

# DataBase Forensics

2007.03.21  
bluearth in N@R

## 1. Oracle Forensics(DBMiner 사용)

### 1.1 개요

Oracle Database에는 발생한 작업에 대한 많은 유용한 정보가 Logfile에 기록되어 있으나 8i 이전 버전에서는 확인하기가 어려웠다.

현재 8i 이상에서는 DBMiner라는 Tool을 이용하여 Online혹은 Offline Redo Log File에 대한 읽기/분석/해석에 관한 작업을 SQL을 사용하여 수행 할 수 있다.

Log file의 분석 작업의 결과로 트랜잭션 별, 사용자 별, 테이블 별, 시간대 별로 Database에 가해진 변경 사항에 대해 추적을 할 수 있다. 즉, 어떤 사용자가 자료를 수정했는지 또 그 작업을 수행하기 이전의 데이터의 값은 무엇이었으며 작업 이후의 데이터의 값은 무엇이었는지를 알 수 있다.

### 1.2 Oracle Redo Log File

Oracle Redo Log File은 데이터베이스에 생긴 모든 변화를 기록하는 파일이다. Redo Log File에 저장되는 이유는, 갑작스러운 시스템 다운이나 데이터베이스 다운 등의 예기치 못한 상황이 발생하면 처리중이던 작업들을 복구하기 위해서 필요한 정보를 보관한다.

Redo Log File은 두 개 이상의 그룹으로 나누어져 있으며, 한 그룹은 Redo Log File에 저장할 공간이 부족하면 데이터를 다른 그룹의 Redo Log File로 저장한다. 하나의 Redo Log Group은 2개 이상의 동일한 Redo Log File 복사본을 가지고 있다.

오라클 서버는 하나의 데이터베이스에 최소한 두 개 이상의 Redo Log File이 있어야 한다.

