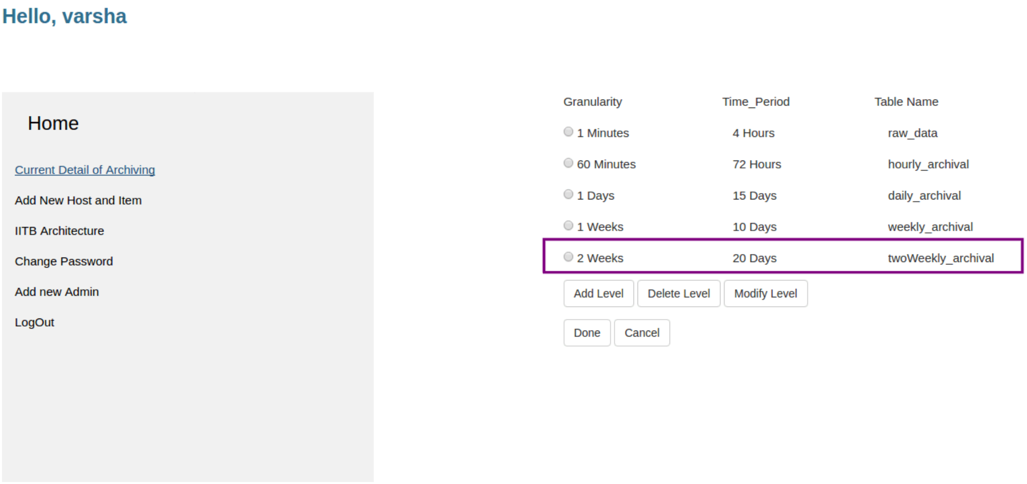
○ 1 Minutes	4 Hours	raw_data
○ 60 Minutes	72 Hours	hourly_archival
○ 1 Days	15 Days	daily_archival
○ 1 Weeks	10 Days	weekly_archival

Granularity of new Level 2 Weeks

Time Period 20 Days

Table Name twoWeekly_archival

Figure 6.13: Level Added Sucessfully

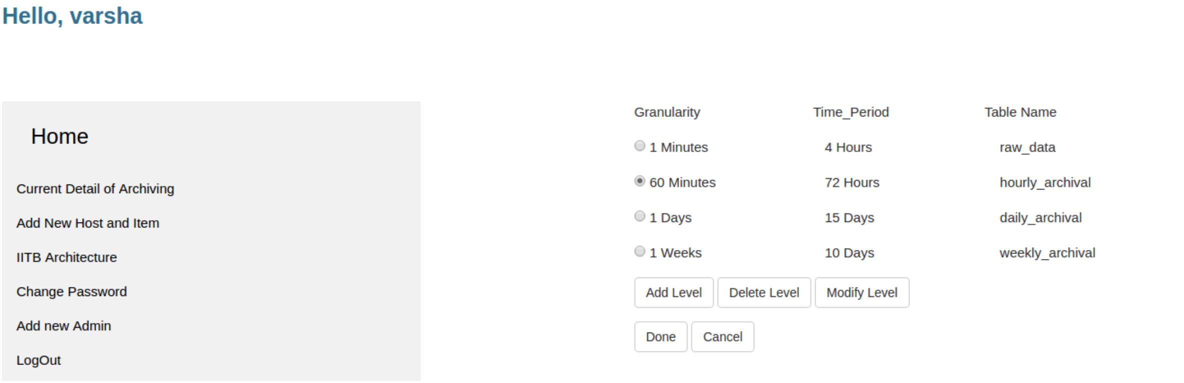


Granularity	Time_Period	Table Name
○ 1 Minutes	4 Hours	raw_data
○ 60 Minutes	72 Hours	hourly_archival
○ 1 Days	15 Days	daily_archival
○ 1 Weeks	10 Days	weekly_archival
○ 2 Weeks	20 Days	twoWeekly_archival

