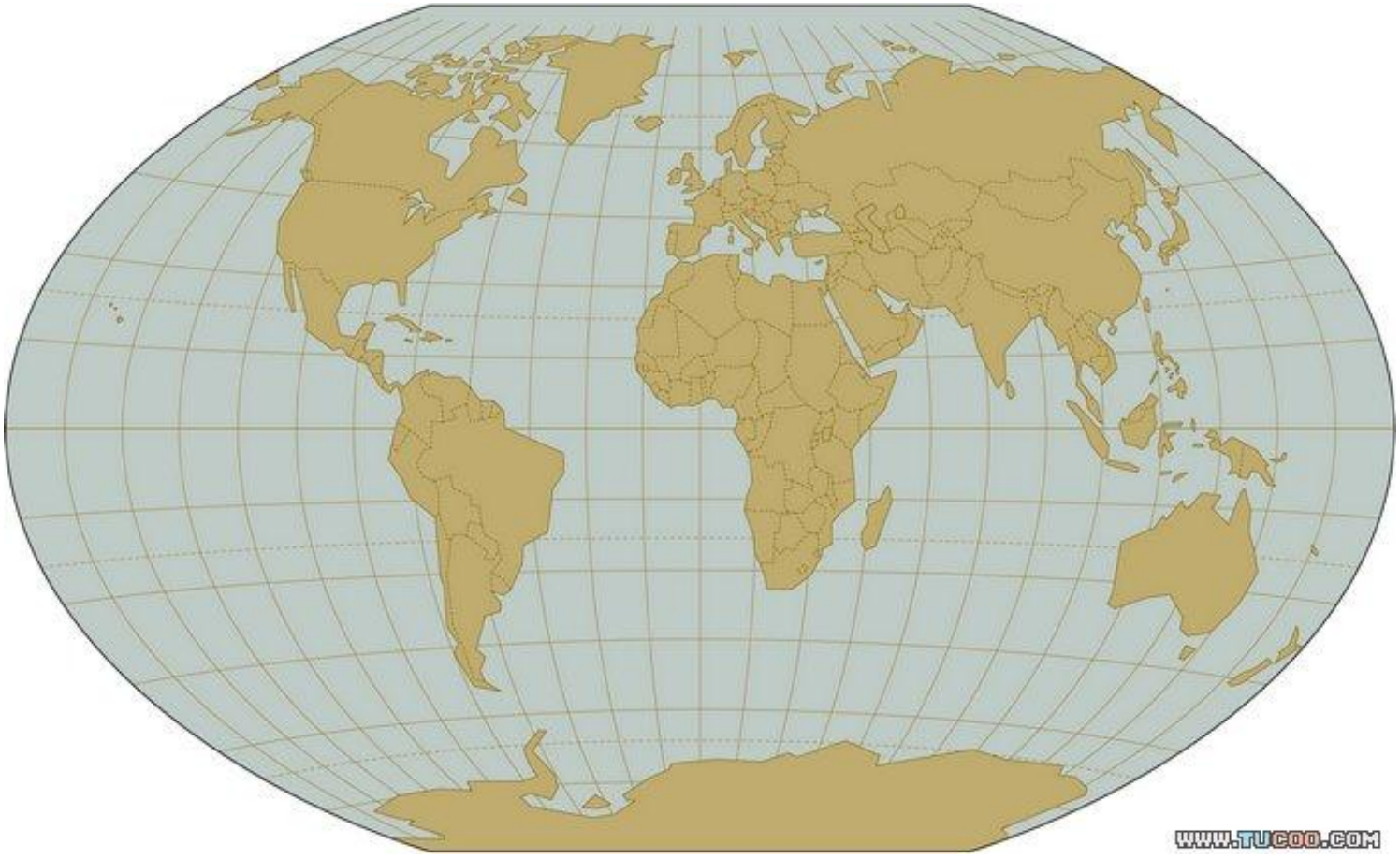
猜测的力量

刘磊（**sundog315**）

什么是猜测?



猜测

释义：推测；凭想象估计。

常作为数学用语

猜测是人们以自己已有的知识为基础，通过对问题的分析、归纳，或将其与有类似关系的特例进行比较、分析，通过判断、推理对问题结果作出的估测。猜测是数学理论的胚胎，许多伟大的数学家都是通过猜测(或猜想)发现了别人都不曾发现的真理

两个案例

- **OTN China Tour 分析案例**
- **ORA-12592: TNS:bad packet**

楔子

OTN China Tour

数据安全 – 意外篡改及防范

- 在某客户数据库系统中，遇到一则数据被恶意篡改的案例. 某用户账户余额为0元，被修改为40000元.

```
SQL> select ABS_FILE#,REL_FILE#,DATA_BLK#,DATA_OBJ#,SEG_NAME ,rs_id
  2  from v$logmnr_contents where session#=90 and seg_name='BROAD_SUBSCRB';

 ABS_FILE#  REL_FILE#  DATA_BLK#  DATA_OBJ# SEG_NAME        RS_ID
---------- ---------- ---------- ---------- --------------- ------------------------
         2         47       7600      66237 BROAD_SUBSCRB    0x00309e.00028a0a.0010
        47         47       7602      66237 BROAD_SUBSCRB    0x00309e.00028b4d.0010
SQL> select TIMESTAMP,ABS_FILE#,REL_FILE#,DATA_BLK#,DATA_OBJ#,sql_redo
  2  from v$logmnr_contents where session#=90 and seg_name='BROAD_SUBSCRB';

TIMESTAMP           ABS_FILE#  REL_FILE#  DATA_BLK#  DATA_OBJ# SQL_REDO
------------------- ---------- ---------- ---------- ---------- --------------------
2011-07-05 16:41:38          2         47       7600      66237 Unsupported
2011-07-05 16:41:54         47         47       7602      66237 Unsupported
```