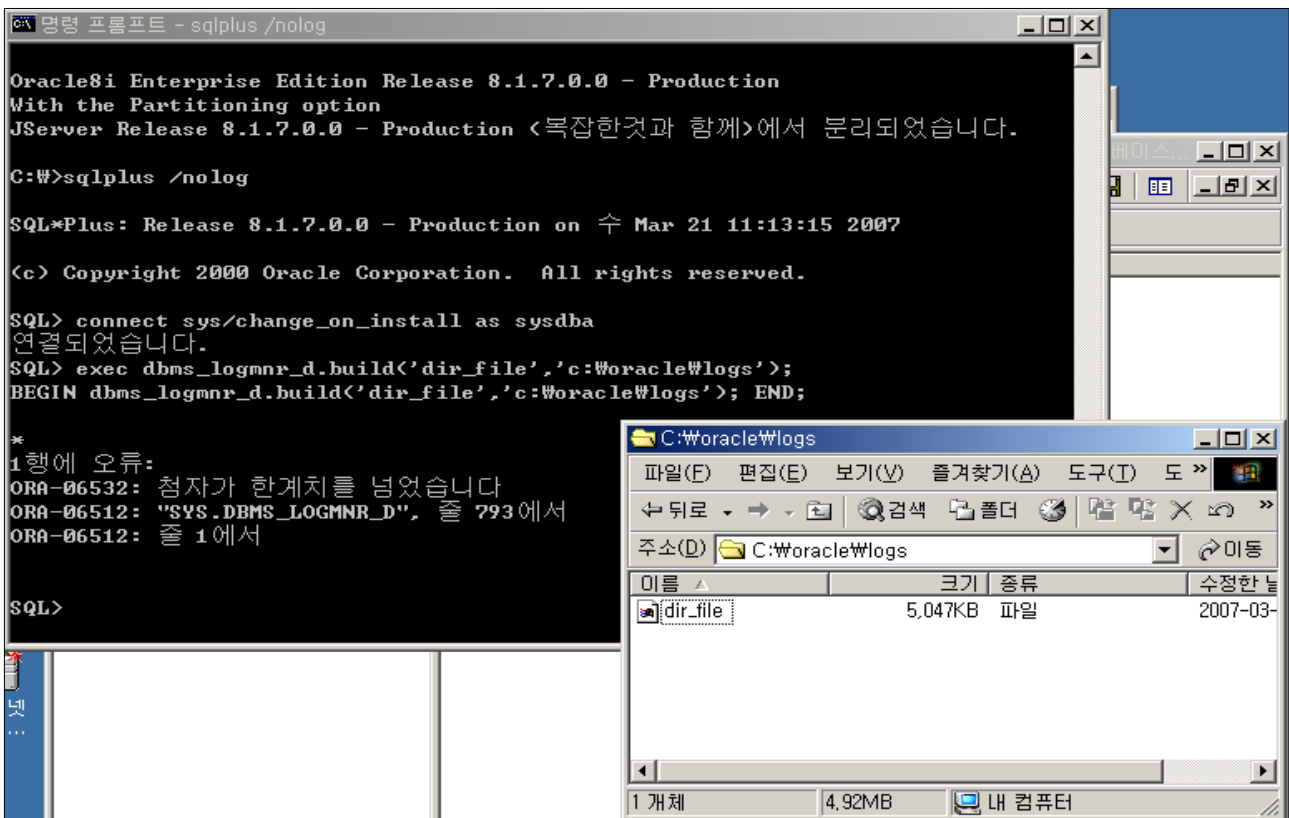
## 1.3 Oracle Dictionary File 만들기

### 1.3.1 Data Dictionary 란 ?

읽기전용 테이블 및 뷰의 집합으로 테이블 구조, 성능 등 데이터베이스 전반에 대한 정보를 제공한다. Data Dictionary에는 오라클 사용자 이름, 권한과 역할, 데이터베이스 스키마 객체(테이블, 뷰, 인덱스, 클러스터 등) 이름과 정의, 무결성 제약 조건에 관한 정보, 데이터베이스의 구조 정보 등이 저장되어져 있다.

### 1.3.2 Data Dictionary File 만들기

- Init.ora 파일 수정(win경우: c:\Woracle\Wadmin\Wbabo\Wpfile\Winit.ora)  
utl\_file\_dir = c:\Woracle\Wlogs ←----- 추가  
    **※ 반드시 설정 후 Oracle Service 중지/재시작**
- 명령어 실행  
    exec dbms\_logmnr\_d.build('dir\_file','c:\Woracle\Wlogs')  
    **※ 명령어 실행 이전에 Logminer Package 실행(1.4 참고)**



- ※ SPFILE은 이번 버전에서 ALTER SYSTEM SET 명령어를 사용하여 매개변수를 설정하는 경우 데이터베이스를 재시작하면 설정된 매개변수가 무효화 되는 문제가 있었음.  
SPFILE에 ALTER SYSTEM SET 명령어에 의해 설정된 매개변수를 저장함으로써 데이터베이스의 시작과 종료 시 재실행하는 문제를 해결함

