

# TABLE OF CONTENTS

<b>1. INTRODUCTION .....</b>	<b>5</b>
1.1 BACKGROUND .....	5
1.2 PURPOSE OF STUDY .....	6
1.3 ASSUMPTIONS .....	6
<b>2. LITERATURE REVIEW .....</b>	<b>8</b>
<b>2.1. CRYPTOGRAPHY .....</b>	<b>8</b>
2.1.1 SYMMETRIC AND ASYMMETRIC CRYPTOSYSTEMS .....	8
2.1.2 SYMMETRIC KEY ENCRYPTION .....	9
I. Data Encryption Standard (DES).....	9
II. International Data Encryption Algorithm (IDEA).....	9
III. CAST.....	10
IV. Skipjack.....	10
V. RC2/RC4.....	10
VI. The Blowfish Encryption Algorithm.....	11
VII. The ICE Encryption Algorithm.....	11
2.1.3 ASYMMETRIC KEY ENCRYPTION .....	11
I. RSA .....	11
II. Digital Signature Standard (DSS).....	12
III. Message Digest Algorithms – MD2, MD4 and MD5.....	12
IV. known as Secure Hash Standard/Secure Hash Algorithm (SHS/SHA) .....	13
V. Certificates .....	13
Encryption / Decryption Tools / Scheme .....	14
<b>2.2 AUTHENTICATION.....</b>	<b>15</b>
2.2.1 WHAT IS AUTHENTICATION?.....	15
2.2.2 AUTHENTICATION PROTOCOL .....	15
2.2.3 AUTHENTICATION METHODS .....	20
<b>2.3 FIREWALLS .....</b>	<b>22</b>
2.3.1 FIREWALL DEFINITIONS .....	22
2.3.2 FIREWALL COMPONENTS [23].....	23
2.3.3 FIREWALL TECHNOLOGY .....	25
I. Packet filtering .....	25
II. Application level proxy servers .....	28
2.3.4 FIREWALL CONFIGURATIONS .....	29
Packet filtering firewalls .....	30
Application-level firewalls.....	31
Hybrid firewalls (Hybrid Gateways).....	38
Circuit-level Firewalls.....	38
Other Firewall configurations.....	39
2.3.4 FIREWALL DESIGN, IMPLEMENTATIONS AND OTHER CONSIDERATIONS .....	41
I. Internet Services to be configured at Firewall .....	41
II. Authentication and access control needed at the Firewall.....	42
III. How to select a firewall?.....	42
IV. To build a firewall or set up the firewall using the firewall package outside.....	43
V. How to build a firewall?.....	44
VI. How to select the type of firewall products on the markets? .....	44
VII How to maintain firewalls? .....	45
2.3.5 FIREWALL SECURITY POLICY .....	46
- Firewall design policy.....	46
- Service access Policy.....	46

- *Information Policy*.....46
- *Dial-in and Dial-out policy* .....46
- *Flexibility policy*.....47
- 2.3.6 INTRUSION DETECTION SYSTEM.....47
- 2.3.7 INTRUSION DETECTION METHODS.....48
- 2.3.8 VULNERABILITY ASSESSMENT .....49
- 3. METHODOLOGY .....50**
- 3.1 SETTING UP FIREWALL WITH DIFFERENT SECURITY LEVELS.....50
  - 3.1.1 *Hardware and software components*.....50
  - 3.1.2 *Security zones in the testing network*.....51
- 3.2 SECURITY TESTING .....53
  - 3.2.1 *Network Scanning/Monitoring tools for Firewall Testing*.....54
  - 3.2.2 *Testing Procedures and Details*.....58
- 3.3 PERFORMANCE TESTING .....58
  - 3.3.1 *Tools* .....59
  - 3.3.2 *Assumptions*.....59
  - 3.3.3 *Measurement* .....60
- 4. FIREWALL CONFIGURATION AND POLICY SETUP.....64**
- 4.1 FIREWALL POLICY AND SCREENING RULES SETUP.....64
- 5. ANALYSIS OF RESULTS AND DISCUSSIONS .....74**
- 5.1 SECURITY TESTING .....74
  - 5.1.1 *Summary of results* .....75
  - 5.1.2 *Analysis*.....78
- 5.2 PERFORMANCE TESTING .....79
- 5.3 RELATIONSHIP OF SECURITY TO PERFORMANCE .....100
  - 5.3.1 *For performance tests of HTTP data transfer with 395K data*.....101
  - 5.3.2 *For performance tests of FTP data transfer with 5M data*.....102
  - 5.3.3 *For performance tests of ftp data transfer with 1M data*.....104
  - 5.3.4 *For performance tests of FTP data transfer with 38.9K*.....105
- 6. LIMITATIONS .....107**
- 7. FUTURE WORK .....108**
- 8. CONCLUSION .....110**
- 9. REFERENCES .....112**
- 10. APPENDICS .....115**
- APPENDIX A - FIREWALL POLICIES IMPLEMENTATION BY SCREENING RULES .....115
- APPENDIX B – PLUGIN LIST OF NESSUS .....121
- APPENDIX C – RAW DATA SET .....147
- APPENDIX E .....157
- APPENDIX F.....158
- APPENDIX G .....166
- APPENDIX H - A COMPARISON BETWEEN PROXY GATEWAY AND PACKET FILTER.....168

## LIST OF TABLES

TABLE 1: THE AVERAGE TOTAL HTTP TRANSACTION TIMES IN SECOND .....	79
TABLE 2: THE BEST TOTAL HTTP TRANSACTION TIMES IN SECOND.....	79
TABLE 3: LATENCY CALCULATED WITH AVERAGE TOTAL HTTP TRANSACTION TIME TRANSACTION TIMES IN SECOND .....	81
TABLE 4: LATENCY CALCULATED WITH BEST TOTAL HTTP TRANSACTION TIME TRANSACTION TIMES IN SECOND .....	81
TABLE 5 : AVERAGE TOTAL TRANSACTION FIGURES .....	86
TABLE 6: BEST FIGURES WITH MINIMUM TRANSACTION TIME OF THE RESULT .....	86
TABLE 7 : LATENCY CALCULATED FROM THE TOTAL AVERAGE TRANSACTION TIMES OF FTP 5M DATA.....	88
TABLE 8 : LATENCY CALCULATED FROM THE TOTAL (BEST) MINIMUM TIMES OF FTP 5M DATA .....	88
TABLE 9 : AVERAGE FIGURES OF TOTAL TRANSACTION TIMES.....	90
TABLE 10: BEST FIGURES WITH MINIMUM TRANSACTION TIME OF THE RESULT.....	90
TABLE 11 : LATENCY CALCULATED FROM THE AVERAGE FIGURES ABOVE .....	92
TABLE 12 : LATENCY CALCULATED FROM THE (BEST) MINIMUM FIGURES ABOVE .....	92
TABLE 13 : TOTAL AVERAGE TRANSACTION TIMES VS NO. OF CONNECTION IN A TRANSACTION .....	94
TABLE 14: TOTAL BEST TRANSACTION TIME VS NO. OF CONNECTION IN A TRANSACTION.....	94
TABLE 15 : LATENCY CALCULATED FROM THE AVERAGE FIGURES ABOVE .....	96
TABLE 16 : LATENCY CALCULATED FROM THE (BEST) MINIMUM FIGURES ABOVE .....	96

## LIST OF FIGURES

FIGURE 1: A TYPICAL PACKET FILTERING SYSTEM, BY USING BASTION INSTALLED WITH PACKET FILTERING SOFTWARE OR A SCREENING ROUTER .....	25
FIGURE 2: PROXY: THE ACTUAL CONNECTION IS FROM CLIENT – PROXY SERVER – REAL SERVER; .....	28
FIGURE 3: FIREWALL USING SCREENING ROUTER .....	30
FIGURE 4: APPLICATION LAYER GATEWAYS WORKING AT APPLICATION LAYER [14] .....	32
FIGURE 5: A TYPICAL DUAL HOMED-HOST FIREWALL.....	33
FIGURE 6: A SCREENED HOST FIREWALL.....	35
FIGURE 7: A TYPICAL SCREENED SUBNET WITH INTERIOR AND EXTERIOR ROUTERS .....	37
FIGURE 8: TEST BED CONFIGURATION.....	52
FIGURE 9: THE HTTP TOTAL AVERAGE TRANSACTIONS TIMES VS THE NO. OF CONNECTION(S) UNDER DIFFERENT FIREWALL SECURITY LEVELS .....	80
FIGURE 10: THE HTTP TOTAL BEST TRANSACTIONS TIMES VS THE NO. OF CONNECTION(S) UNDER DIFFERENT FIREWALL SECURITY LEVELS.....	80
FIGURE 11: THE AVERAGE HTTP LATENCY OF A TRANSACTIONS VS THE NO. OF CONNECTION(S) UNDER DIFFERENT FIREWALL SECURITY LEVELS .....	82
FIGURE 12: THE BEST HTTP LATENCY OF A TRANSACTIONS VS THE NO. OF CONNECTION(S) UNDER DIFFERENT FIREWALL SECURITY LEVELS.....	82
FIGURE 13 : TOTAL TRANSACTION TIME ON AVERAGE FOR DATA TRANSFER BY FTP VS NO. OF CONNECTION.....	87
FIGURE 14 : MINIMUM TOTAL TRANSACTION TIME FOR DATA TRANSFER BY FTP VS NO. OF CONNECTION .....	87
FIGURE 15 : LATENCY CALCULATED FROM THE AVERAGE TL TRANSACTIONS TIMES VS NO. OF CONNECTION(S).....	89
FIGURE 16 : MINIMUM LATENCY CALCULATED FROM THE (BEST) MINIMUM TIMES VS NO. OF CONNECTION(S).....	89
FIGURE 17 : TOTAL AVERAGE TRANSACTIONS TIMES VS NO. OF CONNECTION(S) FOR IM DATA TRANSFER .....	91
FIGURE 18 : TOTAL MINIMUM TRANSACTIONS TIMES VS NO. OF CONNECTION(S) FOR 1M DATA TRANSFER .....	91
FIGURE 19 : LATENCY CALCULATED FROM THE AVERAGE TL TRANSACTIONS TIMES VS NO. OF CONNECTION(S) FOR 1M DATA TRANSFER .....	93
FIGURE 20 : LATENCY CALCULATED FROM THE MINIMUM TL TRANSACTIONS TIMES VS NO. OF CONNECTION(S) FOR 1M DATA TRANSFER .....	93
FIGURE 21 : TOTAL AVERAGE TL TRANSACTIONS TIMES VS NO. OF CONNECTION(S) FOR 38.9KM DATA TRANSFER.....	95
FIGURE 22 : TOTAL MINIMUM TRANSACTIONS TIMES VS NO. OF CONNECTION(S) FOR 38.9KM DATA TRANSFER.....	95
FIGURE 23 : LATENCY CALCULATED FROM THE AVERAGE TL TRANSACTIONS TIMES VS NO. OF CONNECTION REQUEST(S) FOR 38.9KM DATA TRANSFER .....	97
FIGURE 24 : LATENCY CALCULATED FROM THE MINIMUM TL TRANSACTIONS TIMES VS NO. OF CONNECTION REQUEST(S) FOR 38.9KM DATA TRANSFER .....	97
FIGURE 25: SECURITY-PERFORMANCE MATRIX .....	100
FIGURE 26 SECURITY-PERFORMANCE MATRIX .....	102
FIGURE 27: SECURITY-PERFORMANCE MATRIX .....	103
FIGURE 28: SECURITY-PERFORMANCE MATRIX .....	105
FIGURE 29: SECURITY-PERFORMANCE MATRIX .....	106
FIGURE 30: SECURITY-PERFORMANCE MATRIX .....	109

# **1. INTRODUCTION**

## **1.1 Background**

With the fast growing of Internet access in Hong Kong and everywhere else, network security comes up to be a major concern in doing business on the web or protecting individual or company privacy. The security issues on distributed systems have been widely discussed. Basically the security requirement of an organization covers the aspects including user identification, authentication, data encryption and decryption as well as protection which finally results in building and protecting a private network securely against any intrusion and losses. However, it was stated in news (15April1999, the Oriental Daily News about "Hackers") that less than 50% of small to medium size companies in Hong Kong adopts any security measure such as firewall, with a view to protecting their network sites against any external attacks or intrusion. It seems that most of these companies are not aware of the severity of their security problems.

As a matter of fact, one of the most effective ways of securing an internal network is using firewall. Many large organizations gaining access to the Internet would have their firewall built up. Once a firewall system is build up, hard testing has to be started before the live-run, to see if the firewall is effective in protecting the internal network. Testing on firewall is important and would be made as part of an audit or assessment on the firewall.

In this project, the common security issues and interesting topic are researched, in order to get a clearer picture of how the security problems in the web are usually dealt with. Furthermore by setting up a firewall system with the Linux TIS firewall package and a router, it is expected that a secured firewall could be implemented with all the necessary security features in this project. Furthermore, is there any tradeoff between higher security and network performance? How much performance gains or loss if a firewall is used for security concern? This paper examined the impact on performance of firewall by doing some testing on the firewall system.

In fact, Security is more or less a "people problem" since most of a company's real security problems will be related to the company's staff and their attitudes, not to the technical security. In this way, if a company hired a hacker, the company may be exposed to the possible dangerous people. In addition to sounded technical security, proper and well known "usage guidelines" for the network is important in ensuring the network security level. Also good procedures for handling calls from users asking for passwords to be renewed and for handling private information have to be

carefully established. The procedures, practice and guidelines of an organization have to be evaluated from time to time, to ensure they are conformed to organization security policy and standard. In this project, various security policies were covered in more details in the section of literature review. Also they were implemented in the firewall system of this project. Security and performance tests are used to determine how effective the policies are in securing a private network.

Finally with all the support from the security and performance testing results, it is interesting that a security to firewall performance relationship matrix is proposed and presented. Further works on studying the many combination of various security levels and firewall performance were suggested.

## **1.2 Purpose of Study**

This project is to study the security issues on distributed systems. By desk research on the various security related topics such as identification, encryption and decryption, and by some experiments on firewalls, an in-depth approach about securing an internal network with firewall will be presented in the project. Moreover, different firewall policies and configurations will be attempted to determine the impact from added security on the firewall performance with respect to data transfer. The objectives of this research are specified as follows.

- I. To survey on the various distributed systems security related topics such as encryption and decryption schemes, network authentication protocols and firewall in the literature review.
- II. To evaluate the security control and performance of different firewall configurations by doing some testing on firewall with different firewall security levels and proxy services.
- III. To investigate the impact of different levels of firewall security and measures on the performance of firewall system and try to quantify the performance difference.
- IV. To determine how well the various firewall systems in guarding the private network against some potential external attacks and scanning from network scanners such as 'nessus'.
- V. To examine and try to deduce a relationship between security and performance from the testing result.

## **1.3 Assumptions**

- As the firewall system for this project is set up as a small intranet attached to the department network, it is assumed that the computer LAN is the Internet and the department's complex

network is good enough to play as the real Internet. All the testing would be done under such a testing environment in the department laboratory.

- The firewall system for this project would be built up by using the limited resource from the department, it is supposed to be simulated as a real firewall as possible. However, it may not be the same as the real working firewall under certain extent, at least the intranet may not be comparable to an real internal network, and the department users may not be perceived as the various Internet users including the network hacker.
- The network intrusion would be simulated as the real cases as possible in order to audit the firewall system. Some publicly available network scanners would do attacks and network scanning on firewall from outside and they would be adopted in the testing for this project. It is believed that the scanners adopted in the project would be effective in determining any possible the vulnerabilities and security flaws of the firewall from outside.
- The security level of the firewall system is assessed according to no published security level on computer systems. However it is setup with appropriate network components and security measures needed to implement the seven security levels and policies suggested in this project, without seeking any professional assistance or expertise. Based on only my personal assessment and opinion, the assessment may be somewhat subjective, but is assumed to be adequate.
- This project aimed at exploring the difference of network performance among different security levels. It is assumed that the performance figures measured under a particular firewall policy was accurate and suitable for comparison, even no effort was made to validate the figures and there was no need to do so.

In live cases, tests and experiments to be done are usually used in auditing, assessing and determining the security level of a secured firewall. However, on the other way around, the tests can be used to determine the performance of the firewalls of different security levels. For this project, it is assumed that the security level of a particular firewall is predefined with some security measures and firewall policies, testing is only used to ensure that the actual security implementation is expected. The term “firewall policy”, “configuration”, “level” are always referred to a particular firewall setup.

## 2. LITERATURE REVIEW

### 2.1. CRYPTOGRAPHY

In the old times, cryptography was developed by military to conceal the content of secret message from enemies which could not understand the message even they got it without a key. Nowadays, computer is the main tool for cryptography and distributed communication system depends a lot on cryptography to assure communication authenticity and message integrity. In other words, cryptography is applied in dealing with that various security threats encountered such as address spoofing attack in the Internet communications.

#### 2.1.1 Symmetric and Asymmetric Cryptosystems

In general, a cryptosystem comes with two important procedures, encryption and decryption [19]. Cryptosystems can be divided into two classes, the symmetric and asymmetric. For symmetric cryptosystem (also called shared key or private key cryptosystem), encryption and decryption key are the same and must be kept secret. For asymmetric cryptosystem (also called public key cryptosystem), the encryption key is different from the decryption key. The encryption key can be made public whereas the decryption key has to be kept secret.

Encryption is the function, which encrypts arbitrary messages with encryption key while decryption function is to recover the message into its original form from its encrypted form by using the decryption key. Encryption and decryption satisfy the relation as:

$M$  is message space,  $K_E \times K_D$  is the set of encryption and decryption keys [19].

$$\forall m \in M : \forall (k, k^{-1}) \in K_E \times K_D : \{ \{m\}_{k^{-1}} \}_k = m$$

$k, k^{-1}$  are the decryption and encryption respectively.  $\{m\}_{k^{-1}}$  can be used as a signature on message  $m$  by  $P$  which is supposed to be the only principal knows  $k^{-1}$ . As seen in the above relation,  $P$ 's signature on  $m$  can be verified by anyone with the knowledge of  $k$ .



## 2.1.2 SYMMETRIC KEY ENCRYPTION

### I. Data Encryption Standard (DES)

DES is one of the most popular **private key** algorithms. It is developed by IBM and became an official U.S. government standard in 1976. The U.S. government forbids export of hardware and software product that contains DES implementations even though the implementations of DES are widely available outside U.S.[20]. Kerberos uses DES algorithm to encrypt data for various transactions.

DES is very fast, at least 100 times faster than RSA algorithm when implemented in software, and even 1000 times faster when implemented in hardware where DES uses S-boxes and simple table look-up functions, while RSA depends much on very-large-integer arithmetic.

The key of DES can be just about any 64-bit number. The effective length is regarded as 56 bits. There is only one way to break DES, through an exhaustive search of the keyspace with  $2^{56}$  total possible keys which have to take 2000 years if one millions keys are tried for every second.[20].

Although DES is very secure, many attempts had been tried to break it. One group known as DES Challenge (DESCHALL) was set up to meet the challenge. They used the techniques called brute-force with many computers participating to try every possible decryption key, located at <http://www.frii.com/~rcv/deschall.htm>.

### II. International Data Encryption Algorithm (IDEA)

Xuejia Lai and James Massey of the Swiss Federal Institutes of Technology developed it. IDEA uses block size of 64 bites and cipher feedback operation, which made the algorithm stronger. It spreads out the content of a plain-text over many ciphertext bits, thus hides the statistical structure of the plain text completely.

The key length is 128 bits, the longer the key, and the better the algorithm. Due to the use of 64-bit block size, IDEA works fine for FTP by which large amount of data is transferred, but performs poorly with Telnet,

There is secure file encryption program uses IDEA developed by Fauzan Mirza, called Tiny IDEA (<http://www.dcs.rhbnc.ac.uk/~fauzan/tinyidea.html>).

### III. CAST

Carlisle Adams and Stafford Tavares developed it. CAST uses a block size of 64 bits and a 64-bit key. Also it uses 6 S-boxes with 8-bit input and 32-bit output data. The encryption algorithm has 8 rounds, half of the plaintext block is combined with some key material using a function  $f$  and then XORed with the other block in each round, the left one to form a new right block and the old right one to form the new left block. The function  $f$  of the algorithm can be described as follows[20]:

1. Divide a 32-bit input into 4 8-bit quarter i.e. a, b, c, d
2. Divide the 16-bit subkey (the 64-bit key is divided into 4) into 2 8-bit halves i.e. e,f.
3. Process a through S-box1, b through S-box 2, c through S-box 3, d through S-box 4, e through S-box 5 and finally f through S-box 6.
4. XOR the 6 S-box outputs together to get the final 32-bit output.

\* S-box (selection box) is a set of highly nonlinear functions, which are implemented in DES as lookup tables.

After the 8 rounds, the two halves will become a ciphertext. For further reference, check <http://www.cs.wm.edu/~hallyn/des/sbox.html>

### IV. Skipjack

It was developed by the NSA for the Clipper chips, which is a commercial chip for encryption using Skipjack algorithm. This encryption algorithm uses an **80-bit key** and there are 32 rounds of processing in each encryption or decryption operation.

Actually not much is known about this algorithm because it is regarded as secret by U.S. government [20]. For further reference, check

[http://www.cpsr.org/cpsr/privacy/crypto/clipper/skipjack\\_interim\\_review.txt](http://www.cpsr.org/cpsr/privacy/crypto/clipper/skipjack_interim_review.txt) or

<http://www.austinlinks.com/Crypto/non-tech.html> for more about Clipper wiretap chip.

### V. RC2/RC4

It was designed by RSA Data Security, Inc. and is a very fast algorithm. Even it is regarded as a strong algorithm, some independent group had taken about 8 days to break the exportable version of Netscape's SSL which uses RC-4-40. It has key of 40 bits and 128 bits. It has been using by

Microsoft in their communication service for dial-up and VPN connections using Microsoft Point to Point Encryption.

## **VI. The Blowfish Encryption Algorithm**

Blowfish was designed in 1993 by Bruce Schneier as a fast, free alternative to existing encryption algorithms. Since then it has been analyzed considerably, and it is slowly gaining acceptance as a strong encryption algorithm. Blowfish is a symmetric block cipher that can be used as a drop-in replacement for DES or IDEA.

It takes a variable-length key, from 32 bits to 448 bits, making it ideal for both domestic and exportable use. Blowfish is unpatented and license-free, and is available free for all uses. A Java implementation of Blowfish is available as part of Cryptix-Java.

(<http://www.counterpane.com/blowfish.html>)

A reference implementation of Blowfish (ECB, CBC, CFB, and OFB modes) is available at <ftp.psy.uq.oz.au>, <fractal.mta.ca>, or <ftp.ox.ac.uk>.

## **VII. The ICE Encryption Algorithm**

ICE is a 64-bit private-key block cipher, similar to DES. The code implements the class IceKey, which carries out encryption, decryption, and key changes, using the ICE algorithm.

The algorithm and source code are public domain. (<http://www.cs.mu.oz.au/~mkwan/ice>).

## **2.1.3 ASYMMETRIC KEY ENCRYPTION**

This key encryption helps to eliminate the problems of distributing key to users. However, the keys used for the algorithm are usually large, with 100 or more digits. As a result it incurs key management and computing overhead problems.

### **I. RSA**

It was developed by 3 scientists, Ron Rivest, Adi Shamir and Leonard Adleman in 1977. It is well known as widely used in public key cryptosystem. The keys of RSA are devised as follows.

- Choose 2 large primes say  $p$  and  $q$ , and then find their product  $n = pq$ .
- Choose another number  $e$ , which is  $< n$ , but relatively prime to  $(p-1)(q-1)$ , then find its inverse,  $d$ ,  $\text{mod}(p-1)(q-1)$ . That is  $ed = 1$ .
- $e$  is the public exponent and  $d$  is called the private exponent.
- The public key pair is  $(n, e)$ . The private key is  $d$ . The factors  $p$  and  $q$  must be kept secret.

RSA is combined with MD5 hashing function to sign a message in the RSA-MD5 Signature Suite. For details, please refer to [http://www.w3.org/TR/1998/PR-DSig-label-19980403/RSA-MD5-1\\_0.htm](http://www.w3.org/TR/1998/PR-DSig-label-19980403/RSA-MD5-1_0.htm).

## II. Digital Signature Standard (DSS)

It is a **standard** for digital signing by U.S. government. This standard specifies a Digital Signature Algorithm (DSA) which can be used to generate a digital signature. Digital signatures are used to detect unauthorized modifications to data and to authenticate the identity of the user who generates the signature. In addition, the recipient of signed data can use a digital signature in proving to a third party that the signature was in fact generated by the signer of the data. This is known as non-repudiation since the signer of data cannot, at a later time, repudiate the signature.

For more details, please refer to:

[http://www.eff.org/pub/Privacy/Digital\\_money/Anonymity/Digital\\_money/Anonymity/Digital\\_signature/fips\\_dss\\_proposed.standard](http://www.eff.org/pub/Privacy/Digital_money/Anonymity/Digital_money/Anonymity/Digital_signature/fips_dss_proposed.standard)

For the Bulletin of DSS, please refer to:

[http://www.eff.org/pub/Privacy/Digital\\_money/Anonymity/Digital\\_money/Anonymity/Digital\\_signature/nist\\_dss.bulletin](http://www.eff.org/pub/Privacy/Digital_money/Anonymity/Digital_money/Anonymity/Digital_signature/nist_dss.bulletin)

## III. Message Digest Algorithms – MD2, MD4 and MD5

Message Digest is the representation of text in the form of a single string of digits, created using a formula called a one-way hash function. Encrypting a message digest with a private key creates a digital signature, which is an electronic means of authentication. In order to avoid intruder attaching any false message onto any other person's valid message or signature, it should not be possible to find two or more than two messages that hash to a same value. The hash function MD5 was designed specifically to have the property that finding a match mentioned above is infeasible. ([http://webopedia.internet.com/TERM/m/message\\_digest.html](http://webopedia.internet.com/TERM/m/message_digest.html))

The MD5 Message Digest Algorithm is the latest version of the MDs and is considered to be more stable.

For more details about MD5, please go to the URL below:

- <http://www.cert.org/security-improvement/implementations/i002.01.html>
- <http://www.alternic.net/rfcs/1300/rfc1321.txt.html>

Source code and additional information are available via FTP from <ftp://info.cert.org/pub/tools/md5>

.

There had been the security weakness found in Windows NT, which involved the security of the MD4. To crack the password on Windows NT, there are the utilities available on the Internet. ( **PWDDUMP** – from <http://www.masteringcomputers.com/util/nt/pwdump.htm> and **NTCRACK** – <http://www.masteringcomputers.com/util/nt/ntcrack.htm>. )

If running Internet Explorer, which also exposes security flaws, one can try the cracking tools by accessing <http://www.efsl.com/security/ntie> .

For other information about security cracking, please go to

- <http://www.lullaby.demon.co.uk/rtech/pi/nt.htm>
- <http://mssg.rutgers.edu/langroup/online/nt/hack.htm>

#### **IV. known as Secure Hash Standard/Secure Hash Algorithm (SHS/SHA)**

SHA, also SHS, was developed by U.S. government. It is capable of producing a 160-bit hash value from an arbitrary length string. The structure of it is similar to MD4/MD5. Because SHA produces 25% longer message digest than MD does, it is 25% slower but 25% more secure to brute-force attack than MD function.

#### **V. Certificates**

A public-key certificate is a data structure used to securely bind a public key to attributes, which are the identification information such as name, permission. A standard for identification is contained within the international standards for directories. For example X.509 certificate binds a public key to a directory name [19]. Privacy Enhanced Mail (PEM) also employs X.509 certificates.

On the other hand, A digital certificate is viewed as an electronic "credit card" that establishes your credentials when doing business or other transactions on the Web. A certification authority (CA) is responsible to issue it. It contains your name, a serial number, expiration dates, a copy of the certificate holder's public key (used for encrypting and decrypting messages and digital signatures), and the digital signature of the certificate-issuing authority so that a recipient can verify that the certificate is real. Some digital certificates conform to a standard, X.509. Digital certificates can be kept in registries so that authenticated users can look up other users' public keys. [26

*Several well-known Certificate Servers are listed as follows:*

- Netscape's: [http://home.netscape.com/comprod/server\\_central/support/faq/certificate\\_faq.html#1](http://home.netscape.com/comprod/server_central/support/faq/certificate_faq.html#1)
- OpenSoft's: <http://www.opensoft.com/products/expressmail/overview/certserver/> which is based on Distributed Certificate System (DCS).
- Thawte is a leading global provider of digital certificates and digital certificate services for secure SSL web servers, email clients and browsers. ( <http://www.thawte.co.za/contents.html> )

### **Encryption / Decryption Tools / Scheme**

There are many different kinds of commercial tools for encryption and decryption on the market.

For example, the *Pretty Good Privacy* (PGP) (

<http://web.its.smu.edu/~dmcnickl/miscell/warnzimm.html> ) for e-mail privacy, *CodeDrag* (

<http://www.fim.uni-linz.ac.at/codeddrage/codeddrage.htm> ) for general data encryption and

decryption. Furthermore, *Netscape's Secure Sockets Layer (SSL)* is a popular encryption scheme that is now widely mentioned and adopted. Also Microsoft's encryption tool, *Private*

*Communications Technology (PCT)* protocol is well known as another kind of protocol for secured communications.

## 2.2 AUTHENTICATION

### 2.2.1 What is Authentication?

It is identification plus verification [19]. Identification is defined as the procedure by which one claims its certain identity while verification is the procedure by which the identify of the one claimed is to be checked. For distributed communications, the reliable authentication depends heavily on verification procedure, which in turn greatly relies on effective cryptography and authentication protocols.

In a distributed system, there are mainly three kinds of authentication [19], they are:

- *Message content authentication* – verifying that the content of a message received is the same as when it was sent.
- *Message origin authentication* – verifying that the sender of a received message is the same one specified in the sender field of the message.
- *General identity authentication* – verifying that a principal's identity is as claimed. Any entity in a distributed system, which we can distinctly identified, is regarded as principal such as a Certification Authorities CA or a client X.

### 2.2.2 Authentication Protocol

This is the protocol, which carry out authentication involving message exchange. For more detail reference, please refer to [24, 19] as well as <http://www.w3.org/People/Raggett/security/Authentication.html>

Two Popular Authentication services, Kerberos and SPX, are covered in the followings.

#### I. **Kerberos** [19,20,21]

Kerberos is a popular authentication service and it adopts the symmetric cryptosystem together with trusted third-party authentication servers.

Kerberos uses two main protocols, the credential initialization protocol and the client-server authentication protocol, which the clients used to request services from a server. These two protocols are discussed in the followings.

The **credential initialization protocol** authenticates user login and installs initial tickets at the login host. The processes are as follows.

*Assumptions:*

U : User who want to into a host H  
 T : Timestamp of ticket  
 Shared key  $k_U$  of U :  $k_U = f(\text{password}_U)$

H : Host  
 L : ticket's lifetime

Step 1: U → H : U  
 User U initiates login by entering its user name U.

Step 2: H → Kerberos : U, TGS  
 The login host H forwards the login request to a Kerberos server.

Step 3: Kerberos : retrieve  $k_U$  and  $k_{TGS}$  from database  
 : generate new session key k  
 : create ticket-granting ticket  
 tick<sub>TGS</sub> = {U, TGS, k, T, L}<sub>kTGS</sub>

Kerberos server retrieves the user record of U and generates the ticket-granting ticket.

Step 4: Kerberos → H : {TGS, k, T, L, tick<sub>TGS</sub>}<sub>kU</sub>  
 With the ticket-granting ticket, Kerberos server returns the ticket-granting ticket, together with its identity, user name U, session key k, timestamp T, lifetime of ticket T, encrypted with the public key of U, back to U.

Step 5: H → U : “password ? “  
 H asked U for its password.

Step 6: U → H : passwd  
 and U responses with its valid password.

Step 7: H : compute  $p = f(\text{passwd})$   
 : recover k, tick<sub>TGS</sub> by decrypting  
 {TGS, k, T, L, tick<sub>TGS</sub>}<sub>kU</sub> with p.  
 As p is supposed to be equal to  $k_U = f(\text{password}_U)$  .  
 : if decryption fails, abort login; otherwise retain  
 tick<sub>TGS</sub> and k  
 : erase passwd from memory

The **client-server authentication protocol** is used by the clients users to request services from a server. The steps of authentication are as follows.

Step 1: C → TGS : S, tick<sub>TGS</sub>, {C, T1}<sub>k</sub>  
 Client C (the user U above) presents its ticket-granting ticket to the ticket server (TGS) to request a ticket.

Step 2: TGS : recover k from tick<sub>TGS</sub> = {U, TGS, k, T, L}<sub>kTGS</sub> by  
 decrypting with  $k_{TGS}$   
 : recover T1 from {C, T1}<sub>k</sub> by decrypting with k  
 : check timeliness of T1 with respect to local clock  
 generate  
 a new session key k'



: create server ticket  $tick_S = \{C, S, k', T', L'\}_{k_S}$   
 If decryption is successful and T1 is timely, TGS creates a ticket  $tick_S$  for server S.

Step 3: TGS  $\rightarrow$  C :  $\{S, k', T', L', tick_S\}_k$   
 TGS presents C with the  $tick_S$  for server S, the new timestamp and lifetime of new ticket

Step 4: C : recover  $k', tick_S$  by decrypting with k

Step 5: C  $\rightarrow$  S :  $\{C, T2\}_{k'}$   
 C presents the server S with  $tick_S$  and a new authenticator.

Step 6: S : recover  $k'$  from  $tick_S = \{C, S, k', T', L'\}_{k_S}$ , by decrypting with  $k_S$   
 : recover T2 from  $\{C, T2\}_{k'}$  by decrypting with  $k'$   
 : check if T2 is timely with respect to the local clock.

The protocol requires loosely synchronized local clock for the verification of timestamps T?.

Step 7: S  $\rightarrow$  C :  $\{T2 + 1\}_{k'}$   
 Server S send back C with the encrypted new timestamp to assures C of the server's identity.

## II. SPX [19]

SPX adopts the both the symmetry and asymmetric cryptosystems technology to enhance security in open-network. [19]. This is used for Telnet authentication ( <http://intranet.www-kr.org/RFC/rfc/rfc1412.html> ).

It is a major component of Digital Distributed System Security Architecture. It has a credential initialization protocol, a client-server authentication protocol and an enrollment protocol that registers new principals. Only the first two protocols will be discussed in more details in this paper. SPX has a Login Enrollment Agent Facility (LEAF) and Certificate Distribution Center (CDC) that corresponds to Kerberos servers and TGSs. LEAF is used in the credential initialization protocol. CDC is an on-line depository of encrypted private keys of principals and of public-key certificates for and principals and certification authorities. There are also the hierarchically organized certification authorities (CAs) which are to issue public-key certificates and to operate offline and are selectively trusted by principals. Global trust is not needed in SPX. Each principal P typically trusts only a subset of all CAs, referred to as the trusted authorities of P. In fact, the scalability of the system is greatly enhanced without the global trust and on-line trusted components.

The *SPX credential initialization protocol* is performed as followings.

*Assumptions:*

U : User

H : Host



S's public-key certificate be signed by AS, where AC denotes a trusted authority of C

*Step 1:*  $C \rightarrow CDC$  : S  
C requested S's public-key certificate from CDC.

*Step 2:*  $CDC \rightarrow C$  :  $\{S, k_s\}_{K_{AS}^{-1}}$   
CDC returns the requested certificate C then can decrypt it with  $K_{AS}$  (the public-key of AS obtained by C when it is executed the credential initialization protocol) and verify it.

*Step 3:*  $C \rightarrow S$  :  $T, \{k\}_{k_s}, tick_C, \{k_d^{-1}\}_k$   
 $tick_C$  (refers to the  $tick_U$  in the credential initialization protocol) and the private delegation key  $k_d^{-1}$  (generated in step 7 of the credential initialization protocol), with a new session key  $k$ , are sent to S. Only S can recover  $k$  from  $\{k\}_{k_s}$ , and so recover  $k_d^{-1}$  from  $\{k_d^{-1}\}_k$  using  $k$ . Possession of  $tick_C$  and the knowledge of the private delegation key  $k_d^{-1}$  constitute sufficient proof of the delegation from C to S.

*Step 4:*  $S \rightarrow CDC$  : C  
S requests C's public-key certificate from CDC, which is used to verify  $tick_C$  later

*Step 5:*  $CDC \rightarrow S$  :  $\{C, k_C\}_{K_{AC}^{-1}}$   
CDC returns the requested public-key certificate to S

*Step 6:* S : recover  $k$  from  $\{k\}_{k_s}$   
: recover  $k_d^{-1}$  from  $\{k_d^{-1}\}_k$   
: recover  $k_d$  from  $tick_C$   
: verify that  $k_d$  and  $k_d^{-1}$  from a delegation key pair

S uses the C's public-key certificate to verify  $tick_C$

*Step 7:*  $S \rightarrow C$  :  $\{T + 1\}_k$   
S returns  $\{T + 1\}_k$  to C to complete the mutual authentication between C and S.

For SPX, it eliminates on-line trusted authentication servers and the extensive use of hierarchical trust relationships, and so are intended to make SPX scalable for very large distributed systems. However, it is relatively new and is to be researched more extensively.

Besides these two service there are also the Pretty Good Privacy PGP Signature Authentication, for which there is the International PGP Home page: <http://www.pgpi.com/>. Moreover, Netscape Communication Secure Socket Layer (SSL) protocol [7] is well known as it was designed to protect confidential data sent by Web browsers. For more information, please refer to <http://home.netscape.com/newsref/std/SSL.html>, and <http://pauillac.inria.fr/~doligez/ssl/>. For challenge, please go to <http://www.portal.com/~hfinney/sslchal.html>.

## 2.2.3 Authentication Methods

Much discussion on the authentication for distributed computing had been made in the past and its methods are varied. Here below are the several simpler ones for our reference [22].

### I. Password Authentication

This is usually the first line of defense against unauthorized access, using a login name and a password. In fact, further protection on password should be made such as encrypted password, instead of sending password as plain text for login authorization. It is because the protocol of password authentication is easily defeated using eavesdropping. If a hacker has access to the transmission media, the password message can be listened and recorded by him/her for later intrusion into the host system. One-time password (OTP) and smart card authentication are considered as another secure alternatives.

### II. Address Resolution

This kind of authentication relies on the address of the packet at the network level, packet with authorized address is supposed to be routed correctly to the destination. However, a vader can lie to a host about his address by changing the address in the packet of data sent to host. In TCP/IP protocol suite the address of communicating entity is easily forged, with duplicating IP numbers on a subnet with machines masquerading as the other machines.

### III. Trusted Host Authentication

If using Data Encryption Standard(DES) and the public key distribution, certificate hierarchy is the only trusted entity for identifies verification and public key acquisition. This relies ver much on a trusted root that everyone (at least the sender and receiver) believes it to be trustworthy. However, the trusted host can have duplicates on the network, thus authentication is still possible. On the other extreme, mutual trust among clients would exists if the clients themselves trusting each other can distribute public keys on their own.

