Web Proxy Server

And Caching

Submitted To:

Dr.Buddha Singh

Ms. Arpita Jadav

Submitted By:

Priyanka (11103534)

Kshama Singh (11103493)

Varsha Agarwal (11103616)

Karan Dua (11103485)

Batch-B7

INTRODUCTION

WHAT IS A WEB PROXY SERVER?

In computer networks, a **proxy server** is a server (a computer system or an application) that acts as an intermediary for requests from clients seeking resources from other servers. A client connects to the proxy server, requesting some service, such as a file, connection, web page, or other resource available from a different server and the proxy server evaluates the request as a way to simplify and control its complexity. Proxies were invented to add structure and encapsulation to distributed systems. Today, most proxies are **web proxies**, facilitating access to content on the World Wide Web and providing anonymity.

Web proxies forward HTTP requests. Some web proxies allow the HTTP CONNECT to set up forwarding of arbitrary data through the connection; normally this is only allowed to port 443 to allow forwarding of HTTPS traffic.

Examples of web proxy servers include Apache (with mod_proxy or Traffic Server), HAProxy, IIS configured as proxy (e.g., with Application Request Routing), Nginx, Privoxy, Squid, Varnish(reverse proxy only) and WinGate.

WHY IS IT NEEDED?

- **Monitoring and filtering**

A web proxy server can be used to monitor the content that is passing over the network. This content can also be filtered by the owner of the server

- **Content-control software**

A content-filtering web proxy server provides administrative control over the content that may be relayed in one or both directions through the proxy. It is commonly used in both commercial and non-commercial organizations (especially schools) to ensure that Internet usage conforms to acceptable use policy.

Requests may be filtered by several methods, such as a URL or DNS blacklists blacklist, URL regex filtering, MIME filtering, or content keyword filtering. Some products have been known to employ content analysis techniques to look for traits commonly used by certain types of content providers. Blacklists are often provided and maintained by web-filtering

companies, often grouped into categories (pornography, gambling, shopping, social networks, etc.).

.

- **Filtering of encrypted data**

Web filtering proxies are not able to peer inside secure sockets HTTP transactions, assuming the chain-of-trust of SSL/TLS has not been tampered with.

- **Bypassing filters and censorship**

If the destination server filters content based on the origin of the request, the use of a proxy can circumvent this filter. For example, a server using IP-based geolocation to restrict its service to a certain country can be accessed using a proxy located in that country to access the service.

In some cases users can circumvent proxies which filter using blacklists using services designed to proxy information from a non-blacklisted location.

- **Logging and eavesdropping**

Proxies can be installed in order to eavesdrop upon the data-flow between client machines and the web. All content sent or accessed – including passwords submitted and cookies used – can be captured and analyzed by the proxy operator.

- **Improving performance**

A **caching proxy** server accelerates service requests by retrieving content saved from a previous request made by the same client or even other clients. Caching proxies keep local copies of frequently requested resources, allowing large organizations to significantly reduce their upstream bandwidth usage and costs, while significantly increasing performance.

- **Accessing services anonymously**

An anonymous proxy server (sometimes called a web proxy) generally attempts to anonymize web surfing. There are different varieties of anonymizers. The destination server (the server that ultimately satisfies the web request) receives requests from the anonymizing proxy server, and thus does not receive information about the end user's address. The requests are not anonymous to the anonymizing proxy server, however, and so a degree of trust is present between the proxy server and the user. Many proxy servers are funded through a continued advertising link to the user.

- **Security**

A proxy can keep the internal network structure of a company secret by using network address translation, which can help the security of the internal

network. This makes requests from machines and users on the local network anonymous. Proxies can also be combined with firewalls.

LITERATURE SURVEY

History of Proxy Servers:

Proxy servers have been around since the dawn of the Web but are now facing competition from NAT and firewalls. To prevent extinction, their role is evolving from providing guard-dog security and connection sharing to content caching and authentication.

Proxy Past

At its simplest, a proxy server is a layer sitting between a local-area network (LAN) and an external network such as the Internet. Proxy servers came about to meet several needs.

1. They enabled several machines to share a single Internet connection by accepting and forwarding requests from client applications.
2. They could regulate, allowing or disallowing certain communications with the outside world, such as through site filtering.
3. They could conserve bandwidth and increase network efficiency by caching content for repeated local delivery.

Proxy servers share Internet access at the application level, which means every client program must be individually configured to talk to the proxy server. This is an effective way to allow extremely limited kinds of Internet access, but many organizations found the configuration requirements to be a burden. With the development of Network Address Translation (NAT), organizations could share an Internet connection at the network level, which greatly simplified the process.

Proxy Present

But proxy servers aren't completely dead. The role of the proxy server is shifting away from guard-dog security and connection sharing toward content caching and authentication.

Unlike firewalls and NAT, proxy servers can extend their reach beyond a physical LAN. There is increasing demand for portable authentication. An example of this is a university that allows students to access subscription-based third-party services. When students are not on campus, they might access the services through a university proxy server that passes their authentication to the third-party.

Content caching remains a valuable tool for organizations wishing to conserve bandwidth and provide a speedier connection to local users. One of the more popular caching proxy servers is the venerable free and open source Squid, which still enjoys active development. Traditional firewalls don't usually include content caching. Instead, hybrid suites are emerging that pack both firewall features and proxy server features under a larger umbrella.

Microsoft ISA, Kerio WinRoute Firewall, Avirt Soho, and 602LAN Suite all represent this latest creature. These suites emphasize the NAT and firewall features that networks rely on today — while also retaining proxy features, including SOCKS support (the proxy protocol) — to support client-proxy connections for content caching and authentication.

Motivation

A web cache proxy server, or caching proxy is essentially a shared web cache. It is a web cache program running on a dedicated server that archives and returns documents frequently requested by a group of web clients. The proxy server returns an object from its archive if it locates a local copy of the object (a hit), or retrieves and archives the requested object if it does not find a local copy. It uses the absolute URL of archived objects to locate and identify hits. By returning local copies of objects whenever possible, a caching proxy can improve response time, reduce network traffic and increase the effective bandwidth available to end-users. Cache servers avoid delivering stale objects and free up space for new objects by expiring objects and by not caching dynamically generated content. Caching proxies typically implement a Least Recently Used (LRU) policy to determine when to remove objects from their archive

Work Distribution

Name	Roll No.	Work
Karan dua	111	Content Caching, Content Blocking, GUI, Conditional Request Cache Replacement Connected Clients Log, Request-Response Logging
Varsha agrawal	11103616	Cache replacement,Request-Response Logging, Testing, GUI (server interface), Hit-Miss Ratios, Project Report, Cache Replacement ,Connected Client Record
Kshama singh	11103493	Cache replacement, GUI (file browser), Multi-threading, Testing, Cache Replacement ,Connected Client Record
Priyanka luthra	11103534	Logging,Research paper, GUI (server interface), Cache Replacement (based on number), Testing, Hit-Miss Ratios

Requirements

- ## Hardware Requirements

Microprocessor	2.4 GHz Intel Core i5-2430M
Microprocessor Cache	3 MB L3 cache
Memory	4 GB 1333 MHz DDR3
Memory Max	Upgradeable to 16 GB DDR3
Display	1366 x 768
Hard Drive	750 GB SATA (5400 rpm)

Network Card	Integrated 10/100/1000 Gigabit Ethernet LAN

- **Software Requirements**

  • Linux Operating System, Windows Operating System
  • Connectify
  • Documentation: Microsoft Word

Problem Definition

A web cache proxy server, or caching proxy is essentially a shared web cache. It is a web cache program running on a dedicated server that archives and returns documents frequently requested by a group of web clients. The proxy server returns an object from its archive if it locates a local copy of the object (a hit), or retrieves and archives the requested object if it does not find a local copy. It uses the absolute URL of archived objects to locate and identify hits. By returning local copies of objects whenever possible, a caching proxy can improve response time, reduce network traffic and increase the effective bandwidth available to end-users. Cache servers avoid delivering stale objects and free up space for new objects by expiring objects and by not caching dynamically generated content. Caching proxies typically implement a Least Recently Used (LRU) policy to determine when to remove objects from their archive

Terminology: We will now present how we interpret some central terms that are used throughout the report.