## 1.4 Logminer Package File 실행(Test Ver Oracle 8.1.7 for win)

### 1.4.1 Oracle 8.1.5

```
sql>connect sys/change_on_install as sysdba
```

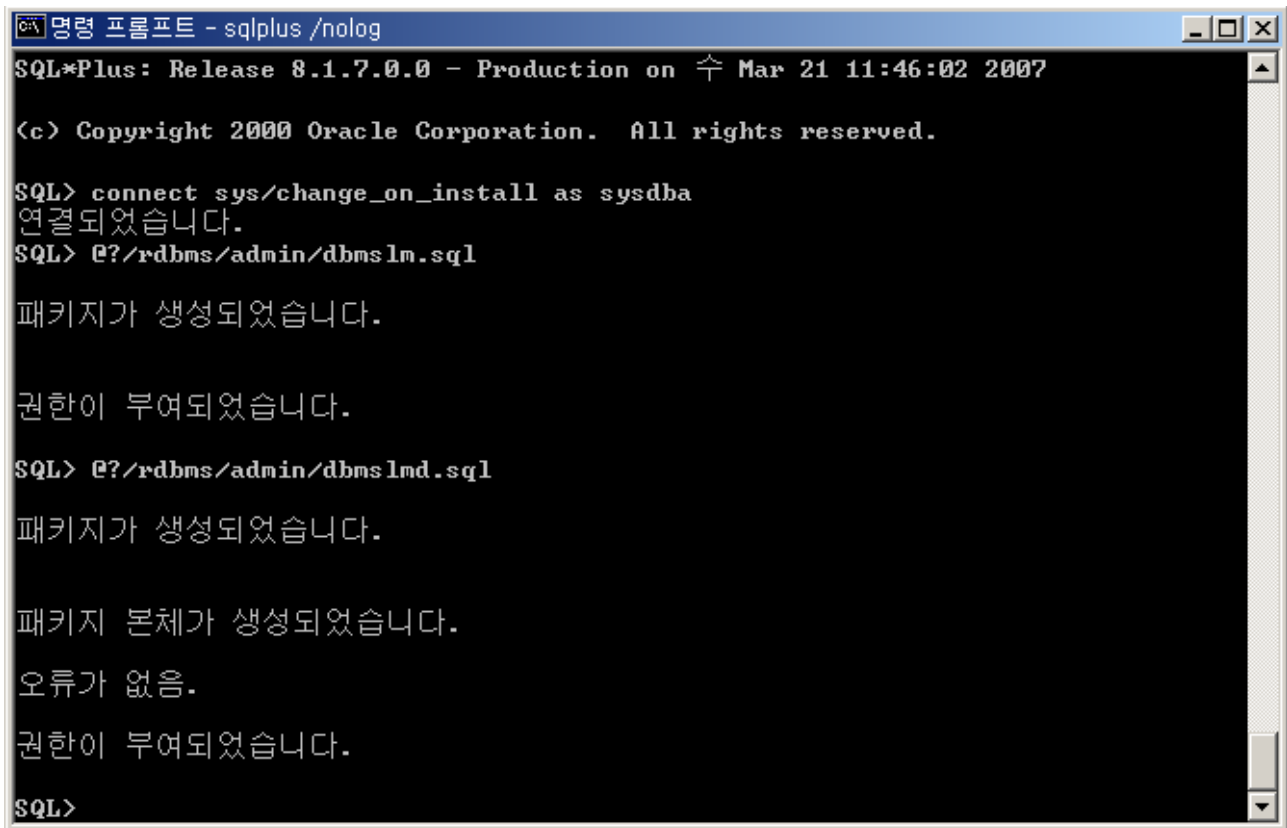
```
sql>@?/rdbms/admin/dbmslogmnr.sql
```

### 1.4.2 Oracle 8.1.6-8.1.7, Oracle 9i

```
sql>connect sys/change_on_install as sysdba
```

```
sql>@?/rdbms/admin/dbmslm.sql
```

```
sql>@?/rdbms/admin/dbmslmd.sql
```



```
명령 프롬프트 - sqlplus /nolog
SQL*Plus: Release 8.1.7.0.0 - Production on 수 Mar 21 11:46:02 2007
(c) Copyright 2000 Oracle Corporation. All rights reserved.

SQL> connect sys/change_on_install as sysdba
연결되었습니다.
SQL> @?/rdbms/admin/dbmslm.sql

패키지가 생성되었습니다.

권한이 부여되었습니다.

SQL> @?/rdbms/admin/dbmslmd.sql

패키지가 생성되었습니다.

패키지 본체가 생성되었습니다.

오류가 없음.

권한이 부여되었습니다.

SQL>
```

### 1.4.3 Dictionary File 생성 - 1.3.2 참조

- Init.ora 파일 수정(win경우: c:\Woracle\Wadmin\Wearth\Wpfile\Winit.ora)

```
utl_file_dir = c:\Woracle\Wlogs ←----- 추가
```

- 명령어 실행

```
exec dbms_logmnr_d.build('dir_file','c:\Woracle\Wlogs')
```

※ init.ora 에 추가후 지정한 디렉토리가 없을 시 미리 만들어야 함  
(여기서는 c:\Woracle\Wlogs 디렉토리)

```
C:\WINDOWS\system32\cmd.exe - sqlplus /nolog
SQL*Plus: Release 9.0.1.0.1 - Production on 수 Mar 21 15:03:21 2007
(c) Copyright 2001 Oracle Corporation. All rights reserved.

SQL> connect sys/qhdk2k as sysdba
연결되었습니다.
SQL> exec dbms_logmnr_d.build('dir_file','c:\oracle\logs');
BEGIN dbms_logmnr_d.build('dir_file','c:\oracle\logs'); END;

*
1행에 오류:
ORA-01336: 지정된 디렉터리 파일을 열 수 없음
ORA-06510: PL/SQL: 처리되지 않은 user-defined 예외 상황
ORA-06512: "SYS.DBMS_LOGMNR_D", 줄 1758에서
ORA-06512: 줄 1에서

UTL_FILE_DIR 설정 오류시

UTL_FILE_DIR 설정 성공시

SQL> exec dbms_logmnr_d.build('dir_file','c:\oracle\logs');

PL/SQL 처리가 정상적으로 완료되었습니다.

SQL>
```

## 1.5 분석을 위한 Redo Log 지정하기

Dictionary File 생성후 분석 하기 위한 Log 파일을 ADD\_LOGFILE Procedure를 이용하여 지정함.