### IV. Public Key Encrypted Authentication

This kind of authentication usually goes with the one-way hash function, used one time. For example Kerberos and SPX systems. For details, please refer to the public key encryption scheme discussed previously. Even it seems to be a secure enough approach, it also exposes weaknesses. But the details of it would not be covered in this research.

V. Biometrics Authentication

By using the personal physical features such as human retina or fingerprint, for authentication. However, the weak point is that it relies on a device that converts the physical feature of a person into bits. Moreover, this kind of information cannot be changed for the person, the system will no longer secure if the information is compromised by vader. Consequently, security relies on a secure communication media used for the transfer of the person's biometric password.

## **2.3 FIREWALLS**

With increasing number of companies connecting to the Internet, on-line security becomes more and more important. Firewalls were designed to protect the private networks from assaults and unauthorized access from the Internet. Because a firewall server functions in reducing the zone of risk to a single point of failure, it is designed to be the only door open to the Internet and data traffic must go through it in order to go to the Internet. As a result the firewall server becomes the bottleneck for any transactions and communications between the Internet and the LAN, also it is the entry point of any assaults from the Internet into the LAN.

### **2.3.1 Firewall Definitions**

Logically, a firewall is a separator, a restricter, an analyzer that are used to protected the internal network against any attack. We can image it as a castle used to prevent us from the outside attacks, or it is a blanket that protects use from fire. It mainly serves the following goals [28]:

- to restrict people to entering at a carefully controlled point;
- to prevent intruders from getting close to your other defenses;
- to restrict people to leaving at a carefully controlled point.

Because of the above purposes, a firewall is often installed at a point where the protected internal network connects to the Internet. All the traffic from the internal network is supposed to pass through the firewall. Basically it is a set of components that restricts access between a protected network and the Internet, or between other sets of network. When in physical implementations, there are many different configurations of firewall. As often as not, a firewall is composed of a set of hardware components such as a router or a computer, or some combination of routers, computers and networks with appropriate software installed. The specific firewall configuration for an internal network will depend a lot on the security policy, budget as well as the overall operations of a site.

Simply speaking, a firewall is a system, either software or hardware or both, that enforces access control policy between two networks. It is the manifestation of a company security policy [27].

### **2.3.2 Firewall components [23]**

It describes main the physical components of a firewall system.

#### **I. Screening Router**

A screening router is a basic component of most firewalls and it can be a commercial router or a host-based router with some kind of packet filtering capability. Typical screening routers have the ability to block traffic between networks or specific hosts, on an IP port level. Some firewalls consist of NOTHING more than a screen router between a private network and the Internet..

#### **II. Bastion host**

It usually is a computer running proxy software that is exposed to the world outside the internal network to be protected. A bastion can be used in all the firewall configurations except the 'screened network' in which a proxy server is not used [27].

Another kind of bastion host is called a victim machine (or a sacrificial lamb). A victim machine is the victim as all the communication or attacks are supposed to directed to it because it is the first machine for the internal network exposed to the outside world. Only the information that is supposed to share freely with anyone and only minimal service should be placed in the victim machine.

A bastion host is a system identified by the administrator of firewall as a critical strong point in the network's security, to keep intruders out of the internal network. Also the security of a bastion host is a matter of concern, it may undergo regular audits and have modified software.

#### **III. Dual Homed Gateway**

It is a system or host bastion placing between the private network and the Internet, and disabling TCP/IP forwarding. This kind firewall is implemented without a screening router. This system or host called a dual homed gateway, is, by definition a bastion host. The hosts on the private network, as well as the host on the Internet, can communicate with the gateway, but there is no direct traffic between the two networks [23].

#### **IV. Screened Host Gateway**

Screened Host Gateway is the most common type of firewall configuration. This is implemented by using a screening router and a bastion host. As often as not, the bastion host is on the private network and the screening router is configured such that the bastion host is the only system on the private network that is reachable from the Internet. The screening router is configured to block traffic to the bastion host on specific ports, allowing the authorized services to communicate with the LAN.

#### **V. Screened Subnet**

This is an isolated subnet is created and it is situated between the Internet and the private network. Typically a screening router which implement varying levels of filtering, is used to block the traffic across the screened subnet. A screened subnet is configured such that both the Internet and the private network can access to the hosts on the screened subnet, provided that the traffic from the networks can go through a screening router. In some firewall configurations, a bastion host will be added to the screened subnet to support interactive terminal sessions or application level gateway.

#### **VI. Application Level Gateway (or Proxy Gateway)**

Much of the software on the Internet works in a stored-and-forward mode such as mailers and USENET. Application level gateways are the service-specific forwarders or reflectors, which usually operate in user mode rather than at a protocol level. In fact, running this kind of forwarding service is important to the security of the whole. For example, the sendmail hole that was exploited by the Morris Internet worm is one of the security problems an application level gateway can present. Some kinds of applications gateway are interactive, such as the FTP and Telnet gateways, which run on the Digital Equipment Corporation firewalls. In general, the crucial applications level gateways are run on bastion hosts [23].

#### **VII. Hybrid Gateways**

This kind of gateway is somehow different from that mentioned above. For instance, the hosts connected to the Internet, but accessible only through serial lines connected to an ethernet terminal server on the private network. Such kinds of gateways may take advantage of multiple protocols, or tunneling one protocol over another. Routers might maintain and monitor the complete state of all TCP/IP connections, or examine traffic to try



to detect and prevent an attack. The AT & T corporate firewall is a hybrid gateway combined with a bastion host.

### 2.3.3 Firewall Technology

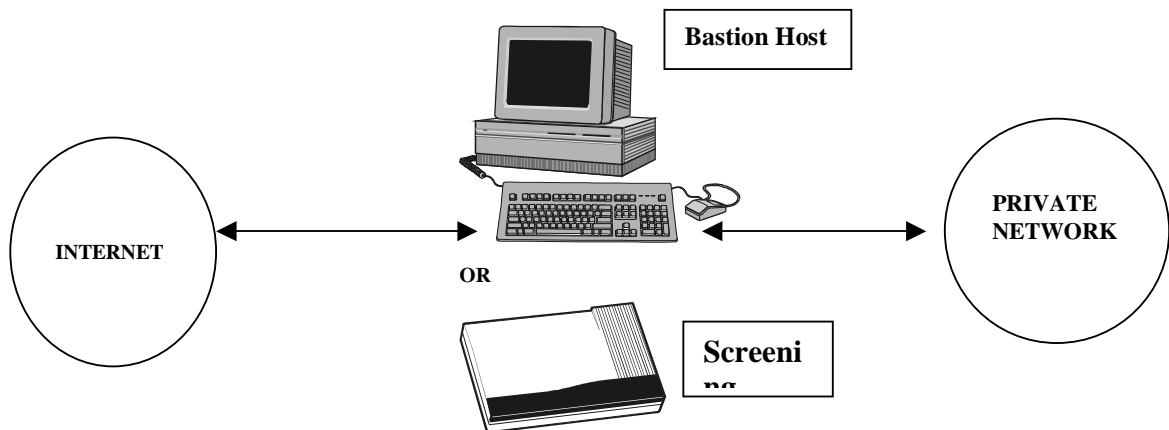
There are two main firewall technologies, they are packet filtering and application level proxy servers. Basically they differ in many aspects and are discussed in the following parts.

#### I. Packet filtering

Packet filtering is the process of allowing and denying any flow of traffic between two networks, based on the information found in the header of each data packet, such as the source/destination IP address and the port/service number. It is used in setting some rules to accept or deny the communications between two networks.

As often as not, it makes use of a packet filtering router (or packet filtering software running on a screening router or a computer) to control data transfer between internal network and the Internet. All traffic into and out of the internal network must pass through the router for data scanning. Usually we call the type of router, which is used in a packet filtering firewall as screening router.

*There is no direct traffic between the 2 networks, with a screening or a bastion Host in between*



*Figure 1: A typical packet filtering system, by using bastion installed with packet filtering software or a screening router*

The main information a router need for packet screening are:

- IP source address (found in packet header)
- IP destination address (found in packet header)

- Protocol ( if the packet is a TCP, UDP or ICMP Packet, found in packet header)
- TCP or UDP source port number (found in packet header)
- TCP or UDP destination port number (found in packet header)
- TCP ACK flag (use to indicate it the packet is the first packet in a connection or is a response to another packet, found in packet header)
- ICMP message type (found in packet header)
- The interface the packet arrives on
- The interface the packet will go out on

According to all the above information, we can do the packet filtering by source /destination IP address, by inbound or outbound service, by port and so on. The screening router will compare the header information with a table of rules set by the network administrator to determine whether or not to send the packet on to its destination. If no rules allow a packet to be sent, the router should discard the packet.

When configuring a router, we should always make it as simple as possible, The more complex the filtering router and its configuration are, the more likely that we will make mistakes in its configurations. When setting rules for packet filtering for a firewall, we should generally find out whether the purposes of the firewall is either:

- “permit any service unless it is expressly denied” or
- “deny any service unless it is expressly permitted”.

The latter one is safer and should be always applied if an internal network security is important.

When we have to set up the packet filtering rules, we may set up a table to illustrate the allowed or disallowed packet as the followings.

Rule	Action	Local Host	External Host	Local Port	External Port	Descriptions
1	Deny	!	Trouble-Host	!	!	Block packet form Trouble-Host
2	Pass	SMPT-Mail	!	25	>1023	Allow packets to our mail Gateway
3	Deny	!	!	!	!	Block everything else

Obviously, the 1<sup>st</sup> rule blocks all the packet coming from the trouble host, the 2<sup>nd</sup> rule allow the inbound connections from any external host using port above 1023 to the internal SMPT mail server at port 25. For all other cases not met by rule 1 and 2, connections from outsides will be blocked with the 3<sup>rd</sup> rule.

When we put the above rules into commands added in a screening router, for example, Cisco router, the rules will be set as the one described below:

Assume that the internal mail server is 132.23.60.0 and an external trusted host is 185.12.30.1, the external internal is “serial1”

*Rule 1: Allow inbound connections from an external trusted host to our mail server. Reject any others.*

```
access-list 101 permit ip 185.12.30.1 0.0.0.0 132.23.60.0 0.0.0255
access-list 101 deny ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
interface serial1
access-group 101 in
```

*Rule 2: Allow outbound connections from our mail server to the external trusted host. Reject any other outbound connections.*

```
access-list 102 permit ip 132.23.60.0 0.0.0255 185.12.30.1 0.0.0.0
access-list 102 deny ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255
interface serial1
access-group 102 out
```

*Rule 3: Deny all service that list connections in with the designated port numbers.*

```
access-list 101 deny tcp any any range 6000 6003
```

### ***A router or a computer with routing packages for packet filtering?***

In general, we can either implement packet filtering by using a single-purpose router or a general-purposed computer dedicated to routing and packet filtering.

If there is a large number of networks or multiple protocols to be handled, a single-purpose router is suggested. It is because the routing packages for general-purpose computer may not have the speed or flexibility to accommodate the necessary interface boards as a router does.

But when do we use a computer for packet filtering? It is used when we are filtering a single Internet link and we need no more than IP packet routing between two or three Ethernets. In this case, it will be more economical to use a cheaper computer installed with the routing and filtering packages. Some commercial firewall packages combine packet filtering with proxying on a machine which acts like a screening router. In addition, the packet filtering software had been included in Linux in the kernel since Linux version 1.3X.

### ***Static versus dynamic packet filtering [30]***

Static packet filtering is the first generation packet filtering, it is ‘static’ because any desired method of connecting between the internal and external network must be left open at all times to allow

desired traffic. It introduced the weakness of making static packet filters open to a wide range of attacks preying on the security of hosts on the internal networks.

Dynamic packet filtering is the second generation of packet filtering. It opens and closes doors in the firewall based on the header information in the data packet. Once a series of packets has passed through the door to its destination, the firewall closes the door. Clearly dynamic packet filter is incorporated with the enhancement to address the weakness of the static packet filter.

## II. Application level proxy servers

This is an application-level technology and the devices used are called “application gateways”. Application gateways are in fact, computers running proxy server software.

In common term, a proxy is one thing act on behalf of another thing. In a proxy system, the hosts that have access act as proxies for the machines that don't, doing what these machines want done. A proxy server is a software that acts on behalf of an application, to try to access or communicate from one network to another. Applications on both the internal and external network sides can communicate with the proxy server, but they cannot communicate directly.

With proxying, the user clients program talks to its proxy server instead of directly to the real server, which resides out in the Internet. The proxy server receives communications from one side, evaluate the request to make sure the communications is authorized to proceed. If the communication is an authorized one, the proxy server will initiate a connection to the communication's destination and relay the packets to the destination. However, a proxy system is only effective when they are used in conjunction with some method of restricting IP-Level traffic (such as screening router) between the clients and the real servers outside.

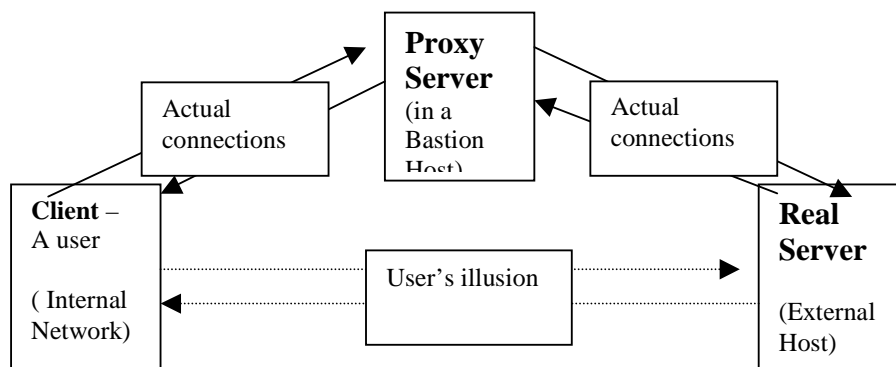


Figure 2: Proxy: the actual connection is from client – proxy server – real server; The illusion of the clients: the client – real server. [10].

Some services support proxying without a proxy server, especially for the “store-and forward” services such as SMTP, NNTP and NTP. For example, the email messages for SMTP are received by a server, then stored until they can be forwarded to another appropriate server or email messages’ destination. As a mail is usually sent through many intermediate servers (the mail gateways) between the source and destination mail servers, each of the intermediate servers act as a proxy server for the sender.

There are pros and cons of using proxy servers and they are listed as the followings.

*Pros of Proxying:*

- Allow users to access Internet services ‘directly’
- Good at logging

**Cons of Proxying**

- Lag behind between the introduction of service and the availability of proxying server for it.
- Require different servers for each service
- Require modifications to clients, procedures, or both
- Don’t work for some service
- Don’t protect from all the protocol weaknesses

Both type of the firewall technologies have well-known pros and cons. As seen from the above sections, they differ in many aspects such as ease of configuration, degree of encryption and so on. The comparison of these two firewall technologies is summarized in Appendix H for further reference.

### **2.3.4 Firewall Configurations**

A firewall can be configured as simple as using a screening router, or as complicated as setting up a screened subnet with internal and external routers. In fact, we would come up with different kinds of firewalls, which have unequal strengths and weaknesses by using the same components and arranging them in different configurations. Generally, there are four kinds of firewalls, one is called **packet filtering firewalls**, the other ones are **application-level firewalls**, **circuit-level firewalls** and **hybrid firewalls**.

Moreover, there are many different configurations of application-level firewalls, the specific configuration to be adopted depends on the level of integrity and security to be implemented for a LAN or Intranet. Here below we will mainly cover the basic and popular firewall configurations.

## Packet filtering firewalls

### - *Screened Network ( Firewalls using Screening Routers)*

This kind of firewalls uses only the screening router to achieve packet filtering, to let authorized communications and reject those unauthorized. Below is the typical firewall configuration using a screening router.

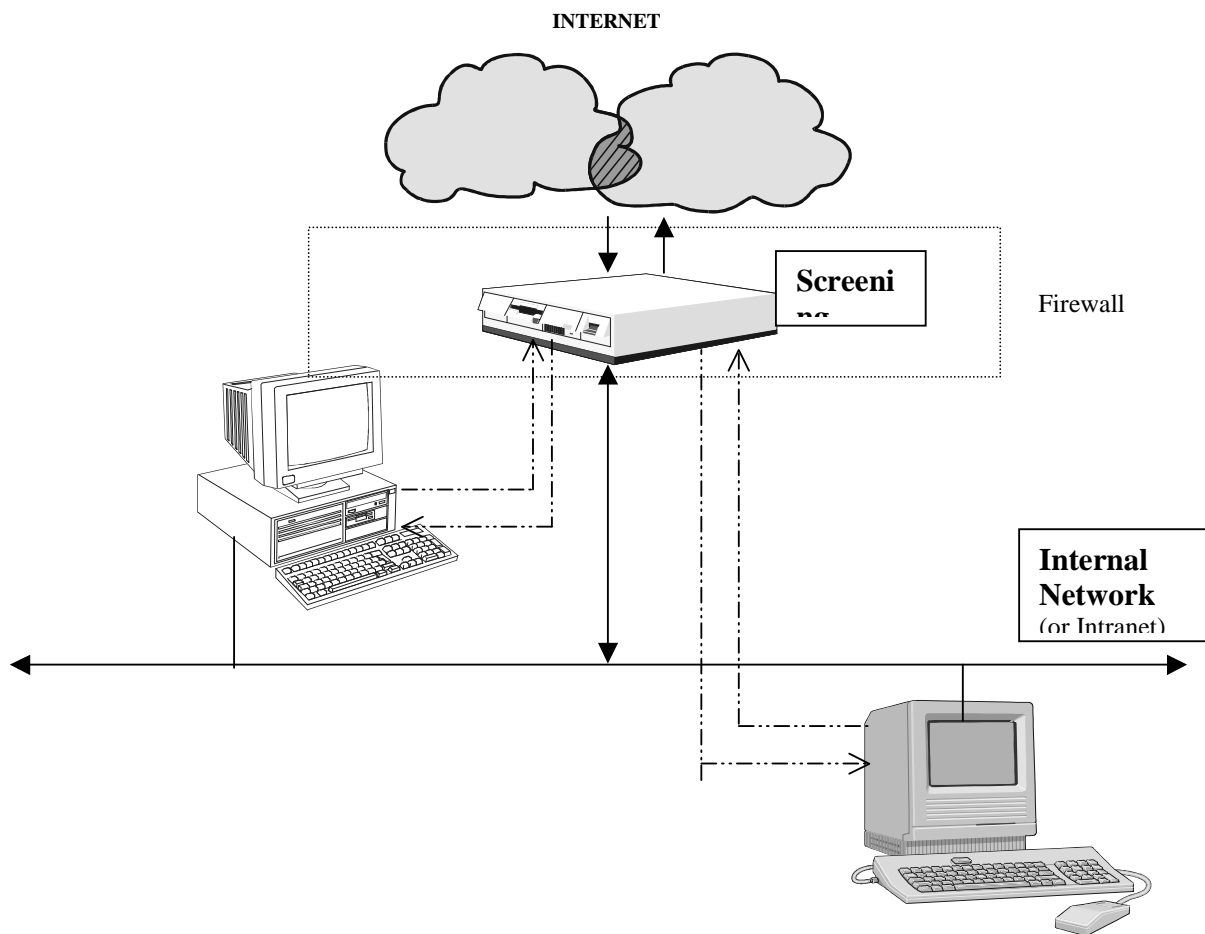


Figure 3: Firewall using screening router

With this kind of configuration, there is a direct communication permitted between multiple hosts on the internal private network, and multiple hosts on the Internet. In other words, any host on the private net side can open connections to any host on the Internet side as long as the connections satisfies the screening rules set up in the router which help to filter those unwanted communications. The internal network topology cannot be hidden from outsiders as each host can be accessed from any hosts on the Internet.

If the router's administrative password is compromised, the entire internal network is laid open to attack easily. In cases where screening router's screening rules are set up with errors without immediately attention from network administrator, the damage to the internal network is beyond expectation. As there is no logging capability, damage control is difficult. Network Administrators need to examine every host for traces of break-in regularly. It will be even harder to trace or discover the in case of total destruction of the firewall.

Even it is popular, it is not the most secure solution as it is permeable and permits quite free Internet access from any point within the internal network. It is not recommended in protecting the sensitive and secret information. It is suitable for small sites with easy screening rules, in which the internal network is supposed to be known to public for information sharing.

Zone of risk: number of hosts on the internal network, the number and type of services the screening router permits. For each service provided via peer-to-peer connection, the size of zone of risk increases sharply.

Pros & cons of this category of firewalls are listed as followings.

*Pros:*

- Simple to implement and relatively inexpensive
- Provide high level of performance
- Transparent to users

*Cons:*

- Vulnerable to attacks aimed at protocol higher than network level. As only the network level protocol is understood by it;
- More difficult to configure and verify, more opportunity for system mis-configurations, security holes and failures;
- Cannot hide the private network topology and therefore expose the private network to the outside world;
- Limited in auditing capabilities as well as logging facilities;
- Cannot support all the Internet applications with packet filtering firewalls (because some services are operated through protocol higher than network layer);
- Don't support some of the security policy's clauses such as user-level authentication and time-of-day access control.

### **Application-level firewalls**

It provides access control at application-level layer and acts as an application-level gateway between two networks. Because it is capable of working at application layer, it can examine the

traffic in details, resulting in a more secure firewall, even more secure than the packet filtering firewall.

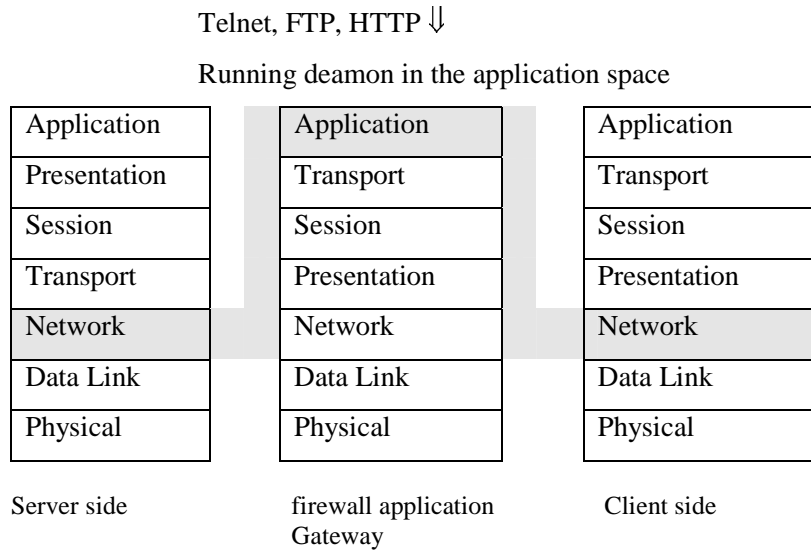


Figure 4: Application layer gateways working at application layer [14]

Application gateways examine all application layers and bring context information into the reject or accept decision process, thus improve on security.

It provides logging facilities to monitor the information such as source, destination network address, application type, user identification and password, size of information transferred, the start and end time of access and on. Also it may provide auditing tools to manipulate the log files.

Pros:

- Capable of defend against all attacks at application-level protocol;
- Much easier to configure than packet filtering firewalls as it don't require the knowing of details about lower level protocols;
- Hiding the private network topology;
- With auditing and logging facilities to get useful information for trace of attacks and audits;
- Supporting user-level authentication and time-of-day access control and many other security policies.

Cons:

- Slower than packet filtering firewalls due to the scrutiny of traffic;
- Intrusive, restrictive at certain extent. As it require to use specialized software , or to change user behavior to achieve policy objectives;
- Not transparent to users.



In fact, there are variations of application level firewall configurations. Only the most common types of application level firewalls are covered. The dual-homed gateway, the screened host and the screened subnet are regarded as the application-level firewalls and they will be described in more details below.

- **Dual Home Gateways**

This kind of firewalls is called a Dual Home Gateway because it is established with a dual home host computer which has at **least** two network interfaces. The dual home host can act like a router; it routes the data from one interface to another one. But the direct routing of IP packets from one interface to another interface is disabled in order to avoid the direct communication from the internal private network with the Internet.

The Dual-Homed Gateway would provide services by acting as a proxy server to provide application gateway such as telnet or FTP. Otherwise, it can allow users to log into the system of the dual-homed host directly to access the Internet. Here below is a typical configuration of a dual-homed gateway.

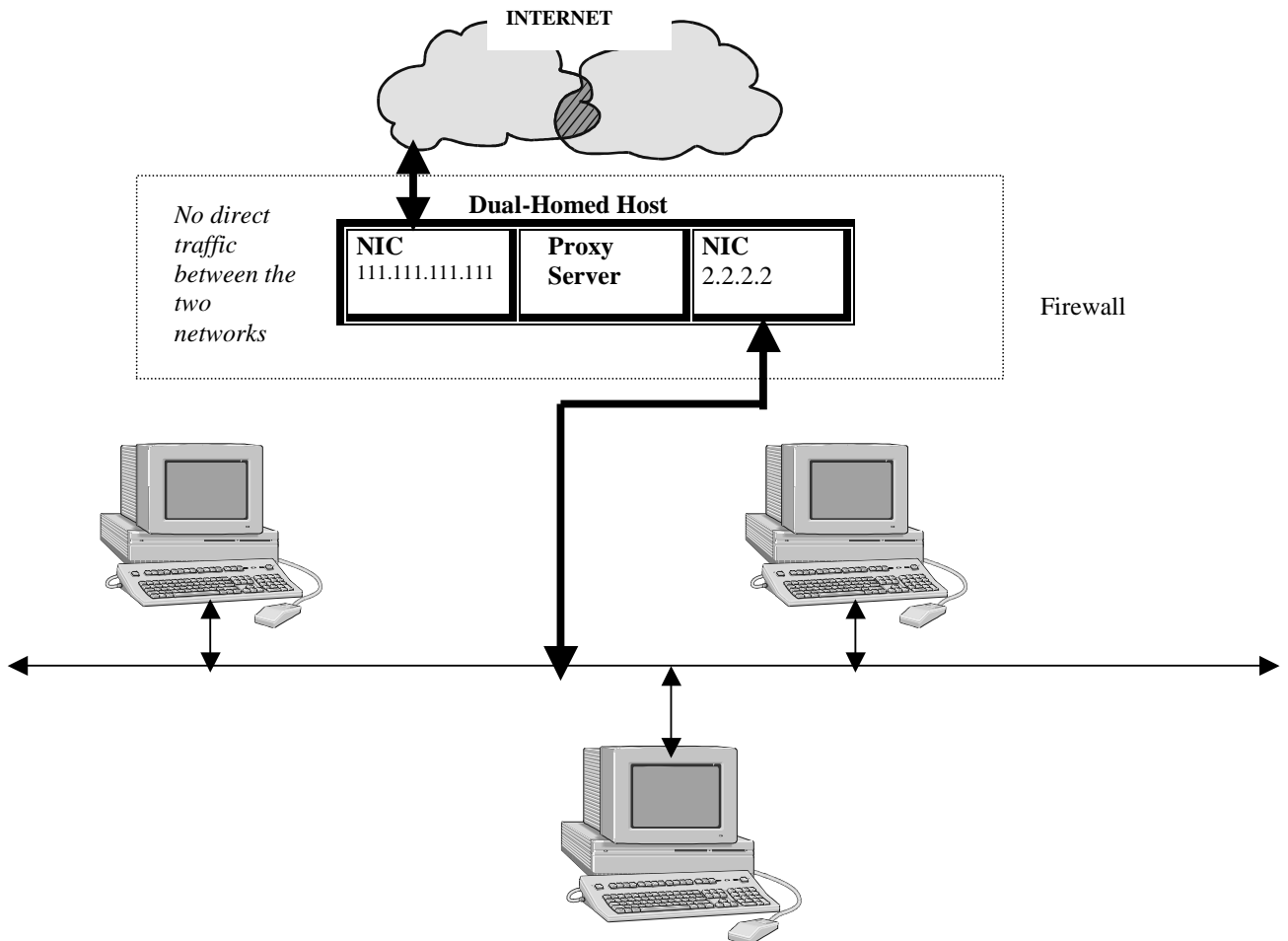


Figure 5: A typical Dual homed-host firewall

There are two network interfaces, 111.111.111.111 is connected directly to De-Militarized Zone (DMZ) which is in turn connected to Internet. The 2.2.2.2 is connected to the internal network. In other words, this kind of firewalls must have two IP network numbers.

There are two common alternatives to setup the firewall, as follows.

- I. A Bastion Host with two network cards, one network interface card (NIC) connected to the private LAN and the other one connected to the Internet. (As seen in Figure 5)

Private LAN --- NIC1 - Bastion Host - NIC2 --- - Internet

- II. A Bastion Host with one network card and a modem with PPP to the Internet

Private LAN --- NIC - Bastion Host - Modem <sup>PPP connection</sup> -----Internet

- Zone of risk:
- *the gateway host*, during normal operation, – since it is the only host reachable from the Internet.
  - *entire private network* if the firewall is destroyed, users accounts is compromised.

*Pros:*

- Separate the protected network from the outside world completely. Hide the names and IP addresses of the site systems from Internet system as with no DNS information being passed out from the internal network
- Provide logging capability that helps in detecting attack (proxy server)
- Use for authentication servers as well as proxy servers

*Cons:*

- Slower than packet filtering firewalls due to the scrutiny of traffic.
- Have problem if adding services which the proxy server cannot handle
- The gateway being the single point of failure if it is the only component of firewall
- All security lost if IP forwarding is enabled in cases such as operating system reinstallation, by human mistakes.

- ***Screened Host Gateways***

For a Screened Host Gateway, there us usually a bastion host and a screening router. The primary security from packet filtering is done with the screening router. The bastion host sits on the internal network. The screening router is configured in such a way that the outside hosts on the Internet can

only open connections to the bastion host and the bastion host is perceived as the only system on the internal network. The hosts on the Internet must connect to the bastion host in order to access the internal network or services. As a result, it is necessary that a high level of host security is maintained in the bastion host.

In real practice, the traffic from the Internet is directed to the screening router first. For those traffic satisfied the rules set in the screening router, it is forwarded to the bastion host or application gateway. All those traffic addressed to machines other than the application gateway, are rejected. For that permitted traffic arrived at the application gateway, the proxy server software on the gateway would examine the traffic again by using its own rules, and pass the permissible traffic to the internal network. For the application gateway, or the bastion host, only one network interface card is required to connect to the internal network.

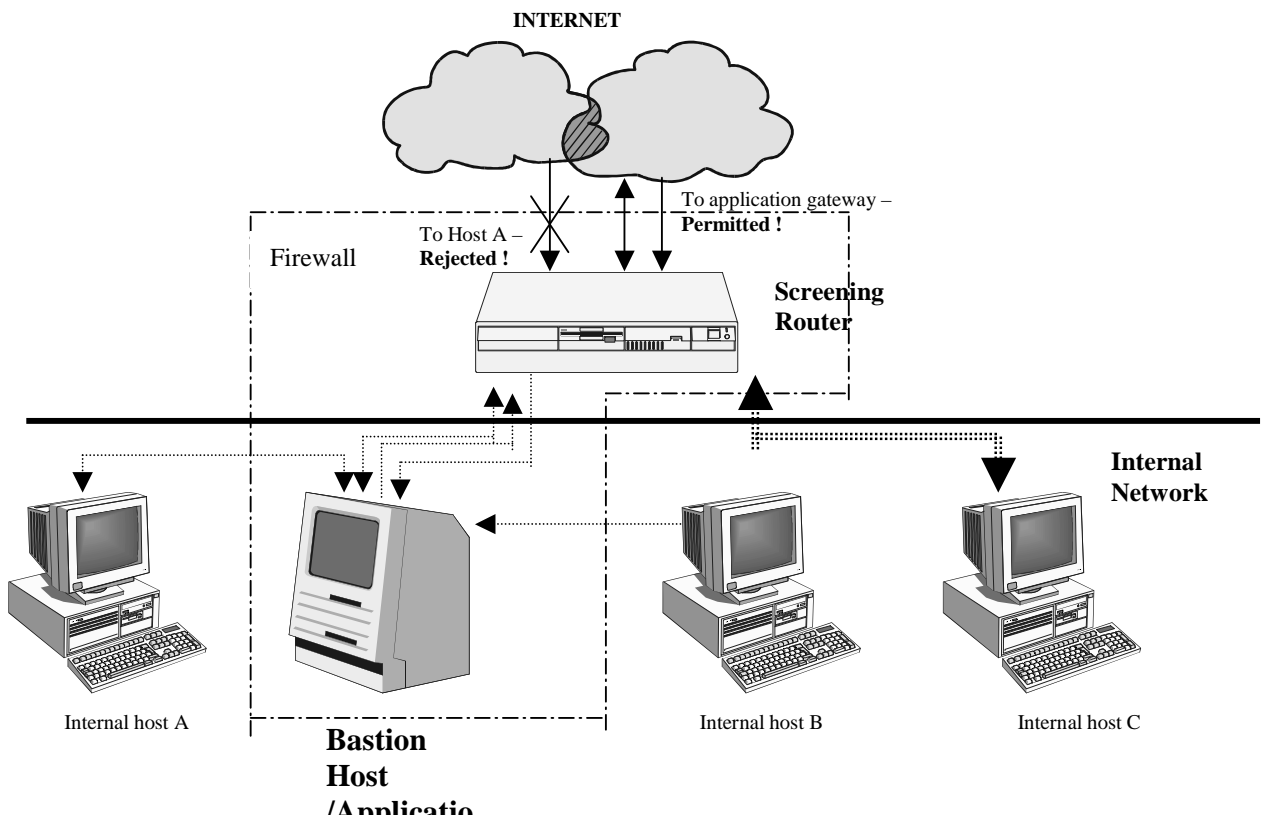

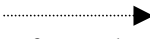


Figure 6: A screened Host Firewall

The configuration of this kind of firewall is slightly different with different packet filtering configurations in the screening router. Packet filtering configurations may adopt either one of the choices below.

- I.  It permits internal host such as the host C in the figure 6, to connect directly to hosts on the Internet for certain services through the screening router. It is used, because some services do not support proxy.
- II.  It forces the internal host such as the internal host A & B, to use the proxy services via the bastion host to connect to the Internet indirectly. It blocks all the connection requests from internal host to hosts on the Internet.

In fact, a screened host gateway can achieve a higher level of security than we could get with either a router only or the bastion host (or application gateway) only.

Zone of risk: the bastion host and the screening router

*Pros:*

- It is rather secure and easy to implement.
- If either component (the router or the application gateway) fails, the other component still affords some measure of protection.
- The rules of screening packets are less complex when compared with that for screened network configuration.

*Cons:*

- The screening router and the application gateway need to be carefully configured, in order to make them work correctly.
- As the system is so flexible that users may take shortcut to make connections directly to the routers to avoid proxy server. This led to impossible logging of such kind of traffic if the router is not capable of logging network traffic.

- ***Screened Subnet***

If the screened host architecture firewall is added with an interior screening router, it becomes a Screened Subnet Firewall. The external router, the bastion host together with and the interior router created a subnet, and are usually called the Demilitarized Zone (DMZ). This approach forces all the services through the firewall to be provided by applications gateways. Also it takes the advantage of routing to reinforce the existing screening.

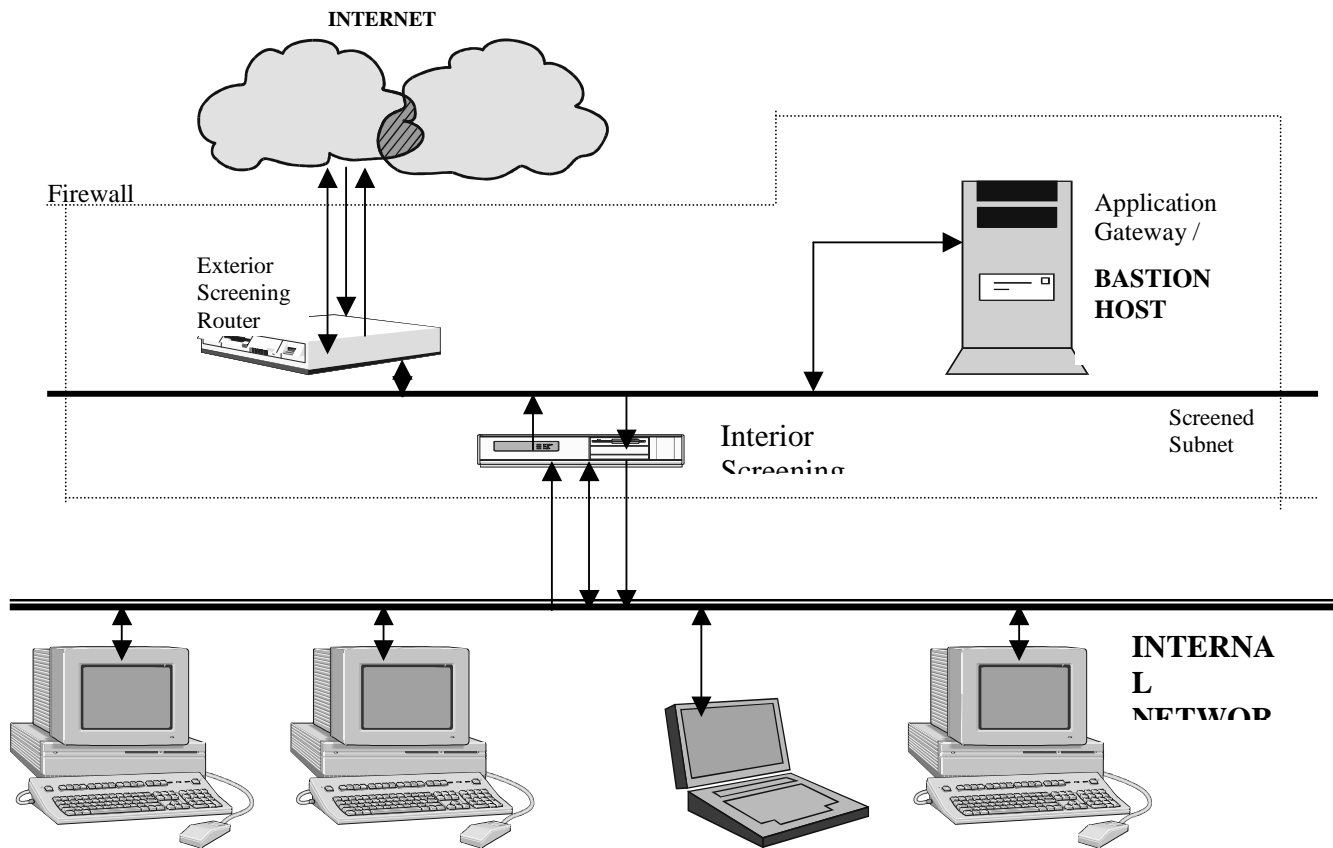


Figure 7: a typical screened subnet with interior and exterior routers

The exterior screening router together with the application gateway or bastion host functions like a screened host firewall. With an addition of an interior screening router, further protection between the application gateway and the internal network is ensured. It is because the internal traffic is still safe with the interior router in case hackers break into the exterior router and the application gateway only. This configuration makes the attack more difficult as intruders must manage to get through all the three protection layers before going into the internal network [28].