- The concept of caching is storing a copy of data close to the client in order to allow faster access. Another benefit is that communication with the place from which the data originated is reduced.
- A proxy is a node in the network between a client and a server which the traffic passes through. A proxy can be used for several purposes, such as caching, firewalling and content adaption. A proxy which performs caching will often be called a caching proxy.

- Proxy servers can dramatically improve performance for groups of users. This is because it saves the results of all requests for a certain amount of time.

Project Objective

The objective of the project is to build a Web Proxy Server that can block content, IP Addresses and websites. It should also be able to cache requested object and also able to replace the object from cache.

Features

1. Client program **will take as input the URL** being requested by the client and **generate a HTTP request** using the typed in URL.

2. Proxy server program that has two functions
   -- First function is to accept HTTP requests from client programs using a TCP socket. The proxy server will first check if the received HTTP request is properly formed and then do further processing. If the HTTP request is poorly formed then the proxy server will retur an **error message** back to the client.
   -- Second function is to process the HTTP request. The proxy server opens a TCP connection to the corresponding host mentioned in the client HTTP request and downloads the requested URL object. The server will maintain a local copy of the object which includes the last modified time of the object. The server then transmits this data back to the client program using the TCP socket. All further requests to the same URL are processed by the server itself, until the client issues a conditional-GET request.

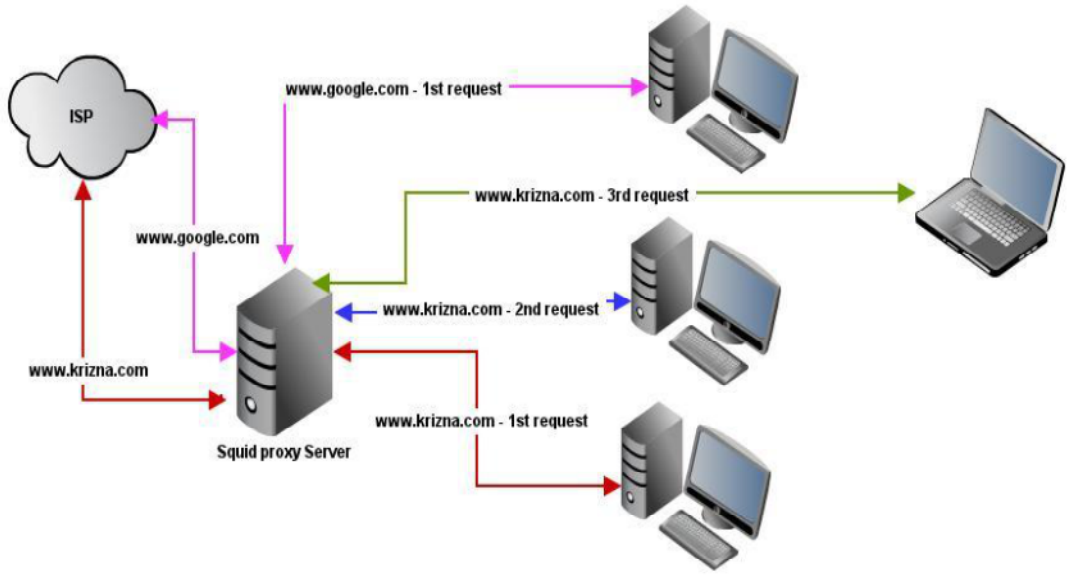
3. For a single HTTP object request the client can issue a normal request to the proxy. For multiple HTTP objects, each request must be sent in a **different thread** and processed accordingly.
4. There can be **more than one client (from the same host machine)** that can issue HTTP requests (single or multiple object requests).
5. The cache will be limited to 100 HTTP requests. If more requests are made then the cache needs to pick one object from the **cache for replacement**.
   Policy used for object replacement is : **LEAST RECENTLY USED**

