

# 원격지 시스템 상태를 고려한 안정적인 백업시스템에 관한 연구

김범재\*, 이준호\*\*

## A Study of Stable Backup System Considered the Status of Remote System

Bum-Jae, Kim\* Jun-Ho Lee\*\*

### 요 약

본 논문은 네트워크를 이용한 안정적이고 빠른 백업 방법에 관한 연구이다. 본 논문에서는 기존 네트워크의 백업방식과 다른 3가지 방식을 통하여 백업을 할 수 있도록 네트워크 및 소프트웨어 구조를 제안하였다. 첫째, 보안에 취약한 네트워크 백업방식을 Socket통신을 이용하여 보완하였으며, 둘째, Target System의 Agent를 항상 구동시켜 Target System의 Resource를 계속 점유하는 방식이 아닌, 백업 시에만 Utility를 구동시켜 Resource 사용율을 낮춰 백업하는 방식, 셋째, 전송데이터 size(byte/1회)를 고려한 네트워크 분할/순차백업방식이다. 위 세 가지 백업방식을 통하여 기존의 백업방식과 비교하여 안정적이고 빠른 백업을 할 수 있었다.

### ABSTRACT

In this dissertation, in the light of the above situations, we intend to use new approach for stable and fast backup system. The approach that we propose network and software infrastructure has three different ways by comparison with existing network backup systems. First of all, we use a socket communication so that resolve the shortcoming of weakness of securities; second, TBD utility for minimizing target system loads in the way of instant operating; and third, time division and bandwidth division backup method considered transfer data size(byte) might be useful more stable and fast comparing with FTP Transfer.

**Keywords :** Network Backup System, Online Backup, Schedule Backup, Auto Backup System, Backup Method, Data Backup.

### 1. 서 론

IT 기술의 발달로 인해 전산 시스템에 존재하

는 데이터들의 형태가 다양해지고 복잡해졌다. 그로 인해 시스템 관리자들은 무 중단 서비스와 백업 및 복구에 많은 관심을 가지고 있다. 현재의

\* 서울과학기술대학교전자공학과

\*\* 교신저자 서울과학기술대학교 전자공학과 교수

접수일자 : 2011년 4월 21일, 수정일자 : 2011년 5월 12일, 심사완료일자 : 2011년 6월 1일

IT서비스(증권/전산/은행 등)는 실시간 서비스가 주류를 이루고 있으며 기업에서는 각종 전산화된 시스템을 도입하여 처리하기 때문에 업무의 효율화를 높이고 있다. 이러한 시스템이 갑작스런 외부의 환경에 의해 중단되었다고 가정을 하면, 그로 인한 손실은 기업의 존폐를 결정할 정도로 크다고 하겠다.[1]

[장애유발 요인]

- 시스템 노후화로 인한 장애
- 자연재해로 인한 시스템 유실
- 운용자의 실수
- 기타(Virus, 외부공격)으로 인한 자료 유실

컴퓨팅 시스템을 안정적으로 운용하기 위해서 전 세계 재해복구 시장에서 스토리지 소요량이 IT 예산의 5%~17% 이상의 높은 성장을 전망하고 있다.[2] 또한, SAN(Storage Area Networks)과 NAS (Network Attached Storage) 그리고 하드디스크의 용량 증가로 인한 대용량의 안정적인 백업 시스템 구현이 가능해져 효율적인 백업관리 시스템 구축을 용이하게 하고 있다.

백업하고, 시스템 장애 시 원활한 복구를 할 수 있는 방법에 관한 연구와 Target System의 상태(CPU/Memory/Network)를 고려한 안정적인 백업에 대하여 연구하고자 한다.

