Qian Zhou

Comparing Dedicated and Integrated Firewall Performance

Bachelor's Thesis
Information Technology

May 2013

MIKKELIN AMMATTIKORKEAKOULU
Mikkeli University of Applied Sciences

DESCRIPTION

MIKKELIN AMMATTIKORKEAKOULU
Mikkeli University of Applied Sciences

Date of the bachelor's thesis

May 30th, 2013

Author(s)

Qian Zhou

Degree programme and option

Information Technology

Name of the bachelor's thesis

Comparing Dedicated and Integrated Firewall Performance

Abstract

Because of the popularity and benefits of using Internet, corresponding Internet and network security threats also increase year by year. Firewalls are used to be the main protector for an organization. Initially, we have the packet filtering firewall to control traffic and provide securities. Nowadays, we have many kinds of firewall solutions and they can provide different security levels. How to choose a suitable firewall solution for a company according to their needs is worth to search.

This thesis will cover topics about secure mechanisms in the firewall, different types of firewall technology and commonly used firewall architectures nowadays. Based on the theory part, I will implement different firewall solutions and measure their performance. Cisco router with CBAC and SmoothWall placed inside a router will be mainly compared and analysed. Then I will give my suggestions for a company whether choose a dedicated firewall or an integrated firewall.

Subject headings, (keywords)

Pages	**Language**	**URN**
48	English	

Remarks, notes on appendices

Tutor

Matti Koivisto

Employer of the bachelor's thesis

Mikkeli University of Applied Sciences

CONTENTS

1. INTRODUCTION

There is no doubt that nowadays data communication through networks has become an indispensable infrastructure resource for our daily life and we get many benefits from it. However, increasing number of adherent network threats for business, governments, educational institutions and other organizations imposes unexpected heavy risks to us.

According to KSN data, Kaspersky Lab detected 1,347,231,728 threats in Q3 2012 and among which the total number of URLs serving malicious code was 91.9 million [1]. Those threats cause uncountable losses for not only the organizations but also individuals. In order to protect our information and assets, firewalls are used to protect against network-based attacks. A firewall is a device or software program used to control whether an incoming traffic or outgoing traffic could be passed or not according to the predefined security criteria [30]. In this way, it can provide security for the internal network by leaving potential threats outside the firewall and still let authorized users access the private network resources. There are various types of firewalls such as a packet filtering firewall, a stateful inspection firewall and an application layer filtering firewall. All of them have been published and built by various firewall vendors.

Firewalls can be classified different ways. Classification can be based on cost such as commercial and freeware, or be named according to implementation methods like hardware-based firewall and software-based firewall. We can also have dedicated hardware firewall and integrated ones.

In my thesis, I will analyze the performance of different firewall implementations and my practical study is comparing dedicated and integrated firewalls. In my final thesis, the dedicated firewall means it concentates on firewall functions no matter they are software based or hardware based firewall. Examples can be SmoothWall and Cisco ASA. While intergrated firewall here means a device has its own main responsiblity but it can be interated with some firewall features. For instance, the main job of a Cisco Router is routing but it can be a integrated firewall via inserting some firewall features. The theoretic aim of the thesis is to understand the firewall security mechanism, firewall history, advantages and limitations in different types of firewall technologies, and major firewall archititetures. The more practical purpose is to respectively implement firewall solution with Cisco IOS firewall (CBAC-Router) and SmoothWall inside a Cisco router, then compare their related performances.

The structure of the thesis is organized as follows. Chapter 2 gives an introduction of firewall security mechanism. Chapter 3 will cover the major types of firewall solution and comparisons between them. Chapter 4 will discuss common used firewall architectures. In Chapter 5, firewall implementation with CBAC-router and SmoothWall will be compared and analyzed. Network without firewall solution will be also included as reference in the lab work. Finally there will be a summary of the whole thesis.

2. FIREWALL SECURITY MECHANISM

In this chapter, I will firstly explain firewalls' important roles in Internet security. Then I will simply summarize some basic and commonly used firewall security components such as access control list, network address translation, and firewall authentication.

2.1 Role of Firewalls in Internet security

We live in an Internet-dependent era and the trend of Internet expansion continues to grow due to lager number of businesses using Internet as their sales and information channel. On the negative side, computer and network security issues are also growing. From Figure 1, we can obviously notice that with the tremendous growth trend, 83 million pieces of malware has already been discovered in March of 2012. And the continuingly worse situation is that new malware still occurs and topped at 7 million in the first quarter 2012. We do not know when the total number will top the 100 million but certainly that it will not be long.

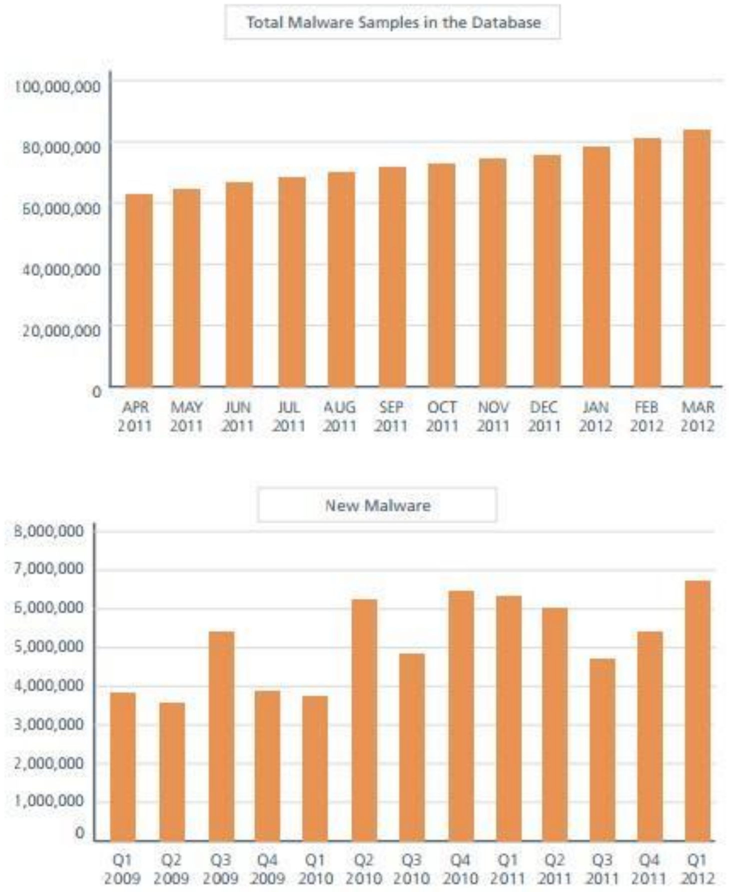


Figure 1 McAfee Threats Report: First Quarter 2012 [2]

Computer networks are vulnerable to many threats such as denial of service attacks, password guessing, protocol-based attacks, social engineering, war dialing and so on [3].

Below are detailed explanations of the threats mentioned above [3]:

- Denial of service attack: The attacker sends many requests to overwhelm a computer or network and in this way the valid users can not normally access or use the resources from the computer or the network.
- Password guessing: Attackers use random combination of the characters and repeatedly tries the password.
- Protocol-based attacks are method which takes advantage of protocols' weakness. For example, DNS attack can be implemented by the attacker who makes the attaker's device pretend to be the DNS server. Users who sends requests to the fake DNS server can receive data with malicious content.[29]
- Social engineering: It means the attacker tricks users to reveal confidential information without letting users noticing being cheated. And then the attacker can make use of the

collected information to access the targeted system. Malicious attacker may make use of social means like Facebook to get needed information.

- War dialing: Malicious people use some software to dial telephone numbers to search computers that provides a potential path towards a corporate network. Today the access is more often gained through unsecured wireless methods. Intentional attackers can bypass the firewall and intrusion detection system to access internal network resources.

In addition, users' lack of awareness in network security can also lead them to unintentionally access dangerous websites or reveal confidential information to potential hackers and then cause losses for them. Except teaching and training their network security knowledge, appropriate access controls on the firewall is needed to avoid attacks launched by professional hackers.

Firewalls have been used for a long time and they can protect against some of the major attacks mentioned above but not all unfortunately. By using firewalls, incoming and outgoing traffic can be restricted. This can help the organizations to make their resources only available to authorized parties and keep information in a private and secure way. If firewalls are used to monitor and limit connections to a network, hackers and unwanted programs have difficulties to invade your network or computer.

In conclusion, even firewall cannot protect against all the threats, but they still play extremely important roles in almost every network.

2.2 Firewall Characteristics

No matter what kind of firewall vendors you choose or what kind of firewall solutions you implement for your network, there are some common rules or characteristics for firewalls mentioned below [4]:

- All traffic from inside to outside or from outside towards inside must pass by the firewall.
- Regardless of the firewall types you use and types of security policies you define on the firewall, only authorized traffic predefined by the local security policy will be permitted to pass. Traffic that is unmatched by the defined rules will be blocked.
- Firewall is immune to penetration; this means that firewall uses a secure and trusted operating system.

2.3 Access Control Lists (ACLs)

Access control list is a list of permit or deny statements defined in order and a statement is combined with information like IP addresses, port numbers and protocols [5]. ACLs can be used by network administrators to deny traffic or permit specific traffic in or out of the networks. As a result, using ACLs on the firewall for basic traffic filtering block unauthorized or potentially malicious packets towards the trusted network. As the devices I will use in the practical part of the thesis are Cisco products, the ACL rules are explained below using Cisco syntax. In this thesis, ACLs are Cisco related and may be a little different from other vendor's ACLs settings.

2.3.1 Characteristics of ALCs

There are some common rules mentioned below for all kinds of Cisco ACLs and I will introduce them in details later [5]:

- One ACL per protocol. This means each ACL is applied to only one protocol rather than two or more protocols.
- One ACL per direction. It means the traffic controlled by an ACL is in one directional either inbound or outbound at a time. It cannot apply an ACL in two directions at the same time.
- One ACL per interface. This stands for each ACL only controls traffic for one interface.
- ACL statements operate in sequential order. This means that the ACL statements are checked from top to bottom. If the incoming packet matches the first statement, then the remaining statements will not be checked. If there is no match, the packet will be examined by the next ACL statement until it matches or be dropped finally. As a result, it is better to put the most important rule at the top and then the second important one.
- Final statement of the ACL is "implicit deny any statement", so an ACL should have at least one permit statement otherwise all traffic will be blocked.

2.3.2 Types of Cisco ACLs

ACLs on Cisco IOS routers can be either named or numbered, standard or extended. Other complex ACLs are based on those ACLs. Named ACLs must be specified as either standard or extended.

- Numbered ACLs

Numbered ACLs are more effectively used to determine ACL type. ACLs with numbers from 1 to 99 and from 1300 to 1999 are Standard IPv4 ACL, while range 100 to 199 and range 2000 to 2699 belong to Extended IPv4 ACL. Below are details about types of ACLs [6] [31]:

- Standard ACLs

Standard ACLs allow a router or a firewall to permit or deny traffic only based on source IP addresses information. All other traffic which does not match the rule will be blocked as the implied "deny any" at the end of the ACL. This type of ACL is easy and fast to implement on devices to provide simple access control. However, rules are too simple and the checked information is in small limited range. As a result, traffic with malicious content but matching this rough rule can still pass the firewall.

- Extended ACLs

Extended ACLs filter IP packets according to many attributes like protocol types, source and destination IP address, source and destination TCP or UDP ports. Compared with Standard ACLs, it is more complex and therefore safer as more detailed information will be examined. Only packets matching all the attributes can be passed or denied.

- TCP Established ACLs

TCP Established ACLs were introduced in 1995. All the traffic from outside are not allowed to enter the inside network. This causes inconvenience for returning traffic that is initiated from the inside network. In addition, as many commonly used applications are based on TCP. A great deal of needed information using TCP is unavailable when returning traffic is blocked by the Extended ACLs. So TCP Established ACLs are introduced to still block all traffic coming from Internet but to allow TCP reply traffic initiated from internal network. TCP Established ACLs use the “established” keyword to force the router to check whether the TCP ACK or RST control flag is set or not. If the ACK flag is set, the TCP traffic will be allowed in. Otherwise traffic will be denied as the traffic will be deemed to associate with a new connection initiated by the outside network.

TCP Established ACLs are more complex than the two ACLs above but they are more flexible when provide same level of security for the network. However, the “established” parameter permits packets with the appropriate control flag setting on the TCP segment. The router does not keep track of conversations. Therefore, it cannot guarantee that all traffic coming from outside are initiated from the inside. Professional hackers can change the control flags and send malicious packets pretending as the returning traffic to the inside network.

- Reflexive ACLs

A reflexive ACL is a kind of session filter and it was introduced in 1996. Standard and extended ACLs do not keep track of the state of a connection. Reflexive ACL were developed to safely allow returning traffic back into internal network while block traffic originated from outside network according to session. It can be only defined with extended named IP ACLs rather than named ACLs or standard named IP ACLs.

A reflexive ACL is safer compared to TCP Established ACL because more filter criteria must be matched before a packet is permitted. For example, spoofing and some types of denial of service attacks could be prevented by using reflexive ACL. In addition, the session filter is also temporary. This means that after the session is over, the filter will be removed. As a result, hackers have less opportunity to attack the network because of the limited time.

- Dynamic ACLs

Dynamic ACLs are also known as lock-and-key ACLs and are applied for IP traffic only. They are dependent on Telnet or SSH connectivity, authentication (local or remote), and Extended ACLs. Traffic wanting to pass through the router will be blocked first until the senders connect the router by Telnet or SSH as well as being authenticated either locally or remotely. With dynamic ACLs, remote connectivity and resources access in a secure level between two remote networks are available. For example, a remote user A on the other side of the Internet can access to a host in a local network protected by a firewall and vice versa.

One of the main advantages of the dynamic ACLs is that it uses the authentication mechanism to verify the users. It also makes the management process simpler in large internetworks because of local or remote authentication. Dynamic user access by the firewall reduces opportunities of the network hacker and also does not need to compromise other security settings.

- Time-based ACLs

Time-based ACLs were introduced in 1998 and had similar function of Extended ACLs. Time-based ACLs restrict traffic based on time range like number of hours, days, weeks or months and so on. This adds flexibility for the administrator to control the traffic or resource access according to the time. For example, employees can surf the Internet during the lunch time rather than the working time. It also simplifies network administrators' workload on analyzing the logging messages in the peak time if they deny the traffic accordingly.

- IPv6 ACLs

IPv6 ACLs are introduced due to needs for IPv6 environment and increasing number of IPv6 attacks. In this thesis, I will not cover details about IPv6 ACLs as lab work will be done in the IPv4 environment.

2.3.3 Context-based Access Control (CBAC) ACLs

CBAC is a software-based firewall feature that uses application layer protocol session information to filter TCP and UDP packets. Like the reflexive ACLs which inspect traffic for sessions that originates from outside, CBAC also inspects traffic from inside network. By tracking the connection status, a CBAC firewall can guarantee packets are not modified before entering the network.

A CBAC firewall could be classified as both stateful inspection firewall and application firewall as it filters traffic at multiple layers such as network layer (IP addresses and protocols), transport layer (ports, TCP and UDP sessions), session layer (conversation state) and application layer (protocols for specific applications). In the Cisco IOS Firewall solution, four main functions are provided by CBAC as follows [7]:

- Traffic filtering: CBAC can inspect traffic according to sessions and permit specified TCP or UDP returning traffic initiated from the inside network.
- Traffic inspection: CBAC inspects packet sequence numbers in TCP connections to see if they are within expected ranges and drops any suspicious packets. This helps protect against SYN-flood attack that means a network attacker floods a server with a large of connection requests while does not complete the connection.

- Intrusion detection: It can protect against specific SMTP attacks by reviewing syslog message and monitoring specific attack signatures.
- Generation of audits and alerts: Audit trails use syslog to track all network actions and timestamps while real-time alerts send syslog error messages to central console upon suspicious activity.

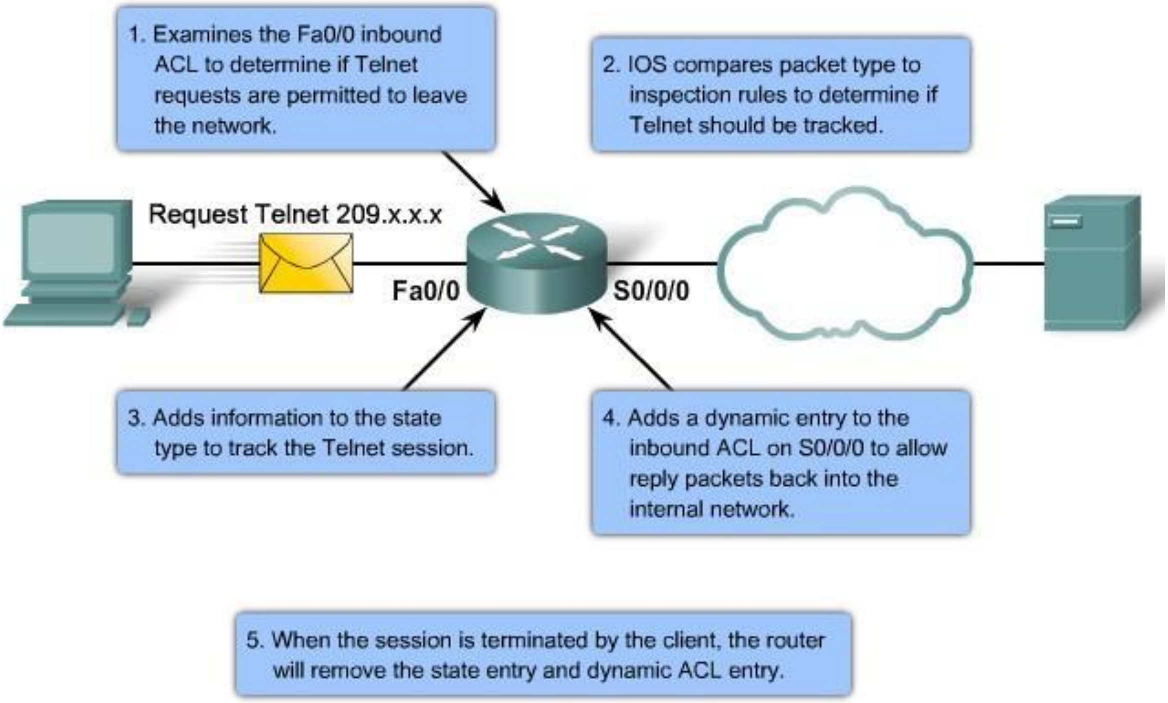

1. Examines the Fa0/0 inbound ACL to determine if Telnet requests are permitted to leave the network.
2. IOS compares packet type to inspection rules to determine if Telnet should be tracked.
Request Telnet 209.x.x.x
Fa0/0
S0/0/0
3. Adds information to the state type to track the Telnet session.
4. Adds a dynamic entry to the inbound ACL on S0/0/0 to allow reply packets back into the internal network.
5. When the session is terminated by the client, the router will remove the state entry and dynamic ACL entry.

