

# 웹서버의 고가용성 보장 기술

이철훈\*, 이일신\*\*

(\*충남대 컴퓨터공학과, \*\*디스크뱅크(주))

## 1. 서론

1980년대 후반 WWW(World Wide Web)의 출현에 의해 인터넷은 일반 사용자들이 쉽게 접근하여 사용할 수 있는 대중적이 매체로 발전하였으며, 다양한 인터넷 응용프로그램들이 웹 기반으로 통합화되는 한편 통신 기기, 가전/멀티미디어 기기, 자동화/제조 기기 및 운송 기기 등의 다양한 응용 분야에서 인터넷을 통한 웹 서비스가 제공되기 시작하면서 이를 서비스하기 위해 인터넷에 접속된 웹 서버의 수 또한 폭발적으로 증가하여 왔다. 앞으로도 지속적인 인터넷의 발전과 더불어 웹 서버의 수 및 이들을 이용한 기업의 수는 더욱 증가할 것이 분명하며, 데이터 웨어하우징, 온라인 트랜잭션 처리, 의사 결정 지원 시스템 등과 같은 웹 상의 핵심 비즈니스 오퍼레이션을 운영하고 관리하는 웹서버의 고가용성(High Availability)에 이들 기업의 성패가 달려 있다고 할 수 있다.

여기에서 가용성(Availability)이란 사용자(client)에게 주어진 서비스를 제공하는 서버의 능력에 대한 척도로서, 전체 경과시간(즉, 가동시간 + 휴지시간)에 대한 가동시간의 비로 정의된다. 휴지시간에는 계획된 휴지시간(planned downtime)과 비계획 휴지시간(unplanned downtime)으로 구분되며, 계획된 휴지시간은 백업, 유지 보수, 업그레이드 등의 미리 계획된 작업들에 의한 것이며, 비계획 휴지시간의 원인은 서버 가동을 중지시키는 사건들로서 하드웨어나 소프트웨어 고장, 오퍼레이터의 실수 또는 지진 홍수 등 환경 요인들이 있다. 일반적인 가용성에 따른 시스템의 분류는 [표 1]에 나타나 있다.

이 표에서 나타난 바와 같이 고가용성이란 전체 경과시간 중 99.9% 이상 주어진 서비스를 제공할 수 있는 서버

표 1. 가용도별 시스템의 분류

시스템 구조	가용도 (%)	일년당위 휴지시간
Conventional	99	3.5 days
High Availability	99.9	8.5 hours
Fault Resilient	99.99	1 hour
Fault Tolerant	99.999	5 minutes

시스템을 의미하며, 이것은 웹 비즈니스 환경에서의 지원되어야 할 최소 서비스 품질로 인식이 되고 있다. 99.999% 이상의 가용성을 요구하는 고장허용(Fault Tolerant) 시스템에 비해 고가용 서버는 다음의 특징들을 가질 수 있다.

- 고장 복구가 순간적이지 않고 수초 이상 걸릴 수도 있다.
- 시스템의 모든 요소들을 중복하지는 않는다.
- 고장 이후 성능 저하가 따를 수 있다.

고장허용 시스템은 고가용 시스템에 비해 보다 높은 가용성을 제공하지만, 시스템이 고가라는 단점이 있으며, 또한 표준이 아닌 독자적인 하드웨어 및 소프트웨어 구조를 지니므로 개발 및 업그레이드 시 많은 시간과 인력이 소요되므로 오늘날의 클라이언트/서버 환경에 부적합한 면이 있다. 따라서 웹서버를 공급하는 국내의 대부분 기업들은 고가용성과 최적의 성능을 제공하되 보다 유연하고(flexible) 값싼 클러스터(Cluster) 구조의 서버를 제공하고 있다.

제 2 장에서는 클러스터링의 장점과 고가용 클러스터링 구조에 대해 설명하고, 제 3 장에서는 국내외 대표적인 고가용 클러스터 서버 시스템을 소개한다.