In order to make the protection of internal network more effective, the number of service allowed between the bastion host and the internal network has to be limited to just permit those are really needed to get into the network, for example, SMTP, DNS and so on. Further limitation can be imposed on allowing the services only to and from particular hosts, for instance, the SMTP would only be limited to connection between the bastion host and the internal mail servers. It supports the stance “which is not expressly permitted is prohibited” the best.

**Zone of risk:** It is small, with the bastion host or hosts and screening routers that make up the connections between the screened subnet, the internal network and the Internet.

*Pros:*

- It provides larger protection than other configurations of firewalls. If the screened subnet firewall with inter-networking routing blocked is attacked, the attacker must reconfigure the routing on the three network (the Internet, the screened subnet and the internal net), without disconnecting or locking himself out, and without routing changes being noticed. This is very difficult, although not impossible.

*Cons:*

- It's most expensive when compared with others configurations. The number of machines, routers, software modules involved is also larger than that in other configurations.
- The screening rules set for the two rules and the bastion hosts will be very complicated and not be easy to maintain.

### **Hybrid firewalls (Hybrid Gateways)**

In order to take the advantage of the packet filtering and application gateways, some vendor introduced hybrid firewalls that combine both packets filtering with application-level techniques[20].

However this kind of firewall still relies on packet filtering mechanism to support certain applications, it still incurs the same security weakness introduced by packet filtering firewall.

### **Circuit-level Firewalls**

This kind of firewall applies security mechanisms when a TCP connection is established. It validates TCP and, in some products validates TCP and, in some products, User Datagram Protocol (UDP) sessions before opening a connection or circuit through the firewall. Also it inserts generic transport-layer proxy into connection and there will have no further packet filtering after connection establishment [34]. But the state of the session is monitored, and traffic is only allowed while the session is still open.

This is more secure than packet filtering but allows any kind of data through the firewall while the session is open, creating a security hole. This is better than packet filtering but still falls short of

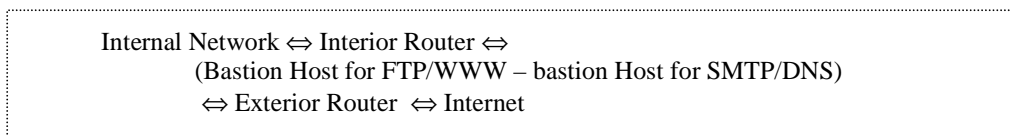
total security. If this gateway does not support UDP, it cannot support native UDP traffic such a domain name service (DNS) and SNMP [35].

### Other Firewall configurations

With all the basic firewall components and configurations in mind, we can come up with different variations on the common configurations of firewall to suite different company's security policy. For example using more bastion hosts to separate traffic for different services in the screened subnet.

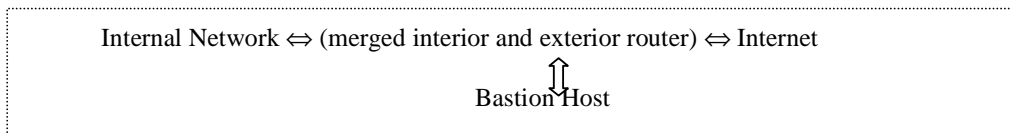
More common variation of the basic firewalls will be covered in followings[20].

#### I. More bastion Hosts (for a screened subnet)



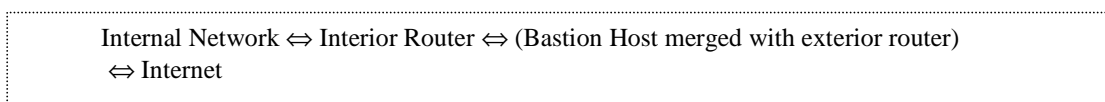
This can avoid the single point of failure of the proxy server. If one bastion host is unavailable or overloaded, the other one can switch to act as a fallback system for the activity of failed bastion host.

#### III. Merged the exterior and interior routers



With this kind of firewall, some traffic would flow directly between the Internet network and the Internet, the bastion hosts would handle the other traffic.

#### IV. Merged the bastion host and exterior router



As a matter of fact, a dual-homed gateway can be used as both the bastion host and the exterior router. For example, if the bastion host is connected to Internet via PPP connection or dial-up SLIP connection, the communication packages for PPP or SLIP would run on the dual-homed host. The dual homed host acts as both the bastion host and exterior router in this case.

However, it is no good to merge the interior router with the bastion host, as it would compromise the overall security.

**V. Use multiple exterior routers**

If there are multiple connections from the private net to the Internet through different service providers, or connections to Internet plus connections to other sites, an exterior router with multiple interfaces or multiple exterior routers, may have to be adopted.

**VI. Use multiple interior routers**

Unless several interior routers have to support several internal networks, multiple interior router should not be considered as it is difficult to configure and maintain the complex screening rules for multiple interior router

**VII. Have multiple perimeter networks**

Different perimeter network (the firewall) would be connected to the Internet and other supplier networks separately. In other words, separate firewall system is connected to different Internet Service Providers (ISPs).

**VIII. Use dual-homed host and screened subnet**



The security of private network would increase significantly if putting the dual-homed host and screened subnet together.

The exterior router still provides the first-hand packet filtering, the dual-homed gateway, would provide finer control on the connections than packet filtering. It provides multi-layered protection but requires rather complex and careful configurations.



## **2.3.4 Firewall Design, Implementations and other Considerations**

There are many design and implementations considerations we should take when a firewall is to be set up. For more references material and discussions, please refer to

Firewall Mailing List: <http://lists.gnac.net/firewalls/>

- Firewall FAQ: <http://www.cis.ohio-state.edu/hypertext/faq/usenet/firewalls-faq/faq.html> .

### **I. Internet Services to be configured at Firewall**

Once the firewall hardware and configuration are ready, we have to think about the service to be provided in the firewall and the LAN. A variety of Internet services are provided in the market and most of them are widely adopted in the Internet. However, an improper use of the Internet service would deteriorate the security of the whole internal network. So the services have to be configured properly in order to work together with the firewall, to make it work as safe as possible.

In fact for each kind of the Internet service, there are two ways to use. First, the service runs directly on a client in the internal network, through the screening router, to communicate with any hosts in the Internet. Second, the service can be a proxy service available at the proxy server for the private LAN. Users can approach the proxy servers for the service. Here below is the common Internet services provided for firewall system. For more information, please refer to [19,20].

- I. E-mail : Simple mail transfer Protocol (SMTP), Post Office Protocol (POP), and Multimedia Internet mail Extensions (MIME)
- II. File transferring issues : File Transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP), Unix-to-Unix Copy Protocol (UUCP)
- III. Terminal Access : Telnet
- IV. News : Network News Transfer Protocol (NNTP)
- V. The World Wide Web (WWW) and the HTTP Protocol
- VI. Information look up service : Finger, whois, talk, Domain Name System(DNS)
- VII. Timing service : Network Time Protocol (NTP)
- VIII. File System : Network during normal operation, during normal operation, File System (NFS)

## **II. Authentication and access control needed at the Firewall**

We can make use of the effective authentication methods and access control to provide access to authorized users and discourage those unauthorized network attacks.

Different authentication Mechanism such one-time passwords, time-based passwords, challenge-response schemes and encrypted password would be considered when setting up an effective authentication system for a LAN. Also the authentication server in TIS FWTK is also commonly used in enhancing the system security. For details, please refer to part I.

## **III. How to select a firewall?**

When constructing or purchasing a firewall, several major standards, many other aspects about further requirements have to be considered seriously in order to have the most suitable firewall for a company LAN. We have to ask how the following features the company security policy would like and how important the features are, to a network's functionality and good performance. Once the following characteristics of firewall are graded in priority, the future firewall characteristics are clear.

### **Security Assurance**

Is there the assurance that the relevant firewall technology fulfills its specifications and it is properly installed? Is the firewall product certified by the National Computer Security Association (NCSA – <http://www.ncsa.com/>) ? Does it have one Communication Security Establishment (CSE) evaluation?

### **Privilege Control**

The degrees to which the product can impose user access restrictions. How much access restrictions a user would be imposed on when using the firewall?

### **Audit Capabilities**

The capability to monitor network traffic, to get to know the unauthorized access attempts, to generate logs and to provide statistical reports and alarms.

### **Flexibility**

Is the firewall open enough to accommodate the security policy of the company? Does it allow changes in features and procedures in the light of the new Internet applications?

### **Performance**

The firewall product should be fast enough such that the screening of packets at firewall server transparent to users. The volume of data throughput, the transmission speed should be consistent to the company bandwidth to the Internet.

### **Scalability**

The firewall should be scaleable to adapt to the multi-platforms and instances within the protected network. The operating systems (OSs), machines and security configuration is to be modified anytime to adapt to any changes.

### **Ease of Use**

Ideally there is the Graphic User Interface (GUI) to facilitate the installing, configuring and managing tasks.

### **Transparency**

The more transparent the firewall is to users, the less confusing the firewall is to users, and the more likely users will support and use the product.

### **Customer support**

The extent to which a vendor supports customers needs, for example, providing prompt access to technical expertise or on-line help for technical solutions about firewall operations, and support for installation, use and maintenance,

## **IV. To build a firewall or set up the firewall using the firewall package outside**

For those company with rich IT expertise and internal resource, building a firewall with tailor-made security features for the special needs of the company would be the best choice.

For those company without in-house IT professional, they must have to outsource the set-up and maintenance of firewall to outside vendor.

Even for company with its IT people, an in-house firewall can be more expensive. If all the costs associated with building a firewall, in term of time required to build and document, to maintain, to add features as required, are added together, accounting may show that it is more economical to buy a firewall outside.

In addition to the dimensions mentioned in the above section about firewall selection, it is necessary to answer the questions as follows, before the decision to buy or purchase a firewall is made. These questions are suggested in [20]

- How will the firewall be tested?
- Who is responsible in verifying that the firewall performs as expected?
- Who will do the general maintenance of the firewall, such as backups and repairs?

- Who will install updates such as new patches, new proxy services to the firewall?
- Can security-related-problems be checked and corrected in a timely manner?
- Who will do the users support and training

If the answers for most of these questions are disappointed and the limited internal resource cannot satisfy the needs from building an in-house firewall, purchasing a commercial firewall is also a good alternative.

## **V. How to build a firewall?**

### *Issues to consider about*

- Physical security of the firewall / network
- Access control
- Authentication
- Encryption
- Security Auditing

### *General steps or guidelines for setting-up a simple firewall system:*

1. Select the hardware required
2. Install the necessary software (NOS and so on)
3. Connect and configure your machine on the network
4. Test it out
5. Add security (through firewalling software)
6. Set up and configure the proxy server

## **VI. How to select the type of firewall products on the markets?**

If you decided to buy a firewall product, there are many choices available in the market with various characteristics such different ease of administration, access control and degree of authentication. Some product even comes with intrusion detection system, security scanning system and log monitoring capability for easier firewall maintenance. A detailed research on the products had to be made to seek the one which best suite the company security policy and needs. There are books and many literatures talking about the various commercial firewall products published as books and found on the web. Please refer to [19,20,21] for more details about the selection.

## VII. How to maintain firewalls?

Here below are the points of maintenance that should be checked out regularly in order to keep the firewall working properly.

### Preventative and Curative Maintenance [20]

1. *Back up all the firewall components regularly*
2. *Be careful when adding new management accounts and services on the firewall*
3. *Watch the log reports of traffic passing through the firewall periodically.*
4. *Monitor the system to determine any attack or unexpected changes to the system.*
5. *Be alert for abnormal conditions of your firewall because they are the signals that the system may be under attack.*
6. *To Do list in case of an incident*
  - Don't panic, document everything if possible
  - Access the situation - to check if the identify of the attacker, if there is any damages to the system, the seriousness of the break-in, if the attack an inside threat, the current status of the hacker and so on, as soon as possible.
  - Cut off the link – to stop the intrusion if possible, and to do or not depends on your environment. However, can you afford shutting down the server, or shutting down some services only?
  - Analyze the problem – add up all the information you got, think carefully the action you are about to take and try to understand the problem. Hopefully you already identified the security hole or the root cause of the problem and will be fixing it. But make sure the fix of the problem won't create another security hole.
  - Take action – implement the emergency response plan if possible and if needed. If the problem cannot be rectified in a short period of time, advise a reasonable timeframe for the restoration and bug fixing of the system. Also notify CERT([info@cert.org](mailto:info@cert.org)) and exchange the information with them.
  - Catch the Intruder – even it is very difficult to do. Try to catch the hacker attack through shell script, logging facility.
  - Review Security to see if it requires any improvement and if there is any hole needed to be cover with the experiences from attack before.
7. *Recycle the firewall – to update and cover new services under the firewall*

### **2.3.5 Firewall security Policy**

The functions of a good and efficient firewall have to prove with an effective security policy. For instance: Some common policies are covered below.

- **Firewall design policy**

It adopts either the stance of:

Permit any service unless it is expressly denied or

Deny any service unless it is expressly permitted.

For the first policy, all services are allowed to pass into the Internet network by default, except for those determined to be disallowed. It exposes the private network to more threats coming from the bad services.

For the second policy, all services are denied by default, except for those services that was determined to be permitted. The system administrator gets more control about the services access.

- **Service access Policy**

It is concerned about the procedures and regulations of user access to the network resource, the dial-in policies and how users can effectively and correctly use the network services. It should strike a balance between protecting the private network and providing users access to the network resources.

- **Information Policy**

The LAN administrator or web master must determine if they intend to provide information access to the public. If the site can provide some information to public, a policy to determine the access to the server must be developed and included in the firewall design. Security on the information server is a big concern. It should not compromise the security of other protected sites that access the server.

- **Dial-in and Dial-out policy**

Remote access system would create big security threats if it were not under control.

The unauthorized access that a dial-in capability generated is a threat to the security of a site. A user's dial-out capability might become an intruder dial-in threat. This dial-in and dial-out capability must be considered in the firewall design. Any outside users must be forced to pass through the advanced authentication of the firewall before they can access the internal network resource.

- **Flexibility policy**

In fact, all the above policy must be flexible enough to meet the new services and changing risk faced on the Internet. However, a security policy almost never changes, but procedures should always be reviewed to incorporate any new environment changes and challenges.

### **2.3.6 Intrusion Detection System**

Intrusion detection is considered by many to be the logical complement to network firewalls [16]. Due to the failures of firewalls to adequately protect network assets from computer-based attacks, intrusion detection tools, i.e. Intrusion Detection System (IDS) had been developed to evaluate the degree of tolerance to intrusion and help in discovering vulnerability and various security problems of a system.

One may wonder how IDS could complement firewalls, which are supposed to be good enough in protecting a private network against outside attacks. Simply speaking, why do we need IDS? In fact, the function of firewalls is not really sufficient in protecting private network, the reasons are as follows [15].

1. Not all access to the Internet occurs through the firewall.  
Those "back doors" of a internal system, such as authorized modem connections between one outside system and the internal system, would pose risk and vulnerabilities beyond our imagination. Also firewalls cannot mitigate those risk associated with such back doors it do not aware of.
2. Not all threat originates outside the firewalls

Firewalls only examine traffic across the boundaries between the internal network and the Internet. If insiders make any security violation or attacks inside the internal network, there is no way for firewall to uncover it.

3. Firewalls are subject to attack themselves

There are some common attacks and strategies for circumventing firewalls, such as using tunneling to bypass the firewall protection. Tunneling is the encapsulation of a message in one protocol (which might be disallowed by firewall), inside a second message.

Most of the Intrusion Detection System (IDS) collect various information from target systems and networks, analyze the information for symptoms of security problems. It can also allow users to specify real-time responses to any malicious and destructive attacks. In general, intrusion detection and vulnerability assessments are two major tasks performed by IDS to achieve the goals of security.

### **2.3.7 Intrusion Detection Methods**

Intrusion detection techniques can be categorized into "misuse detection" and "anomaly detection".

**Misuse detection methods** attempt to model attacks on a system as specific patterns, then systematically scan the system for occurrences of these patterns. The process involves encoding any previous intrusion or malicious behaviors and actions. The cons of this approach is that it will only detect the attacks for which they are trained to detect. Novel attacks or variants of common attacks would easily go undetected [13]. Most of the commercial IDS are using this approach to scan for known attacks.

**Anomaly detection methods** assume that intrusions are highly correlated to abnormal behavior exhibited by either a user or an application. The basic idea is to baseline normal behavior of the object being monitored and flags behaviors that are significantly different from this baseline as abnormalities, or possible intrusions. Unlike misuse detection, this approach is capable of detecting novel attacks against software systems, variants of known attacks, and deviations from normal usage of programs regardless of whether the source is a privileged internal user or an authorized external user.



However, there are drawbacks for this method. First, well-known attacks may not be detected, particularly if they fit the established profile of the user. Once detected, it is often difficult to characterize the nature of the attack for forensic purpose. Also a malicious user who knows he or she is being profiled can change his or her profile slowly over time to essentially train the anomaly detection method to learn his or her malicious behavior as normal. The application of learning machine or neural network technology may be applied for training the detection algorithm. However, a high false positive rate may result for a narrowly trained anomaly detection algorithm, or a high false negative rate may result for a broadly trained detection approach. [13]

### **2.3.8 Vulnerability Assessment**

In addition to intrusion detection, vulnerability assessment is also carried out by most if the IDS. Unlike intrusion detection, vulnerability assessment is the process to determine any system weakness that might allow security violations. It seems to be a precaution measure taken to avoid any real occurrence of attacks and minimize damages due to attacks. Similar to intrusion detection, vulnerability assessment could be divided into two main strategies, the active and passive strategies for performing system examinations.

Passive strategy is host-based mechanisms that inspect system configuration files for unwise settings, system password files for weak passwords, and other system objects for security policy violations. While active strategy is in deed network-based assessment, which reenacts common intrusion scripts, recording system response to scripts [15].

The vulnerability assessment tools such as ISS, produce results about the situation of system security at a point in time. It functions not to reliably detect an attack in progress or any trace of attack happened in an internal system, but could determine whether a specific attack is possible or not.

## **3. METHODOLOGY**

### **3.1 Setting up firewall with different security levels**

In order to determine the impact from different security levels on network performance, different firewall security policies were proposed such that the firewall system for the project was qualitatively set up phase by phase with various security levels.

Starting with the four basic components of building a firewall, i.e. policy, advanced authentication, packet filtering as well as application gateway [20], the security of the firewall system is supposed to be built up, gradually in seven phases by using seven different firewall policies, under a constant experimental environment. During the process, some security as well as performance testing would be done on the firewall system of different security levels.

In other words, different security levels are defined by incorporating different firewall policies, security measures and firewall components. For this project, there is a total of seven configurations and security policies of firewall defined with seven different security levels. It's supposed that the more secured the firewall is, the poorer it performs. The security levels established in the project would be validated against the specifications stated in the security policies in security tests. The setup details are covered in the next section "Firewall configuration and policy setup".

#### **3.1.1 Hardware and software components**

##### **1. Router**

- This router would connect the firewall server (or proxy server under security policy 3 to 7) to the Internet (the department network). Basically, only traffic to the firewall server (pc89250) would be accepted, otherwise, the router would discard it.
- It restricts access from the Internet to the private network as well as internal access requests for using the Internet services.

##### **2. Firewall server**

- It is a Pentium 133 PC, with 32M RAM, connected to the protected network of 10Mb network throughput, and is directly connected to the router. The MTU (Maximum Transfer Unit) for it is 1500, a default MTU size for Ethernet.

- The operating system of it is Linux, and it is installed with Linux FWTK (Firewall Toolkit) package.
- It's also the application proxy server for almost all kinds of services with proxy available, such as TELNET, FTP, HTTP under firewall policy 3,4,5,6 and 7.
- It is the single entry point for outsiders from the Internet, to access the private network.
- It would route the traffic from the router to the internal network, and from the internal network to the router properly.
- There is only one NIC connecting to the internal network.

### **3. HOME – a Linux client beside the firewall server**

- It is a Linux PC called "HOME" located behind the firewall and inside the protected network. It is installed with X Windows, FTP/TELENT/HTTP and other client programs running on it.
- It is a Linux PC, installed with x-window and a program called “expect” for the ftp data transfer through the proxy server, such that the ftp and http proxy could be transparent to users when doing ftp data transfer in policy 3,4,5,6,7.
- Users are supposed to use the Internet services at HOME, freely and directly without going through any manual procedures to login firewall server.

### **3.1.2 Security zones in the testing network**

There are three security zones identified [18] as:

#### **Private Zone**

The internal area of the testing network which is protected from the Internet with a firewall. This is also called a private network.

#### **DNZ (De-Militarized Zone)**

This area is not secured by the firewall. Usually Internet servers are located here such as Web Servers, News Servers, DNS Servers and so on, as these servers are supposed to be accessible easily and could be rebuilt or reinstalled easily in case of being attacked and spoiled. However, this zone is not included in any of the testing for this project.

### Hostile Zone

It is in fact, the outside network, the cloud of the Internet. In this project, it is the department's network.

The FTP or HTTP requests would be made at 'HOME' to outside servers, through the firewall systems in the testing LAN. Here below is the testing bed setup for this project.

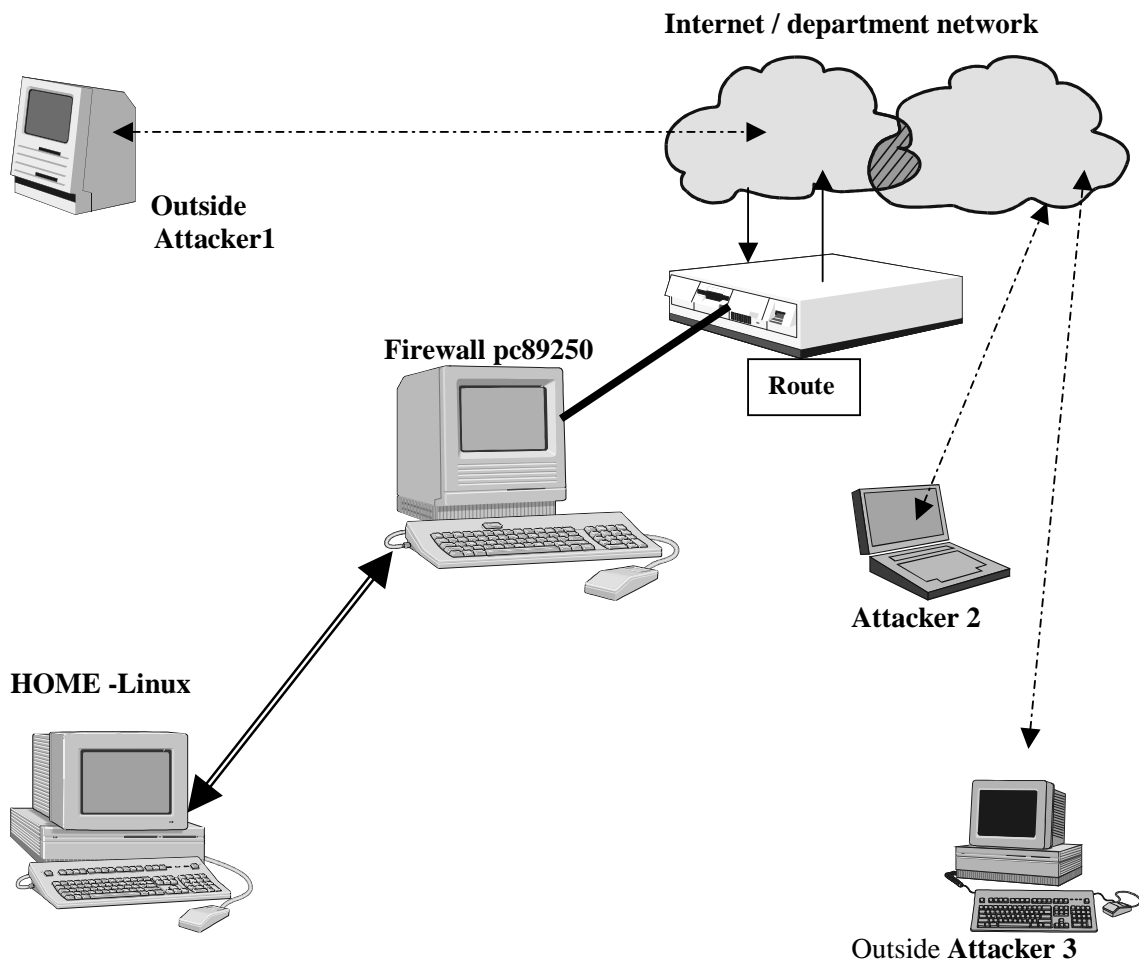


Figure 8: Test Bed Configuration

After the firewall system is incorporated with a certain security level, say level 1, two major kinds of testing would be carried out. The first type of testing is 'security and penetration testing'. The reason to have this testing is neither to determine if there are any attack or scanning activities on the firewall, nor to invent a new method to intrude the firewall. In fact, this kind of testing is to measure the security levels of the firewall system in the project, or in the other words, to validate the security level of the firewall systems and make sure the upper security levels are more secured than the lower levels in this project. The scanning reports together with the system information gained from some hacking techniques could tell us more than enough about how secured the firewall system was in the security testing LAN.

The second type of testing is 'performance testing'. With the firewall system defined with a specified security level, this type of testing is to quantitatively measure how the internal network performance, is affected.

### **3.2 Security testing**

Some security check-up and penetration testing would be applied in testing the security of firewall. Penetration test uses techniques designed to defeat and bypass security mechanisms in order to determine the effectiveness of such mechanisms. As a matter of fact, it is difficult to simulate the real network attacks with the testing LAN and the school network as the Internet. However, the vulnerability of a specified firewall setup to certain intrusion or attacks could be checked with network scanning tools or some techniques which intruders use for hacking and attacking firewall. Most of these attacks in fact, make use of a particular system weakness or vulnerability such as a system bug. As a result, the success of such network attack depends very much on the system and firewall setup correctness as well as the reliability of the running software at the firewall and the protected network.

Currently, there is a number of scanning tools available in the Internet (please check <http://sites.inka.de/lina/freefire-1/tools.html> for more details), some of the tools such as 'nessus', 'saint' would simulate real network attack and intrusion on the target system in order to break into the system. The following scanning and monitoring tools are very useful and have been employed (except 'tripwire) in this project, to check and ensure the security level of a particular firewall setup. Once any vulnerability is reported in a test phase, effort would be made in the next test phase to remedy and rectify the problem or potential security hole.

### 3.2.1 Network Scanning/Monitoring tools for Firewall Testing

Here below are some of the free Internet scanners adopted in playing attacks against the firewall, testing and ensuring the security level of each firewall policy is up to its specification stated in the firewall policies.

Moreover there are some more tools, which can be downloaded for the firewall testing, monitoring as well as intrusion detection. For more resources details, please refer to the following web address.

- ◆ <http://www.cs.purdue.edu/coast/firewalls/fw-body.html#testing>
- ◆ <http://www.sans.org>
- ◆ <http://www.geek-girl.com/ids/index.html>

#### A. For host attacking / scanning:

##### 1. SAINT - Security Analysis Tool (the updated version of SATAN)

It is the "Security Administrator's Integrated Network Tool" and remote network security auditing tool. In its simplest mode, it gathers as much information about remote hosts and networks as possible by examining such network services as finger, NFS, NIS, ftp and tftp, rexd, statd, and other services. The information gathered includes the presence of various network information services as well as potential security flaws—usually in the form of poor setup or incorrectly configured network services, well-known bugs in system or network utilities, or poor or ignorant policy decisions.

It reports and analyzes the data gathered from scanning and produce useful information of

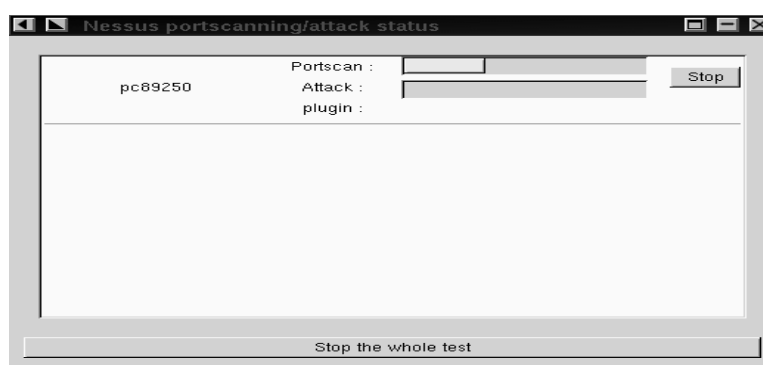
- 1. System vulnerabilities
- 2. Host Information
- 3. Trust (It follow the web of trust between systems, trust through remote login.)

which are very useful information for attackers to launch their attacks. Please refer to <http://www.wwdsi.com/saint/> for more information.

##### 2. Nessus

This is originated from a project called Nessus. It has been developed and started by Renaud Deraison of cvs.nessus.org with www site at <http://www.nessus.org/>. It is a free, open-sourced and easy-to-use security scanner as the aim of the Nessus project is to provide the Internet community a free, open-sourced and easy-to-use security auditing tool. The first version of it was released to the public on the 4th April 1998.

It performs port scanning and network intrusion on the targets. Below is the window pop-up when attack or port scanning from an outside host on the firewall pc89250 is in process.



Basically 'nessus' would check the target system for any vulnerabilities and try to simulate the real attacks, it's operations (or called plugins programs) can be classified as the following major categories. Please refer to Appendix D for more details about nessus.

**Major Nessus plugins:**

- CGI abuses
- Remote file access
- Denial of Service
- Misc
  - auth enabled
  - default system accounts
  - Services
  - FSP Daemon
  - guess operating system
  - HP Laserjet printer has no password
  - HP Printer Remote Print
  - icmp broadcast check
  - icmp netmask request
  - icmp timestamp request
  - HP JetDirect TCP/IP problems: display hack
  - HP JetDirect TCP/IP problems: single thread
  - lpd is active

- Motorola Cable router vulnerability
- QueSO - Guess the Remote Operating System
- rexecd check
- SSH Insertion attack
- Standard System holes
- TCP Chorusing
- TCP Sequence Prediction
- wingate
- X11-Checker
- Gain root remotely
- Backdoors
  - Finger backdoors check
  - Rootkit
  - Hidesource
- NIS
- Finger abuses
- Firewalls
- FTP
- RPC programs
- Sendmail tests

As at 1 August 1999, there has been 209 plugins. Please refer to the Appendix B for the list of plugins available.

## **B. For System security checking and monitoring**

### **1. COPS(Computer Oracle and Password System)**

It performs risk assessment, scanning on various aspects of system configurations. Then it produces a vulnerability report on the systems, it is somewhat like ISS (Internet System Scanner).

COPS is a collection of programs that each attempt to tackle a different problem area of UNIX security. The checking of COPS performs in the testing, are listed as follows. Please also refer to <http://www.fish.com/cops/> for more information.

1. file, directory, and device permissions/modes.
2. poor passwords.
3. content, format, and security of password and group files.
4. the programs and files run in /etc/rc\* and cron(tab) files.
5. existance of root-SUID files, their writeability, and whether or not they are shell scripts.
6. a CRC check against important binaries or key files to report any changes therein.
7. writability of users home directories and startup files (.profile, .cshrc, etc.)
8. anonymous ftp setup.



9. unrestricted tftp, decode alias in sendmail, SUID uudecode problems, hidden shells inside inetd.conf, rexd running in inetd.conf.
10. miscellaneous root checks -- current directory in the search path, a "+" in /etc/host.equiv, unrestricted NFS mounts, ensuring root is in /etc/ftpusers, etc.
11. dates of CERT advisories vs. key files.
12. the Kuang expert system. This takes a set of rules and tries to determine if your system can be compromised (for a more complete list of all of the checks, look at the file "release.notes" or "cops.report"; for more on Kuang, look at "kuang.man".)

As stated by COPS "this checks the dates that various bugs and security holes were reported by CERT against the actual date on the file in question. A positive result doesn't always mean that a bug was found, but it is a good indication that you should look at the advisory and file for further clues. A negative result, obviously, does not mean that your software has no holes, merely that it has been modified in SOME way (perhaps merely "touch"ed) since the advisory was sent out." As a matter of fact, all of the checks above only warns users about the existence of a potential problems, this tools would not correct or exploit any problems it finds.

Also this tools is effective in checking those common configuration problems which is likely to be made by human mistakes.

## **2. BSB Monitor**

This is a simple network monitor which scans the network periodically for an overview over the whole network and it can scan each TCP port e.g. database servers for its status. It produces an HTML formatted report about the status of the network services in use. Also it can send an SMS to a user's mobile phone when one of the critical services is down by using a script 'alerter' that acts as a gateway to a pager service. For more information, please refer to <http://www.bsb-software.com/download/bsb-monitor/> .

## **3. Tripwire**

-It has been developed by the COAST project and is a file integrity assessment tool, a utility that compares a designated set of files and directories against information stored in a previously generated database. This utility flags all the differences, including added or deleted entries. This is to ensure a set of files remain free of unauthorized modifications if tripwire reports no changes. Please refer to <http://www.tripwiresecurity.com/> for more details.

Besides, there are also some other useful network scanning tools available such as argus, but not included in this project. For more details, please refer to the web address <http://www.tripwiresecurity.com/index.html> .

### 3.2.2 Testing Procedures and Details

The network scanners would be run for each different firewall setup with a particular security level. Once any security hole or warnings, was found in a particular test phase incorporated with a specified security level, it would be eliminated or rectified in the next test phase by adding some more security controls such as screening rules to discard any problem traffic.

The seven security levels of the firewall are to be progressively incorporated and increased from level 1 to level 7 for testing. This is achieved theoretically by setting up different firewall policy and practically configuring the firewall system to achieve the requirements of the various firewall policies.

The testing on the firewall security is supposed to be a proof for validating a particular security level of the firewall system.

## 3.3 Performance testing

Performance tests would be done on the firewall to measure the relative performance degradation of mainly two services 'HTTP' and 'FTP' of the firewall. It would simulate real usage of the firewall by directing various loads of FTP and HTTP traffic through the firewall. The data transfer requests would be initiated inside the private network to a FTP or HTTP server in the hostile zone (the internet).

The performance would be evaluated by using one-performance indicator, "**latency**". Latency is the time required by a system to complete a single transaction, from start to finish [1].

The addition of data inspection at firewall would lengthen the time required for data communication, and thus increase the network or transaction latency. Experimentally, this indicator would be measured by executing a bunch of transactions sequentially in a single

thread and the result would be obtained by the taking the elapsed time used for processing each transaction.

In other words, latency of a transaction refers to the amount of time it takes to open 1 or more than 1 connection from the client to the server, request data from the server, download the data from the server to the client, and finally close the connection from the client to the server. If authentication is taken during the process of data download, the overhead due to it would be automatically included in the latency of a transaction. Transaction time or latency would be measured in second only for this paper. Strictly speaking, the transaction time is the total processing time including the transfer time, connection setup and tear-down time as well as any overhead for authentication if any, it may involve many NRTT (Network Round Trip Times).

The reason to have FTP and HTTP services included in testing is that only the service with relatively heavy traffic load would be considered in this performance test. As a result, the low-bandwidth service TELNET would not be included. E-mail, which is a store-and-forward service, would not be considered also due to its queuing nature and variability of mail protocol. For this project, only FTP and HTTP are the primary services to be measured.

Furthermore, as the testing would involve the connection from the internal network to the outside network, the experiments would be carried out during the period no body is using the testing LAN for testing, in order to minimize the risk of interference of internal traffic. Even though the interference from external traffic is difficult to control.

### **3.3.1 Tools**

A tool called "workload" together with some shell scripts would be adopted in synthesizing the desired traffic workload. For details about "workload", please find it in the archives of the firewall-performance mailing list, on <ftp.greatcircle.com> in /pub/firewalls-performance/digest/v01.n011.Z.

### **3.3.2 Assumptions**

It is supposed that the variables in the testing environment involved when evaluating the performance of the firewall are consistent and would not play an important role in the variance of some of the results. The variables are the bandwidth of the network, connection setup time on

initiating host and the receiving host, the workload of the testing machines, stray noise on the network and so on.

In fact, the testing is mainly to find out the difference of performance with different security level, but not to accurately measure the actual performance details about the network. As long as all the testing are carried out in the same testing LAN, using the same hardware, under the same environment, they could still be applied. As a result, the comparison of performance among different firewall configurations would still be valid.

### **3.3.3 Measurement**

Test scenarios are designed in the following sections for testing the firewall performance by using FTP and HTTP session tests.

During the experiments, data transfer requests would be issued at a ftp client, "HOME" located inside the private network. The requests would be passed to the firewall, which is responsible to communicate with the outside servers and contact the outside servers for the processing of data download requests.

If no proxy service is adopted, the clients beside the firewall could go directly into the outside network through the NAT (Network Address Translation) done by IP Masquerader. Also ftp and http requests could be initiated directly from the clients to outside servers for data transfer. This is the case when the firewall system is implemented with policy 1 and 2.

On the other hand, if the firewall is incorporated with proxy services for data transfer, the ftp and http requests would be handled differently. When using proxy server, the ftp-gw or http-gw proxy process for data transfer in the experiments of the project are used. This is the case when the firewall system is implemented with the firewall policy 3,4,5,6 and 7.

Clients beside the proxy server, i.e. the firewall, pass the ftp data transfer requests to the FTP proxy gateway and wait for the proxy server to pass the result back to them. As the proxy server becomes the middleman or agent between the service clients and the outside server, extra overhead for traffic handling is incurred. In this project, when testing with FTP protocol, a program called "expect" was used to automate the logon process to the proxy server for data download. Likewise, for HTTP data retrieval request, the firewall server would act as the http

proxy server for all the clients beside the firewall under firewall policy 3 to 7. But the clients can use the outside proxy server for http data transfer when under firewall policy 1 and 2.

At least 10 trials for each set of test scenarios were run and at least 3 valid set of data set would be used for analysis. The average and the best value (minimum for total time) for each test case would be considered in analysis. The highest or extreme values of result (which deviated from the other measurements of the same reference point) would be discarded with a view to minimizing the noise from the use of other users as well as interference from outside traffic.

### 3.3.3.1. HTTP session tests

Tests would be carried out to transfer large amount of data using HTTP to see how the firewall performs under different firewall policies. A simple HTTP session script is written to perform HTTP GET protocol requests. The HTTP tests examined the environment of high volume of connections and comparatively small data size in a transaction. Ten test events are designed and described as follows. Please note that there is 3 connection requests made in a transaction.

Event	No of sequential transaction(s)	No of sequential connections	Total Data size
1	1	3	395K ~ 0.38M
2	10	30	3.8M
3	20	60	7.6M
4	30	90	11.4M
5	40	120	15.2M
6	50	150	19M
7	60	180	22.8M
8	70	210	26.6M
9	80	240	30.4M
10	90	270	34.2M
11	100	300	38M

This design is to determine how the network performance would be affected when progressively larger and larger workload of traffic is to be handled from event one to event ten. To download the total of 395K document, 3 'GET' requests are made in event 1 because 3 GET requests are executed in a transaction. By the same token, 30 'GET' requests are made to download the 3.8M document in event 2.

