

MANAGEMENT FRONT END FOR CACHE SERVER

CHAN YUNN CHYUAN

A thesis submitted in partially fulfillment of the requirements for the award of degree of
Bachelor of Computer Science (Computer Systems & Networking)

Faculty of Computer Systems & Software Engineering
Universiti Malaysia Pahang

APRIL 2010

ABSTRACT

Caching and proxy technology designed to solve the internet issues in organization include security and performance by cache and filter the request between network with another network mostly internal LAN and global network. Squid cache server is one of the most famous cache server that running on multiple platform. Squid cache server use *squid.conf* file to store configuration and setting for the cache server. Whenever administrators want to configure the cache server, they need to interact with the file itself, which require administrator to study and memories a lot of the commands. Administrators also need extra knowledge on how to set up the server to run at platform. SQ Squid cache server management front end is design to overcome this inconvenient by providing graphical user interface design to interact with the *squid.conf* file on behalf of the administrators. All options available for Squid cache server is configure using GUI make it easier to understand and configure. However, SQ Squid cache server management front end is not a new cache server; it's only a graphical user interface management front end for existing cache server-Squid cache server. SQ Squid Management front end help administrators understand and configure the Squid cache server where extra knowledge on command use in Squid cache server is not needed but only knowledge on cache, network, and security.

ABSTRAK

Teknologi caching dan proxy direka untuk menyelesaikan isu-isu internet dalam organisasi merangkumi sekuriti dan performance dengan cache dan menuras permintaan antara satu rangkaian dengan rangkaian yang lain terutamanya internal LAN dengan internet. Squid cache server adalah salah satu cache server yang amat popular yang boleh dikendali dengan berbilang platform. Squid cache server menggunakan fail *squid.conf* untuk menyimpan konfigurasi dan setting untuk cache server. Bila administrator hendak configure cache server, mereka perlu berinteraksi dengan fail itu, dimana memerlukan administrator untuk belajar dan mengingati banyak command. Administrator juga memerlukan pengetahuan tambahan atas cara untuk menyedia server untuk dikendali atas platform. SQ Squid cache server management front end direka untuk mengatasi penyukaran dengan menyediakan graphical user interface reka bentuk untuk berinteraksi dengan fail *squid.conf* file bagi pihak administrator. Semua pilihan, fungsi yang sedia ada dalam Squid cache server boleh configure dengan GUI menyebabkan lebih senang untuk memahami dan configure. Walaubagaimanapun, SQ Squid cache server management front end bukannya cache server yang baru; ia hanyalah graphical user interface front end pengurusan untuk cache server yang sedia ada-Squid cache server. SQ Squid Management front end membantu administrator dalam memahami dan untuk configure Squid cache server di mana pengetahuan tambahan untuk mengguna command dalam Squid cache server tidak diperlukan tetapi hanya pengetahuan pada cache, rangkaian, dan sekuriti.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	TITLE PAGE	i
	DECLARATION	ii
	ACKNOWLEDGEMENTS	iii
	ABSTRACT	iv
	ABSTRAK	v
	TABLE OF CONTENTS	vi
	LIST OF TABLES	x
	LIST OF FIGURES	xi
	LIST OF ABBREVIATIONS	xiv
	LIST OF APPENDICES	xv
1	INTRODUCTION	1
	1.1 Introduction	1
	1.2 Problem Statement	2
	1.3 Objective	2
	1.4 Scope	3
2	LITERATURE REVIEW	4
	2.1 Internet	4
	2.2 Proxy Technologies	6

CHAPTER	TITLE	PAGE
	2.3 Caching	7
	2.4 Squid Cache Server	7
	2.4.1 <i>Squid.conf</i> file	8
	2.5 Technologies	8
	2.5.1 Graphical User Interface	8
	2.5.2 Microsoft .NET Framework	9
	2.5.3 Programming Language – Microsoft Visual C#	10
3	METHODOLOGY	11
	3.1 Project Model	11
	3.2 Project Method	12
	3.2.1 Identification Phase	13
	3.2.2 Planning Phase	14
	3.2.3 Analysis	14
	3.2.4 Design	15
	3.2.5 Implementation and Testing	15
	3.2.6 Maintenance	16
	3.3 System Design	16
	3.3.1 System flowchart	16
	3.3.2 UML Diagram	17
	3.3.2.1 Data-Flow-Diagram (DFD)	18
	3.3.2.2 Use Case Diagram	20
	3.3.2.3 Sequence Diagram	20
	3.3.2.4 Class Diagram	23
	3.3.2.5 Activity Diagram	24
	3.4 Hardware and Software	28
4	IMPLEMENTATION	29
	4.1 Icon and Language Selection	29