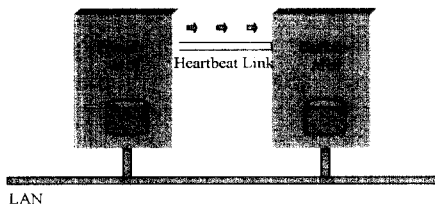
## 2. 클러스터

클러스터링(Clustering)은 고가용성을 제공하고 확장성(scalability)을 높이기 위해 두 개 이상의 서버를 물리적으로 밀접한 구조를 말하며, 각 서버들은 일반적으로 동일하거나 비슷한 시스템들로서 각자 자신의 작업을 스스로 수행할 수가 있다. 기업레벨의 시스템에서 부하가 증가하거나 또는 임계업무(mission-critical) 작업을 수행하는 시스템에서 새로운 요구가 발생할 경우, 보다 많은 서버를 클러스터 시스템에 추가함으로써 이들 요구 사항에 대처할 수 있다. 또한 하나의 서버에 고장이 발생하더라도 고장난 서버의 응용프로그램을 다른 서버나 서버들이 수행하게 함으로써 클러스터의 전체 가용성을 높일 수 있다. 그리고 시스템 관리자는 클러스터를 하나의 시스템처럼 원격 관리할 수 있으므로 보다 높은 관리성(manageability)을 제공한다.

이와 같이 고가용성을 제공하는 클러스터 구조는 passive

backup 서버와 active/active 서버의 두 가지로 구분이 된다. 또한 active/active 서버는 "duplicate everything", "shared nothing", "shared everything" 구조로 세분화된다. 이들 각 클러스터링 구조는 서버가 살아있음을 알리는 "heartbeat" 메커니즘을 가진다. Primary 서버는 상대방 혹은 백업 서버에게 주기적인 메시지를 보내는데, 만약 메시지가 중단이 되면 살아있는 서버는 상대방 서버에 고장이 난 것으로 간주하고, 고장난 서버 대신에 자신이 서비스를 담당하게 된다. 이러한 heartbeat은 서버들 사이에 직렬 케이블상에 메시지를 전송함으로써 쉽게 구현할 수 있으며, 또는 LAN 상에서 다른 메시지에 heartbeat를 실어 보낼 수 있다. 각 클러스터 구조에 대해 보다 자세히 살펴보면 다음과 같다.

## 2.1 Passive Backup 서버



[그림 1] Passive Standby 서버.

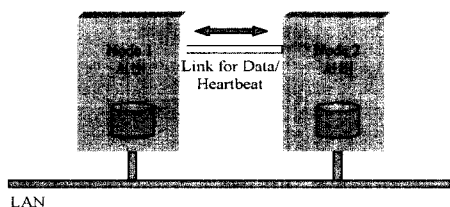
이 구조는 [그림 1] 하나의 서버가 primary 서버로 동작하고 다른 서버는 backup 서버로 primary 서버에 고장이 발생할 경우 사용된다. 이 구조에서는 backup 서버는 primary 서버에 고장이 발생하지 않을 경우에는 아무런 서비스도 하지 않고 있다가 primary에 고장이 발생할 경우에만 primary 역할을 대신한다. 따라서 구현은 쉬운 편이나, 성능 대비 가격이 높은 단점이 있다.

## 2.2 Active/Active 서버

이 구조는 passive backup 서버에 비해 두 서버 모두 평소에 서비스를 각자 제공하다가 하나의 서버에 고장이 발생하면 고장난 서버의 역할을 살아있는 서버가 함께 담당하게 하는 보다 효율적인 구조이다.

### (a) "Duplicate Everything"

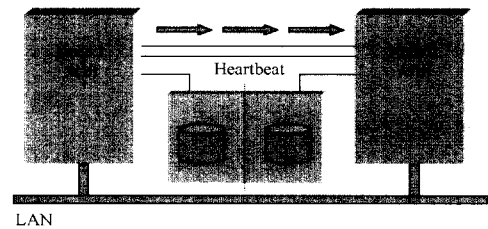
이 구조는 [그림 2] 모든 것을 이중화한 것이다. 즉, 서버가 완전히 이중화되어 있고 각자 자신의 디스크를 가진다. 고장이 발생할 경우 살아 있는 서버가 최근의 데이터를 역세스할 수 있어야 하므로 두 서버 디스크의 모든 데이터는 항상 일관성을 유지하여야 한다. 이 구조는 고가용성은 제공할지



[그림 2] "Duplicate everything" 서버.