Figure 2 CBAC operations [8]

Figure 2 shows how CBAC operates when a user in protected network initiates an outbound connection to external network.

2.4 Network Address Translation

Network Address Translation (NAT) is the process of modifying IP address information in IP packet headers and changing it to another IP address on a routing device [9]. Most of the firewall vendors use NAT in their firewalls not only because of conservation in IP address space but also to provide security. By hiding detailed information such as addresses of the protected network, network will be less likely attacked.

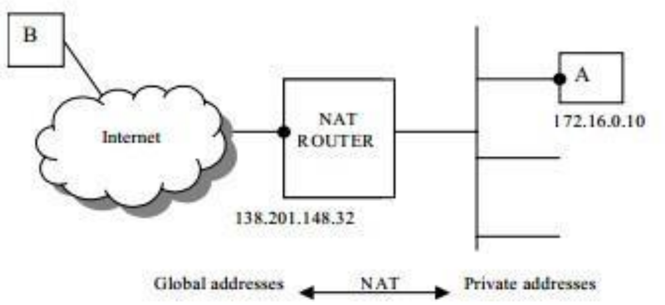


Figure 3 Network Address Translation [10]

Figure 3 simply shows how NAT basically works. When host A with a private address 172.16.0.10 in the internal network wants to connect to host B with a public address assigned by the ISP, a private IP address will be translated to a public IP address. In our case, source IP address 172.16.0.10 should be translated to 138.201.148.32 when packets pass cross the NAT router. Similarly, when host B replies to host A, 138.201.148.32 will be exchanged to 172.16.0.10 by the NAT router [11].

The NAT association can be either static as showed in Figure 3 or dynamic which contains IP addresses and ports number. NAT translates local private addresses into their associated public addresses and vice versa. We can see from the above example that NAT acts as a bridge between your local network and the Internet. It makes all connections seem to be from the NAT address rather than from local addresses of the LAN computer. As a result of this, internal addresses and the network topology are invisible or hidden from outside. A firewall or a router with NAT feature can make the network less susceptible to be directly attacked and therefore reduce risks or threats for the organizations.

However, there are also some disadvantages of the NAT. Due to the complexity of address exchanges, troubleshooting process will be more confusing and performance will be reduced because of the additionally consumed overhead by recalculating checksum. In addition, some protocols like IPSec (IP Security) are designed to detect modifications of the header [35]. As NAT changes the IP address, sometimes it is difficult to distinguish this change from malicious data sending by a hacker. This may cause complexity and other security problems for the Intrusion Detection System (IDS) or Intrusion Prevention System (IPS) when they cannot work well with Network Address Translation (NAT) [12].

2.5 Demilitarized Zone

Demilitarized zone (DMZ) is a physical or logical sub-network that contains and exposes an organization's external-facing services to a larger untrusted network, usually the Internet [13]. DMZ is commonly used with firewall. Basically, DMZ houses servers like an email server, a web server and a Domain Name System (DNS) server and those servers are usually exposed to the public network. DMZ isolates organizations' internal network from the general public network where hackers may exist. Meanwhile, it also enables connections towards the DMZ from Internet and the corporates' internal network. By limiting outsiders' access only to the DMZ rather than the other parts of the corporates' network, DMZ adds an additional network security for the internal network.

There are two types of firewall with DMZ, one is single firewall the other is dual firewall. They will be introduced in the firewall architecture section.

2.6 Virtual Private Network (VPN)

We may think packets inside the firewall network are roughly safe but once they are sent out they should be also safe. However, once the data is out of the firewall and towards the other remote network via the public network like Internet, how could the data still be protected? This issue requires that our firewall is not only controlling access of traffic but also be able to encrypt the outgoing traffic. Similarly, the firewall on the other side of public network should have the ability to decrypt the incoming traffic and filter unneeded traffic between the internal network and public network. The whole process is an example of Virtual Private Network (VPN).VPN is used to provide security for communications between two remote networks across public network like Internet [32]. VPN also involves encrypting and decrypting data, authenticating users and checking integrity.

There are two types of VPN architectures, one is a remote access and the other is a site to site VPN. The remote access VPN also can be thought as a host to a gateway and a VPN gateway usually refers to a firewall or a router. The remote access VPN provides a secure connection between an individual host and a remote network. For example, a salesman who stays at home or travels outside the office can still access the company's intranet resources by using the remote access VPN. It needs the firewall administrator to define rules for the access. Only authorized users can gain access to the company and they are given limited rights by the network administrator to use the company resources. The site to site VPN is also called gateway to gateway. It enables users on different networks to communicate in a secure tunnel while other network users cannot share the resources. For example, employees at different branch offices can access resources between each site without limitation of separate locations. This model also needs authentication either by username and password or digital signatures. [32]

Benefits of VPN are could be providing remote access in a secure way and allowing people to use intranet resources without limitations of the location. It also extends the private network via the virtual tunnels. However, there are also some disadvantages as follows. VPN is some part of the firewall and it usually adds additional workload to the firewall as well as consumes additional resources. Nowadays many firewalls have special hardware to support the VPN and minimize the negative impact of the VPN services.

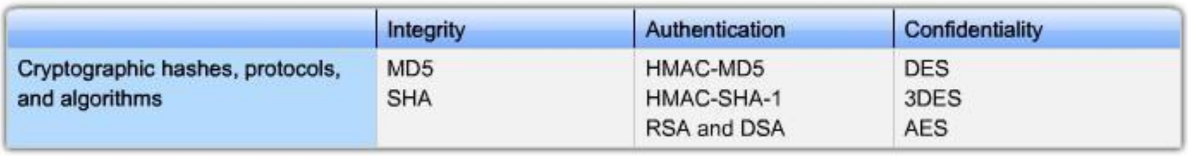
2.7 Firewall and Authentication

We have the sense that authentication depends on verifying users' identity such as username and password. However, as we know simple passwords are not reliable nowadays. Stronger authentication mechanisms are very important on firewalls when protocols such as Telnet, SSH are used. As I mention on the dynamic ACLs, when remote user establishes SSH or Telnet connection to the router, lock-and-key authentication could be used by a VTY line. This password based authentication can use different methods like password-only, a local username database or an AAA (Authentication, Authorization, and Accounting) server with either RADIUS or TACACS+.

When AAA is implemented with Cisco products, either Terminal Access Control Access Control Server Plus (TACACS+) or Remote Authentication Dial-In User Services (RADIUS) can be selected. Both methods can be used by remote users when communicating with AAA servers. Differences are TACAS+ encrypt all information like passwords and usernames while RADIUS only encrypts user passwords. As a result, TACAS+ is more secure and it is supported by most Cisco products but RADIUS is the RFC standard. How to choose appropriate method depends on the organization's needs. If detailed accounting is more important RADIUS maybe selected as it provides extensive accounting. If there are different user groups in the organization and different authorization policies are applied, TACACS+ maybe selected as it separates authentication and authorization processes [17].

2.8 Cryptographic Security Mechanisms

Common authentication methods in modern firewall are one time passwords, hashed passwords and digital signatures with Public Key Infrastructure (PKI). As many security implementations including some firewall implementations are based on cryptographic mechanism, I will explain this topic a little bit. Cryptography consists of authentication, integrity and confidentiality and protects data against exposed to untrusted parties.



	Integrity	Authentication	Confidentiality
Cryptographic hashes, protocols, and algorithms	MD5 SHA	HMAC-MD5 HMAC-SHA-1 RSA and DSA	DES 3DES AES

Figure 6 CIA [18]

From Figure 6 we can see there are various protocols and algorithms for implementing confidentiality, integrity and authentication. HMAC, RSA or DSA can be used in the authentication while integrity is guaranteed by using MD5 or SHA.

Encrypting communication traffic is one example to secure the data confidentiality in firewalls. There are two types of encryption algorithms, one is symmetric and the other is asymmetric. Symmetric algorithms like DES, 3DES and AES assume both communicating parties know the pre-shared key. While asymmetric algorithms like RSA and Public Key Infrastructure (PKI) establish a secure method without previously knowing the secret key. Choosing a suitable algorithm depends on the specific security needs. For example, MD5 is less safe but faster than SHA. In addition, encapsulating security payload (ESP) which offers confidentiality, integrity and authentication is widely used. Those security mechanisms used in the firewall can protect communications traffic to or from firewalls against modification, insertion or deletion.

3. TYPES OF FIREWALL TECHNOLOGIES

When Morris Worm carried by emails firstly attacked Internet in a large scale without expectation, Internet experts beagn to cooperate on designing secure software and systems for the Internet. Firewall began to emerge in late 1980s even Internet related things were still quite new to the majority of people all over the world.

Firewall acts as a safe guard who stands in front of a building entrance and checks people who comes in or out to keep the building safe. This section will give a general overview of different firewall technologies. Nowadays we have various kinds of firewalls which have common properties such as resistance to attacks. All traffic must flow through firewall and using access control policy. However, different types of firewall have different advantages and limitations. This mainly due to firewalls usually also contains other technologies such as network address translation, content filtering and so on. With a good understanding of their differences and their respectively provided features, we can choose and implement appropriate type of a firewall to best meet the network security needs.

3.1 Packet Filtering Firewall

Packet filtering firewall is the first firewall technology introduced by Jeff Mogul from Digital Equipment Corporation (DEC) in 1988 [19]. This first generation of packet filtering firewall mainly works at the network layer of the Open Systems Interconnection (OSI) model. And it also uses transport layer information such as port numbers to deny or permit traffic (e.g. traffic sent by an outside web server to the TCP port 80 could be configured to be blocked).

Packet filtering is based on using ACLs to block connenction to or from a specific network, hosts and ports. It filters traffic according to source and destination IP addresses, source and destination TCP/UDP port numbers, protocols and packet types. However, initial packet filtering firewalls are stateless as they do not know whether a packet is a part of the existing traffic stream or not. As a result, this kind of firewall is also called stateless inspection firewall. Although fewer stateless firewalls are used nowadays, they are indeed the core foundation of most modern and advanced firewalls. Content below are advantages and limitations of the first generation packet filtering firewall.

Advantages of stateless packet filtering firewall are [19]:

- Location flexibility: network devices like routers, switches, and wireless access points configured with ACLs can be packet filtering firewall to control traffic for security [20].

- Packet filters firewalls filter packets faster without inspecting packet content.
- Packet filters have a low impact on network performance and work efficiently.
- Packets filters are easy to implement (a network router with simple access control lists could be a simple packet filtering firewall), maintain, and are supported by most operating systems.
- Security at the Network Layer can be provided by a packet filter.
- A packet filter can perform almost all tasks of a high-end firewall at a much lower cost.

Disadvantages of stateless packet filtering firewall are [19]:

- Packet filter firewall is weak to IP spoofing. If hackers modify the IP address information in the packets header and send that packets pretending from a trusted network to the firewall; and packets fitting the ACL criteria can pass. If those passed packets contain malicious code, the internal network will be in high risk. Some spoofing attacks can be mitigated by verifying the session state in a firewall that works at higher layers. Authenticating the users before allowing the packets is also an effective way to prevent IP spoofing.
- Packets filter firewall may be vulnerable to Overlapping Fragment Attack which is to bypass the firewall and gain access to the targeted network or a host. Because packet filters filter fragmented IP packets based on the TCP header information in the first fragment and all fragments after the first fragment are passed unconditionally. Attacker can overwrite part of the TCP header information in the first fragmentation which can still pass the firewall. Then following fragments can be inserted with malicious data and pass the firewall. Common example is to change service type by changing the destination port number. For example, change port 25 (SMTP) to port 23(Telnet), traffic normally will be denied by the router could pass the packet filtering firewall in this case [21]. Firewalls can be configured to block fragmented packets to mitigate the fragment attack. However fragment packets sent by the legitimate users via the Internet could be blocked as well, so it is not recommend.
- Packet filter firewalls cannot dynamically filter certain services. For example, sessions that use dynamic port negotiations are difficult to filter without opening access to a whole range of ports.
- Packet filter firewalls are stateless. They examine each packet header rather than content showing the state. Anything even like viruses fitting the simple rules can pass and sender of packet is not authenticates, which will cause security issues.

- Packet filtering firewall cannot resist the attacks on application layer as it does not check the packets' contents.

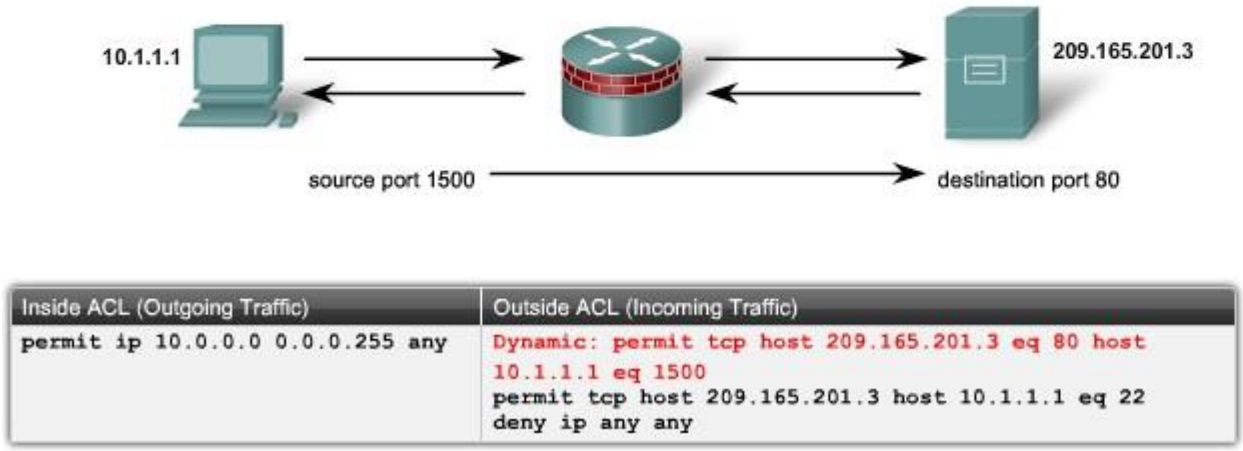
3.2 Stateful Inspection Firewall

Stateless packet filtering also known as static packet filtering cannot recognize whether the packet is part of an existing flow of data or not. Allow or deny decisions are made packet by packet basis rather than based on previous allowed or denied packets. As result, a stateless packet filtering firewall is not smart enough and can be fooled by professional hackers. To solve this problem, stateful inspection firewall was firstly developed by Bill Cheswick and Steve Bellovin at AT&T Bell Laboratories in 1989 [22].

Stateful inspection firewalls are based on packet filtering firewall and can monitor the connection state such as initiation, data transfer and termination. To point out that, the terms to describe the connection state may be a little different from various firewall products but quite similar. This kind of firewall is also known as dynamic packet filtering firewall and it uses information at layer 3, 4 and 5 of the OSI model. [22]

Stateful inspection firewalls track the communication process by using a state table. Naturally, about the firewalls' state table may also vary from a vendor to a vendor. Following information is about the Cisco stateful firewall's state table. Every time when a TCP or UDP connection is established, a firewall will log information in the stateful session flow table. The session flow table includes sorce and destination addresses, port numbers, TCP sequencing information and flags for TCP or UDP connection in each associated session. When firewall receives a packet from outside network, it will compare the received packets with saved state table information to make the allowed or denied decisions [22]. In addition, for stateful inspection firewalls who add the network address translation (NAT) feature, their state tables will also include some NAT information [32].

More advanced stateful inspection firewalls can update state the table and allow returning traffic initiated from inside network. This reduces DNS cache poisoning (which could cause the name server to return incorrect address and make the attackers' device acts as DNS server to trick users communicate with them) and TCP RST flood attacks (a kind of DOS attack by sending succession of SYN requests to targeted systems and cause server unresponsive to legitimate users) [23]. However, as stateful inspection offers transparency, internal IP addresses are exposed as shown in Figure 7 below. This could cause potential threats for the network.



Inside ACL (Outgoing Traffic)	Outside ACL (Incoming Traffic)
permit ip 10.0.0.0 0.0.0.255 any	Dynamic: permit tcp host 209.165.201.3 eq 80 host 10.1.1.1 eq 1500 permit tcp host 209.165.201.3 host 10.1.1.1 eq 22 deny ip any any

Figure 7 Stateful Inspection Firewall [24]

Below are some advantages of the stateful inspection firewall [22]:

- Provide more secure and better performance than packet filtering firewall.
- Stateful packet filtering firewall can prevent some spoofing and DoS attacks by inspecting connection state or verifying whether packets from an authorized source or not.

Disadvantages of the stateful inspection firewall are listed below [22]:

- Compared with stateless firewalls, stateful inspection firewall requires more memory to track the connection state.
- It is also harder to manage than a stateless firewall
- Stateful packet filtering is vulnerable to application layer attacks as it doesn't check contents of the connection.
- Some connectionless protocols like UDP and ICMP do not produce connection information for the state table. They are quite different form connection-oriented TCP who has the three-way handshake process. UDP and ICMP packets do not have sequence number or flags, so stateful inspection firewalls can just check the source and destination IP address, ports and other information according to the rules to allow the packets. What is more, because of UDP and ICMP are connectionless; firewall cannot know when the session is end. So the UDP or ICMP session's entries will be removed from the state table when the predefined timeout value is reached. This prevents entries from filling the state table and may let some UDP or ICMP traffic pass the firewall without entries. [32] [33]
- Stateful inspection firewall doesn't support user authentication.

