

클러스터링을 이용한 무정지서버의 군적용방안에 관한 연구

이정찬, 이상훈
국방대학교 전산정보학과
e-mail:lee1101@dreamwiz.com

A study on Military Adapting Method of Non-stop Server using Clustering

Kyung-Chan Lee, Sang-Hoon Lee
Dept. of Computer Science, Korea National Defence Univeristy

요 약

많은 ISP업체와 서버 관리자들은 자신의 시스템 향상을 위하여 많은 비용을 쏟아 붓고 있지만 그 결과는 비판적이다. 지금까지는 하드웨어적인 성능만을 고려하였으나, 최근에는 리눅스를 탑재하여 낮은 가격과 높은 가용성을 가지면서, 증가하는 네트워크 요청에 효과적으로 대응하기 위한 해결책으로 여러 대의 호스트로 구성되는 클러스터링 기술이 각광을 받고 있다. 또한 클러스터는 저렴한 호스트들로 구성되므로 구축 비용이 절감되는 것 뿐만 아니라, 일부 호스트가 고장 나더라도 다른 호스트는 네트워크 작업 분담을 재설정하여 정상적으로 동작함으로써 가용성이 항상 유지될 수 있다는 것을 보여 준다. 최근 군에서도 정보화 과학화의 열풍에 힘입어 그 어느때보다도 업무의 전산화, 정보화에 박차를 가하고 있는 실정이다. 이로 인한 군업무에 적용되는 중대형서버의 증가와 1인 1PC정책에 의한 클라이언트의 증가, 네트워크 인프라 구축에 심혈을 기울이고 있다. 그러나 사용자의 요구만큼 군환경하에서의 정보화 업무를 담당하는 서버의 역할은 제한이 있다. 군업무의 특수성에 비추어 불때에도 사회의 금융업무와 마찬가지로 중단없는 서비스의 수행과 서버의 안정화는 군전산의 가장 중요한 요소중의 하나임에 불구하고 아직까지도 현실정은 많이 미비한 상태이다. 본 연구는 이와 같은 문제를 해결하기 위하여 NAS와 SAN개념(네트워크기반)을 도입하여 군 서버 구축 새로운 패러다임을 제공하여 업무의 통폐합과 함께 서버의 통폐합의 전초단계인 무정지 클러스터링 서버의 구축방안을 제시하여 군정보화, 과학화의 초석을 다지는 계기로 삼고자 한다.

1. 서론

요즘은 점점 더 많은 회사들이 그들의 사업을 인터넷상으로 옮기고 있어 서버가 제공하는 서비스의 어떠한 단절/중단도 사업상 손실을 의미하게 되었다. 이렇게 서버들의 높은 가용성 (availability)은 점점 더 중요해지고 있다.

그러나 정보 제공자 및 사용자의 기하급수적 증가는 필연적으로 정보 제공 서버의 과부하를 초래한다. 실제로 현재 많은 양의 정보 요청을 처리하는 서버는 과부하를 감당하기 위해서 고가의 고성능 서버를 채택한다. 그러나 이 해결책은 몇 가지 문제점을 가지고 있다. 우선 고비용을 들 수 있다. 일반적으로 고성능 서버는 상당히 고가이다. 예를 들면, 두 배의 성능을 내기 위해서는 두 배를 훨씬 넘는 비용이 들어간다. 또 다른 문제로서 가용성(availability)을 들 수 있다. 고장으로 인해 고성능 서버가 동작

을 중지하였을 때 네트워크 서비스를 할 수 없게 된다. 마지막으로 확장성을 들 수 있다. 단일 서버를 확장하기 위해서는 고비용이 소요될 뿐만 아니라 버스, 디스크, 메모리 등의 병목 현상으로 인해 확장에 제한이 있다.