For each of the above event, the workload of transferring data from an outside ftp site back to the private network would be synthesized for 'no. of sequential transaction' times sequentially. For

example, event 7 would run a script which uses "lynx" program to retrieve total document of 0.38M from an outside web site (say [www.cse.cuhk.edu.hk](http://www.cse.cuhk.edu.hk)) back to the private network in each transaction, and this script runs sequentially for 60 times. The starting time and ending time would be jotted down right before the data transfer is executed and after it is stopped running.

The total average and minimum values of transactions would be chosen for calculating the final result of network latency under a particular firewall security level. The results achieved under the seven different firewall levels would be compared.

### 3.3.3.2. FTP session tests

Test would be carried out to make FTP transfers. Bulk data transfer would be attempted and each bulk transfer involves 5M data in scenario A. Scenario B would try a smaller data size of 1M. In addition, the ftp tests examined the scenarios of low volume of data download and high volume of connections, i.e. scenario C. For the ftp tests, there is only 1 connection in each transaction executed sequentially. Test runs under various scenarios are described in the tables below.

#### Scenario A:

Event	No of sequential connections	Total document total size
1	1	5M
2	2	10M
3	3	15M
4	4	20M
5	5	25M
6	6	30M
7	7	35M
8	8	40M
9	9	45M
10	10	50M

Unlike the HTTP session test, the number of sequential connection is progressively increased by one only due to the large data size of 5M involved in each transaction of data transfer. The data files are placed in the pub directory of the ftp site of the university, i.e. [ftp.cs.cuhk.edu.hk](ftp://cs.cuhk.edu.hk).

#### Scenario B:

Instead of using such as large data size of 5M employed in scenario A, a smaller data size of 1M is used in this scenario.

<b>Event</b>	<b>No of sequential connections</b>	<b>Total document total size</b>
1	1	1M
2	2	2M
3	3	3M
4	4	4M
5	5	5M
6	6	6M
7	7	7M
8	8	8M
9	9	9M
10	10	10M

**Scenario C:**

An even smaller data size of 38.5Kbytes is used and more data connection would be attempted in the testing of this scenario.

<b>Event</b>	<b>No of sequential connections</b>	<b>Total document total size</b>
1	1	38.9K
2	5	194.5K
3	10	389K
4	20	778K
5	40	1.52M

The above three scenarios are used to determine if the data size, no. of connections of data transfer would influence the network performance under a particular firewall policy.

## **4. FIREWALL CONFIGURATION AND POLICY SETUP**

Seven firewall configurations would be attempted in order to have some testing on firewalls of seven security levels, implemented with seven different firewall policies respectively. They would be tested and compared with regards to performance and some other security related aspects. The hardware and software components of the testing LAN was mentioned in section 3.1.1.

### **4.1 Firewall policy and screening rules setup**

Theoretically, the firewall policies stated below could be to implement different security levels. The policy one is the least secured, policy 7 is the most secured. By the same token, the security implemented with policy  $x+1$  would be higher than that with policy  $x$ .

Practically during the testing of the project, once a particular security policy is set up, it would be checked to see if it could deliver all the expectation specified in the policy. The procedures of validating if the rules or services set into the firewall system is so trivial and would not be covered in details. For example, if the rule "deny icmp packets" is specified, the command "ping" executed on the firewall server from outside would result in failure. Starting from security level 1 to 7, the setups, proxy services and screening rules for each of the firewall policy were validated to ensure they delivered the expected security features and control, and thus the expected security level.

#### **4.3.1 Firewall Policy 1**

##### **i. Policy**

- Permit any service unless it is expressly denied
- Provide the maxi flexibility/access for both internal and external users.

##### **ii. Screening rules at router**

- Allow all other traffic from the Internet to destination with IP = firewall server.
- Allow access from internal network to the Internet

##### **ii. Proxy services**

- Nil



(There is the least possible protection from the router with setup 1.)

### 4.3.2 Firewall Policy 2

#### i. Policy

- Permit any service unless it is expressly denied (same as configuration 1)
- Disallow some problem service accesses from outside, but still provide flexible/easy access from outside, but no restriction on access from internal network to the Internet.

#### ii. Screening rules at router

- No ip source routing
- No ip spoofing (e.g. traffic from mail server to pc89180)
- Deny DNS(TCP) traffic from outside
- Deny TFTP(UDP) from outside to port 69
- Deny link (TCP) from outside to port 97
- Deny SunRPC(UDP) & NFS(TCP) from outside to port 111 & 2049
- Deny lpd(TCP) from outside to port 515
- Allow ALL others from outside to the pc89180 and email
- Allow ALL traffic from the internal network to outside

#### ii. Proxy services

- Nil

Other features,

- IP source routing is disabled in the Linux kernel.
- IP spoofing is prevented by the rules set into pc89250 as shown above.
- Disabling of the selected services is achieved by the rules set into the router as shown above.
- **IP Masquerader** is set up such that the workstations inside the private network could access the outside net, with IP being translated at the gateway.

### 4.3.3 Firewall Policy 3

#### i. Policy

- Permit any service unless it is expressly denied (same as configuration 2)
- An additional protection is added with 'proxy service' enabled in the firewall server. Specific traffic is further shielded and screened with the proxy server installed.
- Any traffic going into the private network would be pre-screened at the router first, then it would be passed into the proxy server for further authentication and screening. Security level is raised because both the router and proxy server examine network traffic.

#### ii. Proxy services

- TELNET/FTP/HTTP/RLOGIN

### 4.3.4 Firewall Policy 4

#### i. Policy

- Permit any service unless it is expressly denied (same as configuration 1)
- Allow even more restricted access from outside, and deny from selected bad HOSTs from outside.

#### ii. Screening rules at router

- No ip source routing
- No ip spoofing (e.g. traffic from mail server to pc89180)
- Deny DNS(TCP) traffic from outside
- Deny TFTP(UDP) from outside to port 69
- Deny link (TCP) from outside to port 97
- Deny SunRPC(UDP) & NFS(TCP) from outside to port 111 & 2049
- Deny lpd(TCP) from outside to port 515
- Deny openwindows (TCP & UDP) from outside to port 20
  
- Deny X Windows (UDP & TCP) from outside to port 6000

- Deny ICMP traffic from outside
  
- Deny outside bad HOST access from a range of IP
  - 137.189.88.128 - 137.189.88.191 (sparc28 - sparc91) for testing purpose
  - 137.189.88.65, (solar25) for testing purpose
  - 195.92.23.251, (block from desired sites )  
195.92.23.250,  
208.232.1.130,  
208.232.1.127,  
207.44.192.2,  
209.133.111.124,  
209.235.107.136,  
207.89.178.\*,  
12.10.107.5,  
199.60.229.31,  
203.85.221.120

but these IP would not conflict with any outside servers which the firewall depends on)

- Allow ALL other TCP/UDP traffic from outside to the pc89180 and email (ONLY TCP/UDP traffic, deny all others)
  - Allow ALL traffic from the internal network to outside
- ii. Proxy services**
- TELNET/FTP/HTTP/RLOGIN

#### **4.3.5. Firewall Policy 5**

**i. Policy**

- DENY any service unless it is expressly permitted. (or we say "that is not expressly permitted is prohibited")
- Deny all access from outside by default, but allow access from inside and provide the best possible services to the internal network, by permitting some selected services going into the network.

Although this policy is different from the previous one, it is expected that it could implement the same protection as the previous one and so the rule setting would be different.

**ii. Screening rules at router**

- No ip source routing
- No ip spoofing
  
- Permit ALL traffic from private network (internal hosts) to outside
- Deny ICMP traffic from outside
  
- Permit ANY TCP traffic from outside to port :
  - ◊ 97,
  - ◊ 111,
  - ◊ 2049,
  - ◊ 515,
  - ◊ 20,
  - ◊ 6000

(Permit TCP excluding link, SunRPX, NFS, lpd, & openwindows and x-windows)

- Permit ANY UDP traffic from outside to port
  - ◊ 69,
  - ◊ 111,
  - ◊ 20,
  - ◊ 6000

(Permit UDP excluding TFTP, SunRPX, openwindows, x-windows)

- Permit any outside hosts access from a range of IP EXCLUDING the bad ones as :(the following host IPs should be the ones used before)

- 137.189.88.128 - 137.189.88.255 (sparc28 - sparc91) for testing purpose  
195.92.23.251, (block from desired sites )  
195.92.23.250,  
208.232.1.130,  
208.232.1.127,  
207.44.192.2,  
209.133.111.124,  
209.235.107.136,  
207.89.178.\*,  
12.10.107.5,  
203.85.221.120
  
- Deny All other traffic from outside (make sure NO unwanted traffic entering the internal network)

**ii. Proxy services**

- TELNET/FTP/HTTP/RLOGIN

### 4.3.6. Firewall Policy 6

**i. Policy**

- DENY any service unless it is expressly permitted.
- A more restricted policy to permit outside access to certain port numbers range only.

**ii. Screening rules at router**

- No ip source routing
- No ip spoofing
  
- Permit TCP traffic from outside at port < 1024 to pc89250 at port < 1024 (Permit BSD 'r' commands, rlogin , rsh ..)
- Permit TCP traffic from outside at port >1023 to pc89250 only at port 23, 24 (Permit TELNET to port 23 and 24 only)

- Permit TCP traffic from outside at port > 1023 to only port 20,25)  
(Permit incoming FTP/SMTP traffic from outside)
- Permit TCP from outside at port > 1023 to pc89180 at port 80  
(Permit HTTP from outside at port > 1023 to pc89250 at port 80)
- Permit IP from outside to port 3001, 3002 (reserved from future use)
- Permit NTP(UDP) traffic from outside at port > 1023 to pc89180 at port 123
  
- Permit DNS(TCP+UDP) only from host "beryl, i.e. 137.189.89.250" to port 53
  
- Permit any outside hosts access from a range of IP EXCLUDING the bad ones as :
  - 137.189.88.128 - 137.189.88.255 (sparc28 - sparc91) for testing purpose
  - 195.92.23.251, (block from undesired sites )
  - 195.92.23.250,
  - 208.232.1.130,
  - 208.232.1.127,
  - 207.44.192.2,
  - 209.133.111.124,
  - 209.235.107.136,
  - 207.89.178.\*,
  - 199.60.229.31
  - 12.10.107.5,
  - 203.85.221.120(If these IPs are blocked at the router, the internal users could no longer access the web sites of the bad ip addresses above.)
  
- Deny All other IP traffic from outside
- Allow ALL traffic from the internal network to outside
  
- iii. Proxy services**
  - TELNET/FTP/HTTP/RLOGIN

#### **4.3.7. Firewall Policy 7.**

**i. Policy**

- DENY any service unless it is expressly permitted.
- Provide the least flexibility and services to the internal users, but incorporate maximum protection on the LAN. The internal users are no longer freely access any Internet services as some users are restricted to access of authorized hosts.

**ii. Screening rules at router**

- No ip source routing
- No ip spoofing
- Deny ICMP traffic from outside
- Permit ANY TCP traffic from outside to port :
  - ◊ 97,
  - ◊ 111
  - ◊ 2049,
  - ◊ 515,
  - ◊ 20,
  - ◊ 6000(Permit TCP excluding link, SunRPX, NFS, lpd, & openwindows and x-windows)
  
- Permit ANY UDP traffic from outside to port
  - ◊ 69,
  - ◊ 111,
  - ◊ 20,
  - ◊ 6000(Permit UDP excluding TFTP, SunRPX, openwindows, x-windows)
  
- Permit any outside hosts access from a range of IP EXCLUDING the bad ones as stated in the previous firewall configuration (policy 6).
  
- Permit authorized outside hosts to pc89250, they are:
  - 137.189.88.65, (solar15)
  - 137.189.88.73, (solar23)
  - 137.189.88.153 (sparc53)
  - 137.189.88.154 (sparc54)

137.189.91.165, (venture)  
137.189.91.190, (cucs18.cse.cuhk.edu.hk)  
137.189.90.151 - 137.189.90.159 (linux1 to Linux9)  
137.189.91.189, (sapphire )  
137.189.91.188, (garden)  
137.189.91.187, (beryl.cse.cuhk.edu.hk)  
137.189.91.192, (www.cse.cuhk.edu.hk / fortress)  
137.189.89.136, (pc89136)  
137.189.91.191, (obsidian.cse.cuhk.edu.hk - FTP server)

For Testing:

137.189.172.198, (www.jlm.cuhk.edu.hk)  
143.89.40.4, (www.cs.ust.hk)  
137.189.6.37, (www.cuhk.edu.hk/spring.csc.cuhk.edu.hk)  
147.8.179.15, (www.cs.hku.hk)  
144.214.5.246, (www.cityu.edu.hk)

- Deny All other IP traffic from outside  
(so maybe need to delete 'access-list 101 permit ip any host 137.189.89.250')
  - Deny TFTP (UDP) from pc89250 at port 69 to outside
  - Deny TELNET from pc89250 to linux6 (137.189.90.156) - for testing
  - Allow ALL traffic from the pc89250 to outside
- ii. **Proxy services :**
- TELNET/FTP/HTTP/RLOGIN

#### 4.3.8 Summary of firewall configurations

In short, the security features incorporated in the firewall system so as to realized the pre-defined security level could be up summed up as follows.

Security level /	No of screening	Proxy services	Additional authentication with	Flexibility to users/ Ease of access from	Security Level
------------------	-----------------	----------------	--------------------------------	---	----------------



<b>Policy</b>	<b>rules set into the router</b>	<b>available at firewall ?</b>	<b>proxy services used by clients beside the firewall?</b>	<b>outside</b>	
1	0	✗	✗	The most flexible	The least secured
2	7	✗	✗	Less flexible than 1	More secured than 1
3	7	✓	✓ for FTP, Telnet..	Less flexible than 2	More secured than 2
4	26	✓	✓ for FTP, Telnet..	Less flexible than 3	More secured then 3
5	29	✓	✓ for FTP, Telnet..	Less flexible than 4	More secured than 4
6	37	✓	✓ for FTP, Telnet..	Less flexible than 5	More secured than 5
7	43	✓	✓ for FTP, Telnet..	The least flexible	The most secured

## **5. ANALYSIS OF RESULTS AND DISCUSSIONS**

### **5.1 Security Testing**

In this project, the security level of the firewall system are divided into 7 categories, each level is implemented with a particular firewall policy for security control. The particular details of each of the seven firewall policies are described in section 4. The security level implemented by the firewall policy were tested and validated against those stated in the policy and checked to see if any other security holes found. Even the security policies defined in the project, are implemented with the qualitatively different security levels, it still needs some security testing to quantify the security level and validate the result system of a particular firewall policy.

Starting from the lowest security level, security testing helps in uncovering any security problems or vulnerabilities found in the security level. For any problem found, it would be solved by incorporating more security measures, when proceeding to the next few security levels. Thus the security testing in this project, in fact, helps in improving the security of the firewall system, from one security level to another one.

All the port scanning, attacks run by network scanners were performed on a Linux PC (either Linux1 or pc89136) located in the outside external network. It is supposed that attacks from outside of the private network can more reliably tell us how difficult the outside attackers can intrude the private network.

In fact, each type of the network scanner reports specified the result of scanning and attack on the firewall, and any vulnerabilities or warnings found. So when combining all the results gained from running different network scanners used in the project, it is no longer difficult to understand why intrusion can be possible when the firewall is not secured.

Here below is the specific information adopted when running the network scanners.

- SAINT,

The 'Heavy+' scanning level was used when scanning on the firewall under every firewall configuration. Please refer to Appendix E for the type of attacks and scanning details and results.

- NESSUS

All the attacks and port scanning options were selected when running the network scanners. Please refer to Append D for more detailed about the nessus reports after running the scanner targeted on the firewall pc89250.

- BSB - Monitor

It helps in the monitoring the services of the firewall. The details of it would not be covered. Please refer to Appendix E for the snap shots of its operation and details.

- COPS

As the firewall configuration was changed frequently from one policy to another one, mistake would be easily made when modifying firewall settings. It mainly runs on the firewall and helped to check if there was any problem with the configuration of the firewall. Please refer to Appendix F for the log files and results. So far no problems were reported when running COPS the firewall implemented with the various firewall policies.

### 5.1.1 Summary of results

After the security testing and checking on the firewall system in different phases, the results are summarized as follows. (note: vul means vulnerability)

<u>Security Level</u>	<u>Policy Used</u> (pls refer to sec 4)	<u>Vulnerability Detected / action taken as remedies</u>	<u>Security warnings ./ result</u>	<u>Vul + warning counts</u>
<b>Level 1</b>	1	<p>SAINT: Excess finger info</p> <p>SAINT: Vulnerable services – Sendmail gives out information using EXPN / VRFY</p> <p>NESSUS:</p> <ul style="list-style-type: none"> <li>- Security holes - rlogin service is activated.</li> <li>- Security holes rsh services is activated.</li> </ul> <p>NESSUS: Security holes - the firewall host answered to an icmp ECHO query, which is not a good thing.</p> <p>Potential Risk : IP spoofing, IP source routing and Syn flood attack (as they</p>	<p>SAINT: 4 trusted hosts - the DNS servers (berly, sapphire, garden, cucs) and their vulnerable services are identified</p> <ul style="list-style-type: none"> <li>- The operating system of the remote host appears to be some kind of UNIX</li> <li>- Sendmail supports the EHLO greetings and ESMTP and aloe EXPN command- can be a security flaw.</li> <li>- The firewall host answered to an icmp TIMESTAMP request. This will give away the remote host current time to an attacker, and may help him to bypass time based authentication protocols.</li> </ul>	10

		are not avoided by setting rules in the router)	<ul style="list-style-type: none"> <li>- The services running at the firewall can still be detected e.g. ftp(21/tcp). Also some unknown port numbers e.g. (3464/tcp, 5311/tcp) were reported</li> </ul>	
<b>Level 2</b>	2	<p>SAINT: Excess finger info</p> <p>SAINT: Vulnerable services – Sendmail gives out information using EXPN / VRFY</p> <p>NESSUS:</p> <ul style="list-style-type: none"> <li>- Security holes - rlogin service is activated.</li> <li>- Security holes rsh services is activated.</li> </ul> <p>NESSUS: Security holes - the firewall host answered to an icmp ECHO query, which is not a good thing.</p>	<p>SAINT: 4 trusted hosts - the DNS servers (berly, sapphire, garden, cucs) and their vulnerable services are identified</p> <p>NESSUS:</p> <ul style="list-style-type: none"> <li>- The operating system of the remote host appears to be some kind of UNIX</li> <li>- Sendmail supports the EHLO greetings and ESMTP and aloe EXPN command- can be a security flaw.</li> <li>- The firewall host answered to an icmp TIMESTAMP request. This will give away the remote host current time to an attacker, and may help him to bypass time based authentication protocols.</li> <li>- The services running at the firewall can still be detected e.g. ftp(21/tcp). Also some unknown port numbers e.g. (3464/tcp, 5311/tcp) were reported/</li> </ul>	9
<b>Level 3</b>	3	<p>SAINT: Vulnerable services – Sendmail gives out information using EXPN / VRFY</p> <p>NESSUS:</p> <ul style="list-style-type: none"> <li>- Security hole - rlogin is activated.</li> <li>- Security hole - rsh services is activated.</li> </ul> <p>NESSUS: Security holes - the firewall host answered to an icmp ECHO query, which is not a good thing.</p> <p><b>Action taken:</b> Deny ICMP traffic from outside, such that 'ping' on firewall would result in failure by setting 1 more screening rule into the</p>	<p>SAINT: 4 trusted hosts - the DNS servers (berly, sapphire, garden, cucs) and their vulnerable services are identified</p> <p>NESSUS:</p> <ul style="list-style-type: none"> <li>- The firewall host answered to an icmp TIMESTAMP request. This will give away the remote host current time to an attacker, and may help him to bypass time based authentication protocols.</li> <li>- Sendmail supports the EHLO greetings and ESMTP and aloe EXPN command- can be a security flaw.</li> <li>- The auth service provides</li> </ul>	7

		router.	<p>sensitive information to intruders: it can be used to find out which accounts are running which servers.</p> <ul style="list-style-type: none"> <li>- The services running at the firewall can still be detected e.g. ftp(21/tcp). Also some unknown port numbers e.g. (3464/tcp, 5311/tcp) were reported/</li> </ul>	
<b>Level 4</b>	4	<p>SAINT: Vulnerable services – Sendmail gives out information using EXPN / VRFY</p> <p>NESSUS:</p> <ul style="list-style-type: none"> <li>- Security hole - rlogin is activated.</li> <li>- Security hole - rsh services is activated.</li> </ul>	<p>SAINT: 4 trusted hosts - the DNS servers (berly, sapphire, garden, cucs) and their vulnerable services are identified</p> <p>NESSUS:</p> <ul style="list-style-type: none"> <li>- The auth service provides sensitive information to intruders: it can be used to find out which accounts are running which servers.</li> <li>- The services running at the firewall can still be detected e.g. ftp(21/tcp). Also some unknown port numbers e.g. (3464/tcp, 5311/tcp) were reported/</li> <li>- Sendmail supports the EHLO greetings and ESMTP and aloe EXPN command- can be a security flaw.</li> </ul>	6
<b>Level 5</b>	5	<p>SAINT: Vulnerable services – Sendmail gives out information using EXPN / VRFY</p> <p>NESSUS:</p> <ul style="list-style-type: none"> <li>- Security hole - rlogin is activated.</li> <li>- Security hole - rsh services is activated.</li> </ul> <p><b>Action taken:</b></p> <ul style="list-style-type: none"> <li>- Modify /etc/sendmail.cf setting (please refer to Appendix G for details )</li> <li>- Disable rsh in /etc/inetd.conf of the firewall.</li> </ul>	<p>SAINT: 4 trusted hosts - the DNS servers (berly, sapphire, garden, cucs) and their vulnerable services are identified</p> <p>NESSUS :</p> <ul style="list-style-type: none"> <li>- Sendmail supports the EHLO greetings and ESMTP and aloe EXPN command- can be a security flaw.</li> <li>- The services running at the firewall can still be detected e.g. ftp(21/tcp). Also some unknown port numbers e.g. (3464/tcp, 5311/tcp) were reported/</li> <li>- QueSO has found out that the firewall hosy OS is :Linux</li> </ul>	6

			2.1.xx	
<b>Level 6</b>	6	<p>NESSUS:                      - Security hole - rlogin is activated.                      (rlogin is still needed for users )</p> <p><b>Action taken:</b> Deny all the traffic from the IP of the attackers</p>	<p>SAINT: 4 trusted hosts - the DNS servers (berly, sapphire, garden, cucs) and their vulnerable services are identified</p> <p>NESSUS:                      - The services running at the firewall can still be detected e.g. ftp(21/tcp). Also some unknown port numbers e.g. (3464/tcp, 5311/tcp) were reported/</p>	3
<b>Level 7</b>	7	<p>SAINT: No scanning can be done as all the traffic to the firewall are rejected. (Please refer to Appendix F for the report result.)</p> <p>NESSUS:: No result can be produced as all the traffic from the attacker where this scanner is located.</p>	Nothing can be detected.	0

### 5.1.2 Analysis

As expected or as pre-designed, the  $x + 1$  security level is no less secured than the security level  $x$ . When looking at the details of the firewall policy 4 and 5, these two policies are expected to be more or less the same with regards to security level. They only differ in the basic policy features. So it is expected that the number of vulnerabilities found for policy 4 and 5 would not differ a lot from each other.

As a matter of fact, the testing result is not so representative if it is used to confirm the security standard of a particular firewall policy. However, the testing is an attempt to quantify the difference of security between one security level and the others. Furthermore, it helps to build up the security from one level to another by eliminating some security flaws and warnings found in the firewall of lower level security.

## 5.2 Performance Testing

Please find in the Appendix C for all the raw data sets of the testing result and measurement details.

Testing results obtained from experiments done under the seven firewall security levels would be presented and compared as follows.

### 5.2.1 HTTP session test

Please refer to HTTP testing scenarios described in section 3.3.3.1 and various firewall policies and the definitions for various security levels described in section 4.3.

#### 5.2.1.1 Result

The tables below show the AVERAGE total transaction time (in second) and the BEST total transaction time (in second) for retrieving 395K documents through HTTP.

Note: Cfg x refers to the firewall configuration x with security level defined as level x.

<u>No. of sequential connections</u>	<u>Cfg1</u>	<u>Cfg2</u>	<u>Cfg3</u>	<u>Cfg4</u>	<u>Cfg5</u>	<u>Cfg6</u>	<u>Cfg7</u>
1x3	0.94	1.00	1.50	1.25	2.86	1.33	2.75
10x3	10.40	13.00	63.88	65.33	70.88	63.33	63.25
20x3	22.20	30.14	304.38	313.60	316.38	302.00	304.50
30x3	31.40	40.67	513.75	507.50	531.75	493.33	501.50
40x3	51.60	58.60	653.33	683.67	666.00	761.00	666.00
50x3	60.80	72.83	836.86	837.71	861.14	832.67	839.25
60x3	78.60	90.50	1,014.25	1,012.00	1,002.00	1,004.33	1,011.00
70x3	86.20	105.33	1,201.29	1,219.75	1,235.29	1,185.67	1,198.25
80x3	98.20	117.50	1,351.57	1,386.67	1,377.14	1,361.67	1,371.75
90x3	111.00	125.00	1,558.86	1,538.00	1,552.43	1,536.00	1,526.25
100x3	143.40	150.33	1,710.71	1,716.33	1,743.00	1,674.33	1,737.25

Table 1: The AVERAGE total HTTP transaction times in second

<u>No. of sequential connections</u>	<u>Cfg1</u>	<u>Cfg2</u>	<u>Cfg3</u>	<u>Cfg4</u>	<u>Cfg5</u>	<u>Cfg6</u>	<u>Cfg7</u>
1x3	0.50	0.50	1.00	1.00	1.00	1.00	1.00
10x3	7.00	6.00	61.00	64.00	62.00	63.00	63.00
20x3	14.00	18.00	300.00	305.00	299.00	298.00	293.00
30x3	14.00	31.00	507.00	479.00	499.00	489.00	496.00
40x3	47.00	48.00	555.00	652.00	660.00	761.00	666.00
50x3	30.00	61.00	830.00	750.00	833.00	824.00	835.00
60x3	67.00	75.00	1,008.00	895.00	959.00	990.00	1,007.00
70x3	35.00	83.00	1,196.00	1,189.00	1,200.00	1,173.00	1,169.00
80x3	92.00	108.00	1,303.00	1,360.00	1,344.00	1,343.00	1,339.00
90x3	36.00	110.00	1,554.00	1,515.00	1,524.00	1,508.00	1,429.00
100x3	120.00	128.00	1,617.00	1,645.00	1,692.00	1,662.00	1,686.00

Table 2: The BEST total HTTP transaction times in second

For the table of the best results (i.e. the HTTP total minimum transaction times), the best values were selected, i.e. the minimum time of processing, as the final result. Therefore the values under a particular security level approximate the best-case performance results.

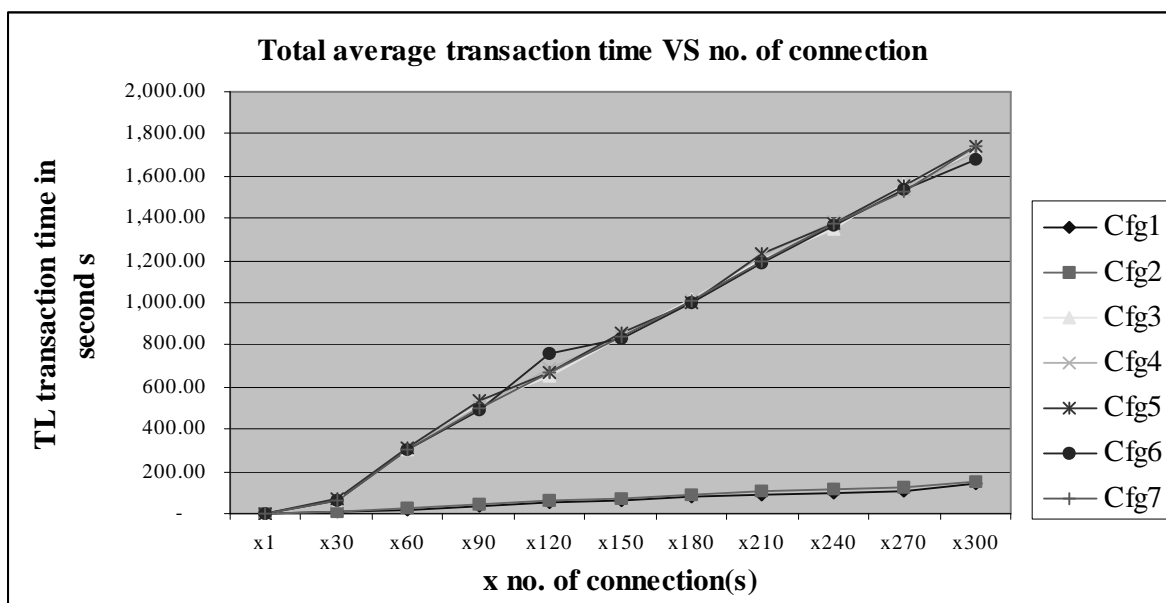


Figure 9: The HTTP total average transactions times VS the no. of connection(s) under different firewall security levels

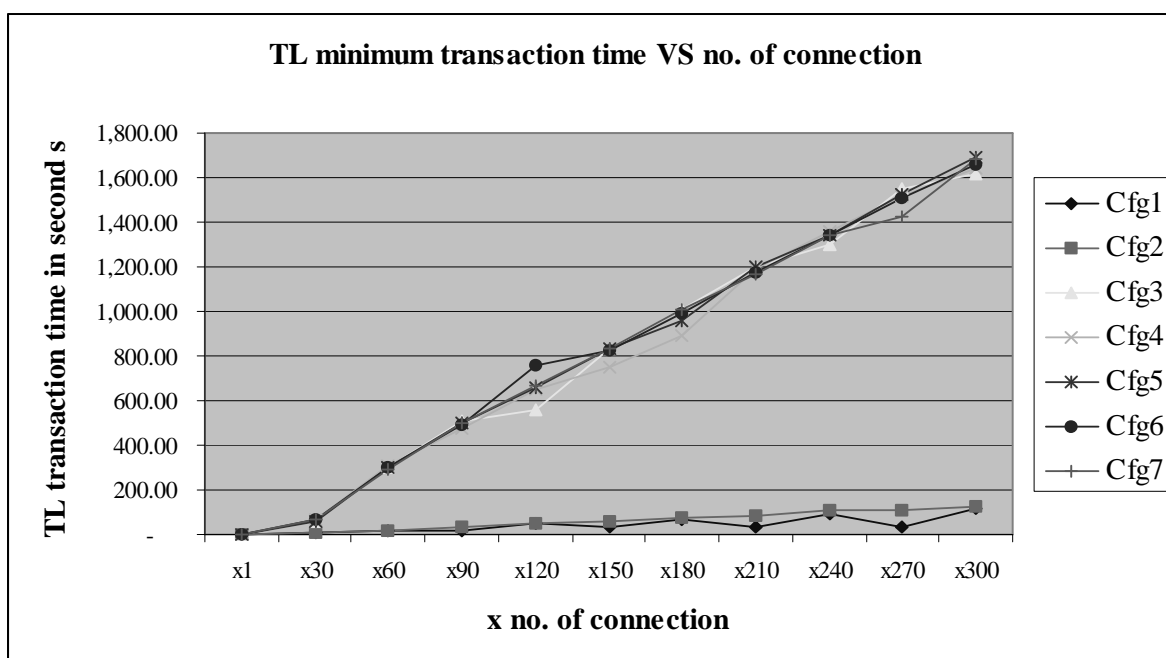


Figure 10: The HTTP total best transactions times VS the no. of connection(s) under different firewall security levels



The tables below show the network latency for processing each transaction of 395K bytes data. Please note that each of the transactions made would involve 3 connections realized with 3 HTTP requests for the transfer of 395K bytes data.

Note: x1 means 1 transaction and 3x 1 connections; x20 means 3x20 connections and so on.

No. of transaction(s) ( x 1 = 1 a transaction)

	<u>X1</u>	<u>X10</u>	<u>X20</u>	<u>X30</u>	<u>X40</u>	<u>X50</u>	<u>X60</u>	<u>X70</u>	<u>X80</u>	<u>X90</u>	<u>x100</u>
<b>Cfg1</b>	0.94	1.04	1.11	1.05	1.29	1.22	1.31	1.23	1.23	1.23	1.43
<b>Cfg2</b>	1.00	1.30	1.51	1.36	1.47	1.46	1.51	1.50	1.47	1.39	1.50
<b>Cfg3</b>	1.50	6.39	15.22	17.13	16.33	16.74	16.90	17.16	16.89	17.32	17.11
<b>Cfg4</b>	1.25	6.53	15.68	16.92	17.09	16.75	16.87	17.43	17.33	17.09	17.16
<b>Cfg5</b>	2.86	7.09	15.82	17.73	16.65	17.22	16.70	17.65	17.21	17.25	17.43
<b>Cfg6</b>	1.33	6.33	15.10	16.44	19.03	16.65	16.74	16.94	17.02	17.07	16.74
<b>Cfg7</b>	2.75	6.33	15.23	16.72	16.65	16.79	16.85	17.12	17.15	16.96	17.37

Table 3: Latency calculated with average total HTTP transaction time transaction times in second

No. of transaction(s) ( x 1 = 1 connection in a transaction)

	<u>x1</u>	<u>X10</u>	<u>X20</u>	<u>X30</u>	<u>X40</u>	<u>X50</u>	<u>X60</u>	<u>X70</u>	<u>X80</u>	<u>X90</u>	<u>x100</u>
<b>Cfg1</b>	0.50	0.70	0.70	0.47	1.18	0.60	1.12	0.50	1.15	0.40	1.20
<b>Cfg2</b>	0.50	0.60	0.90	1.03	1.60	1.53	1.25	1.19	1.35	1.22	1.28
<b>Cfg3</b>	1.00	6.10	15.00	16.90	13.88	16.60	16.80	17.09	16.29	17.27	16.17
<b>Cfg4</b>	1.00	6.40	15.25	15.97	16.30	15.00	14.92	16.99	17.00	16.83	16.45
<b>Cfg5</b>	1.00	6.20	14.95	16.63	16.50	16.66	15.98	17.14	16.80	16.93	16.92
<b>Cfg6</b>	1.00	6.30	14.90	16.30	19.03	16.48	16.50	16.76	16.79	16.76	16.62
<b>Cfg7</b>	1.00	6.30	14.65	16.53	16.65	16.70	16.78	16.70	16.74	15.88	16.86

Table 4: Latency calculated with best total HTTP transaction time transaction times in second

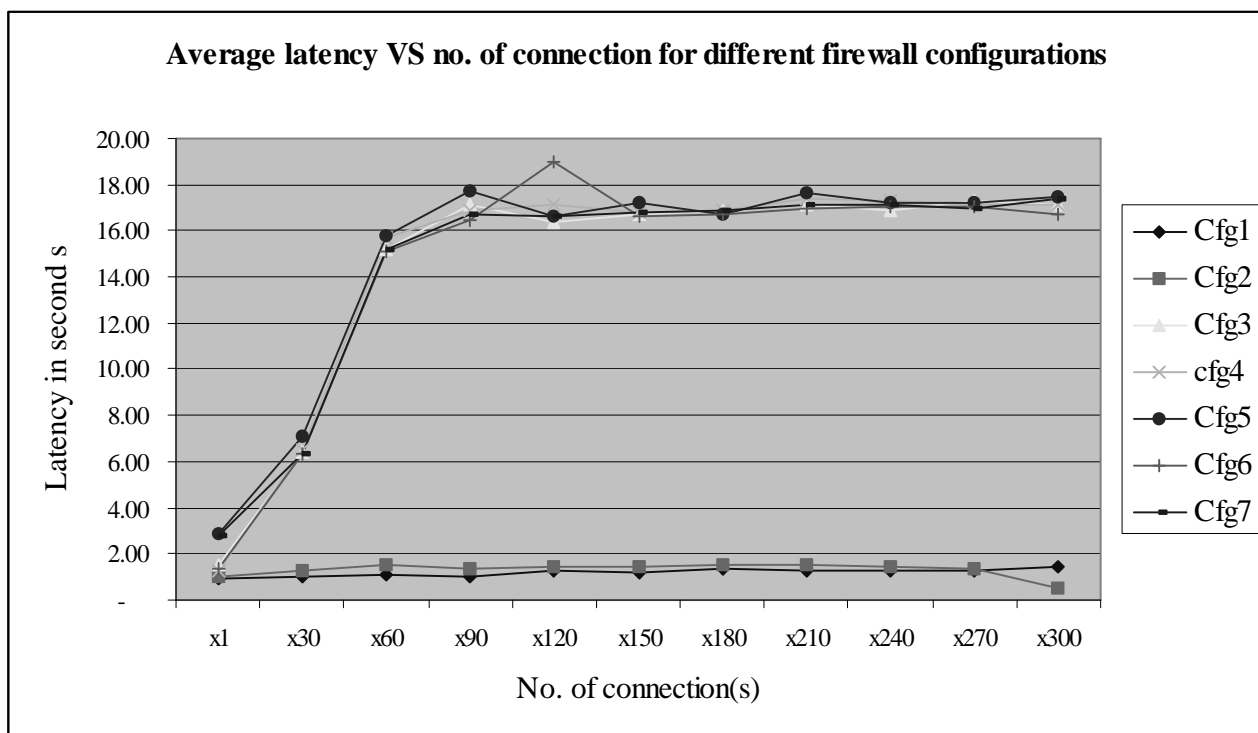


Figure 11: The average HTTP latency of a transactions VS the no. of connection(s) under different firewall security levels

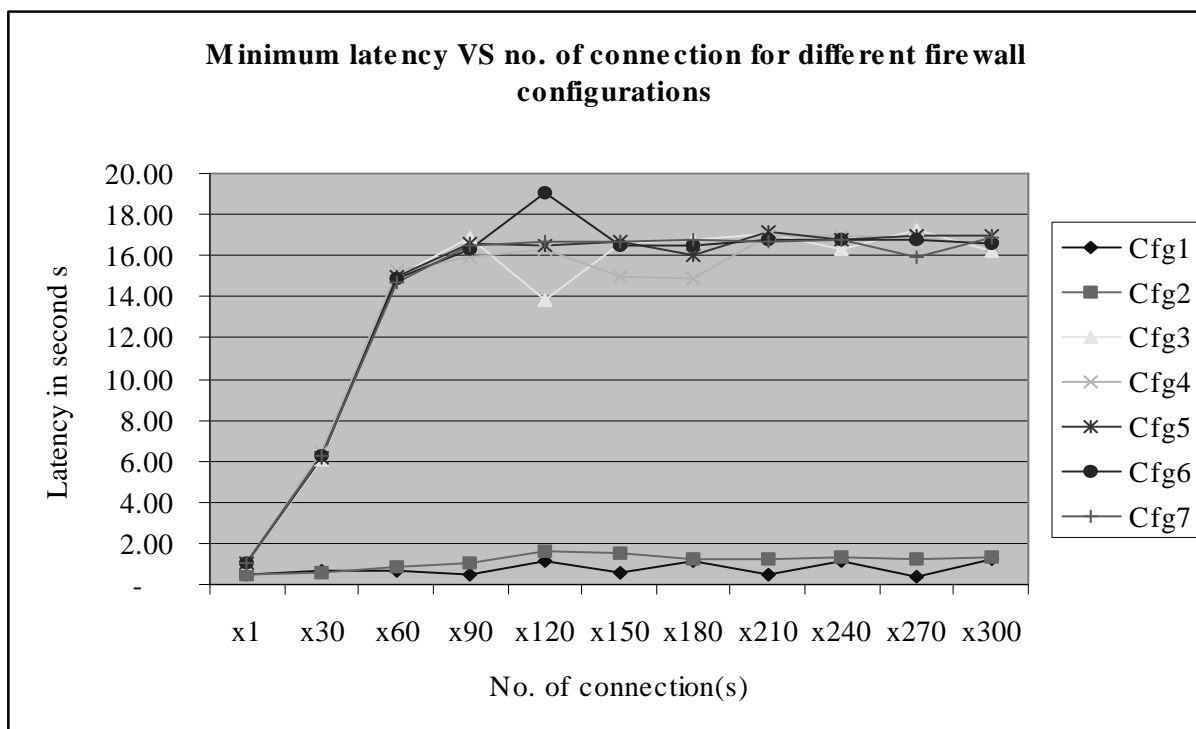


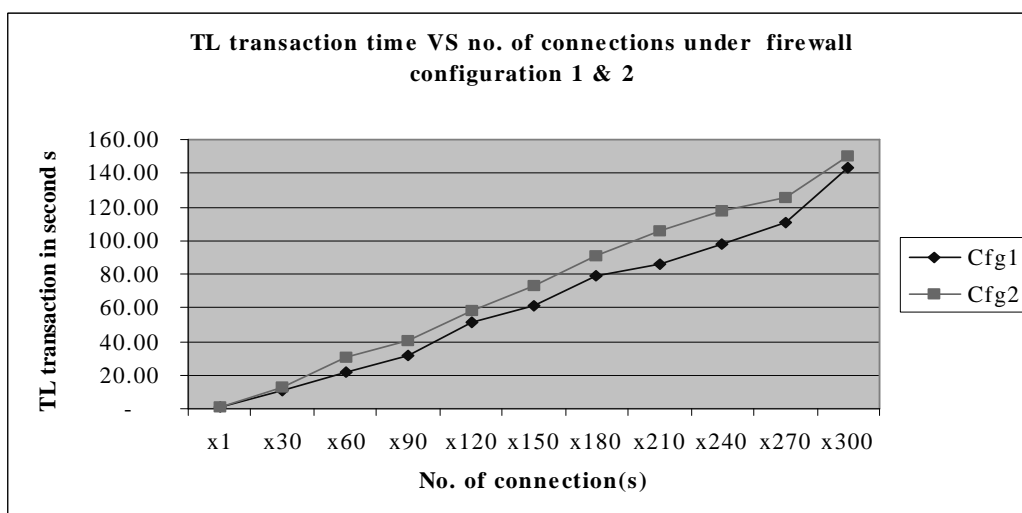
Figure 12: The best HTTP latency of a transactions VS the no. of connection(s) under different firewall security levels

### 5.2.1.2 Analysis

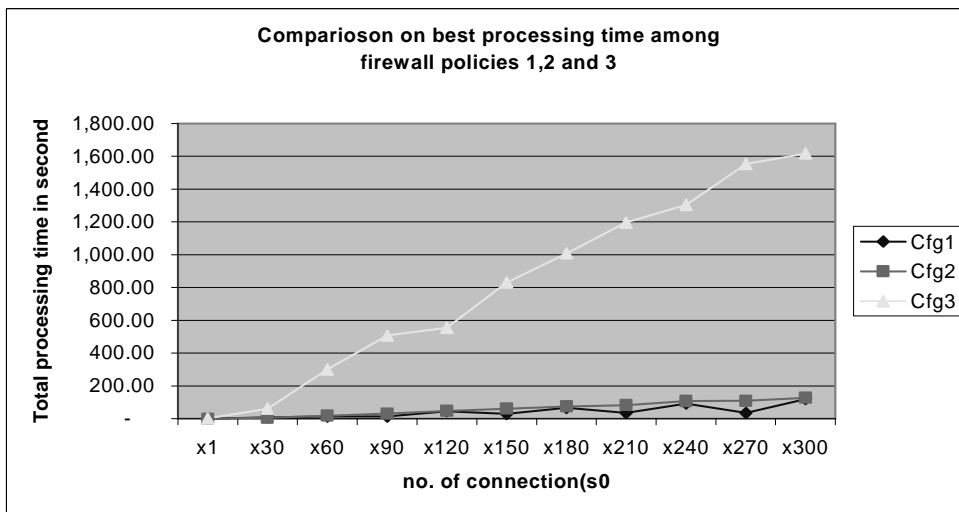
With the design of the high-volume connection and low data size (< 395Kb) file retrieval by using HTTP protocol, the total data transfer or transaction times for the firewall policy 1, 2 and 3 differs significantly. When compared with the result of security level 2 and 3, the tests showed a remarkable increase of latency and processing time with firewall security level 3, with connection > 1. When proceeding from security level 3 to 4, 5, 6 or 7, no obvious and consistent change of performance could be found.