## 11. 백업방식에 관한 방법

### 1. Tape Media 백업

과거의 백업방식은 대부분 Tape Media를 이용한 백업이 주류를 이루었다. Tape Media 백업의 문제점을 살펴보면 데이터 증가에 따른 Tape Drive와 Tape Library의 추가 도입 비용이 증가되고 Tape Media의 장애로 인한 복구 실패율증가(물리적 실패), 2차 백업 및 원격지 복제, 소산시의 제반 문제들이 발생되고 있다. 또한 Tape Drive의 점유시간이 길어지면 Backup Queue가 길어지고, 따라서 목표시간 내에 백업이 완료되지 않을 수 있으므로 다른 중요한 서버가 Backup Mode에서 장시간 대기할 수 있는 상황이 발생할 수 있다.[3] 보관 상태에 따라서도 Tape Media의 수명은 다르다. 이 백업방식은 먼지와 습도에 민감하여 데이터 복구 시 실패할 가능성이 높아진다.

### 2. DAS(Direct Attached Storage) 백업

디스크를 이용한 백업은 Disk 가격의 하락과 가상화 기술의 발달로 기존 Tape Media 백업 방식이 가지고 있는 단점을 많이 보완해 줄 수 있는 방법으로 관심을 갖게 되었다. Server와 전용 Cable로 연결한 외장형 저장장치, Server/Client 환경에서의 부족한 저장 공간을 쉽게 확보하는 방법으로 서버 자체에 물리적 외부 저장 장치를 연결하는 방법이다. 네트워크에 연결된 각 서버에 외부 저장 장치를 추가함으로써 필요한 데이터를 물리적으로 가까운 곳에서 접근할 수 있고 확장이 용이하다.[4] 하지만 데이터의 증가에 따른 외부 저장 장치의 지속적인 추가는 서버의 효율성을 저하시키는 문제요인으로 지적되고 있다.

표.1 Media/SAN/Network 백업 비교  
Table.1 Comparison of Media/SAN/Network Backup[3]

백업방식	미디어	SAN	Network
External Interface	SCSI	Network (TCP/IP)	FC(SCSD)
접속장치	IO Port	FC Port, SAN S/W	NIC Port, 네트워크 S/W
파일시스템 공유	불가능	불가능	가능
접속 속도	IO 포트 속도	채널 속도	LAN과 채널 속도
유지비용	저가	고가	저가
편의성	아주불편	다소불편	쉬움
특징	소규모의 독립된 구성	유연성/확장성/편의성이 우수	파일 공유가 가장 안정적, 신뢰성 높음

본 논문에서는 시스템 장애 사항에 대비해 네트워크를 이용한 Target System을 주기적으로 외부

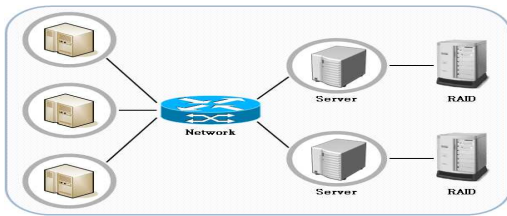


그림.1 Direct Attached Storage 구성  
Figure.1 Direct Attached Storage Configuration

### 3. SAN(Storage Area Network) 백업

Storage Area Network는 특수 목적용 고속 네트워크로서, 대규모 네트워크 사용자들을 위하여 서로 다른 종류의 데이터 저장장치를 관련 데이터 서버와 함께 연결해 별도의 랜(LAN:근거리통신망)이나 네트워크를 구성해 저장 데이터를 관리하는 방식이다. SAN은 Data를 한 공간에 빠르게 전송할 수 있는 저장 방식이다.[3]



그림.2 Storage Area Network 구성  
Figure.2 Storage Area Network Configuration

하지만, 위와 같이 SAN을 이용한 백업 방식은 시스템 구축 시 많은 비용과 장비들의 투자와, 기존 시스템들의 업그레이드가 필수적이다.[6][9]

### 4. NAS(Network Attached Storage) 백업

네트워크가 발달하지 못하고 데이터양이 많지 않던 시절에 부서별로 파일공유와 관리의 편의를 위해 시작되었던 솔루션이다. 파일공유 및 파일 서비스 기능을 위해서는 범용OS에서 제공되는 일부분의 기능(NFS 또는 CIFS)를 이용하였고 데이터 저장장치는 주로 서버에 내장된 디스크를 사용하였다. 파일서버의 한계를 극복한 것이 NAS(Network Attached Storage)이다.[3]

