Mainframe Linux 를 이용한 가상서버환경의 보안성 강화를 위한 Virtual Firewall 의 성능 평가 및 비교

김태균*, 황진옥**, 민성기**
*고려대학교 컴퓨터과학기술대학원
**고려대학교 컴퓨터학과
e-mail : tachyon@korea.ac.kr

Performance Evaluation and Comparison of Virtual Firewall on Virtual Server Farm using Mainframe Linux for Security Enhancement

Tae-Gyoon Kim*, Jin-ok Hwang**, Sung-gi Min**
*Graduate Schools of Computer Science & Technology, Korea University
**Dept. of Computer Science & Engineering, Korea University

요 약

Mainframe Linux 를 이용한 가상서버환경은 점차 기존의 분산서버환경과 유사한 방향으로 진화하고 있다. 이와 같이 구현된 가상의 서버환경에서 보안성의 확보는 필수 사항으로 요구되고 있으며 이와 관련한 여러 방안 중 기존 분산서버환경과 유사한 Virtual Firewall 의 필요성이 대두되고 있으며, 이에 본 논문은 분산서버환경의 Firewall 적용 및 Mainframe Linux 의 가상서버환경 구현에 관한 연구활동을 바탕으로, Firewall 로서의 기능적 요구사항과 Mainframe Linux 환경의 Resource 관리 관련요구사항을 기준으로 Debian 계열의 상용 Virtual Firewall 의 적용과 성능에 대한 평가를 수행하였다. 추가로 기존 분산서버환경의 Appliance 형태의 Gigabit Firewall 의 성능평가 결과를 비교하여 보았다. 기능 및 성능적인 면에서 기존 분산서버환경의 Firewall 제품들과 유사한 수준과, Resource 관리의 측면에서 타 서버들과 공존하는 가상서버환경에 큰 영향을 미치지 않는 결과를 보여주었다.

1. 서론

최근 Linux 의 활용범위가 Mainframe 까지 확대되고 있다. 이는 Mainframe 의 높은 가용성 및 신뢰성에 Linux 의 풍부한 Open Source 를 활용함으로써 유연하고 강력한 시스템 구현을 위함이다.

Mainframe 상의 Linux 구현의 방법으로 전체 Resource 을 하나의 운영체제가 모두 사용하는 방법으로 유휴 Resource 발생의 단점을 가지는 Native 방식, 전자보다 많은 운영체제를 사용할 수 있으나 여전히 제한적인 LPAR (Logical Partition)방식, 그리고 다수의 가상서버환경을 구현하고 실시간으로 CPU, I/O 등 Resource 을 동적으로 할당하는 기능을 사용할 수 있는 Virtual Machine 방식이 있다[10]. 이러한 Mainframe 상의 가상서버환경은 기존의 분산서버환경[1]과 유사한 형태로 발전하고 있으며 Web-WAS-DB 의 3-tier 형태의 구성을 보이고 있다. 따라서 보안성 강화를 위한 방안이 연구되고 있다. 이를 위한 한가지 방안으로 Virtual Firewall 적용과 그 기능 및 성능에 대하여 검토가 필요할 것이다.

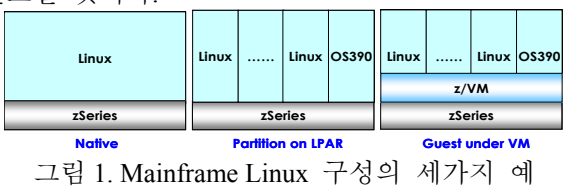

그림 1. Mainframe Linux 구성의 세가지 예

본 논문의 2 장에서는 관련연구 활동을 알아보고, 3 장에서는 가상서버환경의 Virtual Firewall 테스트 기준에 대한 내용을, 4 장에서는 성능 테스트 수행 내용과 그 결과를 알아보고, 5 장에서 결론을 맺는다.

2. 관련 연구

2.1. Firewall 의 구현 방식

Firewall 은 일반적으로 Bastion Host, Screened Host, Dual-Homed Gateway 등으로 구성되며, 기능 수행 계층(Layer)을 기준으로 발전하였다[4][6].

Packet Filtering 방식은 주로 OSI 7 Layer 의 Network Layer 에서 Routing 과 IP Packet Filtering 을 수행하며 빠른 속도와 간단한 구성 그리고 저렴한 비용의 장점이 있는 반면, Traffic 정보가 적으며 Filtering Rule 이 복잡하고 난해한 단점을 가진다[3].