현재에는 이러한 문제점을 보완하기 위해 낮은 가격과 높은 가용성을 가지면서 증가하는 네트워크 요청에 효과적으로 대응하기 위한 해결책으로 여러 대의 호스트로 구성되는 클러스터링 기술이 각광을 받고 있다. 클러스터는 저렴한 호스트들로 구성되므로 구축 비용이 절감된다. 또한 일부 호스트가 고장 나더라도 다른 호스트는 네트워크 작업 분담을 재설정하여 정상적으로 동작함으로써 가용성이 유지된다.

하나의 프로세스가 수행을 완료하기 위해 몇 백 또는 몇 천시간 이상을 요구하는 경우, 정상적으로 종료하기 위해서는 이러한 프로세스가 수행되는 시

시스템이 그 수행시간 동안 내부적인 오류나 유지 관리 등으로 인하여 중단될 없이 지속적으로 사용이 가능해야 한다.

군에서도 이러한 중단 없는 업무의 흐름이 요구되기 때문에 무정지 서비스를 위한 서버의 구축이 필수적이라 할 수 있다.

이에 본 논문에서는 가장 효율적인 클러스터링 서버의 구축방안에 대한 새로운 패러다임을 제공함으로써 군 정보화 과학화의 초석을 다지고자 한다.

본 논문의 구성은 다음과 같다. 제 2장에서는 클러스터 개념에 대하여 설명하고, 제 3장에서는 이러한 클러스터 개념을 바탕으로 클러스터 서비스 아키텍처에 대하여 설명하고, 제 4장에서는 네트워크 기반의 클러스터링 서버의 구축방법에 대하여 살펴보고, 제 5장에서 결론을 맺는다.

2. 클러스터 개념

클러스터는 여러 개의 시스템이 하나의 거대한 시스템으로 보이게 만드는 기술이다. 이런 기술에는 여러 방법이 있기 때문에 각 기술의 특징을 잘 이해하는 것이 클러스터링 기술을 잘 활용할 수 있는 방법이다. 컴퓨터 클러스터링은 디지털 VAX 플랫폼을 시초로 1980년대부터 다양한 형태로 만들어지기 시작했으며, 이러한 클러스터는 디스크 공간 같은 하드웨어 자원을 공유할 수 있었고, 여러 사용자에게 컴퓨팅 자원을 제공할 수 있었다. 원래 클러스터 컴퓨팅이라는 말은 일반적으로 유닉스 계열 운영체제의 컴퓨터에서 고가용성(High Availability, 컴퓨터의 서비스 제공 능력을 극대화하는 것)과 결함내성(fault-tolerance, 시스템 일부가 고장나도 시스템이 서비스를 계속할 수 있는 능력)을 지원하기 위해 고안된 기술에 붙여진 이름이다. 대부분의 유닉스 기반 서버 제조업체들은 이러한 클러스터링 기술과 제품을 보유하고 있는데 SUN의 Solaris 기반 SUN Cluster가 대표적이다. 물론 이 외에도 IBM, HP, NCR, UNISYS등 대부분의 유닉스 서버 업체들은 고유한 고가용성, 결함내성 클러스터링 기술을 가지고 있다. 현재 사용하고 있는 대부분의 클러스터는 노드와 관리자로 구성되어 있다. 노드는 프로세싱 자원을 제공하는 시스템이고, 클러스터 관리자는 노드를 서로 연결하여 단일 시스템처럼 보이게 만드는 로직을 제공한다.

2.1 클러스터 노드

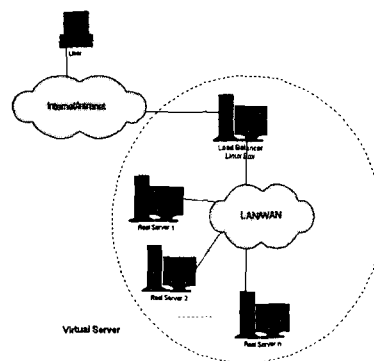
클러스터 노드는 클러스터의 실질적인 작업을 처

리하는 것을 말한다. 일반적으로 클러스터 노드는 클러스터에 속하도록 구성해야 한다. 클러스터의 역할과 업무에 따라 해당 소프트웨어는 독특하게 제작된 것일 수도 있고, 일반적인 소프트웨어 일 수도 있다. 역할에 따른 특정 소프트웨어라면 공학 계산을 위한 각 노드에 맞는 프로그램일 수도 있으며, 일반적인 소프트웨어는 로드밸런싱을 위한 클러스터라면 아파치 같은 소프트웨어를 들 수 있을 것이다.