First, as expected that the firewall of lowest security level (i.e. level 1) performs the best. It is because there is no packet filtering rules set into the router which does very little work outside of routing traffic, extremely low overhead would be incurred in this security level.

Second, the tests showed that security level 2 performs better than security level one. This can be explained by the addition of security in security level 2 implemented by setting 7 screening rules into the router. As there is no rule set for packet filtering for the previous security level i.e. level 1, even the addition of 1 filtering rule would incur traffic overhead as compared with the previous security policy 1. In fact, the traffic may not go through all the 7 screening rules in the router, but the packet would be checked or parsed with at least 1 rule and then be passed to the firewall. The difference of performance is shown in the following figure.



Third, as shown in the above figure, the significant increase of processing time with security level 3 is due to the incorporation of proxy services at the firewall. The additional security made by the proxy server imposes overhead, which is comparatively significant to the total processing time and latency of data transfer. The proxy process run at the firewall (the proxy server) will analyses application commands inside the payload portion of data packets and keeps logs of traffics as well as specific activities. Thus it incurred higher overhead than simple packet filtering firewall, such as the firewall incorporated with security level 2 in this project. Moreover, for each new connection to the Internet, the overhead from proxy process would be introduced. Consequently the time for HTTP transfer jumped up quickly when connection number is increased from 1 to 30. It is illustrated below.



Interestingly, the curves of security 3, 4, 5, 6 and 7 seems to be overlapped each others, whether there is the performance gains or loss is difficult to conclude. In fact, the performances of security level 4 to 7 could be explained by the way that security policies were implemented. The firewall security levels 3 to 7, are mainly implemented and controlled by configuring screening rules in the router. In fact, the security level of 3 (or x) is theoretically better than level 4 (or x+1) and the security policy 3 (or x) is proved in the security testing that is it more secured than the security policy 4 (or x+1). But the difference between the number of screenings rules used for the implementation of firewall policy 3 (or x) and that for the implementation of policy 4 (or x+1) is zero, which is not big enough to show up any significant impact on the performance.

When the no. of screening for security level 7 is increased to 43, 26 more screening rule than level 3, it is found that the performance values at some particular number of connection for security level

7 is larger than that for security level 3. It is interesting to note that there is clearly the performance difference when the no. of screening rule is added from 0 to 7 when security level proceeding from 1 to 2. But if the rules are added again, the performance difference is not obvious at all. This phenomenon could be explained by way the router parses the screening rules. Normally the router parses the rules in sequential order for a match. The more the rules are parsed, the slower the traffic would be. So the speed of traffic going through the router depends very much on the sequence of the rules set into the router. When a particular traffic is matched by a rule in the sequence, the fate of the packet is determined and the rest of rules is ignored, no matter how many rules are set. In other words, if the traffic is matched in the sequence of rules earlier, the faster the traffic goes through the router. As a result, if the FTP traffic simulated in policy 3 to 7 is only parsed with the same number of rules before their fates is determined, it is no wonder why their performance are very close to each others.

Consequently, even it is shown that the performance of security level  $x$  is better than that of  $x+1$  a little bit at certain test point or no. of connections, but on the whole the performance difference among security level 3 to 7 is not obvious and is difficult to conclude.

The irregular shape of the performance curves for firewall policy 3 to 7 also revealed that the performance of them is easily affected by the outside interference and noise. However, it is nearly impossible to predict and control the impact of outside traffic to our results, so it is difficult to estimate how many trials should be run such that a smooth curve for each policy is resulted.

In short, for security policy 3 to 7, because of the impact from external traffic together with the minimal performance difference among their security levels, no plausible evidence could be found in this project that the performance of security level 3 is better than that of level 4, that of level 4 is better than that of level 5 and so on. But with regards to the security features and control, the higher security levels are actually more secured than the lower security levels.

## 5.2.2 FTP session test

Please refer to FTP testing scenarios described in section 3.3.3.2 and various firewall policies and security level definitions described in section 4.3.

As the connections for transferring different data sizes of file are supposed to be made sequentially, the firewall was able to process all the transfer requests. All the connection requests are started at the Linux PC "HOME" for consistent comparison.

### 5.2.2.1 Result

#### Scenario A - Transfer 5M data per transaction

##### Total transaction time

The tables below show the total average and best transaction times taken to complete file transfers in the scenario A.

<b>No. of Connection</b>	<b>Cfg1</b>	<b>Cfg2</b>	<b>Cfg3</b>	<b>Cfg4</b>	<b>Cfg5</b>	<b>Cfg6</b>	<b>Cfg7</b>
1	11.13	12.33	10.00	10.75	12.00	11.60	14.00
2	27.86	26.57	26.75	22.40	25.00	27.39	29.80
3	44.00	44.00	37.67	34.40	38.67	38.23	40.17
4	56.57	56.60	55.00	58.00	48.20	57.51	53.00
5	71.44	73.00	60.67	69.00	66.40	73.88	64.83
6	88.88	87.25	79.25	85.00	78.25	78.71	77.33
7	107.11	103.86	91.00	95.00	91.67	96.08	90.83
8	118.75	119.60	98.33	110.60	98.75	107.51	107.83
9	134.67	134.00	126.00	129.60	116.60	130.27	133.83
10	144.33	143.17	133.75	135.50	133.50	146.89	129.33

Table 5 : Average total transaction figures

<b>No. of connections</b>	<b>Cfg1</b>	<b>Cfg2</b>	<b>Cfg3</b>	<b>Cfg4</b>	<b>Cfg5</b>	<b>Cfg6</b>	<b>Cfg7</b>
1	3.00	12.00	8.00	9.00	9.00	9.30	13.00
2	24.00	23.00	23.00	19.00	20.00	26.65	27.00
3	39.00	42.00	35.00	27.00	38.00	29.22	32.00
4	52.00	52.00	44.00	49.00	43.00	53.92	49.00
5	65.00	67.00	55.00	64.00	62.00	71.30	57.00
6	75.00	85.00	73.00	81.00	75.00	72.06	72.00
7	92.00	100.00	76.00	87.00	87.00	95.26	76.00
8	106.00	115.00	91.00	105.00	86.00	98.57	97.00
9	127.00	127.00	119.00	121.00	114.00	125.37	114.00
10	140.00	135.00	131.00	132.00	130.00	145.85	119.00

Table 6: Best figures with minimum transaction time of the result

Figures below illustrate respectively the total average and minimum FTP transfer times for 5M data.

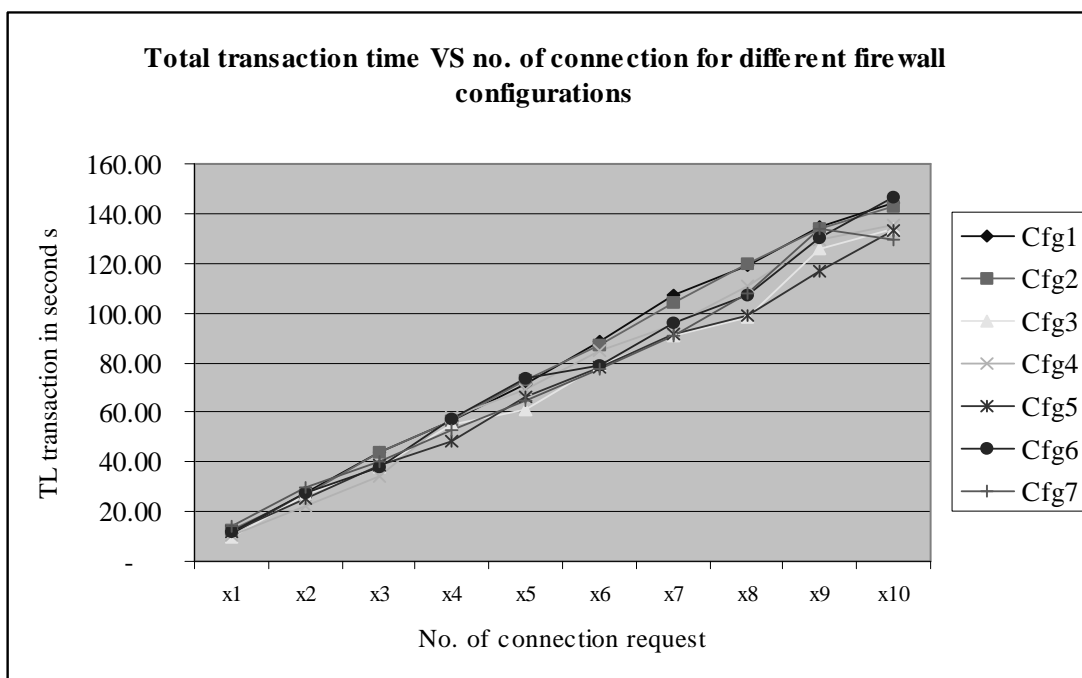


Figure 13 : Total transaction time on average for data transfer by FTP VS no. of connection

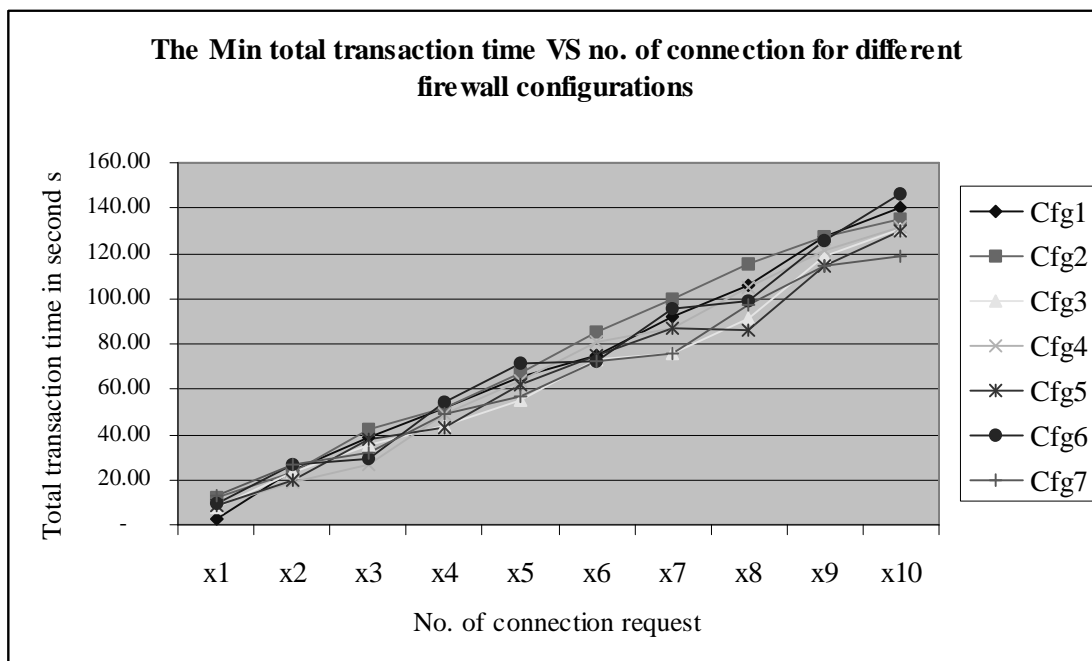


Figure 14 : Minimum total transaction time for data transfer by FTP VS no. of connection

**Latency for each transaction**

The tables below show the average and best latency time taken to complete the file transfer in scenario A

No. of connection(s) ( x 1 = 1 connection in 1 transaction)

	<u>x1</u>	<u>x2</u>	<u>x3</u>	<u>x4</u>	<u>x5</u>	<u>x6</u>	<u>X7</u>	<u>X8</u>	<u>x9</u>	<u>x10</u>
<b>Cfg1</b>	11.13	13.93	14.67	14.14	14.29	14.81	15.30	14.84	14.96	14.43
<b>cfg2</b>	12.33	13.29	14.67	14.15	14.60	14.54	14.84	14.95	14.89	14.32
<b>Cfg3</b>	10.00	13.38	12.56	13.75	12.13	13.21	13.00	12.29	14.00	13.38
<b>Cfg4</b>	10.75	11.20	11.47	14.50	13.80	14.17	13.57	13.83	14.40	13.55
<b>Cfg5</b>	12.00	12.50	12.89	12.05	13.28	13.04	13.10	12.34	12.96	13.35
<b>Cfg6</b>	11.60	13.69	12.74	14.38	14.78	13.12	13.73	13.44	14.47	14.69
<b>Cfg7</b>	14.00	14.90	13.39	13.25	12.97	12.89	12.98	13.48	14.87	12.93

*Table 7 : Latency calculated from the total average transaction times of FTP 5M data*

No. of connection(s) ( x 1 = 1 connection in 1 transaction)

	<u>x1</u>	<u>x2</u>	<u>x3</u>	<u>x4</u>	<u>x5</u>	<u>x6</u>	<u>X7</u>	<u>x8</u>	<u>x9</u>	<u>x10</u>
<b>Cfg1</b>	3.00	12.00	13.00	13.00	13.00	12.50	13.14	13.25	14.11	14.00
<b>cfg2</b>	12.00	11.50	14.00	13.00	13.40	14.17	14.29	14.38	14.11	13.50
<b>Cfg3</b>	8.00	11.50	11.67	11.00	11.00	12.17	10.86	11.38	13.22	13.10
<b>Cfg4</b>	9.00	9.50	9.00	12.25	12.80	13.50	12.43	13.13	13.44	13.20
<b>Cfg5</b>	9.00	10.00	12.67	10.75	12.40	12.50	12.43	10.75	12.67	13.00
<b>Cfg6</b>	9.30	13.33	9.74	13.48	14.26	12.01	13.61	12.32	13.93	14.59
<b>Cfg7</b>	13.00	13.50	10.67	12.25	11.40	12.00	10.86	12.13	12.67	11.90

*Table 8 : Latency calculated from the total (best) minimum times of FTP 5M data*

Figures below illustrate respectively the average and minimum latency of files transfer (FTP) 5M data.



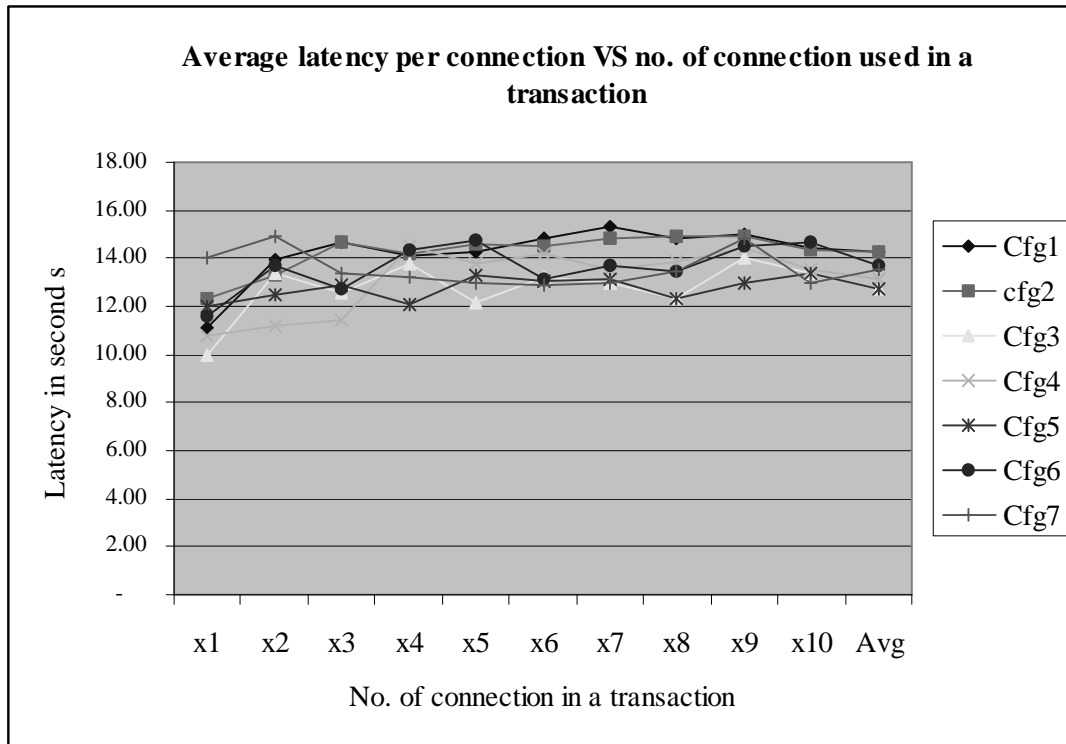


Figure 15 : Latency calculated from the AVERAGE TL transactions times VS no. of connection(s)

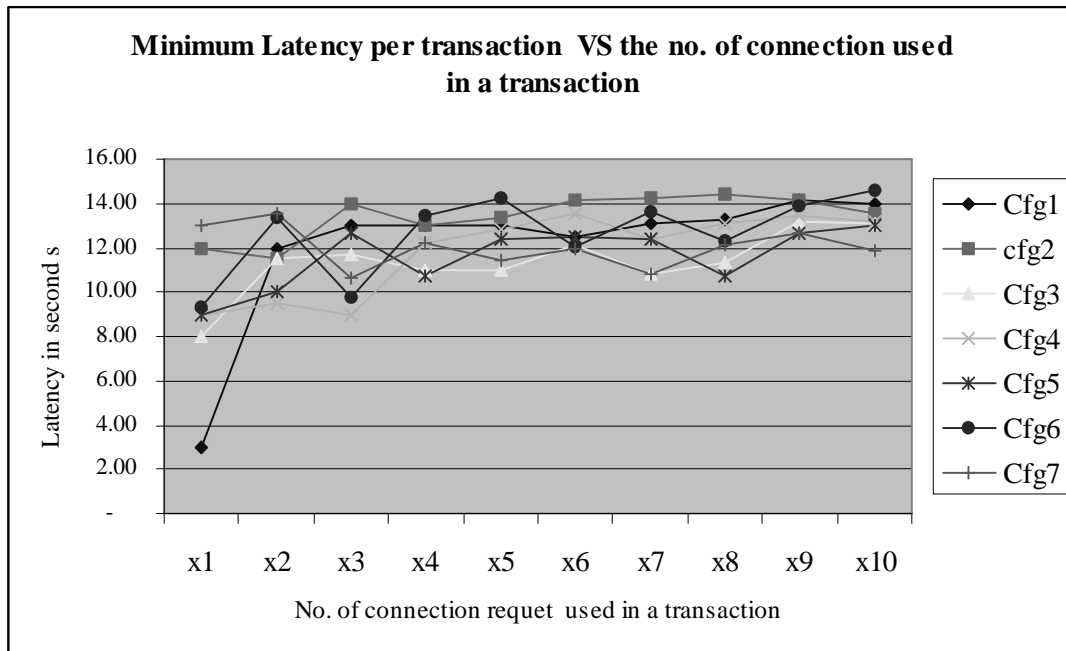


Figure 16 : Minimum Latency calculated from the (best) minimum times VS no. of connection(s)

**Scenario B - Transfer 1M data per transaction**

**Total transaction time**

The tables below show the average and best total time taken to complete the file transfer in scenario B (1M data transfer).

<b>No. of connection</b>	<b>Cfg1</b>	<b>Cfg2</b>	<b>Cfg3</b>	<b>Cfg4</b>	<b>Cfg5</b>	<b>Cfg6</b>	<b>Cfg7</b>
1	3.00	3.67	3.67	2.78	3.60	3.00	3.13
2	7.89	7.33	6.80	6.80	7.89	7.17	7.25
3	12.44	11.20	11.60	11.10	11.89	12.38	11.57
4	16.00	16.43	16.00	15.80	15.22	17.38	15.88
5	20.89	20.86	20.20	19.80	19.56	20.89	20.30
6	25.75	26.17	25.40	23.78	24.13	24.11	24.60
7	32.00	29.71	30.00	27.60	29.11	29.11	28.00
8	36.33	36.71	33.20	31.70	32.00	33.22	33.60
9	39.43	40.43	39.40	37.11	36.11	38.71	37.20
10	42.33	44.00	40.20	41.56	40.67	42.38	41.60

Table 9 : Average figures of total transaction times

<b>No. of connection</b>	<b>Cfg1</b>	<b>Cfg2</b>	<b>Cfg3</b>	<b>Cfg4</b>	<b>Cfg5</b>	<b>Cfg6</b>	<b>Cfg7</b>
1	2.00	3.00	3.00	2.00	3.00	2.00	2.00
2	6.00	5.00	6.00	6.00	7.00	5.00	6.00
3	11.00	8.00	11.00	10.00	10.00	11.00	11.00
4	14.00	14.00	14.00	14.00	14.00	15.00	14.00
5	19.00	18.00	18.00	19.00	17.00	18.00	18.00
6	20.00	24.00	21.00	21.00	20.00	21.00	22.00
7	30.00	25.00	28.00	25.00	27.00	27.00	26.00
8	35.00	33.00	29.00	31.00	28.00	30.00	31.00
9	35.00	38.00	36.00	35.00	32.00	37.00	34.00
10	39.00	40.00	37.00	39.00	37.00	38.00	39.00

Table 10: Best figures with minimum transaction time of the result

Figures below illustrate respectively the average and best (minimum) FTP files transfer times for 1M data transfer.

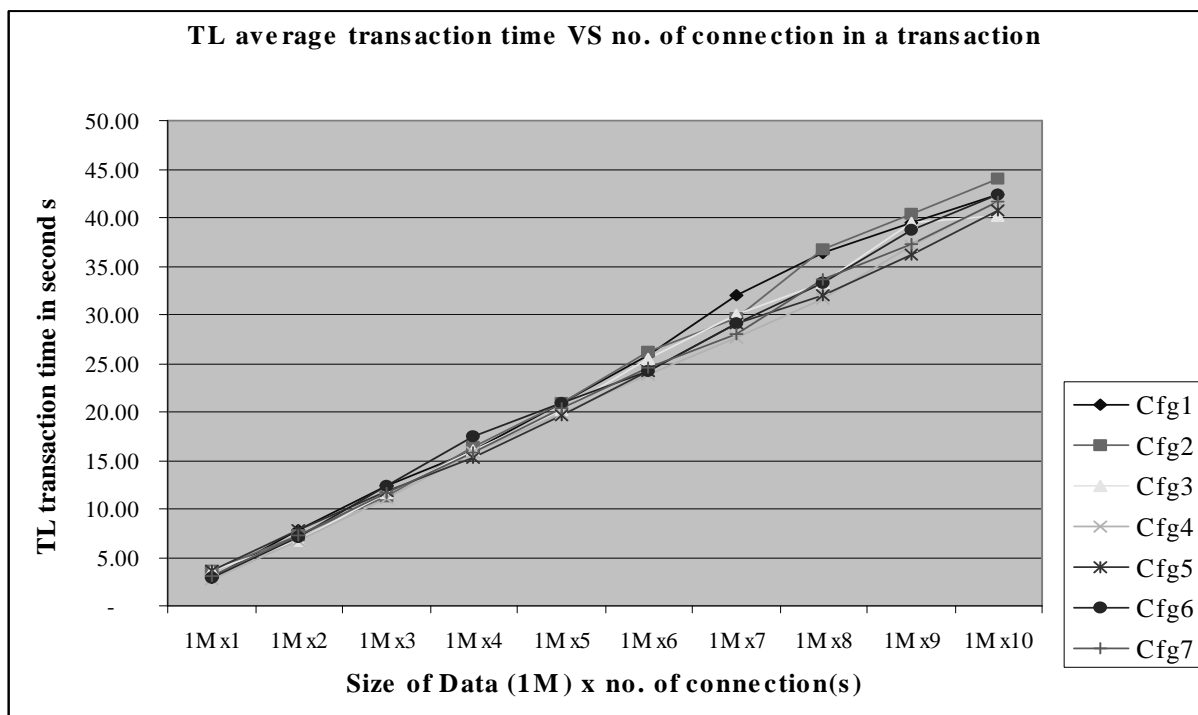


Figure 17 : Total AVERAGE transactions times VS no. of connection(s) for 1M data transfer

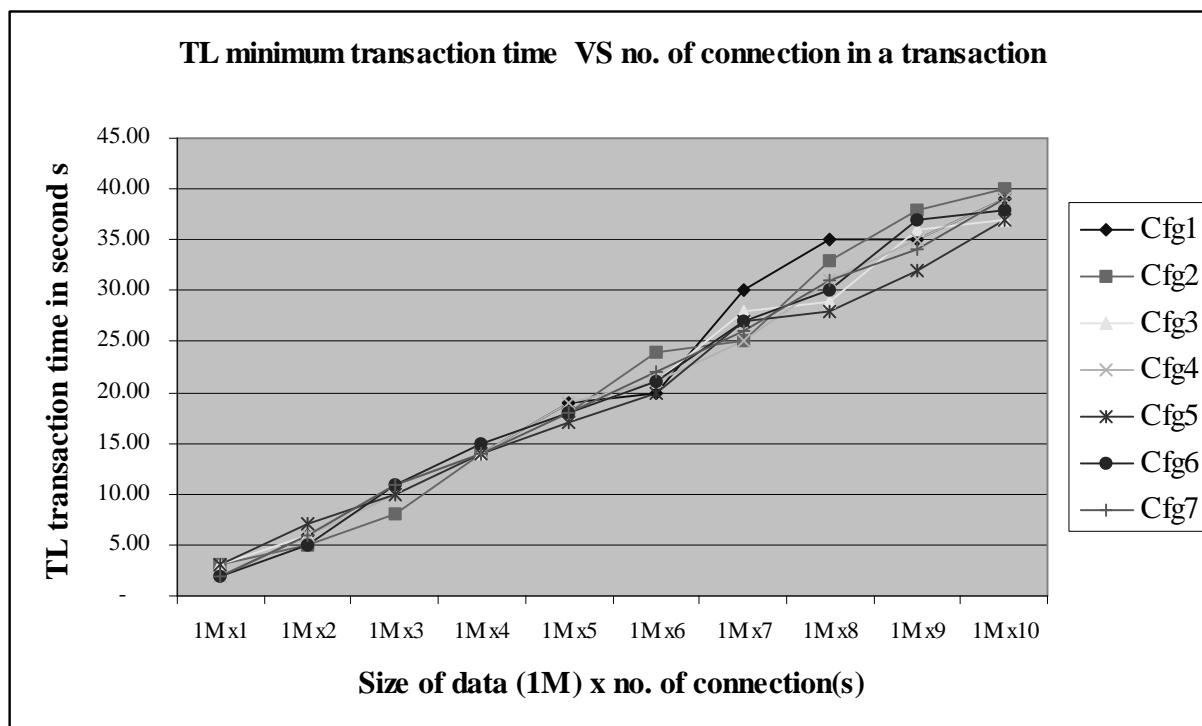


Figure 18 : Total MINIMUM transactions times VS no. of connection(s) for 1M data transfer

**Latency for each transaction**

The tables below show the average and best latency time taken to complete the file transfer in scenario B

<b><u>No. of connection</u></b>	<b><u>Cfq1</u></b>	<b><u>Cfq2</u></b>	<b><u>Cfq3</u></b>	<b><u>Cfq4</u></b>	<b><u>Cfq5</u></b>	<b><u>Cfq6</u></b>	<b><u>Cfq7</u></b>
1	3.00	3.67	3.67	2.78	3.60	3.00	3.13
2	3.94	3.67	3.40	3.40	3.94	3.58	3.63
3	4.15	3.73	3.87	3.70	3.96	4.13	3.86
4	4.00	4.11	4.00	3.95	3.81	4.34	3.97
5	4.18	4.17	4.04	3.96	3.91	4.18	4.06
6	4.29	4.36	4.23	3.96	4.02	4.02	4.10
7	4.57	4.24	4.29	3.94	4.16	4.16	4.00
8	4.54	4.59	4.15	3.96	4.00	4.15	4.20
9	4.38	4.49	4.38	4.12	4.01	4.30	4.13
10	4.23	4.40	4.02	4.16	4.07	4.24	4.16

*Table 11* : Latency calculated from the average figures above

<b><u>No. of connection</u></b>	<b><u>Cfq1</u></b>	<b><u>Cfq2</u></b>	<b><u>Cfq3</u></b>	<b><u>Cfq4</u></b>	<b><u>Cfq5</u></b>	<b><u>Cfq6</u></b>	<b><u>Cfq7</u></b>
1	2.00	3.00	3.00	2.00	3.00	2.00	2.00
2	3.00	2.50	3.00	3.00	3.50	2.50	3.00
3	3.67	2.67	3.67	3.33	3.33	3.67	3.67
4	3.50	3.50	3.50	3.50	3.50	3.75	3.50
5	3.80	3.60	3.60	3.80	3.40	3.60	3.60
6	3.33	4.00	3.50	3.50	3.33	3.50	3.67
7	4.29	3.57	4.00	3.57	3.86	3.86	3.71
8	4.38	4.13	3.63	3.88	3.50	3.75	3.88
9	3.89	4.22	4.00	3.89	3.56	4.11	3.78
10	3.90	4.00	3.70	3.90	3.70	3.80	3.90

*Table 12* : Latency calculated from the (best) minimum figures above

Figures below illustrate respectively the average and best (minimum) latency of file transfer (FTP) 1M data.

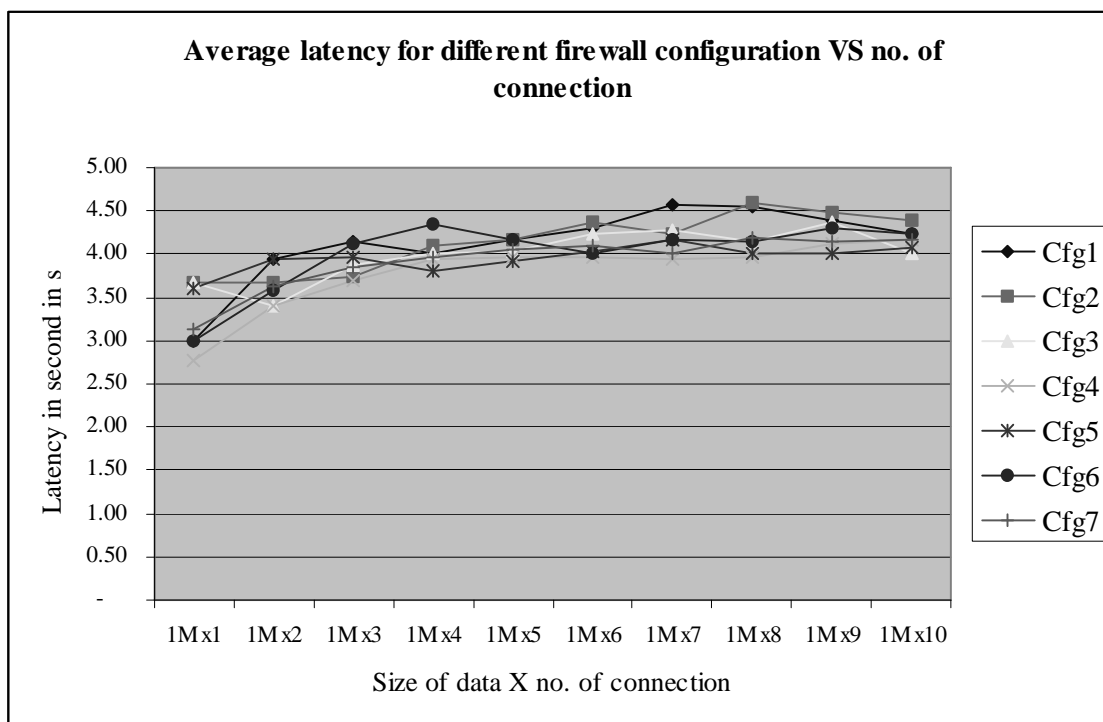


Figure 19 : Latency calculated from the AVERAGE TL transactions times VS no. of connection(s) for 1M data transfer

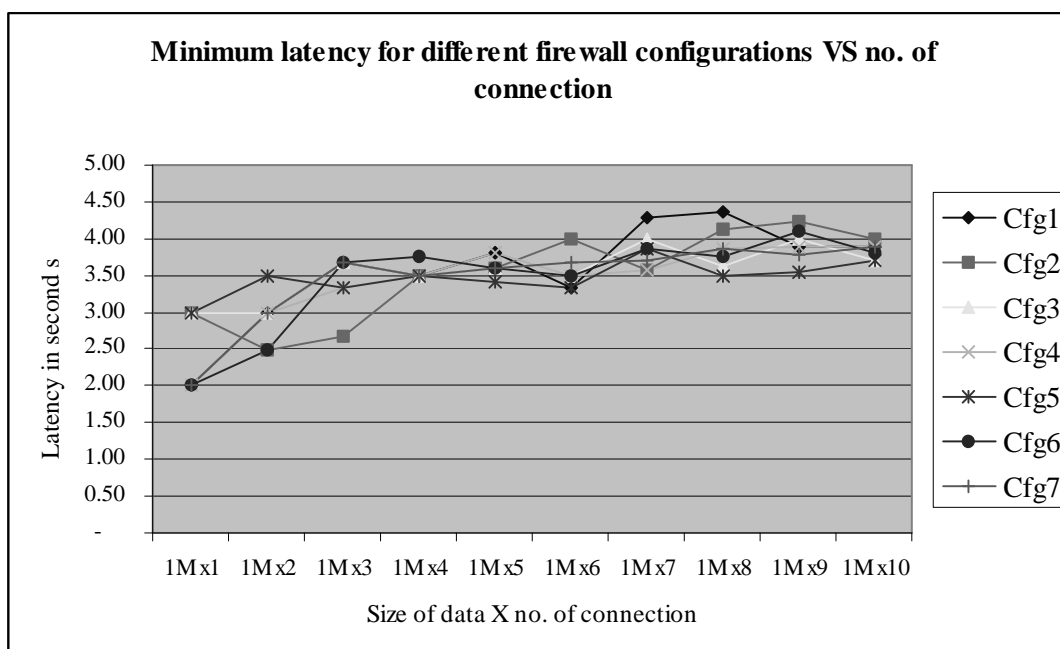


Figure 20 : Latency calculated from the MINIMUM TL transactions times VS no. of connection(s) for 1M data transfer

**Scenario C - Transfer smaller data of 38.9Kb per transaction, large volume of connections**

**Total transaction time**

The tables below show the average and best total time taken to complete the file transfer in scenario C (38.9K, large volume of connections data transfer).

