Alleviating ISP Last Mile Network Performance Using Adaptive Nutrition Label Technique

By

Nooma Roohi	**2009-NUST-BE-BICSE-200**
Sajjal Raza Rao	**2009-NUST-BE-BICSE-205**

Final Year Project Report

Bachelors in Information and Communication System Engineering

Department of Electrical Engineering

School of Electrical Engineering & Computer Science

National University of Sciences & Technology

Islamabad, Pakistan.

2013

CERTIFICATE

It is certified that the contents and form of thesis entitled **"Alleviating ISP Last Mile Network Performance Using Adaptive Nutrition Label Technique"** submitted by *Nooma Roohi (2009-NUST-BE-BICSE-200) and Sajjal Raza Rao (2009-NUST-BE-BICSE-205)* have been found satisfactory for the requirement of the degree.

Advisor: ______________________________

(Dr. Saad Qaisar)

Co-Advisor: ______________________________

(Mr. Abdul Basit)

DEDICATION

To Allah the Almighty

&

To our Parents and Faculty

ACKNOWLEDGEMENTS

Even though it is just the start of our project but it is said that if a starting spark is perfect, the rest goes in the right direction. We are deeply thankful to our advisor and co-advisor, **Dr. Saad Qaisar** and **Sir Abdul Basit** for guiding us in each and every step and guiding us whenever needed. We would have never been able to put our project in this form without their guidance and support.

Table of Contents

List of Abbreviations

AAA	**Authentication, Authorization and Accounting**
ccache	**Compiler cache**
GUI	**Graphical User Interface**
HTML	**Hyper Text Markup Language**
IPC	**ISP Premises Component**
QoS	**Quality of Service**
RADIUS	**Remote Authentication Dial-In User Server**
SDK	**Software Development Kit**
SVN	**Subversion**
ToS	**Type of Service**
UPC	**User Premises Component**

List of Tables

List of Figures

1. INTRODUCTION

Our project is network based whose main objective is to eliminate the bottlenecks at the last mile link and provide round the clock connectivity to its users. The system that we propose is different from the others because it would be incorporated at the users end providing efficient bandwidth utilization as ISPs do not know how to tune up their connectivity with subscribers so that they may experience better service

A lot of progress has been made in the field of networking but the increased internet population can still not be handled by the ISP's. Though Internet Service Providers (ISP) claim to provide promised bandwidth however, network performance varies due to last mile connectivity issues and shared nature of bandwidth. Therefore there is a need to propose such solutions in order to cater this issue

1.1 VISION STATEMENT

The idea is to develop a modified firmware running on the routers to provide efficient service to its user. The proposed system would follow an adaptive QoS tagging technique for the elimination of the last mile links bottlenecks.

1.2 PRODUCT PURPOSE

The purpose of creating this product is to help users enjoy better internet service based on their requirements. Network performance can be improved by traffic monitoring and Traffic engineering. A lot of solutions already exist but have failed to provide with the permanent solution for traffic congestion.

1.1.1 Product Scope

Our product would provide with an adaptive solution i.e. It would observer the traffic pattern generated by the user, assign the nutrition label and then the traffic re-shaping would be done accordingly by the ISP.

1.1.2 Product Perspective

Our product would be a modified firmware running on the router. It would keep on monitoring the traffic pattern generated by the user. The system would run as a daemon.

1.3 PROBLEM STATEMENT

"To provide an adaptive QoS tagging technique for improving home network performance and providing round the clock connectivity for home users."

1.4 PRODUCT DESCRIPTION

We propose a traffic engineered solution comprising of two components:

1. User-Premises component **(UPC)**
2. ISP-Premises Component **(IPC)**

1.1.3 User Premises Component (UPC)

This unit carries out a number of functions at the home router. UPC can be accomplished by is following modules:

i. **Monitor internet traffic** → Sniffs user's traffic for a certain amount of time.

ii. **Calculates Statistics** → Classifies and maintains statistics of different types of traffic based on its priority and pattern.

iii. **Send statistics to IPC** → Opens a socket and send statistics to IPC.

iv. **Send tag to home user** → Notifies home user of assigned tag in form of pop ups.

1.1.4 ISP Premises Component (IPC)

Just like UPC, this block also performs its functions at the ISP end controller. Functions are divided into two modules:

i. **Calculate most generated traffic** → the IPC receives the label via its listening socket and then calculate highly generated traffic pattern of the user.

ii. **Assign Tag** → it assigns a tag to that user accordingly. Purpose of tag is to allocate required bandwidth to user.

iii. **Bill the user** → it also bills the user according to his/her bandwidth consumption.

iv. **Send tag and bill to UPC** → it sends the tag, bill per unit and total bill to UPC.

The UPC would mark subscriber's traffic (nutrition labeling) based on their traffic patterns and will send these patterns for the IPC to tune up its congestion avoidance in an adaptive manner. The system would act as a daemon and perform in dynamic way. Dynamicity implies that the system will periodically check the traffic being generated and will mould the traffic consequently. These components are dependent on each other for their functionality and are further divided into subsections.