Procedure Parameter로 New/ADDFILE/REMOVEFILE 이용

- New : 1, Removefile : 2, Addfile :3

### 1.5.1 Oracle Instance 가동(중지되어 있을시)

```
sql> startup
```

### 1.5.2 Log List 만들기

- List 생성 및 추가

```
sql>exec dbms_logmnr.add_logfile('c:\oracle\arc\redo01.log',1)
```

```
sql>exec dbms_logmnr.add_logfile('c:\oracle\arc\redo02.log',3)
```

```
sql>exec dbms_logmnr.add_logfile('c:\oracle\arc\redo03.log',3)
```

- List 삭제

```
sql>exec dbms_logmnr.add_logfile('c:\Woracle\Warc\Wredo03.log',2)
```

### 1.5.3 LogMiner 사용하기

Dictionary File 생성 및 분석할 Log를 지정하였으면 Logminer를 시작할 수 있고 분석이 가능함. 시작 시 탐색 정보의 범위를 줄이기 위해 Parameter 지정 할 수 있음.

- StartSCN(default 0) / EndSCN(0)
- StartTime(01-JAN-2007) / EndTime(01-March-2007)
- DictFileName(NULL) : 생성 했을시 사용

※ SCN : System Change Number

- Logminer 실행

```
sql>execute dbms_logmnr.start_logmnr(DictFileName=>'c:\Woracle\Wlogs\Wdir_file');
```

```
sql>execute dbms_logmnr.start_logmnr(DictFileName=>'c:\Woracle\Wlogs\Wdir_file',
```

```
StartSCN=>100, EndSCN=>200);
```

- Logminer 종료

```
sql> call dbms_logmnr.end_logmnr();
```

#### ※ LogMiner 시작 후 관련 View 정보

- V\$LOGMNR\_DICTIONARY : 사용중인 Dictionary File
- V\$LOGMNR\_PARAMETERS : Setting된 현재의 Parameter 값
- V\$LOGMNR\_LOGS : 분석되고 있는 Redo log file(또는 Archive 파일)
- V\$LOGMNR\_CONTENTS : 현재 분석되고 있는 Redo log file 내용

## 1.6 다른 Database에서 생성된 로그 파일 분석

### 1.6.1 선제 조건

A : 분석 수행할 Database 시스템(즉, Logminer 설치된 시스템)

B : 분석대상 Database 시스템(즉, 분석당할 Redo log file 있는 시스템)

- ① A와 B는 Character Set이 같아야 함.
- ② B의 Redo log 분석 위해서는 B의 Dictionary File 사용해야함
- ③ A와 B는 같은 Hardware Platform 이어야 함
- ④ Oracle 8.0 이상에서 생성된 Redo log file 이어야 함

#### ※ 전체 과정 요약

1. Oracle Package File 생성(Logminer 설치)
2. Dictionary File 생성
3. 분석 위한 Redo log 지정
4. Logminer 실행
5. 분석 내용 확인 및 복구

## 1.7 실제 예제 : Data 삭제후 Logminer로 확인 하고 삭제된 Record 복구

1) scott/tiger로 접속하여 emp table 한 Row 삭제

```
sql> connect scott/tiger
```

```
SQL> select empno, ename, job, deptno from emp;
```

EMPNO	ENAME	JOB	DEPTNO
7369	SMITH	CLERK	20
7499	ALLEN	SALESMAN	30
7521	WARD	SALESMAN	30
7566	JONES	MANAGER	20
7654	MARTIN	SALESMAN	30
7698	BLAKE	MANAGER	30
7782	CLARK	MANAGER	10
7788	SCOTT	ANALYST	20
7839	KING	PRESIDENT	10
7844	TURNER	SALESMAN	30
7876	ADAMS	CLERK	20
7900	JAMES	CLERK	30
7902	FORD	ANALYST	20
7934	MILLER	CLERK	10

14 개의 행이 선택되었습니다.

```
sql> delete from emp where deptno = 10
```

```
SQL> delete from emp where deptno = 10;
```

3 행이 삭제되었습니다.

```
sql> commit
```

```
SQL> commit;
```

커밋이 완료되었습니다.