Application Proxy(Gateway) 방식은 Application Layer 에서 응용 프로그램의 연결/차단을 수행하며 Traffic 제어 선택사항이 다양하고 Traffic 정보가 풍부한 장점을 가지나, 상대적으로 속도가 느리며 Traffic 이 많은 지점에 부적합하며 개별 Application 마다 Gateway 가 필요한 단점을 가진다[2][5].

Circuit Level Proxy 방식은 Application Proxy 와 달리 Session Layer 에서 동작하며 SOCKS V5 가 대표적인 예로 대부분의 Protocol 을 자동으로 지원해 주는 반면, Application Protocol 을 해석하지 않으므로 Application-Specific Access Control 이나 Logging/Auditing 기능을 제공하지 못한다[3].

이밖에도 Dynamic Packet Filtering 의 형태로 Kernel 단에 Stateful Packet Inspector 모듈을 장착하여 필요한 정보가 축적될 때까지 기다린 후 Filtering 을 수행하는 Stateful Packet Inspection 방식은 Application Proxy 를 모방하여 Packet Filtering 을 적용하되 Application Processing 보다 훨씬 적은 부하로 Application Data 를 해석하여 Filtering 하는 솔루션으로 Application Proxy 보다 훨씬 나은 성능을 가지면서도 이에 상당하는 Protection Capability 를 제공한다[8].

최근에는 전통적인 Packet Filtering 과 Application Proxy 그리고 Stateful Inspection 의 장점을 혼합한 Multi-Layer Inspection 방식이 제시되었다. Multi-Layer Inspection 은 Protocol Agent 라는 component 를 이용하여, Intruder 의 정상적인 SSH 연결을 통한 Malicious Code/Packet 의 전송을 차단하기 위하여 SSH 연결의 첫 번째 Packet 을 검사하여 SSH 표준을 따르는지 확인하는 것과 같은 Protocol 표준 강화 기능, Content Inspection System 을 이용한 Redirection Traffic 검사 기능, VoIP 와 같은 복잡한 Protocol 의 전송을 가능케 하기 위한 Application Data 의 검사와 수정 등의 기능 등 보다 유연한 기능을 제공한다[11].

2.2. 가상서버환경의 Firewall

Linux Embedded Appliance Firewall (LEAF)은 Linux Router Project (LRP)에서 파생된 솔루션으로 최소의 Resource(8MB 메모리, 3~4 장의 플로피 디스크)를 가지고 Routing 과 Firewall 기능을 얻기 위한 활동으로, 주로 구형 PC 의 재활용이 목적이다. 주로 i386 의 아키텍처에 porting 되던 이 솔루션을 Mainframe Linux 의 가상서버환경에 접목하기 위한 연구가 활발하다. i386 의 것과 달리 Boot Loader 및 DASD driver 모듈 등 약간의 커스터마이징과 첨가를 이용하여 가상서버환경에 porting 을 할 수 있다. 가상서버환경에 적용하기 위한 필요 Resource 는 2.57MB 의 kernel file, 4MB 의 initial root device, 16MB 의 storage required for IPL 정도를 요구한다[10]. 그러나 Debian GNU/Linux 를 기반으로 하는 이 가상 Firewall 은 앞의 Firewall 의 여러 방식 중 Packet Filtering 방식에 가깝다. 따라서 고속의 성능은 보여줄 수 있으나 서버환경의 복잡도가 증가될수록 성능저하와 관리의 불편을 가중할 수 있는 단점을 가질 수 있다. 본 논문에서 테스트할 Multi-Layer Inspection 의 Hybrid 방식이라 LEAF 보다 다양하고 우수한 기능과 성능을 가질 것이다.

2.3. Firewall 의 성능 평가 요소

다수의 Firewall 솔루션과 제품들에 대한 객관적인 평가를 위하여 NSS Group 은 그 기준을 제시하고 여러 제품에 대한 평가를 수행한 바 있다. Firewall Group 테스트(Edition1)에서 NSS Group 은 성능 평가를 위해 그 기준을 크게 3 가지로 나누었다[9].

첫 번째는 Basic Competency 항목으로, 일반적인 Firewall Scanning Tools 를 이용하여, ‘Probe the firewall from the Internet’, ‘Probe the protected network from the Internet’, ‘Probe the DMZ from the Internet’, ‘Probe the firewall from the protected network’, ‘Probe the Internet from the protected network (test security policy)’, ‘Probe the DMZ (if available) from the protected network’, ‘Probe the firewall from the DMZ(if available)’, ‘Probe the protected network from the DMZ(if available)’의 항목을 설정하였다.

