7강. 기본 명령

- 파일의 압축
- 아카이브

백업 등의 목적을 위해 여러 파일과 디렉터리를 묶은 것

- tar(Tape ARchive) 명령

여러 개의 파일을 하나의 아카이브 파일로 만들거나 아카이브 파일에서 파일들을 추출
-만들기 : tar cvf 아카이브명 묶을파일명...
-풀기 : tar xvf 아카이브명
-내용보기 : tar tvf 아카이브명

- tar 명령 사용 예
  - tar cvf abc.tar a.txt b.txt c.txt

현재 디렉터리에 있는 a.txt, b.txt, c.txt 파일을 abc.tar 파일로 묶음

  - tar tvf abc.tar

abc.tar에 포함된 파일 및 디렉터리의 목록을 보여줌

  - tar xvf abc.tar

현재 디렉터리에 abc.tar 파일을 풀어놓음

  - tar cvf backup.tar backup

backup이 디렉터리인 경우 디렉터리의 내용을 backup.tar로 묶음

- gzip과 gunzip 명령을 이용한 파일의 압축
- 파일을 압축 / 압축 파일을 해제하는 명령
- gzip [옵션] [파일명]...
  - -n : n은 1~9 사이의 숫자, 압축률 (1:속도 빠름 ~ 9:압축률 높음, 기본 값은 6)
  - -d : 압축 해제, gunzip과 같음
- gzip으로 압축하면 원래 파일명에 .gz 확장자가 추가됨
- gunzip [옵션] [압축파일명]...

.gz, .Z, .tgz, .taz 등의 확장자를 가진 압축파일을 풀어 확장자를 제거 (tgz은 tar.gz, taz는 tar.Z을 의미)

- tar 명령을 이용한 압축과 압축 해제
- 아카이브와 압축을 동시에 실행
  - tar cvfz 압축아카이브명 파일명...
- 압축해제와 아카이브에서 추출을 동시에 실행
  - tar xvfz 압축아카이브명

- bzip2와 bunzip2 명령
- 일반적으로 gzip보다 60~70% 이상의 높은 압축률, 압축 해제 속도는 느림
- 사용방법은 gzip / gunzip과 같음 (확장자는 .bz2)

● 백업 개요

• 백업의 목적

시스템의 장애 또는 여러 가지 이유로 데이터가 손실 또는 손상되었을 때 데이터를 복구하기 위하여 백업을 수행

• 안전한 백업

-정기적으로 백업을 수행

-최근 백업한 내용만 보관하는 것은 위험

이미 손상된 데이터를 최근에 백업하였을 가능성이 있음

• 백업을 수행하는 방법

cpio, dump, tar 등의 명령어를 사용

● 백업 솔루션 결정 시 고려 사항

• 이식성 : 널리 사용되는 dump, cpio, tar 등 사용

• 자동 백업 : 자동으로 정기적 백업

• 사용의 편의성

• 원격 백업 : 원격 터미널에서 작업하려면 텍스트 기반 명령 사용

• 네트워크 백업 : 네트워크로 연결된 백업 장치에 백업

• 매체의 종류 : 안정성, 용량, 전송속도 등

● 시스템 백업의 종류

• 시초 백업(Day-zero Backup)

시스템 설치 후 사용자가 시스템을 사용하기 전에 모든 파일과 프로그램을 백업

• 완전 백업(Full Backup)

주기적으로 시스템의 모든 파일을 백업

• 변경 분 백업(Incremental Backup)

-이전의 백업 후에 변경된 파일만을 백업

-주기적인 백업 또는 프로그램의 추가나 패치 같은 특정한 이벤트 후 수행

● 시스템 백업의 수준

• 단순 백업

-먼저 완전 백업을 수행하고 이후 변경된 부분만을 백업

-개인적인 용도나 작은 규모의 사이트에서 사용하기 적합

• 수준별 백업

-중요한 업무를 다루고 있고 큰 규모의 사이트를 운영할 때 효과적

-완전 백업과 변경 분 백업(수준을 달리함) 두 가지를 사용

-적은 비용으로 백업 보장기간(backup history)을 길게 연장

-파일 시스템을 복원하는 데 드는 시간을 최소화

● 백업의 전략

• 자료 가치에 따라 다른 백업 종류 사용

• 백업 저장 매체는 번갈아 가며 사용

- 영구 보관을 위한 백업 저장 매체를 준비
- 수시로 백업 저장 매체의 상태를 확인
- 백업 저장 매체는 컴퓨터로부터 떨어진 곳에서 보관
- 백업 후에는 백업 저장 매체에 쓰기 방지 설정
- 중요한 백업 자료는 암호화