1.1.5 Block Diagram

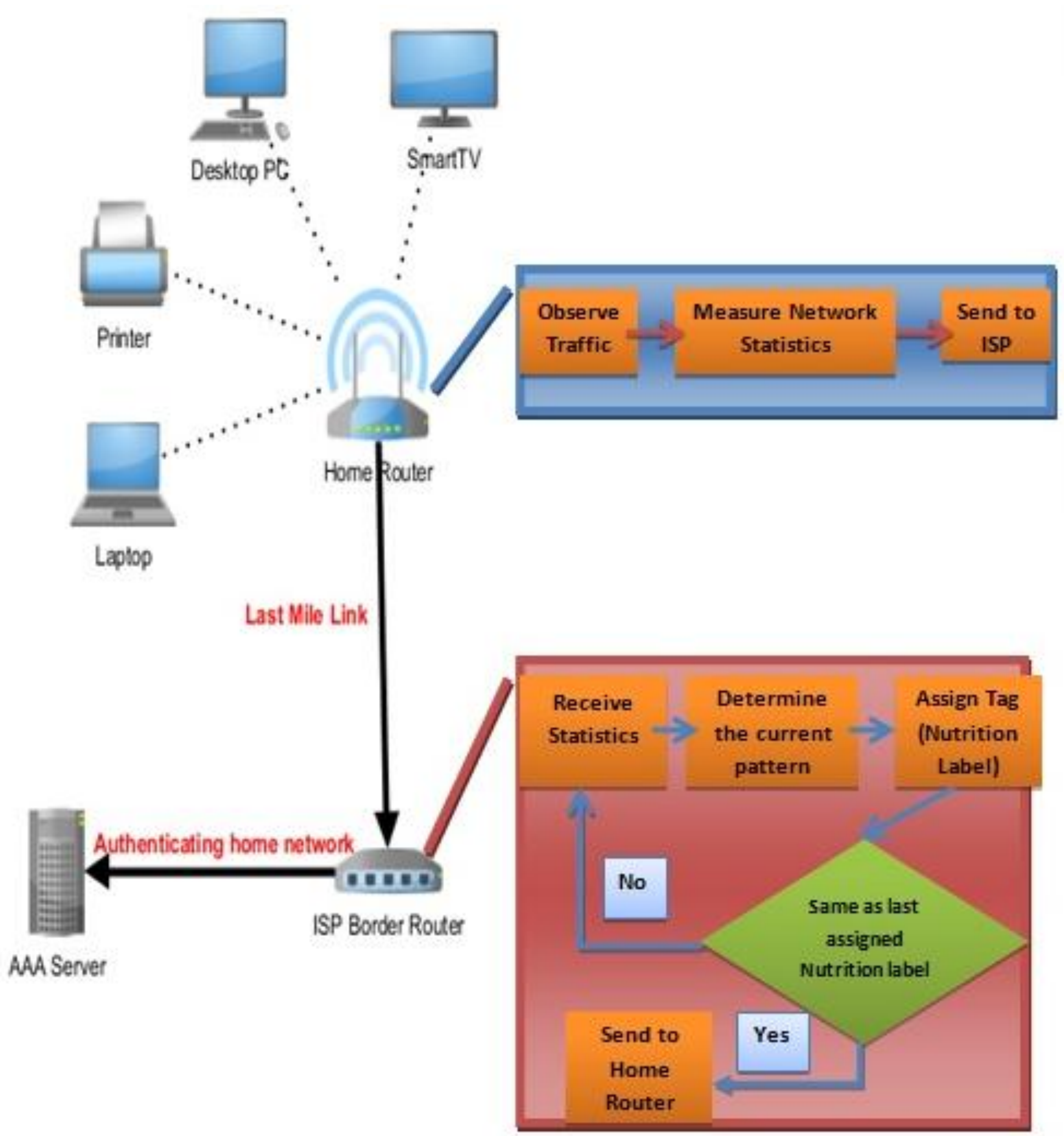

Figure I: Block diagram

Chapter 2

2. RELATED WORK & LITERATURE SURVEY

500 million users across the globe use broadband internet connection [10] for their daily requirements and this number is growing day by day. So, it is important to provide the internet users with the best internet services. ISPs, usually, promise more reliable and fast internet connections by increasing the service charges and spend less on their existing infrastructure. Many researches have been carried out to measure the network parameters and various tools have been developed for this purpose. We present in this section the previous work carried out that is related to our proposed project.

Previously, one time measurements were used to troubleshoot the network performance. These tests were run from websites. Netalyzr [19], Network Diagnostic Tool (NDT) [25] and Network Path and Application Diagnostics (NPAD) [17] are examples of such tools.

K. Cho *et al.* [13], G. Maier *et al.* [6] and M. Siekkinen *et al.* [18] characterized access networks from access ISP using passive measurements. They inferred RRT and throughput of the access network. But it did not measure actual performance of the access network.

D. Croce *et al.* [3] and M. Dischinger *et al.* [16] characterized access networks using probing techniques from servers in wide area. The readings from this technique were inconsistent as servers are far away from the home networks.

The Grenouille project [7] used monitoring agent that were run from a user's home network system. Neti@Home [1] and BSense [5] also used this approach. This approach was not scalable. D. Han *et al.* [4] used open wireless access points to measure the network performance but it did not collect continuous measurements and no insight on the network configuration.