3.3 Application Layer Filtering Firewall

With development of the attacking methods, threats move from lower layers to the higher layers like application layer. As we know from above, a packet filtering firewall and a stateful inspection firewall analyze a packet's addresses, ports, protocols, or a packet state but still not that deep in application protocols and content of the packets. Absence in checking contents including payload gives hackers opportunities to attack the application layer directly. So the third generation of the firewall called application layer filtering firewall is invented to defend application layer attacks.

Application layer filtering firewall filters traffic based on information on layers 3, 4, 5 and 7 of the OSI model. Application level firewall can examine contents of the header information according to specific keywords or character strings like words or phrases and then block messages matching the rules [25]. Most of the application firewalls control and filter traffic based on software. This means that the application firewall filters content and blocks unwanted traffic with the help of software program.

Advantages of the application layer filtering firewall are listed as follows:

- Provide higher level security when compared with other types of firewall. Application layer filtering firewall examines content of the data, so it can prevent attacks based on application layer protocols. Examples are SMTP, POP3 and DNS buffer overflows; web attacks based on HTTP header and request information; attack code hidden with SSL tunnels[25]. For a detiled example like web based attacks, we can use keyword ”aaa” specified by the firewall to filter URL http://www.badthings.com/aaa.html.

Disadvantage of the application layer filtering firewall are showed as below [25]:

- A main shortcut of the application firewall is that the processing is slow due to checking contents of the packets. As a result, network performance may be degraded in certain degree.
- Application firewall needs support from powerful hardware when compared with packet filtering firewall.
- In addition, it is more expensive than the other types of firewall.
- The last disadvantage is its complexity and relatively difficult to implement and maintain.

3.4 Other kinds of Firewalls

Except firewalls I mention above, there are also other kind of firewalls such as host-based firewall and hybrid firewalls in modern networks. Those firewalls add flexibility and diversity to the network according to different specification and needs.

3.4.1 Host-based Firewall

Host-based (server and personal) firewall means a host or server with firewall software running on it. Perimeter-based firewalls have to deal with large number of applications and services. Sometimes it is possible for them to leave out some malicious threats which access into the internal network. And it is widely known that perimeter-based firewalls don't protect against threats inside the network. So a host-based firewall can be used to prevent threats mentioned before. It can block incoming or outgoing traffic but only on that host or server and not the network it resides [28]. With the firewall software, host can be less likely affected by malware or other hosts. Host-based firewalls for servers usually use rule sets to deny or allow traffic like the network firewall does. Host-based firewall also cooperates with other software like antivirus software and intrusion prevention software to provide security.

3.4.2 Distributed Firewall

Except the personal firewall, more advanced firewall like a distributed firewall is developed to prevent inside network attacks. Unlike a traditional firewall that assumes the internal network is trusted while the external network is untrusted, a distributed firewall thinks both internal and external network are untrusted [26]. By filtering traffic from both the Internet and inside network, attacks launched from the inside network can be prevented.

A distributed firewall is based on host-resident software applications, and it can provide highly targeted security policy for the hosts which run specific applications. A distributed firewall is usually behind the perimeter firewall and acts as the second screen. Like a big building within many rooms, perimeter firewall acts as the main guard in the building entrance and distributed firewall provides features that act as guard in the room entrance.

Distribution firewall is not a single product but an entire system that has a perimeter firewall, a central management server, and end-user machines. There are three main components of the distributed firewall. The first one is the central management system which centrally deploys, monitors, or updates security policies. The central management feature reduces workload for the administrator and also reduces costs on maintance. The second component is to distribute the policies to end-users in a secure way. Finally, when security polies sent by the central management server arrive to the hosts, hosts will implement the polies to allow or deny traffic. [26]

A distributed firewall is similar to personal firewall but it has more advantages as it offers central management, logging and other features [26]. Security policy on the personal firewall can be configured by the system users themselves and the policies or management is only valid on that individual host. Security policies on end-users within the distributed firewall are managed by the network administrator. And those security polices cannot be changed by the client users. Security policy could be implemented on every detailed corner of the network where the distributed firewall resides rather than just generally between networks.

In summary, the distributed firewall strengthens system security in terms of intrusion detection and intrusion protection systems for the hosts. By implementing comprehensive security policy on the hosts, inside attacks can be prevented. Central management also enhances network administrators’s work efficiency. In addition, it reduces bottleneck problem on the perimeter firewall and improves network performance like faster processing speed. Network monitoring, syslog management and other tools also enhance the security level.

3.4.3 Application Proxy Gateway

A proxy server can also acts as a firewall to allow or deny packets and works on behalf of the network users. It can be a dedicate hardware or a general machine running some software. Proxy is a special network service used by clients to indirectly connect to another host. Many firewalls and routers have this kind of proxy function to provide privacy and security for protected network.

Firewalls that have a proxy agent act as a bridge between the internal network and Internet. When host A in the internal network wants to communicate with host B via the Internet, it should firstly connect to the proxy server and then the proxy server connects to the host B. Both host A and host B are invisible to each other but they are respectively visible to the proxy server. They never communicate directly but always via proxy server. As host B on the external network cannot see the inside IP addresses, the internal network is less likely to be attacked by the hacker. This is quite similar to the network address translation.

The proxy agent uses the firewall rules to deny or allow traffic. There are different types of proxy gateways or proxy firewalls according to which layer they mainly operate.

An application proxy gateway is similar to an application firewall and works at the application layer. It also examines the content of the traffic and checks application protocol headers or payloads to allow or deny traffic. There are some differences between the application proxy gateway and the application firewall. Firstly, the application proxy gateway is safer as it prevents direct connection between two hosts [32]. Secondly, it provides safer mechanism like decrypting, and re-encrypting packets before sending them [32].

There are different dedicated application proxy servers according to different protocols.
And different dedicated proxy servers provide different services. For example, a HTTP proxy server provides web services and a Simple Mail Transport Protocol (SMTP) proxy server provides email services. Normally, a proxy server is placed behind a firewall and deals with traffic to or from the main firewall. Before forwarding traffic to the internal hosts or passing traffic to the firewall, the proxy server will check and filter the traffic [32]. Dedicated proxy servers only accept packets generated by the service that the proxy server dedicates [27]. For example, Telnet traffic is only handled by the Telnet proxy [27]. An application proxy gateway that only runs Telnet proxy can allow packets generated with Telnet service and deny all other packets with other services. Dedicated proxies can reduce workload for the firewall as it can filter and log some application traffic that maybe hard to check for the firewall [32].

There are some disadvantages of the application proxy firewall, it is more difficult and complex to configure. For the dedicated application proxy, an administrator should be familiar with the protocols to configure related security policies. Otherwise, misconfiguration would negatively affect the network security level. In addition, the processing speed is slow and the performance is degraded.

Because of those problems, application proxy servers are gradually ignored by many vendors and replaced by adaptive proxies like Cisco Adaptive Security Appliance (ASA). Adaptive proxy firewalls combine security of the proxy firewall and speed of the dynamic packet firewall as well as other features.

3.4.4 Unified Threats Management (UTM)

Many modern firewalls are combined with several features to apply on a single device or single system. Similarly, Unified Threats Management (UTM) also adapts the idea to integrate multiple security products to perform multidimensional security features on a single appliance. For example, it usually combines firewall, anti-virus, network intrusion detection and other features.

With those all-inclusive security features, organizations especially small and medium size organizations can save money when buying devices. UTM also reduces the complexity and simplifies installation as well as maintenance of the device. Web-based GUI management also reduces workload of the network administrator. As every coin has two sides, one main limitation of the UTM is the performance. With a single device and single system to deal with multiple tasks will take longer time and consumes CPU resources. As a result, the processing may be slower and it needs support of powerful hardware. When compared with some single feature provided by an individual device, the performance of that feature on UTM may also degrade because of compromising with other multiple functions. [32]

4 FIREWALL AND NETWORK ARCHITECTURE

After we now know secure components in firewalls and types of firewalls, it is time to analyze where we should put the firewall in the network topology. This section is mainly to introduce commonly used firewall architectures in the network.

4.1 Simple Network with Firewall or Bastion Host

In Chapter 3 we already learnt that packet filtering is the basic feature of a firewall and devices like router can also acts as simple firewall. Below Figure 8 shows a simple packet filtering router acting as a firewall between a trusted network (private network) and an untrusted network (public network like Internet). The router is directly connected to both networks to provide simple network security features. You can apply different types of Cisco IOS ACLs to control network access.

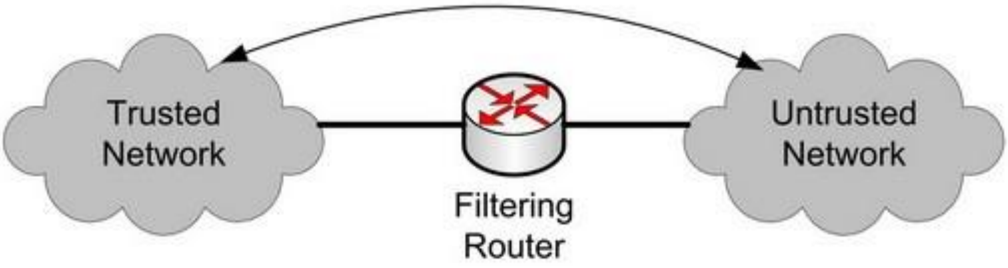


Figure 8 A Simple Router as a Firewall [36]

4.2 Firewall with Proxy Server

As Internet is an open system, sometimes we do not want to reveal our internal network. We can use a proxy server as a “fog screen” on the network to minimize exposed connection to the Internet.

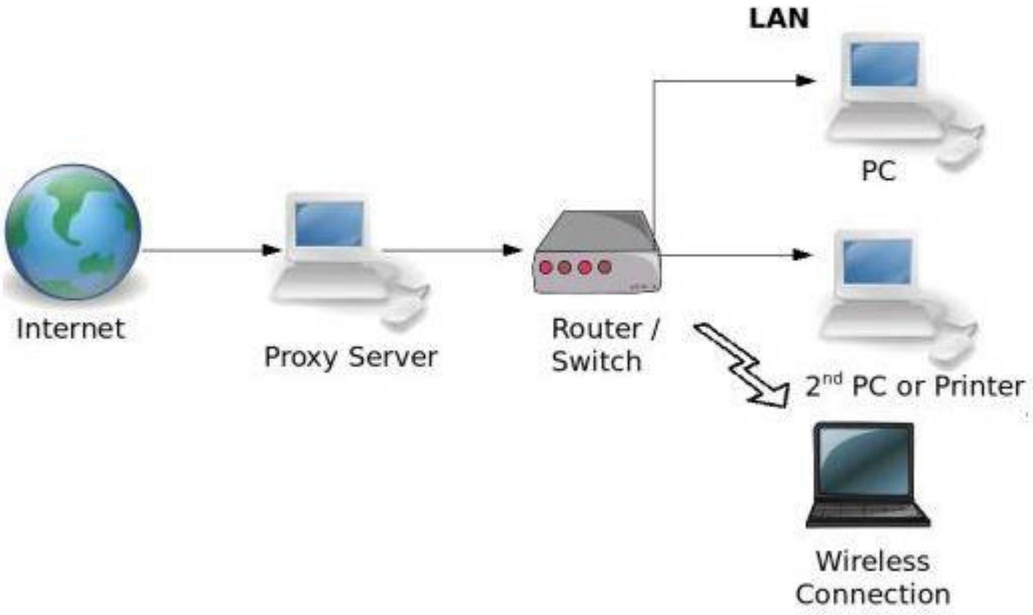


Figure 9 Proxy Server [38]

Figure 9 shows network topology with a proxy server. Hosts in the LAN want to communicate with outside world and packets should be processed by the proxy before being forwarded to the Internet. Similarly, returning packets from the Internet should also be processed by the proxy server before sending them to the internal host. A router in the network acts as a firewall and can control network access by utilizing access lists. Except allowing and denying traffic to a network, router also routes IP packets.

4.3 Firewalls with DMZ

We learnt in Section 2.5 that DMZ is a subnet network to place servers and it also separates the internal network from the public network. No matter how to implement a network with DMZ, basic access control principles are as follows [14]:

- Internal network can freely access the Internet but this needs the firewall to translate internal source addresses to the public one.
- Internal network can access the DMZ; this makes convenience for internal users administrate the servers in DMZ as well as share resources provided by the DMZ servers.
- External network can access the DMZ as primary aim of the servers in DMZ is to provide services for the public network. Additionally, when outsiders access the DMZ, the firewall needs to translate public address to address recognized in the DMZ.
- External network cannot access the internal network as data in the internal network is private and not public.
- DMZ cannot access the internal network otherwise when hackers from Internet intrudes DMZ, they can attack the internal network in further.
- Generally, DMZ cannot access the Internet except when DMZ houses email server which needs to access the Internet.

Below are two types of firewall architectures within the DMZ implantation.

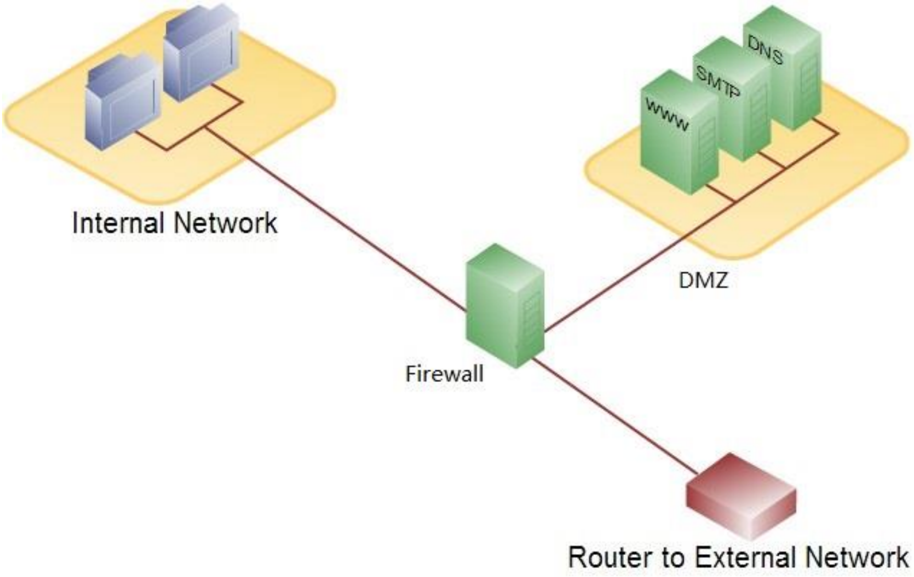


Figure 4 Single Firewall [15]

Figure 4 shows the single firewall topology. There is only one firewall with three network interfaces which respectively connect the external network, DMZ and internal network. Any communications between servers in different zones must pass through the firewall under the predefined network security policies.

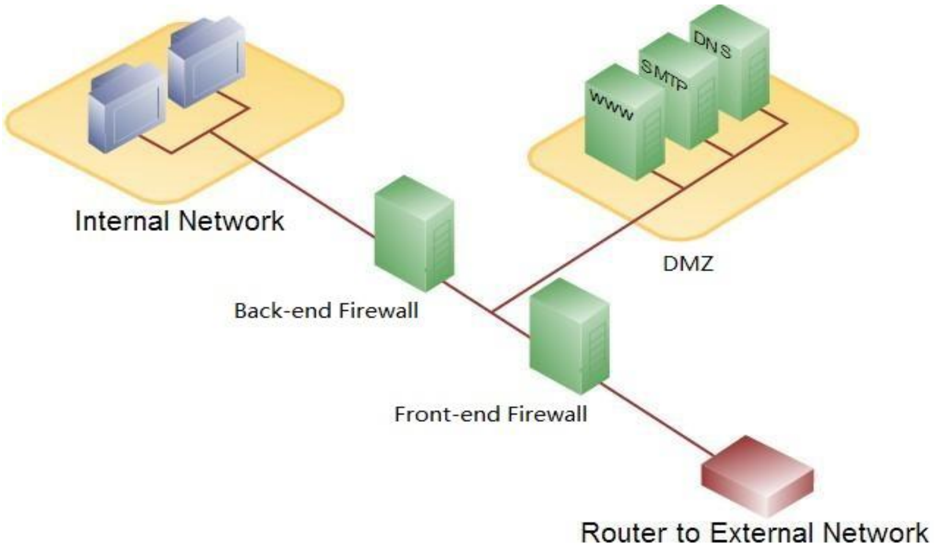


Figure 5 Dual Firewall [15]

Figure 5 shows a more secure way to create a DMZ with two firewalls. The first firewall (also called front-end firewall) connects the DMZ and public network. It is configured to allow traffic destined to the DMZ only. The second firewall (also called back-end firewall) connects internal network and DMZ. It controls traffic between internal network and DMZ. This method is more secure as hackers have to break two firewalls in order to access the internal network but it is also more expensive.

4.4 Firewall Appliance

A firewall appliance can be dedicated hardware and software system that implements the access control policy. It can contain many features like Network Address Translation (NAT), Virtual Private Network (VPN), and De-Militarized Zone (DMZ) and so on. Cisco ASA is one of the commonly used Firewall Appliance for small and medium size organizations. Figure 10 shows a simple topology of the Firewall Appliance.

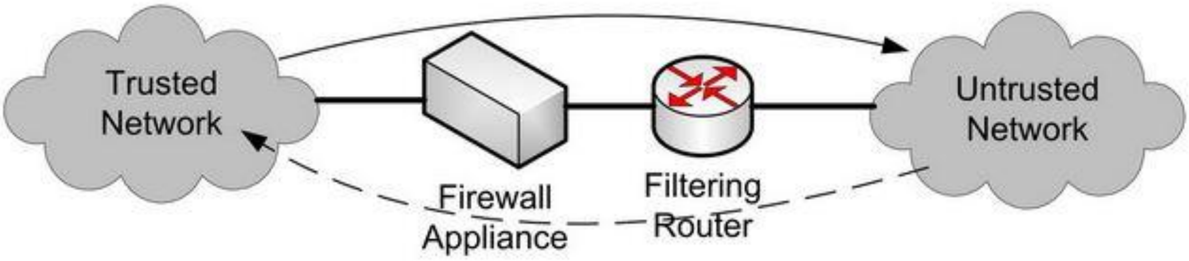


Figure 10 Firewall Appliances [36]

4.5 Firewall with VPN

As I mention in Section 2.6, VPN is added to the modern firewall architecture to provide secure connection for two remote networks over the Internet. Basically, there are two types of VPN networks, one is site-to-site VPN and the other is remote access VPN.