No. of connection(s) ( x 1 = 1 connection in a transaction)

<u>No. of connection</u>	<u>Cfg1</u>	<u>Cfg2</u>	<u>Cfg3</u>	<u>Cfg4</u>	<u>Cfg5</u>	<u>Cfg6</u>	<u>Cfg7</u>
1	0.05	0.80	0.80	0.83	0.80	0.67	0.88
5	5.60	5.50	6.60	6.67	6.40	7.33	6.63
10	11.60	11.00	14.80	14.67	13.20	14.33	14.00
20	24.60	24.17	28.80	29.67	27.40	28.67	28.11
40	49.00	43.33	59.40	59.67	56.25	56.50	57.67

*Table 13* : Total average transaction times VS no. of connection in a transaction

No. of connection(s) ( x 1 = 1 connection in a transaction)

<u>No. of connection</u>	<u>Cfg1</u>	<u>Cfg2</u>	<u>Cfg3</u>	<u>Cfg4</u>	<u>Cfg5</u>	<u>Cfg6</u>	<u>Cfg7</u>
1	0.05	0.50	0.50	0.50	0.50	0.50	0.50
5	5.00	5.00	6.00	6.00	6.00	7.00	6.00
10	11.00	10.00	14.00	14.00	13.00	13.00	13.00
20	23.00	22.00	28.00	29.00	25.00	28.00	25.00
40	48.00	33.00	57.00	58.00	53.00	55.00	56.00

*Table 14*: Total best transaction time VS no. of connection in a transaction

Figures below illustrate respectively the average and best (minimum) FTP files transfer times for 38.9K, large volume of connections data transfer.

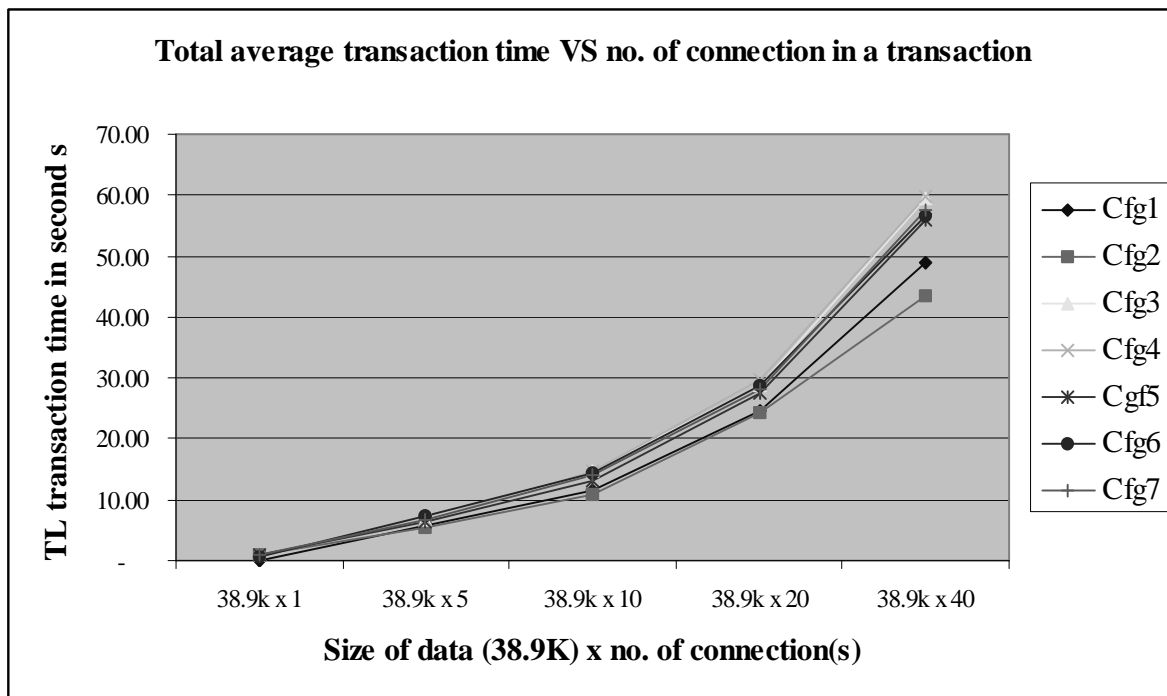


Figure 21 : Total AVERAGE TL transactions times VS no. of connection(s) for 38.9KM data transfer

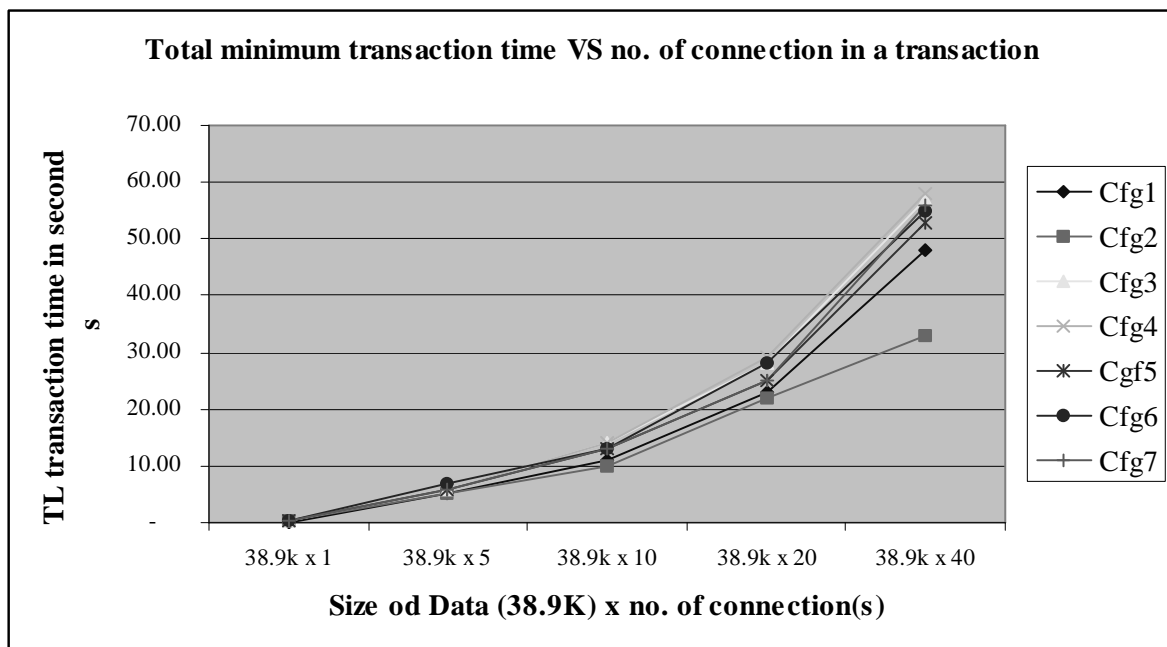


Figure 22 : Total MINIMUM transactions times VS no. of connection(s) for 38.9KM data transfer

**Latency for each transaction**

The tables below show the average and best latency time taken to complete the file transfer in scenario C (38.9K, large volume of connections data transfer) .

	<u>Cfq1</u>	<u>Cfq2</u>	<u>Cfq3</u>	<u>Cfq4</u>	<u>Cqf5</u>	<u>Cfq6</u>	<u>Cfq7</u>
38.9k x 1	0.05	0.80	0.80	0.83	0.80	0.67	0.88
38.9k x 2	1.12	1.10	1.32	1.33	1.28	1.47	1.33
38.9k x 3	1.16	1.10	1.48	1.47	1.32	1.43	1.40
38.9k x 4	1.23	1.21	1.44	1.48	1.37	1.43	1.41
38.9k x 5	1.23	1.08	1.49	1.49	1.40	1.41	1.44

*Table 15* : Latency calculated from the average figures above

	<u>Cfq1</u>	<u>Cfq2</u>	<u>Cfq3</u>	<u>Cfq4</u>	<u>Cqf5</u>	<u>Cfq6</u>	<u>Cfq7</u>
38.9k x 1	0.05	0.50	0.50	0.50	0.50	0.50	0.50
38.9k x 5	1.00	1.00	1.20	1.20	1.20	1.40	1.20
38.9k x 10	1.10	1.00	1.40	1.40	1.30	1.30	1.30
38.9k x 20	1.15	1.10	1.40	1.45	1.25	1.40	1.25
38.9k x 40	1.20	0.83	1.43	1.45	1.33	1.38	1.40

*Table 16* : Latency calculated from the (best) minimum figures above

Figures below illustrate respectively the average and best (minimum) latency of file transfer (FTP) 38.9K data.



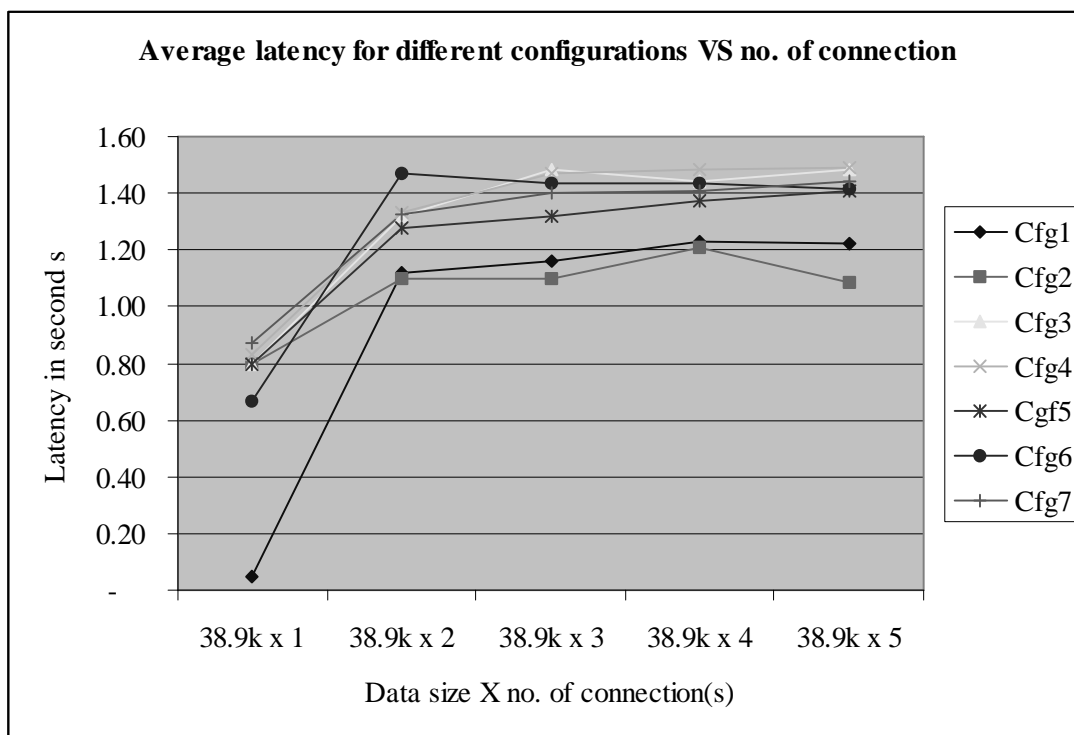


Figure 23 : Latency calculated from the AVERAGE TL transactions times VS no. of connection request(s) for 38.9KM data transfer

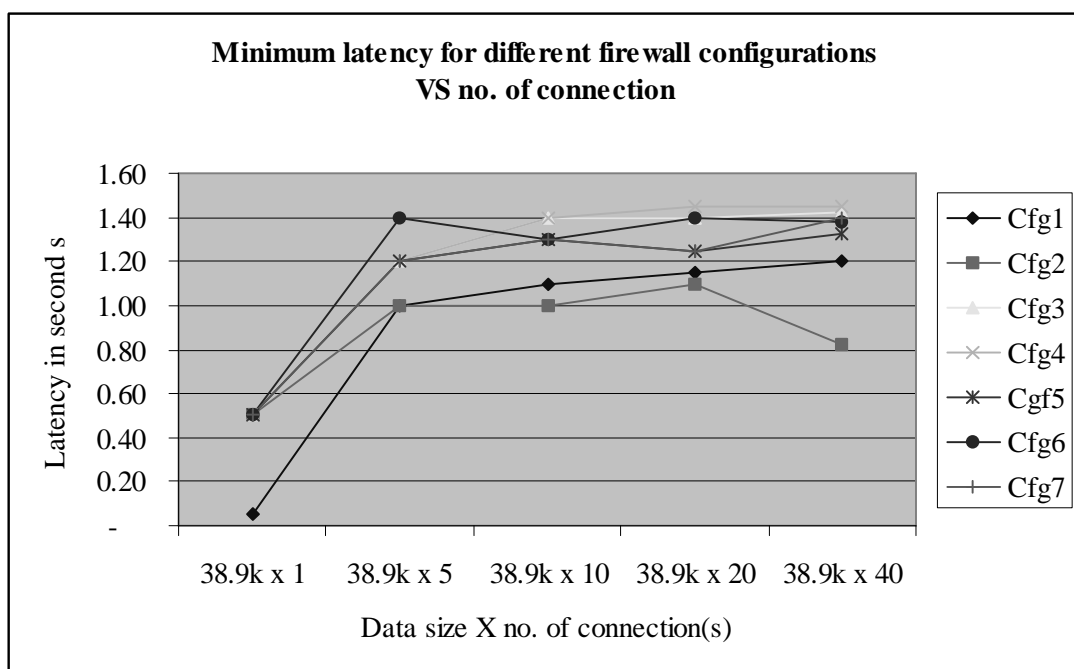


Figure 24 : Latency calculated from the MINIMUM TL transactions times VS no. of connection request(s) for 38.9KM data transfer

### **5.2.2.2 Analysis**

First, from the test results of scenario 1, 2 and 3, it is as expected that the latency for a particular security level remains more or less a constant value, as with 1 connection to 10 connections. This is because the FTP connection request is run sequentially and one connection should not influence another if more than 1 connections are attempted. However, just like the HTTP testing, it is not the case with the scenario 3 in which the latency of connection was increased since connection number is larger than 2 and it is much higher than that for connection 1. The irregularities of latency times for a particular security level is most probably due to the noise and interference from external traffic.

Second, we look at the difference of latency or total transaction times among different security level. It is found that the result values from low-volume connections testing, still remain more or less a constant no matter the file size is 5M or 1M. 'Low-volume connection' means 1 to 10 connections, whereas 'high-volume connection' means 1 to 40 connections as implemented in this project.

However, for high-volume connection and small data size tests, the latency values found under firewall configuration 3 to 7, are clearly larger than that under firewall configurations 1 and 2. This interesting result is similar to that found with HTTP protocol described in the previous section.

If the data size is large, the transaction time for each connection would be comparatively longer, and the overhead time added by proxying at the firewall would become insignificant with respect to the long time taken to complete a transaction.

On the contrary, if data size is small, the transaction time for each new connection would be shorter and the risk of collision and outside interference would be smaller. If the number of connections is high with small data size, the additional overhead coming from the proxy process would become significant when compared with the small value of latency. Also the overhead time from proxy process is incurred whenever each new connection is made. If the connection no. is high, the data traffic collision rate grows, the accumulated overhead from the connections would be high and this accumulated overhead would weigh a lot when compared to the total transaction time or latency without the overhead. In this way, with small file size and frequent connections to the Internet, the network performance would be affected if the firewall policy 3 or above is adopted. In other words,

the using of proxy server for more security at the firewall (as implemented in the firewall policy 3 to 7 in this project) would have to scarify the network performance.

Third, it is also clear that the performance difference among the firewall policy 3 to 7, is not large. Just like the results from HTTP tests, the fluctuations of the performance curves for firewall policy 3 to 7 appear very often in testing all the 3 scenarios of data transfer with FTP protocol. They imply that the overhead added by 'more security' or 'higher security level' does not outweigh the influences from outside traffic interference and noise. As a result, it is very difficult to determine how big difference of the performance impact among the implementations of firewall security policy 3 to 7.

### 5.3 Relationship of Security to Performance

With the findings mentioned above, it is interesting to relate the security of firewall defined in this project with the performance results. Here below is a simple matrix used to illustrate the security-performance relationship.

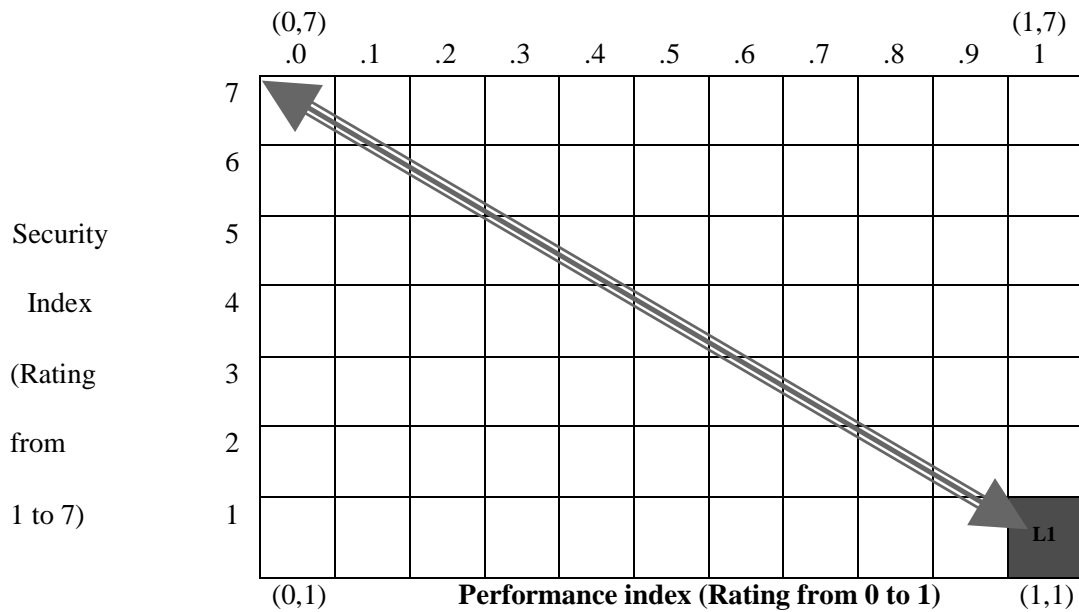


Figure 25: Security-performance matrix

Notes for the coordinate x,y : (where x is from 0 to 1, y is from 1, 7 )

- 0, 7    poorest performance, highest security
- 1, 7    best performance, highest security
- 0, 1    poorest performance, lowest security
- 1, 1    best performance, lowest security

Traditionally, the security-performance relationship is expected to be the path of the 2-heads "arrow" on the matrix above. That is, the better the firewall performs the poorer the security. By the same token, the poorer the firewall performs the better the security.

As refer to the design of the security testing, the seven security levels implemented in this project can be assigned with a security index from 1 to 7. As concluded from the results of the testing, the average latency values from the performance tests could be used to calculate the performance indices. In this way, each of the performance testing could be put into the security-performance

matrix proposed above. The security-performance relationship found in each of the testing for this project could be depicted with the security-performance matrices described below.

### 5.3.1 For performance tests of HTTP data transfer with 395K data

The performance indices were calculated by using the average latency as follows.

*The Average Latency for the transfer of 395Kb data using HTTP protocol*

	<u>x1</u>	<u>x10</u>	<u>x20</u>	<u>x30</u>	<u>x40</u>	<u>x50</u>	<u>x60</u>	<u>x70</u>	<u>x80</u>	<u>x90</u>	<u>x100</u>	<u>TL</u>	<u>Perf index</u>
<b>Cfg1</b>	0.94	1.04	1.11	1.05	1.29	1.22	1.31	1.23	1.23	1.23	1.43	13.08	<b>1.00</b>
<b>Cfg2</b>	1.00	1.30	1.51	1.36	1.47	1.46	1.51	1.50	1.47	1.39	0.50	14.46	<b>0.90</b>
<b>Cfg3</b>	1.50	6.39	15.22	17.13	16.33	16.74	16.90	17.16	16.89	17.32	17.11	158.69	<b>0.08</b>
<b>cfg4</b>	1.25	6.53	15.68	16.92	17.09	16.75	16.87	17.43	17.33	17.09	17.16	160.10	<b>0.08</b>
<b>Cfg5</b>	2.86	7.09	15.82	17.73	16.65	17.22	16.70	17.65	17.21	17.25	17.43	163.60	<b>0.08</b>
<b>Cfg6</b>	1.33	6.33	15.10	16.44	19.03	16.65	16.74	16.94	17.02	17.07	16.74	159.40	<b>0.08</b>
<b>Cfg7</b>	2.75	6.33	15.23	16.72	16.65	16.79	16.85	17.12	17.15	16.96	17.37	159.90	<b>0.08</b>

(Please refer to section 5.2.2.1 for the detail analysis of it )

The total latency for all the transactions under configuration 1 is used as the reference point X as it is assumed that the performance of firewall configuration 1 is the best and is assigned with performance index 1. So the calculation of the indices is as follows.

$$X = 13.08$$

- Configuration 1:  $1/(13.08/X) = 1$
- Configuration 2:  $1/(14.46/X) = 0.9$
- Configuration 3:  $1/(158.69/X) = 0.08 \sim 0.1$
- Configuration 4:  $1/(160.10/X) = 0.08 \sim 0.1$
- Configuration 5:  $1/(163.6/X) = 0.08 \sim 0.1$
- Configuration 6:  $1/(159.4/X) = 0.08 \sim 0.1$
- Configuration 7:  $1/(159.9/X) = 0.08 \sim 0.1$

In this way, the testing results in the project achieved the security-performance relationship: (1.1), (0.9,2), (0.1, 3), (0.1,4), (0.1,5), (0.1,6) and (0.1, 7)

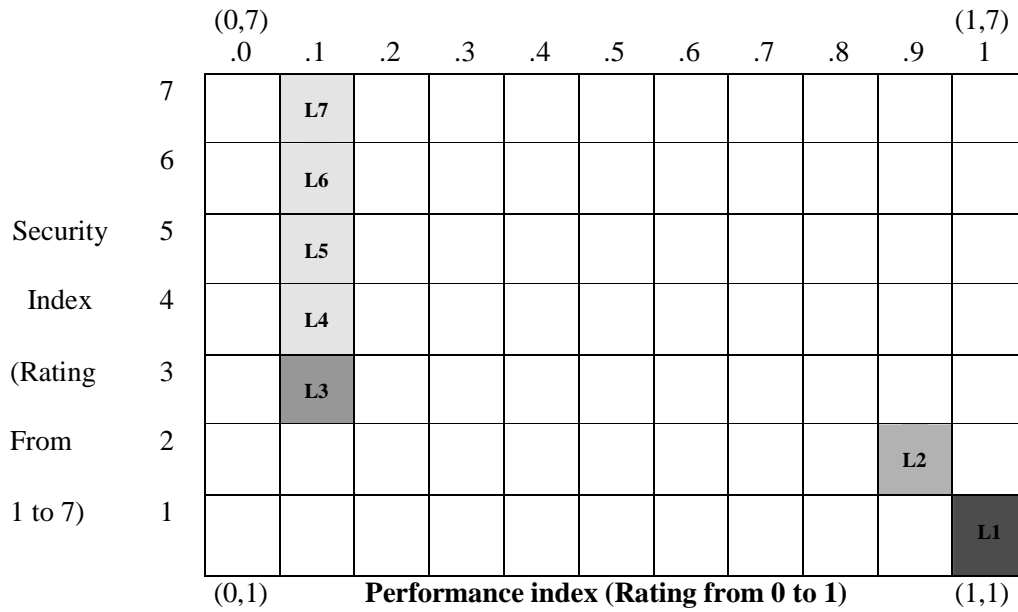


Figure 26 Security-performance matrix

### 5.3.2 For performance tests of FTP data transfer with 5M data

The performance indices were calculated by using the average latency as follows.

#### The Average Latency for the transfer of 5M data using FTP protocol

	<u>x1</u>	<u>x2</u>	<u>x3</u>	<u>x4</u>	<u>x5</u>	<u>x6</u>	<u>x7</u>	<u>x8</u>	<u>x9</u>	<u>x10</u>	TL	Perf. Index
Cfg1	11.13	13.93	14.67	14.14	14.29	14.81	15.30	14.84	14.96	14.43	142.51	<b>1.00</b>
cfg2	12.33	13.29	14.67	14.15	14.60	14.54	14.84	14.95	14.89	14.32	142.57	<b>1.00</b>
Cfg3	10.00	13.38	12.56	13.75	12.13	13.21	13.00	12.29	14.00	13.38	127.69	<b>1.12</b>
Cfg4	10.75	11.20	11.47	14.50	13.80	14.17	13.57	13.83	14.40	13.55	131.23	<b>1.09</b>
Cfg5	12.00	12.50	12.89	12.05	13.28	13.04	13.10	12.34	12.96	13.35	127.51	<b>1.12</b>
Cfg6	11.60	13.69	12.74	14.38	14.78	13.12	13.73	13.44	14.47	14.69	136.64	<b>1.04</b>
Cfg7	14.00	14.90	13.39	13.25	12.97	12.89	12.98	13.48	14.87	12.93	135.65	<b>1.05</b>

(Please refer to section 5.2.2.1 for the detail analysis of it )

The total latency for all the transactions under configuration 1 is used as the reference point X as it is assumed that the performance of firewall configuration 1 is the best and is assigned with performance index 1. So the calculation of the indices is as follows.

$X = 142.51$

- Configuration 1:  $1/(142.51/X) = 1$
- Configuration 2:  $1/(142.57/X) = 1$
- Configuration 3:  $1/(127.69/X) = 1.12 \sim 1$
- Configuration 4:  $1/(131.23/X) = 1.09 \sim 1$
- Configuration 5:  $1/(127.51/X) = 1.12 \sim 1$
- Configuration 6:  $1/(136.64/X) = 1.04 \sim 1$
- Configuration 7:  $1/(135.65/X) = 1.05 \sim 1$

Interesting, the testing results in the project achieved the security-performance relationship: (1,1), (1,2), (1, 3), (1,4), (1,5), (1,6) and (1, 7)

		(0,7)										(1,7)	
		.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	1	
Security Index (Rating From 1 to 7)	7											L7	
	6											L6	
	5											L5	
	4											L4	
	3											L2	
	2											L2	
	1											L1	
		(0,1)	Performance index (Rating from 0 to 1)										(1,1)

Figure 27: Security-performance matrix

### 5.3.3 For performance tests of ftp data transfer with 1M data

The performance indices were calculated by using the average latency as follows.

#### The Average Latency for the transfer of 1Mb data using FTP protocol

	<u>Cfg1</u>	<u>Cfg2</u>	<u>Cfg3</u>	<u>Cfg4</u>	<u>Cfg5</u>	<u>Cfg6</u>	<u>Cfg7</u>
1Mx1	3.00	3.67	3.67	2.78	3.60	3.00	3.13
1Mx2	3.94	3.67	3.40	3.40	3.94	3.58	3.63
1Mx3	4.15	3.73	3.87	3.70	3.96	4.13	3.86
1Mx4	4.00	4.11	4.00	3.95	3.81	4.34	3.97
1Mx5	4.18	4.17	4.04	3.96	3.91	4.18	4.06
1Mx6	4.29	4.36	4.23	3.96	4.02	4.02	4.10
1Mx7	4.57	4.24	4.29	3.94	4.16	4.16	4.00
1Mx8	4.54	4.59	4.15	3.96	4.00	4.15	4.20
1Mx9	4.38	4.49	4.38	4.12	4.01	4.30	4.13
1Mx10	4.23	4.40	4.02	4.16	4.07	4.24	4.16
TL	41.29	41.43	40.04	37.94	39.48	40.10	39.23
<b>Perf index</b>	<b>1.00</b>	<b>1.00</b>	<b>1.03</b>	<b>1.09</b>	<b>1.05</b>	<b>1.03</b>	<b>1.05</b>

(Please refer to section 5.2.2.1 for the detail analysis of it.)

The total latency for all the transactions under configuration 1 is used as the reference point X as it is assumed that the performance of firewall configuration 1 is the best and is assigned with performance index 1. So the calculation of the indices is as follows.

$$X = 41.29$$

Configuration 1:	$1/(41.29/X) = 1$
Configuration 2:	$1/(41.43/X) = 1$
Configuration 3:	$1/(40.04/X) = 1.03 \sim 1$
Configuration 4:	$1/(37.94/X) = 1.09 \sim 1$
Configuration 5:	$1/(39.48/X) = 1.05 \sim 1$
Configuration 6:	$1/(40.10/X) = 1.03 \sim 1$
Configuration 7:	$1/(39.23/X) = 1.05 \sim 1$

The result of it is similar to the testing of 5Mb data transfer using FTP protocol.

In this way, the testing results in the project achieved the security-performance relationship: (1,1), (1,2), (1,3), (1,4), (1,5), (1,6) and (1,7)



		(0,7)	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	(1,7)	1
Security Index (Rating From 1 to 7)	7													L7
	6													L6
	5													L5
	4													L4
	3													L3
	2													L2
	1													L1
		(0,1)	Performance index (Rating from 0 to 1)										(1,1)	

Figure 28: Security-performance matrix

### 5.3.4 For performance tests of FTP data transfer with 38.9K

The performance indices were calculated by using the average latency as follows.

#### The Average Latency for the transfer of 38.9Kb data using FTP protocol

	<u>Cfq1</u>	<u>Cfq2</u>	<u>Cfq3</u>	<u>Cfq4</u>	<u>Cfq5</u>	<u>Cfq6</u>	<u>Cfq7</u>
38.9k x 1	0.05	0.80	0.80	0.83	0.80	0.67	0.88
38.9k x 2	1.12	1.10	1.32	1.33	1.28	1.47	1.33
38.9k x 3	1.16	1.10	1.48	1.47	1.32	1.43	1.40
38.9k x 4	1.23	1.21	1.44	1.48	1.37	1.43	1.41
• 38.9k x 5	• 1.23	• 1.08	• 1.49	• 1.49	• 1.40	• 1.41	• 1.44
TL :	4.79	5.29	6.53	6.61	6.17	6.41	6.45
Perf. index	1.00	0.90	0.73	0.72	0.78	0.75	0.74

(Please refer to section 5.2.2.1 for the detail analysis of it.)

The total latency for all the transactions under configuration 1 is used as the reference point X as it is assumed that the performance of firewall configuration 1 is the best and is assigned with performance index 1. So the calculation of the indices is as follows.

X = 4.79

- Configuration 1:  $1/(4.79/X) = 1$
- Configuration 2:  $1/(5.29/X) = 0.9$
- Configuration 3:  $1/(6.53/X) = 0.73 \sim 0.7$  (ignore the 2<sup>nd</sup> decimal digit)
- Configuration 4:  $1/(6.61/X) = 0.72 \sim 0.7$  (ignore the 2<sup>nd</sup> decimal digit)
- Configuration 5:  $1/(6.17/X) = 0.78 \sim 0.7$  (ignore the 2<sup>nd</sup> decimal digit)
- Configuration 6:  $1/(6.41/X) = 0.75 \sim 0.7$  (ignore the 2<sup>nd</sup> decimal digit)
- Configuration 7:  $1/(6.45/X) = 0.74 \sim 0.7$  (ignore the 2<sup>nd</sup> decimal digit)

In this way, the testing results in the project achieved the security-performance relationship: (1,1), (0.9,2), (0.7, 3), (0.7,4), (0.7,5), (0.7,6) and (0.7, 7)

		(0,7)										(1,7)	
		.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	1	
Security Index (Rating From 1 to 7)	7								L7				
	6								L6				
	5								L5				
	4								L4				
	3								L3				
	2										L2		
	1											L1	
		(0,1)	Performance index (Rating from 0 to 1)										(1,1)

Figure 29: Security-performance matrix

## 6.LIMITATIONS

### 1. Outside interference to performance testing

With a view to simulating the data transfer process as close to real cases as possible, tests were run under real working environment, which is not disconnected ourselves from the rest of the world. Documents or data files used in data transfer by using http or ftp protocols are all located at the outside public area, i.e. the ftp server of the CS department (<ftp.cse.cuhk.edu.hk>) and the pc89136. However, the testing environment, in this way, would become out of our control.

In order to minimize the risk of interference from outside traffic on the results, most of the testing were carried during the night times and the quiet times of the day of testing. But it is still difficult to know the condition of outside traffic and how busy the public ftp server is, results with abnormal and inconsistent values under the same configuration would sometime obtained (as seen in the pattern of raw data values in Appendix C). In this case, many more tests have to be repeated in order to obtain a smoother curve. In this way more effort has to be made in eliminating the interference from outside traffic.

Certainly the time for testing in this project is not enough and repeated testing in achieving smooth curves for the comparison of network performance among security level 3 to 7, is a problem.

### 2. Security level definition for firewall

In this project, it is attempted to define the 7 levels of security by the implementation of 7 different firewall policies, which are supposed to impose different levels of restriction and security controls on the network. However, "security" itself is very difficult to ensure and defined. Not only technically sound design and protection, human co-operation to conform the published guideline and policies is very important in achieving the expected "security level". Any minor security loophole could ruin all the effort previously paid in guarding the firewall system and downgrade security level easily. Even though "security level" could be quantified as an index and be measured with respect to "tolerance to some particular attacks", it is only true at the time it is tested. It is due to the fact that the Internet changes very often, new bugs, new Trojan horses, technologies as well as hacking techniques evolves over the time quickly, no one can guarantee that your well-designed and protected system is 100 % secured today, would not be broken into by intruders tomorrow. In a nutshell, more considerations have to be made in defining an achievable and reliable security level.

## **7.FUTURE WORK**

With regards to the data sets and the limitations mentioned above, possible research opportunities are as follows.

### 1. More repeated testing on different size of data

In this project, only data size of 1M, 5M and 38.9K for data transfer using ftp protocol and of 395K for data retrieval by using http protocol are attempted. It is recommended that more different size of data and number of connections requests are tested. This is to determine the threshold value about the data size and number of connection, above these values the network performance would be significantly affected. Of course, it may have to be associated with a particular firewall configuration or feature such that the result could be more meaningful.

In this project, time for testing is really limited and for sure it is inadequate for repeating the testing many times until the smooth curves of results are obtained. So future work could be made on fine tuning the testing parameters such as the data size, number of connections, data transfer protocols used, and achieving a smooth curve on the results by performing intensive and more repeated testing.

In order to eliminate the interference from outside traffic, testing of file transfer could be done under a more stable network environment in which an isolated or a less busy the ftp or http server could only be employed for the testing.

### 2. Security of seven levels

As seen in the result, concerning the difference of performance under security level 3 to 7, it is difficult is little to conclude. In the future work, it is recommended that the 7 levels of security should be redesigned such that their level of security would differ significantly from each others and the incorporated security measures for the different 7 security levels should weigh more in security checking on data traffic. In other words, stronger authentication protocol for communication and strong cryptography for protecting all transmitted confidential data, including passwords, binary files, and administrative commands, could be added.

For example, security level 3 should be adopted with a more advanced authentication protocol, such as SSH 2 , with the "secure ftp" for data transfer communication. SSH 2 is based on SSH (Secured

Shell) and SSH2 is intended as a complete replacement for ftp, telnet, rlogin, rsh, rcp, and rdist, please refer to <http://www.ssh.fi/sshprotocols2/specs.html> for more details.

In this way, the difference of security features among different secured level is enlarged, the performance overhead should be more or less be affected. But of course, many factors such as technology used, firewall configuration and setup, will also matter when measuring firewall performance.

### 3. Security VS performance relationship

As discussed in section 5.3, an interesting security-performance matrix is derived and it shows us some implication about how much combination of the security and performance could be.

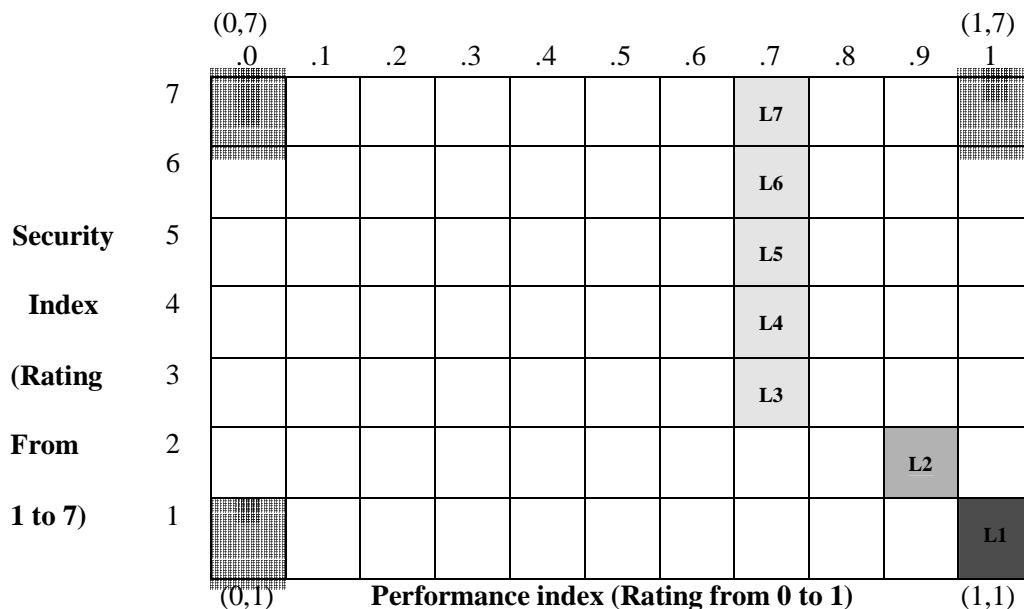


Figure 30: Security-performance matrix

Notes for the coordinate x,y : (where x is from 0 to 1, y is from 1, 7 )

- 0, 7 poorest performance, highest security
- 1, 7 best performance, highest security
- 0, 1 poorest performance, lowest security
- 1, 1 best performance, lowest security

The relationships colored with red, yellow, blue and green in the matrix have already been implemented in this project. Research on the security-performance relationship of (1,7), (1,7), (0,0) that were not covered in this project, is recommended in the future work.

## **8. CONCLUSION**

For all of the testing on the firewall system implemented with the seven proposed firewall policies, the performance of firewall has been analyzed and quantified. In the security testing, the security levels were not only built up qualitatively with different security policies, but also tested and validated by using some network scanning tools such that the higher level security delivers less security control and is more secured than the lower security level theoretically and practically.