모르나, 각 서버와 LAN에 부가적인 오버헤드(overhead)를 유발하며 따라서 성능에 영향을 끼칠 수 있다. 또 다른 단점은 두 서버 사이의 정보 전달이나 사용자 연결 전달 시 지연이 발생한다는 것이다. 이로 인해 서버 고장 시 일관성이 깨어질 위험이 있다. 이 구조의 장점은 모든 데이터가 완전히 이중화되어 있기 때문에 클라이언트는 어떤 서버를 액세스해도 무방하며 따라서 부하 균등(load balancing) 면에서 유리하다는 것이다.

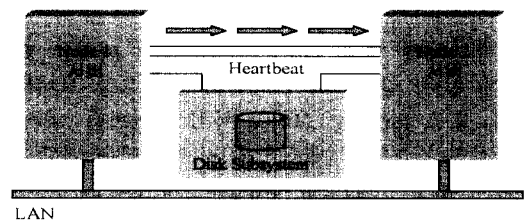
### (b) "Shared Nothing"



[그림 3] "Shared Nothing" 서버.

이 구조에서는 [그림 3] 어떠한 것도 공유하지 않는다. 즉, 두 서버가 동일한 디스크에 물리적으로 연결이 되어 있지만, 정상 상태에서 각 서버는 자신의 디스크들을 소유하고 소유자만이 그 디스크를 액세스할 수 있다. 만일 서버 고장이 발생하면, 상대방 서버는 고장난 서버의 디스크들에 대한 소유를 주장하고 액세스하게 된다. 이 구조는 디스크들 사이의 일관성 유지가 필요 없으므로 서버와 LAN의 오버헤드를 크게 줄일 수 있다. 이 구조에서는 디스크들이 단일점고장(single point of failure)이 되므로, 일반적으로 데이터 무결성(integrity)을 위해 디스크 미러링(disk mirroring) 또는 RAID(redundant array of inexpensive disks) 기술을 이용한다.

### (c) "Shared Everything"



[그림 4] "Shared Everything" 서버.

마지막으로 이 구조는 [그림 4] 모든 것을 공유하게 한 것으로서, 서버들은 동일한 디스크를 동시에 공유한다. 이 구조는 한 순간에 하나의 서버만이 데이터를 액세스해야 하므로 복잡한 락(lock) 관리자가 요구된다. 이 구조 역시 디스크 미러링 또는 RAID 기술을 사용하는 것이 일반적이다.

## 3. 고가용 클러스터 시스템

고가용성 클러스터 시스템으로는 SUN, HP 등의 상용 유닉스 업체에서 제공하는 상용 유닉스 클러스터들과 마이크로



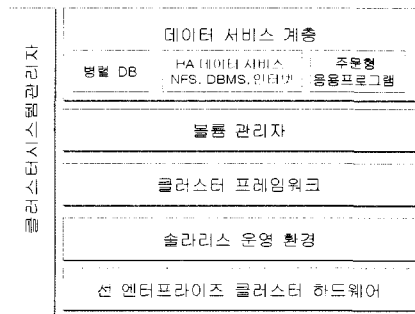
로소프트의 Windows NT나 Windows2000을 기반으로 한 NT 클러스터, 그리고 요즘 급부상하고 있는 Linux 기반의 클러스터와 그 외 FreeBSD를 비롯한 여러 가지 운영체제들을 기반으로 한 클러스터 시스템들이 선보이고 있다.

해외의 클러스터 시스템 업체로는 SUN, HP, IBM, Compaq, SGI, Intel, Microsoft, Legato, Veritas 등이 있으며, 국내의 클러스터 업체로는 LinuxOne, Clunix, 아라기술, C-EISA 등이 있다.

본 장에서는 이들 국내의 업체들 중 대표적으로 해외 업체에서는 Sun microsystems의 선 클러스터와 Microsoft의 NT 클러스터에 대하여 소개를 하고, 국내 업체중에서는 Clunix의 EnCluster와 아라기술의 SmartIP에 대하여 소개를 한다.

### 3.1 선 클러스터

선 클러스터는 단일 시스템이 제공할 수 있는 것 보다 더 높은 수준의 가용성과 서비스를 응용프로그램에 제공하기 위하여 설계되었다. 선 클러스터는 다중 서버와, 스토리지 그리고 네트워크 인터페이스를 하나의 논리적인 단위로 통합하여 클라이언트와 사용자들에게 더 높은 수준의 가용성을 제공해 줄 수 있다.