2.2 클러스터 관리자

리눅스의 커널이 모든 프로세스에 대한 스케줄과 자원관리를 하는 것처럼 클러스터 관리자 역시 이런 관리자의 역할로써 각 노드에 대한 자원분배 및 관리를 할 수 있는 기능을 가지고 있다. 기본적으로 한개의 관리자가 필요하지만 때에 따라서는 클러스터 노드가 클러스터 기능을 하기도 하며, 대규모 환경의 경우에는 여러대의 클러스터 관리자가 있기도 하다.

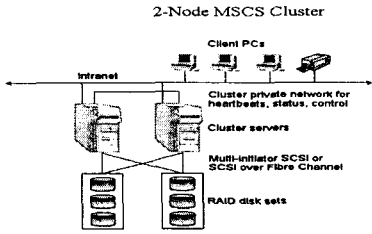
일반적인 클러스터 가상서버의 구조는 (그림1)과 같다. <그림 1>에서 보는 것과 같이 서버들은 고속 LAN 또는 지역적으로 분산된 WAN으로 연결된다. 서버들의 외부연결부는 부하분산기(loadbalancer)이다. 부하분산기는 요구를 자기 다른 서버들에게 작업할당하고 클러스터의 병렬서비스들을 하나의 IP주소상의 하나의 가상서비스로 보이도록 만든다. 높은 가용성은 노드또는 데몬의 장애나 시스템의 구조변화를 적절히 감지해 냄으로써 이루어진다.[1]



<그림 1> 일반적인 클러스터링 구조

3. 클러스터 서비스 아키텍처

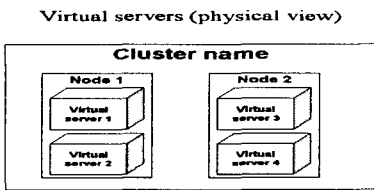
클러스터 서비스는 클러스터의 모든 서버가 액세스해야 하는 외부 일반 장치를 위해 일부 연결 미디어를 지원한다. <그림2>에서는 2노드 서버 클러스터의 구성 요소를 보여준다.



<그림 2> 2노드 서버 클러스터

클러스터 서비스의 한 가지 이점은 서버 클러스터에서 실행 중인 응용 프로그램과 서비스가 가상 서버로서 사용자와 워크스테이션에 표시될 수 있다는 것이다. 사용자와 클라이언트에게 클러스터링된 가상 서버를 실행하는 응용 프로그램이나 서비스로의 연결은 하나의 실제 서버로의 연결 과정과 동일하게 보여진다. 실제로 클러스터 내의 모든 노드에서 가상 서버로 연결할 수 있다. 사용자와 클라이언트 응용 프로그램은 실제로 가상 서버를 호스트하는 노드를 알 필요가 없다.

<그림3>에서는 4개의 가상 서버를 포함하는 2노드 클러스터를 설명한다. 각 노드에는 2개의 가상 서버가 있다. 클러스터 서비스는 가상 서버를 리소스 그룹으로서 관리하며 각 가상 서버 리소스 그룹은 IP 주소와 이 IP 주소에 매핑되는 네트워크 이름의 2가지 리소스를 수록한다.



<그림 3> 클러스터 서비스 내의 실제 가상 서버 가상 서버로의 응용 프로그램 클라이언트 연결은 클러스터 서비스가 가상 서버 주소로 발행하는 IP 주소만 알고 있는 클라이언트 세션을 사용하여 이루어진다. 클라이언트는 단순히 개별 네트워크 이름과 IP 주소이다. <그림 4>는 4개의 가상 서버를 지원하는 2노드 클러스터의 예제를 사용하여 클라이언트 노드의 클라이언트 보기와 4개의 가상 서버를 설명한다.

