**ORA-600 16703故障解析—tab$表被清空**

发表于 [2017 年 07 月 05 日](http://www.xifenfei.com/2017/07/ora-600-16703-recovery.html) 由 [惜分飞](http://www.xifenfei.com/author/xifenfei)

联系：**[手机(+86 13429648788) QQ(107644445)](http://www.xifenfei.com/about_xifenfei)**

[IMG_256](http://wpa.qq.com/msgrd?v=3%26uin=107644445%26site=qq%26menu=yes)

[IMG_257](http://wpa.qq.com/msgrd?v=3%26uin=107644445%26site=qq%26menu=yes)

标题：**[ORA-600 16703故障解析—tab$表被清空](http://www.xifenfei.com/2017/07/ora-600-16703-recovery.html)**

作者：**[惜分飞](http://www.xifenfei.com/)**©版权所有[未经本人同意,不得以任何形式转载,否则有进一步追究法律责任的权利.]

最近连续遇到两次数据库启动报ORA-600 16703错误,而导致数据库无法正常启动的问题

**ORA-600 16703报错**

Completed: ALTER DATABASE RECOVER  database

alter database open

Beginning crash recovery of 1 threads

 parallel recovery started with 32 processes

Started redo scan

Completed redo scan

 read 0 KB redo, 0 data blocks need recovery

Started redo application at

 Thread 1: logseq 993752, block 2, scn 14872376881763

Recovery of Online Redo Log: Thread 1 Group 6 Seq 993752 Reading mem 0

  Mem# 0: /oracle/oradata/xifenfei/redo06.log

Completed redo application of 0.00MB

Completed crash recovery at

 Thread 1: logseq 993752, block 3, scn 14872376901765

 0 data blocks read, 0 data blocks written, 0 redo k-bytes read

Tue Jul 04 23:13:29 2017

Thread 1 advanced to log sequence 993753 (thread open)

Thread 1 opened at log sequence 993753

  Current log# 7 seq# 993753 mem# 0: /oracle/oradata/xifenfei/redo07.log

Successful open of redo thread 1

MTTR advisory is disabled because FAST\_START\_MTTR\_TARGET is not set

Tue Jul 04 23:13:29 2017

SMON: enabling cache recovery

Errors in file /oracle/diag/rdbms/xifenfei/xifenfei/trace/xifenfei\_ora\_15886.trc  (incident=163595):

ORA-00600: internal error code, arguments: [16703], [1403], [20], [], [], [], [], [], [], [], [], []

Incident details in: /oracle/diag/rdbms/xifenfei/xifenfei/incident/incdir\_163595/xifenfei\_ora\_15886\_i163595.trc

Tue Jul 04 23:13:30 2017

Use ADRCI or Support Workbench to package the incident.

See Note 411.1 at My Oracle Support for error and packaging details.

Errors in file /oracle/diag/rdbms/xifenfei/xifenfei/trace/xifenfei\_ora\_15886.trc:

ORA-00704: bootstrap process failure

ORA-00704: bootstrap process failure

ORA-00600: internal error code, arguments: [16703], [1403], [20], [], [], [], [], [], [], [], [], []

Errors in file /oracle/diag/rdbms/xifenfei/xifenfei/trace/xifenfei\_ora\_15886.trc:

ORA-00704: bootstrap process failure

ORA-00704: bootstrap process failure

ORA-00600: internal error code, arguments: [16703], [1403], [20], [], [], [], [], [], [], [], [], []

Error 704 happened during db open, shutting down database

USER (ospid: 15886): terminating the instance due to error 704

Instance terminated by USER, pid = 15886

ORA-1092 signalled during: alter database open...

opiodr aborting process unknown ospid (15886) as a result of ORA-1092

这里报错比较明显ORA-600 16703,而且是在启动时bootstrap$中的对象出现该问题.