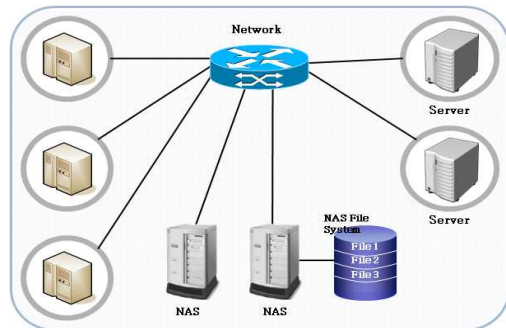


그림.3 Network Attached Storage 구성  
Figure.3 Network Attached Storage Configuration

NAS는 저장장치의 기능을 강조한 것으로 저장장치 부분의 하드웨어적 성능/기능뿐 아니라 소프트웨어적 기능이 예전의 파일서버와는 차별화되었다. 그리고 I/O측면에서도 범용 OS대신에 파일 서비스에 특화된 전용의 OS를 채택함으로써 보다 나은 I/O성능을 제공하고 있다. 그리고 역할에 있어서도 기존의 파일서버가 End-user단말에 대한 파일 서비스를 제공하는 역할을 강조한 반면 NAS는 End-user단말에 대한 기존 파일서버의 역할뿐만 아니라 Application Server의 데이터 네트워크가 연결된 곳에서는 어디서라도 스토리지를 접속해서 사용할 수 있는 Application Server에 대한 저장장치로서의 역할을 하고 있다.[3]

표.2 DAS/NAS/SAN 비교  
Table.2 Comparison of DAS/NAS/SAN

구분	DAS	NAS	SAN
구성요소	App 서버, 스토리지	App 서버, 파일 서버, 스토리지	App 서버, 스토리지
접속장치	없음	Eth S/W	FC Ch S/W
스토리지 공유	가능	가능	가능
파일시스템 공유	불가능	가능	불가능
파일관리	App 서버	File 서버	App 서버
접속속도	채널속도	Eth채널속도	채널속도
특징	소규모 독립된 구성에 적합	파일 공유를 위한 안정적이고 신뢰성 높은 솔루션	유연성/확장성/편의성이 가장 뛰어남

### III. 네트워크를 통한 백업 방법

#### 1. Veritas NetBackup

##### 1) 개요

VERITAS NetBackup은 대기업의 UNIX 및 PC 클라이언트를 위한 고성능, 산업 첨단의 백업 및 저장 복구, 공간 관리 서비스를 제공한다. 대용량 데이터베이스의 신속한 백업과 중앙 집중식 관리 기능, Main Frame급 볼륨 관리, 고급 테이프 라이브러리와 로보트 공학을 지원함으로써, 기업의 컴퓨팅 환경을 지원하고 데이터 보호를 제공한다.[6]

##### 2) 주요 특징

- (1) 다중 서버 백업, 스탠드얼론.
- (2) 중앙 관리 백업 및 복구 솔루션 제공.
- (3) Unix/윈도우NT 기능 지원

#### 2. Legato Networker

##### 1) 개요

Legato Networker는 Global 1,000 회사들 중 90%가 사용하는 데이터 보호 솔루션이며, 네트워크상에서의 온라인 백업, 무인 자동백업, 데이터의 효과적인 관리 및 처리를 제공하는 소프트웨어로써 단순 서버 백업에서부터 Enterprise Storage Management에 이르기까지 사용자 환경에 맞는 최적의 기능을 제공한다.[7]

##### 2) 주요 기능

- (1) 네트워크 환경에서 백업 및 복구
- (2) 간단한 Backup Schedule 사용
- (3) 다양한 Server(UNIX, Win NT) 지원

#### 3. 제안하는 백업 방식

##### 1) 개요

본 논문에서 제안하는 백업관리 방식은 장애 사항에 대비해서 Target 시스템을 스케줄에 의해서 백업하는 방식이다. 백업 대상 시스템의 내적인 요인에 의한 시스템 Down 및 데이터 유실을 최소화 하고, 복구를 원활히 하기위해 외부 백업을 수행한다. 또한, 백업 시 Target System의 서비스에 영향 없이 안정적으로 백업 할 수 있도록 Target System의 Resource를 체크하여 백업 한다.