2) 분석할 Log file 확인

```
sql> connect system/manager
```

커밋이 완료되었습니다.

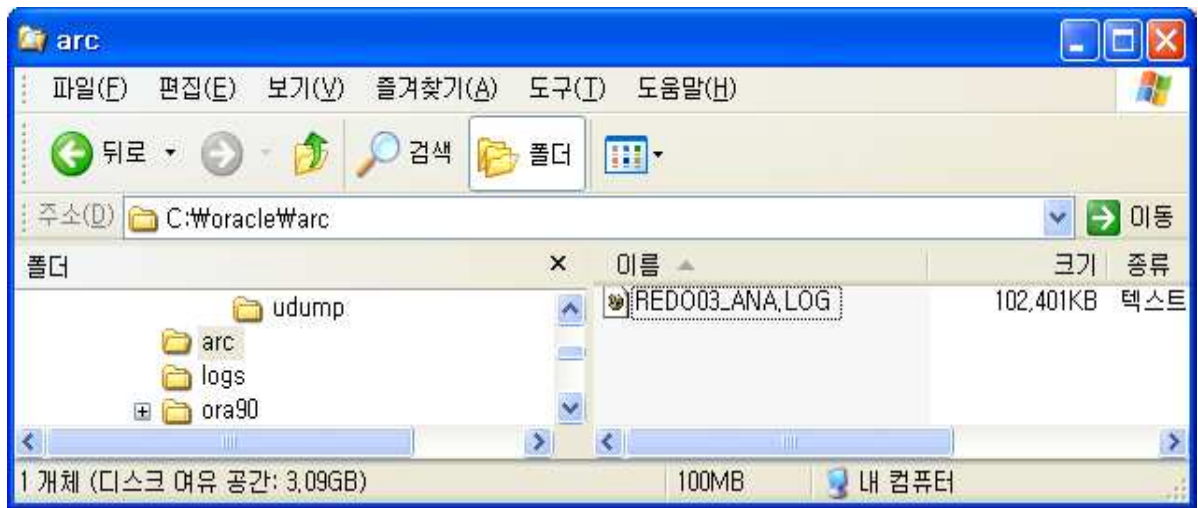
```
SQL> connect system/ql k  
연결되었습니다.
```

```
sql> select a.group#, a.sequence#, b.member
      from v$log a, v$logfile b
      where a.status = 'CURRENT' and a.group# = b.group#;
```

```
SQL> select a.group#, a.sequence#, b.member
      2 from v$log a, v$logfile b
      3 where a.status='CURRENT' and a.group#=b.group#;

  GROUP# SEQUENCE#
-----
MEMBER
-----
          1          2
C:\ORACLE\ORADATA\WEARTH\REDO01.LOG
```

※ redo03.log 파일을 c:\Woracle\Warc\redo03\_ana.log로 복사



3) Dictionary File 생성 및 delete 되었을때의 redo log file 또는 archiving file 등록

3-1) dictionary file 생성(반드시 관리자 권한으로 실행해야함)

```
sql> exec dbms_logmnr_d.build('dir_file','c:\Woracle\logs')
```

```
SQL> exec dbms_logmnr_d.build('dir_file','c:\Woracle\logs');
PL/SQL 처리가 정상적으로 완료되었습니다.
```



### 3-2) 분석할 Log 파일 등록

```
sql> exec dbms_logmnr.add_logfile('c:\Woracle\Warc\Wredo03_ana.log',1)
```

```
SQL> exec dbms_logmnr.add_logfile('c:\Woracle\Warc\Wredo03_ana.log',1);  
PL/SQL 처리가 정상적으로 완료되었습니다.
```

※ 보통의 경우 Redo log 파일은 Overwrite 되기 때문에  
Archiving File 파일 사용

### 4) Logminer 실행

```
sql>execute dbms_logmnr.start_logmnr(DictFileName=>'c:\Woracle\Wlogs\Wdir_file');
```

```
SQL> execute dbms_logmnr.start_logmnr(DictFileName=>'c:\Woracle\Wlogs\Wdir_file');  
PL/SQL 처리가 정상적으로 완료되었습니다.
```