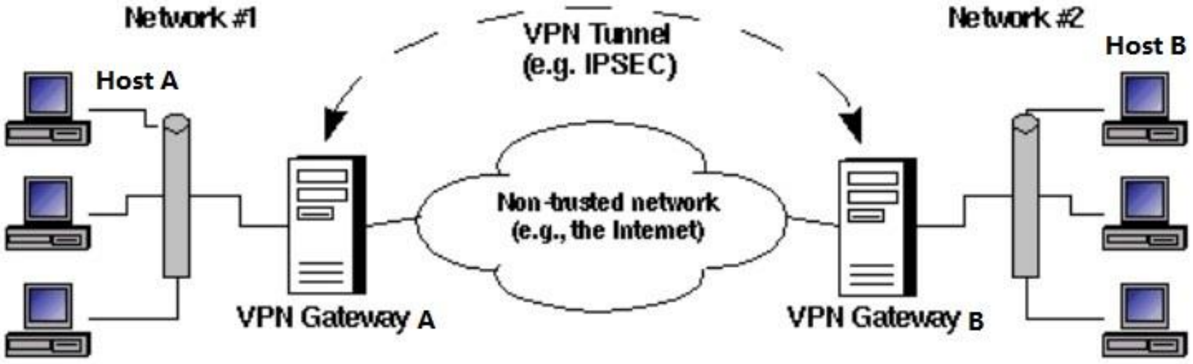


Figure 11 Site-to-site VPN [39]

As mentioned earlier, a site-to-site VPN solution can connect two or more sites in different fixed locations in a secure way. By using site-to-site VPN, an organization with different branch offices can extend its network and internal information resources to its employees. In the site-to-site VPN network, hosts on each side of the network send and receive traffic through the VPN gateway. A VPN gateway is usually placed at the edge of a network and it can be a router, a firewall or Cisco ASA.

Figure 11 shows how the site-to site VPN works. If host A in the network 1 wants to communicate to host B on network 2 via the Internet, it firstly sends traffic to VPN gateway A. Then the VPN gateway A will encrypt the traffic before sending it to the Internet. When the peer VPN gateway, which is VPN gateway B on the network 2 receives the traffic; it will first decrypt the traffic and then forwards the traffic to host B. [40]

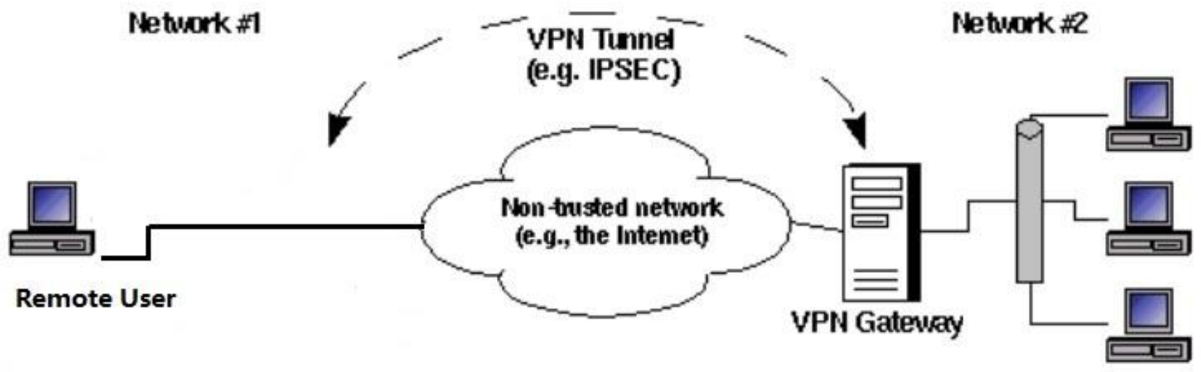


Figure 12 Remote Access VPN [based on 39]

Site-to-site VPN is suitable for network to network connections and the places of the networks are usually fixed. However, how to provide a secure solution for cases that VPN users change their locations frequently? For example, some business man like salesman who needs to travel to different places but still need to access information from the remote company via the Internet.

In this case, remote access VPN is developed for those telecommuters. They can flexibly connect VPN connection by the computer without limitation of the location. In addition, they do not need to set up the VPN connection all the time and can close the connection when unnecessary and establish the connection when needed.

Remote-access VPN could be implemented as client and server architecture and usually the client should download corresponding VPN client software. The VPN client software acts as the VPN gateway as I mention tin the site-to-site VPN. When client sends traffic, VPN client software will encrypt and encapsulate the traffic before sending it to the Internet. When the VPN gateway on the target network receives the traffic, it decrypts the traffic like the VPN gateway in the site-to-site VPN. [40]

5. FIREWALL IMPLEMENTATION

After we get familiar with the firewall technology and architecture, it is time to do the practical firewall implementation. A firewall can be a simple device like a router with integrated firewall capabilities. It can also be a dedicated firewall like the SmoothWall to provide some security features. My practical aim in this section is to implement application layer filtering based on two different firewall solutions. They are a Cisco IOS router configured with CBAC, and a firewall as a proxy by using SmoothWall. Then I will compare the performance differences between them.

To analyze their performance, I will use a network connected by a router but without any firewall capabilities as a reference. For each test, I will choose ICMP, TCP and UDP traffic to test whether the traffic is allowed before or after I configure related rules. In this basic testing environment, I will also choose referred parameters on TCP traffic for evaluating bandwidth. Topics about my testing environment, performance comparisons among different firewall implementations, and conclusion or additional analysis summary for the practical work will be covered below.

5.1 Cisco Router without Firewall Capabilities

In my practical work, I simplify the network topology by only using two networks. One network stands for the trusted network and the other one represents the untrusted network. Only one firewall will be used and I will not use DMZ in the network. The four testing environments and details are shown below.

5.1.1 Tested Environment

Before we evaluate the firewall performance, it is extremely importmant to test its baseline environment. This guarantee that the firewall performance will not be affected by the other factors in the baseline and make the results more reliable. The other purpose of the baseline environment is to decide measurement methods and referenced parameters for latter scenerios.

Figure 13 shows the toplogy of the small baseline network. Two computers with Windows XP and one Cisco 2911 router will be used. Server computer will stand in the outside network while Client computer will be placed in the inside network.

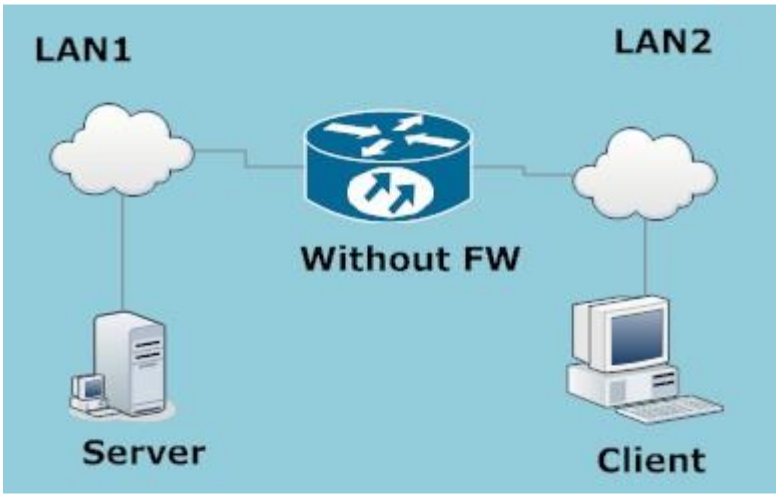


Figure 13 Basic Tested Environment

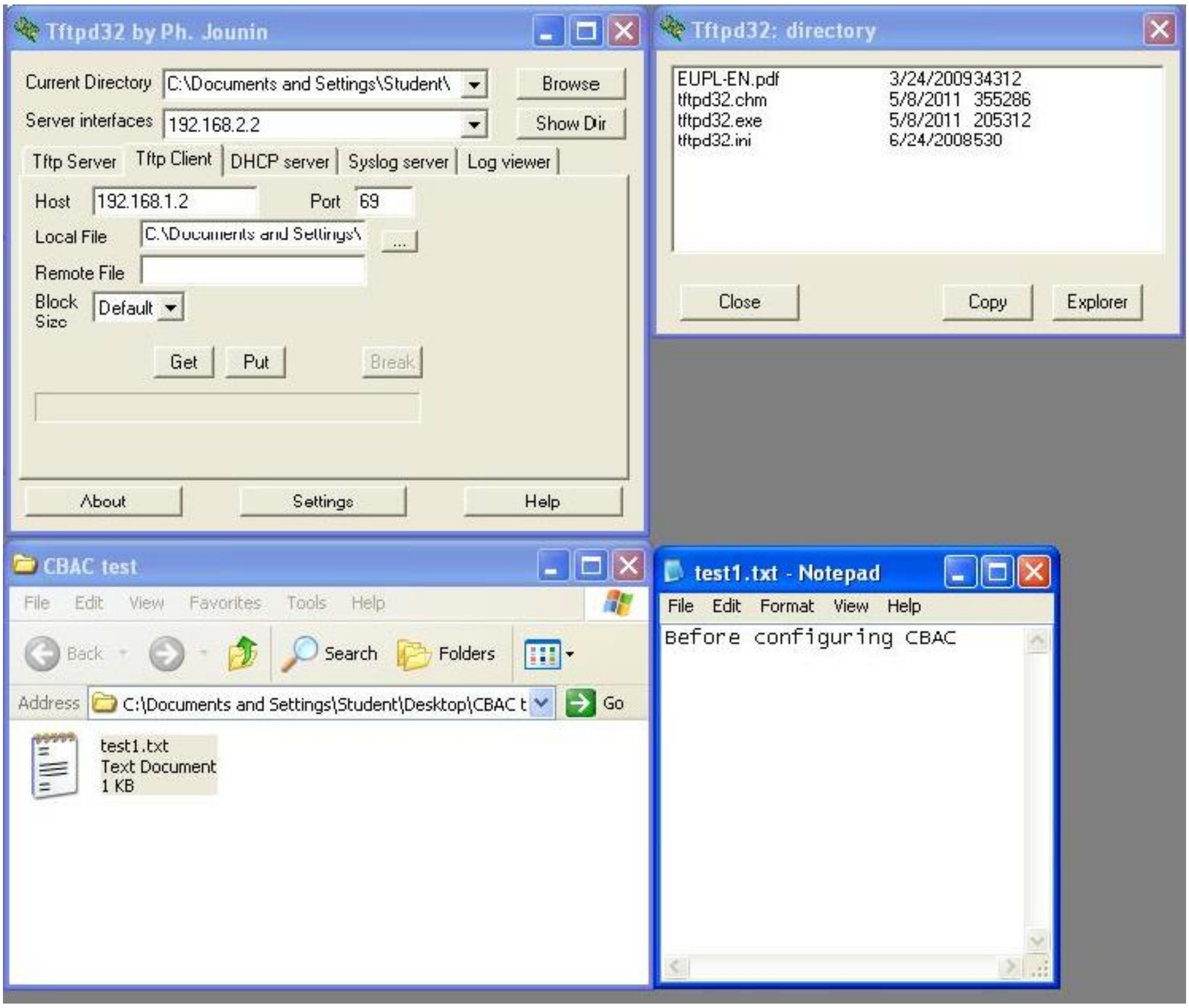
Table 1 shows ip addresses and subnet masks of the network devices according to the topology.

Table 1 Address Plan

Name	IP Address	Subnet mask	Gateway
Server	192.168.1.2	255.255.255.0	192.168.1.1
Client	192.168.2.2	255.255.255.0	192.168.2.1
Outside Interface	192.168.1.1	255.255.255.0	NO
Inside Interface	192.168.2.1	255.255.255.0	NO

I test ICMP, TCP and UDP traffic before I measure the network performance by sending echo request, web request and TFTP file transfer request. I ping from the client computer to the server computer and vice versa, both pings are successful. Then I use a web browser on the client computer and type the IP address of server computer to test the TCP traffic. The client computer can successfully access to the homepage of the server computer and vice versa.

Finally, to test the UDP traffic, I download tftpd32 software on both server computer and client computer. On the server computer, I firstly open the tftpd32 software and choose TFTP Server. Then on the client computer, I create some text file which will be transferred during the test. Figure 14 shows the content of the text file I will transfer and configuration of the client TFTP. Firstly, I open the software and choose the TFTP client to send TFTP file. Host is the IP address of the server on which the TFTP server is installed. The port number is 69 and the protocol is UDP. Then I locate the test1 text file and click the Put button. We can see the settings on the TFTP client and text file content from Figure 14.



Tftpd32 by Ph. Jounin

Current Directory: C:\Documents and Settings\Student\

Server interfaces: 192.168.2.2

Host: 192.168.1.2 Port: 69

Local File: C:\Documents and Settings\

Remote File:

Block Size: Default

Tftpd32: directory

```
EUPL-EN.pdf      3/24/200934312
tftpd32.chm      5/8/2011  355286
tftpd32.exe      5/8/2011  205312
tftpd32.ini      6/24/2008530
```

CBAC test

Address: C:\Documents and Settings\Student\Desktop\CBAC t

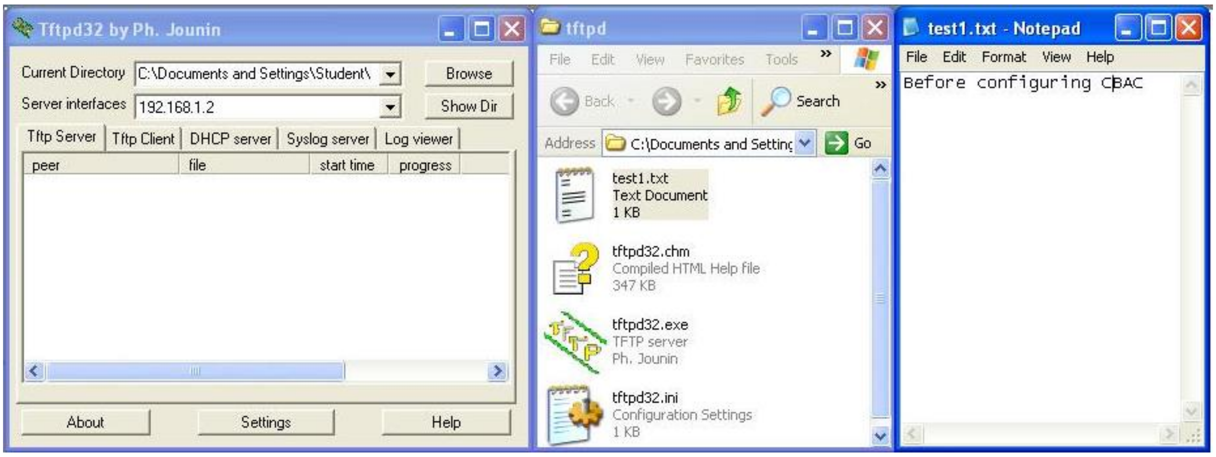
test1.txt
Text Document
1 KB

test1.txt - Notepad

```
Before configuring CBAC
```

Figure 14 TFTP Client Configurations

Then the TFTP server will receive the text file and we can see it on the directory. Figure 15 shows that the TFTP Server receives the text file and we can also see the content of the received file.



Current Directory C:\Documents and Settings\Student\

Server interfaces 192.168.1.2

Address C:\Documents and Settinç

test1.txt
Text Document
1 KB

tftpd32.chm
Compiled HTML Help file
347 KB

tftpd32.exe
TFTP server
Ph. Jounin

tftpd32.ini
Configuration Settings
1 KB

```
Before configuring CBAC
```

Figure15 Results on TFTP server

In summary, all the traffic like TCP, UDP and ICMP are verified normal before I measure the network performance.

5.1.2 Network Performance Measurement Methods

After the basic environment is tested, network performance will respectively be evaluated by network latency and packet lost rate. Bandwidth will also be used as a reference of the network performance. Then based on the results, I can choose referred parameters for other firewall implementations and compare or analyze the results of the tests.

5.1.2.1 Network Latency and Packet Lost Rate

I simply use ping command to test the network latency and packet lost rate. Here the network latency will use Round-trip Time (RTT) as a reference. The unit of RTT is millisecond and I will use the ms as its abbreviation. In general, three different packet sizes like small, medium and large will be used. For each packet size, I will test 5 times and for each time I send 500 ICMP requests. Then I calculate the average RTT, average of the minimum RTT and average maximum RTT as well as average of packet lost rate according to the results of the 5 tests.

Among the 5 tests, the result of the each test mostly varies a little different in terms of maximum RTT when one or two ICMP requests among the 500 requests take longer time. Usually, the number of those packets is few. As a result, the average RTT is almost same to the minimum RTT when majority of the packets use the minimum RTT.

Table 2 Network Latency and Packet Lost Rate

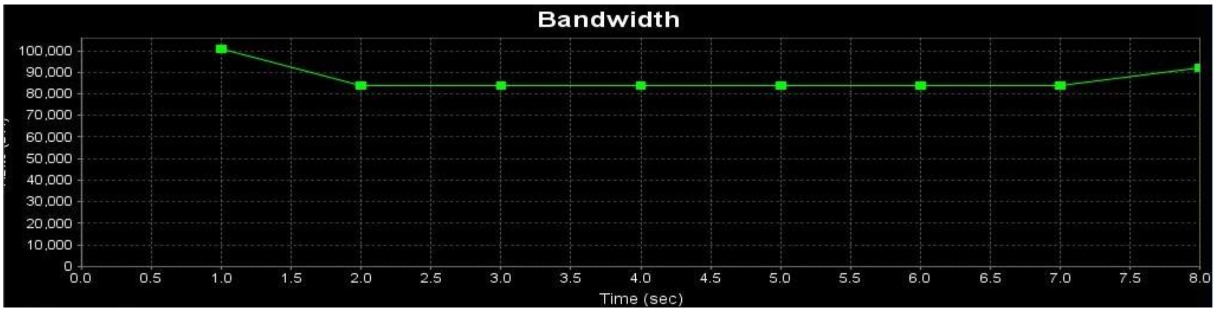
Packet size (Bytes)	Number of Requests	Lost	Min (ms)	Max (ms)	Average (ms)
32	500	0 (0%)	0	0.8	0
35000	500	1.1 (0%)	7	8.4	7
65000	500	2.6 (0%)	13	21.8	13

Table 2 shows that in the base environment with same number of requests, the bigger the packet size the larger dealy will the network has. Packet lost rate will also increase correspondingly when the packet size grows. In each case of packet size, the average value is equal to the minimum value. This means that only few ICMP requests have higher RTT.