##### 2) 주요 기능

- (1) Backup Schedule 사용
- (2) UNIX, Win NT 지원
- (3) Target System의 Resource 고려한 백업
- (4) 동시시간대에 하나의 네트워크 리소스를 하나의 백업 자원이 사용
- (5) 고유의 네트워크 Architecture를 이용한 Utility 구동으로 외부접속을 차단

표.3 백업솔루션 제조사별 비교

Table.3 Product Comparisons

기능	Veritas NetBackup	Legato NetWorker	FTP 백업	제안 방식
중앙집중관리	○	○	○	○
파일시스템백업	○	○	X	○
다양한 압축	○	○	X	○
온라인 백업	○	○	○	○
LAN백업	○	○	○	○
분산백업	X	X	X	○
동시백업	○	○	○	X
Agent 사용	○	○	X	○
Agent 구동	상시	상시	X	백업

### IV. 실험 장치 및 실험 방법

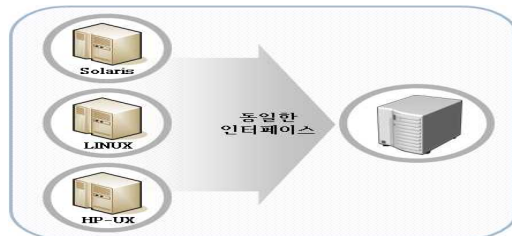


그림.4 백업서버 네트워크 구성  
Figure.4 Backup Server Network Configuration

본 논문에서 구성하려고 하는 백업 서버의 구성은 앞서 말한바와 같이 하드웨어적인 추가 구성없이 네트워크를 이용한 백업 방식으로, 사용할 수 있는 주변의 자원을 활용하여 빠르고 안정적인 백업이 가능하도록 여러 가지 측면에서 고려하였다.

#### 1. Resource 체크

네트워크를 이용한 백업 방식에서의 Resource 체크는 Target System의 상태를 고려하여 백업하

는 방식으로 기존의 FTP를 이용한 백업에서 할 수 없는 Resource 사용율을 체크하여 백업을 수행한다. Resource 체크는 Target System의 CPU, Memory, IO의 사용량을 시스템에서 제공하는 운영자원 정보를 이용하여 필요한 데이터를 얻는다.

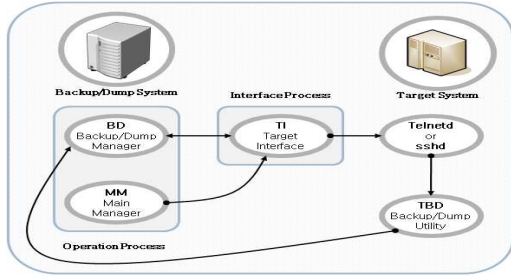


그림5. 프로세서 흐름도  
Figure.5 Processor flow

Figure.5 와 같이 백업서버는 Backup/Dump Manager 와 Main Manager, Target Interface 프로세서가 유기적으로 정보를 교환하고 있다. Target System의 백업을 수행할 경우, TI를 이용하여 Target System의 TBD Utility를 구동하고, 구동된 TBD Utility는 (Solaris 시스템의 경우) sar, vmstat, iostat의 명령어를 수행하여 Target System의 Resource정보와 데이터를 Backup/Dump Manager에게 전송한다. 이때, Backup/Dump Manager는 정해진 임계치와 비교하며 백업을 수행하고, 백업 시 Target System의 Resource 사용율이 임계치를 초과할 경우 데이터 전송을 Pause하였다가 Resource 사용율이 낮아지면 백업을 재계하여 안정적으로 백업할 수 있다.

