Oracle Forensics Part 7:
Using the
Oracle System Change Number
in Forensic Investigations

David Litchfield [davidl@ngssoftware.com]
24th November 2008

An NGSSoftware Insight Security Research (NISR) Publication

http://www.ngssoftware.com

Introduction

This paper will examine the internals of the Oracle System Change Number (SCN) in 10g and demonstrate how it can be used in the forensic examination of a compromised database server. It will also demonstrate how to use orablock and oratime, part of cadfile [1], a forensic toolkit for database servers, to discover when an Oracle data block was changed.

The System Change Number

The System Change Number or SCN is a number Oracle uses internally to keep track of changes made to the database server. With each change the SCN is incremented. The database's SMON background process keeps track of these SCNs and their timestamps in the SMON_SCN_TIME table. Approximately 5 days worth of SCNs are stored in this table and older entries are deleted. Though introduced in Oracle 9, under 10g this table has undergone significant changes. The table itself is created in the SMON_SCN_TO_TIME cluster and a new column of type RAW has been added. The column, called TIM_SCN_MAP, has a maximum size of 1200 bytes and can track up to 100 SCN to time mappings. As the SMON process writes to the SMON_SCN_TIME table every five minutes this allows for a new SCN to time mapping to be recorded every 3 seconds. Each 12 byte mapping contains a 32bit timestamp and the 4 byte SCN. The purpose of the remaining 4 bytes has not yet been ascertained. The timestamp is calculated using an algorithm that records the time from midnight of the 1st January 1988. The full details of this algorithm are explained in "Oracle Forensics: Part 1". The SCN_TO_TIMESTAMP function analyzes this data in this raw column to return the timestamp for the given SCN.

The dump below shows four 12 byte mappings taken from the TIM_SCN_MAP column.

```
...
...
93 D9 04 28 5C 54 05 00 00 00 95 6B
96 D9 04 28 60 54 05 00 00 00 E6 04
99 D9 04 28 CD 54 05 00 00 00 E6 04
9E D9 04 28 66 55 05 00 00 00 9D 6B
...
...
```

If we look at the first mapping we can split out the timestamp and the SCN:

```
93 D9 04 28 5C 54 05 00 -> 0x2804D993, 0x0005545C
```

Respectively in decimal these are 671406483 and 349276. If we supply 349276 as the SCN to the SCN_TO_TIMESTAMP() function we get the following:

```
SQL> SELECT SCN_TO_TIMESTAMP(349276) FROM DUAL;

SCN_TO_TIMESTAMP(349276)
-------------------------------
21-NOV-08 09.48.03.000000000 PM
```

Thus we can see this SCN correlates to 9:48:03 PM on the 21st of November 2008. Now, if we supply the timestamp, 671406483, to oratime [2], we get the same date and time:

```
C:\cadfile>oratime 671406483
21/11/2008 21:48:03
```

In Oracle, data is stored in tables and, at the file level, these tables are split across data blocks. Amongst other things each data block contains a header, a row directory and the data itself which is stored in rows. The row directory contains a list of offsets pointing to each row of data. Located at bytes 9 to 12 of the data block header is a 4 byte SCN. The SCN is updated each time the data block is written to with the value of the SCN at the time of the last committed update, insert or delete to occur on data in that block. So if an INSERT is occurs at 17:00:00 hrs but is committed at 17:01:00 hrs then the timestamp and SCN will reflect the latter time, i.e. that of the commit.

During a forensic examination of a compromised Oracle database server the SCN and its timestamp can help tell the investigator whether a block of data has been changed. This is especially useful in those cases where there is an absence of other evidence such as the redo logs or audit trail. As with all forensic examinations it's critical not to change any evidence so any investigation should take place on a cold data file and not a live data file. Orablock [1] can be used for this purpose.

Using Orablock to determine when a data block was last changed.

For the purposes of this example, let's assume that an investigator is concerned that some data in a data block may have been changed by an attacker and they want to know the time the data was last changed. The SCN for the block in question is 349280. Starting with only the object ID of the OBJ$ table, typically 18, orablock can be used to determine when a block of data was last changed. Firstly, the investigator needs to retrieve the object ID of the SMON_SCN_TO_TIME cluster, where the SMON_SCN_TIME table is located:

```
C:\cadfile>orablock

Orablock v1.0

(c) David Litchfield
(david@davidlitchfield.com)

-h (show help)
-f data_file (required)
-c column_template
-z block_size (default 8192)
-o object_id
-b block_number
-s seperator (default newline)
-a action

Actions are:
A DUMPALL
D SHOWDELETED
O DUMPNOTVIAOFFSETS
S SHOWDELETEDNOTVIAOFFSETS
C DUMPSCNS
```

```
C:\cadfile>orablock.exe -f system01.dbf -a A -o 18 -c objects.txt -s "," | findstr
/C:"SMON_SCN_TO_TIME"
563,SMON_SCN_TO_TIME,
564,SMON_SCN_TO_TIME_IDX,
```