**10046分析启动过程**

=====================

select rowcnt,blkcnt,empcnt,avgspc,chncnt,avgrln,nvl(degree,1), nvl(instances,1) from tab$ where obj# = :1

END OF STMT

PARSE #140048443935120:c=0,e=390,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905161433

=====================

select blevel, leafcnt, distkey, lblkkey, dblkkey, clufac,        nvl(degree,1), nvl(instances,1) from ind$ where bo# = :1 and obj# = :2

END OF STMT

PARSE #140048443934176:c=1000,e=601,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905162088

=====================

PARSING IN CURSOR #140048443933232 len=70 dep=1 uid=0 oct=3 lid=0 tim=1499185905162444 hv=3377894161 ad='84f13d70' sqlid='32d4jrb4pd4sj'

select charsetid, charsetform from col$  where obj# = :1 and col# = :2

END OF STMT

PARSE #140048443933232:c=0,e=294,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905162443

=====================

PARSING IN CURSOR #140048443932288 len=52 dep=1 uid=0 oct=3 lid=0 tim=1499185905247020 hv=429618617 ad='84f0f2d0' sqlid='4krwuz0ctqxdt'

select ctime, mtime, stime from obj$ where obj# = :1

END OF STMT

PARSE #140048443932288:c=0,e=549,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905247019

BINDS #140048443932288:

select blevel, leafcnt, distkey, lblkkey, dblkkey, clufac,        nvl(degree,1), nvl(instances,1) from ind$ where bo# = :1 and obj# = :2

END OF STMT

PARSE #140048443934176:c=1000,e=601,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905162088

=====================

PARSING IN CURSOR #140048443933232 len=70 dep=1 uid=0 oct=3 lid=0 tim=1499185905162444 hv=3377894161 ad='84f13d70' sqlid='32d4jrb4pd4sj'

select charsetid, charsetform from col$  where obj# = :1 and col# = :2

END OF STMT

PARSE #140048443933232:c=0,e=294,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905162443

=====================

PARSING IN CURSOR #140048443932288 len=52 dep=1 uid=0 oct=3 lid=0 tim=1499185905247020 hv=429618617 ad='84f0f2d0' sqlid='4krwuz0ctqxdt'

select ctime, mtime, stime from obj$ where obj# = :1

END OF STMT

PARSE #140048443932288:c=0,e=549,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905247019

BINDS #140048443932288:

 Bind#0

  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00

  oacflg=00 fl2=0001 frm=00 csi=00 siz=24 off=0

  kxsbbbfp=7f5f91b87bd0  bln=22  avl=02  flg=05

  value=20

EXEC #140048443932288:c=2000,e=2686,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=1218588913,tim=1499185905249810

WAIT #140048443932288: nam='db file sequential read' ela= 6205 file#=1 block#=337 blocks=1 obj#=36 tim=1499185905256132

WAIT #140048443932288: nam='db file sequential read' ela= 3739 file#=1 block#=338 blocks=1 obj#=36 tim=1499185905266704

WAIT #140048443932288: nam='db file sequential read' ela= 4966 file#=1 block#=241 blocks=1 obj#=18 tim=1499185905271761

FETCH #140048443932288:c=0,e=21976,p=3,cr=3,cu=0,mis=0,r=1,dep=1,og=4,plh=1218588913,tim=1499185905271820

STAT #140048443932288 id=1 cnt=1 pid=0 pos=1 obj=18 op='TABLE ACCESS BY INDEX ROWID OBJ$ (cr=3 pr=3 pw=0 time=21993 us)'

STAT #140048443932288 id=2 cnt=1 pid=1 pos=1 obj=36 op='INDEX RANGE SCAN I\_OBJ1 (cr=2 pr=2 pw=0 time=16923 us)'

CLOSE #140048443932288:c=0,e=63,dep=1,type=0,tim=1499185905271941

BINDS #140048443935120:

 Bind#0

  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00

  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0

  kxsbbbfp=7f5f91c07de8  bln=22  avl=02  flg=05

  value=20

EXEC #140048443935120:c=1000,e=795,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=2970138452,tim=1499185905272802