Figure 6.14: New Archival details displayed

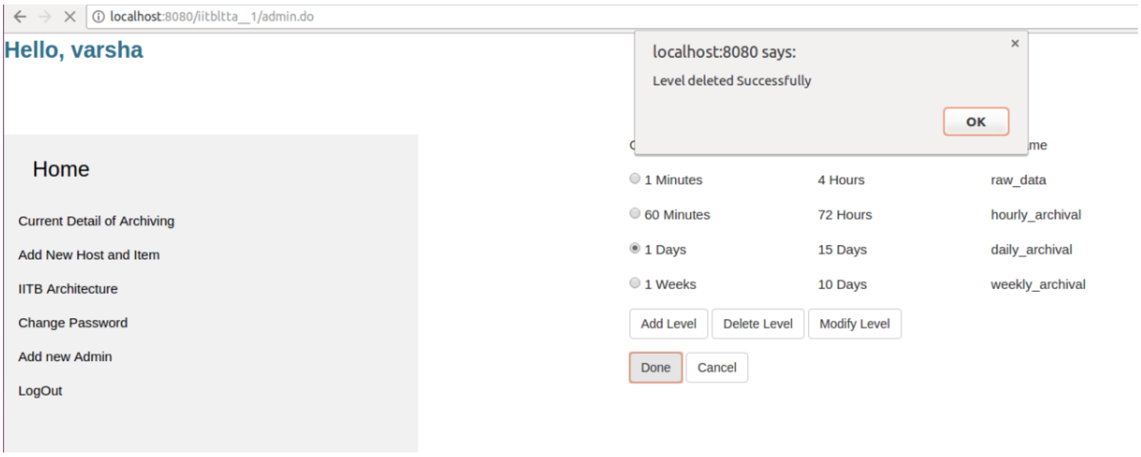
- **Deletion of a Level**

  If admin wants to delete a level he or she can do it. In this case all data at this level would be aggregated according to granularity of next level and put in next table. For example, in current scenario if we delete daily archival table then all data in this table would be aggregated at weekly level and then be put in next table i.e weekly_archival table.



Granularity	Time_Period	Table Name
○ 1 Minutes	4 Hours	raw_data
● 60 Minutes	72 Hours	hourly_archival
○ 1 Days	15 Days	daily_archival
○ 1 Weeks	10 Days	weekly_archival

Figure 6.15: Select a level to delete

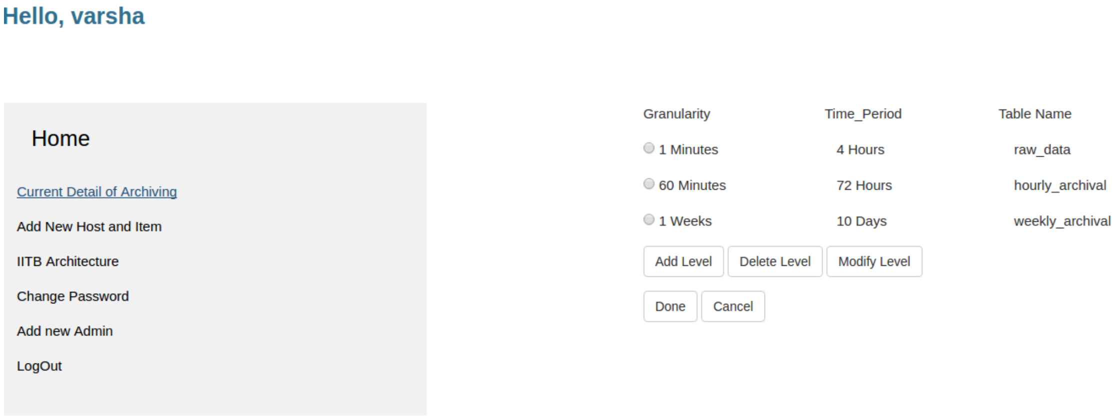


localhost:8080 says:

Level deleted Successfully

		me
○ 1 Minutes	4 Hours	raw_data
○ 60 Minutes	72 Hours	hourly_archival
● 1 Days	15 Days	daily_archival
○ 1 Weeks	10 Days	weekly_archival

Figure 6.16: Level Deleted Successfully



Granularity	Time_Period	Table Name
○ 1 Minutes	4 Hours	raw_data
○ 60 Minutes	72 Hours	hourly_archival
○ 1 Weeks	10 Days	weekly_archival

Figure 6.17: New Archival details displayed

6.3.3 Other Functionalities for Admin

- **IITB Architecture**

  Under this section admin can see all flow diagram shown above in chapter 2.