두 번째는 성능 평가를 위한 항목으로, Basic Throughput (Throughput, Frame Loss, Latency), Maximum Connection Rate, Goodput, Goodput with UDP Load, Goodput with UDP Load and DoS Handling 의 항목을 설정하였다.

마지막으로 Client 의 요청에 의한 대량의 HTTP Content 전송 정도를 보는 Goodput 의 항목을 추가하였다. NSS Group 는 이상의 기준으로 여러 제품에 대한 평가를 수행하였다.

국내에서는 정보보호 솔루션의 평가를 위해 ‘정보통신망 침입차단 시스템 평가기준’[7]을 설정하였으며 Common Criteria 에 대한 기준도 마련하여 객관적인 평가 기준을 마련하고 있다.

3. 본론

본 연구에서는 Mainframe Linux 를 이용한 가상서버환경의 보안성을 향상시키고자 기존의 분산서버환경과 유사한 방법을 강구하던 중, Virtual Firewall 의 적용을 고려하게 되었으며 이의 적절한 구성을 위하여 성능에 대한 연구를 수행하게 되었다. 이에 따른 성능 평가를 위한 테스트 지표로써 NSS Group 의 ‘Firewall

Group Test -Edition1'[9]과 국내 '정보통신망 침입차단 시스템 평가기준'[7]을 참고하게 되었다. 평가기준은 기능, 성능, 안정성의 세 가지의 범위와 가상서버환경의 안정적 서비스를 고려한 Resource 사용량 측정으로 나누었다.

기능 테스트는 Firewall 의 침입차단방식과 환경 및 기능적인 사항을 알아보기 위하여 표 1 과 같이 방식, 지원사항과 DoS 를 고려한 항목을 선정하였다[6].

표 1. 기능 테스트

항목	내용
Firewall 방식	Proxy or Packet 방식, Inspection, Gateway
환경지원	Platform, Network Protocol/ Bandwidth, 동시접속 가능세션수
기능지원	VPN, ACL, Port Scanning · ICMP 공격 탐지/차단, Anti-Spoofing, Packet Fragment Attack, Trin00/SYN flooding 차단, Anti-Virus 기능
DoS Stress 테스트	SYN flooding, UDP flooding, Port Scan, Land Attack, Ping of death

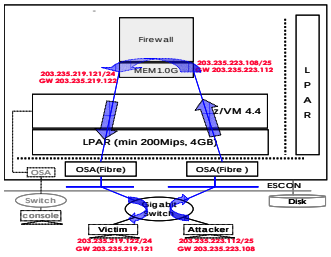

그림 2. 기능 테스트 수행 구성도

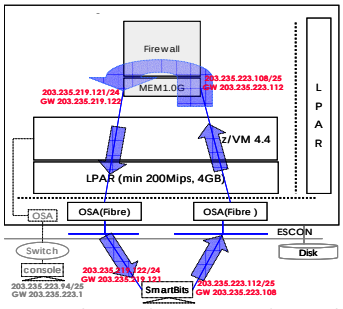

그림 3. 성능 테스트 수행 구성도

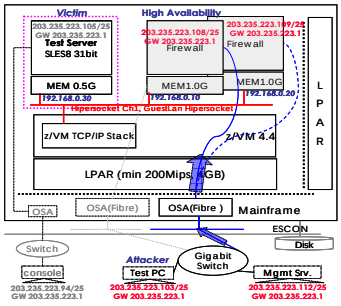

그림 4. 안정성 테스트 수행 구성도

성능 테스트는 Firewall 로 인한 전체 서비스에 대한 영향도를 알아보기 위한 Network 부하 성능 테스트로 Throughput, Latency 를 측정한다.

안정성 테스트는 장애시 대처능력을 검증하기 위한 항목으로 FW 재부팅시간을 측정하였다.

마지막으로 Mainframe Linux 를 이용한 가상서버환경의 관점에서 Virtual Firewall 의 CPU 사용량과 가상서버환경의 전체 CPU 사용량을 추가하였다. 메모리의 경우 Buffer, Cash 등의 이유로 항상 일정 수준이상의 사용률을 보이기에 항목에서 제외하였다. 이러한 고려는 가상서버환경에서 Firewall 의 기능 수행간 타 서버들에 미치는 영향도를 알아보기 위함이다.