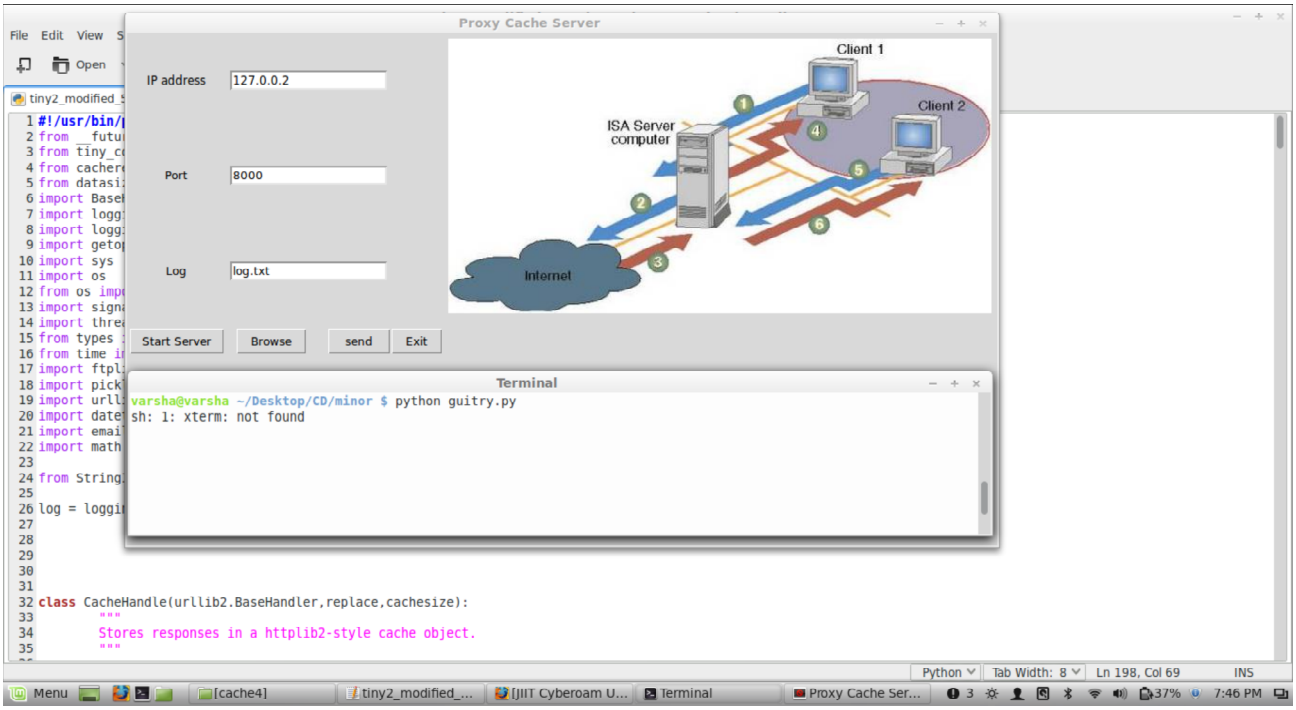
Algorithm Used

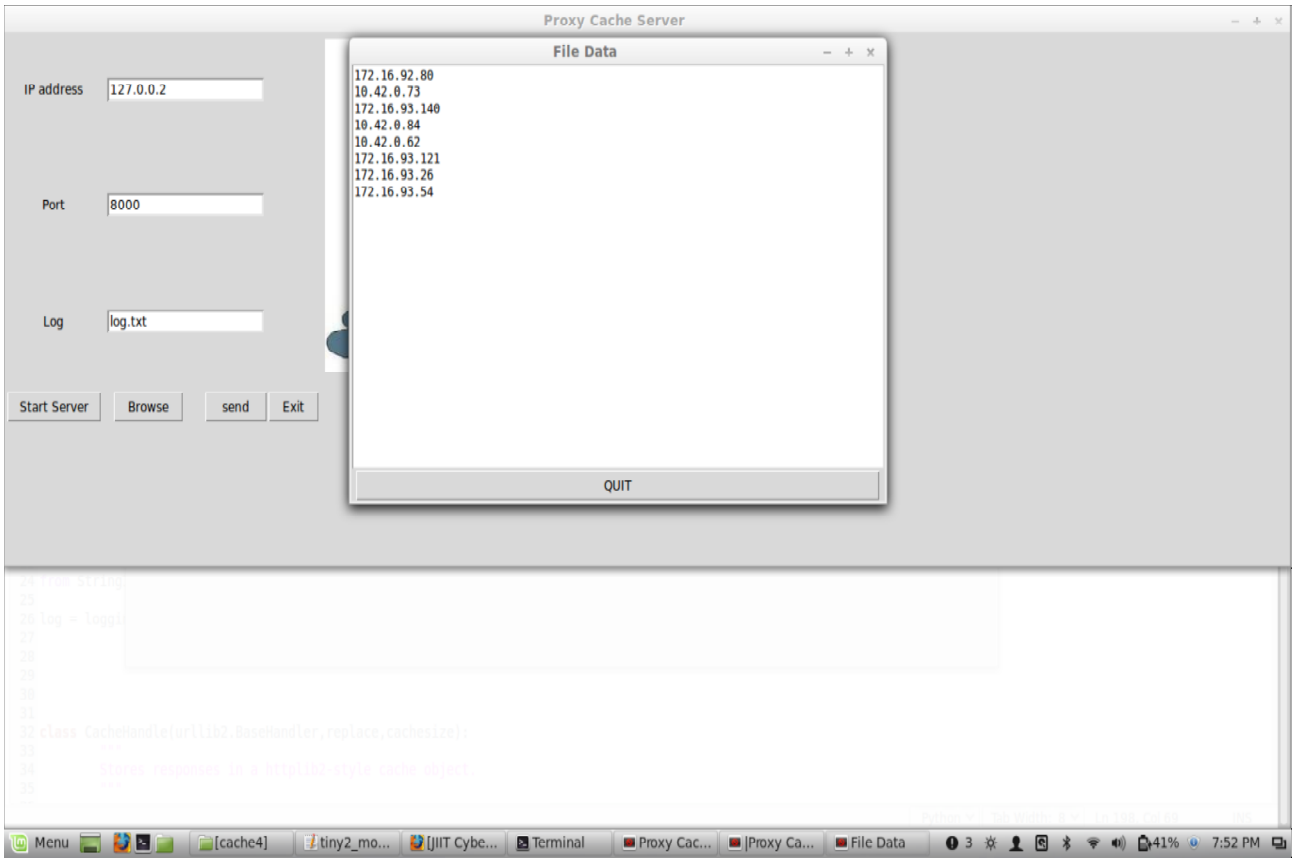
Least Recently Used:

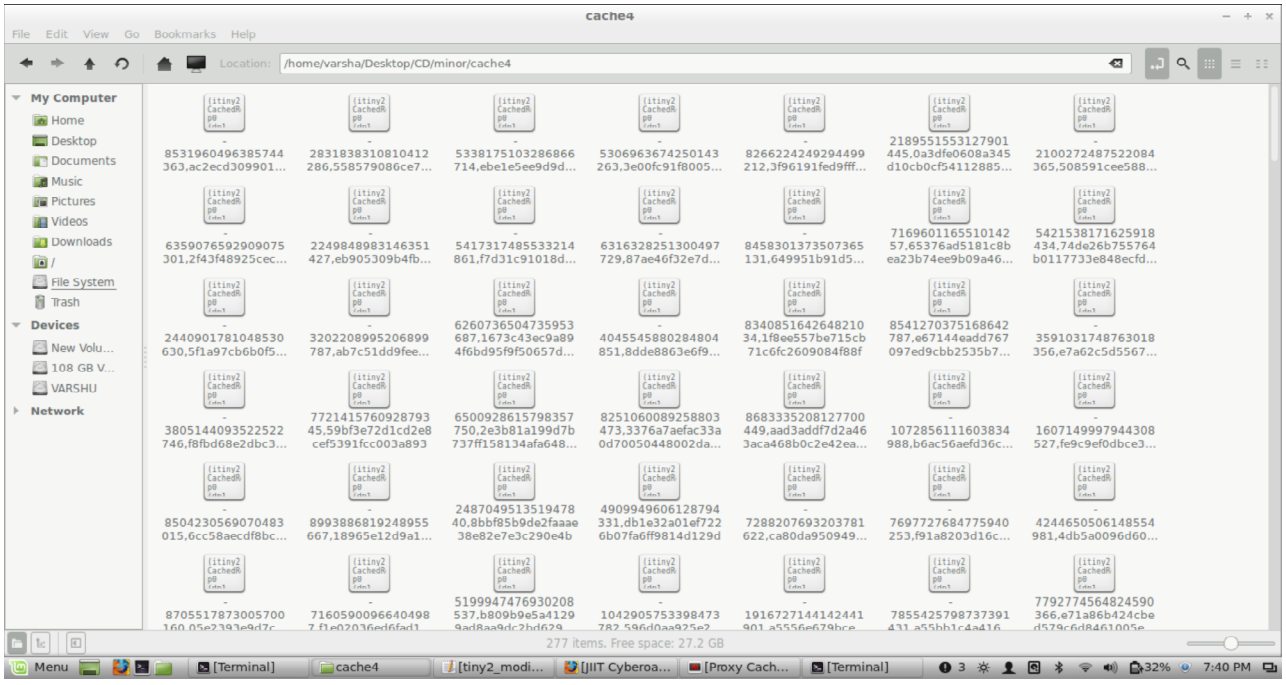
Discards the least recently used items first. This algorithm requires keeping track of what was used when, which is expensive if one wants to make sure the algorithm always discards the least recently used item. General implementations of this technique require keeping "age bits" for cache-lines and track the "Least Recently Used" cache-line based on age-bits. In such an implementation, every time a cache-line is used, the age of all other cache-lines changes. LRU is actually a family of caching algorithms with members including: 2Q by Theodore Johnson and Dennis Shasha and LRU/K by Pat O'Neil, Betty O'Neil and Gerhard Weikum.