#### 4-1) v\$logmnr\_contents view를 통한 Log file 정보 확인

```
sql> select seg_owner, seg_name, operation, sql_redo, sql_undo  
from v$logmnr_contents  
where seg_owner = 'SCOTT' and seg_name = 'EMP';
```

```
C:\WINDOWS\system32\cmd.exe - sqlplus /nolog  
SEG_NAME  
-----  
OPERATION  
-----  
SQL_REDO  
-----  
SQL_UNDO  
-----  
delete from "SCOTT"."EMP" where "EMPNO" = '7876' and "ENAME" = 'ADAMS' and "JOB"  
= 'CLERK' and "MGR" = '7788' and "HIREDATE" = TO_DATE('23-MAY-1987 00:00:00', '  
DD-MON-YYYY HH24:MI:SS') and "SAL" = '1100' and "COMM" IS NULL and "DEPTNO" = '2  
SEG_OWNER  
-----  
SEG_NAME  
-----  
OPERATION  
-----
```

※ 계정명 및 테이블 명은 반드시 대문자를 넣어줄 것!!!!

#### 4-2) 실행한 session의 정보 확인

```
sql> select to_char(timestamp,'yyyy/mm/dd hh24:mi:ss') "Time",  
           session_info from v$logmnr_contents  
           where seg_name='EMP'
```

```
C:\WINDOWS\system32\cmd.exe - sqlplus /nolog  
SEG_NAME  
-----  
OPERATION  
-----  
SQL_REDO  
-----  
SQL_UNDO  
-----  
delete from "SCOTT"."EMP" where "EMPNO" = '7876' and "ENAME" = 'ADAMS' and "JOB"  
= 'CLERK' and "MGR" = '7788' and "HIREDATE" = TO_DATE('23-MAY-1987 00:00:00', '  
DD-MON-YYYY HH24:MI:SS') and "SAL" = '1100' and "COMM" IS NULL and "DEPTNO" = '2  
SEG_OWNER  
-----  
SEG_NAME  
-----  
OPERATION  
-----
```

- delete/drop 등 내용 확인이 가능하므로 insert 등의 sql로 복구 가능함

#### 5) Logminer 종료

```
sql> call dbms_logmnr.end_logmnr();
```

※ 종료하기 전에 반드시 v\$logmnr\_contents view나 등록된  
log file를 모두 백업 받아야 함

## 예) 현재 사용중인 Redo Log File 바로 분석하기

### 1. Logminer 설정 및 실행(Dictionary File 생성 버전)

```
C:\WINDOWS\system32\cmd.exe - sqlplus /nolog
SQL> connect system/qhdk2k as sysdba
연결되었습니다.
SQL> exec dbms_logmnr.d.build('dir_file','c:\oracle\logs');
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> exec dbms_logmnr.add_logfile('c:\oracle\oradata\earth\redo01.log',1);
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> exec dbms_logmnr.add_logfile('c:\oracle\oradata\earth\redo02.log',3);
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> exec dbms_logmnr.add_logfile('c:\oracle\oradata\earth\redo03.log',3);
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> execute dbms_logmnr.start_logmnr(DictFileName=>'c:\oracle\logs\dir_file');
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> select scn, seg_owner, seg_name, operation, sql_redo, sql_undo from v$logmnr_contents where seg_owner='SCOTT' and seg_name='EMP';
```

### 2. Logminer 설정 및 실행(현재의 Dictionary File 사용)

```
C:\WINDOWS\system32\cmd.exe - sqlplus /nolog
SQL> execute dbms_logmnr.add_logfile('c:\oracle\oradata\earth\redo01.log',1);
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> execute dbms_logmnr.add_logfile('c:\oracle\oradata\earth\redo02.log',3);
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> execute dbms_logmnr.add_logfile('c:\oracle\oradata\earth\redo03.log',3);
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> execute dbms_logmnr.start_logmnr(options=>dbms_logmnr.dict_from_online_catalog);
PL/SQL 처리가 정상적으로 완료되었습니다.
SQL> select scn, timestamp, seg_owner, operation from v$logmnr_contents where seg_name='EMP';