数据在这里

数据安全 - Redo信息解密

- Oracle通过Redo日志进行事务重演

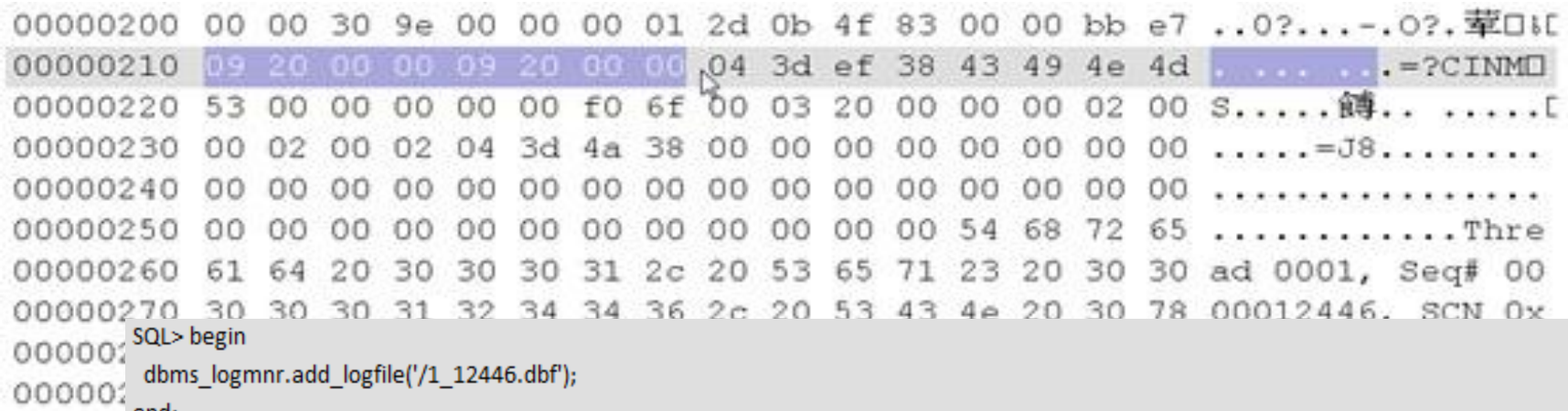
```
KDO Op code: URP row dependencies Disabled
  xtype: XA  bdba: 0x0bc01db2  hdba: 0x0900e509
itli: 3  ispac: 0  maxfr: 4863
tabn: 0 slot: 47(0x2f) flag: 0x0c lock: 0 ckix: 0
ncol: 33 nnew: 2 size: -1
col  7: [ 1]  35
col 15: [ 1]  80
>>>>这里记录了修改前的值,分别修改了第7和第15列信息(由0编号,等于第8和16列)
>>>>80就是十进制的0
CHANGE #2 TYP:2 CLS: 1 AFN:47 DBA:0x0bc01db2 SCN:0x0001.4be1efdb SEQ:  1 OP:11.5
>>>>OP:11.5 指更新行记录信息
KTB Redo
op: 0x01  ver: 0x01
op: F  xid:  0x0003.022.0019482c    uba: 0x00801542.af17.05
KDO Op code: URP row dependencies Disabled
  xtype: XA  bdba: 0x0bc01db2  hdba: 0x0900e509
itli: 3  ispac: 0  maxfr: 4863
tabn: 0 slot: 47(0x2f) flag: 0x0c lock: 3 ckix: 0
ncol: 33 nnew: 2 size: 1
col  7: [ 1]  31
col 15: [ 2]  c3 05
>>>>这里是更新后的值,c3 05就是40000
```

```
SQL> select dump(0,16) "0",dump(40000,16) "40000" from dual;

0                40000
---------------- ----------------
Typ=2 Len=1: 80 Typ=2 Len=2: c3,5
```

Why ?

猜想的依据



```
00000200 00 00 30 9e 00 00 00 01 2d 0b 4f 83 00 00 bb e7 ..0?....-.O?.茕□i[
00000210 09 20 00 00 09 20 00 00 04 3d ef 38 43 49 4e 4d . ... ..=?CINM□
00000220 53 00 00 00 00 00 f0 6f 00 03 20 00 00 00 02 00 S.....餺.. .....[
00000230 00 02 00 02 04 3d 4a 38 00 00 00 00 00 00 00 00 .....=J8........
00000240 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
00000250 00 00 00 00 00 00 00 00 00 00 00 00 54 68 72 65 ............Thre
00000260 61 64 20 30 30 30 31 2c 20 53 65 71 23 20 30 30 ad 0001, Seq# 00
00000270 30 30 30 31 32 34 34 36 2c 20 53 43 4e 20 30 78 00012446, SCN 0x
```

```
SQL> begin
 dbms_logmnr.add_logfile('/1_12446.dbf');
end;
/

PL/SQL procedure successfully completed.

SQL> begin
 dbms_logmnr.start_logmnr();
end;
/

PL/SQL procedure successfully completed.

SQL> select t.RBABLK,t.RBABYTE,t.DATA_OBJ#,t.ROW_ID,t.OPERATION,t.SQL_REDO,t.INFO from v$logmnr_contents t where t.RBABLK=166733;

166733 16 66237 AAAQXKAAAAAAAAAAAA UNSUPPORTED Unsupported Object or Data type Unsupported
```