S. Sundaresan *et al.* [27] studied different access network metrics like throughput, latency, packet loss and jitter for benchmarking purposes by taking continuous measurements. Home gateway was used to evaluate the home network performance. They determined performance measurements from two deployments (most of the users had either DSL or Cable connections):

1. **SamKnows**

   It specializes in performance evaluation of access networks. The study consisted of 4200 devices connected across 16 different ISP in US. Gateways were deployed either directly behind the home user's router or behind the home wireless router and were managed and configured remotely.

2. **BISMark**

   It comprises gateways in home, centralized management and data collection servers and measurement server. It collected measurements form a smaller focused group of users from different ISPs and service plans in Atlanta. Device performed active and passive measurements. The gateway was based on the NOX Box [20], and ran a standard Debian Linux distribution.

   For throughput, measurement techniques used were

   - Single/Multi threaded HTTP
   - Passive Throughput
   - UDP Capacity

None of the techniques achieved the value specified in the service plan. Single threaded downloads were adversely affected by the loss rate on the access link than multi threaded TCP. Consistency in throughput was determined by defining a metric (**Avg/P95**). Higher the value of this metric, more consistent the throughput was. However, upload throughput performance was more consistent. For all ISPs, performance variability increased during peak hours. It was observed that loss rate increased for under provisioned ISPs. They also studied effect of traffic shaping on different ISPs, time and users. Many cable ISPs implemented Power Boost, which distorts speed test like measurements. So, any throughput benchmark should aim to characterize both burst rates and steady state throughput rates.

These researchers defined two latency metrics

1. **Last mile latency** → latency to first hop inside ISP's network.
2. **Latency Under load** → latency user experiences during upload/download.

Last mile latency was measured using 'traceroute' command. It was usually quite high. Latency for DSL depends on distance to DSLAM or line quality. Cable providers had lower last mile latency and jitter. Latency introduced under load was affected by capacity of uplink. This factor of increase when uplink is saturated is much higher than when the downlink is saturated. Delay experienced by packets indicated size of the buffer. For a fixed buffer, latency is inversely proportional to draining rate of buffer. Packets saw lower latency during Power Boost. So, latency can be improved by using different data transfer behaviors.

The paper concluded with three important lessons learnt:-

1. **One measurement does not fit all** → ISPs use different policies & traffic shaping behaviors, so cannot be compared to each other.
2. **One ISP does not fit all** →'no' best ISP for all users.
3. **Home network equipment matters** → network topology and hardware affects performance significantly.

S. Sundaresan *et al.* in "**Helping Users shop for ISP's with Internet Nutrition label"** [26] focuses on ISP service plan that carry a nutrition label , that gives more information about the network performance other than the only metrics upload speed and download speed such as throughput, jitter [11] and latency [15] etc. Nutrition label wants user to realize the fundamental network properties .When shopping for ISP only throughput does not reflect the actual performance that the users would receive. They foresee that the ISP service plan should be purchased on their usage patterns through a portal. Label should be accurate, measurable and biggest challenge would be to create awareness among the users.

The metrics defined in the label are:

Throughput: It varies depending upon the condition of the network.

1. **Short-term throughput**: Avg. throughput a user that a user experiences during initial period of transfer.
2. **Sustainable throughput**: Throughput a user should receive after an initial period of higher throughput.
3. **Minimum throughput**: Captures network load.

Last-mile Latency:

1. **Baseline last-mile latency**: The minimum latency to the border router in ISP network.
2. **Maximum last-mile latency**: Larger value of last mile latency which a user experiences due to congestion.

Jitter → Important for multimedia and interactive applications

Loss Rate [11] → Capture the average loss over a sustained period.

It is Difficult for an ISP to predict about the performance a user will receive, performance may get affected by local loop, wiring equipment.

A survey was carried out in 6 houses, 2 of them used cable net, 3 DSL users and 1 Wimax in order in order to observe the metrics. Wimax and DSL users had their throughputs same as the advertised value as there is no power boost [23][24] in them. Cable net, power boost due to which the short term throughput is higher than the sustained throughput. In both cable networks they have their peak values different from each other as compared to the single advertised service plan.

The paper suggested that the future work is to create awareness among the users and asking users about the internet usage, then user can choose the service plan in a better way and also nutrition label should be dynamic.

K. Calvert *et al.* in **"Instrumenting Home networks"** [12] discussed about troubleshooting the problems at home network. Users experience the network problems and are unfamiliar with them so management of the home network is a global security issue as home users are unable to properly configure and secure their home networks. Such tools need to be developed that diagnose, recognize and correct the home network problem by eliminating the factors leading to power degradation [14]. So an autonomous, comprehensive and scalable logging platform is required, the researchers named such system named as HDNR (home network data recorder). Home network data recorder would record the home network's activities .In this case a simple home network was considered with a laptop, printer, PC and a game controller all connected to the router.

A sensor monitoring all the activities & events taking place in the home network, it collects the raw data at the packet level which gets processed into higher level summaries. The primary application of the troubleshooting can be provided by the ISP's themselves or the 3rd party subscribers.

Such services would require access to the original data of the user but only the trusted applications would have access to do that but the un-trusted applications would not be able to see the full data but see it after the data is aggregated with other homes and by trusted components.