[그림 5] 선 엔터프라이즈 클러스터 소프트웨어 구조

선 클러스터의 핵심 구조는 [그림 5]에 나타난 요소들로 구성되어 있다. 클러스터 프레임워크는 고장 모니터링, 고장 감지, 복구, failover, 그리고 동적 클러스터 재구성과 같은 핵심적인 기능들을 제공해 준다. 시스템은 CPU, 메모리, 내부버스, 디스크, 디스크 제어장치, 네트워크 제어장치, 운영체제, 전원공급장치, 냉각장치, 그리고 케이블등의 고장으로부터 복구될 수 있다. 고장 모니터는 위와 같은 시스템의 기본적 수준의 요소들에 대하여 작동하기 때문에, 다른 많은 서비스 모듈들에 통합되어질 수 있다.

데이터 서비스 계층은 NFS와 데이터베이스를 포함한 다수의 응용프로그램들을 위한 failover 서비스를 제공한다. 이 데이터 서비스 계층에서는 응용프로그램들에 특화된 고장 모니터나, 복구를 위한 프로그램이 제공되기 때문에, 빠른 고장 감지나 복구를 수행할 수 있다. 데이터 서비스 계층에 포함되지 않은 응용프로그램을 엔터프라이즈 클러스터 환경에 구축하기 위하여 고가용 어플리케이션 프로그래밍 인터페이스(HA API: High Availability Application Programming Interface)를 제공한다. 각 사용자는 이 HA

API를 이용하여서, 각자의 어플리케이션에 특화된 고장 모니터나 복구기능을 만들 수 있다.

또한, 선 클러스터 시스템은 클러스터 관리자와 볼륨 관리자 그리고, 사이몬(Solstice SyMON)같은 모니터링 도구들을 제공함으로써 보다 효율적인 클러스터와 스토리지 관리를 수행할 수 있게 해준다.

### 대체 경로와 동적 재구성

동적 재구성(DR: Dynamic Reconfiguration)은 시스템 구성 요소를 온라인 또는 오프라인으로 변환함으로써, 운영 시스템의 중단이나 재부팅 없이 실행중인 시스템의 구성을 변경하는 기능으로 프로세서 보드나 네트워크 제어기 등의 시스템 요소들을 시스템 구성에 논리적, 물리적으로 포함시키거나, 제거할 수 있다. 대체 경로(AP: Alternate Pathing)은 동적 재구성의 확장 개념으로, 네트워크나 외부 스토리지의 연결 중단 없이 메타네트워크, 메타디스크 구성을 통하여 I/O 보드, 네트워크 요소의 동적 재구성을 지원해 준다.

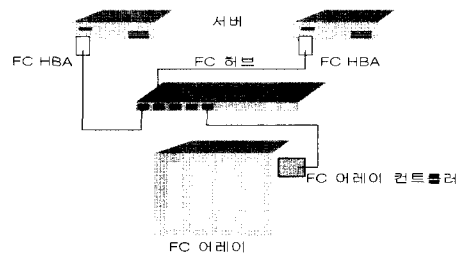
선 엔터프라이즈 x500 시리즈 서버들은 모든 시스템의 구성 요소가 복수로 설계되어 있기 때문에, 다중 모듈 환경에 대체 경로와 동적 재구성을 통하여 내부 시스템 자원에 대하여 더 확장된 개념으로 고가용성을 위한 클러스터링을 실현할 수 있다.

### 3.2 윈도우즈 클러스터링

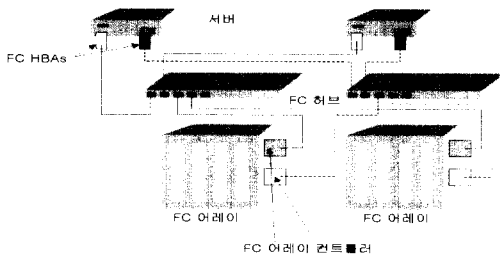
MSCS(Microsoft Cluster Server)는 1997년 MS의 윈도우즈 NT4.0 엔터프라이즈에 기본 탑재되는 형태로 처음 선보였으며, 윈도우즈 2000 어드밴스드 서버로 이어지면서, 여러 가지 성능면에서 향상을 가져왔다. 운영체제와 함께 제공되는 이점을 가졌지만 클러스터링의 기본적인 기능만을 제공하고 있으며, 현재는 2노드만 지원되고 차후, 윈도우즈 2000 데이터센터 서버에 탑재될 MSCS 는 4노드까지 지원하게 된다.