<그림 4>에서 볼 수 있듯이 클라이언트는 IP 주소와 이름만 볼 수 있으며 가상 서버의 실제 위치에 대한 정보를 알 필요가 없다.

Virtual servers (client view)

Node 1	Node 2	Virtual server 1	Virtual server 2	Virtual server 3	Virtual server 4
IP address 1.1.1.1 Network name WINCCNode1	IP address 1.1.1.2 Network name WINCCNode2	Internet information Server IP address 1.1.1.1 Network name WINCC.VS1	HTTP Server IP address 1.1.1.2 Network name WINCC.VS2	Microsoft Exchange IP address 1.1.1.1 Network name WINCC.VS3	SQL Server IP address 1.1.1.2 Network name WINCC.VS4

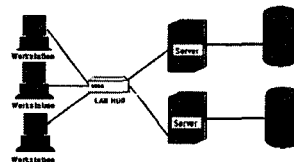
<그림 4> 클러스터 서비스 가상서버 클라이언트 따라서 클러스터 서비스는 가상 서버를 실행하는 응용 프로그램에 높은 가용성을 제공할 수 있다.

응용 프로그램이나 서버에 장애가 발생하는 경우 클라이언트 서비스는 전체 가상 서버 리소스 그룹을 클러스터의 다른 노드로 이동시킨다. 이러한 장애가 발생하면 클라이언트는 응용 프로그램의 해당 세션에서 장애를 발견하여 원래 연결과 동일한 방법으로 재연결을 시도한다. 클러스터 서비스는 단순히 가상 서버의 IP 주소를 복구 작업 중 클러스터에서 사용 가능한 노드로 매핑하기 때문에 성공적으로 재연결된다. 클라이언트 세션은 현재 응용 프로그램이 클러스터의 다른 노드에서 실제로 호스트되고 있는지 알 필요 없이 연결을 재설정할 수 있다.

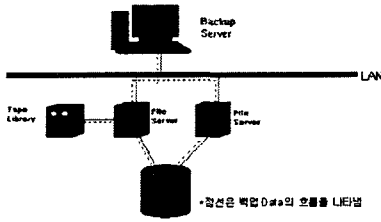
이렇게 하면 응용 프로그램이나 서비스의 고 가용성을 유지할 수 있지만 장애가 발생한 클라이언트 세션과 관련된 세션 상태 정보는 응용 프로그램 복구 중 디스크의 클라이언트 세션 데이터를 복구를 위해 저장하도록 응용 프로그램이 설계되거나 구성되지 않는 한 손실된다. 클러스터 서비스는 고 가용성을 제공하지만 응용 프로그램 자체적으로 무결성 트랜잭션을 지원하지 않으면 응용 프로그램 무결성을 제공하지는 않는다.

4. 네트워크 기반의 클러스터링 서버 구축방법

기존의 서버 구축은 <그림5>와 같이 각 서버당 스토리지가 종속되는 형태로 서버 클러스터링에 있어 근원적으로 클러스터링 서버의 구축이 제한된다.



<그림5> 전통적인 네트워크-스토리지 구조 그러나 <그림6>과 같이 서버가 스토리지에 종속되는 형태에 있어서는 문제가 다르다.



<그림6> 스토리지 중심의 네트워크 구조(SAN)

한마디로 고속네트워크라고 할 수 있다.

즉, 호스트 컴퓨터에서 SCSI를 통한 스토리지 서버와 신속하게 데이터를 주고 받을 수 있는 것처럼 네트워크상에서 Fiber Channel(FC)의 이점인 고속 전송과 장거리 연결 및 멀티 프로토콜 기능을 활용 실현하는 기술이다.