WAIT #140048443935120: nam='db file sequential read' ela= 3197 file#=1 block#=169 blocks=1 obj#=3 tim=1499185905276069

WAIT #140048443935120: nam='db file sequential read' ela= 3459 file#=1 block#=170 blocks=1 obj#=3 tim=1499185905404084

WAIT #140048443935120: nam='db file sequential read' ela= 6358 file#=1 block#=145 blocks=1 obj#=4 tim=1499185905410548

FETCH #140048443935120:c=999,e=137805,p=3,cr=3,cu=0,mis=0,r=0,dep=1,og=4,plh=2970138452,tim=1499185905410635

STAT #140048443935120 id=1 cnt=0 pid=0 pos=1 obj=4 op='TABLE ACCESS CLUSTER TAB$ (cr=3 pr=3 pw=0 time=137810 us)'

STAT #140048443935120 id=2 cnt=1 pid=1 pos=1 obj=3 op='INDEX UNIQUE SCAN I\_OBJ# (cr=2 pr=2 pw=0 time=131330 us)'

\*\*\* 2017-07-05 00:31:46.094

Incident 176395 created, dump file: /oracle/diag/rdbms/xifenfei/xifenfei/incident/incdir\_176395/xifenfei\_ora\_51261\_i176395.trc

ORA-00600: internal error code, arguments: [16703], [1403], [20], [], [], [], [], [], [], [], [], []

报错信息明显,由于select rowcnt,blkcnt,empcnt,avgspc,chncnt,avgrln,nvl(degree,1), nvl(instances,1) from tab$ where obj# = :1无法正常执行.从而出现了ORA-600 16703的错误,更加直接一点的解释就是obj#=20的对象在tab$中找不到记录，从而出现此类报错.和官方解释ORA-600 TAB$和obj$不匹配一致.

**分析system文件**

通过dul等工具分析system文件发现tab$表记录为空

Data UnLoader: 11.2.0.1.5 - Internal Only - on Wed Jul 05 01:28:53 2017

with 64-bit io functions and the decompression option

Copyright (c) 1994 2017 Bernard van Duijnen All rights reserved.

 Strictly Oracle Internal Use Only

Found db\_id = 1334610369

Found db\_name = xifenfei

DUL> unload table TAB$( OBJ# number, DATAOBJ# number,

  2      TS# number, FILE# number, BLOCK# number,

  3      BOBJ# number, TAB# number, COLS number, CLUCOLS number,

  4      PCTFREE$ ignore, PCTUSED$ ignore, INITRANS ignore, MAXTRANS ignore,

  5      FLAGS ignore, AUDIT$ ignore, ROWCNT ignore, BLKCNT ignore,

  6      EMPCNT ignore, AVGSPC ignore, CHNCNT ignore, AVGRLN ignore,

  7      AVGSPC\_FLB ignore, FLBCNT ignore,

  8      ANALYZETIME ignore, SAMPLESIZE ignore,

  9      DEGREE ignore, INSTANCES ignore,

 10      INTCOLS ignore, KERNELCOLS number, PROPERTY number)

 11      cluster  C\_OBJ#(OBJ#)

 12      storage ( tablespace 0 segobjno 2 tabno 1 file 1 block 144);

. unloading table                      TAB$       0 rows unloaded

DUL> unload table OBJ$( OBJ# number, DATAOBJ# number, OWNER# number,

  2      NAME clean varchar2(30), NAMESPACE ignore, SUBNAME clean varchar2(30),

  3      TYPE# number, CTIME ignore, MTIME ignore, STIME ignore,

  4      STATUS ignore, REMOTEOWNER ignore, LINKNAME ignore,

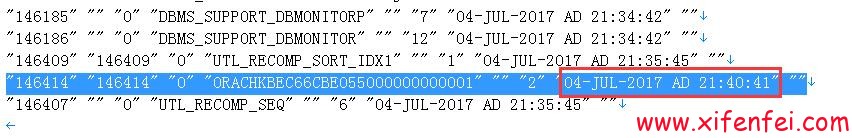
  5      FLAGS ignore, OID$ hexraw)