It has some potential following applications:

1. Internet performance measurement(NANO)

2. Network security

3. Network trouble shooting

The researchers have implemented the initial prototype HNDR data collection system in home networks. The prototype design is based on "NOX Box" [20] which runs the linux kernel. It has a 500MHz Geode processor, 3 Ethernet ports and an 802.11b/g access point. TCP dump runs on the each of the wired Ethernet. The packets get captured but the packets got dropped when three different types of internet loads were observed due to the reduced buffer size .But after increasing the buffer size no packets were dropped.

So, the conclusion drawn from this paper is that home network management can be made easy by deploying the HDNR and monitoring the home network traffic.

Now researchers are focused on studying the access networks and different parameters that affect its performance. Based on the above three research papers of Nick [12][26][27] and his co-workers, we decided to take a step further. The three techniques applied in the research papers will be used to develop such a system that will monitor the network traffic and restructure the traffic using an effective marking technique.

3. METHODOLOGY

We divided our project into two main parts, namely UPC and IPC. Functionality and design of each component is given below.

3.1 USER-PREMISES COMPONENT (UPC)

We started working on the User-Premises Component (UPC) and accomplished the following tasks:

1. Identifying hardware
2. Identifying firmware
3. Creation of a package
4. Init script
5. Modifying GUI

Details of these tasks are given below:

3.1.1 Identifying hardware

Hardware, basically, involves the router. We have used TP-Link TL-WR1043ND [29]. It is an ultimate N Gigabit wireless router. It is also a common home router.

3.1.2 Identifying firmware

The router's firmware that we have used for our project is "OpenWrt" [21] which is a Linux based distribution for embedded systems. It consists of the following four versions:

1. White Russian
2. Kamikaze
3. Backfire
4. Attitude Adjustment

Attitude Adjustment is the latest distribution. We flashed its binary file on the router and configured its network settings using its LUCI (Web GUI). Now, the router became OpenWRT supported.

3.1.3 Creation of Package

For creating an entire package, we created three sub packages, which are as follows:

1. Sniffer
2. Client
3. Server

All these packages were created using:

- **a. Source file** → source code was coded in C language.
- **b. Makefile** → we created our programs in Ubuntu 12.10, so we compiled and build it for OpenWRT by using SDK [30]. SDK is a precompiled toolchain which is used to cross-compile [1] a single user defined package for a specific target. For building our program for the router we defined a Makefile [9]. A Makefile tells the SDK how to compile and build the package. Finally, we got an **".ipk"** file after successful compilation of the c code.

Packages are discussed in detailed below.

3.1.3.1 Sniffer

It sniffs incoming packets on LAN interface "eth0.1" of router for a certain time, in our case 10 seconds. The captured packet is characterized according to seven different defined categories and a count is maintained for each category. These categories are as follow:

Table 1: Categories defining TOS of outgoing packets

Category (TOS)	Process name (Example)	Application Protocol
Web	Firefox, Chrome, Safari, Opera, httpd, plugin-container, WebKitPluginHost	HTTP(S)
File Transfer	ftp, dropbox, svn, git, SW updates	ftp, rsync, svn, cvspserver
Email	Mail, Outlook, Thunderbird	IMAP(S), POP3(S), (S)SMTP
Chat	Skype, iChat, Adium, Pidgin	ircd, SIP, msnp, snpp, xmpp
Personal	Media players, games, productivity	Rtsp
CMD	Ssh, telnet	Ssh, telnet
Besides IP	Every packet that is not IP based	ARP, RARP, echo-reply etc.

After 10 seconds, the program saves the statistics in a file (.txt) and sends it to IPC by opening a socket on port 51717. Afterwards, it again starts sniffing.

For sniffing packets, we have used ***libpcap*** [28] library. Pcap (packet capture) library contains built in functions and supports filtering of packets.

3.1.3.2 Client

Client's sole purpose is to send the assigned tag received by UPC from IPC to home user's computer. For this purpose, it uses port 51719.

3.1.3.3 Server

Server listens for response from IPC on port 51718. Above mentioned client package is also included in it.

3.1.4 Init Script

Init scripts in OpenWRT are quite easy to build. We created an init script by the name of "main.sh" to run server and sniffer package on start up. This file is place in "/etc/init.d" directory of OpenWRT running on the router. The permissions to run this script are set by "chmod +x /etc/init.d/main.sh" command. To start the script manually type "/etc/init.d/main.sh start" by accessing console (CLI) of firmware through ssh. To start the script on start up type "/etc/init.d/main.sh enable" in console panel.

3.1.5 Modifying GUI

We created an html page, for home user, to access his/her current traffic pattern, assigned tag and bill easily. Traffic patterns are displayed in form of graphs. The page refreshes itself after 10 seconds to get the latest readings from the obtained/received files. This html page can be accessed from routers default page by typing router's URL in address bar of any browser. In our case, we access the Luci of router using "192.168.2.1".

HTML and JQuery are used for creation of the html page and accessing data from files. Highcharts [8] plug in is used for plotting graphs. Following figures show traffic graphs and total generated traffic from home router.