CHAPTER	TITLE	PAGE
	4.1.1 Icon	29
	4.1.2 Language	30
	4.1.3 Name	30
4.2	Storyboard and Mechanism	31
	4.2.1 Start Application mechanism	31
	4.2.2 Configuration Wizard Operation Process 1	33
	4.2.3 Configuration Wizard Operation Process 2	35
	4.2.4 Configuration Wizard Operation Process 3	37
	4.2.5 Saving Configuration Mechanism	38
4.3	Interface and Database	39
	4.3.1 Start form	39
	4.3.2 Summary form	40
	4.3.3 Configuration Wizard	42
	4.3.4 Network setting form	45
	4.3.5 Cache setting form	47
	4.3.6 File setting form	48
	4.3.7 General setting form	49
	4.3.8 Parent Proxy setting form	50
	4.3.9 External Plug-in form	52
	4.3.10 Time-out setting form	52
	4.3.11 Location setting form	53
	4.3.12 Users or Groups setting form	55
	4.3.13 Schedule setting form	58
	4.3.14 Protocol setting form	59
	4.3.15 Port setting form	61
	4.3.16 Access Control Policies form	62
	4.3.17 History form	65
	4.3.18 Database	66
4.4	Testing	67

CHAPTER	TITLE	PAGE
5	RESULT AND DISCUSSION	70
	5.1 Introduction	70
	5.2 Result	71
	5.3 Discussion	72
	5.4 Assumption	74
	5.5 Future Work	74
6	CONCLUSION	75
	REFERENCES	76
	APPENDIXEX	78

LIST OF TABLES

TABLE NO.	TITLE	PAGE
2.1	Compare GUI and Command prone systems	9
3.1	Modules for SQ Management Front End	27
4.1	Result Testing	68

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE
2.1	Internet	5
2.2	How Proxies Work	6
3.1	Project Model	12
3.2	Project Method	13
3.3	System Flowchart	17
3.4	Context Diagram	18
3.5	DFD Level 0	19
3.6	Use Case Diagram	20
3.7	Sequence Diagram for Display and Configuring system	21
3.8	Sequence Diagram for Start Service	22
3.9	Sequence Diagram for Stop Service	22
3.10	Class Diagram	24
3.11	Activity Diagram for Configure the Service	25
3.12	Interface Design for the SQ Squid Management front End	26
4.1	SQ Squid Cache Server Management front end	29
4.2	Starting Application	32
4.3	Storyboard for Configuration Wizard 1	34

FIGURE NO.	TITLE	PAGE
4.4	Storyboard for Configuration Wizard 2	36
4.5	Storyboard for Configuration Wizard 3	37
4.6	Save Mechanism	38
4.7	Start form	40
4.8	LoadConfiguration() method	40
4.9	Summary form	41
4.10	Summary list box	42
4.11	Configuration Wizard 1	43
4.12	Configuration Wizard 2	44
4.13	Configuration Wizard 3	44
4.14	SaveConfiguration() method	45
4.15	Network setting form	46
4.16	“Save” button click event handler	47
4.17	Cache setting form	47
4.18	File setting form	48
4.19	General setting form	49
4.20	Parent Proxy setting form	50
4.21	sub-Parent Proxy setting form	51
4.22	Double click on DataGrid	51
4.23	External Plug-in setting form	52
4.24	Time-out setting form	53
4.25	Location setting form	54
4.26	sub Location setting form	55
4.27	Users setting form	56
4.28	sub Users setting form	56
4.29	Groups setting form	57
4.30	sub Groups setting form	57
4.31	Schedule setting form	58
4.32	sub Schedule setting form	59
4.33	Protocol setting form	60