收集信息

初次猜想

是否Oracle Logminer功能的限制？

- **1) Simple and nested abstract datatypes (ADTs)**
- **2) Collections (nested tables and VARRAYs)**
- **3) Object Refs**
- **4) Index organized tables (IOTs)**
- **5) CREATE TABLE AS SELECT of a table with a clustered key**

UNSUPPORTED Value In Sql_redo,Operation Columns Of V$Logmnr_contents [ID 282994.1]

初次猜想-失败

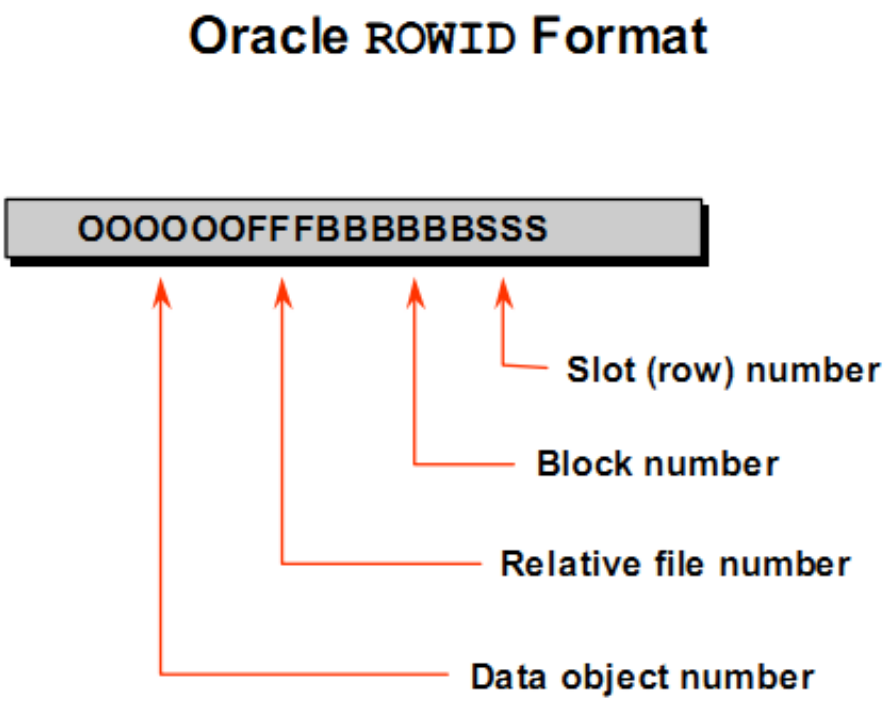
再次收集信息

```
REDO RECORD - Thread:1 RBA: 0x00309e.00028b4d.0010 LEN: 0x0114 VLD: 0x01
SCN: 0x0001.4be86fb9 SUBSCN:  1 07/05/2011 16:41:54
CHANGE #1 TYP:0 CLS:22 AFN:2 DBA:0x00801542 OBJ:0 SCN:0x0001.4be86efe SEQ:  1 OP:5.1
ktudb redo: siz: 116 spc: 7530 flg: 0x0022 seq: 0xaf17 rec: 0x05
         xid:  0x0003.022.0019482c
ktubu redo: slt: 34 rci: 4 opc: 11.1 objn: 66237 objd: 67018 tsn: 8
Undo type:  Regular undo       Undo type:  Last buffer split:  No
Tablespace Undo:  No
          0x00000000
KDO undo record:
KTB Redo
op: 0x04  ver: 0x01
op: L  itl: xid:  0x000a.01e.001a0c96 uba: 0x00800c0a.b193.29
                 flg: C---    lkc:  0     scn: 0x0001.4bb7744a
KDO Op code: URP row dependencies Disal
  xtype: XA flags: 0x00000000  bdba: 0x0bc01
itli: 3  ispac: 0  maxfr: 4863
tabn: 0 slot: 47(0x2f) flag: 0x0c lock: 0 ckix: 0
ncol: 33 nnew: 2 size: -1
col  7: [ 1]  35
col 15: [ 1]  80
```

```
CHANGE #2 TYP:2 CLS: 1 AFN:47 DBA:0x0bc01db2 OBJ:0 SCN:0x0001.4be1efdb SEQ:  1 OP:11.5
KTB Redo
op: 0x01  ver: 0x01
op: F  xid:  0x0003.022.0019482c    uba: 0x00801542.af17.05
KDO Op code: URP row dependencies Disabled
  xtype: XA flags: 0x00000000  bdba: 0x0bc01db2  hdba: 0x0900e509
itli: 3  ispac: 0  maxfr: 4863
tabn: 0 slot: 47(0x2f) flag: 0x0c lock: 3 ckix: 0
ncol: 33 nnew: 2 size: 1
col  7: [ 1]  31
col 15: [ 2]  c3 05
```

再次猜想

- ROWID : AAAQXKAAAAAAAAAAAA
- CHANGE #1 TYP:0 CLS:22 AFN:2 DBA:0x00801542 OBJ:0
- SCN:0x0001.4be86efe SEQ: 1 OP:5.1



试验验证

```
BBED> set offset 50
      OFFSET        50

BBED> m /x deb8
File: /1_12446.dbf (0)
Block: 166733        Offsets:   50 to  511        Dba:0x00000000
------------------------------------------------------------------------
deb80010 00140018 0020001d 00040001 00010074 1d6a0022 00000003 00220019
...
BBED> set offset 210
      OFFSET        210

BBED> m /x f050
Warning: contents of previous BIFILE will be lost. Proceed? (Y/N) y
File: /1_12446.dbf (0)
Block: 166733        Offsets:  210 to  511        Dba:0x00000000
------------------------------------------------------------------------
f050000c 0018001d 00040001 00020101 00000000 00000003 00220019 482c0080
...
```