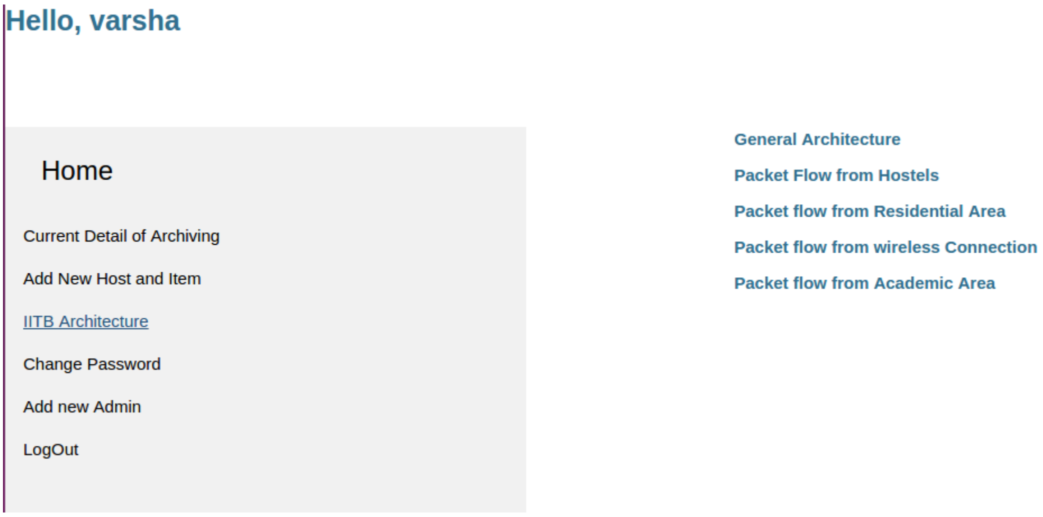
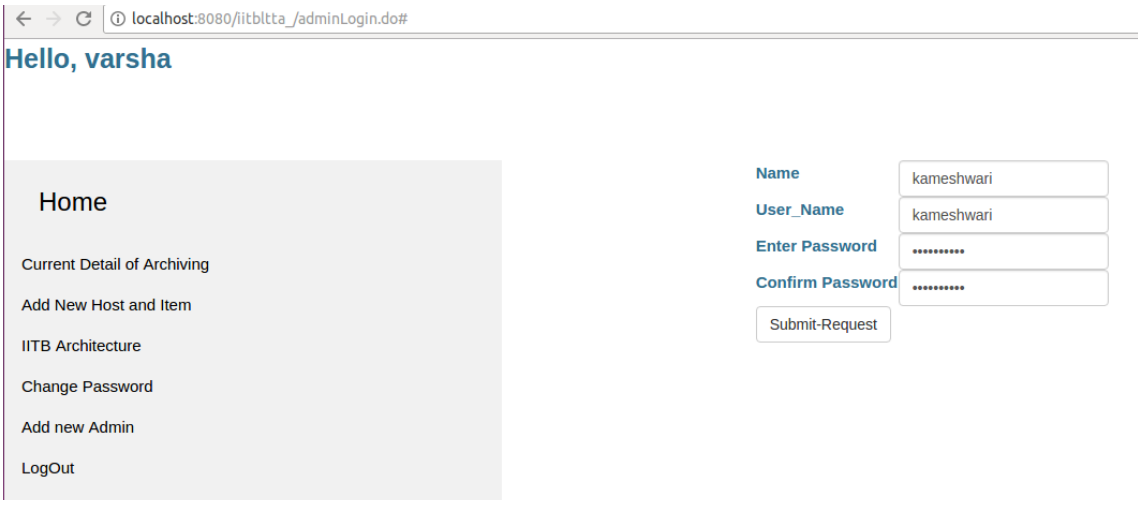