FIGURE NO.	TITLE	PAGE
4.34	sub Protocol setting form	60
4.35	Port setting form	61
4.36	sub Port setting form	62
4.37	Define Access Rule	63
4.38	Select user or group	63
4.39	Select location	64
4.40	Select schedule	64
4.41	FillList() method	65
4.42	Restore method	65
4.43	History form	66
4.44	ConfSetting.xsd	66
4.45	<i>squid.conf</i> file statement	68
4.46	Denied Message	69
4.47	start configure	83
4.48	Configure Network Setting	84
4.49	Enter parameter and save	85
4.50	Apply configuration	86
4.51	Configure ACL	87
4.52	Click Location	87
4.53	Add new ACL	88
4.54	Save ACL	89
4.55	Start configure ACP	90
4.56	Add new ACP	90
4.57	select deny access	91
4.58	Select ACL categories 1	92
4.59	Select ACL categories 2	92
4.60	Finish ACP	93
4.61	Apply ACL and ACP	93
4.62	Restoration	94
5.1	Testing Topology	71

LIST OF ABBREVIATIONS

ACL	-	Access Control List
ACP	-	Access Control Policies
IP	-	Internet Protocol
MAC	-	Media Access Control

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	<i>squid.conf</i> file	89
B	Project Gantt chart	91
C	User Manual	93

CHAPTER 1

INTRODUCTION

1.1 Introduction

Advancement in Internet technologies and the low price of internet access make it easier to access to internet, hence increase the number of internet users. When the number of internet users increase, the performance of internet has dropped significantly. Most of the big organization in order to increase the speed of internet, they set up a cache or proxy server. Cache server store a copy of the recent files that access, and when another user access the same file, they can access it more quickly since they no need to request again from file server, but directly communicate with cache server. One (1) of the famous cache servers is Squid cache server. Squid cache server using *squid.conf* to make a configuration, so the network administrator or network engineer need to know how and where to write the configuration command, and then save to file. After that, the network administrator or network engineer needs to restart Squid service so that it can apply the configuration.

SQ Squid cache server Management front end is a graphical user interface to configure the *squid.conf* file. All option provided by Squid cache server include the access control list, port number to listen and etc is configure by using graphical user interface which make it simple to understand, and easier to configure. When network

administrator or network engineer apply the configuration, it will automatically restart the squid service, so that the squid service will run with new configuration.

1.2 Problem Statement

Network administrator or network engineer need extra knowledge on squid cache server, before they can start the Squid service as cache server. Network administrator or network engineer need to learn how to set up the port number listen to the internet traffic.

Network administrator or network engineer need to be expertise in using Squid cache server, that they need to learn all the command and where to put the command in order to apply the configuration to the Squid cache server.

The *squid.conf* file is configuration file for Squid cache server. Network administrator or network engineer need to deal with the file in order to configure the cache server, which means directly deal with the database for setting for the server without specific applications.

1.3 Objective

- i. Create application to configure *squid.conf* file.
- ii. Build graphical user interface management front end for Squid cache server.
- iii. Ensure accurate and correctness of the configuration parameters entered.

1.4 Scope

The SQ squid management front ends not a new cache server, but only application to configure the existing cache server – Squid.

SQ squid management front end is developing using Microsoft C# language, so the platform to run the program limited to the Windows. (Since Linux platform cannot run Execution file).

The users of this software limited to the network engineers, network administrator, and mostly used in big organization, since there is no need to set up the cache server in organization when the number of employees are limited.

CHAPTER 2

LITERATURE REVIEW

2.1 Internet

Advance of the Computer Technologies has led to further advancement in Internet Technologies, nowadays, internet becomes an essential component, and even some of us consider it's as part of our life. Internet is a worldwide public network that connects multiple individuals' networks.

Internet originated from ARPANET, a U.S. Government funded project that aimed for inter-networking technologies that use for information sharing at 1970s. A standard that called TCP/IP (Transmission Control Protocol/ Internet Protocol) was adopted as protocol for Internet. Every Internet users have assigned with unique IP address that defined by the protocol used (IP, AppleTalk, IPX and etc), and each users communicate with each other's with that unique IP address.

Development of the Internet makes our life easy. For example, Internet makes it possible to us to communicate, meet with new people by using World Wide Web technologies regardless the geographic area as long as we have internet access. Internet allows people to share their knowledge, information through forum, web site. Internet connects us with our beloved one (1) by mail, or instant messaging. Search engine like

live.com search engine, google.com, yahoo.com make it even easier to search for information we want from the internet.