遭遇CheckSUM

ERROR at line 1:
ORA-00368: checksum error in redo log block
ORA-00353: log corruption near block 166733 change 5568491448
time 07/05/2011 16:41:54
ORA-00334: archived log: '/1_12446.dbf'

```
BBED> m /x 0000          --将checksum清空为0
Warning: contents of previous BIFILE will be lost. Proceed? (Y/N) y
File: /1_12446.dbf (0)
Block: 166733        Offsets:   14 to  511        Dba:0x00000000
------------------------------------------------------------------------
00000000 01140101 00014be8 6fb90501 00160000 00020080 15424be8 6efe0001
```

失败

```
SQL> begin
  dbms_logmnr.add_logfile('/1_12446.dbf');
end;
/

PL/SQL procedure successfully completed.

SQL> begin
  dbms_logmnr.start_logmnr();
end;
/

PL/SQL procedure successfully completed.

SQL> select t.RBABLK,t.RBABYTE,t.DATA_OBJ#,t.ROW_ID,t.OPERATION,t.SQL_REDO,t.INFO from v$logmnr_contents t where t.RBABLK=166733;

166733  16  66237  AAAQXKAAAAAAAAAAAAA  UNSUPPORTED  Unsupported  Object or Data type Unsupported
```

再次猜想-失败

山穷水尽疑无路-第三次猜想

```
tabn: 0 slot: 47(0x2f) flag: 0x0c lock: 0 ckix: 0
ncol: 33 nnew: 2 size: -1
col  7: [ 1]  35
col 15: [ 1]  80

CHANGE #2 TYP:2 CLS: 1 AFN:47 DBA:0x0bc01db2 OBJ:0 SCN:0x0001.
KTB Redo
op: 0x01  ver: 0x01
op: F  xid:  0x0003.022.0019482c    uba: 0x00801542.af17.05
KDO Op code: URP row dependencies Disabled
  xtype: XA flags: 0x00000000  bdba: 0x0bc01db2  hdba: 0x0900e509
itli: 3  ispac: 0  maxfr: 4863
tabn: 0 slot: 47(0x2f) flag: 0x0c lock: 3 ckix: 0
```

ROW HEAD

This format is identical between version 8 and Oracle9*i*. This information is well documented.

Row Overhead

The values for the row flag are:

```
#define KDRHFK 0x80  Cluster Key
#define KDRHFC 0x40  Clustered table member
#define KDRHFH 0x20  Head piece of row
#define KDRHFD 0x10  Deleted row
#define KDRHFF 0x08  First data piece
#define KDRHFL 0x04  Last data piece
#define KDRHFP 0x02  First column continues from Previous
piece
#define KDRHFN 0x01  Last column continues in Next piece
```

A hexadecimal dump of a data block showing an entire row has a row flag value of "2c." This sets the bits KDRHFH, KDRHFF, KDRHFL, which would display as --H-FL-- in a logical dump. That is, the row piece contains the header, the first column, and the last column.

If the row is being updated, then the lock byte points to the ITL entry of the transaction involved.

试验验证

```
BBED> f /x 0c000000
File: /1_12446.dbf (0)
Block: 166733        Offsets:  160 to  511        Dba:0x00000000
------------------------------------------------------------------------
0c000000 002f2102 ffff0000 00010000 0007000f 35040014 80010100 0b050001

<32 bytes per line>

BBED> m /x 2c000000
Warning: contents of previous BIFILE will be lost. Proceed? (Y/N) y
File: /1_12446.dbf (0)
Block: 166733        Offsets:  160 to  511        Dba:0x00000000
------------------------------------------------------------------------
2c000000 002f2102 ffff0000 00010000 0007000f 35040014 80010100 0b050001
```

成功

166733 16 66237 AAAQXKAAvAAAB2yAAv UPDATE update "UNKNOWN"."OBJ# 66237" set "COL 8" = HEXTORAW('31'), "COL 16" = HEXTORAW('c305') where "COL 8" = HEXTORAW('35') and "COL 16" = HEXTORAW('80') and ROWID = 'AAAQXKAAvAAAB2yAAv';
Dictionary Mismatch

三次猜想-成功

结论证明

1. 创建一个表
2. 插入数据，并使一条数据产生迁移
3. 修改迁移的数据，观察是否与案例情况相同

```
create table t (a varchar2(4000));
insert into t values ('a');
insert into t values (‘b');
insert into t values (‘c');
commit;

SQL> select t.rowid from t;

ROWID
------------------
AAADZiAABAAAIDBAAA
AAADZiAABAAAIDBAAB
AAADZiAABAAAIDBAAC
```

使第三条记录产生行迁移

```
update t set t.a=lpad('a',4000,'a') where t.a='a';
update t set t.a=lpad('a',4000,'b') where t.a='b';
update t set t.a=lpad('a',4000,'c') where t.a='c';
commit;
```

```
SQL> select t.rowid from t;

ROWID
------------------
AAADZiAABAAAIDBAAA
AAADZiAABAAAIDBAAB
AAADZiAABAAAIDBAAC
```

DUMP数据块

SQL> alter system dump datafile 1 block 32961;

System altered.

tab 0, row 2, @0x3c
tl: 9 fb: --H----- lb: 0x2 cc: 0
nrid: 0x004080c2.0

SQL> alter system dump datafile 1 block 32962;

System altered.

tab 0, row 0, @0xfdc
tl: 4012 fb: ----FL-- lb: 0x1 cc: 1　　**--row flag是----FL--，也就是0x0c**
hrid: 0x004080c1.2
col 0: [4000]
63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63