5.1.2.2 Bandwidth

To measure the network bandwidth, I download Jperf software on two computers. I will use Jperf server on the Server computer and the client mode on the Client computer. TCP with different parameters will be tested individually. For the TCP traffic, buffer length, TCP window size and Max Segment Size will be separately set with different values to determine which value has higher bandwidth. Then I will choose a good combination of those three factors in other tests.

Firstly for the TCP Buffer length test, I use the following ascending buffer sizes 1Mbytes, 2 Mbytes, 4 Mbytes, 8Mbytes, 16Mbytes and 32 Mbytes.



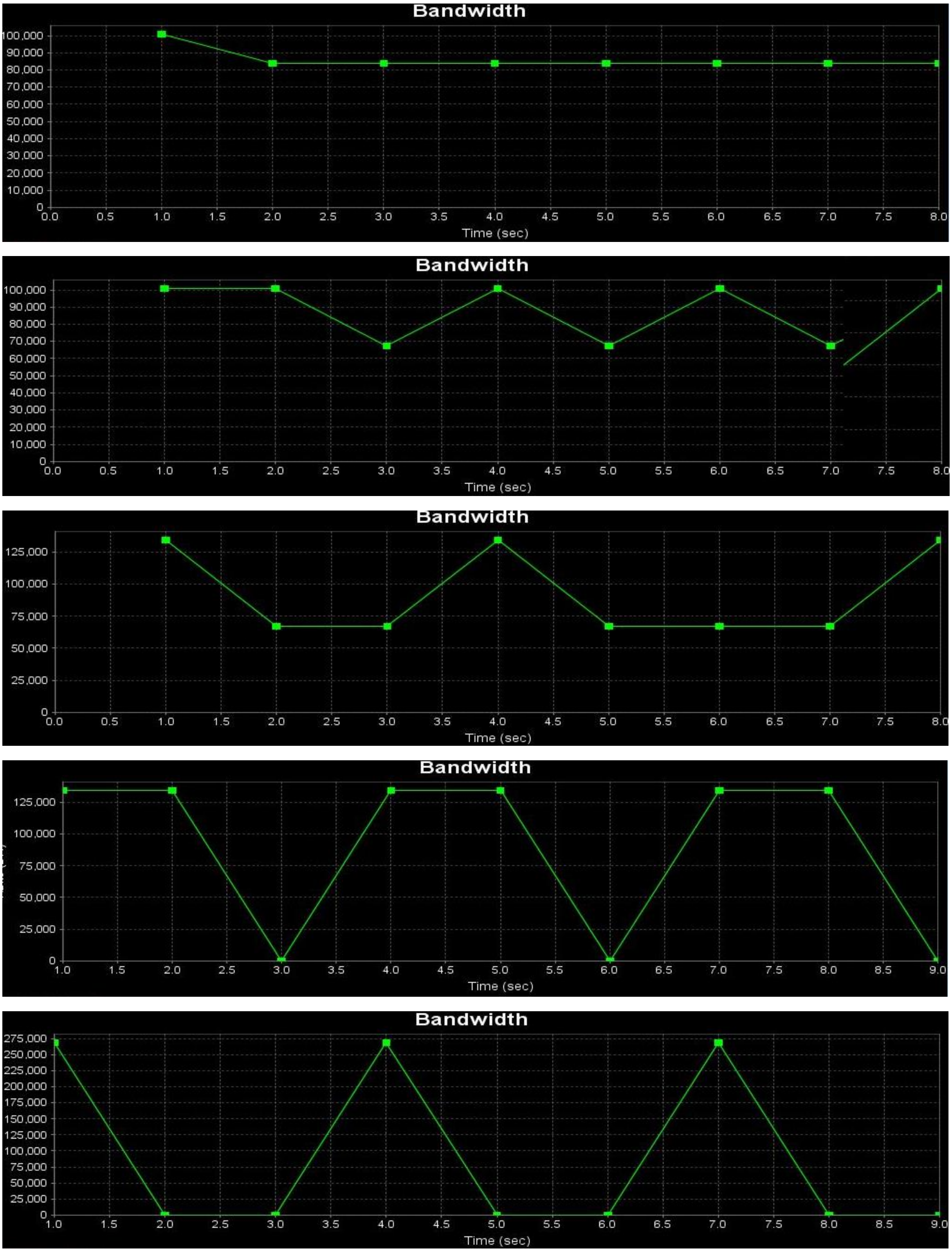


Figure 16 TCP Buffer Size and Bandwidth

From Figure 16 above, we can see that with lower buffer size, the bandwidth is more stable with relatively lower value like 83886 Kbytes. If the buffer size is higher, bandwidth will fluctuate and change more dramatically if the buffer size is more than 16 Mbytes. As a result, I will choose 1 Mbytes for the buffer size to get a stable bandwidth result for the later tests.

Then for the TCP window size, I use different window sizes. Table 3 shows that the bandwidth will gradually become bigger when the window size grows until it exceeds 32 Kbytes. After 32 Kbytes, bandwidth will decrease a little bit. According to the result, I will use 32 Kbytes as a reference value for the TCP window size.

Table 3 TCP Window Size and Bandwidth

Window size (Kbytes)	1	2	4	8	16	32	64	128
Bandwidth (Kbits/sec)	57584	57576	58638	62538	71974	84687	84581	84123

Thirdly, for the TCP Max Segment Size (MSS), I also use varied values ranged from 300 Kbytes to 1800 Kbytes to show the trend of bandwidth when MSS grows. Table 4 shows that bandwidth will become bigger when MSS grows until its value is 1500 Kbytes. If the MSS value is over 1500 Kbytes, bandwidth will decrease a little bit. Based on the results, I will choose 1500 Kbytes as the referred MSS for TCP traffic.

Table 4 TCP MSS and Bandwidth

MSS (Kbytes)	300	600	900	1200	1500	1800
Bandwidth (Kbits/sec)	57805	51823	61884	62097	62277	61917

In summary, for the TCP traffic, bandwidth will be set by buffer size with 1 Mbytes, window size with 32 Kbytes and MSS with 1500 Kbytes. With those settings configured for the TCP traffic, I test 5 times and use the average value for bandwidth in the baseline environment of router without firewall. Table 5 shows that the bandwidth in this testing environment is 84348 Kbits/sec.

Table 5 TCP Bandwidth in Router without Firewall

Times	1	2	3	4	5	Average
Bandwidth	84394	84336	84498	84336	84175	84348

(Kbits/sec)						

5.2 IOS Router configured with CBAC

In this section, Cisco router will be configured with CBAC feature and acts as an integrated firewall to control traffic between two networks. In this CBAC settings, the inside interface will be configured with inbound ACLs to inspect outgoing traffic. And the outside interface will be set with dynamic ACLs to allow traffic originated from inside network to return back. This dynamic ACL is temporary and will be removed when the session is over. All the ACLs will be numbered and extended ACLs.

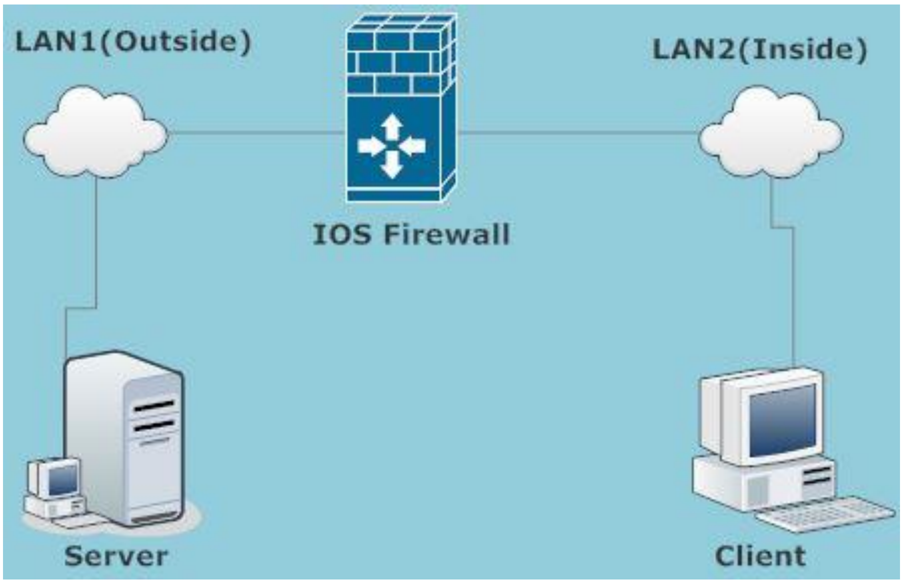


Figure 17 Router with CBAC

Figure 17 shows the topology of this tested network. I just continue to use the baseline network and add extended ACLs on both inside and outside interface in order to control traffic. The network will be implemented with the aims below:

- Client on inside network can send ICMP echo request
- Client on inside network can access the homepage of the server
- Client can send TFTP file to the server and the server can read the information in the received file.
- Inside interface will inspect outgoing traffic and outside interface will allow return traffic initiated by the client

- Other requests from inside network to outside and request from outside to inside network will be denied.

To understand how the CBAC works, I also divide this part into three processes: before configuring the CBAC, before configuring the inspecting rules and after configuring the inspecting rules.

Before configuring the CBAC

Before configure the ACLs, as I mention in the baseline test, I test the ICMP traffic by ping method, UDP traffic by tftp file transfer and TCP traffic by using web request. They all work well from inside network towards outside network and vice versa.

Creating ACLs without Inspection Rules

To allow client in the inside network to send web page requests and to transfer tftp files as well as to ping other hosts, I use the configuration command below:

```
CBAC (config)# access-list 101 permit tcp 192.168.2.0 0.0.0.255 any
CBAC (config)# access-list 101 permit udp 192.168.2.0 0.0.0.255 any
CBAC (config)# access-list 101 permit icmp 192.168.2.0 0.0.0.255 any
CBAC (config)# access-list 101 deny ip any any
```

Then I apply those rules on the inside interface with the inbound direction, so all those three requests are allowed to go out this interface and other undefined traffic will be blocked.

```
CBAC (config)# int gigabitethernet 0/2
CBAC (config)# ip access-group 101 in
```

To only allow the icmp traffic to return from outside and deny all the other traffic from outside, I use the command below:

```
CBAC (config)# access-list 102 permit icmp any any echo-reply
CBAC (config)# access-list 102 permit icmp any any unreachable
CBAC (config)# access-list 102 permit icmp any any time-exceeded
CBAC (config)# access-list 102 deny ip any any
```

Then I apply those ACLs rules on outside interface with the inbound direction.

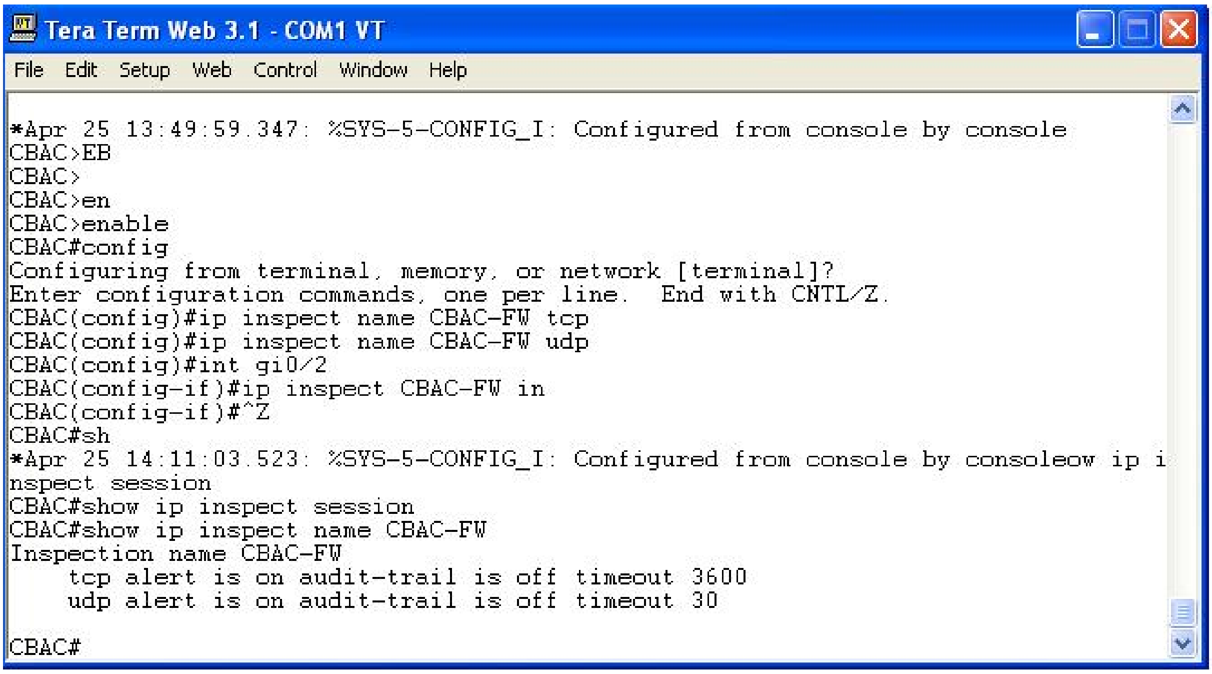
```
CBAC (config)# int gigabitethernet 0/1
CBAC (config)# ip access-group 102 in
```

Then I test the icmp traffic and web services as well as tftp file transfer services before I apply the inspection rules.

Pinging from the server to the client is failed because of the deny ip any any command. However, ping from the client to the server is allowed because of the command permit icmp any any echo-reply. For the tcp traffic test, requests on webpage at either the client computer or the server computer are not allowed. For the file transfer, the file can be transferred via the UDP 69 port but the contents of the text file are invisible.

➢ Adding Inspection Rules

In order to let the returning tcp and udp traffic back to inside network, I add the commands as Figure 19 shows.


Tera Term Web 3.1 - COM1 VT
File Edit Setup Web Control Window Help

```
*Apr 25 13:49:59.347: %SYS-5-CONFIG_I: Configured from console by console
CBAC>EB
CBAC>
CBAC>en
CBAC>enable
CBAC#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line.  End with CNTL/Z.
CBAC(config)#ip inspect name CBAC-FW tcp
CBAC(config)#ip inspect name CBAC-FW udp
CBAC(config)#int gi0/2
CBAC(config-if)#ip inspect CBAC-FW in
CBAC(config-if)#^Z
CBAC#sh
*Apr 25 14:11:03.523: %SYS-5-CONFIG_I: Configured from console by consoleow ip i
nspect session
CBAC#show ip inspect session
CBAC#show ip inspect name CBAC-FW
Inspection name CBAC-FW
    tcp alert is on audit-trail is off timeout 3600
    udp alert is on audit-trail is off timeout 30

CBAC#
```

Figure 19 Commands on Inspection

Ping from the client to the server is still allowed and from the server to the client is still denied. The web request from the server to the client is continuously denied while the web request from the client to the server is allowed this time. For the TFTP file transfer, even I inspect the UDP traffic; the content of the text file received by the TFTP server is still invisible until I apply inspection rules on the tftp traffic.

After I ensure those fundamental ACLs works, I reconfigure the router and add 300 lines of unless ACL statements before the valid ACL statements. Then I test the network performance to see whether length of the ACLs affect the network bandwidth.

5.2.1 Network Latency and Packet Lost Rate

The testing method on network latency and packet lost rate is same as that I use in the baseline environment. The results are showed in Table 6.

Table 6 Latency and Packet Lost Rate in Router with Firewall

Packet size (Bytes)	Number of Requests	Lost	Min (ms)	Max (ms)	Average (ms)
32	500	0 (0%)	0	0.8	0
35000	500	1.4 (0%)	7	9.8	7
65000	500	2.6 (0%)	13	54.2	13

5.2.2 Bandwidth

Table 7 shows TCP bandwidth when I use 1 Mbytes buffer length, 32 Kbytes in window size and 1500 Kbytes in max segment size. I test in 30 seconds and 5 times.

Table 7 TCP Bandwidth in Router with Firewall

Times	1	2	3	4	5	Average
Bandwidth (Kbits/sec)	83600	83538	83686	84139	84199	83832.4

5.3 Firewall implemented with Proxy Server (SmoothWall)

SmoothWall is a Linux-based open source firewall and easy to use because of friendly web-based GUI. In this section I will use SmoothWall as proxy server to filter traffic. Figure 20 shows the topology of the network. At this part, I use the third computer to install and to run the SmoothWall. In the installation, Green interface is configured with IP address 192.168.2.1/24 while the red one is set with IP address 192.168.1.1/24. They just like inside and outside interface I use in previous scenarios. Client and Server configurations are same as before.

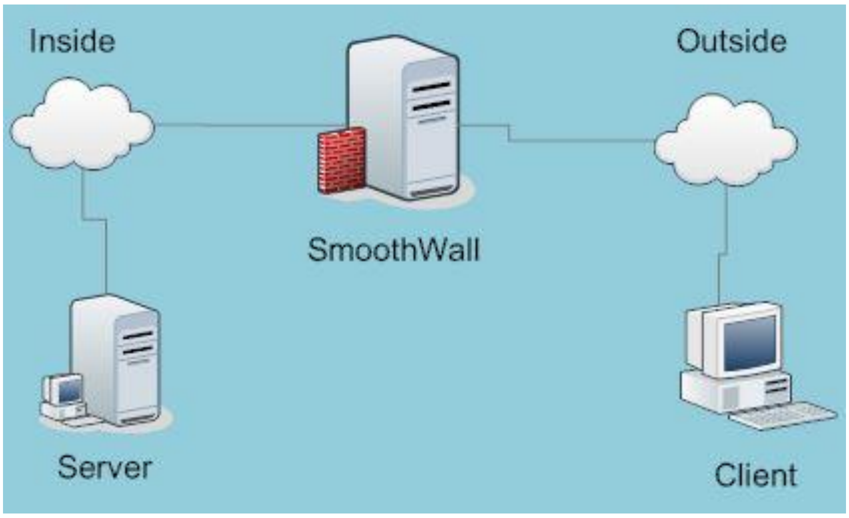


Figure 20 SmoothWall Topology

5.3.1 Network Performance with Default settings of SmoothWall

After I create the simple network, I simply test the traffic control between inside network and outside network. By default, traffic inside can go outside while the opposite traffic is denied. I can successfully ping from client to server but fail on pinging from server to client. Similarly to the TCP traffic, web request from client to server works but the other way around does not work well. For UDP traffic, TFTP server can receive the text file but fail to read the information in the text file.