● 백업 관련 명령어
- cpio 명령

-파일을 테이프에 저장하기 위한 유틸리티
-디렉터리를 다루지 못하기 때문에 파일 목록은 find 등을 사용

- cpio 명령의 사용 방법

-생성 : cpio -o < 파일목록을 가진 파일 > 아카이브명
-추출 : cpio -i < 아카이브명

● cpio 명령 사용 예
- find 명령의 결과를 cpio의 표준 입력으로 사용
- /home 디렉터리 아래 모든 파일을 테이프 드라이브(/dev/st0)로 백업
  - find /home -print | cpio -o > /dev/st0
- find 명령의 -mtime, -newer 옵션을 사용하여 변경분 백업
  - find . -mtime -1 -name '*.bak' -print | cpio -o > /dev/st0
  - find . -newer timefile -print | cpio -o > /dev/st0

● dump 명령
- 파일 시스템 전체를 백업 가능, 복구는 restore 명령 사용
- 수준별 백업 기능 제공
  - 0 은 모든 파일을 백업
  - n 수준 백업은 더 낮은 수준의 이전 백업 이후에 생성, 변경된 파일만 백업
- dump 명령의 사용 방법
  - dump [옵션] 파일시스템 (파일시스템이 mount된 디렉터리)
  - dump [옵션] 파일명
- dump 명령의 옵션
  - -n : n이 0이면 전체 백업, 0보다 크면 더 낮은 수준의 백업 이후에 추가, 변경된 내용만 백업
  - -f file: 지정한 파일(아카이브) 또는 디바이스(/dev의 디렉터리 파일)에 백업
  - -u : 백업 정보를 기록함
- 예 1 : mydir 디렉터리를 mydump 파일로 백업
  - dump -0f mydump mydir
- 예 2 : /work에 마운트된 파일 시스템을 5 수준에서 테이프 드라이로 백업
  - dump -5u -f /dev/st0 /work

● dump 명령의 특징

• dump 명령의 장점
-수준별 백업을 제공
-여러 개의 테이프에 백업 가능, 어떤 타입의 파일도 백업 및 복구 가능
-파일의 접근 권한, 소유자, 수정시간 등의 사항도 복구됨
• dump 명령의 단점
-각 파일 시스템은 개별적으로 dump 되어야 함 (파티션마다 별도)
-NFS 파일 시스템은 dump 불가 (로컬 파일 시스템만 dump 가능)
-활동 중인 파일 시스템은 제대로 백업이 되지 않을 수 있음

● 파일 시스템
• 정보를 저장하기 위해 저장 공간에 어떻게 구성할 것인가에 관한 규칙
파일에 파일명과 경로를 부여하고 저장이나 검색을 위해 어디에 위치시켜야 하는지 등을 나타내는 방법
• 리눅스 파일 시스템의 기본적 기능
-파일과 디렉터리 개념으로 구성된 트리를 관리
-모든 것을 파일로 취급
• 파일의 종류
-정규 파일 : 일반적인 텍스트 파일이나 이진 파일
-디렉터리 파일 : 디렉터리 정보를 나타내는 파일
-특수 파일 : 주변 장치나 프로세스 간 통신에 이용되는 파일

● 파일 시스템 유형
• minix : Minix의 파일 시스템, 기본적인 파일 시스템
• xiafs : minix의 제한을 보완한 수정 버전
• msdos : FAT(File Allocation Table) 파일 시스템
• hpfs : OS/2의 파일 시스템
• isofs : ISO 기준을 따르는 표준 CD-ROM의 파일 시스템
• umsdos : msdos를 긴 파일명, 소유자, 접근 권한, 링크와 장치 파일 등을 사용할 수 있도록 확장
• nfs : 네트워크 상 컴퓨터의 파일 시스템을 공유
• sysv : System V/386, Xenix 등의 파일 시스템