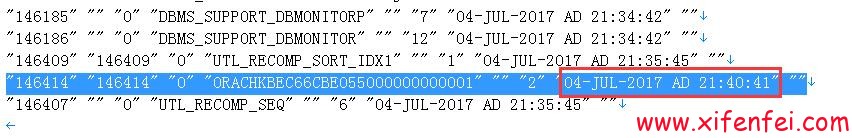
  6      storage ( tablespace 0 segobjno 18 file 1 block 240);

. unloading table                      OBJ$   89804 rows unloaded

DUL>

发现在obj$中有创建表ORACHKBEC66CBE055000000000001(ORACHK+16进制24位)的一个表名字

[](http://www.xifenfei.com/2017/07/ora-600-16703-recovery.html)

[](http://www.xifenfei.com/2017/07/ora-600-16703-recovery.html)

该表用途通过分析数据库日志发现

create table ORACHKBEC66CBE055000000000001 tablespace system as select \* from sys.tab$;

也就是说,这个orachk的表是用来备份tab$的,然后进一步发现有delete from tab$.至此基本上分析清楚,tab$表备份到ORACHK表中,然后delete tab$表数据.实现数据库破坏以及难以恢复的效果.有点类似[plsql dev引起的数据库被黑勒索比特币实现原理分析和解决方案](http://www.xifenfei.com/2016/11/plsql-dev-hacker-bitcoin.html)的破坏案例

通过最近案例的分析,我们已经具备了恢复这种破坏的能力,实现业务数据0丢失的恢复,如果需要请联系我们,提供技术支持

**Phone:13429648788    Q Q:107644445**

[IMG_260](http://wpa.qq.com/msgrd?v=3%26uin=107644445%26site=qq%26menu=yes)

**E-Mail:[dba@xifenfei.com](mailto:dba@xifenfei.com)**

由于接触的case都存在不完整行,目前该种破坏源头未分析情况,请尽可能保护好现场,防止二次破坏

[IMG_261](file:///C:\\Users\\xians\\AppData\\Local\\Temp\\enhtmlclip\\临沂公安人口库恢复ora-600%2016703问题20171010.docx)

[IMG_262](file:///C:\\Users\\xians\\AppData\\Local\\Temp\\enhtmlclip\\山东省公安厅治安人口库病毒问题分析20170915.docx)

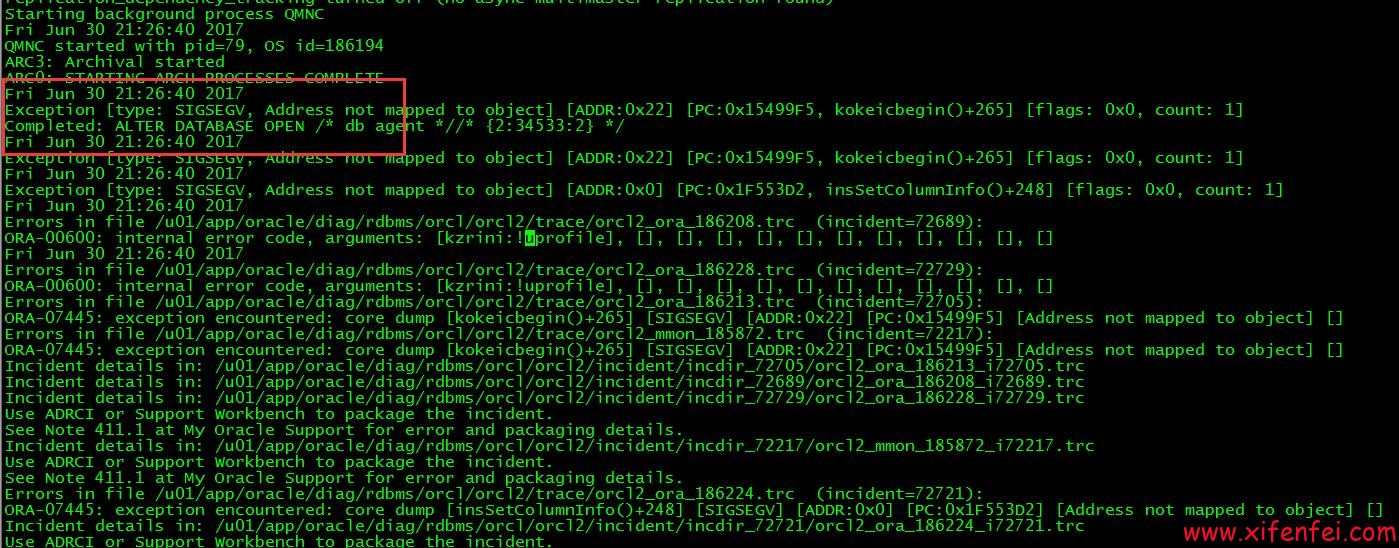
来自 <<http://www.xifenfei.com/2017/07/ora-600-16703-recovery.html>>