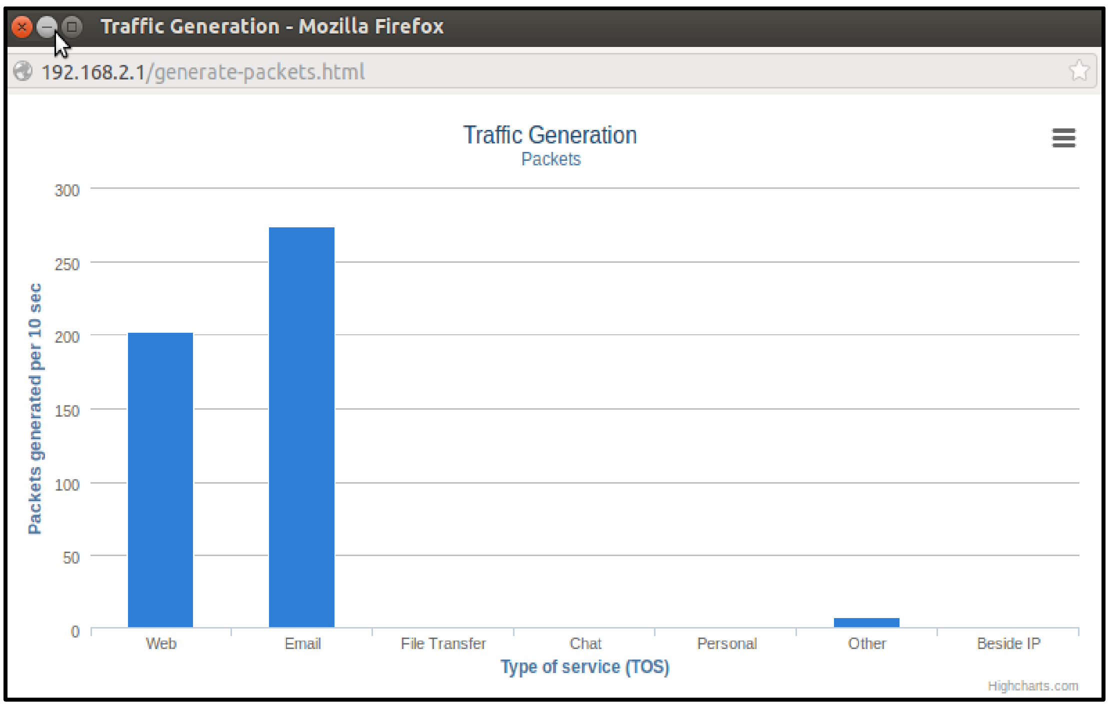

Figure II: Graph showing traffic generated

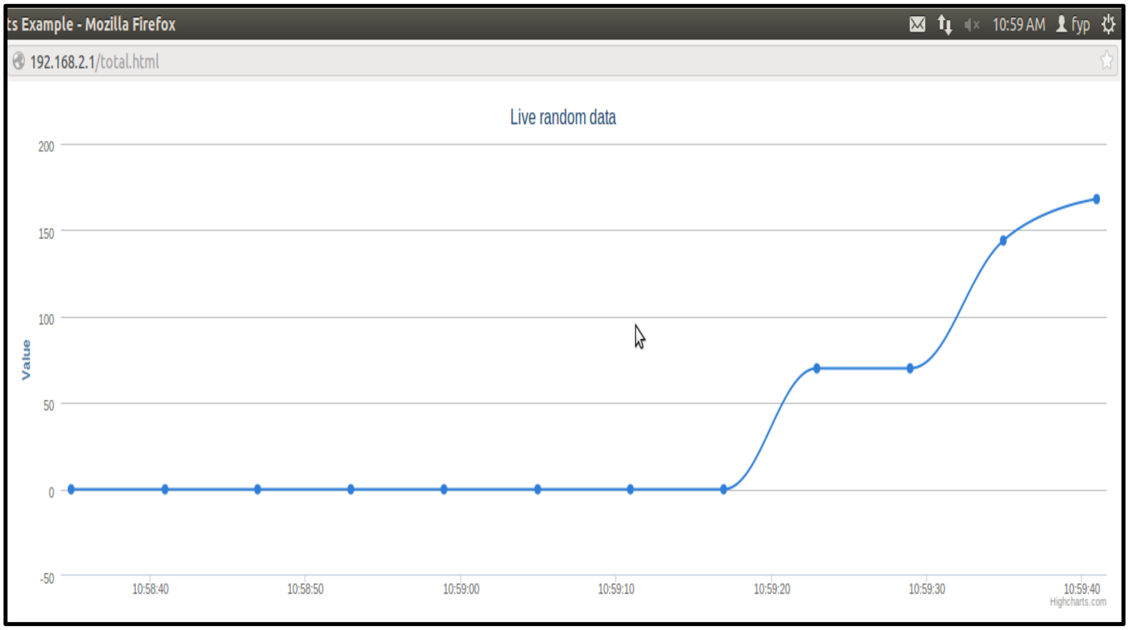

Figure III: Graph showing total generated traffic

3.2 ISP PREMISES COMPONENT (IPC)

IPC, mainly, consists of a controller/server at the ISP end. This controller performs following functions:

1. Listens for statistics from UPC on port 51717.
2. Identifies the current traffic pattern
3. Assigns a tag to most generated traffic pattern. These tags represent what type of characteristics the bandwidth should have. Following table entails the tags assigned and nutrition label metrics for bandwidth.