1) CPU Resource Check

시스템에서 제공하는 명령어를 이용하여 Sys, User, Idel, Wio와 같은 Target System의 CPU Resource 사용율을 읽어 들인다.

```
% sar -P ALL 1 2
AIX aix: 1 5 0001001F4C00 05/20/03
12:54:00 cpu      %usr    %sys    %wio    %idle
12:54:02 0         32      52      15      1
1         31      45      19      4
12:54:03 0         31      49      17      20
1         35      34      9       21
평균      0         29      50      11      9
1         34      39      13      14
-         32      45      12      11
```

그림6 CPU 사용율 체크  
Figure.6 System CPU Resource Check

2) Memory Resource Check

```
%vmstat -p 1 10
Usage: vmstat [-cisS] [disk...] [interval] [count]

%vmstat 1 1
procs memory page disk faults cpu
r b w swp free re mf pi po fr de sr s6 sd sd sd in sy cs us sy id
0 0 0 44048 2396 3 959 44 34 43 0 1 0 2 2 1 867 6394 3662 18 9
72
```

그림7 메모리 사용율 체크  
Figure.7 System Memory Resource Check

시스템에서 제공하는 명령어를 사용하여 Target System의 Memory Resource 사용율을 읽어 들인다.

3) IO Resource Check

```
% iostat 3 1
tty: tin tout avg-cpu: % user % sys % idle % iowait
0.0 388.0 23.6 16.0 54.7 5.8

디스크: % tm_act Kbps tps Kb_read Kb_wrtn
hdisk1 0.1 5.8 0.1 14287 8650449
hdisk0 0.0 0.0 0.0 0 0
hdisk2 0.0 0.0 0.0 0 0
cd0 0.0 0.0 0.0 0 0
```

그림8 IO사용율 체크  
Figure.8 System IO Resource Check

IO Resource도 CPU/MEMORY Resource 사용율과 마찬가지로 방법으로 시스템에서 제공하는 "iostat" 명령어를 사용하여 필요한 Resource정보를 읽고, 읽은 데이터로 백업 시 데이터 전송을 제어한다.

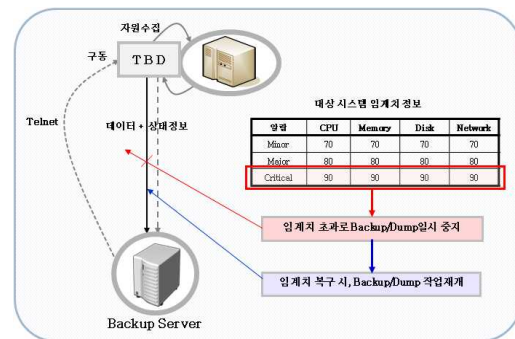


그림9 TBD Utility 백업 제어  
Figure.9 TBD Utility Backup Control Flow

Target System의 Resource 사용율을 고려한 백업방식은 Figure.9 과 같이 백업서버에서 Telnetd 이나 sshd를 이용하여 Target System의 TBD Utility를 구동시킨다. 이때 구동된 TBD Utility는 백업데이터 + 상태정보(CPU, Memory, IO)를 백업서버로 전송한다. 백업 서버의 Backup/Dump Manager는 시스템 별 정해진 CPU, Memory, IO

임계치와 비교하며 백업을 수행하는데, 상태정보가 임계치를 초과할 경우 데이터 전송을 잠시 Pause 하였다가 Resource 사용율이 임계치보다 낮아지면 다시 백업을 재계한다. 이러한 방식으로 백업을 수행하게 되면, 과부하로 인한 서비스 장애를 예방할 수 있다.

2. Data 전송

1) FTP 데이터 전송 방식

FTP Command는 data connection을 위해 data port, transfer mode, representation type, structure 등과 같은 파라미터와 store, retrieve, append, delete 등과 같은 파일 시스템 operation의 성격을 지정한다. User-DTP는 지정된 데이터 포트에서 "listen"해야 하고, server는 지정된 파라미터에 따라 data connection을 초기화 하고 데이터 전송을 시작한다. 데이터 포트는 control connection을 통해서 FTP command를 초기화한 호스트와 같을 필요는 없다. 그러나 user와 user-FTP 프로세스는 반드시 지정된 데이터 포트에서 "listen"하고 있어야 한다. 또한, Data connection은 전송과 수신의 시퀀스 레이아웃에 사용될 수도 있다.