현재의 구조는 active/standby 형태로 두 대의 서버에 동일한 형태의 IIS 나 SQL 서버가 함께 탑재되어서, IIS 와 익스체인지, 혹은 IIS 와 SQL 서버가 서로 다른 서버에서 동작하며 문제가 발생할 경우 단일 서버에서 두 가지 작업을 이뤄지도록 구성하는 것이 일반적인 형태이며, standby 서버에서도 작업이 이루어지기 때문에, 부하 균등의 의미도 있다.

전체적인 클러스터의 구성은 스토리지의 공유형태에 따라서 [그림 6-1]과 [그림 6-2]의 형태로 구성이 된다.



[그림 6-1] 스토리지 공유 선택의 예 1



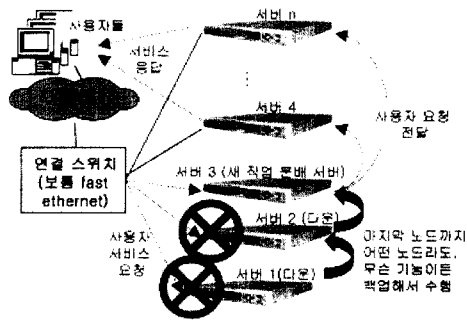
[그림 6-2] 스토리지 공유 선택의 예 2

선으로 표시된 것은 모두 광케이블이며, 이와 별도로 서버와 서버사이에는 크로스 오버케이블로 연결하거나 별도의 이더넷 허브를 통해 연결해야 한다.

이밖에 윈도우 NT/2000을 지원하는 클러스터웨어로는 레가토의 클러스터링 솔루션과 OPS(Oracle Parallel Server) 그리고 베리타스의 클러스터-X, NSI 소프트웨어의 더블 테이크 지오클러스터등을 꼽을 수 있다.

### 3.3 Clunix EnCluster

리눅스 가상 서버(LVS, Linux Virtual Server)기술에 기반한 EnCluster의 주요 특징은, 우선 기존 하드웨어 고기능 스위치만큼의 고확장성 클러스터 기능을 사용할 수 있다는 것이다. 그리고 기존 LVS 소프트웨어 클러스터링과 같은 경제성을 갖으며, 브로드캐스트/패킷 필터링 방식 이상의 고가용성을 보장하는 대칭적 결합 복구 기능을 제공한다. 또한 클러스터 내에 여러 개의 작업 분산 서버 설정 기능, 웹 브라우저를 이용한 관리 기능 등을 제공한다.



[그림 7] 대칭적 결합복구

[그림 7]은 대칭적 결합 복구에 대한 설명을 나타낸다. N대의 서버로 구성된 클러스터의 경우, LVS를 사용했다면 작업 분배 서버가 다운될 경우 전체 서버가 사용이 불가능해지는 문제가 있다. HA 패키지나 터보 클러스터의 다중 백업 기능을 이용해서 백업 서버를 정해둔다면 문제가 없겠지만, 백업 서버마저 다운된다면 문제는 피할 수 없을 것이다. 클루닉스의 대칭적 결합복구 기술은 백업 서버를 따로 지정하