Figure 6.18: IITB Architecture

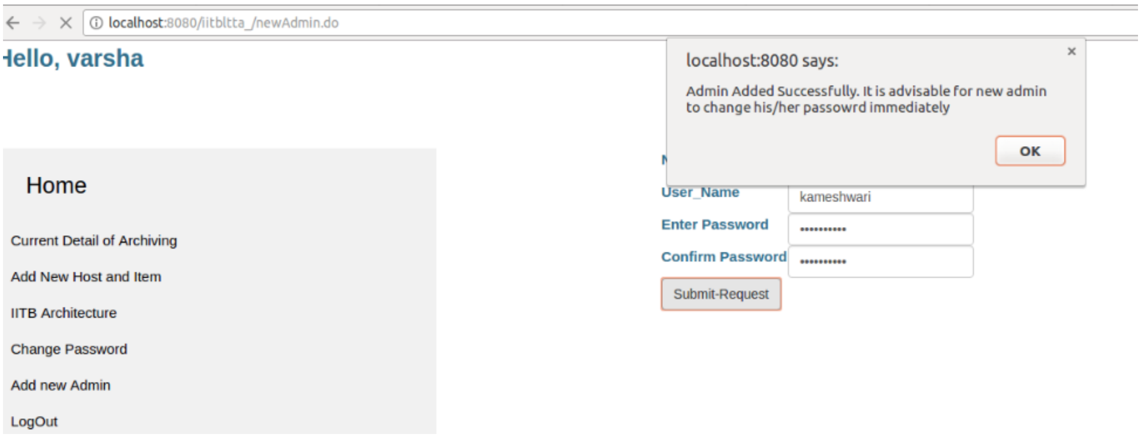
- **Add new Admin**

  Under this section admin can add a new admin. With help of username, an email id and password new admin would be created. First time password would be provided by admin. Later, new admin can change his/her password.



Name: kameshwari
User_Name: kameshwari
Enter Password: ••••••••••
Confirm Password: ••••••••••

Figure 6.19: Add new Admin



localhost:8080/iitbltta_/newAdmin.do

Hello, varsha

localhost:8080 says:

Admin Added Successfully. It is advisable for new admin to change his/her passowrd immediately

User_Name: kameshwari

Enter Password: ••••••••••

Confirm Password: ••••••••••

Figure 6.20: Admin added successfully

Chapter 7

Conclusion and Future Work

IITB has a very huge traffic network and many tools are used for managing such a huge network, One such tool is Zabbix which is very good in monitoring and alerting tasks but not efficient in maintaining long term trends. We see a number of questions which can not be answered by zabbix but are very important to get answered. We see why IIT need a tool which can store traffic data over a long period of time. This tool should not only be able to store traffic data but as data become old, granularity of data should also increase. We also justify need of such tool and then we propose IITBLTTA - Long Term Archival tool. This tool can be use to remove raw data and store useful long term trends out of raw data. We proposed it's architecture, design and working of tool. We also designed algorithm for it's back-end and front-end. We fully implemented back-end of tool where raw data is being pulled continuously and data is being archived at multiple granularity. We also implemented support for front-end where queries can be put and trends can be seen at front-end.

Future Work

1. Adding support for trend Comparison

   As described in Chapter 6, as of now LTTA only supports two kinds of queries. one is, query in a continuous time period and other is query over a repeated time period. For future, it would be very useful to add support for traffic comparison among two devices. Even on a single devices traffic comparison among different interfaces. These kind of statistics are very helpful for checking load balance or finding other network abnormalities. By adding this support, following kind of traffic queries would be answered.