结果基本一致

1. **Logminer无法挖掘出内容**

2. **Logfile Dump内容与案例相同**

```
CHANGE #1 TYP:0 CLS:22 AFN:3 DBA:0x00c0096c OBJ:4294967295
        SCN:0x0000.00043ec4 SEQ:1 OP:5.1 ENC:0 RBL:0
tabn: 0 slot: 0(0x0) flag: 0x0c lock: 0 ckix: 0
        ncol: 1 nnew: 1 size: 4001
        col  0: [4000]
        63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63


CHANGE #2 TYP:2 CLS:1 AFN:1 DBA:0x004080c2 OBJ:13922 SCN:0x0000.00043def
        SEQ:5 OP:11.5 ENC:0 RBL:0
tabn: 0 slot: 0(0x0) flag: 0x0c lock: 2 ckix: 0
        ncol: 1 nnew: 1 size: -4001
        col  0: [ 1]  64
```

开启附加日志真的可以

alter database add supplemental log data;

修改行迁移记录

Logminer…

```
SQL> select
t.RBABLK,t.RBABYTE,t.DATA_OBJ#,t.ROW_ID,t.OPERATION,t.SQL_REDO,t
.INFO from v$logmnr_contents t ;

13159  16  13922  AAADZiAABAAAIDBAAC  UPDATE  update
"UNKNOWN"."OBJ# 13922" set "COL 1" = HEXTORAW('656565') where
"COL 1" = HEXTORAW('64') and ROWID =
'AAADZiAABAAAIDBAAC';  Dictionary Mismatch
```

Why?

```
05 01 20 00 03 00 ff ff fa 00 c0 00 67 43 04 00
00 00 00 00 01 00 ff ff 10 00 14 00 18 00 20 00
1d 00 02 00 01 00 1c 00 8c 00 a4 1c 22 00 00 00
08 00 01 00 cd 00 00 00 2c 00 06 00 62 36 00 00
62 36 00 00 00 00 00 00 00 00 00 00 0b 01 01 05
00 00 cd 00 04 0d 00 00 01 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 80 00 00
00 00 00 00 c2 80 40 00 c0 80 40 00 ff 12 25 01
03 00 00 00 0c 00 00 00 00 00 01 01 fe ff 00 00
00 0c 40 00 00 00 40 00 64 12 02 01 01 04 00 00
01 00 01 00 01 00 00 00 00 10 00 00 00 00 00 00
c1 80 40 00 02 00 00 00
```

这部分的内容是不是很眼熟？看看下面这段就明白了，红色这部分内容对应的就是**hrid**：

tab 0, row 0, @0xfdc
tl: 4012 fb: ----FL-- lb: 0x1 cc: 1　　**--row flag是----FL--，也就是0x0c**
hrid: 0x004080c1.2
col 0: [4000]
63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63 63