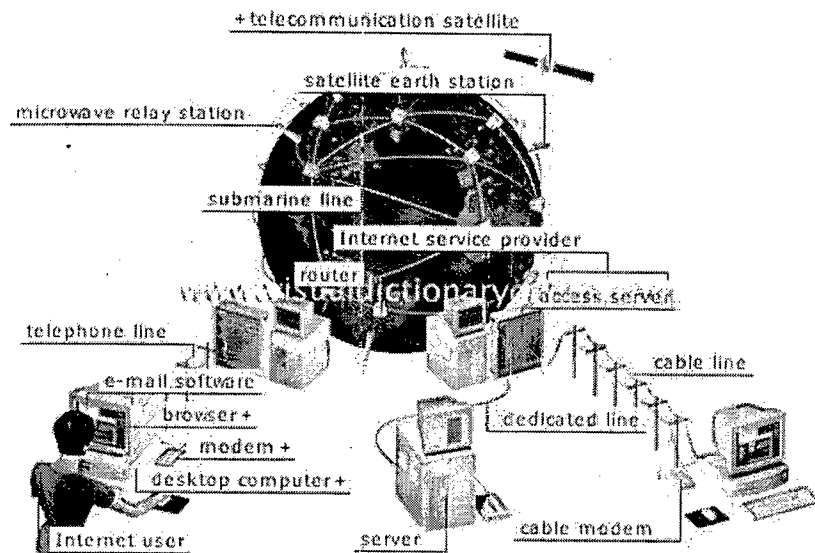


Figure 2.1: Internet

Although internet has brought ease to our life, but at the same time, another issue arises. Firstly, is about the internet performance, when number of internet users increase, logically, the performance of internet decreased. When number of users increase, the number of requests to server also increase, the response time of the server or latency to process the request will also increase, so, the performance will down. Besides that, security also one (1) of the issues that created by Internet. The main point why Internet brings security issues is worldwide and public. When user connects to internet, the user also exposed to the network threat like hacking, denial of service, viruses, spamming, hijacking information by unintended parties. Furthermore, there's lack of appropriate policy or regulation. There is no clear regulation or policy on how user used the internet. So, the user might use internet for illegal purpose, for example, underage children access to pornography through web site when parents is not around.

2.2 Proxy Technologies

As internet arise the issues about the security, lack of regulation and policy, many solutions has being developed to solve the problem. Proxy technology is one (1) of the effective solutions for one (1) organization to implement policy and regulation, and security for their users.

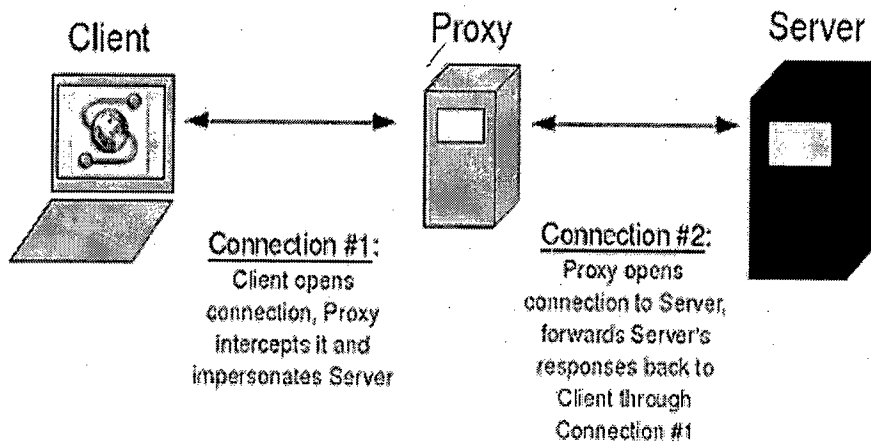


Figure 2.2: How Proxies Work

Proxy server resides between the organizational network and the external network (internet). Any request to the Internet from the organizational network or vice versa must process by proxy server before they started to communicate. In other words, proxy server act as request filter, that it's decide which request can be proceed or which request need to be deny access. Proxy server also act like security guard which hide server and user's IP address from external network, so that hacker or external threat cannot directly communicate with the users or server, which lead to stronger security.

2.3 Caching

Caching is one (1) of the methods that used by big organization to improve performance over internet. Cache server (actually also called proxy server) stores a request and result for first request until timer expired. When second or later same request to the server, the cache server immediately return the result to the users, hence reduce the latency and response time to process request by the server, and at the same time reduce the load on internet. So, the performance of internet will increase.