4. 테스트 결과 및 분석

테스트 환경으로 Windows 또는 Linux 를 운영체제로 사용한 Pentium Ⅲ 700MHz (512MB Memory, 20G HDD)급의 PC 를 사용하였으며, Virtual Machine 은 z/VM 4.4, 1CP Shared(약 400MIPS), 4GB Memory Shared 환경으로 Virtual Firewall 을 위하여 1CP Shared, 128MB Memory, 4.6GB Disk 의 Resource 를 할당하였다. 그리고 Industry Standard Network Performance Analysis System 으로, 일반적인 Packet 을 발생하고 그 Packet 을 Count 하며, Trigger Packet 을 생성하여 전송하고 Count 할 수 있는, 'Spirent Communications SmartBits SMB-600'을 테스트 장비로 활용하였다[9].

여러 공격 Tool 을 이용한 실제 테스트 수행 결과, Firewall 본연의 기능을 충실히 수행하였다. 동시접속 가능 세션수는 1GB 의 main Storage 를 확보할 시 최대 500,000 세션으로 조사되었다.

표 2. 기능 테스트

항목	내용
Firewall 구조/방식	Multi-Layer Inspection
지원 Platform	Intel, Mainframe
지원 Network Protocol	IP, ICMP, TCP, UDP
지원 Network Bandwidth	Gigabit Interface
동시접속 가능세션수	최대 500,000 세션
SYN Flooding (by SynFlooder)	탐지
UDP Flooding (by UDP Flooder)	탐지
Port Scan (by SuperScan)	탐지
Land Attack (by SynFlooder)	탐지/차단
Ping of death (by HAKTEK)	탐지

Throughput 및 Latency 를 통한 가상서버환경에 대한 영향도를 살펴본 바 기존 분산환경에서 사용 중인 Fast Ethernet 용 Appliance Firewall 의 성능을 상회하는 수준이었다.

표 3. 성능 테스트(Route mode, 보안정책: All allow)

Frame Size(UDP)	Throughput(Mbps)	Latency(㎲)
64 Byte	47.86	463
128 Byte	83.63	520
256 Byte	155.28	660
512 Byte	310.21	593
1024 Byte	572.59	874
1518 Byte	619.47	684

*Throughput=((FrameSize+Preamble:7Byte+StartBit:1Byte)×8bit+InterFrameGap:96bit)×(SmartBit pps result)

장애를 감안한 안정성 테스트의 경우, Intel Platform 의 경우 High Availability 기능을 지원하나 2004 년 2Q 현재 Mainframe Linux 용 버전이 HA 기능을 완벽히 지원하지 못하여, 장애시 대처능력으로 재부팅 시간을

측정하였다. 평균 약 40 초 정도의 재부팅 시간이 소요되었으며 이는 Severity 가 높지 않는 경우에 적용이 가능한 것으로 판단된다. 차후 차기 버전의 안정화 버전이 발표된다면 HA 기능도 무난히 구현 가능할 것이다.

표 4. Resource 사용량 평가(성능 테스트 수행간)

구분(테스트)		성능테스트 간 Max
CPU 사용량 -Max	Virtual Firewall	75.91MIPS
	LPAR(약 400MIPS 이상)	139.12MIPS
	BOX(전체 3874MIPS)	220.82MIPS

성능 테스트 수행 중 가상서버환경에서의 Resource 사용량 변화 추이를 살펴본 바, Virtual Firewall 의 CPU 사용량은 최고 75MIPS 로 Mainframe 전체(3874MIPS)의 1.96%, Peak 시 Logical Partition 환경(LPAR) 할당되는 양(139.12MIPS)에 대해서는 약 54.57%로, 전체 가상서버환경의 Resource 활용에도 무리가 없었다. 이는 실시간으로 동적인 Resource 할당과 제어를 하는 Dynamic Resource Sharing 기능의 효과일 것이다.

이 기능을 고려한다면 High Availability 기능이 꼭 필요하지 않을 것이다.