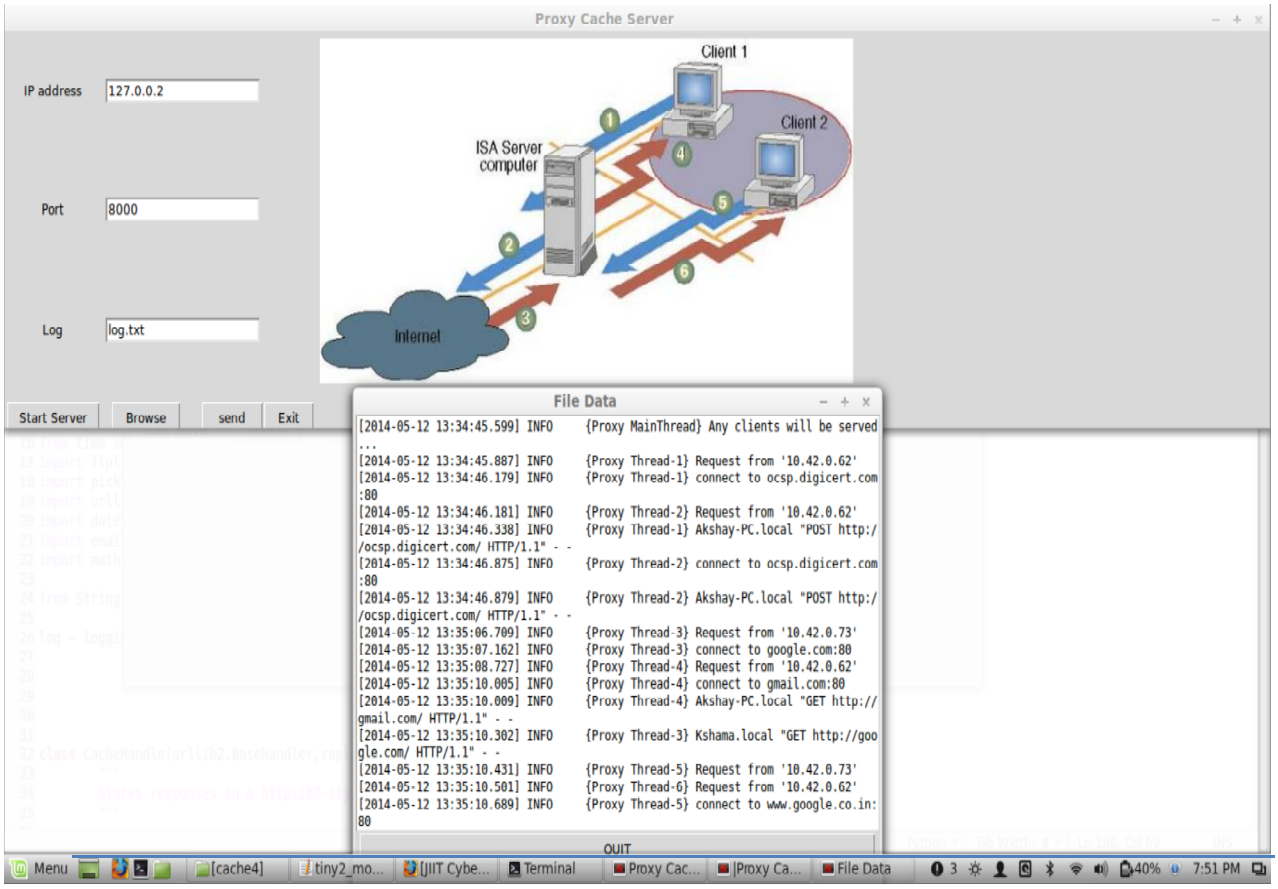
IMPLEMENTATION

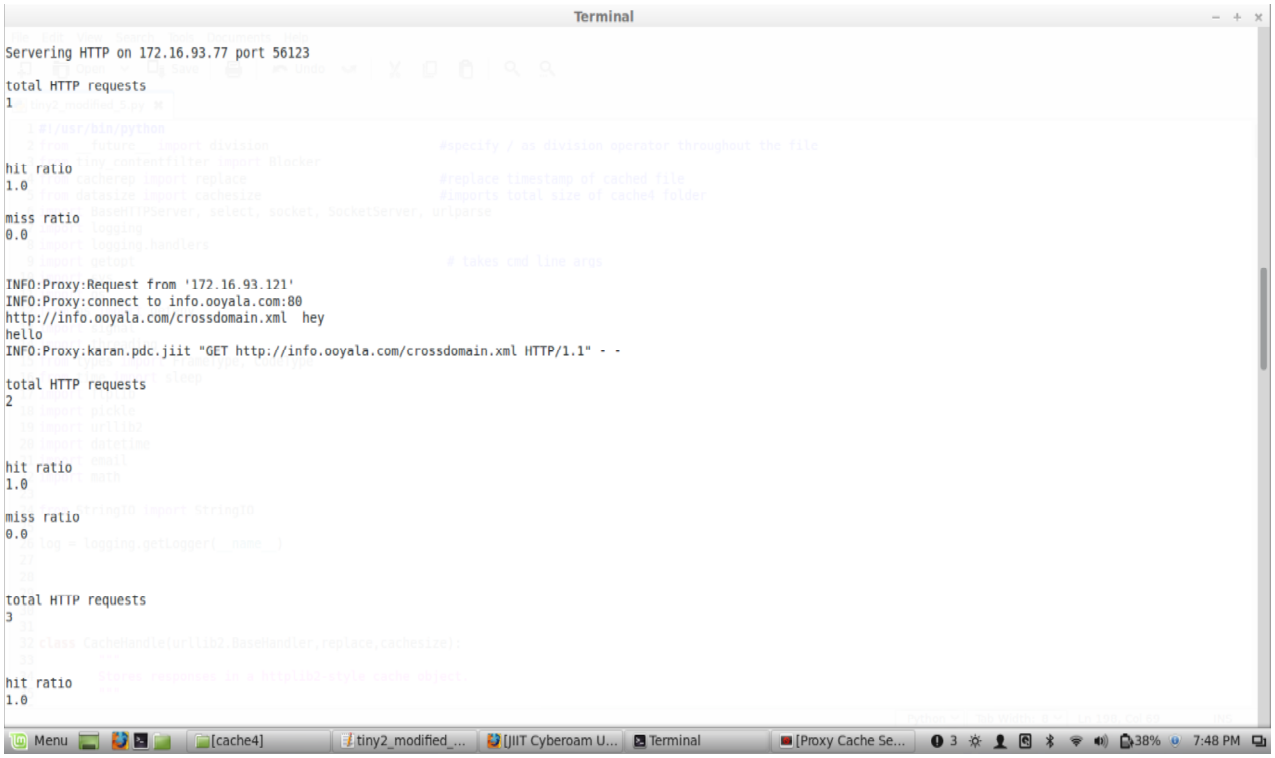


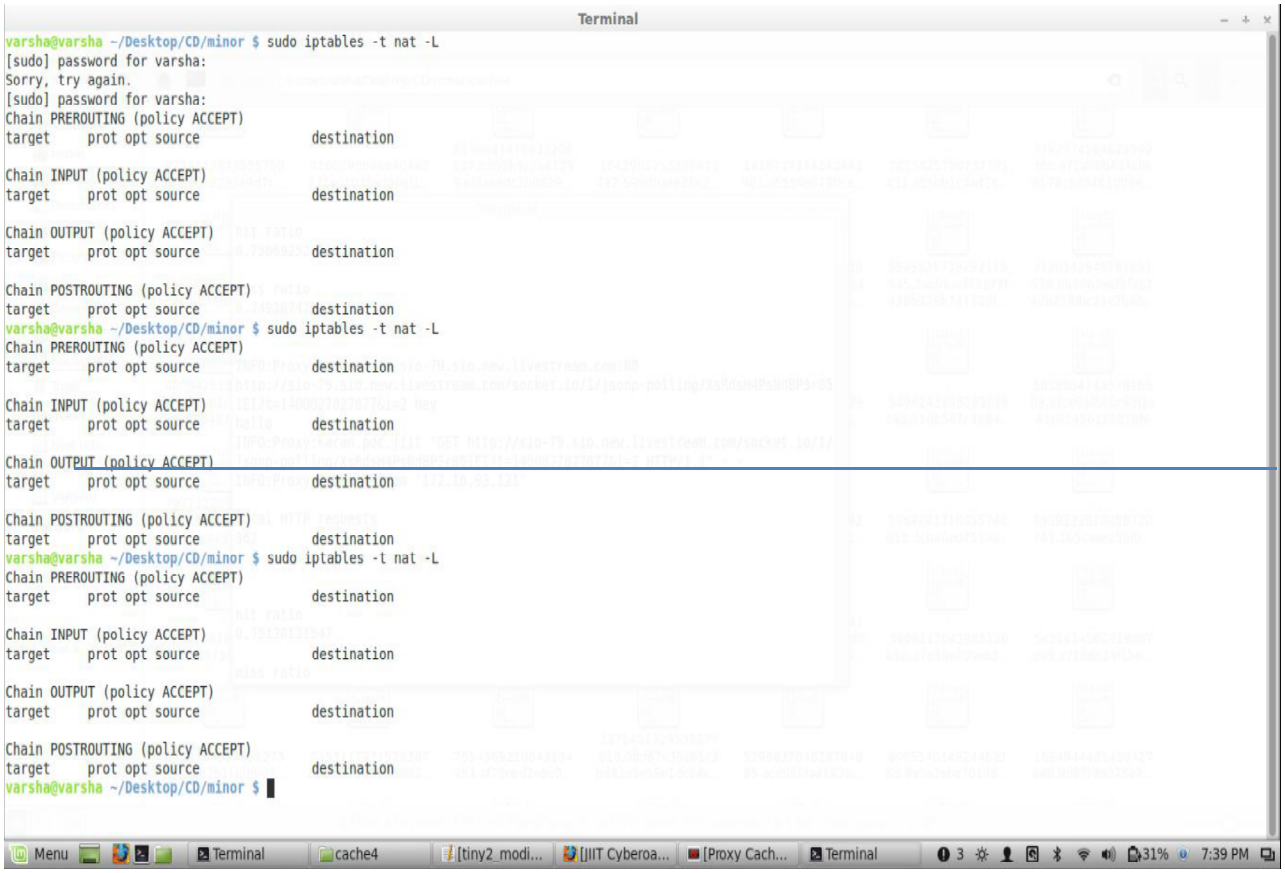

Proxy Cache Server
IP address
127.0.0.2
Port
8000
Log
log.txt
Start Server
Browse
send
Exit
Client 1
Client 2
ISA Server computer
Internet
Terminal
varsha@varsha ~/Desktop/CD/minor $ python guitry.py
sh: 1: xterm: not found
class CacheHandle(urllib2.BaseHandler,replace,cachesize):
Stores responses in a httplib2-style cache object.


Proxy Cache Server
IP address
127.0.0.2
Port
8000
Log
log.txt
Start Server
Browse
send
Exit
File Data
172.16.92.80
10.42.0.73
172.16.93.140
10.42.0.84
10.42.0.62
172.16.93.121
172.16.93.26
172.16.93.54
QUIT


cache4
File Edit View Go Bookmarks Help
Location: /home/varsha/Desktop/CD/minor/cache4
My Computer
Home
Desktop
Documents
Music
Pictures
Videos
Downloads
File System
Trash
Devices
Network
277 items. Free space: 27.2 GB