User는 파일을 로컬 호스트가 아닌 두 개의 호스트 간에 파일을 전송하고 싶을 수 있다. User는 두 개의 서버간에 control connection을 setup하고 두 서버간의 data connection을 정리한다. 이 경우에 control 정보는 user-PI를 거치지만 데이터는 두 서버간에서만 전송된다.[8]

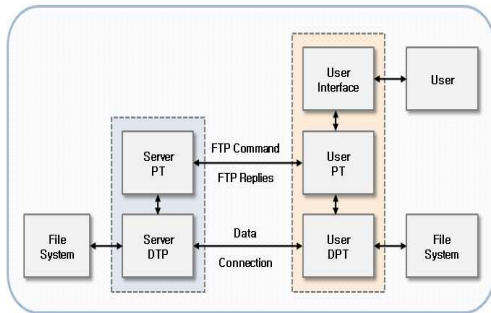


그림.10 FTP 데이터 전송  
Figure.10 FTP Data Transfer Flow

프로토콜은 데이터 전송이 진행중인 동안

control connection이 open되어 있기를 요구한다. 두 개의 서버가 데이터를 전송을 마친 후에 FTP 서비스의 사용을 끝내고 control connection을 close하는 것은 user의 몫이다. Command없이 control connection이 close되면 server는 데이터 전송을 중단할 것이다.[10]

2) Socket을 이용한 TBD 데이터 전송 방식

Figure.10 과 같이 FTP는 Data Connection을 통해 Data를 Client에서 Server로 전송을 한다. 대부분의 상용 FTP통신은 21번 Port를 이용하여 연결제어를 하고, 20번 Port를 이용하여 Data를 전송한다. 이때 Data Size를 1회 64k의 size로 통신하여 데이터를 전송하고, 전송된 데이터 다시 파일로 만든다. 하지만 본 논문에서 연구한 바와 같이 Socket통신을 이용한 128k/1회 전송 시 데이터 전송 속도 및 분할된 Data를 다시 파일로 만드는 시간이 단축되어 결과적으로 빠르게 백업할 수 있다.

표.4 백업 방법별 비교  
Table.4 Comparison of Backup Methods

방식	전송방식	전송속도	리소스	비고
TBD	TCP/IP Socket Binary	128k/1회로 Size한정	(CPU) 4 % 이 내	Target System 사용율
FTP	FTP Protocol Binary	64k/1회로 Size한정	(CPU) 4 % 이 내	파일전 송방식

Table.4는 백업방식 별 전송방식의 차이점으로 TBD Utility를 구동하여 백업하는 방식과 FTP를 이용한 백업 방식의 차이점을 비교하였다.



표.5 TBD Utility 백업 시험 결과  
Table.5 TBD Utility Backup Test Result

시스템(Name)	백업결과 판정(CPU %)	
	TBD Utility	FTP
HLR9(Solaris)	30초(4%)	1분31초(CPU4%)
HLR8(Solaris)	33초(4%)	1분29초(CPU4%)
HLR38(Solaris)	30초(4%)	1분29초(CPU4%)
CDMA-GLRH-Ux)	3분 43초(4%)	1분57초(CPU4%)
GSM-GLRH-Ux)	7분 53초(4%)	5분 18초(4%)
RFC(Linux)	25초(4%)	1분25초(2%)
CSMS(Hp-Ux)	23초(4%)	47초(3%)
LDAP(Linux)	31초(4%)	52초(4%)

성능비교 시 같은 네트워크 환경(네트워크 속도 100M Byte FULL)에서 시험을 하였으며, 네트워크를 통한 백업 테스트는 Table.5와 같은 결과를 얻었다. TBD Utility를 사용하는 방식 백업은 리소스 사용율이 낮은 Target System의 경우 Target System의 Resource 사용을 정보를 읽어오면서도 FTP백업 방식과 비교하여 1/3정도의 백업시간이 단축되었다. 하지만, 상대적으로 Resource 사용율이 높은 시스템의 경우 오히려 백업시간이 증가하여 백업의 효율은 떨어지지만, 실시간 서비스 시스템의 안정성을 고려하여 서비스에 지장을 초래하지 않고 안정적으로 백업을 할 수 있었다.