   - Compare total traffic for Tp0 to tp5 for today

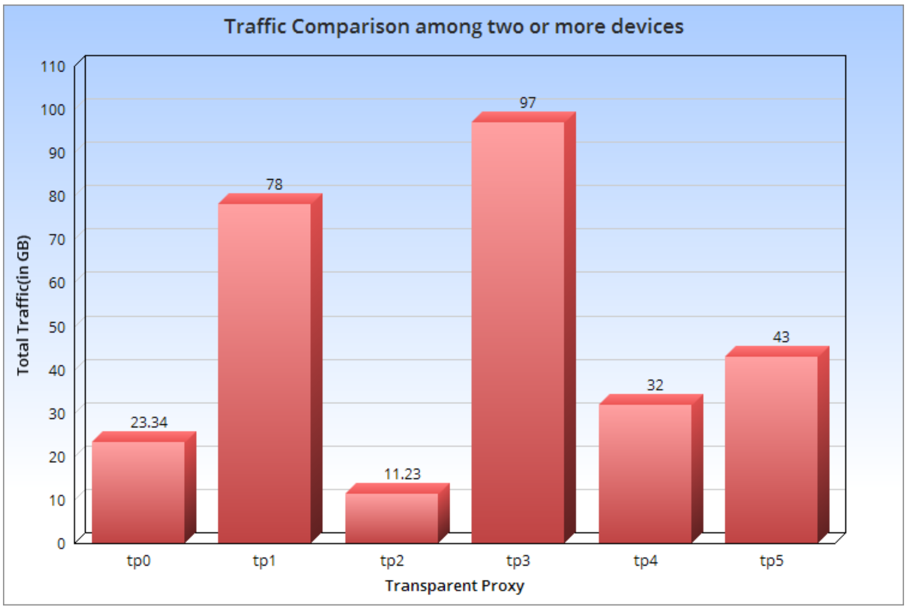


Figure 7.1: Traffic Comparison among Transparent Proxies

- Tell Maximum traffic for this week per day for all interface of BGP

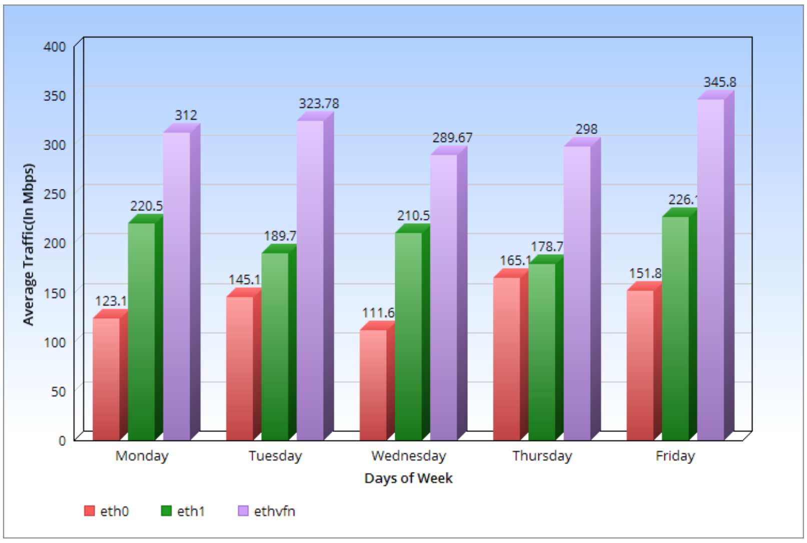


Figure 7.2: Comparison of Traffic among interfaces of BGP

2. Implementation of Machine Learning Algorithm

   As of now tool is only storing statistics and later on demand is showing these stored statistics after little calculation. There is no learning or any other kind of intelligence. Once tool collects huge amount of data , machine learning algorithm and techniques like regression or reinforcement learning can be applied to predict future network traffic and behaviour. Machine learning techniques can be use to find beautiful pattern in stored data and on basis of this pattern a lot of information can be extracted.

3. Adding support for trends other traffic intensity

   Besides Zabbix, IIT has other monitoring tool. One such tool is mns tool which is monitoring switches of IITB. During monsoon, switch failure is common problem. By saving trends for switches it would become very helpful to predict and diagnose network problems in rainy season.

Bibliography

[1] “https://www.zabbix.com/documentation/2.2/manual”.

[2] “https://www.caida.org/tools/taxonomy/anontaxonomy.xml”.

[3] ”http://dev.mysql.com/doc/”.

[4] ”http://searchnetworking.techtarget.com/tip/analyzing-your-network-statistical-monitoring-vs-real-time-performance”.

[5] ”https://www.zabbix.com/forum/showthread.php?t=5524”.

[6] ”https://www.zabbix.org/mw/images/a/ad/zabbixdbschema-2.4.3-mysql.pdf”.

[7] ”http://www.unixgeeks.org/security/newbie/unix/cron-1.html”.

[8] ”https://msdn.microsoft.com/en-in/library/dn282389.aspx”.