● 리눅스 파일 시스템
• ext2 파일 시스템 특징
-파일 시스템 생성 시 블록 크기를 선택 가능. 1,024~4,096byte
-파일 시스템 생성 시 i-node 개수 결정 가능. 주어진 크기의 파티션에 얼마나 많은 파일을 저장할 수 있는가에 따라 결정됨
-디스크 블록을 그룹으로 분할
-정규 파일이 저장되기 전에 데이터 블록을 미리 할당 가능
-고속의 심볼릭 링크를 지원

• ext3 파일 시스템
-ext2 파일 시스템과 같은 구조
-리눅스에서 사용 가능한 파일 시스템 중 가장 신뢰할 수 있는 저널링 파일 시스템 중 하나
-저널링 : 변경을 기록하는 로그를 두어 시스템 비정상 종료 시 파일 시스템 복구를 쉽게 하는 방법
• ext4 파일 시스템
-현재 리눅스의 기본 파일 시스템
-Extents와 Online Defragmentation 지원
-1EB 파일 시스템 사이즈, 16TB 사이즈의 파일 가능
• reiser 파일 시스템
-처음으로 저널링을 지원한 리눅스 파일 시스템
-사용 중에 파일시스템의 크기 확장이 가능
-B*Tree에 기반으로 모든 파일 시스템의 객체가 하나의 B*Tree 안에 존재

● 다음 문제에 대한 정답을 고르시오.
13. 다음은 어떠한 백업 명령을 설명하는 것인가?
파일을 백업하는 것이 아니라, 파일 시스템 전체를 백업하며, 이전에 파일 시스템을 백업한 이후에 변경된 파일의 목록을 작성하여, 그 목록에 있는 파일을 새로운 백업 파일로 백업 받을 수 있는 점진적인 백업 기능을 제공한다.
① cpio ② taper
③ dump ④ rmt

14. tar 파일로 묶여 있는 sensor.tar 파일의 리스트를 보고자 할 때 사용하는 것은?
① tar cvf sensor.tar ② tar xvf sensor.tar
③ tar tvf sensor.tar ④ tar rvf sensor.tar

15. gzip 명령어에서 압축률을 높이기 위한 옵션은?
① -h ② -9
③ -t ④ -1

● 다음문제에 대한 정답을 서술하시오.
16. 시스템을 이용하면서 백업은 굉장히 중요한 작업이다. 이러한 백업 작업을 할 때 알아두어야 할 수칙들을 작성하시오.
① 자료 가치에 따라 다른 백업 전략을 취하라.
② 백업 저장 매체는 번갈아가며 사용하라.
③ 오랫동안 보관하기 위한 백업 저장 매체를 준비하자.
④ 가끔 백업 저장 매체의 상태를 확인하자.
⑤ 백업 저장 매체는 될 수 있으면 컴퓨터로부터 떨어진 곳에서 보관하자.
⑥ 백업을 한 후에는 백업 저장 매체에 쓰기 방지를 해두자.
⑦ 중요한 백업 자료에는 암호화를 해두자.

17. gzip이 bzip2와 가장 크게 다른 점을 설명하고, 그 특징을 살펴보시오.
(답) gzip은 사용자가 보유한 저장 공간을 절약하거나 백업을 할 때, 또는 데이터 전송시간을 줄이려고 데이터의 크기를 압축하는 역할을 하는 리눅스의 표준 압축(해제) 유틸리티이다. gzip을 이용한 압축은 'Lempel-Ziv' 라는 인코딩 방법을 사용하며, 텍스트 파일을 기준으로 60%에서 70% 정도의 압축 효율을 가진다. 'gzip -d'는 gunzip과 동일하며, 원본 파일의 소유주와 소유 그룹, 파일 액세스 시간, 파일 변경 시간, 퍼미션은 그대로 유지한 채로 압축을 풀게 된다.
bzip2는 자료를 압축하기 위하여 버로우스-윌러(Burrows-Wheeler) 블록 정렬 텍스트 압축 알고리즘(Block-sorting text compression algorithm)과 허프만 코딩(Huffman coding)을 사용하고 있으며, 일반적으로 gzip으로 한 것 보다 60-70% 이상의 높은 압축률을 보이며 사용방법도 동일하지만, 압축 해제 속도는 상당히 느린 단점을 가지고 있다. bzip2로 압축을 하게 되면 '.bz2'라는 확장자를 갖게 된다.