본 실험에서는 Target System의 보안 및 안정성을 고려하여 Agent를 이용한 Target System의 자원정보를 읽어오는 방법을 사용하지 않았다. Agent를 이용할 경우 Target System의 일정량의 부하를 지속적으로 점유하거나, Target System의 환경변화 즉, Virus or 불특정 Processor의 오작동이 발생할 경우 구동중인 Agent에 영향을 주고, 이로 인한 Agent의 오류로 CPU점유율이 증가하여 Target System에 영향을 줄 수 있는 요인으로 작용할 수 있다. 하지만, TBD Utility 구동방식은 이러한 외부적인 불안 요소로 인한 시스템 장애를 미연에 방지하기 위해, 백업서버로부터 백업 요청을 받을 경우에만 TBD Utility가 구동하여 백업함으로써 Target System에 영향을 주지 않는다. 또한, TBD Utility 구동하고자 할 경우, Socket을 통한 내부규격에 맞는 조건(Schedule Code, System IP, Backup Path, Date, 백업시스템 IP 등)이 성

립되어져야만 TBD Utility가 정상적으로 구동되어 백업이 실행되어지도록 설계하여 시스템 운용자에 의한 오류로 인한 장애 발생 여지를 제한하였다.

### 3. 대역별 분산백업 방식

네트워크를 통한 백업은 하나의 Resource를 어떻게 사용하느냐에 따라 데이터의 전송속도에 많은 영향을 끼친다. 본 논문에서 제안하는 Network 전송방식은 Figure.11 과 같이 Network 대역별 한 번에 하나의 File을 전송한다.

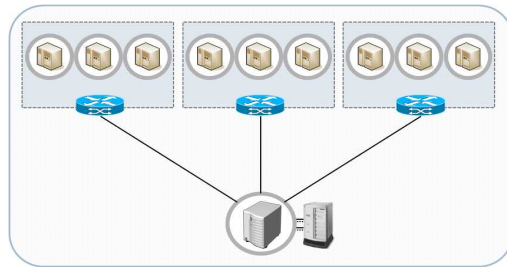


그림.11 대역별 네트워크 백업 구성도  
Figure.11 Band by the Network Configuration

하나의 Network자원을 통하여 File 전송 시, Client Socket 생성자에서 File객체에 해당하는 하나의 Socket을 Open() 한다. 이 하나의 Socket이 Close()하기전에, 다른 Socket을 Open()을 하고 File을 전송할 경우, Network 및 I/O 부하가 상당히 많이 발생한다. 이렇게 부하가 많이 발생할 경우 File 전송 시 하나의 파일을 전송하는 시간이 증가하고, 전체적으로 File전송시간이 많이 소요되어 전체적인 백업 Size가 감소하게 되는 현상이 있다. 이에, 본 논문에서 제안하는 전송방식은 Network를 세분화 하여, 하나의 Network을 통하여서는 하나의 Socket만을 Open()하고, File에 대한 전송이 완료되어 Socket을 Close() 한 후, 다시 Socket을 Open()하는 순차적 File 전송방식을 사용하여, 동시에 다중의 접속을 통한 File 전송 방식 보다 많은 데이터를 적은시간에 전송할 수 있었다.

표.6 동시백업 vs 순차백업 비교  
Table.6 Comparison of Concurrent and Sequential Backup

시스템(Name)	백업결과 판정	
	동시백업	순차백업
HLR9(Solaris)	49초(CPU10%)	24초(CPU4%)
HLR8(Solaris)	51초(CPU11%)	26초(CPU5%)
HLR38(Solaris)	1분 08(CPU10%)	28초(CPU4%)
CDMA-GLR (HP-UX)	46초(CPU12%)	19초(CPU4%)
GSM-GLR (HP-UX)	48초(CPU13%)	25초(CPU5%)
RFC(Linux)	50초(CPU11%)	25초(CPU6%)
CSMSS (HP-UX)	1분 02초 (CPU10%)	29초(CPU4%)
LDAP(Linux)	45초(CPU12%)	30초(CPU4%)

Table.6 의 결과와 같이 각 각의 100M Byte x 4 Ea 의 File을 하나의 Network를 통해 전송 시 시스템 Resource 사용을 및 데이터의 전송시간을 단축할 수 있었다.