Then I use the same measurement method to measure the network latency and bandwidth before I add any rules. Table 8 shows latency and packet lost rate when the Smoothwall is configured with default settings. By comparing Table 8 and Table 6, using SmoothWall will cause bigger latency when compared with IOS Firewall. But using SmoothWall is more reliable as the packet lost rate is smaller.

Table 8 Latency and Packet Lost in Default SmoothWall

Packet size (Bytes)	Number of Requests	Lost	Min (ms)	Max (ms)	Average (ms)
32	500	0 (0%)	0	0.4	0
35000	500	0.4 (0%)	13	14.4	13
65000	500	0 (0%)	23.8	31.4	24

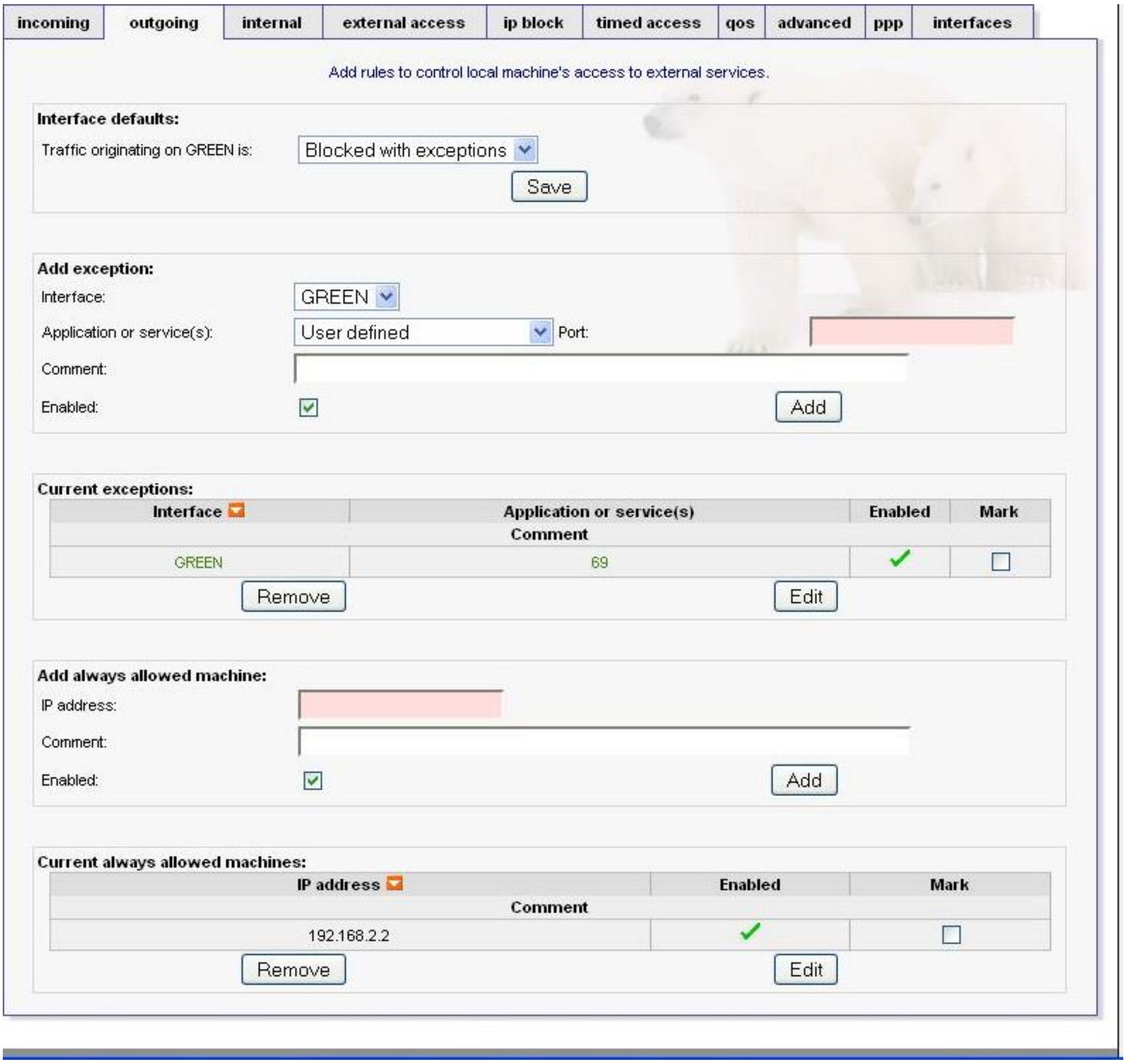
Table 9 shows the results of the TCP bandwidth. Comparing Table 9 and Table 7, TCP bandwidth in network with default SmoothWall decreases a little bit. In a word, it shows SmoothWall reduces the throughput.

Table 9 TCP Bandwidth in Default SmoothWall

Times	1	2	3	4	5	Average
Bandwidth(Kbits/sec)	83087	82938	82938	83064	83044	83014.2

5.3.2 Network Performance after adding rules

In this part, I test whether length of rules added in the SmoothWall will affect the network performance. Before adding those rules, I first try to configure some rules to test how the SmoothWall controls traffic. And I find tftp traffic does not work as I expected. Take the TFTP file transfer as an example, by default the client on inside network sends the text file and the server can receive it but cannot read its content. As a result, I individually allow the client computer to use UDP port 69 to successfully send the text file. Figure 21 shows traffic sent by the inside client via port 69 (for TFTP file transfer) is allowed to go outside. However, this still does not help the server read the received text.



Add rules to control local machine's access to external services.

Interface defaults:

Traffic originating on GREEN is: Blocked with exceptions

Add exception:

Interface: GREEN

Application or service(s): User defined Port:

Comment:

Enabled: ☑

Current exceptions:

Interface	Application or service(s)	Enabled	Mark
	Comment		
GREEN	69	✓	□

Add always allowed machine:

IP address:

Comment:

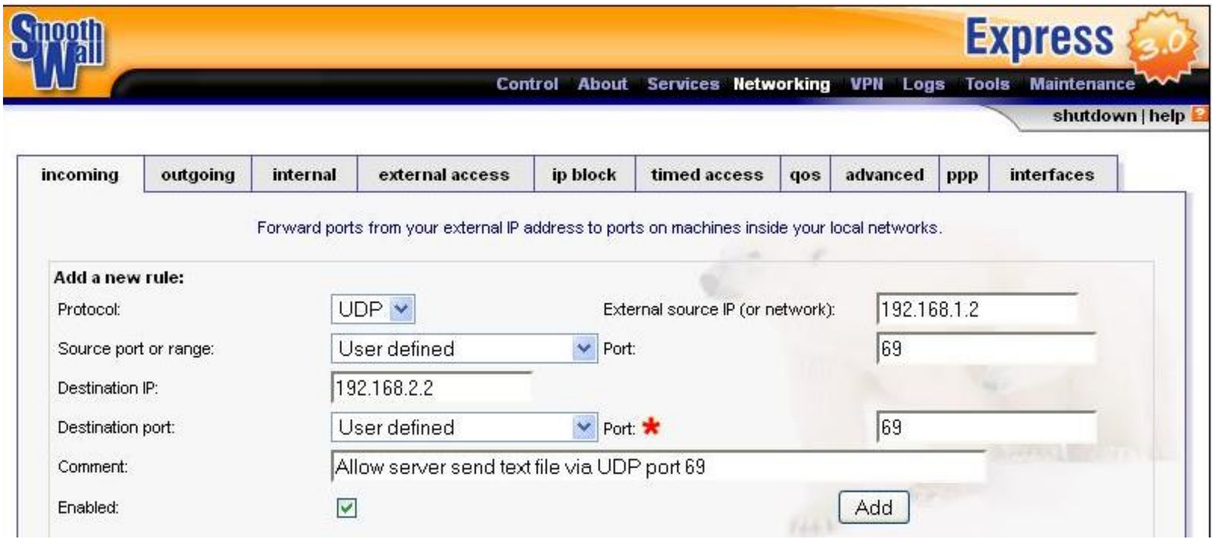
Enabled: ☑

Current always allowed machines:

IP address	Enabled	Mark
Comment		
192.168.2.2	✓	□

Figure 21 Allow Outing Traffic via Port 69 Service

In order to successfully transfer text file between two networks, I change the mode of the TFTP server and TFTP client. Server computer on the outside network is switched to TFTP client mode while the Client computer on inside network is changed to be a TFTP sever. Then I decide to let outside sever send TFTP file to inside client.


Control About Services Networking VPN Logs Tools Maintenance
shutdown | help
incoming outgoing internal external access ip block timed access qos advanced ppp interfaces

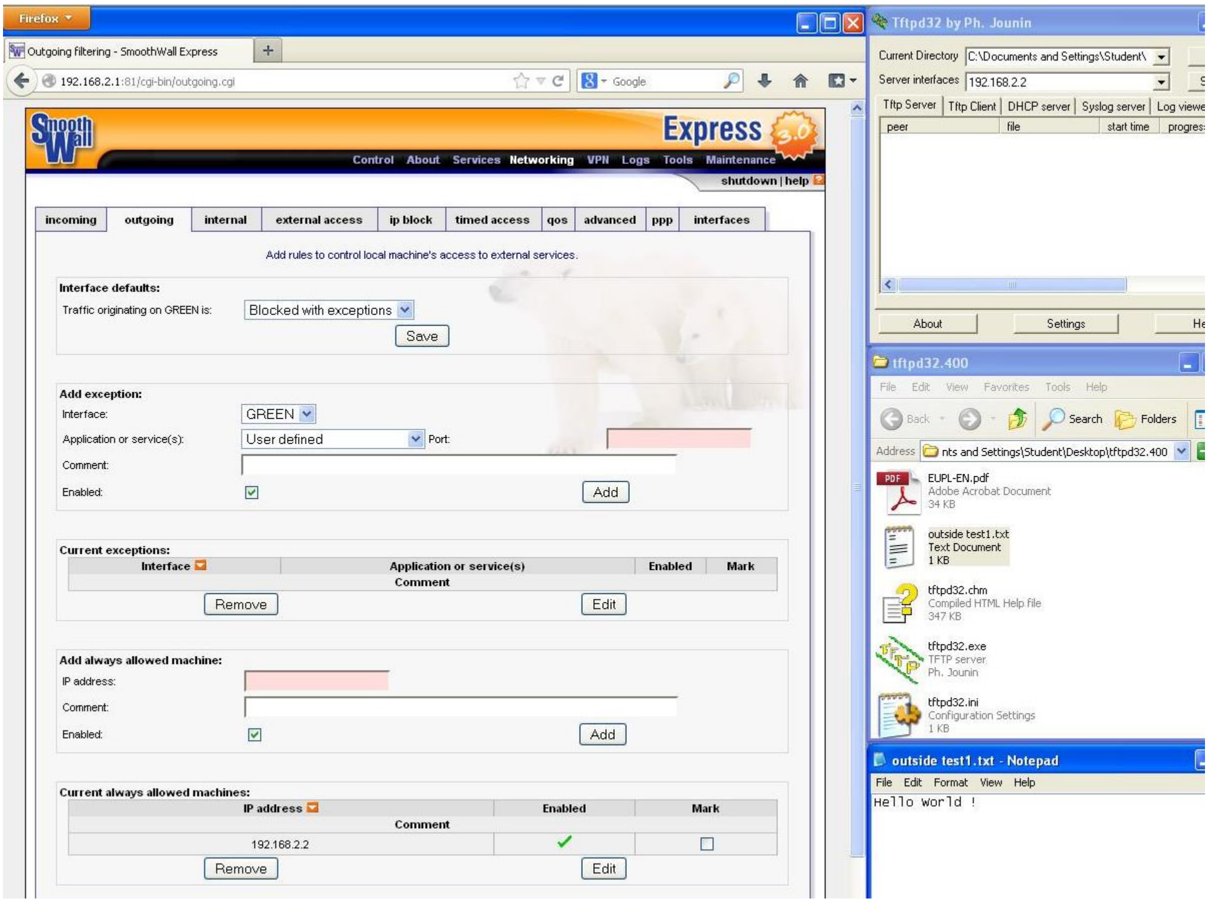
Forward ports from your external IP address to ports on machines inside your local networks.

Add a new rule:

Protocol:	UDP	External source IP (or network):	192.168.1.2
Source port or range:	User defined	Port:	69
Destination IP:	192.168.2.2		
Destination port:	User defined	Port: *	69
Comment:	Allow server send text file via UDP port 69		
Enabled:	☑		Add

Figure 22 Incoming Traffic Control

Figure 22 shows how I configure the incoming traffic to enable the server to send the file to the client. After I apply this rule, normally we would expect that TFTP sever (at inside network) can successfully receive the text and read its data. But it does not like that. The TFTP client can receive the text file but cannot read the data. The data is readable only I add a rule which allows client computer to access the outgoing service.


Firefox
Outgoing filtering - SmoothWall Express
192.168.2.1:81/cgi-bin/outgoing.cgi
Google
Control About Services Networking VPN Logs Tools Maintenance
shutdown | help
incoming outgoing internal external access ip block timed access qos advanced ppp interfaces

Add rules to control local machine's access to external services.

Interface defaults:

Traffic originating on GREEN is: Blocked with exceptions

Save

Add exception:

Interface:	GREEN		
Application or service(s):	User defined	Port:	
Comment:			
Enabled:	☑		Add

Current exceptions:

Interface	Application or service(s)	Enabled	Mark
	Comment		

Remove Edit

Add always allowed machine:

IP address:		
Comment:		
Enabled:	☑	Add

Current always allowed machines:

IP address	Enabled	Mark
Comment		
192.168.2.2	✓	□

Remove Edit

Tftpd32 by Ph. Jounin
Current Directory C:\Documents and Settings\Student\
Server interfaces 192.168.2.2
Tftp Server Tftp Client DHCP server Syslog server Log viewe
peer file start time progres
About Settings He
tftpd32.400
File Edit View Favorites Tools Help
Back Search Folders
Address nts and Settings\Student\Desktop\tftpd32.400
EUPL-EN.pdf Adobe Acrobat Document 34 KB
outside test1.txt Text Document 1 KB
tftpd32.chm Compiled HTML Help file 347 KB
tftpd32.exe TFTP server Ph. Jounin
tftpd32.ini Configuration Settings 1 KB
outside test1.txt - Notepad
File Edit Format View Help

```
Hello world !
```

Figure 23 Outing Traffic Control

Figure 23 shows I only add a rule allowing client computer (run as TFTP server) to access outside service. Without this rules, the text file can be received but cannot be readable. This is a little different thing when I test tftp traffic with SmoothWall. Compared with Cisco IOS Firewall, Smoothwall seems more friendly and easier to use. However, some functions are limited and at least the TFTP service is not that easy to understand and configure correctly.

To test whether length of rules affects SmoothWall performance, I add around 300 IP Block rules. Table 10 shows the network latency and packet lost rate. Compared Table 8 and Table 10, there is no much differences on latency between default SmoothWall and SmoothWall with rules. However, the packet lost rate is bigger when more rules are configured. In addition, the average maximum latency is also bigger when add more rules even those rules are useless.

Table 10 Network Latency and Packet Lost Rate in SmoothWall with 300 Rules

Packet size (Bytes)	Number of Requests	Lost	Min (ms)	Max (ms)	Average (ms)
32	500	0 (0%)	0	2.2	0
35000	500	1.2 (0%)	13	13.2	13
65000	500	0.2 (0%)	24	32.4	24

Table 11 shows that the TCP bandwidth is around 83073.8 when rules are configured and this value is a little higher according to Table 9 when the network is with default SmootWall. This result is not expected when more rules are added. My guess here is 300 rules are not enough to affect the bandwidth performance when the memory of SmoothWall is big.

Table 11 TCP Bandwidth in SmoothWall with 300 Rules

Times	1	2	3	4	5	Average
Bandwidth(Kbits/sec)	83001	83130	83001	83173	83064	83073.8

5.4 SmoothWall inside a Router

As SmoothWall is dedicated as firewall and normally companies may not directly connect it to the Internet. If they do like this, SmoothWall performance may be affected when it works for routing and firewall issues. Before I have implemented router intergrated with CBAC feature, now i will also implemnet a network integrated a SmoothWall inside a router. For the SmoothWall inside the router, the router focuses on routing and SmoothWall act as firewall. Then I would like to figure out pefromance differences between router with firewall features and Smoothwall inside a router.