In the performance testing, for the scenarios of data transfer of small data size and high volume of HTTP or FTP connection requests, the firewall is capable of showing some performance difference under the implementation of different firewall policies. It is found that the security level 1 which was incorporated with no screening rule and the least security control as specified in the firewall policy 1, performed better than security level 2, which was more secured than level 1 with the setting of screening rules. Likewise, security level 2 performed better than level 3, which was more secured than level 2 and installed with proxy services at the firewall. For each connection request to the Internet, the overhead from more security would be incurred. When there are frequent connection requests to the Internet, the accumulated overhead for the completion of all the requests would be substantial, especially when the data file for transfer is small.

However, since the implementation of security level 4, 5, 6 and 7, the impact from the interference of outside traffic dominated the performance influences incurred by more security. Only fluctuations of performance values for the security levels have been observed. Consequently, it is difficult to conclude whether the security level of one policy is more or less secured than that of others under firewall security policy 4, 5, 6 and 7.

As seen from the overall result of testing, it is obvious that the firewall performance will actually be affected with more security only if the overhead incurred by the added security control and measures is significant enough when compared with the normal transaction time without the added security control. As long as the accumulated overhead, which is induced from the addition of a particular security control, could outweigh the interference from outside traffic and the traffic processing time without the added security, performance degradation would result. Moreover, it can be confirmed that the increased security have to scarify network performance with respect to data transfer by using the FTP and HTTP protocols in the project.

Furthermore, a security to performance matrix is proposed in the project. This matrix showed us the various combinations of security to performance relationship and could be used to illustrate the traditional relationship of security and performance, that is, the better the security, the poor the performance. This relationship could be applied to the security level 1,2, and 3 implemented in this project. It is mainly due to the overhead added by more security control with respect to higher level of security. However, it also depends very much on the way the added security is incorporated into the system. For example, the security level 4 and 5, which were implemented by screening rules set into a router and did not obviously differ from one another in performance. Actually the numbers of screening rules used in level 4 and 5 are ver close, it is very likely that the added security in level 5 only incurred a little more overheads than that of level 4.

These all are very interesting results and further studies using other security measures for the definition of various security levels and the exploration of security to performance relationship in the non-traditional aspects are recommended.

## 9. REFERENCES

1. A. Molitor. *Measuring Firewall Performance*. Network System Corporation.
2. M. J. Ranum. *On the Topic of Firewall Testing*. 1995
3. [Mjr@v-one.com](mailto:Mjr@v-one.com) . *Firewall Performance Measurement techniques: A scientific approach*. Available at <http://www.clark.net/pub/mjr/pubs/fwperf/intro.html>
4. OUTLINK. *The Market Research Process and Methodology*. 1997. Outlink, Inc. Available at <http://outlink.com/fireanal.html>
5. C. Kostick and M. Mancuso. *Firewall Performance Analysis Report*. August 1995. Computer Science Corporation CSC.
6. B. Cheswick. *The Design of a Secure Internet Gateway*. 1990. At&T Bell Laboratories, Murray Hill, New Jersey.
7. R. Farrow. *RIK Farrow's 1997 Firewall Product Analysis - An Analysis of Current Firewall Technologies*. 1997. Available at <http://www.gocsi.com/farrowpa.html>
8. D. Farmer and W. Venema. *Improving the Security of Your Site by Breaking Into it*. Appeared as Admin-guide-to-cracking\_101 document at
9. D. Newman. *Super Firewalls - Lab Tests*. Data Communications . 21 May 1999. Available at : <http://www.data.com/issue/990521/firewalls.html>
10. E. E. Schultz. *How to Perform Effective Firewall Testing*. Computer Security Journal. Spring 1996.
11. V. N. Padmanabhan and J. C. Mogul. *Improving HTTP Latency*. 1994. Available at : <http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/Dday/mogul/HTTPLatency.html> .
12. E. Jonsson and T. Olovsson. *A Quantitative Model of the Security Intrusion Process Based on Attacker Behavior*. IEEE Transaction on Software Engineering. Vol. 23 No. 4. April 1997.
13. A. K. Ghosh, J. Wanken and F. Charron. *Detecting Anomalous and Unknown Intrusions Against Programs*. Computer Security Applications Conference, 1998 Proceedings. P259-267. 1998
14. M. Greenwald, S. K. Singhal, J. R. Stone and D. R. Cheriton. *Designing an Academic Firewall: Policy, Practice, and Experience With SURF*. Network & Distributed System Security 1996. 1996.
15. R. Bace. *An Introduction to Intrusion Detection - ASSESSMENT*. The security assurance company ICSA Inc. 1999.



16. D. Wreski. *Linux Security Administrator's Guide*. V0.98, 22 August 1998. Available at <ftp://sunsite.unc.edu/pub/Linux/doc/HOWTO> or <http://sunsite.unc.edu/LDP/>
17. S. W. Lodin and C. L. Schuba. *Firewalls fend off invasions from the Net*. *IEEE Spectrum*, vol 352, p26-34. Feb 1998.
18. Compaq Computer Corporation. Performance Analysis and Tuning of Raptor's Eagle NY 3.06 Firewall on Compaq Servers. April 97.
19. D.E. Denning and P.J. Denning. *Internet Besieged – Countering Cyberspace Scofflaws*. Addison Wesley, 1998.
20. M. Goncalves. *Firewalls*. McGraw-Hill, 1998.
21. J.R. Vacca, *Internet Security SECRETS*. IDG Books World, Inc. 1996.
22. A.K. Dippel, *Authentication of Computer Communications*. Available from : <http://www.cs.indiana.edu/~www/hyplan/adippel/authent.html#SEVEN>. May, 1996.
23. M.J. Ranum. *Thinking About Firewalls*. Trusted Information Systems, Inc. Glenwood, Maryland. 1998.
24. B. Lampson et al. *Authentication in Distributed Systems: Theory and Practice*. *ACM Trans. Computer Systems*. Nov. 1992, pp.265-310.
25. A. D. Rubin, D.E. G. Jr. *A Survey of Web Security*. *IEEE Computer*. 1998, pp 34-41.
26. A.K. Dippel. *Authentication of Computer communications*. Available from: <http://whatis.com/digitace.htm>
27. K. Pagan and S. Fuller. *Intranet Firewalls – Planning & Implementing Your Security System*. VENTANA. 1997.
28. D. B. Chapman and E. D. Zwicky. *Building Internet Firewalls*. O' Reilly. 1995.
29. M. Grennan. *Firewalling and proxy Sever HOWTO*, 8 Nov 1996.
30. Avolio and Blask. *Application Gateways and Stateful Inspection: A Brief Note Comparing and Contrasting*. Trusted Information System, Inc. 22Jan1998.
31. C. H. Rowland. *Network Attack Trend Analysis*. 19Nov1997. ( web address: [http://www.geek-girl.com/bugtraq/1997\\_4/0352.html](http://www.geek-girl.com/bugtraq/1997_4/0352.html) )
32. *Stateful Inspection Firewall technology*. Check Point Co. 1998. Available from <http://www.checkpoint.com/products/technology/stateful1/html>
33. P. Galvin. *Distinguish firewall hype*. SunWorld Online. Dec 1996. Available from <http://www.sunworld.com/sunworldonline/swol-12-1996/swol-12-security.html>

34. Z. Zhang. Firewalls – An Internet Security Approach. 1995 Available from [http://erebus.bentley.edu/students/z/zhang\\_zhih/firwal95/tsld001.htm](http://erebus.bentley.edu/students/z/zhang_zhih/firwal95/tsld001.htm)
35. L. Lin. Intranet Security. 1998-99. Available from <http://www.isaca.org/art6.htm> ..
36. M. J. Ranum and M. Curtin. Internet Firewalls Frequently Asked Questions. Technical Incursion Countermeasures. May 1998. Available from [http://www.ticm.com/kb/faq/faqfw.html#head\\_types](http://www.ticm.com/kb/faq/faqfw.html#head_types) .

## 10. APPENDICES

### **APPENDIX A - firewall policies implementation by screening rules**

#### **1. Firewall Policy 1**

- i. Policy - Permit any service unless it is expressly denied
  - Provide the maxi flexibility/access for both internal and external users.

No screening rules are set into the router. The router is only responsible for routing traffic to the firewall. Also no proxy service is set at the proxy server.

ipchains route is set with the rule :

```
ipchains -A input -s any -i ! eth0 -j DENY
```

- ii. Proxy Services: Nil

#### **2. Firewall Policy 2**

- i. Policy - Permit any service unless it is expressly denied (same as 1)
  - Disallow some problem service accesses from outside, but still provide flexible/easy access from outside, but no restriction on access from internal network to the Internet.

The list of rules set into the router:

---

```
access-list 100 deny    udp any host 137.189.89.250 eq tftp
access-list 100 deny    tcp any host 137.189.89.250 eq 97
access-list 100 deny    tcp any host 137.189.89.250 eq sunrpc
access-list 100 deny    udp any host 137.189.89.250 eq sunrpc
access-list 100 deny    tcp any host 137.189.89.250 eq 2049
access-list 100 deny    tcp any host 137.189.89.250 eq lpd
access-list 100 permit ip  any any
```

---

\* 100 means the list list 100

The list of rules set into the firewall server, pc89250:

---

```
ipchains -A input -s 192.168.168.0/27 -i ! eth2 -j DENY
ipchains -A input -s 192.168.168.32/27 -i ! eth1 -j DENY
```

---

```
! ipchains -A input -s any -i ! eth0 -j DENY (removed)
```

---

- IP source routing is disabled by the linux kernel.
- IP spoofing is prevented by the rules set into pc89250 as shown above.
- Disabling of the selected services is achieved by the rules set into the router as shown above.

- **IP Masquerader** is set up such that the workstations inside the private network could access the outside net, with IP being translated at the gateway.

ii. Proxy Services: Nil

### 3. Firewall Policy 3

- i. Policy - Permit any service unless it is expressly denied (same as configuration 1)
  - An additional protection is added with 'proxy service' enabled in the firewall server. Specific traffic is further shielded and screened with the proxy server installed.
- The list of rules set into the router remained as that of configuration 2.
- No more screening rule is added for it. Screening rule setting is the same as that for configuration 2.
- Proxy services is enabled with TELNET/FTP/HTTP/WWW/SMTP/DNS/X-WINDOWS

### 4. Firewall Policy 4

- i. Policy - PERMIT any service unless it is expressly denied (same as configuration 1)
  - Allow even more restricted access from outside, and deny from selected bad HOSTs from outside.
- ii. Policy Setting Details

List of screening rules set into the router (Total 26 rules)

---

Phase 4:

```
access-list 101 deny    udp any host 137.189.89.250 eq tftp
access-list 101 deny    tcp any host 137.189.89.250 eq 97
access-list 101 deny    tcp any host 137.189.89.250 eq sunrpc
access-list 101 deny    udp any host 137.189.89.250 eq sunrpc
access-list 101 deny    tcp any host 137.189.89.250 eq 2049
access-list 101 deny    tcp any host 137.189.89.250 eq lpd
access-list 101 deny    tcp any host 137.189.89.250 eq ftp-data
access-list 101 deny    udp any host 137.189.89.250 eq 20
access-list 101 deny    tcp any host 137.189.89.250 eq 6000
access-list 101 deny    udp any host 137.189.89.250 eq 6000

access-list 101 deny    ip 137.189.88.0 0.0.0.128 host
137.189.89.250
access-list 101 deny    ip host 195.92.23.250 host 137.189.89.250
access-list 101 deny    ip host 195.92.23.251 host 137.189.89.250
access-list 101 deny    ip host 208.232.1.130 host 137.189.89.250
access-list 101 deny    ip host 208.232.1.127 host 137.189.89.250
access-list 101 deny    ip host 207.44.192.2 host 137.189.89.250
access-list 101 deny    ip host 209.133.111.124 host 137.189.89.250
access-list 101 deny    ip host 209.235.107.136 host 137.189.89.250
access-list 101 deny    ip host 12.10.107.5 host 137.189.89.250
access-list 101 deny    ip host 199.60.229.31 host 137.189.89.250
access-list 101 deny    ip host 203.85.221.120 host 137.189.89.250
```

```
access-list 101 deny ip 207.89.178.0 0.0.0.255 host
137.189.89.250
access-list 101 deny icmp any host 137.189.89.250
access-list 101 permit tcp any host 137.189.89.250
access-list 101 permit udp any host 137.189.89.250
access-list 101 permit ip any any
```

---

- Proxy services is enabled with TELNET/FTP/HTTP/FINGER/RLOGIN

## 5. Firewall Policy 5

- i. Policy - DENY any service unless it is expressly permitted.  
(or we say "that is not expressly permitted is prohibited")  
- Deny all access from outside by default, but allow access from inside and provide the best possible services to the internal network, by permitting some selected services going into the network.

- ii. Policy Setting Details

List 101 handles traffic from the ROUTER to PC89250.

List 102 handles traffic from PC89250 to the ROUTER.

any: any host

host \*.\*.\*.\*: the specific host.

eq: equal

\*\*\* Cisco Router append a "deny all any any" at the end of all  
access-list group.

List of rules set into the router (Total 29 rules including "deny all any any")

---

Phase 5:

```
access-list 101 deny icmp any host 137.189.89.250

access-list 101 deny ip 137.189.88.0 0.0.0.128 host
137.189.89.250
access-list 101 deny ip host 195.92.23.250 host 137.189.89.250
access-list 101 deny ip host 195.92.23.251 host 137.189.89.250
access-list 101 deny ip host 208.232.1.130 host 137.189.89.250
access-list 101 deny ip host 208.232.1.127 host 137.189.89.250
access-list 101 deny ip host 207.44.192.2 host 137.189.89.250
access-list 101 deny ip host 209.133.111.124 host 137.189.89.250
access-list 101 deny ip host 209.235.107.136 host 137.189.89.250
access-list 101 deny ip 207.89.178.0 0.0.0.255 host
137.189.89.250
access-list 101 deny ip host 199.60.229.31 host 137.189.89.250
access-list 101 deny ip host 12.10.107.5 host 137.189.89.250
access-list 101 deny ip host 203.85.221.120 host 137.189.89.250

access-list 101 deny tcp any host 137.189.89.250 eq ftp-data
```

---

```
access-list 101 deny tcp any host 137.189.89.250 eq 97
access-list 101 deny tcp any host 137.189.89.250 eq sunrpc
access-list 101 deny tcp any host 137.189.89.250 eq lpd
access-list 101 deny tcp any host 137.189.89.250 eq 2049
access-list 101 deny tcp any host 137.189.89.250 eq 6000
* access-list 101 permit tcp any host 137.189.89.250
access-list 101 deny udp any host 137.189.89.250 eq tftp
access-list 101 deny udp any host 137.189.89.250 eq sunrpc
access-list 101 deny udp any host 137.189.89.250 eq 20
access-list 101 deny udp any host 137.189.89.250 eq 6000
* access-list 101 permit udp any host 137.189.89.250

access-list 101 permit ip any host 137.189.89.250

access-list 102 permit ip host 137.189.89.250 any

access-list 101 deny ip any any
access-list 102 deny ip any any
```

---

- Proxy services is enabled with TELNET/FTP/HTTP/FINGER/RLOGIN

## 6. Firewall Policy 6

- i. Policy - DENY any service unless it is expressly permitted.  
- A more restricted policy to permit outside access to certain port numbers range only.
- ii. Policy Setting Details

Total 37 screening rules .

---

Phase 6:

```
access-list 101 deny icmp any host 137.189.89.250
access-list 101 deny ip 137.189.88.0 0.0.0.128 host
137.189.89.250
access-list 101 deny ip host 195.92.23.250 host 137.189.89.250
access-list 101 deny ip host 195.92.23.251 host 137.189.89.250
access-list 101 deny ip host 208.232.1.130 host 137.189.89.250
access-list 101 deny ip host 208.232.1.127 host 137.189.89.250
access-list 101 deny ip host 207.44.192.2 host 137.189.89.250
access-list 101 deny ip host 209.133.111.124 host 137.189.89.250
access-list 101 deny ip host 209.235.107.136 host 137.189.89.250
access-list 101 deny ip 207.89.178.0 0.0.0.255 host
137.189.89.250
access-list 101 deny ip host 199.60.229.31 host 137.189.89.250
access-list 101 deny ip host 12.10.107.5 host 137.189.89.250
access-list 101 deny ip host 203.85.221.120 host 137.189.89.250

access-list 101 deny tcp any host 137.189.89.250 eq ftp-data
access-list 101 deny tcp any host 137.189.89.250 eq 97
access-list 101 deny tcp any host 137.189.89.250 eq sunrpc
access-list 101 deny tcp any host 137.189.89.250 eq lpd
```

---

```
access-list 101 deny tcp any host 137.189.89.250 eq 2049
access-list 101 deny tcp any host 137.189.89.250 eq 6000
* access-list 101 permit tcp any lt 1024 host pc89250 lt 1024

* access-list 101 permit ip host 137.189.89.136 host 137.189.89.250
* access-list 101 permit tcp any gt 1023 host 137.189.89.250 eq ftp
* access-list 101 permit tcp any gt 1023 host 137.189.89.250 eq ftp-data
* access-list 101 permit tcp any gt 1023 host 137.189.89.250 eq telnet
* access-list 101 permit tcp any gt 1023 host 137.189.89.250 eq 24 eq 513
eq 514
* access-list 101 permit tcp any gt 1023 host 137.189.89.250 eq smtp
* access-list 101 permit tcp any gt 1023 host 137.189.89.250 eq domain
* access-list 101 permit tcp any gt 1023 host 137.189.89.250 eq www

access-list 101 deny udp any host 137.189.89.250 eq tftp
access-list 101 deny udp any host 137.189.89.250 eq sunrpc
access-list 101 deny udp any host 137.189.89.250 eq 20
access-list 101 deny udp any host 137.189.89.250 eq 6000

* access-list 101 permit udp any gt 1023 host pc89250 eq 123
* access-list 101 permit udp any gt 1023 host pc89250 eq 53,54,80

access-list 102 permit ip host 137.189.89.250 any

access-list 101 deny ip any any
access-list 102 deny ip any any
```

---

- Proxy services is enabled with TELNET/FTP/HTTP/FINGER/RLOGIN

## 7. Firewall Policy 7

- ii. Policy - DENY any service unless it is expressly permitted.
  - Provide the least flexibility and services to the internal users, but incorporate maxi protection on the LAN. The internal users are no longer freely access any Internet services as users are restricted to access of authorized hosts.

### iii. Policy Setting Details

There is total 43 rules set into the router

```
-----
Phase 7
access-list 101 deny icmp any host 137.189.89.250
access-list 101 deny tcp any host 137.189.89.250 eq ftp-data
access-list 101 deny tcp any host 137.189.89.250 eq 97
access-list 101 deny tcp any host 137.189.89.250 eq sunrpc
access-list 101 deny tcp any host 137.189.89.250 eq lpd
access-list 101 deny tcp any host 137.189.89.250 eq 2049
access-list 101 deny tcp any host 137.189.89.250 eq 6000

access-list 101 deny udp any host 137.189.89.250 eq tftp
```

```
access-list 101 deny    udp any host 137.189.89.250 eq sunrpc
access-list 101 deny    udp any host 137.189.89.250 eq 20
access-list 101 deny    udp any host 137.189.89.250 eq 6000
access-list 101 permit  udp any gt 1023 host pc89250 eq 123
access-list 101 permit  udp any gt 1023 host pc89250 eq 53,54,80

access-list 101 permit  ip host solar15 host pc89250
access-list 101 permit  ip host solar23 host pc89250
access-list 101 permit  ip host sparc53 host pc89250

access-list 102 deny    tcp host 137.189.89.250 host sparc54 eq 23
access-list 101 permit  ip host sparc54 host pc89250
access-list 101 permit  ip host venture host pc89250
access-list 101 permit  ip host cucs18 host pc89250
access-list 101 permit  ip host linux1 host pc89250
access-list 101 permit  ip host linux2 host pc89250
access-list 101 permit  ip host linux3 host pc89250
access-list 101 permit  ip host linux4 host pc89250
access-list 101 permit  ip host linux5 host pc89250
access-list 101 permit  ip host linux6 host pc89250
access-list 101 permit  ip host linux7 host pc89250
access-list 101 permit  ip host linux8 host pc89250
access-list 101 permit  ip host linux9 host pc89250

access-list 101 permit  ip host garden host pc89250
access-list 101 permit  ip host beryl host pc89250
access-list 101 permit  ip host www host pc89250
access-list 101 permit  ip host pc89136 host pc89250
access-list 101 permit  ip host ftp host pc89250
access-list 101 permit  ip host 137.189.172.198 host pc89250
access-list 101 permit  ip host 143.89.40.4 host pc89250
access-list 101 permit  ip host 137.189.6.37 host pc89250
access-list 101 permit  ip host 147.8.179.15 host pc89250
access-list 101 permit  ip host 144.214.5.246 host pc89250
(implicit: deny all, as "deny all ip" would be appended at the end
of list)

access-list 102 deny    udp host 137.189.89.250 any eq tftp
access-list 102 permit  ip host 137.189.89.250 any

(Appended at the end of the list 101/102)

access-list 101 deny    ip any any
access-list 102 deny    ip any any
```

---

- Proxy services is enabled with TELNET/FTP/HTTP/FINGER/RLOGIN



## ***APPENDIX B – Plugin List of Nessus***

## Appendix C – Raw Data Set

### FTP - Sequential FTP Testing with 5.0M data

#### Config 1 FTP Seq

x15	3	12	x14	12	12	13	12	13	12
24	34	x32	26	30	27	x32	x32	28	26
43	x60	51	39	43	39	43	39	50	49
55	x65	x67	57	57	60	55	52	60	x65
72	70	70	77	70	x84	65	73	71	75
98	96	87	89	x100	75	95	88	x102	83
112	111	113	105	115	100	115	92	x138	101
116	118	120	x133	121	117	x133	106	117	135
x144	x149	141	x159	131	135	x144	127	134	140
x163	144	x148	x157	147	145	x148	144	146	140

#### Config 2 FTP

##### Seq

12	13	X17		x16	x29		12 x17	x15	
28	27		26		29	28	23	25 x37	
42	43	X52		x55		42	47	47	43
57	X66		x62		59	57 x63		52	58
68	X85		79		77 x78		73	74	67
X97		87	x93		86 x96		85 x96		91
102	106		102		109	100 x119		104	104
115	122	X 123			120	120 x123		121 x123	
127	X134		129		138	138	138 x147	x174	
143	144		145		x150	145	135	147 x151	

#### Config 3 FTP Seq

x38	X120		x103		10	x110	13	9	8
x32	X243		x231		23	x203	23	31	30
x58	X333		X356		35	x336	x58	35	43
x77	X482		X468		60	x443	x63	44	61
x108	x212		X576		55	x387	63	64	x79
x115	x131		X684		79	90	75	x108	73
x159	x177		X863		96	94	76	98	x109
x185	x213		X985		103	91	x114	101	x109
x175	x225		X1037		119	130	x134	x135	129
x229	x208		X1102		x152	135	137	132	131

#### Config 4 FTP Seq

x16		9	x101			11 x130		10 x15		13
x63.5		22	x218.8			25 x252		23	23	19
x268.7		27	x294.1			37 x351		43	36	29
x373.5		65	x353.5			64 x367		49 x67		54
x443.2		64	x481.7		x82	x597		74	68	70
x548.1		85	x616			87 x703	x88		87	81
x658.5		96	x700		x106	x825		87	92	105

x812.2	120	x1045	109 x730	111	108	105
x913.2	126	x1178	139 x143	121	123	139
x1010.9	x165	x1312	135 x122	141	134	132

**Config 5 FTP Seq (1x5M to 10x5M)**

x129	x15	x15	14	13	9 x15	
x212	x30		26	27	25	27 20
x344	x44		40	38 x42	x43	38
x510		43 x61	51	51	45	51
x636		69	64	68	69 x74	62
x598		80 x99		77	81	75 x92
x279	x109		87	94 x100		94 x99
x308		96	86	116 x120	x125	97
x365	x129		120	115	117	114 117
x392		134	133	130	136	135 133

**Config 6 FTP Seq**

15	9.3	10.5	11.60	11.60
26.8	28.71	26.65	27.39	13.69
40.18	45.3	29.22	38.23	12.74
56.6	53.92	62	57.51	14.38
74	71.3	76.33	73.88	14.78
72.06	79.06	85	78.71	13.12
95.26	96.09	96.9	96.08	13.73
98.57	114.56	109.41	107.51	13.44
137.43	125.37	128	130.27	14.47
145.85	147.55	147.28	146.89	14.69
			Min lat	11.60
			Avg lat	13.66

**Config 7 FTP Seq**

x16		15 x74	13	15	13
27		31 x78	30	31	30
41		44	40	44	40
55		52	55	52	49
65		63	69	63	57
78		78	81	78	77
96		97	95	97	84
102		114	112	114	108
114		155	118	155	135
130		119	146	119	129

**FTP - Sequential FTP Testing with 38.9Kb data**

**Config 1 FTP Seq - 0.3M**

x0.05	0.05	0.05	0.05	0.05	0.05
x6	6.00	6.00	5.00	6.00	5.00
x11	11.00	12.00	12.00	12.00	11.00
x25	25.00	25.00	25.00	23.00	25.00
x1268	x1268	49.00	48.00	50.00	x650

**Config 2 FTP Seq - 0.3M**

0.50	1.00	1.00	1.00	0.50	x1
5.00	5.00	5.00	5.00	7.00	6.00
12.00	11.00	11.00	x14	10.00	11.00
26.00	x47	24.00	24.00	22.00	24.00
				25.00	25.00

**Config 3 FTP Seq - 0.3M**

1.00	1.00	0.50	1.00	0.50
6.00	6.00	7.00	7.00	7.00
14.00	15.00	15.00	15.00	15.00
29.00	31.00	28.00	28.00	28.00
57.00	63.00	59.00	59.00	59.00
91.00	92.00	89.00	x683	x682
x713	x712	x712	x	x713

**Config 4 FTP Seq - 0.3M**

1.00	1.00	0.50
7.00	6.00	7.00
15.00	15.00	14.00
29.00	29.00	31.00
58.00	60.00	61.00

**Config 5 FTP Seq - 0.3M**

1.00	1.00	0.50	0.50
6.00	6.00	7.00	7.00
13.00	13.00	13.00	14.00
28.00	27.00	29.00	28.00
x176			64.00
x1554	x685	x864	x625
x711	x711	x708	x1313

**Config 6 FTP Seq - 0.3M**

0.50	1.00	0.50
8.00	7.00	7.00
13.00	14.00	16.00
28.00	28.00	30.00
58.00	x64	55.00

**Config 7 FTP Seq - 0.3M**

1.00	0.50	0.50	1.00	1.00	1.00	1.00	1.00	1.00	x
6.00	6.00	7.00	7.00	7.00	6.00	7.00	7.00	7.00	x
15.00	14.00	13.00	14.00	15.00	15.00	13.00	13.00	13.00	x
25.00	28.00	28.00	29.00	28.00	28.00	29.00	29.00	29.00	x
59.00	56.00	57.00	58.00	58.00	59.00	58.00	57.00	x58	57.00

**FTP - Sequential FTP Testing with 1Mb data**

**Config 1 FTP Seq - 1M**

x6	3	4	x6	2	4	3	3	3	2
9	7	10	6	7	9	x12	8	8	7
12	x18	15	12	13	11	13	13	11	12
x20	x19	15	16	14	x20	17	18	17	15
21	19	22	19	22	21	21	x26	21	22
24	29	27	27	20	24	x31	x31	28	27
33	30	34	x38	32	33	31	32	31	32
37	35	37	36	x42	35	35	39	35	38
x45	x45	41	40	39	35	43	41	x45	37
44	46	x48	41	x51	x47	41	x54	43	39

**Config 2 FTP Seq - 1M**

x7	3 -	x5	x6		4	4
7 x13		8	5	8	8	8
12 x16		8	13	13	10 x15	
15	16	14	16	18	18	18
20	25	21	21	23	18	18
26 x28		27	27	26	24	27
32	25	30	25	32	28	36
35	34	39	33	42	38	36
42	38	40	39	43	39	42
46	42	40	46	46	46	42

**Config 3 FTP Seq - 1M**

x5	4	4 x5		3
6	6	7	7	8
11	11	13	11	12
17	17	14	17	15
21	21	21	18	20
27	27	26	26	21
28	32	30	30	30
33	33	34	29	37
37	38	42	36	44
42	43	37	38	41

**Config 4 FTP Seq - 1M**

3	3	2	2	4	2	4	3	2	x5
8	8	6	6	6	6	7	7	7	7
10	10	12	12	12	11	11	11	11	11
19	x17	x17	15	14	15	x17	16	x17	x17
x21	19	x22	21	x22	x23	20	19	20	x21
22	25	x26	25	24	22	25	21	25	25
25	28	29	31	27	28	28	28	27	25
31	31	32	32	32	31	32	32	33	31
36	35	37	37	39	35	37	39	x40	39
39	41	43	45	42	41	40	41	42	x44

**Config 5 FTP Seq - 1M**

6	3	3	3	3	4	3	4	3	4
x12	7	8	8	8	10	7	8	8	7
x14	11	13	16	11	10	13	11	11	11
x23	15	15	16	14	18	15	15	15	14
x22	19	22	20	18	19	20	17	20	21
x34	x28	23	25	25	24	25	26	20	25
x43	29	29	29	29	30	31	28	30	27
x47	30	28	33	31	32	33	32	33	36
x41	36	36	38	37	38	32	36	34	38
39	x44	39	41	42	41	37	42	45	40

**Config 6 FTP Seq - 1M (sample data 9 is not used)**

x5	3	3	2	3	4	4	x7	2
8	x9	8	x9	7	7	5	x10	8
13	11	13	12	12	x14	12	13	13
	22	18	17	17	18	15	15	17
20	25	21	18	24	20	21	19	20

24	26	26	22	21	27	24	25	22
28	28	27	30	27	28	34	33	27
30	40	35	30	36	32	34	32	30
38	x51	37	39	42	37	37	41	x57
44	38	45	40	x50	43	44	39	46

**Config 7 FTP Seq - 1M**

3	x1	4	3	x6	2	4	3	3	3
x9	x11	x9	x9	x9	6	7	8	8	x9
12	x13	12	12	11	11	x13	x13	12	11
17	14	16	16	x19	15	16	x20	18	15
25	20	20	20	18	21	19	20	20	20
23	25	22	22	24	25	25	26	25	29
27	27	26	29	x34	32	30	26	27	28
33	36	34	36	33	35	35	32	31	31
42	38	39	36	38	36	34	34	36	39
39	40	43	43	43	40	42	43	42	41

**HTTP - Sequential HTTP Testing - with ~ 395K data**

**Config 1 HTTP Seq**

1	0.5	0.5	2	1	1	1	0.5
12	x13	12	7	x15	x4	12	9
17	27	25	x45	x29	x12	28	14
29	x40	38	x57	38	x12	38	14
47	58	52	x62	49	52	x58	x63
x82	77	52	x78	80	30	65	x26
67	96	86	70	x103	x24	74	x26
99	94	108	x110	x115	x28	95	35
102	97	x109	99	101	x37	92	x44
x151	125	142	x145	134	36	118	x44
120	149	140	147	x163	x41	161	x53

**Config 2 HTTP Seq**

1	1	1	0.5	0.5	1	2
6	15	12	11	16	20	11
41	27	35	28	29	18	33
44	53	42	34	31	40	x72
-	74	48	53	66	52	-
91	61	65	77	67	76	x122
x122	75	78	90	95	84	121
147	88	107	96	83	111	x150
141	120	109	108	119	108	x148
155	111	129	130	110	115	x158
167	128	150	131	145	181	x191

**Config 3 HTTP Seq**

x12	2	1	2	1	2	1	2	1	x3
61	63	65	64	x66	63	65	64	66	x81
x329	305	305	x313	301	305	305	313	301	300
x579	517	511	507	520	517	511	507	520	x573
	659	679			555	671	684	672	
x867	840	838	x840	830	842	838	840	830	x937
x1030	1018	1008	1012	1019	1018	1008	1012	1019	x1159
1202	x1203	x1210	1196	1197	1203	1210	1204	1197	x1320
1373	x1389	1303	1325	x1383	1389	1303	1385	1383	x1540
x2040	1554	1562	x1564	1560	1554	1558	1564	1560	x2059
1617	x1727	x1803	1723	1661	1727	1803	1723	1721	x2565

**Config 4 HTTP Seq**

2		1	1	1	x5		x4
x75	67	68	64	65		64	64
x351	316	320	310	317	x339		305
528	479	488	525	x529		528	497
652			x709	705		694	
883	819	818	879	877		750	838
1055			x1115	1049		1049	895
1252			x1320	1220		1218	1189
1399			x1525	1401	x1409		1360
1515			x1707	1580	x1622		1519
1683	1645	1652	x1909	1761		1859	1698

**Config 5 HTTP Seq**

2	5	1	4	4	2	-	2	X 7
62	63	65	66	64	63	-	64	120
302	306	306	300	306	299	-	313	399
499	509	516	514	518	516	-	597	585
660	667	671	667	669	662			
835	836	835	838	841	833	x836	X1698	1010
1012	1010	1015	999	1009	1010	x1028	959	X1217
1201	1205	1200	1205	1208	1205	x1200	X1508	1423
1372	1378	1372	1381	1389	1404	x1383	1344	X1619
1524	1571	1559	1559	1557	x1656	1565	1532	X1849
1692	1724	1711	1727	1797	1830	x1831	1720	X 2275

**Config 6 HTTP Seq**

1		2	1
64		63	63
308		298	300
489 x396		496	495
761			
838		836	824



1014		1009	990
1196		1188	1173
1384		1343	1358
1522		1578	1508
1686		1675	1662

**Config 7 HTTP Seq**

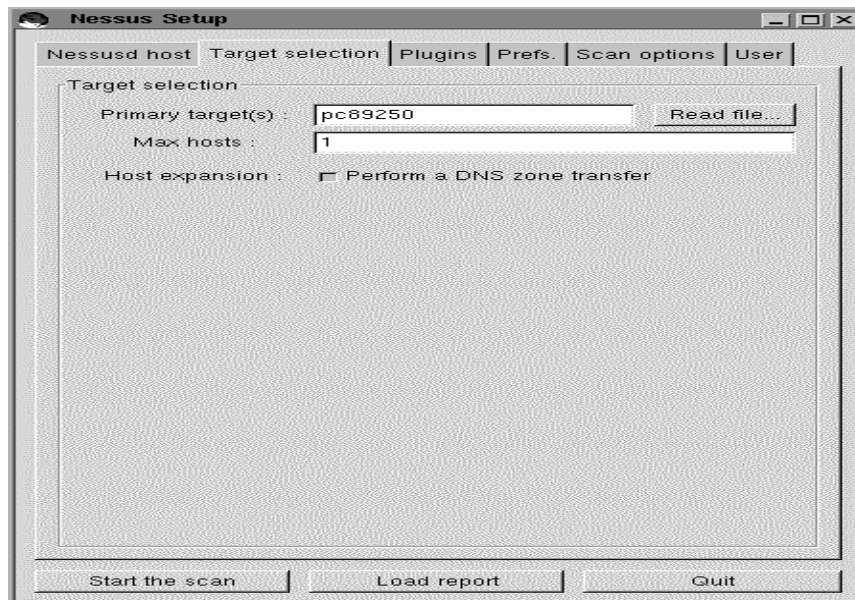
2	5	1	3
63	64	63	63
310	310	293	305
496	512	500	498
	666		
843	844	835	835
1011	1015	1007	1011
1169	1204	1251	1169
1410	1388	1339	1350
1598	1563	1429	1515
1825	1735	1703	1686

It is supposed that all the analysis of the testing could be obtained again by using all the above raw data.

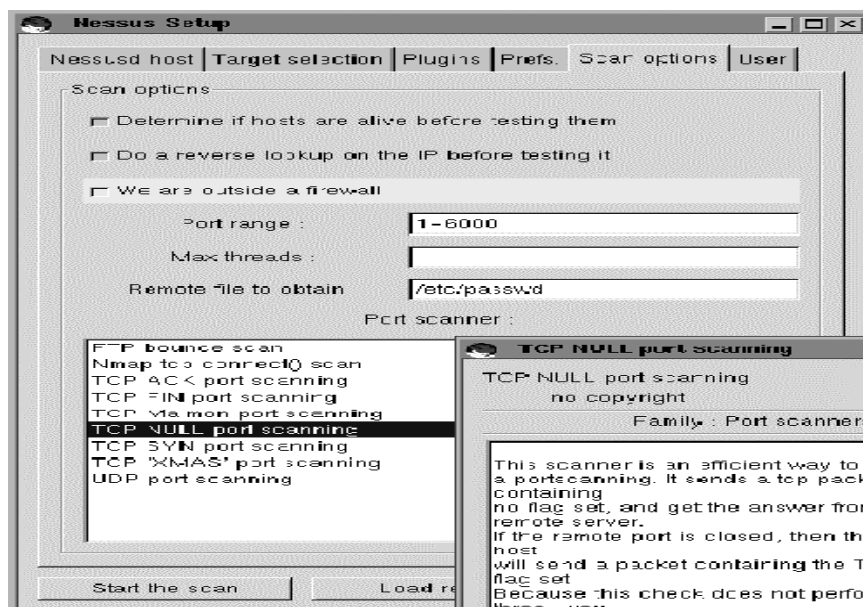
## Appendix D

### Nessus

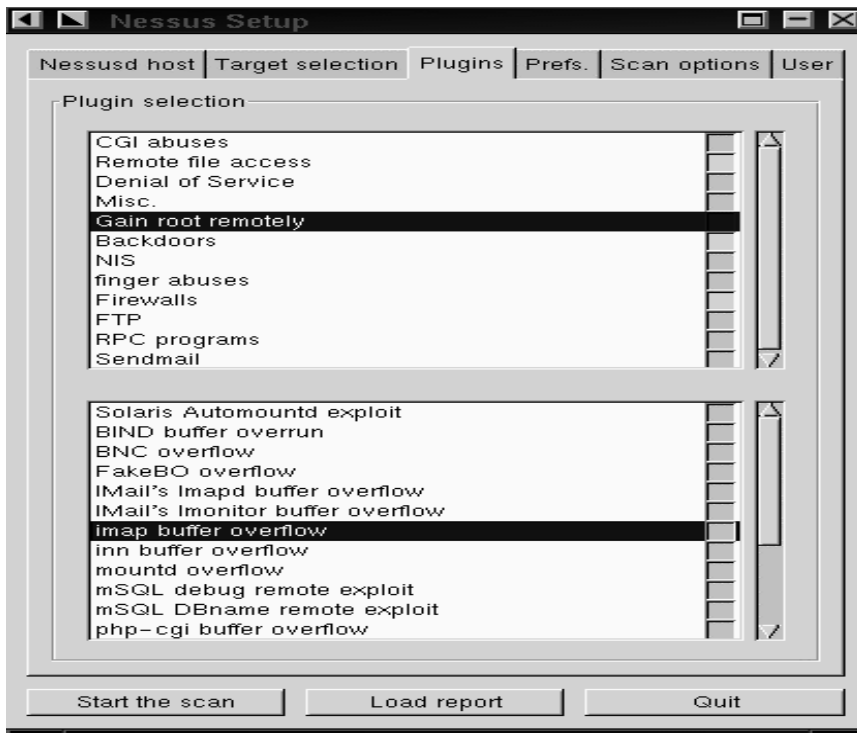
1. Nessus Setup screen to specify the target system to attack or scan



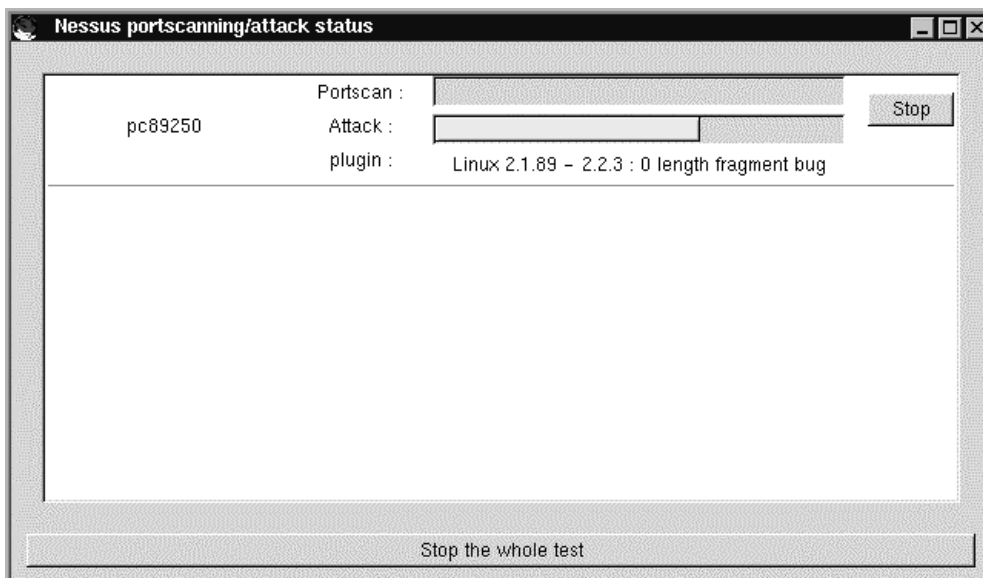
2. The screen to select type of port scanning on the target system



3. Many attacks / plugins could be selected in the following nessus windows. In this project, all the plugins./attacks would be chosen and used in attacking the firewall.