This command passes the name of the data file for the SYSTEM tablespace "system01.dbf" and dumps the object ID and object name of all objects from the OBJ$ table (object ID 18). The column template file, "objects.txt" contains the following:

```
C:\cadfile>type objects.txt
NUMBER
NOPRINT
NOPRINT
VARCHAR2
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
NOPRINT
```

This tells orablock only to print the 1st and 4th columns as NUMBER and VARCHAR2 respectively. The output from running this command is piped to the findstr command to search for SMON_SCN_TO_TIME. This result shows that the object ID for this cluster is 563.

Using this object ID the investigator can dump SCNs and timestamps.

```
C:\cadfile>orablock.exe -f system01.dbf -a A -o 563 -c scn.txt -v
```

Note the "-v" option used in this command line. This will return the block number and file offset, etc. The column template file used in this command, scn.txt, contains the following contents:

```
C:\cadfile>type scn.txt
NOPRINT
DATE
NOPRINT
NUMBER
NOPRINT
NOPRINT
NOPRINT
NOPRINT
```

This tells orablock to print the second and fourth columns as a date and a number. After running this command we get the following output, truncated for brevity.

```
...
...
----------------------------------------------------------
Block ID = 4443, Row Offset = 1055 [41F], File offset = 36398203, Row Header =
[6C 00 08]
2008/11/21 21:45:38
348445
----------------------------------------------------------
Block ID = 4444, Row Offset = 6896 [1AF0], File offset = 36412236, Row Header =
[6C 00 08]
2008/11/21 21:45:44
348605
----------------------------------------------------------
Block ID = 4444, Row Offset = 5825 [16C1], File offset = 36411165, Row Header =
[6C 00 08]
2008/11/21 21:51:32
350591
...
...
```

Searching for an earlier SCN closest to the SCN in question, 349280, the investigator finds it to be 348605, highlighted in bold above. This means that the SCN the investigator is looking for can be found in the TIM_SCN_MAP column of this row. The block ID where this can be found is 4444. The investigator obtains this so they don't have to search through the whole data file. Editing the scn.txt file, the investigator changes it so the TIM_SCN_MAP raw column is now printed:

```
C:\cadfile>type scn.txt
NOPRINT
DATE
NOPRINT
NUMBER
NOPRINT
RAW
NOPRINT
NOPRINT
```

Now the investigator runs the following command, with the -b option to specify the block ID.

```
C:\cadfile\orablock.exe -f system01.dbf -a A -o 563 -c scn.txt -b 4444
```

The output again has been truncated for clarity:

```
2008/11/21 21:45:44
348605
1140:
0C D9 04 28 D1 51 05 00 00 00 3A 6C
...
...
93 D9 04 28 5C 54 05 00 00 00 95 6B
96 D9 04 28 60 54 05 00 00 00 E6 04
99 D9 04 28 CD 54 05 00 00 00 E6 04
9E D9 04 28 66 55 05 00 00 00 9D 6B
...
```

```
...
```

Highlighted in bold above is one of the SCN to timestamp mappings. Extracting this we find the SCN in question and can now get its timestamp using oratime:

96 D9 04 28 -> 0x2804D996 -> 671406486
60 54 05 00 -> 0x00055460 -> 349280

```
C:\cadfile>oratime 671406486
21/11/2008 21:48:06
```

Having done this the investigator now knows the time at which the data block in question changed to within a 3 second accuracy, namely 21:48:06 on the 21st of November 2008.

Recall that the SMON process keeps track of the past 5 days SCNs. Older SCNs are deleted. However, just because they have been deleted does not mean that they're not still available in the data file. When Oracle deletes data it doesn't remove it from the data file; it's just marked as deleted - see "Oracle Forensics: Part 2". Using the SHOWDELETED and SHOWDELETEDNOTVIAOFFSETS option all deleted data can be found. The DUMPSCNS option was used initially to get the SCN of the block in question.

Conclusion

This paper has shown how the Oracle SCN can be useful to forensic investigations and how orablock and oratime can be used instead of loading an Oracle data file in the database and thus preserves the evidence.

[1] Cadfile
http://www.databasesecurity.com/cadfile.zip