However, changing of web nature from static to dynamic (web content for web site change every time we access) make caching impossible, and hence its caching to improve performance become an issues. However, since there are still many static web content, so, caching still relevant to improve the performance.

2.4 Squid Cache Server

Squid Cache Server is open source proxy or cache server that originally develop for Linux platform. Nowadays, there being developed a version that can run at Microsoft Windows platform. Squid cache server is standalone cache server that support authentication via a number of methods, including LDAP, SMB and PAM., access control list, filter request both based on destination or source address. Squid cache server can use by network administrator or network engineer to monitor the traffic or network utilization of their user.

2.4.1 *squid.conf* File

Squid caches server configuration file, *squid.conf* is where all setting includes basic configuration: port number to listen to request, access control list, policy and regulation. The network administrator or network engineer that want to configure the

Squid cache server need to open the *squid.conf* file with text editor, like notepad or WordPad in order to write a command to configure the server. Next step is save the file, and restart the Squid service. There is no Graphical User Interface (GUI) configuration interface to directly configure the Squid cache server, so extra knowledge on using Squid command is required before network administrator or network engineer can configure the Squid cache server. Examples of configuration line we can make using *squid.conf* file are shown in appendix A.

2.5 Technologies

Technologies that used to develop the application include Microsoft .NET framework, Visual C# language and in Graphical User Interface (GUI).

2.5.1 Graphical User Interface

Nowadays, most of the applications are designed with GUI, since it's make the application systems itself becomes more attractive, easier to used compare to command prone.

GUI, Graphical User Interface is hierarchical graphical front end for our systems that accept users-generated and system-generated events, from a fixed set of events as input and produce deterministic graphical output. In others words, the GUI is a front end for your systems, that contain graphical widget, that have its own properties, for example button, image, link and etc, which then respond to the user's input, or systems event, to produce deterministic graphical output.

Graphical user interface becomes more popular from time to time, Microsoft propriety programming language like Visual C#, Visual Basic.NET and etc almost all even come with tool box for programmer to drag and drop to design interface for their

systems that even shorten systems developing time. However, those don't mean command prone already stop their usages, the open sources software, like Linux still using command prone for their systems. Table below show the comparison of command prone systems and graphical user interface systems.

Table 2.1: Compare GUI and Command prone systems

	GUI	Command Prone
Graphical Widget	Have	No
Input	Can be Text, picture, or set of events either from systems or users.	Text, or more specifically command
Output	Can be Text, picture, or set of events that generated by application.	Text

2.5.2 Microsoft .NET Framework

From MSDN library, .NET Framework is an integrated Windows component that support building and running of the next generation application, and XML web services. In other words, .NET Framework is designed as programming code execution environment for .NET Application. .NET Framework contains two (2) main components that are common language runtime (CLR) and class library. Common Language Runtime is run-time environment for application code while the class library contains collection of reusable code that integrates with CLR.

2.5.3 Programming Language – Microsoft Visual C#

C# (pronounced “C sharp”) is an object oriented programming language that is designed for building variety of .NET applications. Compare to other language like C or Java, C# is simple, powerful, and type safe. The innovations in C# enable rapid development for application while retaining the expressiveness and elegance of C-style languages.

Microsoft Visual C# is an implementation of the C# language by Microsoft. Visual Studio either is Professional edition or Express edition that supports Visual C# with a full-featured code editor, compiler, project templates, designers, code wizards, a powerful and easy-to-use debugger, and other tools. The .NET Framework class library provides access for C# application to many Microsoft operating system services and other useful, well-designed classes that speed up the development cycle significantly.

CHAPTER 3

METHODOLOGY

3.1 Project Model

Refer to figure 3.1 Project Model, the system administrator responsible to configure the cache server (Squid cache server) by deal with SQ management front end which is GUI based management front end. SQ management front end mainly have two (2) functions, one (1) is to deal with *squid.conf* file, which is the configuration file for Squid cache server, and one (1) is to deal with Squid cache server itself. SQ management front end can read the existing configuration (if exist) or write the user configuration to the file. In this case, the SQ management front end treat *squid.conf* file as database. The second function is deal with Squid cache server, which used to start or stop the Squid service.

Squid cache server side resides between internal LAN and external Internet. The firewall will force the connection between internal LAN and external Internet through the cache server in order to apply policy, regulation, and to filter request.