

Distributed Network Traffic Controller

Student: Chadwick Johnson

Course number: 22c:198

Credits: 3 sh

Course Advisor: Dr. Ramon Lawrence

Internet access is a resource that is vital to the every day functionality of business and campus activity. Bandwidth utilization is often a problem for companies and campuses where abuse is overwhelming and causes all traffic to come to a stand still. A possible solution would be setting some form of fair constraints on this traffic that allows for the people who are less abusive to receive faster access.

My senior year as an undergraduate, I had created a system that would act as an Internet Gateway, and allow traffic to be placed in “pipes” based on how much usage an individual had created by using an IP address as a point of reference. This past project was successful with its ability to throttle bandwidth and create the pipes, but was found lacking in many areas. For instance, it used a flat file for storage, did not have an administrative interface and all traffic was valued the same. Another drawback was that this design only allowed for a single gateway. My plan for this project is to expand on the old project, and allow for multiple gateways, an administrative interface, and everything will be coordinated through redundant SQL servers.

This problem has two major aspects, which I will refer to as “what” and “why”. “What” references the system that will do the throttling, and is important because a single gateway means if that single device fails, the internet is down for everyone. Also, if there is too much traffic for a single gateway, a new bottleneck is created. The “why” refers to the reason an IP address is throttled. A system should allow for variation on how much is considered fair usage.

The main reason for this problem is simply too much demand and a lack of resources. This problem is commonly created with peer-to-peer file sharing programs, but can happen for a variety of other reasons. Currently this problem has a variety of solutions, such as blocking certain types of traffic or watching traffic for abuses like saturation of an internet connection and then giving warnings and blocking computers. Supposedly there are pieces of software that can be purchased for large sums of money that offer similar features, but only allows for a single gateway. I have not yet found a setup that allows for multiple gateways.

Some improvements to this project are well outside the scope of this class and could include an implementation of a protocol, such as IPSec, that verifies the user so all traffic is associated with that user, and spoofing is almost impossible. Any optional components that are not integrated would also be improvements, but may not happen due to time constraints.

Project Summary - (see figure 1 at the end of this document)

The basic plan is to start with a set of two computers set up as gateways that are synchronized through two redundant SQL servers. This system will be managed through a web front end, and will allow for modifications to the distributed system. These systems maintain records on how much traffic is created per IP, and allows for throttling based on amount of traffic by placing users into either an open pipe or a slow pipe. A web interface will also allow each system to see how much traffic they have created, and their history. A presentation of this project will be given along with all documentation necessary to make this count as a substitute for the Software Engineering project so I could get my Software Engineering subtract. I am going to be attempting a Hybrid Incremental / Extreme Programming approach.

Should the above level be completed, hopefully an A level grade would hopefully be negotiated. If enough time is available in the semester, I have other things I would like to see happen, but my not be possible. These “add-ons” would border on optimism for a project this size. One thing is I would really like to make this system scalable so it allows for gateways to be added or removed. Also I would like the administrative interface to have the ability to retrieve displays of performance per server. I would like to see the ability to have the gateways act as firewalls, so certain ports and IP addresses can be blocked. Besides that, it would be nice if management had the ability to allow for certain ports and addresses to count as different weights. For example, download.com may count for 75% of normal traffic, while FTP traffic counts for 120% of the actual cost. Also, it would be nice to have more pipes to be allowed. Should all this be completed, I would like to have a set of instructions and all code published as an Open Source solution.

Project Details

Resources will include Operating Systems, Software, Hardware, and references. The operating systems I have available to me are Fedora 2.0, Windows XP, and Windows 2003. Software I have available include PostgreSQL, Jude, DHCP, IPTables, and CBQ. DHCP is open source and allows for distribution of IP addresses upon request when a computer connects to the network. IPTables is built into the Kernel of the Linux OS, and allows for IPforwarding after a Kernel recompiling. CBQ (also open source) is an open source shell script which does the actual throttling. Hardware consists of network cables, switches, and network cards. All of the above I have and am ready to commit to this project. I currently have a laptop, and two mid to low end computers that I can commit, but I expect this project to require a minimum of three more computers. I was hoping to request the additional computers from surplus, but I am willing to purchase them myself. I also have several Python, Java, Linux, Database, and Security books. Should I need any additional references, I will find them online, borrow them, or purchase them. As for space, I was planning on having these computers in my office unless a problem arises.

The most difficult issues will probably be the interfaces. Traffic reports will be updated in the database through JDBC or Python and the web interfaces will be done in JSP. My project in the past used Python, but the only interfaces were with plain text files or a webpage that displayed a barely-dynamic webpage, and no administrative GUI or options were established. My project is unique because of the use of Open-source tools, which I am hoping will make this a project that could be freely used by anyone in the world.

Deliverables

- Project Plan – Goes into detail about what the project is supposed to accomplish.
- High Level Design – Describes how the things in the project plan work.
- Test Plans – A description of the various techniques used to test the final project.
- Distributed Network Traffic Controller + instructions on use – The actual program written. Chances are this will actually be several individual programs that all work together to
- Test Results – Results collected from the test plans which include bugs found and performance results.
- *Instructions on how to create project from the start and all code.
- Short Report containing assessments of data collected
- Concluding overview of whole project

* = **Optional**

Timeline

Week 1+2	Project Plan, High Level Design, Test Plans
Week 3	Get together hardware and set up systems
Week 4	DHCP + IP forwarding
Week 5+6	Database setup + Throttling
Week 7+8	Administrative Screens
Week 9+10	Setup and integration of second gateways
Week 11	User screens
Week 12+13	Attempt optional components if time allows
Week 14	Testing
Week 15	Assessment of data collected and overview of whole project.

Conclusion

This project is something I have personal interest in. It involves extending a networking project I already have knowledge of and integrates a database for ease of use and efficiency. If I am successful, this project will be a great exercise in database design, along with bettering my understanding in the networking and database field, along with the possibility of delivering an Open Source solution to a problem any individual with more than one computer attached to an Internet connection may have.

Figure 1