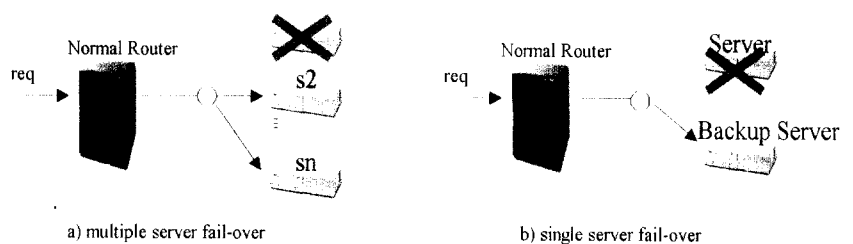
지 않고, 기본적으로 모든 노드들이 어떤 역할이든 수행할 수 있다는 점에서 대칭적이다. 즉, 작업 분배 서버가 다운되었던 일반 서버가 다운되었던, 최후의 서버가 다운될 때까지 다운되지 않은 서버들이 협의하여 다운된 서버를 대신할 서버를 선출한다. 이를 위해서는 클러스터 구성 정보를 모든 클러스터 구성 서버들이 공유할 필요가 있고, 그 내용이 최신 정보로 일관되게 유지할 필요가 있다. 클루닉스 EnCluster에서는 이러한 정보들을 데이터베이스화 하여 주기적으로 heartbeat에 필요한 정보를 담아 주고 받음으로써 모든 서버들이 내용을 일관성과 무결성을 확보한다.

EnCluster의 구성요소는 크게 3개의 구성 부분으로 이루어진다. 작업 분산을 위한 소프트웨어인 LVS 부분과 시스템 자원을 DB화하여 시스템 감시와 관리의 핵심적 기능을 수행하는 CMDB 부분, 관리 툴 부분이다. LVS는 외부에서 들어오는 서비스 요청을 실제 서버들에게 분산해 주는 소프트웨어 부분이다. CMDB는 시스템 감시와 결합 복구를 담당하는 클러스터의 핵심 부분으로 모든 시스템 자원을 데이터베이스로 구성하여 모든 노드들이 시스템 전체의 상태에 대해 동일한 내용 정보를 공유하도록 해 주는 소프트웨어이다. 마지막으로, 클러스터 시스템 전체를 관리할 수 있도록 해주는 Cluman이 있다. 이것은 LVS, CMDB 모두를 제어, 관리할 뿐 아니라 사용자 계정 관리와 네트워크 서비스까지 관리 수행해 준다.

### 3.4 아라기술 SmartIP

아라기술의 SmartIP는 자용/대칭적 부하 분산 방법을 이용한 클러스터링 솔루션으로써, 클러스터링의 특징인 부하 분산과 failover 기능을 안정적으로 지원하고 있다. 특히 SmartIP는 IP 필터링 방식을 채택하여 높은 가용성과 안정적인 부하 분산을 제공한다. SmartIP 부하 분산기의 주요 기능은 다음과 같다. 클러스터링 및 서버 부하 분산 기능으로 복수개의 서버를 묶어, 가상의 고성능 서버로 운영할 수 있도록 한다. 그리고 복수개의 서버를 서비스 무정치 상태로 운영할 수 있는 기능인 고장 허용(Fault Tolerance, Fail-Over) 기능과 복수개의 서버의 조합으로 복수개의 클러스터를 운영할 수 있는 다중 클러스터 운영 기능을 제공한다. 또한 전용 관리도구를 이용한 상태 감시 및 운영기능을 제공한다.

SmartIP는 서버 자율적 분산 기법을 사용한 failover 기능이 지원되며, 두 가지 종류의 failover를 제공한다. 하나는 부하 분산을 행하며 이루어지는 다중 서버 failover이다. 부



[그림 8] SmartIP failover



하 분산중 일부 호스트에 고장이 발생하면 작업 분담을 재 설정하여 나머지 호스트들이 모든 요구들을 정상적으로 처리한다. 또한, 부하 분산은 하지 않고 failover만을 필요로 하는 응용을 위해 단일 서버 failover가 제공된다. 서비스 중이던 호스트가 고장이 나서 중지되면 다음 호스트가 서비스를 하고 만약 이 호스트가 고장이 나면 또 다른 호스트가 서비스하는 식의 failover이다. SmartIP의 failover 기능의 특징은 클러스터 내의 호스트들이 자율적이고 대칭적으로 failover를 한다는 점이다. 따라서, 여타의 failover 솔루션이 가지는 문제점인, failover를 담당하는 마스터 서버가 오류 발생할 경우 전체 failover가 이루어지지 않는다는 점을 원천적으로 봉쇄한다. [그림 8]은 SmartIP의 failover 기능을 나타낸다.