Table 2: Nutrition Label Metrics

Nutrition Label Metric	Type of Service	Tag No.
Loss Rate	Web	0
Short-term throughput	Email	1
Sustainable throughput	File Transfer	2
Max. Jitter	Chat	3
Baseline last-mile latency & Max. Jitter	Personal	4
Normal throughput	CMD/Besides IP	5/6

4. Calculates bill for the user. Our system bills according to usage of bandwidth which is determined by the current traffic generation of the user.

5. Send the bill and assigned tag back to IPC on port 51718.
6. IPC also has a gui for ISPs. This GUI contains graphs for the total generated traffic for a particular user. Again,GUI is built using HTML, JQuery and Highcharts [8] .

3.3 ADDITIONAL FEATURE

For convenience of home user, we have also added pop up service displaying the currently assigned tag to the user without accessing router's web page from a browser.

Home user's machine listens for response from the home router. A server running on port 51719 serves this purpose. The data received is pushed into a utility named "notify-send" for generating pop ups on a linux machine. Figure shows the popup generated when the assigned tag is changed to "Tag6: Besides IP" along with router's modified webpage.

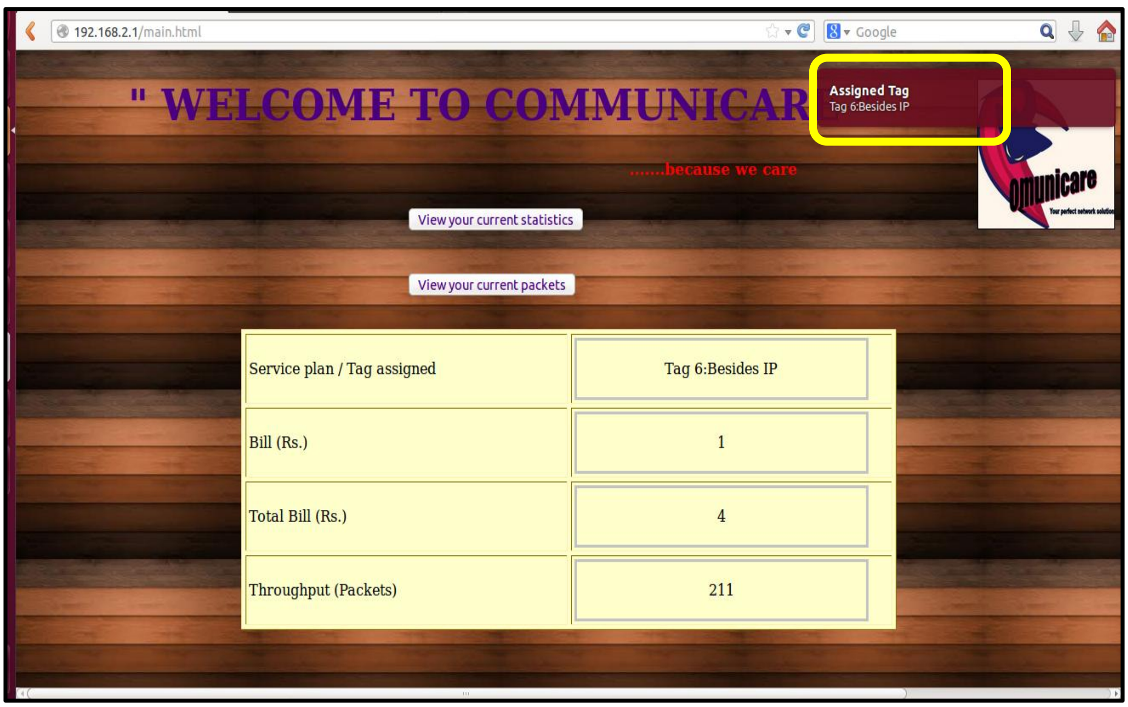

Figure IV: Router's webpage & popup

4. RESULTS

We completed our project step by step and were able to setup a test bed. The test bed consisted of IPC server, UPC router and systems connected to UPC representing home users. A wi-tribe [31] connection of 1Mbps was used for accessing the internet. The internet connection and wan port of UPC i.e. eth0.2 was connected to an unmanageable D-LINK switch. The three servers running on ports 51717 (IPC), 51718 (UPC) and 51719 (home user machine) always listen for incoming connections from their respective clients. The information received on these servers is interpreted and a certain task is performed. For example, the server running on IPC calculates most generated traffic; total bill after it receives the data and updates its records and IPC records too. Similarly, server running on UPC receives this updated information from IPC and sends this record to home user machine which displays a popup.

We intend to make home user and ISP aware of bandwidth usage. The graphs generated from our system represent statistics for a single home network. The ISP workers and the home user can monitor the traffic patterns. Observer can also see the current assigned tag for the home network and can verify whether he/she is assigned the proper, required bandwidth or not. The tags, we have used, are comprehendible by a lay man. This way we can get home users educated technically.

The system can also avoid bottle necks in the last mile link. ISPs can monitor present condition of the link by the help of visual graphs. They can observe which home network is congesting the link and what type of service is consuming the bandwidth. Hence, ISPs can take steps to avoid such situations.