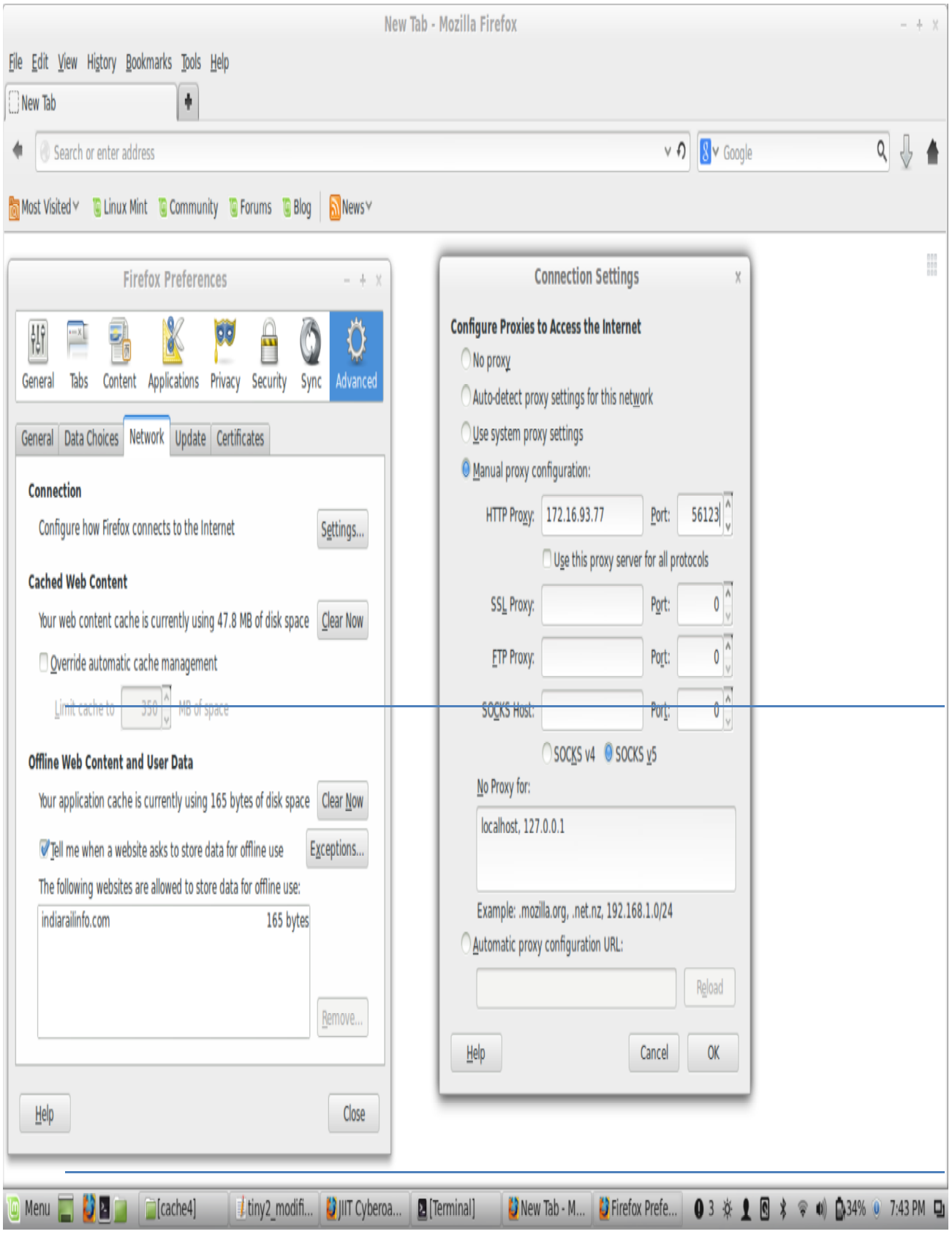

Proxy Cache Server
IP address
127.0.0.2
Port
8000
Log
log.txt
Start Server
Browse
send
Exit
Client 1
Client 2
ISA Server computer
Internet
File Data
[2014-05-12 13:34:45.599] INFO {Proxy MainThread} Any clients will be served
[2014-05-12 13:34:45.887] INFO {Proxy Thread-1} Request from '10.42.0.62'
[2014-05-12 13:34:46.179] INFO {Proxy Thread-1} connect to ocsp.digicert.com :80
[2014-05-12 13:34:46.181] INFO {Proxy Thread-2} Request from '10.42.0.62'
[2014-05-12 13:34:46.338] INFO {Proxy Thread-1} Akshay-PC.local "POST http:/ /ocsp.digicert.com/ HTTP/1.1" - -
[2014-05-12 13:34:46.875] INFO {Proxy Thread-2} connect to ocsp.digicert.com :80
[2014-05-12 13:34:46.879] INFO {Proxy Thread-2} Akshay-PC.local "POST http:/ /ocsp.digicert.com/ HTTP/1.1" - -
[2014-05-12 13:35:06.709] INFO {Proxy Thread-3} Request from '10.42.0.73'
[2014-05-12 13:35:07.162] INFO {Proxy Thread-3} connect to google.com:80
[2014-05-12 13:35:08.727] INFO {Proxy Thread-4} Request from '10.42.0.62'
[2014-05-12 13:35:10.005] INFO {Proxy Thread-4} connect to gmail.com:80
[2014-05-12 13:35:10.009] INFO {Proxy Thread-4} Akshay-PC.local "GET http:// gmail.com/ HTTP/1.1" - -
[2014-05-12 13:35:10.302] INFO {Proxy Thread-3} Kshama.local "GET http://goo gle.com/ HTTP/1.1" - -
[2014-05-12 13:35:10.431] INFO {Proxy Thread-5} Request from '10.42.0.73'
[2014-05-12 13:35:10.501] INFO {Proxy Thread-6} Request from '10.42.0.62'
[2014-05-12 13:35:10.689] INFO {Proxy Thread-5} connect to www.google.co.in: 80
QUIT


Terminal
Servering HTTP on 172.16.93.77 port 56123
total HTTP requests
1
hit ratio
1.0
miss ratio
0.0
INFO:Proxy:Request from '172.16.93.121'
INFO:Proxy:connect to info.ooyala.com:80
http://info.ooyala.com/crossdomain.xml hey
hello
INFO:Proxy:karan.pdc.jiit "GET http://info.ooyala.com/crossdomain.xml HTTP/1.1" - -
total HTTP requests
2
hit ratio
1.0
miss ratio
0.0
total HTTP requests
3
hit ratio
1.0
Menu
[cache4]
tiny2_modified_...
[JIIT Cyberoam U...
Terminal
[Proxy Cache Se...
38%
7:48 PM


Terminal
varsha@varsha ~/Desktop/CD/minor $ sudo iptables -t nat -L
[sudo] password for varsha:
Sorry, try again.
[sudo] password for varsha:
Chain PREROUTING (policy ACCEPT)
target prot opt source destination
Chain INPUT (policy ACCEPT)
target prot opt source destination
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
Chain POSTROUTING (policy ACCEPT)
target prot opt source destination
varsha@varsha ~/Desktop/CD/minor $ sudo iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target prot opt source destination
Chain INPUT (policy ACCEPT)
target prot opt source destination
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
Chain POSTROUTING (policy ACCEPT)
target prot opt source destination
varsha@varsha ~/Desktop/CD/minor $ sudo iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target prot opt source destination
Chain INPUT (policy ACCEPT)
target prot opt source destination
Chain OUTPUT (policy ACCEPT)
target prot opt source destination
Chain POSTROUTING (policy ACCEPT)
target prot opt source destination
varsha@varsha ~/Desktop/CD/minor $
Menu
Terminal
cache4
[tiny2_modi...
[JIIT Cyberoa...
[Proxy Cach...
Terminal
31%
7:39 PM

Terminal

```
Servering HTTP on 172.16.93.77 port 56123

total HTTP requests
1

hit ratio
1.0

miss ratio
0.0

INFO:Proxy:Request from '172.16.93.121'
INFO:Proxy:connect to info.ooyala.com:80
http://info.ooyala.com/crossdomain.xml  hey
hello
INFO:Proxy:karan.pdc.jiit "GET http://info.ooyala.com/crossdomain.xml HTTP/1.1" - -

total HTTP requests
2

hit ratio
1.0

miss ratio
0.0


total HTTP requests
3


hit ratio
1.0
```


New Tab - Mozilla Firefox
File Edit View History Bookmarks Tools Help
New Tab
Search or enter address
Google
Most Visited
Linux Mint
Community
Forums
Blog
News
Firefox Preferences
General
Tabs
Content
Applications
Privacy
Security
Sync
Advanced
General
Data Choices
Network
Update
Certificates
Connection
Configure how Firefox connects to the Internet
Settings...
Cached Web Content
Your web content cache is currently using 47.8 MB of disk space
Clear Now
Override automatic cache management
Limit cache to 350 MB of space
Offline Web Content and User Data
Your application cache is currently using 165 bytes of disk space
Clear Now
Tell me when a website asks to store data for offline use
Exceptions...
The following websites are allowed to store data for offline use:
indiarailinfo.com 165 bytes
Remove...
Help
Close
Connection Settings
Configure Proxies to Access the Internet
No proxy
Auto-detect proxy settings for this network
Use system proxy settings
Manual proxy configuration:
HTTP Proxy: 172.16.93.77 Port: 56123
Use this proxy server for all protocols
SSL Proxy: Port: 0
FTP Proxy: Port: 0
SOCKS Host: Port: 0
SOCKS v4 SOCKS v5
No Proxy for:
localhost, 127.0.0.1
Example: .mozilla.org, .net.nz, 192.168.1.0/24
Automatic proxy configuration URL:
Reload
Help
Cancel
OK
Menu
[cache4]
tiny2_modifi...
JIIT Cyberoa...
[Terminal]
New Tab - M...
Firefox Prefe...
34%
7:43 PM