3. Nessus - attack or port scanning process window



## Appendix E

A report produced by running SAINT on the firewall.

The screenshot shows the SAINT web interface. On the left is a navigation menu with links: SAINT Home, Data Management, Target selection, Data Analysis, Configuration Mgt, Documentation, and Troubleshooting. The main content area is titled "Results - pc89250.cse.cuhk.edu.hk" and contains the following sections:

- General host information :**
  - Host type: unknown type
  - NB Name:
  - Subnet 137.189.89
  - 4 Trusted host(s)
  - Scanning level: all out
  - Last scan: Mon Jul 19 18:35:09 1999
- Network Services:**
  - SMTP server
- Vulnerable Services :**
  - Sendmail gives out information using EXPN
  - Sendmail gives out information using VRFY
- Actions :**
  - Scan this host

At the bottom of the main content area, there are links: "Back to the SAINT start page" and "Back to SAINT Reporting and Analysis". The browser's status bar at the bottom shows "nxterm", "gv: Results - pc8925...", and "xv 3.10a(PNG) <un".

The result produced with BSB-monitor is showed below:

The screenshot shows the BSB-Monitor web interface in a Netscape browser window. The title bar reads "Netscape: BSB-Software - Monitor". The address bar shows "file:/data/bsb-monitor/htdocs/network-cfg7.html". The main content area is titled "Network Monitor" and displays a "Network status overview as of 99-7-19 21:4:31".

Server	Description	Services
<b>Routers</b>		
cs7200-1.cse.cuhk.edu.hk	Main Router (Leased Line)	ICMP-Ping Telnet Connection (goto)
137.189.89.250	PC89250 - connected to router	ICMP-Ping Telnet Connection (goto)
<b>Servers</b>		
137.189.89.250	Firewall Server PC89250	ICMP-Ping (goto) File Transfer Protocol (goto) Network File System Postgres RDBM WebMin Administrator Sendmail SMTP Internet Relay Chat
<b>Workstations</b>		
home.cs.cuhk.hk	Workstation of home (Linux)	ICMP-Ping
web.cs.cuhk.hk	Workstation of www (Linux)	ICMP-Ping
mobile1.cs.cuhk.hk	Workstation of mobile1 (Unix)	ICMP-Ping
mobile2.cs.cuhk.hk	Workstation of mobile2 (Windows 98)	ICMP-Ping

Legend:

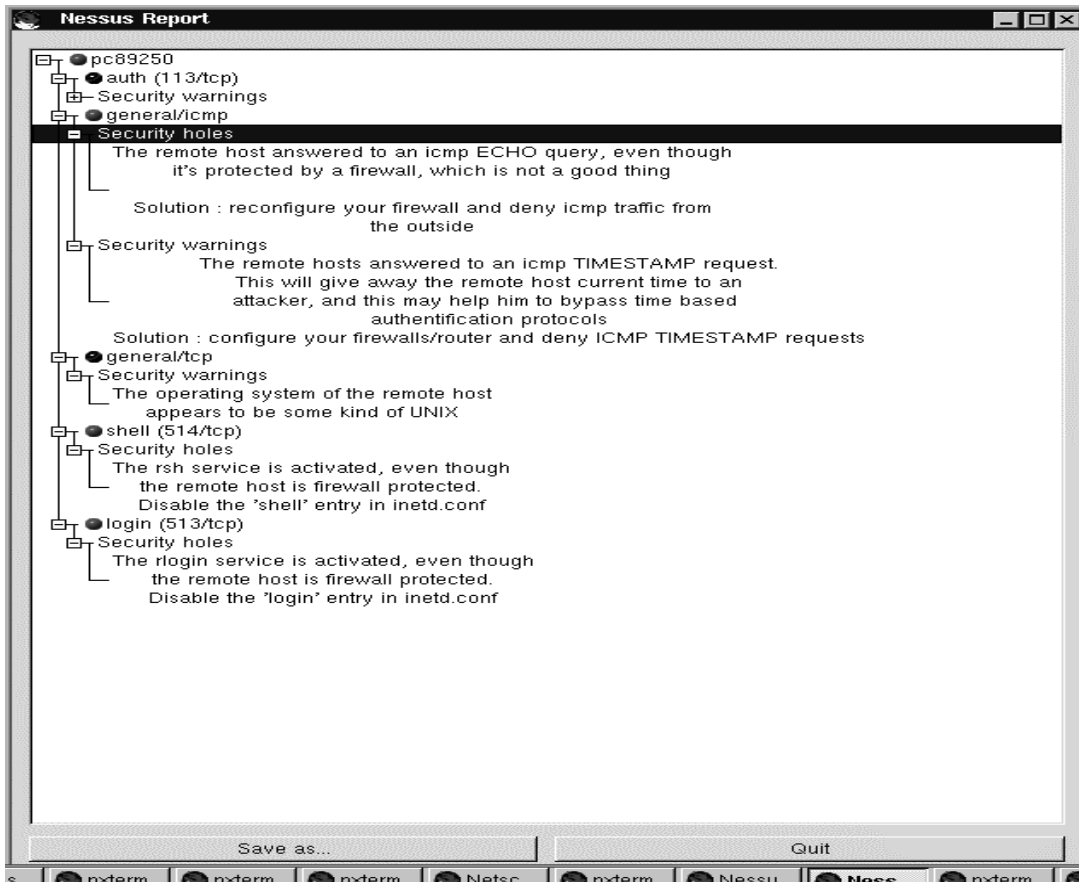
- Service is up
- Critical service is up
- Service is down
- Critical service is down

BSB - Monitor 1.2, © 1998 Dazko Krzicic, BSB - Software

## Appendix F

Some of the reports produced during the testing of security and the screen snap shots are captured and presented as follows.

For security level 2:



### For security level 3

Report generated by COPS

ATTENTION:

Security Report for Thu Jul 1 17:02:00 CST 1999  
from host pc89250.cs.cuhk.hk, COPS v. Version 1.04+

```
**** root.chk ****
**** dev.chk ****
**** is_able.chk ****
Warning! /etc/security is _World_ readable!
**** rc.chk ****
**** cron.chk ****
**** group.chk ****
**** home.chk ****
```

```
**** passwd.chk ****
**** user.chk ****
**** misc.chk ****
**** ftp.chk ****
**** pass.chk ****
**** kuang ****
**** crc.chk ****
**** bug.chk ****
```

ATTENTION:

Security Report for Sat Jul 10 14:38:44 CST 1999  
from host pc89250.cs.cuhk.hk, COPS v. Version 1.04+

Warning! /etc/security is \_World\_ readable!

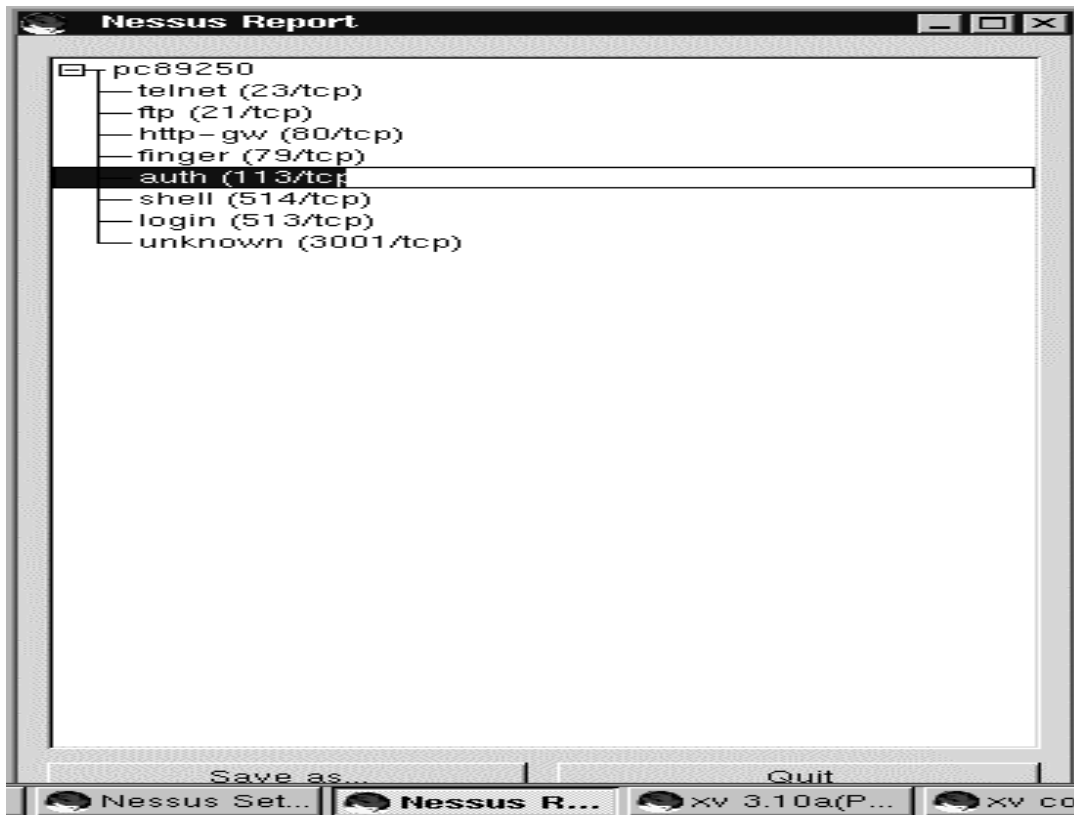
ATTENTION:

CRC Security Report for Sat Jul 10 14:38:26 CST 1999  
from host pc89250.cs.cuhk.hk

```
replaced -rwxr-xr-x root root Jul 2 02:22:24 1999 /usr/bin/captainfo
replaced -rwxr-xr-x root root Jul 2 02:22:24 1999 /usr/bin/clear
replaced -rwxr-xr-x root root Jul 2 02:22:24 1999 /usr/bin/infocmp
replaced -rwxr-xr-x root root Jul 2 02:22:24 1999 /usr/bin/infotocap
replaced -rwxr-xr-x root root Jul 2 02:22:25 1999 /usr/bin/reset
replaced -rwxr-xr-x root root Jul 2 02:22:24 1999 /usr/bin/tic
replaced -rwxr-xr-x root root Jul 2 02:22:24 1999 /usr/bin/toe
replaced -rwxr-xr-x root root Jul 2 02:22:25 1999 /usr/bin/tput
replaced -rwxr-xr-x root root Jul 2 02:22:25 1999 /usr/bin/tset
added -rw-r--r-- root root Jul 2 02:22:22 1999 /usr/lib/libcurses.so
added -rw-r--r-- root root Jul 2 02:22:27 1999 /usr/lib/libform.a
added -rw-r--r-- root root Jul 2 02:22:27 1999 /usr/lib/libform.so
replaced -rw-r--r-- root root Jul 2 02:22:27 1999 /usr/lib/libform.so.4
permisss -rw-r--r-- root root Jul 2 02:22:27 1999 /usr/lib/libform.so.4
replaced -rw-r--r-- root root Jul 2 02:22:27 1999 /usr/lib/libform.so.4.2
permisss -rw-r--r-- root root Jul 2 02:22:27 1999 /usr/lib/libform.so.4.2
added -rw-r--r-- root root Jul 2 02:22:27 1999 /usr/lib/libform_g.a
added -rw-r--r-- root root Jul 2 02:22:26 1999 /usr/lib/libmenu.a
added -rw-r--r-- root root Jul 2 02:22:26 1999 /usr/lib/libmenu.so
replaced -rw-r--r-- root root Jul 2 02:22:26 1999 /usr/lib/libmenu.so.4
permisss -rw-r--r-- root root Jul 2 02:22:26 1999 /usr/lib/libmenu.so.4
replaced -rw-r--r-- root root Jul 2 02:22:26 1999 /usr/lib/libmenu.so.4.2
permisss -rw-r--r-- root root Jul 2 02:22:26 1999 /usr/lib/libmenu.so.4.2
added -rw-r--r-- root root Jul 2 02:22:26 1999 /usr/lib/libmenu_g.a
added -rw-r--r-- root root Jul 2 02:22:23 1999 /usr/lib/libncurses.a
added -rw-r--r-- root root Jul 2 02:22:22 1999 /usr/lib/libncurses.so
replaced -rw-r--r-- root root Jul 2 02:22:22 1999 /usr/lib/libncurses.so.4
permisss -rw-r--r-- root root Jul 2 02:22:22 1999 /usr/lib/libncurses.so.4
replaced -rw-r--r-- root root Jul 2 02:22:22 1999 /usr/lib/libncurses.so.4.2
permisss -rw-r--r-- root root Jul 2 02:22:22 1999 /usr/lib/libncurses.so.4.2
added -rw-r--r-- root root Jul 2 02:22:24 1999 /usr/lib/libncurses_g.a
added -rw-r--r-- root root Jul 2 02:22:25 1999 /usr/lib/libpanel.a
added -rw-r--r-- root root Jul 2 02:22:25 1999 /usr/lib/libpanel.so
replaced -rw-r--r-- root root Jul 2 02:22:25 1999 /usr/lib/libpanel.so.4
permisss -rw-r--r-- root root Jul 2 02:22:25 1999 /usr/lib/libpanel.so.4
replaced -rw-r--r-- root root Jul 2 02:22:25 1999 /usr/lib/libpanel.so.4.2
permisss -rw-r--r-- root root Jul 2 02:22:25 1999 /usr/lib/libpanel.so.4.2
added -rw-r--r-- root root Jul 2 02:22:25 1999 /usr/lib/libpanel_g.a
added drwxr-xr-x root root Jul 2 02:22:43 1999 /usr/lib/terminfo
```

Note, the warnings would be avoided by setting the right permissions then.

The nessus report when only port scanning is run on the firewall.



#### For security level 4

##### With SAINT scanning.

No vulnerability found

No service found

Unknown system of pc89250 - its subnet 137.189.89

Internet domain found cse.cuhk.edu.hk (0/5)

2 Subnets :           137.189.89 (0/1)  
                  137.189.91 (0/4)   No vulnerable host contained

4 Trusted Hosts: ( Hosts trusted by pc89250) -

DNS - Domain Name Services

	subnet
garden.cse.cuhk.edu.hk	137.189.91
cuucs18.cse.cuhk.edu.hk	137.189.91
beryl.cse.cuhk.edu.hk	137.189.91
sapphire.cse.cuhk.edu.hk	137.189.91

##### With COPS checking on firewall

ATTENTION:

Security Report for Thu Jul 1 17:02:00 CST 1999

from host pc89250.cs.cuhk.hk, COPS v. Version 1.04+

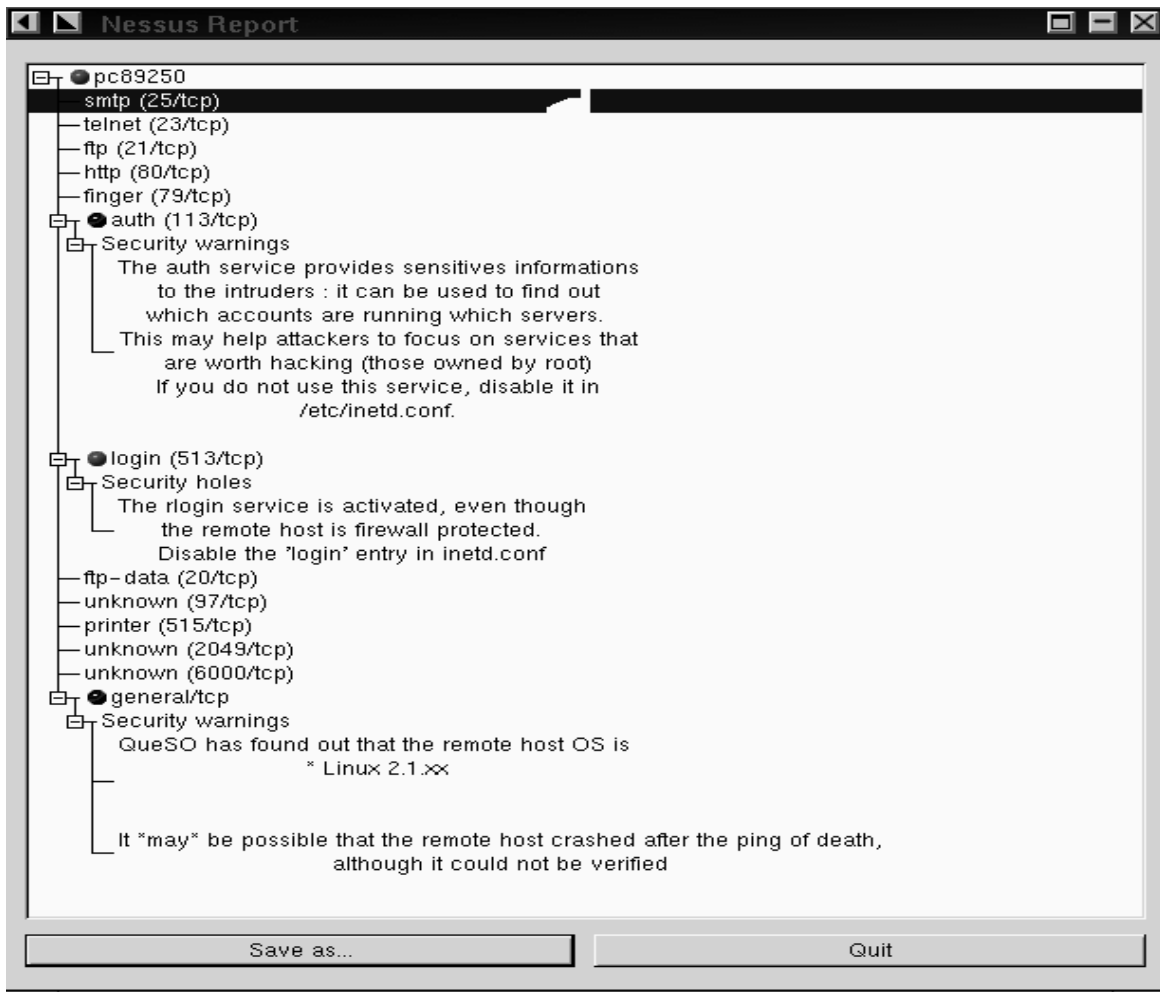
```

**** root.chk ****
**** dev.chk ****
**** is_able.chk ****
Warning! /etc/security is _World_ readable!
**** rc.chk ****
**** cron.chk ****
**** group.chk ****
**** home.chk ****
**** passwd.chk ****
**** user.chk ****
**** misc.chk ****
**** ftp.chk ****
**** pass.chk ****
**** kuang ****
**** crc.chk ****
**** bug.chk ****

```

### For security level 6

This is the report after running the network scanner 'nessus' on the firewall pc89250, under **firewall configuration 6**. The “AUTH “ & “RLOGIN” warned below cannot be eliminated as it is for the flexibility of the LAN. users. Also it is impossible to ping the firewall form outside, the last warning can be ignored.





With BSB monitor, the result is :

**Network Monitor**

Network status overview as of 99-7-19 21:4:31

Server	Description	Services
<b>Routers</b>		
cs7200-1.cse.cuhk.edu.hk	Main Router (Leased Line)	ICMP-Ping Telnet Connection (goto)
137.189.89.250	PC89250 - connected to router	ICMP-Ping Telnet Connection (goto)
<b>Servers</b>		
137.189.89.250	Firewall Server PC89250	ICMP-Ping Apache (goto) File Transfer Protocol (goto) Network File System Postgres RDBM WebMin Administrator Sendmail SMTP Internet Relay Chat
<b>Workstations</b>		
home.cs.cuhk.hk	Workstation of home (Linux)	ICMP-Ping
web.cs.cuhk.hk	Workstation of www (Linux)	ICMP-Ping
mobile1.cs.cuhk.hk	Workstation of mobile1 (Unix)	ICMP-Ping
mobile2.cs.cuhk.hk	Workstation of mobile2 (Windows 98)	ICMP-Ping

**Legend**

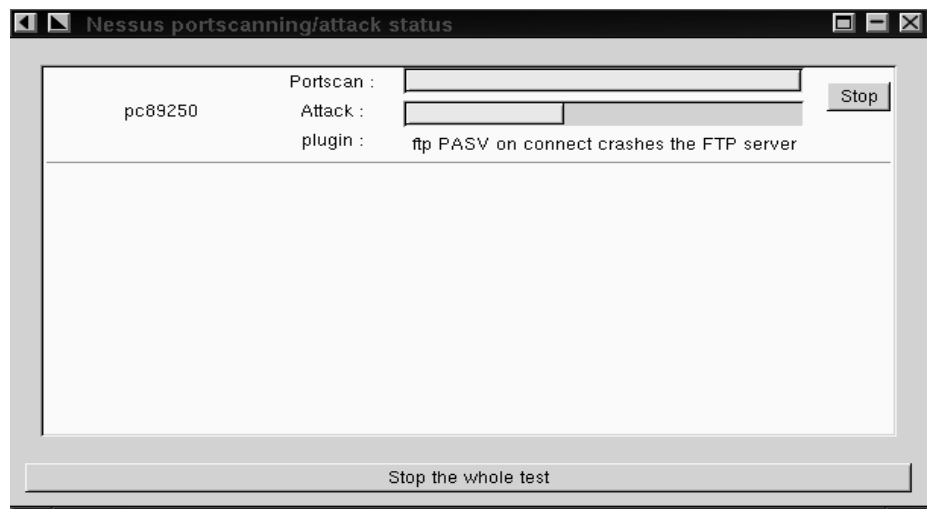
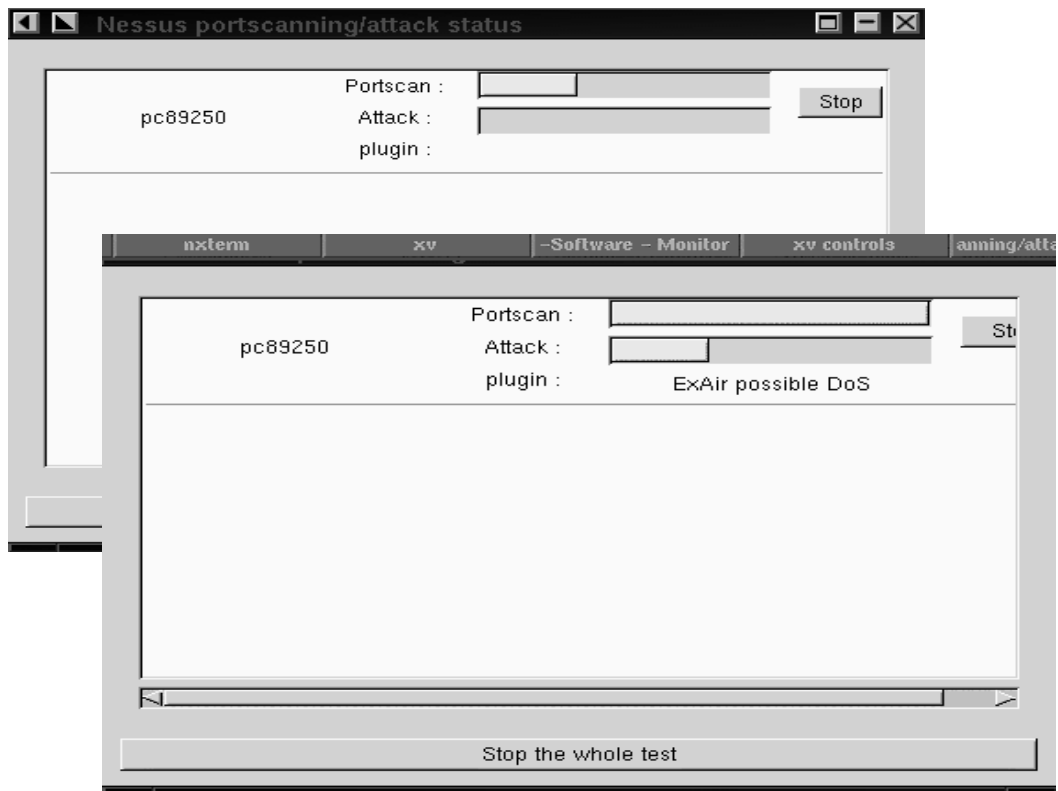
- Service is up
- Critical service is up
- Service is down
- Critical service is down

BSB-Monitor 1.2, © 1998 Darko Krizic, BSB-Software

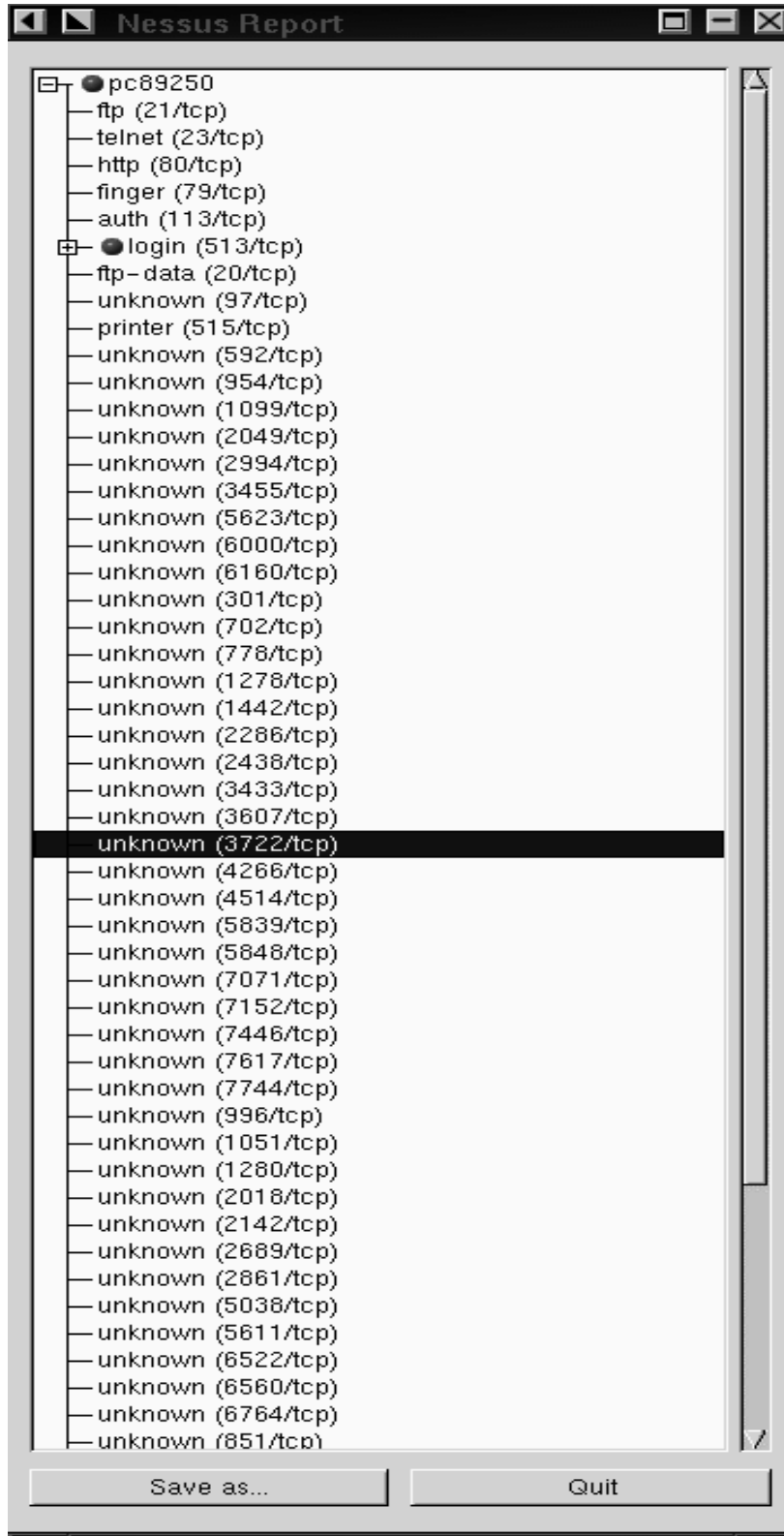
### For security level 7

After the firewall configuration 7 was set up, bsb-monitor is used to check to see the status of the firewall. As expected, we cannot 'ping' the firewall from outside as ICMP packet was blocked. Also SMTP setting was reconfigured so the SMTP sendmail service is not known from outside. Furthermore, all the hosts beside the firewall did not exist from the eyes of outsider.

The network scanner 'Nessus' is in process of attacks on the firewall host pc89250.



The result from the attacks and port scanning of 'Nessus ' was



With BSB monitor result:

**Network Monitor**

Network status overview as of 99-7-11 17:51:26

Server	Description	Services
<b>Routers</b>		
cs7200-1.cse.cuhk.edu.hk	Main Router (Leased Line)	ICMP-Ping
137.189.89.250	Router behind Leased Line	ICMP-Ping Telnet Connection (goto)
<b>Servers</b>		
pc89250	Master Server	ICMP-Ping Apache (goto) Apache SSL (goto) File Transfer Protocol (goto) Network File System Postgres RDBM Sendmail SMTP Post Office Protocol 3 Internet Mail Access Protocol Internet Relay Chat
<b>Workstations</b>		
home	Workstation of home (Linux)	ICMP-Ping
web	Workstation of www (Linux)	ICMP-Ping
mobile1	Workstation of mobile1 (Unix)	ICMP-Ping
mobile2	Workstation of mobile2 (Windows 98)	ICMP-Ping

**Legend**

- Service is up
- Critical service is up
- Service is down
- Critical service is down

BSB-Monitor 1.2, © 1998 Dazko Krizic, BSB-Software

## Appendix G

Here below showed the steps of reconfiguring the vulnerability about SMTP.

Account information probing by malicious users using sendmail and SMTP protocol  
=====

```
solar15.cs.cuhk.hk:/uac/ptmsc/kylau> telnet pc89250 25
Trying 137.189.89.250...
Connected to pc89250.cs.cuhk.hk.
Escape character is '^]'.
220 pc89250.cs.cuhk.hk ESMTP Sendmail 8.9.3/8.9.3; Mon, 19 Jul 1999 15:25:01
+0800
```

```
EXPN root
250 System Administrator <root@pc89250.cs.cuhk.hk>
EXPN guest
550 guest... User unknown
EXPN lpr
550 lpr... User unknown
EXPN ftp
250 FTP User <ftp@pc89250.cs.cuhk.hk>
EXPN mail
250 mail <mail@pc89250.cs.cuhk.hk>
EXPN kylau
550 kylau... User unknown
EXPN www
550 www... User unknown
QUIT
221 pc89250.cs.cuhk.hk closing connection
Connection closed by foreign host.
solar15.cs.cuhk.hk:/uac/ptmsc/kylau>
```

After the /etc/sendmail.cf is modified and the above attack is simulated again, the 'EXPN' operation is disallowed and the attack is avoided in the way as :

```
solar15.cs.cuhk.hk:/uac/ptmsc/kylau> !1
telnet pc89250 25
Trying 137.189.89.250...
Connected to pc89250.cs.cuhk.hk.
Escape character is '^]'.
220 pc89250.cs.cuhk.hk ESMTP Sendmail 8.9.3/8.9.3; Mon, 19 Jul 1999 15:33:33 +08
00
EXPN root
502 Sorry, we do not allow this operation
EXPN lpr
502 Sorry, we do not allow this operation
EXPN mail
502 Sorry, we do not allow this operation
QUIT
221 pc89250.cs.cuhk.hk closing connection
Connection closed by foreign host.
solar15.cs.cuhk.hk:/uac/ptmsc/kylau>
```

In fact the attack can also be traced and discovered in the /etc/log/maillog as :

```
^^^^^^^^^^^^^^^^^^
. [137.189.88.51], stat=Deferred: Connection refused by solar1.cs.cuhk.hk.
```

```
Jul 19 14:39:44 pc89250 sendmail[24195]: NOQUEUE: pc89136.cse.cuhk.edu.hk
[137.189.89.136]: EXPN root
Jul 19 14:39:55 pc89250 sendmail[24197]: NOQUEUE: pc89136.cse.cuhk.edu.hk
[137.189.89.136]: EXPN gust
Jul 19 14:40:05 pc89250 sendmail[24202]: NOQUEUE: pc89136.cse.cuhk.edu.hk
[137.189.89.136]: EXPN guest
Jul 19 14:40:12 pc89250 sendmail[24203]: NOQUEUE: pc89136.cse.cuhk.edu.hk
[137.189.89.136]: EXPN lpr
Jul 19 14:40:20 pc89250 sendmail[24204]: NOQUEUE: pc89136.cse.cuhk.edu.hk
[137.189.89.136]: EXPN kylau
Jul 19 14:40:30 pc89250 sendmail[24194]: NOQUEUE: pc89136.cse.cuhk.edu.hk
[137.189.89.136]: EXPN attack?
Jul 19 14:40:31 pc89250 sendmail[24205]: NOQUEUE: pc89136.cse.cuhk.edu.hk
[137.189.89.136]: EXPN ftp
Jul 19 14:48:59 pc89250 sendmail[24233]: VAA06013: to=kylau@solar1, ctladdr=root
(0/0), delay=1+17:17:31, xdelay=00:00:00, mailer=esmtpl, relay=solar1.cs.cuhk.hk.
[137.189.88.51], stat=Deferred: Connection refused by solar1.cs.cuhk.hk.
Jul 19 15:25:42 pc89250 sendmail[24278]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN root
Jul 19 15:25:52 pc89250 sendmail[24279]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN guest
Jul 19 15:26:09 pc89250 sendmail[24280]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN lpr
Jul 19 15:26:15 pc89250 sendmail[24281]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN ftp
Jul 19 15:27:03 pc89250 sendmail[24282]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN mail
Jul 19 15:27:03 pc89250 sendmail[24282]: NOQUEUE: forward
/var/spool/mail/.forward.pc89250: Group writable directory
Jul 19 15:27:03 pc89250 sendmail[24282]: NOQUEUE: forward
/var/spool/mail/.forward: Group writable directory
Jul 19 15:27:30 pc89250 sendmail[24277]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN attack?
Jul 19 15:27:31 pc89250 sendmail[24283]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN kylau
Jul 19 15:27:38 pc89250 sendmail[24284]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN www
Jul 19 15:33:06 pc89250 sendmail[24296]: alias database /etc/aliases rebuilt by
root
Jul 19 15:33:06 pc89250 sendmail[24296]: /etc/aliases: 14 aliases, longest 10 by
tes, 152 bytes total
Jul 19 15:33:07 pc89250 sendmail[24311]: starting daemon (8.9.3):
SMTP+queueing@01:00:00
Jul 19 15:33:07 pc89250 sendmail[24314]: VAA06013: to=kylau@solar1, ctladdr=root
(0/0), delay=1+18:01:39, xdelay=00:00:00, mailer=esmtpl, relay=solar1.cs.cuhk.hk.
[137.189.88.51], stat=Deferred: Connection refused by solar1.cs.cuhk.hk.

Jul 19 15:33:42 pc89250 sendmail[24317]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN root [rejected]
Jul 19 15:33:47 pc89250 sendmail[24317]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN lpr [rejected]
Jul 19 15:33:59 pc89250 sendmail[24317]: NOQUEUE: solar15.cse.cuhk.edu.hk
[137.189.88.65]: EXPN mail [rejected]
```

## APPENDIX H - A comparison between proxy gateway and packet filter

A comparison between proxy gateway and packet filter [33]

	Proxy Gateway	Packet Filter	Details
<b>TCP traffic examination</b>	Applies rules to TCP session, monitors data flow to determine if each command within the session is allowed or not.	Applies rules to each packet, based on source and destination address and port	Proxy gateways are more thorough and more efficient for TCP traffic
<b>UDP traffic examination</b>	Generic proxies allow UDP traffic to be controlled between fixed ports. Cannot handle varying port addresses (RPC-based traffic, like NFS)	Maintains state for UDP communications, keeping a channel open between sender and receiver, handling even RPC traffic	The UDP protocol is even less secure than TCP, so no firewall provides thorough UDP security
<b>Flexibility</b>	A strength and weakness is the lack of flexibility. Proxy exists for a protocol, or us the generic proxy for other protocols, or protocol can't pass	Very flexible. Unfortunately can leave room for mis-configuration	A proxy gateway is best if it supports all the protocols you are passing
<b>Ease of configuration</b>	Fewer choices, so generally easier to configure	Many choices	A more expert hand is needed to guide packet filter configuration. For instance, both types can do address hiding, but proxy gateways do it by default and packet filters must be configured to hide addresses
<b>Ease of management</b>	Again, simpler means easier	Keeping the protocols passed and rules Implemented to a minimum can simplify management	Both require skill, knowledge, and specific firewall training to be securely managed
<i>Miscellaneous</i>	Address hiding for free, good logging and alerting potential	Address hiding via configuration options, reasonable logging potential and good alerting potential	

- The end of the report -