The system provides a mechanism for dynamic billing. The user will pay for when and how much he is using. This relives the user from giving a constant bill that he pays whether he uses the bandwidth or not. This satisfies the user as he knows what he is paying for.

Besides that the system tends to focus on efficient and effective bandwidth utilization for the home network as well as the ISPs as bandwidth is a scarce resource and is very expensive. This is done by assigning minimum bandwidth to the network when the last mile link for it is idle and essential bandwidth for different types of traffic pattern when the home users are active on the network. ISPs can also manipulate home users' behavior and come up with such packages or schemes that will allow resourceful usage of their bandwidth and at the same time increase their earnings. Home user can also get an improved quality of service (QoS) as they will get necessary bandwidth for their activities.

In short, our system not only provides a better solution to ISP clients who buy service plans with less bandwidth than required but also to those who purchase high bandwidth but their required internet traffic needs far less than that. This wastage of bandwidth can be tactfully handled by the system we have designed.

5. DISCUSSION

We started off the project by first reading and understanding the three research papers of Nick Feamster [26][27][12]. The research papers gave us an overview of our project. We came up with the project idea by consulting and brainstorming with our advisor and co-advisor.

We created an initial SRS document to identify requirements, different components, their functions and their relationships. This was done using entity relationship diagrams, test case diagram. We identified that our system will have two components i.e., UPC and IPC. These components were interdependent and performed a number of functions. Since this document, our requirements and functionalities changed as the project progressed.

Our first task was implementing the UPC component. For this purpose we first identified the hardware i.e., the router we were going to use. The router was TP-Link's TL-WR1043ND. Next we identified the firmware we are going do it. We came up with OpenWRT. It is Linux distribution for embedded devices and is highly customizable. So we flashed our router with Attitude Adjustment (a version of OpenWRT) and configured the internet settings according to SEECS network. Later we found out that router has limited memory storage and we need to build our own customized image.

Building an actual firmware was one hell of a task. As this was a totally new thing to us and none of our peers had done it before. After sending hours of searching we finally figured how to build the image. We build two images.

1. RAMdisk image → used for testing purposes and can run without flashing it on router using serial port of router.

2. Squashfs image → it is burnt on router to run it and increases the nand count of router.

Next step was to create a package for sniffing the packets from routers and identify the types of packets. For it, we searched and came up with libpcap. This packet capturing library sniffs the packets on desired interface and can also analysis the packet as well as filter it. We decided to filter packets based on destination ports. From ports we identified the type of service user was requesting like web, e-mail, file transfer etc. After creating a program which worked fine on Ubuntu 12.10, our next challenge was to make it OpenWRT compatible so that it can run on the router.

We found a method to write the required makefiles and got to know the SDK for building the package. The building of package was quite time consuming. We encountered countless errors while building. We had errors of makefiles, ccache program, libraries compatibility issues. We tried each and every possible solution and finally solved the error. Finally, we got our first package prepared to be installed in the router. We installed, ran it and it worked according to our expectations.

Next was to send the statistics to IPC. Initially we made a simple client and server program and tested it on one of the computers. The client code was integrated with our package and the other computer was made server to accept the connections from router. After this worked, we explored on how to setup a hypothetical isp. Due to lack of time we thought of setting up AAA server that will authenticate the home networks and allocated them bandwidth. We did setup a RADIUS server but unfortunately lack of a IEEE802.1X switch and unavailability of bandwidth to allocate to home users, we had to drop idea of setting up this server.

Finally we decided to build a simple server that will perform the functionality of an ISP controller. We coded for the server functionalities and then synchronized the programs and packages. Now we wanted to create a GUI and webpage for both home users and ISP server. We coded in HTML and Jquery and got our webpage along with graphs representing total traffic and currently generated traffic. We finished our project by adding a popup service for the home user. Using this popup service user can easily see his/her presently assigned tag.

One of the major problems was SEECS internet connection. It redirects the entire traffic to proxy server. This way sniffer could sniff only traffic for web. So we decided to use wi-tribe [31] connection to get the traffic for email, file transfer and other services. Moreover, blocked ports of svn and git gave us problems as we had to build our program from homes.

6. CONCLUSION

Internet usage increased exponentially during last century and is still on rise. This increase in demand puts high pressures on ISPs as they thrive to proficiently utilize their limited bandwidth. Whereas users want a better quality of service more than ever. They desire for improved home network performance.

We present a solution of implementing such a system that will dynamically assign bandwidth to home networks according to type of traffic the user is generating. The system consists are two dependent components namely IPC and UPC. UPC will determine the type of traffic the user is generating and send to IPC for assignment of a tag called nutrition label. IPC will identify the highly generated traffic and assign the corresponding tag and assign the required bandwidth. The system also provides the home users and ISPs with visual graphs and popups for better understanding the behavior of the home users.

The system has a number of advantages both for ISPs and home users. ISPs can avoid last mile link congestion and ensure throttle free service. ISPs can maneuver service plans according to home user usage. They can also effectively and efficiently use their bandwidth. User can become more technically educated. He/She is provided with a facility dynamic billing and user will pay for when and how much he/she uses. Moreover, ISPs and home users will be aware of traffic patterns because of the visual aids provided in form of graphs and popup services.