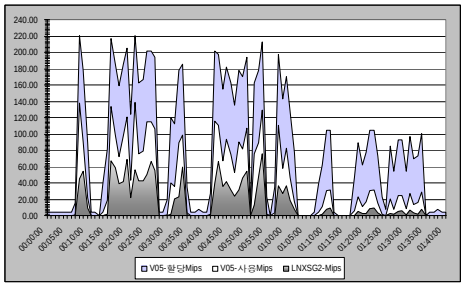

그림 5. CPU 사용량 추이 (성능 테스트 수행간)

표 5. Appliance Gigabit Firewall 테스트 결과 참고 (국내 FreeBSD 기반 제품)

항목		내용
동시접속 가능세션수		최대 200,000 세션
Packet Size - Throughput/Latency	64 Byte-	356 Mbps/26.37 ㎲
	256 Byte	714 Mbps/27.20 ㎲
	512 Byte	1366 Mbps/42.53 ㎲
초당/최대 처리 세션수		17914/999045
SYN Flooding, UDP Flooding, Ping of death		탐지/차단
Port Scan, Land Attack		탐지
1st FW fail 시 2nd FW 로 자동기능전이		5~15 초
1st FW fail 시 Auto 세션 Fail-over		Yes
1st FW 복원시 1st FW 로 자동기능전이		8~15 초
1st FW 복원시 Auto 세션 Fail-over		Yes

5. 결론

본 논문에서 테스트한 Debian 계열의 Virtual Firewall 은 기존 분산서버환경에 적용되는 Fast Ethernet Appliance 제품과 비슷한 성능을 보여 주었으며 가상서버환경의 서비스에도 큰 영향을 미치지 않았다. 단, '04 년 2Q 현재 Test 환경에 이용되는 Virtual Machine(z/VM 4.4)의 버전에 대하여 Virtual Firewall 의 대응 버전이 개발 상태이라 Hot-Standby 의 High Availability 의 기능이 완벽히 지원되지 않았으며 이는 향후 안정화 버전에서 지원될 것이다. 그러나 Mainframe 의 Dynamic Resource 기능에 따른 탄력적인 자원할당기능을 생각한다면 현재 상태에서도 그 적용이 가능하다고 본다. 따라서 Virtual Firewall 으로 가상서버환경의 보안성을 강화시켜 주는 한 방안으로써 그 활용 가능성을 보여주었다.

향후 Mainframe Linux 의 가상서버환경의 진화에 대한 보다 다양하고 지속적인 접근과 보안성 강화의 연구가 필요할 것이며, 또한 가상서버환경의 다양한 구성에 따른 보다 최적화된 Virtual Firewall 의 적용과 그 기준에 대한 연구(HA 포함)가 계속되어야 할 것이다. 나아가 Virtual IPS 등 새로운 솔루션에 대한 연구와, 통합 보안관제 환경과의 연계에 대한 연구도 필요할 것이다.

참고문헌

[1] 유진근, 박근수, "기업 내부 보안 시스템에서 보안성 개선 방안", 한국정보과학회 추계학술대회 논문집, 2003

[2] 김선정, 나현식, "시스템 보안을 위한 프락시 애플리케이션 방화벽 시스템 구축", 한국정보처리학회 추계학술발표논문집 제 8 권 제 2 호, 2001

[3] 최준호, 김판구, "네트워크상에서 바이러스 차단을 위한 방화벽 시스템의 설계 및 구현", 한국정보처리학회 논문지 C 제 8-C 권 제 4 호, 2001.8

[4] 송병욱, 김홍철, 박인성, 김상욱, "네트워크 트래픽 상태 기반의 방화벽 시스템", 한국정보처리학회 춘계 학술발표논문집 제 8 권 제 1 호, 2001

[5] 기우서, 김병철, 오재철, "학교전산망 보호를 위한 스크린드 호스트 게이트웨이 방화벽 구축에 관한 연구", 한국정보처리학회 추계 학술발표논문집 제 7 권 제 2 호, 2000

[6] 이용준, 김봉한, 박천용, 오창석, 이재광, "전산망 보호를 위한 혼합형 방화벽 시스템 구현", 한국정보처리학회 논문집 제 5 권 제 6 호, 1998

[7] 정보통신부, "정보통신망 침입차단시스템 평가기준 개정", 한국정보보호진흥원, 2000

[8] Robert Zalenski, "Firewall Technologies", IEEE Potentials, 2002

[9] http://www.nss.co.uk, "Firewall Group 테스트 (Edition1)", NSS Group, 2002.12

[10] http://publib-b.boulder.ibm.com/redbooks.nsf/portals/S390Redbooks "Linux on IBM eServer zSeries and S/390: Porting LEAF to Linux on zSeries"외 IBM Redbooks, IBM, 2002~2004

[11] http://www.stonesoft.com/products/StoneGate/Firewall/, "Multi-Layer Inspection", "StoneGate 2.2 Administrator's Reference", StoneSoft, 2002~2003

[12] http://leaf.sourceforge.net, "Linux Embedded Appliance Firewall Project"