## V. 결론

본 논문에서는 하드웨어적인 추가 구성없이 네트워크 환경 하에서 Target System의 보안과 안전을 고려한 안정적인 백업 방식에 대하여 연구하였다. TBD Utility를 이용한 백업 방식과 대역별 분산 백업 방식을 이용하여 아래와 같은 결론을 얻었다.

1) 베라타스나, HP에서 개발한 사용 백업소프트웨어 경우와 같이 Agent가 Target System에 존재하는 방식이 아닌, Utility를 이용하여 필요시에만 TBD Utility를 구동시켜 백업하는 Resource를 체크하는 방식으로 TBD Utility로 인한 Target System에 영향을 주는 문제가 발생하지 않는다.

2) TBD Utility 구동 시 Socket통신을 이용한 Connection으로 Utility가 구동되고, 작업 종료 시 네트워크 Port가 정상 종료되도록 구현하여 네트워크를 이용한 백업 시 취약한 시스템 보안적인 사항을 고려하였다.

3) 하나의 네트워크 자원을 이용하여 여러 Target

System을 백업하는 방식이 아닌, 순차 백업 방식으로 보다 빠른 백업이 가능하였다.

4) 망 별 분산백업 방식을 이용하여 네트워크 자원별 동시 백업이 가능하며, 백업시간을 단축할 수 있다.

5) TCP/IP Socket Binary Mode를 이용한 백업데이터 전송방식을 이용하여 Packet Size를 증가시켜 상용의 FTP백업 방식과 비교하여 속도개선을 이루었다.

## 참고 문헌

- [1] 박진영, 유혁, <스토리지 가상화를 활용한 VTL 백업시스템>, 고려대학교 컴퓨터학과, 2008, pp619 ~ 623
- [2] 주우철, 최홍식, <정보시스템 백업현황과 구축방법에 관한 연구>, 국민대학교 정보관리학부, pp275 ~ 280
- [3] W. C. Preston, Using SANs and NAS, O'REILLY, 2001, pp22 ~ 37, pp101 ~ 105, pp.66 ~ '88
- [4] 김규삼, <SAN과 NAS에 대하여>, 방송공학회지, 제 6권, pp101 ~ 105
- [5] SAN, NAS, DAS의 개념비교, 2010년 7월 21일, <http://gojump0713.blog.me/140111144418>, 2011-05-11
- [6] VERITAS NetBackup 5.0 System Administrator's Guide, Volume I for UNIX pp.385 ~ 398
- [7] Legato Networker Introduce pp.4 ~ 8
- [8] Feinler, Elizabeth, "Internet Protocol Transition Workbook", Network Information Center, SRI International, March 1982.
- [9] SAN을 이용한 백업구성 [http://blog.naver.com/omniback?Redi\\_rect=Log&logNo=150006543587](http://blog.naver.com/omniback?Redi_rect=Log&logNo=150006543587)
- [10] FTP Model [http://cafe.naver.com/infonuri.cafe?iframe\\_url=/ArticleRead.nhn%3Farticleid=80&](http://cafe.naver.com/infonuri.cafe?iframe_url=/ArticleRead.nhn%3Farticleid=80&)



---

저자약력

---

**김 범 재(Bum-Jae Kim)**                      **학생회원**



1996년 서일전문대학교  
전자과(전문학사),  
2007년 한국방송통신대학교  
정보통계학과  
(공학학사),  
2009년 3월 ~ 현재  
서울과학기술대학교  
전자공학 석사과정

<관심분야> DBMS, 데이터 통신, 모바일 통신,  
센서 네트워크 등

**이 준 호(Jun-Ho Lee)**                      **정회원**



1988년 연세대학교  
전자공학(공학학사),  
1990년 연세대학교  
전자공학(공학석사),  
1996년 연세대학교  
데이터통신 공학박사  
현. 서울과학기술대학교  
전자공학 교수,

<관심분야> 회로이론, 데이터 통신, 인터넷 프로  
토콜