7. RECOMMANDTIONS

1. The system can be implemented for multiple home networks by creating a database using MySQL.
2. It can be improved by adding a proper billing mechanism with or close to market rates.
3. The test bed can be implemented to actual home networks. This can be done by contracting with ISPs.
4. We can manipulate the user's usage pattern by devising service plans that can not only benefit users but also ISPs.
5. Such service plans can also help reduce congestion of last mile link during peak hours.

9. REFERENCES

[1] C. R. S. Jr. and G. F. Riley. Neti@home: A distributed approach to collecting end-to-end network performance measurements. In the Passive and Active Measurement Conference (PAM), 2004.

[2] Cross Compile, http://wiki.openwrt.org/doc/devel/crosscompile

[3] D. Croce, T. En-Najjary, G. Urvoy-Keller, and E. Biersack. Capacity estimation of adsl links. In CoNEXT, 2008.

[4] D. Han, A. Agarwala, D. G. Andersen, M. Kaminsky, K. Papagiannaki, and S. Seshan. Mark-and-sweep: Getting the inside scoop on neighborhood networks. In Proc. Internet Measurement Conference, Vouliagmeni, Greece, Oct. 2008.

[5] G. Bernardi and M. K. Marina. Bsense: a system for enabling automated broadband census: short paper. In Proc. of the 4th ACM Workshop on Networked Systems for Developing Regions (NSDR '10), June 2010., 2010

[6] G. Maier, A. Feldmann, V. Paxson, and M. Allman. On dominant characteristics of residential broadband internet traffic. In ACM Internet Measurement Conference, 2009.

[7] Grenouille. Grenouille. http://www.grenouille.com/.

[8] Highcharts JS, http://www.highcharts.com/

[9] Image/Makefile Details, http://wiki.openwrt.org/doc/techref/image.makefile

[10] Internet World Stats. http://www.internetworldstats.com/dsl.htm.

[11] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP Throughput: A Simple Model and its Empirical Validation. In Proc. ACM SIGCOMM, pages 303–323, Vancouver, British Columbia, Canada, Sept. 1998.

[12] K. Calvert, W. K. Edwards, N. Feamster, R. Grinter, Y. Deng, X. Zhou. Instrumenting Home Networks. In ACM SIGCOMM Computer Communications Review Volume 41, Issue 1, January 2011.

[13] K. Cho, K. Fukuda, H. Esaki, and A. Kato. The impact and implications of the growth in residential user-to-user traffic. In ACM SIGCOMM 2006, 2006.

[14] M. bin Tariq, M. Motiwala, N. Feamster, and M. Ammar. Detecting Network Neutrality Violations with Causal Inference. In Proc. CoNEXT, Dec. 2009.

[15] M. Dick, O. Wellnitz, and L. Wolf. Analysis of factors affecting players' performance and perception in multiplayer games. In Proceedings of 4th ACM SIGCOMM workshop on Network and system support for games, NetGames '05, pages 1–7, New York, NY, USA, 2005. ACM.

[16] M. Dischinger, A. Haeberlen, K. P. Gummadi, and S. Saroiu. Characterizing residential broadband networks. In Proc. ACM SIGCOMM Internet Measurement Conference, San Diego, CA, USA, Oct. 2007.

[17] M. M. et al. Network Path and Application Diagnosis. http://www.psc.edu/networking/projects/pathdiag/.

[18] M. Siekkinen, D. Collange, G. Urvoy-Keller, and E. Biersack. Performance limitations of adsl users: A case study. In the Passive and Active Measurement Conference (PAM), 2007.

[19] Netalyzr. http://netalyzr.icsi.berkeley.edu/.

[20] NOX Box. http://noxrepo.org/manual/noxbox.html.

[21] OpenWRT, www.openwrt.org.

[22] OpenWRT. https://openwrt.org/.

[23] PowerBoost FAQs – Charter, http://myaccount.charter.com/customers/Support.aspx?SupportArticleID=2338.

[24] PowerBoost FAQs – Comcast, http://customer.comcast.com/help-and-support/?CatId=377.&SCRedirect=true

[25] R. Carlson. Network Diagnostic Tool. http://e2epi.internet2.edu/ndt/.

[26] S. Sundaresan, N. Feamster, R. Teixeira, A. Tang, W. Edwards, R. Grinter, M. Chetty, W. De Donato. Helping Users Shop for ISPs with Internet Nutrition Labels. In ACM SIGCOMM HomeNets Workshop Toronto, Ontario, Canada, August 2011.

[27] S. Sundaresan, W. De Donato, N. Feamster, R. Teixeira, S. Crawford, A. Pescape. Broadband Internet Performance: A View From the Gateway. In ACM SIGCOMM Toronto, Ontario, Canada, August 2011.

[28] The Sniffer's guide to raw Traffic, http://yuba.stanford.edu/~casado/pcap/section1.html

[29] Ultimate Wireless N Gigabit Router TL-WR1043ND, http://www.tp-link.com/en/products/details/?model=TL-WR1043ND

[30] Using the SDK, http://wiki.openwrt.org/doc/howto/obtain.firmware.sdk?s[]=makefile

[31] Wi-tribe, http://www.wi-tribe.pk/