验证结论

两个案例

- **OTN China Tour 分析案例**
- **ORA-12592: TNS:bad packet**

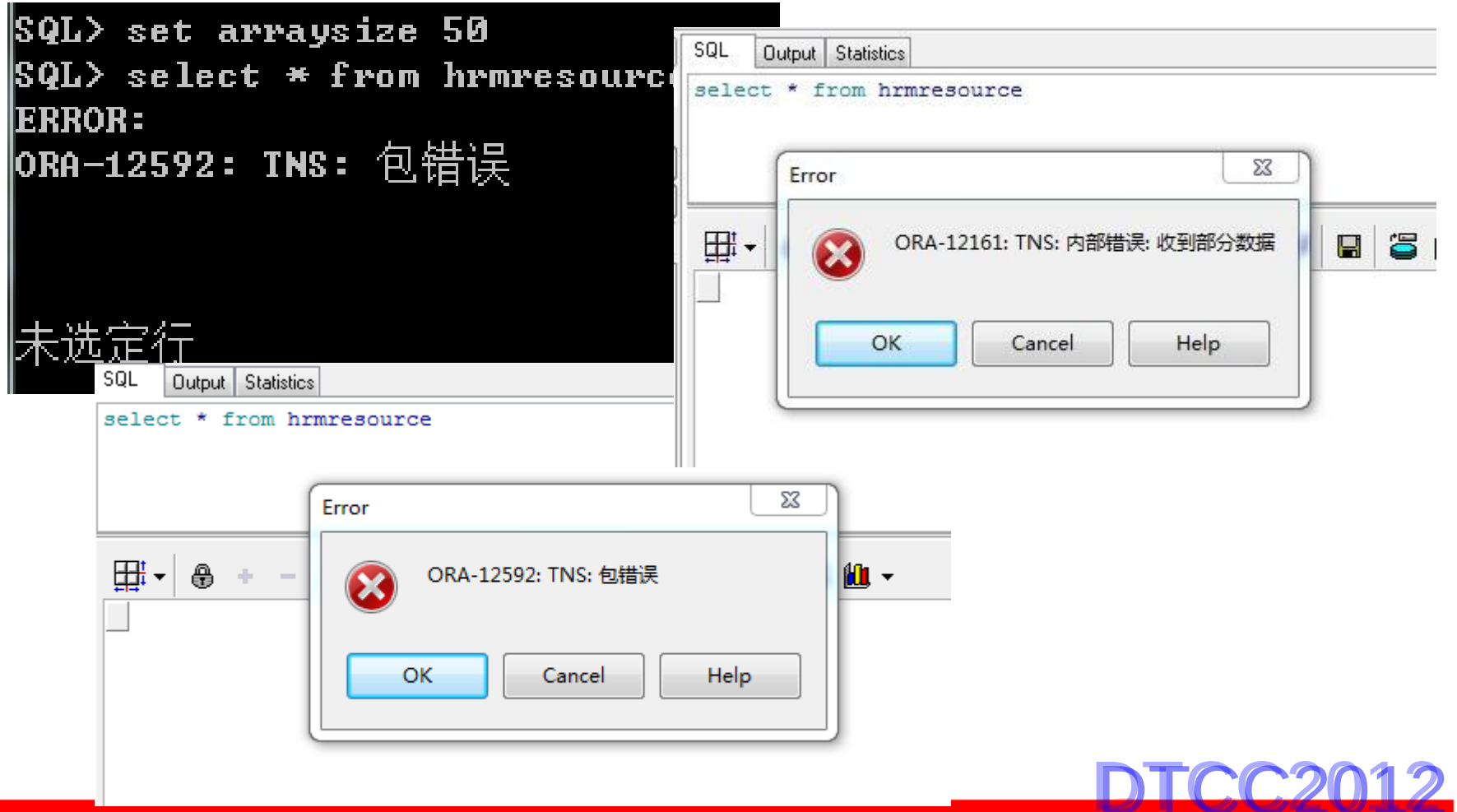
ORA-12592: TNS:bad packet

ORA-12592: TNS:bad packet

Cause: An ill-formed packet has been detected by the TNS software.

Action: For further details, turn on tracing and reexecute the operation. If error persists, contact Oracle Customer Support.

查询一个表报错


SQL> set arraysize 50
SQL> select * from hrmresourc
ERROR:
ORA-12592: TNS: 包错误
未选定行
SQL
Output
Statistics
select * from hrmresource
Error
ORA-12161: TNS: 内部错误: 收到部分数据
OK
Cancel
Help
SQL
Output
Statistics
select * from hrmresource
Error
ORA-12592: TNS: 包错误
OK
Cancel
Help