SAN의 표준화를 위한 단체인 SNIA(Storage Networking Industry Association)에서는 "SAN이란 호스트 컴퓨터의 종류에 구애받지 않고 별도의 연결된 저장장치 사이에 대용량의 데이터를 전송시킬 수 있는 고속네트워크"라고 정의하고 있다. 데이터량의 관리라는 것은 데이터가 필요한 곳으로 이동시키거나 또는 과부하가 걸린 서버의 데이터를 보다 여유가 있는 다른 서버로 옮겨 줌으로서 서버간의 가장 빠른 전송로를 찾아주는 고도의 정교한 소프트웨어가 요구되어진다. 이러한 소프트웨어의 지원을 받아 SAN은 항상 지속적인 데이터를 이용하도록 하고 관리의 편리성을 제공하는 스토리지 아키텍처이다.

기존 시스템에서는 서버들이 파일 그 자체 대신에 어떤 하나의 파일 카피들을 공유했었다.

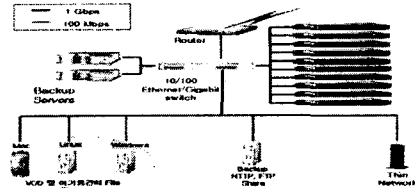
하지만, SAN은 파일이 이동중일때 데이터를 신속히 주고 받을 수 있으며 정보를 관리할 수 있는 것이다. 그러나 SAN을 구축하기 위해서는 저장장치만으로 해결할 수 없다. 특정 HOST 컴퓨터가 소유하고 있는 정보는 공유될 수 없기 때문이다. 소유되지 않은 정보에 접근하기 위해서는 네트워크의 체중, 스토리지 비용의 증가와 CPU에 체중을 더할뿐 아니라, 데이터의 교환 또는 COPY를 필요로 한다.

또한 SAN에 관한 모든 장치간의 SAN 네트워크 연결을 위하여 케이블, 스위치, 허브, 호스트어댑터 등의 물리적 장치가 필요 하다.

이에 보완적인 개념으로 NAS를 들 수 있다.

NAS는 Ethernet과 같은 LAN 인터페이스를 통해 네트워크에 직접 연결되는 새로운 개념의 데이터 저

장 시스템이다. NAS는 네트워크상에서 서로 다른 기기종 플랫폼간의 효율적인 데이터 통합 사용과 온라인 데이터의 전송/저장이 서버를 거치지 않음으로써 서버 성능 향상에도 기여하는 획기적인 저장 장치이다. NAS의 일반적인 구성은 <그림7>과 같다.



<그림 7> NAS의 일반적인 구성

NAS가 네트워크에 연결된 시스템이 유닉스나 윈도우든 상관없고 확장성도 무한대라는 이점 때문에 현재 ATM기반의 각 서버당 스토리지를 유지하고 있는 우리 군의 고비용의 서버구조를 혁신할 것으로 전망된다.

5. 결론

이상으로 네트워크기반의 클러스터링 서버 구축방안을 살펴보았다. 이번 연구를 통해 군 서버 구축의 패러다임을 재구성하여 중단없는 군 정보화 서비스에 기여할수 있는 가능성을 제시하였다.

그러나 네트워크기반(SAN, NAS)의 클러스터링 서버 구축은 아직까지는 완벽하게 검증된 기술이라기 보다는 시험단계인 기술이라고 할 수 있다.

향후 연구에서는 실업무에서 운영간 도출되는 보안 사항들에 대한 최적의 대안제시방법과 가장 효율적인 서버 구축방안에 대한 지속적인 연구가 진행되어야 할 것이다.

참고문헌

- [1] 이선중, Creating Linux Virtual Servers, 2000.10
- [2] 김동호, 리눅스상에서 고가용성 웹서버 클러스터의 설계 및 구현, 2002.02
- [3] 김남주, 체크포인팅 기법을 이용한 신뢰성 있는 클러스터 컴퓨팅 환경 구축, 2000.02
- [4] 신순철, 부하분산 클러스터 제작과 Fail Over, 2002.03
- [5] <http://www.linuxvirtualserver.org>
- [6] <http://www.mti.com>
- [7] <http://www.clusterkorea.org>
- [8] <http://www.oracle.com>