```

SCN	TIMESTAM	SEG_OWNER	OPERATION
274612	07/03/21	SCOTT	DELETE
274612	07/03/21	SCOTT	DELETE

※ 현재 운영(사용)중인 Redo Log File은 계속 Write 됨을 유의 해야함

```

sql> execute dbms_logmnr.add_logfile('c:\Woracle\Woradata\Wearth\Wredo01.log',1);
sql> execute dbms_logmnr.add_logfile('c:\Woracle\Woradata\Wearth\Wredo02.log',3);
sql> execute dbms_logmnr.add_logfile('c:\Woracle\Woradata\Wearth\Wredo03.log',3);
sql> execute
        dbms_logmnr.start_logmnr(options=>dbms_logmnr.dict_from_online_catalog);
sql> select scn, timestamp, seg_owner, seg_name, operation
        from v$logmnr_contents;

```

### 3. Logminer 실행 결과 확인

```

EMP
DELETE
delete from "SCOTT"."EMP" where "EMPNO" = '7566' and "ENAME" = 'JONES' and "JOB"

        SCN SEG_OWNER
-----
SEG_NAME
-----
OPERATION
-----
SQL_REDO
-----
SQL_UNDO
-----

= 'MANAGER' and "MGR" = '7839' and "HIREDATE" = TO_DATE('02-APR-1981 00:00:00'
'DD-MON-YYYY HH24:MI:SS') and "SAL" = '2975' and "COMM" IS NULL and "DEPTNO" =
'20' and ROWID = 'AAAH14AABAAAO+HAAD';

        SCN SEG_OWNER
-----

```

## ※ 참고사항(Forensics 관점에서 주의 사항)

1. Logminer을 이용하여 Redo Log File 이나 Archiving File 분석시 파일의 무결성이 깨지지 않느냐 ?

결론적으로 말해서 무결성은 깨지지 않는다. Oracle은 Logminer 사용시 내부적인 Checksum을 사용하여 파일의 무결성이 깨지지 않게 하고 있다.

2. Logminer로 분석하여 복구시 순서대로 제대로 복구 할 수 있는가 ?

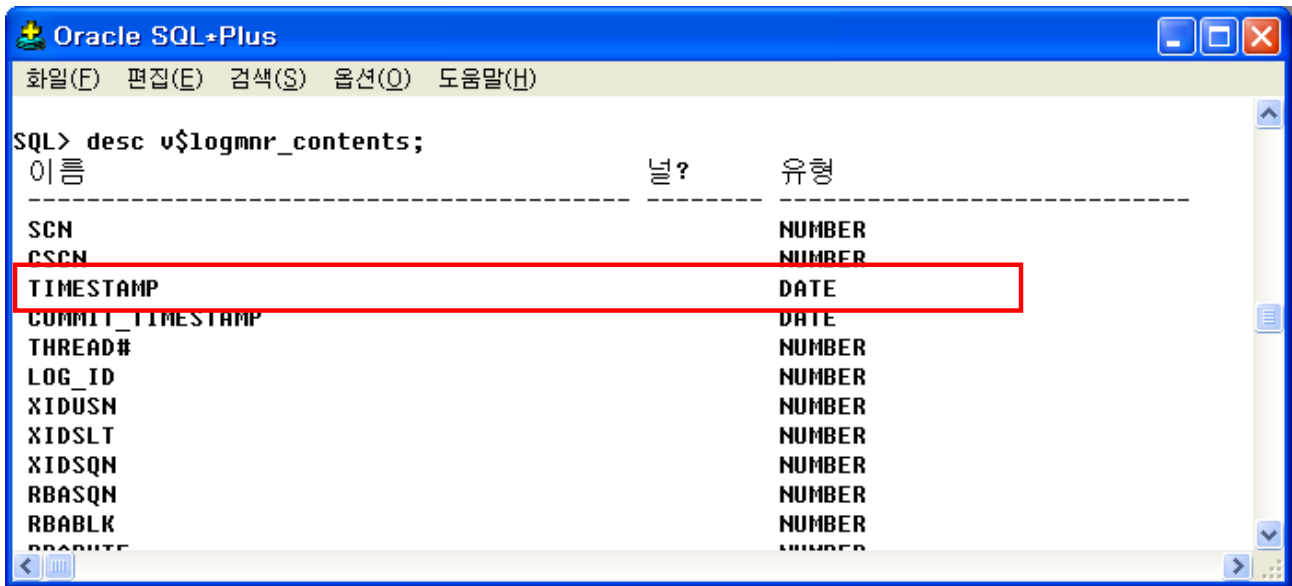
이 문제는 Oracle에서 지원하는 Time 과 관련된 문제이다.