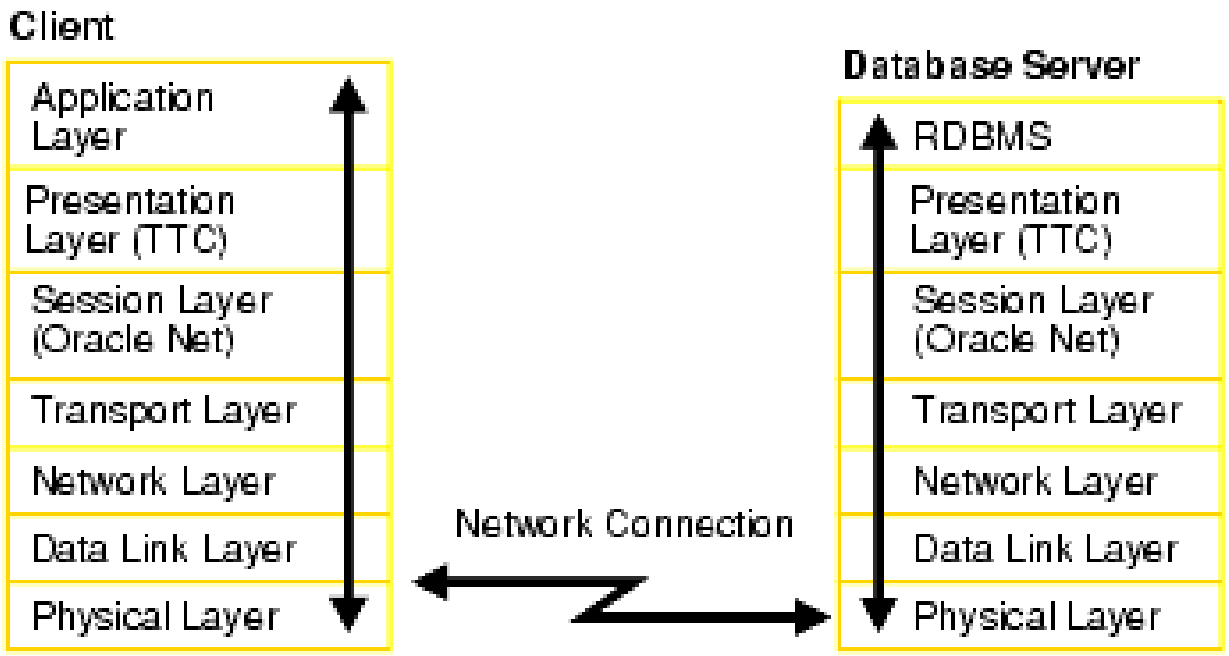
怎么办？- 收集信息

- 应用程序没有变更
- 数据库版本没有变更
- 操作系统没有变更
- 服务器没有变更
- 网络发生了变化

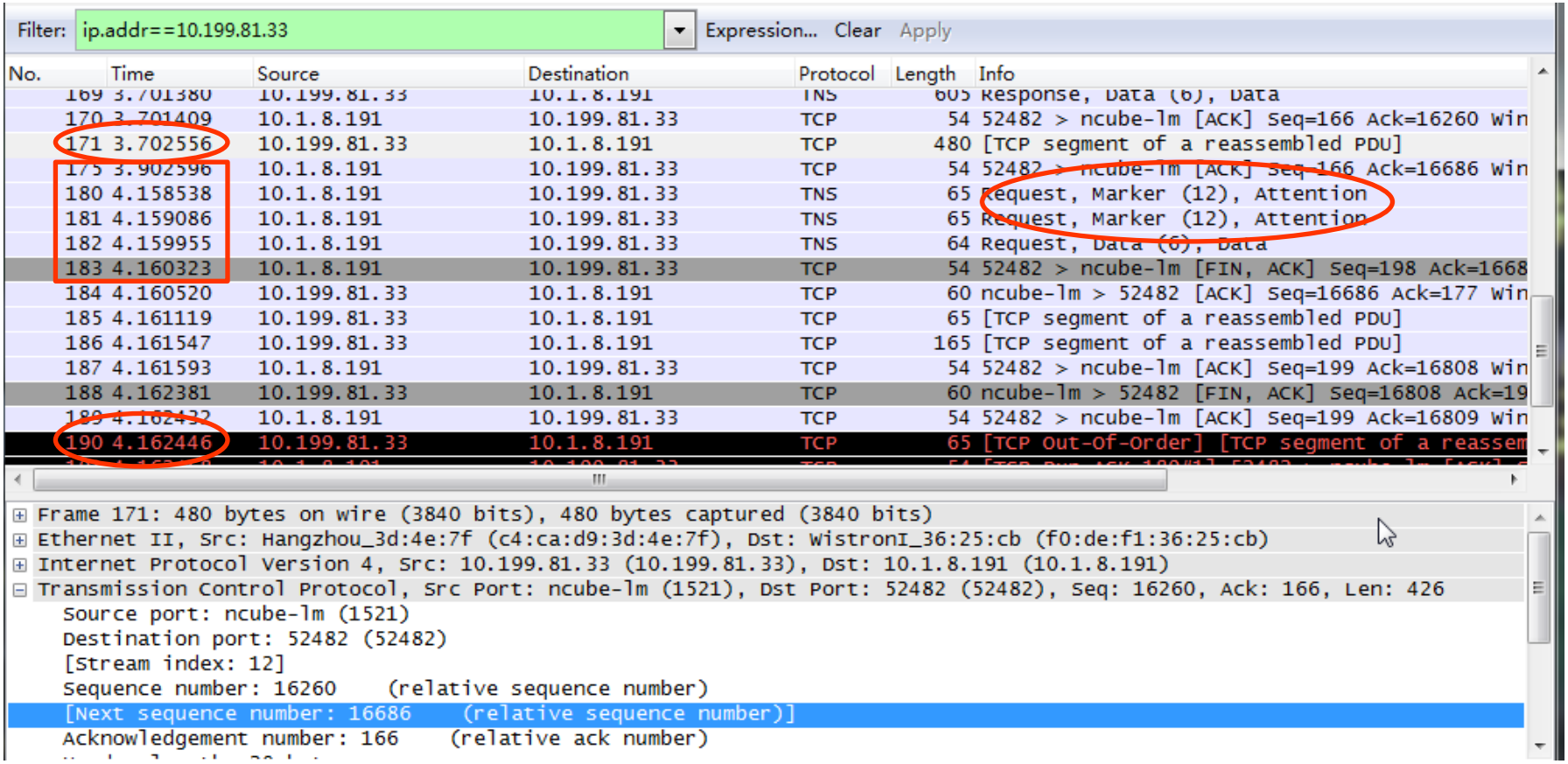
网络问题？出在哪里？- 分析问题

- **Ping**
- **Tnsping**
- **select操作**
- **Sql.net trace**
- 抓包

Oracle Net Stack & OSI



抓包分析


Filter: ip.addr==10.199.81.33
Expression... Clear Apply
No. Time Source Destination Protocol Length Info
170 3.701409 10.1.8.191 10.199.81.33 TCP 54 52482 > ncube-lm [ACK] Seq=166 Ack=16260 Win
171 3.702556 10.199.81.33 10.1.8.191 TCP 480 [TCP segment of a reassembled PDU]
180 4.158538 10.1.8.191 10.199.81.33 TNS 65 Request, Marker (12), Attention
181 4.159086 10.1.8.191 10.199.81.33 TNS 65 Request, Marker (12), Attention
182 4.159955 10.1.8.191 10.199.81.33 TNS 64 Request, Data (6), Data
183 4.160323 10.1.8.191 10.199.81.33 TCP 54 52482 > ncube-lm [FIN, ACK] Seq=198 Ack=1668
184 4.160520 10.199.81.33 10.1.8.191 TCP 60 ncube-lm > 52482 [ACK] Seq=16686 Ack=177 Win
185 4.161119 10.199.81.33 10.1.8.191 TCP 65 [TCP segment of a reassembled PDU]
186 4.161547 10.199.81.33 10.1.8.191 TCP 165 [TCP segment of a reassembled PDU]
187 4.161593 10.1.8.191 10.199.81.33 TCP 54 52482 > ncube-lm [ACK] Seq=199 Ack=16808 Win
188 4.162381 10.199.81.33 10.1.8.191 TCP 60 ncube-lm > 52482 [FIN, ACK] Seq=16808 Ack=19
190 4.162446 10.199.81.33 10.1.8.191 TCP 65 [TCP Out-Of-Order] [TCP segment of a reassem
Frame 171: 480 bytes on wire (3840 bits), 480 bytes captured (3840 bits)
Ethernet II, Src: Hangzhou_3d:4e:7f (c4:ca:d9:3d:4e:7f), Dst: WistronI_36:25:cb (f0:de:f1:36:25:cb)
Internet Protocol Version 4, Src: 10.199.81.33 (10.199.81.33), Dst: 10.1.8.191 (10.1.8.191)
Transmission Control Protocol, Src Port: ncube-lm (1521), Dst Port: 52482 (52482), Seq: 16260, Ack: 166, Len: 426
Source port: ncube-lm (1521)
Destination port: 52482 (52482)
[Stream index: 12]
Sequence number: 16260 (relative sequence number)
[Next sequence number: 16686 (relative sequence number)]
Acknowledgement number: 166 (relative ack number)

抓包分析

No.	Time	Sequence number	Next sequence number
171	3.702556	16260	16686
185	4.161119	16686	16697
186	4.161547	16697	16808
190	4.162446	16686	16697

抓病机

1.只有网络做了变更

2.小表没有问题

1.Oracle SQL*Net调整

2.修改监听端口

SUD & TDU

Session data unit (SDU)

A buffer that Oracle Net uses to place data before transmitting it across the network. Oracle Net sends the data in the buffer either when requested or when it is full.