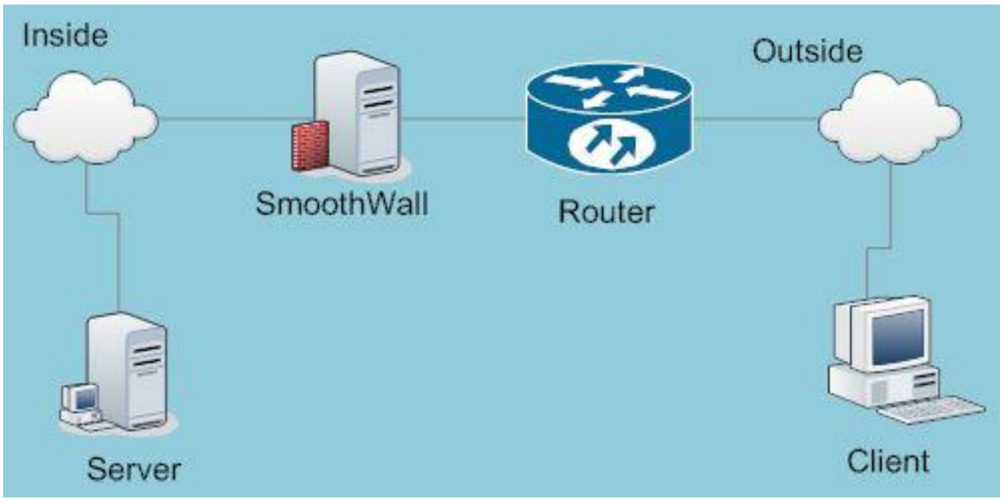
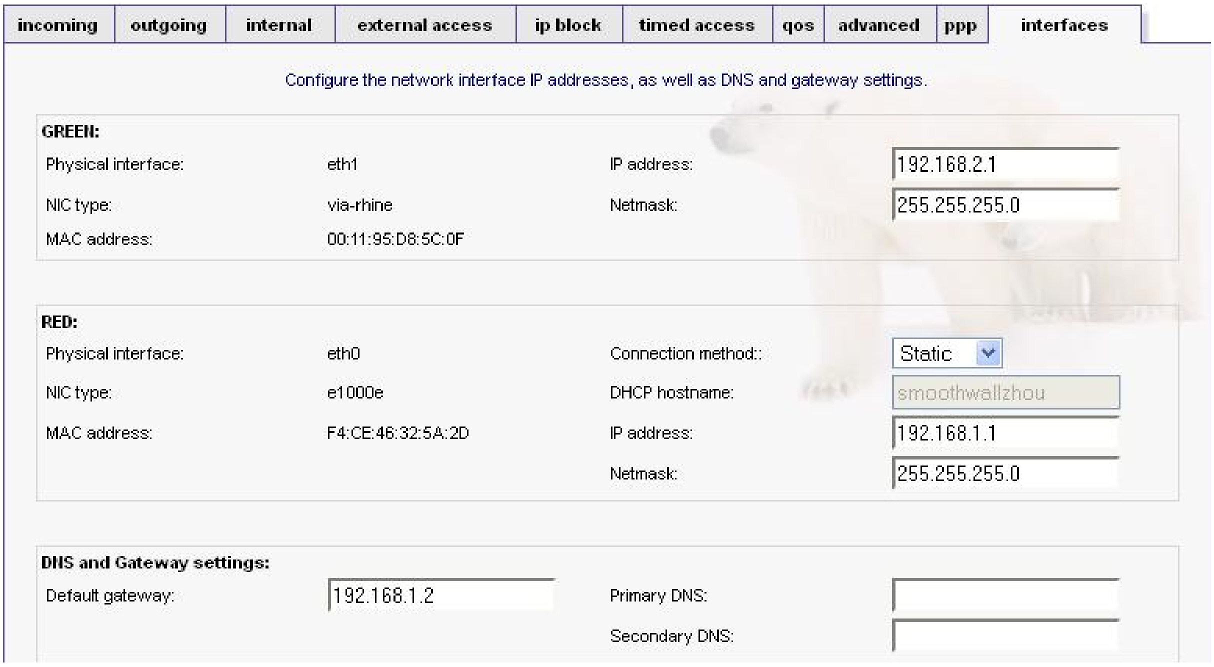


Figure 24 SmoothWall inside a Router

Figure 24 shows the netowork topology I use for the SmoothWall inside the router. The setting on Green side are not changed and Just connect the Red interface on the SmoothWall via port GE0/1 of the router and port GE0/2 of the router is connected to server compoter. IP address of GE0/1 is 192.168.1.2/24 while IP address of GE0/2 is 192.168.3.1/24. Server IP address is 192.168.3.2/24.



Configure the network interface IP addresses, as well as DNS and gateway settings.

GREEN:

Physical interface: eth1

NIC type: via-rhine

MAC address: 00:11:95:D8:5C:0F

IP address: 192.168.2.1

Netmask: 255.255.255.0

RED:

Physical interface: eth0

NIC type: e1000e

MAC address: F4:CE:46:32:5A:2D

Connection method:: Static

DHCP hostname: smoothwallzhou

IP address: 192.168.1.1

Netmask: 255.255.255.0

DNS and Gateway settings:

Default gateway: 192.168.1.2

Primary DNS:

Secondary DNS:

Figure 25 SmoothWall Interfaces Configuration

The configuration is quite simple, I just firstly configure interfaces of the router. Then SmoothWall Default Gateway is assigned with IP address of GE0/1 on router as this port connects the red interface of the SmoothWall. You can see the interaces I configure for the SmoothWall. To test the connectity, I ping from inside to outside server, send web request to the server. After all are successfull, I start to test the latency, the packet lost rate and the TCP bandwidth.

5.4.1 Network Performance for SmoothWall Inside Router

Comparing Table 10 and Table 12, there is no much differences in delay and packet lost rate. Except few small-size and medium-size packets have longer delay when after i add router before the SmoothWall connects to the outside network.

Table 12 Latency and Packet Lost for SmoothWall Inside Router

Packet size (Bytes)	Number of Requests	Lost	Min (ms)	Max (ms)	Average (ms)
32	500	0 (0%)	0	3	0
35000	500	0 (0%)	13	16.8	13
65000	500	0.2 (0%)	24	28.8	24

From Table 13 and Table 11 we can notice that bandwidth decrease a little bit when a router is added.

Table 13 TCP Bandwidth for SmoothWall Inside Router

Times	1	2	3	4	5	Average
Bandwidth(Kbits/sec)	82638	82681	82617	82403	82596	82587

5.5 Comparisons and Summaries

This section is mainly about making conclusion according to the outcomes of the practical work and giving some suggestions on how to choose a firewall solution. I should point out that, to test the security level of the firewall performance is not included in my final thesis.

Table 14 Latency and Packet Lost Rate in medium and bigger packet size

Packet size (35000 Bytes)			
Router with IOS Firewall	1.4 (0%)	9.8	7
SmoothWall with rules	1.2 (0%)	13.2	13
SmoothWall inside a router	0 (0%)	16.8	13
Packet size (65000 Bytes)			
Router with IOS Firewall	2.6 (0%)	54.2	13
SmoothWall with rules	0.2 (0%)	32.4	24
SmoothWall inside a router	0.2 (0%)	28.8	24

For convenience, I put all the data together and analyze them. Table 14 shows latency and packet lost in different tests. As there is no big difference when the packet size is small, I just use the medium and big packet size. I will mainly compare cases on router with IOS firewall and SmootWall inside a router. The former could be seen as integrated firewall while the later one is dedicated firewall with combination of router. Both have firewall and routing capabilities. Generally, delay will increase for all situations when packet size grows.

Basically, difference on latency and packet lost rate is not that obvious when packet size is small. When go to the medium packet size, router with CBAC firewall features has smaller delay than network with SmoothWall inside a router. However, we can see that using SmoothWall alone and using SmoothWall with router, packets are less likely to get lost when comparing with router with IOS firewall. For bigger packet size, SmoothWall inside a router still ensures lower packet lost rate but takes longer RTT. A little different thing is maximum RTT for few individual packets are lower.

Table 15 Comparison of the TCP bandwidth

Case	Bandwidth (Kbits/sec)	Percentage
Router without Firewall	84348	NO
Router with IOS Firewall	83832.4	99.39%
SmoothWall with rules	83073.8	NO
SmoothWall inside a router	82587	99.41%

Table 15 shows the TCP bandwidth among different tests. When a router is configured with CBAC firewall feature, the network bandwidth decreases a little bit. Similar to SmoothWall, when combine it with a router, TCP bandwidth degrades. Network bandwidth is higher when implementing IOS firewall than that with SmoothWall. When a router is configured with firewall feature, bandwidth is 99.39% of router without firewall. For SmoothWall inside a router, the bandwidth is 99.41% of network only uses SmoothWall. With same number of rules, it seems that router with CBAC firewall feature reduces more bandwidth but the overall bandwidth is still more than the case SmoothWall inside a router.

My aim of final thesis is to finding the answer on should a company choose a dedicated firewall like SmoothWall or an integrated firewall like Cisco Router configured with firewall features. To answer this question, I compare network performances on different cases and mainly focus on Cisco Router with CBAC and SmoothWall Express 3.0 combined with a router. I also test TCP, UDP and ICMP traffic by default mode before I add rules in both tests.

Using dedicated SmoothWall, network has bigger delay and lower TCP bandwidth. However, packets are less likely to get lost when SmoothWall is implemented. Another good point is that the software is free and all you need is a supportive hardware with Pentium class CPU or a better one. Installation of the SmoothWall is also quite easy because of the straightforward guide. For firewall configurations, except the tftp traffic control I test, SmootWall is generally easy to use. It is more friendly use especially for people who do not have enough network knowledge. This is because web-based graphic user interface is simple and people can understand easily. Cisco IOS firewall configuration needs person have prerequisites on the ACL rules and how to troubleshooting.

Functions of SmoothWall are somehow limited when I check the panel. For example, object-based port rules are not contained. This may be not that convenient when we want to control traffic on special group. For IP block rules, it can block outside hosts specified with source IP addresses or source networks. It does not contain access limitation on the specified destination hosts or a specified network. If we want to only outside host to access resources on one inside host and allow it to access other resources, this is not that convenient and maybe time-consuming. This is because we have to add rules in the incoming panel to add IP address of the each allowed hosts. When there are many of those addresses and without object rules, configuration will be not that easy.

We choose a firewall solution we can take the following things into consideration. For example, companies own needs, cost, network performance parameters, provided firewall features, whether easy to configure and supported hardware for the firewall. SmoothWall is suitable for a smaller company who wants to save cost and do not need too many advanced firewall features. It is also good for a company who do not have an IT professional and bandwidth is not a big issue. It is also a good choice for a company who needs smaller packet lost rate to get reliable data transmission.

For Cisco Router, it focuses on routing issues but the firewall features can be configured. If a company needs overall higher bandwidth and assumes that the packet lost as well as latency are a minor issues, the company is better to choose IOS firewall. Company needing more advanced firewall features is suggested to choose router. This will also require the company has professional IT experts to know how to configure those features and how to troubleshooting.

There are also other dedicated firewall products like Cisco ASA, it has more advanced firewall functions. I would like to test its performance in the future if I have time. Then more suggestions could be given for a company who wants to use other kinds dedicated firewall. For my future work, I am also interested in figuring out how complexity of rules configured affect the firewall performance. In addition, firewalls with DMZ are also interesting to explore in deep.

To make a conclusion, it is very interesting to do the projects and I learn many things not only firewall theory knowledge but also how to manage a project and time. How to plan and implement the project are also practical benefits in the whole process.

BIBLIOGRAPHY

[1] Yury Namestnikov (2012). IT Threat Evolution: Q3 2012. Retrieved from http://www.securelist.com/en/analysis/204792250/IT_Threat_Evolution_Q3_2012

[2] McAfee Labs. McAfee Threats Report: First Quarter 2012. Retrieved from http://www.mcafee.com/uk/resources/reports/rp-quarterly-threat-q1-2012.pdf

[3] Frederic Avolio. Firewalls and Internet Security, the Second Hundred (Internet) Years. Retrieved from http://www.cisco.com/web/about/ac123/ac147/ac174/ac200/about_cisco_ipj_archive_article09186a00800c85ae.html

[4] Chapter 22.2. Firewall characteristics. Retrieved from http://mercury.webster.edu/aleshunas/COSC%205130/Chapter-22.pdf

[5] ACLs
Cisco material, Accessing the WAN, chapter 5

[6] Types of ACLs
Cisco material, Network Seciurity, 2012, Chapter 4.1

[7] CBAC four functions
Cisco material, Network Seciurity, 2012, Chapter 4.3

[8] Figure 2 CBAC operation
Cisco material, Network Seciurity, 2012, Chapter 4.3.2.2

[9] Wikipedia, NAT definition. Retrieved from

http://en.wikipedia.org/wiki/Network_address_translation

[10] Wikipedia, Figure 3 NAT. Retrieved from
http://en.wikipedia.org/wiki/Network_address_translation

[11] Fredrik Thernelius. (May 2000). SIP, NAT, and Firewall.Chapter 8.4, NAT. Retrieved from
http://www.cs.columbia.edu/sip/drafts/Ther0005_SIP.pdf

[12] Meharouech Sourour, Bouhoula Adel, Abbes Tarek (December 2011). Network Security Alerts Management Architecture for Signature-Based Intrusions Detection Systems within a NAT Environment. Journal of Network and Systems Management. Volume 19, Issue 4, pp 472-495. Retrieved from
http://link.springer.com/article/10.1007%2Fs10922-010-9195-4

[13] Wikipedia, DMZ definition. Retrieved from
http://en.wikipedia.org/wiki/DMZ_(computing)

[14] DMZ access control rules
Cisco material, Network Security, Chapter 10.1.1.2

[15] Wikipedia, Figure 4 Single firewall. Retrieved from
http://upload.wikimedia.org/wikipedia/commons/6/6f/DMZ_network_diagram_1_firewall.svg

[16] Wikipedia, Figure 5 Dual firewall. Retrieved from
http://upload.wikimedia.org/wikipedia/commons/6/60/DMZ_network_diagram_2_firewall.svg

[17] RADIUS or TACACS+
Cisco material, Network Security 2012, Chapter 3.3.2.1

[18] Figure 6 CIA
Cisco material, Network Security, Chapter 7.1.4.3

[19] Packet Filtering

Cisco Material, Network Security 2012, Chapter 4.2.2.2

[20] Ray Blair, Arvind Durai. (May 21, 2009). Chapter 1: Types of Firewalls. Cisco Press. Retrieved from http://www.networkworld.com/subnets/cisco/060109-ch1-cisco-secure-firewalls.html?page=1

[21] Fragmentation Attacks (Nov 21st, 2008). Retrieved from http://www.bukisa.com/articles/7931_fragmentation-attacks

[22] Stateful packet filtering

Cisco material, Network Security 2012, Chapter 4.2.2.3

[23] Wikipedia, SYN flood. Retrieved from http://en.wikipedia.org/wiki/SYN_flood

[24] Figure 7 Stateful Firewall

Cisco material, Network Security 2012, Chapter 4.2.2.3

[25] Application firewall. Retrieved from http://www.windowsecurity.com/articles-tutorials/firewalls_and_VPN/Application_Layer_Filtering.html

[26] Deb Shinder (Published on 15 Jan. 2004 / Last Updated on 23 Jan. 2013). Application Layer Filtering (ALF): What is it and How does it Fit into your Security Plan? Retrieved from http://en.wikipedia.org/wiki/Distributed_firewall

[27] All about Firewalls. Retrieved from http://firewall-review.narod.ru/application_gateway.html

[28] Host-based firewall. IT Law Wiki. Retrieved from http://itlaw.wikia.com/wiki/Host-based_firewall

[29] Wikipedia, DNS spoofing. Retrieved from

http://en.wikipedia.org/wiki/DNS_spoofing

[30] Wikipedia, Firewall definition. Retrieved from http://en.wikipedia.org/wiki/Firewall_(computing)

[31] Tony Chen(08/20/2008). Access Control Lists (ACLs). Accessing the WAN-Chapter 5.. Retrieved from http://zh.scribd.com/doc/31339214/25/What-are-Dynamic-ACLs

[32] Karen Scarfone , Paul Hoffman (September 2009).Guidenlines on Firewalls and Firewall Policy. Special Publication 800-41 Revision 1. Retrieved from http://csrc.nist.gov/publications/nistpubs/800-41-Rev1/sp800-41-rev1.pdf

[33] Chpater 3 stateful firewall. Retrieved from http://www.pearsonhighered.com/samplechapter/0672327376.pdf

[34] Wikipedia, VPN. Retrieved from http://en.wikipedia.org/wiki/Virtual_private_network

[35] KENNETH INGHAM, STEPHANIE FORREST. A History and Survey of Network Firewalls. Retrieved from http://csis.bits-pilani.ac.in/faculty/murali/compnet-12/papers/History_and_Survey_of_Network_Firewalls.pdf

[36] Dirk (March 27th, 2009). Firewall Architecture. Retrieved from http://www.bluechaos.be/blogs/index.php/2009/03/firewall-architecture

[38] George Garza, Linda Richter (9/13/2010). Top 5 Layers of Information Security. Retrieved from http://www.brighthub.com/computing/enterprise-security/articles/86838.aspx#imgn_1

[39] Mick Bauer (Jan 01, 2003). An Introduction to FreeS/WAN, Part I. Retrieved from http://www.linuxjournal.com/article/6378?page=0,0