Logminer는 Oracle 8i 이상에서 지원된다. 그러나 TIMESTAMP 타입은 Oracle 9i 이상에서 지원되는 시간 타입이다. Logminer가 8i에서 개발되었기 때문에 9i이상에서도 변수명은 TIMESTAMP 이지만 변수 타입을 확인하면 DATE 타입으로 되어 있다. 그러므로 Logminer를 이용하여 복구시에는 비슷한 시간에 진행된 쿼리가 있을시 순서가 뒤바뀔 수 있다. 여기서 또한 알아두어야 할 점은 일반적으로 쿼리 실행시 바로 Commit 되지 않고 일정 상황이 발생하거나 Commit 명령어 실행시 완료된다.

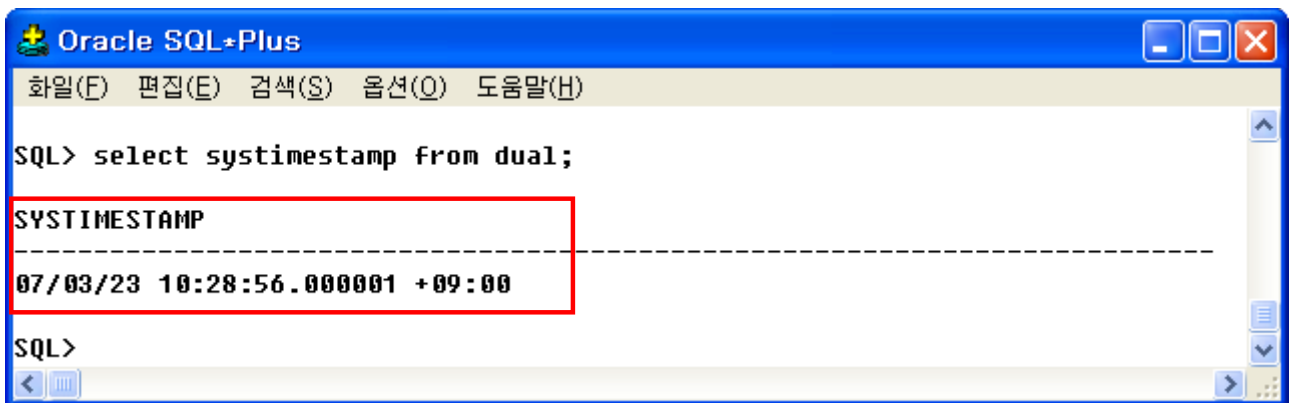
Logminer로 분석해 보면 알겠지만 동일 시간에 여러 명령어들이 실행되어진 것을 확인할 수 있을 것이다. 그러나 SCN은 유일한 값이다 그러므로 Logminer를 이용하여 복구 시에는 SCN 사용을 권고한다.

(실제적으로 REDO LOG 안의 시간은 TIMESTAMP 형식으로 저장된다)

참고1) v\$logmnr\_content의 TIMESTAMP 데이터 형식(8i, 9i)



참고2) TIMESTAMP Type은 Oracle 9i 이상에서 지원



참고3) Oracle 8i에는 TIMESTAMP 지원 하지 않음

```
SQL> select systimestamp from dual;
select systimestamp from dual
*
1행에 오류:
ORA-00904: 열명이 부적합합니다
```

※ Database Forensics이라고 하기엔 민망한 점이 많지만 아직까지 DB 포렌식 분야는 많이 알려지지 않은것 같습니다. Oracle 및 Mssql 등 DB관련 포렌식 자료 있으시는 분들 좋은 자료 공유 부탁드립니다. [To bluearth@null2root.org](mailto:To_bluearth@null2root.org)

참고자료 : [http://www.giac.org/certified\\_professionals/practicals/gcfa/0159.php](http://www.giac.org/certified_professionals/practicals/gcfa/0159.php) (GCFA 문서)