Terminal

```
hit ratio
0.648648648649

miss ratio
0.351351351351
```

```
INFO:Proxy:connect to new.livestream.com:80
http://new.livestream.com/api/servers/sio/leastloaded.json?mode=full    hey
hello
INFO:Proxy:karan.pdc.jiit "GET http://new.livestream.com/api/servers/sio/leastloaded.json?mode=full HTTP/1.1" - -
INFO:Proxy:connect to www.youtube.com:80
http://www.youtube.com/ hey
found

BLOCKED A CONNECTION TO AN UNAUTHORISED DOMAIN:

INFO:Proxy:Request from '172.16.93.121'

total HTTP requests
38
DEBUG:tiny2_modified_5:Request is stale


hit ratio
0.631578947368

miss ratio
0.368421052632


INFO:Proxy:connect to www.youtube.com:80
http://www.youtube.com/ hey
found

BLOCKED A CONNECTION TO AN UNAUTHORISED DOMAIN:
```

```
.....
```

TEST CASES:

Test ID	Description	Output	Expected Output
1	Sudo ProxyCache abcd	Setting up socket Please give port Number as Integer	Executed the command on one of the client Setting up Socket
2	Cache size should be atleast 100	BufferSize full .Server stop working.	Cache size 100 ,takes more URL entry in Cache.
3	When heavy site like youtube.com	Broken pipe Exception	Website should run and show that URL present in Cache or not.
4	Exited The Server	All the clients should get closed	All Clients get closed with an Exception in java

Py.test tool

Category:Unit testing

Part of py.lib

```
Terminal
varsha@varsha ~/Desktop/CD/minor $ py.test
============================= test session starts ==============================
platform linux2 -- Python 2.7.5 -- pytest-2.3.5
collected 0 items

============================== in 0.07 seconds ===============================
varsha@varsha ~/Desktop/CD/minor $ tiny3.py.test
tiny3.py.test: command not found
varsha@varsha ~/Desktop/CD/minor $ python tiny3.py.test
python: can't open file 'tiny3.py.test': [Errno 2] No such file or directory
varsha@varsha ~/Desktop/CD/minor $ py.test
============================= test session starts ==============================
platform linux2 -- Python 2.7.5 -- pytest-2.3.5
collected 0 items

============================== in 0.07 seconds ===============================
varsha@varsha ~/Desktop/CD/minor $
```

Result:No error

Pylint tool

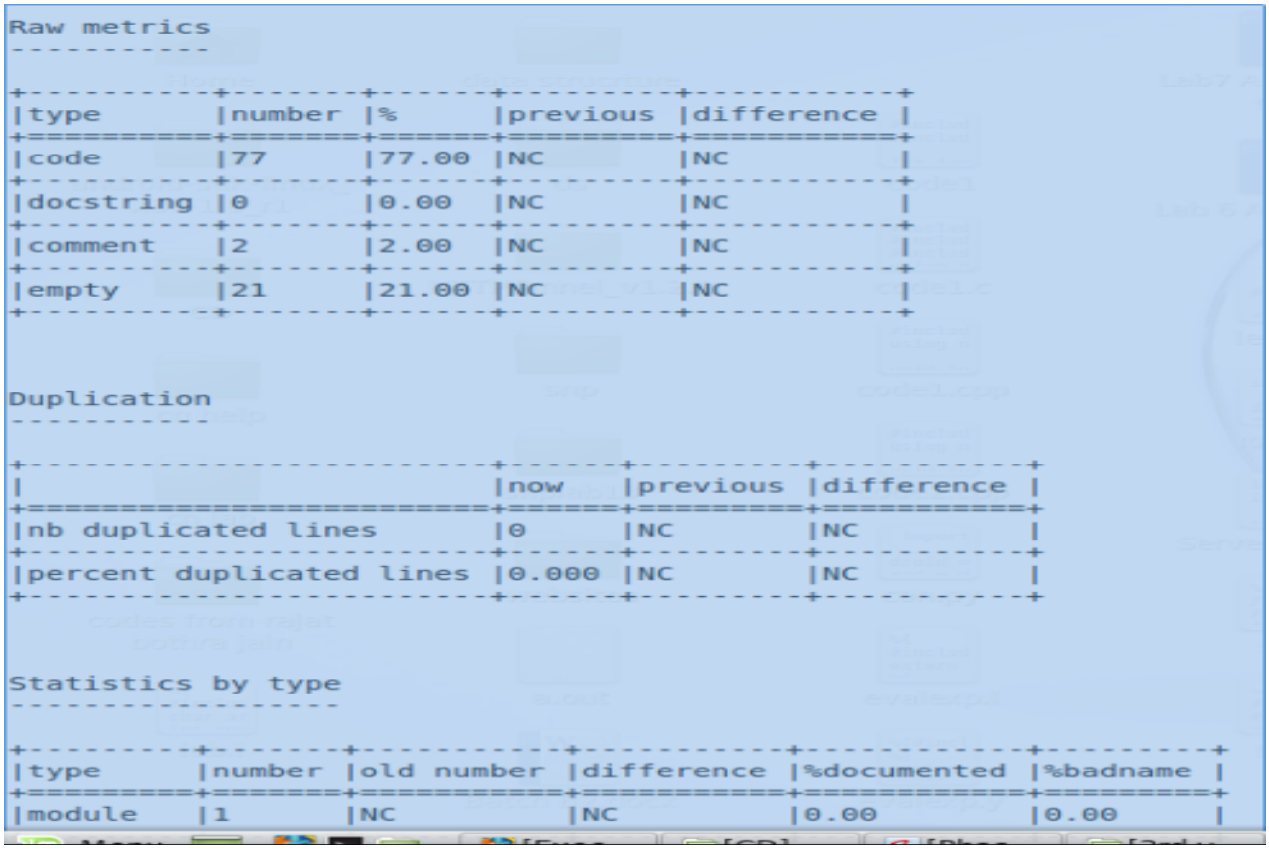
```
varsha@varsha ~/Desktop/CD/minor $ pylint guitry.py
No config file found, using default configuration
************* Module guitry
W: 10,0: Found indentation with tabs instead of spaces
W: 16,0: Found indentation with tabs instead of spaces
W: 25,0: Found indentation with tabs instead of spaces
W: 26,0: Found indentation with tabs instead of spaces
W: 28,0: Found indentation with tabs instead of spaces
W: 29,0: Found indentation with tabs instead of spaces
W: 31,0: Found indentation with tabs instead of spaces
W: 32,0: Found indentation with tabs instead of spaces
W: 34,0: Found indentation with tabs instead of spaces
W: 35,0: Found indentation with tabs instead of spaces
W: 37,0: Found indentation with tabs instead of spaces
W: 38,0: Found indentation with tabs instead of spaces
W: 48,0: Found indentation with tabs instead of spaces
W: 49,0: Found indentation with tabs instead of spaces
W: 51,0: Found indentation with tabs instead of spaces
W: 52,0: Found indentation with tabs instead of spaces
W: 54,0: Found indentation with tabs instead of spaces
W: 55,0: Found indentation with tabs instead of spaces
W: 56,0: Found indentation with tabs instead of spaces
W: 58,0: Found indentation with tabs instead of spaces
W: 59,0: Found indentation with tabs instead of spaces
W: 60,0: Found indentation with tabs instead of spaces
W: 62,0: Found indentation with tabs instead of spaces
W: 63,0: Found indentation with tabs instead of spaces
W: 64,0: Found indentation with tabs instead of spaces
W: 67,0: Found indentation with tabs instead of spaces
W: 68,0: Found indentation with tabs instead of spaces
W: 69,0: Found indentation with tabs instead of spaces
W: 70,0: Found indentation with tabs instead of spaces
W: 71,0: Found indentation with tabs instead of spaces
W: 73,0: Found indentation with tabs instead of spaces
W: 74,0: Found indentation with tabs instead of spaces
W: 75,0: Found indentation with tabs instead of spaces
W: 76,0: Found indentation with tabs instead of spaces
W: 77,0: Found indentation with tabs instead of spaces
W: 79,0: Found indentation with tabs instead of spaces
W: 80,0: Found indentation with tabs instead of spaces
W: 81,0: Found indentation with tabs instead of spaces
```

```
C: 21,0:fbrowse: Missing docstring
C: 22,4:fbrowse.App: Missing docstring
C: 25,9:fbrowse.App.__init__.bclose: Missing docstring
R: 22,4:fbrowse.App: Too few public methods (0/2)
W: 42,4:fbrowse: Unused variable 'app'
W: 48,35:send: Redefining built-in 'file'
C: 45,0:send: Missing docstring
E: 62,7:send: Instance of '_socketobject' has no 'sendall' member
E: 63,7:send: Instance of '_socketobject' has no 'shutdown' member
C: 73,1: Operator not preceded by a space
      L2= Label(root, text="Port")
   ^
C: 79,1: Operator not preceded by a space
      L3= Label(root, text="Log")
   ^
C: 80,1: Comma not followed by a space
      L3.grid(row=2,column=0, pady=2, padx=6)
            ^^
C: 96,1: Invalid name "path" for type constant (should match (([A-Z_][A-Z0-9_]*)
|(__.*__))$)
C: 97,1: Invalid name "img" for type constant (should match (([A-Z_][A-Z0-9_]*)|
(__.*__))$)
C: 98,1: Invalid name "panel" for type constant (should match (([A-Z_][A-Z0-9_]*
)|(__.*__))$)
C: 99,1: Comma not followed by a space
      panel.grid(row=0, column=3, columnspan=2, rowspan=3,sticky=W+E+N+S, padx
=5, pady=5)
                                                ^^
C:102,1: Invalid name "termf" for type constant (should match (([A-Z_][A-Z0-9_]*
)|(__.*__))$)
C:104,1: Invalid name "wid" for type constant (should match (([A-Z_][A-Z0-9_]*)|
(__.*__))$)
W:  2,0: Unused import Canvas from wildcard import
W:  2,0: Unused import MULTIPLE from wildcard import
```