[40] VPN

CCNA Security Material, chapter 8

APPENDIX

Appendix 1- Configurations on CBAC

```
CBAC#show run
CBAC#show running-config
Building configuration...

Current configuration : 17472 bytes
!
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CBAC
!
boot-start-marker
boot-end-marker
!
!
!
no aaa new-model
!
!
no ipv6 cef
ip auth-proxy max-login-attempts 5
ip admission max-login-attempts 5
!
!
!
!
!
```

```
ip inspect name CBAC-FW tcp
ip inspect name CBAC-FW udp
ip inspect name CBAC-FW tftp
ip cef
!
multilink bundle-name authenticated
!
!
!
!
license udi pid CISCO2911/K9 sn FCZ153720T8
!
!
!
redundancy
!
!
!
!
!
!
!
!
!
!
!
!
!
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
```

```
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 ip address 192.168.1.1 255.255.255.0
 ip access-group 102 in
 duplex auto
 speed auto
!
interface GigabitEthernet0/2
 ip address 192.168.2.1 255.255.255.0
 ip access-group 101 in
 ip inspect CBAC-FW in
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
```

```
 no ip address
 shutdown
 clock rate 2000000
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
!
access-list 101 deny     tcp host 172.16.1.1 any
access-list 101 deny     tcp host 172.16.1.2 any
access-list 101 deny     tcp host 172.16.1.3 any
access-list 101 deny     tcp host 172.16.1.4 any
access-list 101 deny     tcp host 172.16.1.5 any
access-list 101 deny     tcp host 172.16.1.6 any
access-list 101 deny     tcp host 172.16.1.7 any
access-list 101 deny     tcp host 172.16.1.8 any
access-list 101 deny     tcp host 172.16.1.9 any
access-list 101 deny     tcp host 172.16.1.10 any
access-list 101 deny     tcp host 172.16.1.11 any
access-list 101 deny     tcp host 172.16.1.12 any
access-list 101 deny     tcp host 172.16.1.13 any
access-list 101 deny     tcp host 172.16.1.14 any
access-list 101 deny     tcp host 172.16.1.15 any
access-list 101 deny     tcp host 172.16.1.16 any
access-list 101 deny     tcp host 172.16.1.17 any
access-list 101 deny     tcp host 172.16.1.18 any
access-list 101 deny     tcp host 172.16.1.19 any
access-list 101 deny     tcp host 172.16.1.20 any
access-list 101 deny     tcp host 172.16.1.21 any
access-list 101 deny     tcp host 172.16.1.22 any
access-list 101 deny     tcp host 172.16.1.23 any
access-list 101 deny     tcp host 172.16.1.24 any
```

```
access-list 101 deny    tcp host 172.16.1.25 any
access-list 101 deny    tcp host 172.16.1.26 any
access-list 101 deny    tcp host 172.16.1.27 any
access-list 101 deny    tcp host 172.16.1.28 any
access-list 101 deny    tcp host 172.16.1.29 any
access-list 101 deny    tcp host 172.16.1.30 any
access-list 101 deny    tcp host 172.16.1.31 any
access-list 101 deny    tcp host 172.16.1.32 any
access-list 101 deny    tcp host 172.16.1.33 any
access-list 101 deny    tcp host 172.16.1.34 any
access-list 101 deny    tcp host 172.16.1.35 any
access-list 101 deny    tcp host 172.16.1.36 any
access-list 101 deny    tcp host 172.16.1.37 any
access-list 101 deny    tcp host 172.16.1.38 any
access-list 101 deny    tcp host 172.16.1.39 any
access-list 101 deny    tcp host 172.16.1.40 any
access-list 101 deny    tcp host 172.16.1.41 any
access-list 101 deny    tcp host 172.16.1.42 any
access-list 101 deny    tcp host 172.16.1.43 any
access-list 101 deny    tcp host 172.16.1.44 any
access-list 101 deny    tcp host 172.16.1.45 any
access-list 101 deny    tcp host 172.16.1.46 any
access-list 101 deny    tcp host 172.16.1.47 any
access-list 101 deny    tcp host 172.16.1.48 any
access-list 101 deny    tcp host 172.16.1.49 any
access-list 101 deny    tcp host 172.16.1.50 any
access-list 101 deny    tcp host 172.16.1.51 any
access-list 101 deny    tcp host 172.16.1.52 any
access-list 101 deny    tcp host 172.16.1.53 any
access-list 101 deny    tcp host 172.16.1.54 any
access-list 101 deny    tcp host 172.16.1.55 any
access-list 101 deny    tcp host 172.16.1.56 any
access-list 101 deny    tcp host 172.16.1.57 any
access-list 101 deny    tcp host 172.16.1.58 any
```

```
access-list 101 deny    tcp host 172.16.1.59 any
access-list 101 deny    tcp host 172.16.1.60 any
access-list 101 deny    tcp host 172.16.1.61 any
access-list 101 deny    tcp host 172.16.1.62 any
access-list 101 deny    tcp host 172.16.1.63 any
access-list 101 deny    tcp host 172.16.1.64 any
access-list 101 deny    tcp host 172.16.1.65 any
access-list 101 deny    tcp host 172.16.1.66 any
access-list 101 deny    tcp host 172.16.1.67 any
access-list 101 deny    tcp host 172.16.1.68 any
access-list 101 deny    tcp host 172.16.1.69 any
access-list 101 deny    tcp host 172.16.1.70 any
access-list 101 deny    tcp host 172.16.1.71 any
access-list 101 deny    tcp host 172.16.1.72 any
access-list 101 deny    tcp host 172.16.1.73 any
access-list 101 deny    tcp host 172.16.1.74 any
access-list 101 deny    tcp host 172.16.1.75 any
access-list 101 deny    tcp host 172.16.1.76 any
access-list 101 deny    tcp host 172.16.1.77 any
access-list 101 deny    tcp host 172.16.1.78 any
access-list 101 deny    tcp host 172.16.1.79 any
access-list 101 deny    tcp host 172.16.1.80 any
access-list 101 deny    tcp host 172.16.1.81 any
access-list 101 deny    tcp host 172.16.1.82 any
access-list 101 deny    tcp host 172.16.1.83 any
access-list 101 deny    tcp host 172.16.1.84 any
access-list 101 deny    tcp host 172.16.1.85 any
access-list 101 deny    tcp host 172.16.1.86 any
access-list 101 deny    tcp host 172.16.1.87 any
access-list 101 deny    tcp host 172.16.1.88 any
access-list 101 deny    tcp host 172.16.1.89 any
access-list 101 deny    tcp host 172.16.1.90 any
access-list 101 deny    tcp host 172.16.1.91 any
access-list 101 deny    tcp host 172.16.1.92 any
```

```
access-list 101 deny     tcp host 172.16.1.93 any
access-list 101 deny     tcp host 172.16.1.94 any
access-list 101 deny     tcp host 172.16.1.95 any
access-list 101 deny     tcp host 172.16.1.96 any
access-list 101 deny     tcp host 172.16.1.97 any
access-list 101 deny     tcp host 172.16.1.98 any
access-list 101 deny     tcp host 172.16.1.99 any
access-list 101 deny     tcp host 172.16.1.100 any
access-list 101 deny     tcp host 172.16.1.101 any
access-list 101 deny     tcp host 172.16.1.102 any
access-list 101 deny     tcp host 172.16.1.103 any
access-list 101 deny     tcp host 172.16.1.104 any
access-list 101 deny     tcp host 172.16.1.105 any
access-list 101 deny     tcp host 172.16.1.106 any
access-list 101 deny     tcp host 172.16.1.107 any
access-list 101 deny     tcp host 172.16.1.108 any
access-list 101 deny     tcp host 172.16.1.109 any
access-list 101 deny     tcp host 172.16.1.110 any
access-list 101 deny     tcp host 172.16.1.111 any
access-list 101 deny     tcp host 172.16.1.112 any
access-list 101 deny     tcp host 172.16.1.113 any
access-list 101 deny     tcp host 172.16.1.114 any
access-list 101 deny     tcp host 172.16.1.115 any
access-list 101 deny     tcp host 172.16.1.116 any
access-list 101 deny     tcp host 172.16.1.117 any
access-list 101 deny     tcp host 172.16.1.118 any
access-list 101 deny     tcp host 172.16.1.119 any
access-list 101 deny     tcp host 172.16.1.120 any
access-list 101 deny     tcp host 172.16.1.121 any
access-list 101 deny     tcp host 172.16.1.122 any
access-list 101 deny     tcp host 172.16.1.123 any
access-list 101 deny     tcp host 172.16.1.124 any
access-list 101 deny     tcp host 172.16.1.125 any
access-list 101 deny     tcp host 172.16.1.126 any
```

```
access-list 101 deny    tcp host 172.16.1.127 any
access-list 101 deny    tcp host 172.16.1.128 any
access-list 101 deny    tcp host 172.16.1.129 any
access-list 101 deny    tcp host 172.16.1.130 any
access-list 101 deny    tcp host 172.16.1.131 any
access-list 101 deny    tcp host 172.16.1.132 any
access-list 101 deny    tcp host 172.16.1.133 any
access-list 101 deny    tcp host 172.16.1.134 any
access-list 101 deny    tcp host 172.16.1.135 any
access-list 101 deny    tcp host 172.16.1.136 any
access-list 101 deny    tcp host 172.16.1.137 any
access-list 101 deny    tcp host 172.16.1.138 any
access-list 101 deny    tcp host 172.16.1.139 any
access-list 101 deny    tcp host 172.16.1.140 any
access-list 101 deny    tcp host 172.16.1.141 any
access-list 101 deny    tcp host 172.16.1.142 any
access-list 101 deny    tcp host 172.16.1.143 any
access-list 101 deny    tcp host 172.16.1.144 any
access-list 101 deny    tcp host 172.16.1.145 any
access-list 101 deny    tcp host 172.16.1.146 any
access-list 101 deny    tcp host 172.16.1.147 any
access-list 101 deny    tcp host 172.16.1.148 any
access-list 101 deny    tcp host 172.16.1.149 any
access-list 101 deny    tcp host 172.16.1.150 any
access-list 101 deny    tcp host 172.16.1.151 any
access-list 101 deny    tcp host 172.16.1.152 any
access-list 101 deny    tcp host 172.16.1.153 any
access-list 101 deny    tcp host 172.16.1.154 any
access-list 101 deny    tcp host 172.16.1.155 any
access-list 101 deny    tcp host 172.16.1.156 any
access-list 101 deny    tcp host 172.16.1.157 any
access-list 101 deny    tcp host 172.16.1.158 any
access-list 101 deny    tcp host 172.16.1.159 any
access-list 101 deny    tcp host 172.16.1.160 any
```

```
access-list 101 deny     tcp host 172.16.1.161 any
access-list 101 deny     tcp host 172.16.1.162 any
access-list 101 deny     tcp host 172.16.1.163 any
access-list 101 deny     tcp host 172.16.1.164 any
access-list 101 deny     tcp host 172.16.1.165 any
access-list 101 deny     tcp host 172.16.1.166 any
access-list 101 deny     tcp host 172.16.1.167 any
access-list 101 deny     tcp host 172.16.1.168 any
access-list 101 deny     tcp host 172.16.1.169 any
access-list 101 deny     tcp host 172.16.1.170 any
access-list 101 deny     tcp host 172.16.1.171 any
access-list 101 deny     tcp host 172.16.1.172 any
access-list 101 deny     tcp host 172.16.1.173 any
access-list 101 deny     tcp host 172.16.1.174 any
access-list 101 deny     tcp host 172.16.1.175 any
access-list 101 deny     tcp host 172.16.1.176 any
access-list 101 deny     tcp host 172.16.1.177 any
access-list 101 deny     tcp host 172.16.1.178 any
access-list 101 deny     tcp host 172.16.1.179 any
access-list 101 deny     tcp host 172.16.1.180 any
access-list 101 deny     tcp host 172.16.1.181 any
access-list 101 deny     tcp host 172.16.1.182 any
access-list 101 deny     tcp host 172.16.1.183 any
access-list 101 deny     tcp host 172.16.1.184 any
access-list 101 deny     tcp host 172.16.1.185 any
access-list 101 deny     tcp host 172.16.1.186 any
access-list 101 deny     tcp host 172.16.1.187 any
access-list 101 deny     tcp host 172.16.1.188 any
access-list 101 deny     tcp host 172.16.1.189 any
access-list 101 deny     tcp host 172.16.1.190 any
access-list 101 deny     tcp host 172.16.1.191 any
access-list 101 deny     tcp host 172.16.1.192 any
access-list 101 deny     tcp host 172.16.1.193 any
access-list 101 deny     tcp host 172.16.1.194 any
```

```
access-list 101 deny      tcp host 172.16.1.195 any
access-list 101 deny      tcp host 172.16.1.196 any
access-list 101 deny      tcp host 172.16.1.197 any
access-list 101 deny      tcp host 172.16.1.198 any
access-list 101 deny      tcp host 172.16.1.199 any
access-list 101 deny      tcp host 172.16.1.200 any
access-list 101 permit icmp 192.168.2.0 0.0.0.255 any
access-list 101 permit tcp 192.168.2.0 0.0.0.255 any
access-list 101 permit udp 192.168.2.0 0.0.0.255 any
access-list 102 deny      icmp host 10.10.10.1 any echo-reply
access-list 102 deny      icmp host 10.10.10.2 any echo-reply
access-list 102 deny      icmp host 10.10.10.3 any echo-reply
access-list 102 deny      icmp host 10.10.10.4 any echo-reply
access-list 102 deny      icmp host 10.10.10.5 any echo-reply
access-list 102 deny      icmp host 10.10.10.6 any echo-reply
access-list 102 deny      icmp host 10.10.10.7 any echo-reply
access-list 102 deny      icmp host 10.10.10.8 any echo-reply
access-list 102 deny      icmp host 10.10.10.9 any echo-reply
access-list 102 deny      icmp host 10.10.10.10 any echo-reply
access-list 102 deny      icmp host 10.10.10.11 any echo-reply
access-list 102 deny      icmp host 10.10.10.12 any echo-reply
access-list 102 deny      icmp host 10.10.10.13 any echo-reply
access-list 102 deny      icmp host 10.10.10.14 any echo-reply
access-list 102 deny      icmp host 10.10.10.15 any echo-reply
access-list 102 deny      icmp host 10.10.10.16 any echo-reply
access-list 102 deny      icmp host 10.10.10.17 any echo-reply
access-list 102 deny      icmp host 10.10.10.18 any echo-reply
access-list 102 deny      icmp host 10.10.10.19 any echo-reply
access-list 102 deny      icmp host 10.10.10.20 any echo-reply
access-list 102 deny      icmp host 10.10.10.21 any echo-reply
access-list 102 deny      icmp host 10.10.10.22 any echo-reply
access-list 102 deny      icmp host 10.10.10.23 any echo-reply
access-list 102 deny      icmp host 10.10.10.24 any echo-reply
access-list 102 deny      icmp host 10.10.10.25 any echo-reply
```

```
access-list 102 deny     icmp host 10.10.10.26 any echo-reply
access-list 102 deny     icmp host 10.10.10.27 any echo-reply
access-list 102 deny     icmp host 10.10.10.28 any echo-reply
access-list 102 deny     icmp host 10.10.10.29 any echo-reply
access-list 102 deny     icmp host 10.10.10.30 any echo-reply
access-list 102 deny     icmp host 10.10.10.31 any echo-reply
access-list 102 deny     icmp host 10.10.10.32 any echo-reply
access-list 102 deny     icmp host 10.10.10.33 any echo-reply
access-list 102 deny     icmp host 10.10.10.34 any echo-reply
access-list 102 deny     icmp host 10.10.10.35 any echo-reply
access-list 102 deny     icmp host 10.10.10.36 any echo-reply
access-list 102 deny     icmp host 10.10.10.37 any echo-reply
access-list 102 deny     icmp host 10.10.10.38 any echo-reply
access-list 102 deny     icmp host 10.10.10.39 any echo-reply
access-list 102 deny     icmp host 10.10.10.40 any echo-reply
access-list 102 deny     icmp host 10.10.10.41 any echo-reply
access-list 102 deny     icmp host 10.10.10.42 any echo-reply
access-list 102 deny     icmp host 10.10.10.43 any echo-reply
access-list 102 deny     icmp host 10.10.10.44 any echo-reply
access-list 102 deny     icmp host 10.10.10.45 any echo-reply
access-list 102 deny     icmp host 10.10.10.46 any echo-reply
access-list 102 deny     icmp host 10.10.10.47 any echo-reply
access-list 102 deny     icmp host 10.10.10.48 any echo-reply
access-list 102 deny     icmp host 10.10.10.49 any echo-reply
access-list 102 deny     icmp host 10.10.10.50 any echo-reply
access-list 102 deny     icmp host 10.10.10.51 any echo-reply
access-list 102 deny     icmp host 10.10.10.52 any echo-reply
access-list 102 deny     icmp host 10.10.10.53 any echo-reply
access-list 102 deny     icmp host 10.10.10.54 any echo-reply
access-list 102 deny     icmp host 10.10.10.55 any echo-reply
access-list 102 deny     icmp host 10.10.10.56 any echo-reply
access-list 102 deny     icmp host 10.10.10.57 any echo-reply
access-list 102 deny     icmp host 10.10.10.58 any echo-reply
access-list 102 deny     icmp host 10.10.10.59 any echo-reply
```

```
access-list 102 deny     icmp host 10.10.10.60 any echo-reply
access-list 102 deny     icmp host 10.10.10.61 any echo-reply
access-list 102 deny     icmp host 10.10.10.62 any echo-reply
access-list 102 deny     icmp host 10.10.10.63 any echo-reply
access-list 102 deny     icmp host 10.10.10.64 any echo-reply
access-list 102 deny     icmp host 10.10.10.65 any echo-reply
access-list 102 deny     icmp host 10.10.10.66 any echo-reply
access-list 102 deny     icmp host 10.10.10.67 any echo-reply
access-list 102 deny     icmp host 10.10.10.68 any echo-reply
access-list 102 deny     icmp host 10.10.10.69 any echo-reply
access-list 102 deny     icmp host 10.10.10.70 any echo-reply
access-list 102 deny     icmp host 10.10.10.71 any echo-reply
access-list 102 deny     icmp host 10.10.10.72 any echo-reply
access-list 102 deny     icmp host 10.10.10.73 any echo-reply
access-list 102 deny     icmp host 10.10.10.74 any echo-reply
access-list 102 deny     icmp host 10.10.10.75 any echo-reply
access-list 102 deny     icmp host 10.10.10.76 any echo-reply
access-list 102 deny     icmp host 10.10.10.77 any echo-reply
access-list 102 deny     icmp host 10.10.10.78 any echo-reply
access-list 102 deny     icmp host 10.10.10.79 any echo-reply
access-list 102 deny     icmp host 10.10.10.80 any echo-reply
access-list 102 deny     icmp host 10.10.10.81 any echo-reply
access-list 102 deny     icmp host 10.10.10.82 any echo-reply
access-list 102 deny     icmp host 10.10.10.83 any echo-reply
access-list 102 deny     icmp host 10.10.10.84 any echo-reply
access-list 102 deny     icmp host 10.10.10.85 any echo-reply
access-list 102 deny     icmp host 10.10.10.86 any echo-reply
access-list 102 deny     icmp host 10.10.10.87 any echo-reply
access-list 102 deny     icmp host 10.10.10.88 any echo-reply
access-list 102 deny     icmp host 10.10.10.89 any echo-reply
access-list 102 deny     icmp host 10.10.10.90 any echo-reply
access-list 102 deny     icmp host 10.10.10.91 any echo-reply
access-list 102 deny     icmp host 10.10.10.92 any echo-reply
access-list 102 deny     icmp host 10.10.10.93 any echo-reply
```

```
access-list 102 deny     icmp host 10.10.10.94 any echo-reply
access-list 102 deny     icmp host 10.10.10.95 any echo-reply
access-list 102 deny     icmp host 10.10.10.96 any echo-reply
access-list 102 deny     icmp host 10.10.10.97 any echo-reply
access-list 102 deny     icmp host 10.10.10.98 any echo-reply
access-list 102 deny     icmp host 10.10.10.99 any echo-reply
access-list 102 deny     icmp host 10.10.10.100 any echo-reply
access-list 102 permit icmp 192.168.1.0 0.0.0.255 any echo-reply
!
!
!
control-plane
!
!
!
line con 0
line aux 0
line 2
 no activation-character
 no exec
 transport preferred none
 transport input all
 transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
 stopbits 1
line vty 0 4
 login
 transport input all
!
scheduler allocate 20000 1000
!
end

CBAC#
```