Transport data unit (TDU)

Transparent Network Substrate Network Transport (TNS NT) layer (the layer that communicates to the Operating System protocol layer)

Configure SDU

- Server Side:
  - sqlnet.ora

    DEFAULT_SDU_SIZE=8192

  - listener.ora

    SID_LIST_listener_name=

    (SID_LIST= (SID_DESC= (SDU=8192) (SID_NAME=sales)))

  - Initialization parameter file

    DISPATCHERS="(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp))(SDU=8192))"

Configure SDU

- Client Side:
  - sqlnet.ora

    DEFAULT_SDU_SIZE=8192

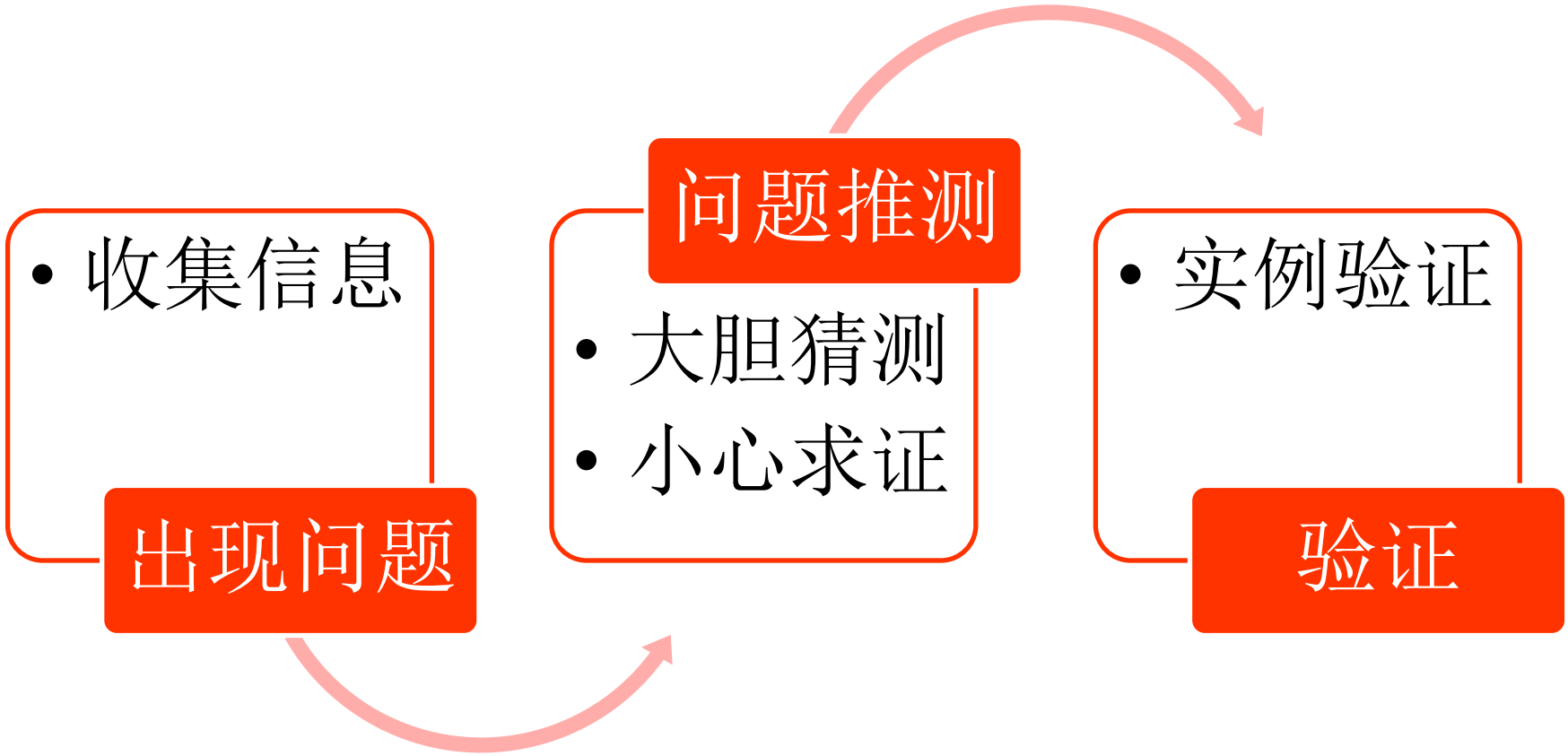
  - tnsnames.ora

    sales.us.example.com= (DESCRIPTION= (SDU=8192) (ADDRESS=(PROTOCOL=tcp)(HOST=sales-server)(PORT=1521)) (CONNECT_DATA= (SERVICE_NAME=sales.us.example.com)))

修改监听端口

将监听端口修改未非默认端口，并进行静态注册

Summary



About Me

- 某商业地产集团数据库管理员
- ACOUG Member
- Itpub Oracle专题深入讨论区版主
- 博客：sundog315.itpub.net
- 微博：www.weibo.com/sundog315

ACOUG
All China Oracle User Group
中国 Oracle 用户组

IT168旗下网站
ITPUB.NET

DTCC2012