Terminal

```
Report
======
77 statements analysed.

Messages by category
--------------------

+-----------+-------+---------+-----------+
|type       |number |previous |difference |
+===========+=======+=========+===========+
|convention |19     |NC       |NC         |
+-----------+-------+---------+-----------+
|refactor   |1      |NC       |NC         |
+-----------+-------+---------+-----------+
|warning    |229    |NC       |NC         |
+-----------+-------+---------+-----------+
|error      |2      |NC       |NC         |
+-----------+-------+---------+-----------+


Messages
--------

+-----------+-------------+
|message id |occurrences  |
+===========+=============+
|W0614      |168          |
+-----------+-------------+
|W0312      |58           |
+-----------+-------------+
|C0111      |7            |
+-----------+-------------+
|C0103      |6            |
+-----------+-------------+
|C0324      |3            |
+-----------+-------------+
|E1101      |2            |
+-----------+-------------+
|C0322      |2            |
+-----------+-------------+
```

Menu [Exec... [CD] [Phas... [3rd y... [Netw... [EN



```
Raw metrics
-----------

+----------+-------+------+----------+------------+
|type      |number |%     |previous  |difference  |
+==========+=======+======+==========+============+
|code      |77     |77.00 |NC        |NC          |
+----------+-------+------+----------+------------+
|docstring |0      |0.00  |NC        |NC          |
+----------+-------+------+----------+------------+
|comment   |2      |2.00  |NC        |NC          |
+----------+-------+------+----------+------------+
|empty     |21     |21.00 |NC        |NC          |
+----------+-------+------+----------+------------+


Duplication
-----------

+--------------------------+------+----------+------------+
|                          |now   |previous  |difference  |
+==========================+======+==========+============+
|nb duplicated lines       |0     |NC        |NC          |
+--------------------------+------+----------+------------+
|percent duplicated lines  |0.000 |NC        |NC          |
+--------------------------+------+----------+------------+


Statistics by type
------------------

+---------+-------+-----------+-----------+------------+---------+
|type     |number |old number |difference |%documented |%badname |
+=========+=======+===========+===========+============+=========+
|module   |1      |NC         |NC         |0.00        |0.00     |
```

Risk Analysis

Risk Id	Description	Risk Area	Problem(p)	Impact(I)	RE(P*I)
1	User Enters Non valid command	System Risk	L(1)	L(1)	1
2	Power Failure	Requirement Risk	M(3)	M(3)	9
3	Server Fails	System Risk	H(5)	H(5)	25
4	Executing Gets Struck	System Risk	M(3)	L(2)	6
5	Executing Program asks for user input or system level libraries calls not present	System Risk	M(3)	H(4)	12
6	Firewall Port Blocked	RequiremenT Risk	H(5)	H(5)	25

Future work :

Conclusion

Security
System is not very secure, it is easy to break in between server and client to initiate attacks, reduce throughput, sniff packets,

Compatibility for OS
Proxy server is optimized to run on Linux OS but in future it will be modified to improve its compatibility on other OS such as Windows or MAC

Portability
System is not platform independent. It can be made independent of platform in future by porting the used modules to other platforms or by changing the programming language used. This will require high skills and effort.

Load Handling capacity of system
As for now system has been tested to work with atleast 10 clients and handle over 150 threads but this is not enough to for system to work in real time environment. In future we can increase the load handling capacity of server and implement it in real time environment.

Client Package
In this implementation we have used clients which are independent of this proxy server. In future we can make a Client that will be highly optimized to work with this Proxy Server.
Caching may partially be implemented in client itself and it can be hard coded to work with Proxy Server to improve performance and user experience.

Extensibility of protocols
This proxy server works only for the HTTP requests, it can be extended to handle HTTPS, FTP, POP3 (email client) and many other protocols.

Handling POST request types
This system works only for GET request but it can be modified to make it work for POST requests

References

http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4498173

http://www.ffi.no/no/Rapporter/13-02926.pdf

http://www.google.com/patents/US20090077254

www.w3.org/Protocols/**rfc**2616/**rfc**2616.html

http://objectclub.esm.co.jp/Jude/

http://argouml.tigris.org/

http://robertbell.articlealley.com/the-advantages-of-using-proxy-servers-841461.html

http://www.slideshare.net/greatbury/advantages-of-proxy-server

http://sharannetwork.blogspot.in/2011/01/proxy-servers-basics-i.html

http://www.articlesbase.com/internet-articles/proxy-server-basics-375667.html