SmartIP는 IP Filter 방식을 사용한 IP 분산기이다. 클라이언트가 클러스터 IP 에 대해 정보요청신호를 보내면, 서버 그룹의 Local Lan의 클러스터에 참여 중인 모든 서버에 요청신호가 전달(Multicast)된다. 정보요청에 대해 참여중인 서버 중 단 한 서버만이 정상적으로 패킷을 수용하고 나머지 서버들은 패킷을 받지 않는다. 패킷을 수용한 서버가 클라이언트에 응답을 하므로, 클라이언트 입장에서는 클러스터중 한 서버와 패킷을 주고 받는 형태가 된다. 각 서버들은 인터넷 트래픽 중 일정한 부분을 수용하도록 설정되어, 서버 클러스터에 주어지는 트래픽을 적절히 분산시키게 된다.

#### 4. 결 론

이상으로 웹서버를 위한 고가용성 클러스터링 기술과 실제로 이런 기술들에 기반한 시스템들에 대하여 살펴보았다. 대부분의 시스템들의 클러스터링 기술은 소프트웨어의 형태로 제공이 되며, 좀 더 높은 고가용성을 제공하기 위하여, 몇몇 하드웨어 모듈을 다중화 하는 경우도 있다. 대부분의 운영체제 벤더에서 자사의 운영체제들을 위한 고가용성 클러스터링 기술들을 제공하고 있으며, Freeware인 리눅스나 FreeBSD 기반의 클러스터링도 손쉽게 고가용성 클러스터링을 구축하는데 기여하고 있다.

이전의 클러스터링 기술이 일반 사용자들이 접하기 힘든 단지 과학 계산용의 고성능을 위하여만 사용된 기술이었다면, 오늘날에 있어서 클러스터링 기술은, 인터넷에 기반한 여러 가지 다양한 서비스를 안정적이고 높은 품질로 제공하기 위한 핵심적인 기반 기술이라고 할 수 있다.

#### 참고문헌

[1] "Intel-based Multi-Server System High Availability", HP NetServer High Availability white paper, Hewlett Packard

[2] "Designing for High Availability", White Paper, Intel Corporation

[3] Peter Weygant, "Clusters for High Availability", Prentice Hall, 1996

[4] "Sun Enterprise Cluster Failover", White Paper, Sun microsystems

[5] "Highly Available and Scalable Solutions on Tru64 UNIX AlphaServer Systems", COMPAQ, <http://www.tru64unix.compaq.com>, April 2000

[6] "AS/400 Clustering Technology White Paper", White Paper, IBM

[7] "Legato HA+ Info", Legato Product Info, <http://www-ftp.legato.com>

[8] "UnixWare 7 NonStop Clusters", SCO, <http://www.sco.com>

[9] "High-Availability Linux Project", <http://www.linux-ha.org>

[10] "Cluster Strategy: High Availability and Scalability with Industry-Standard Hardware", White Paper, Microsoft

[11] "High Availability Solutions for Unix and Real-Time Operating Systems", H.A. Technical Solutions, <http://www.tech-sol.com>

[12] "엔클러스터 기술 백서", <http://www.clinux.co.kr>

[13] "Smart IP 기술 백서", <http://www.aratech.co.kr>

[14] "리눅스윈 고가용성(High Availability) 웹 서버 매뉴얼 Ver. 0.1 ", <http://alpha-cluster.linuxone.co.kr>

[15] "가상의 슈퍼컴, 클러스터링 파워", 마이크로 소프트웨어, 2000. 7.

### 저 자 소개



#### 이철훈(李鐵勳)

1983년 서울대 전자공학과 졸업. 1983년-1986년 삼성전자 컴퓨터개발실 연구원. 1988년 한국과학기술원 전기 및 전자공학과 석사. 1992년 한국과학기술원 박사 1992년-1994년 삼성전자 컴퓨터사업부 선임연구원. 1994년~1995년 Univ. of Michigan 객원연구원. 1995년 2월-현재 충남대학교 컴퓨터공학과 조교수. 관심 분야: 운영체제, 병렬처리, 결합 허용 및 실시간 시스템.



#### 이일신(李日新)

1990년-1995년 한백정보통신 연구원. 1996년-1997년 이네트정보통신기술이사. 1997년-2000년 아라리온(주) 전략기획팀장. 2000년-현재 디스크뱅크(주) 대표이사. 주요연구 분야 : RAID 스토리지