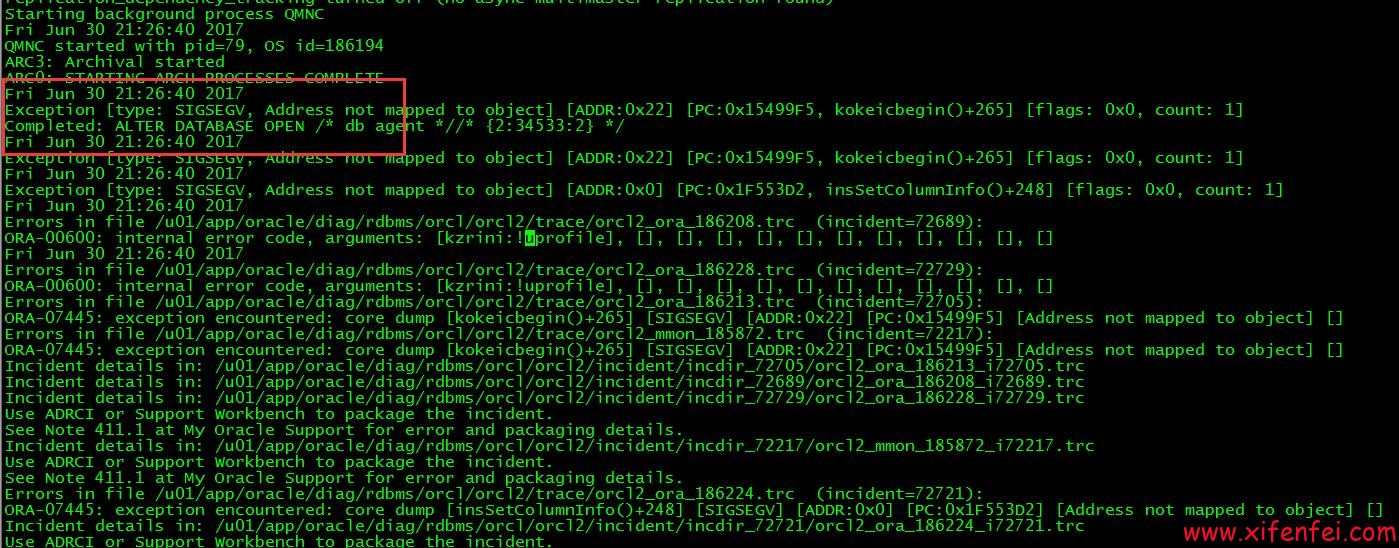
不光是在linux平台，我们在其他平台上也发现了此类问题。希望大家注意自己下载的安装介质…………

继续上篇的tab$被清空([ORA-600 16703故障解析—tab$表被清空](http://www.xifenfei.com/2017/07/ora-600-16703-recovery.html)),导致数据库启动异常的case

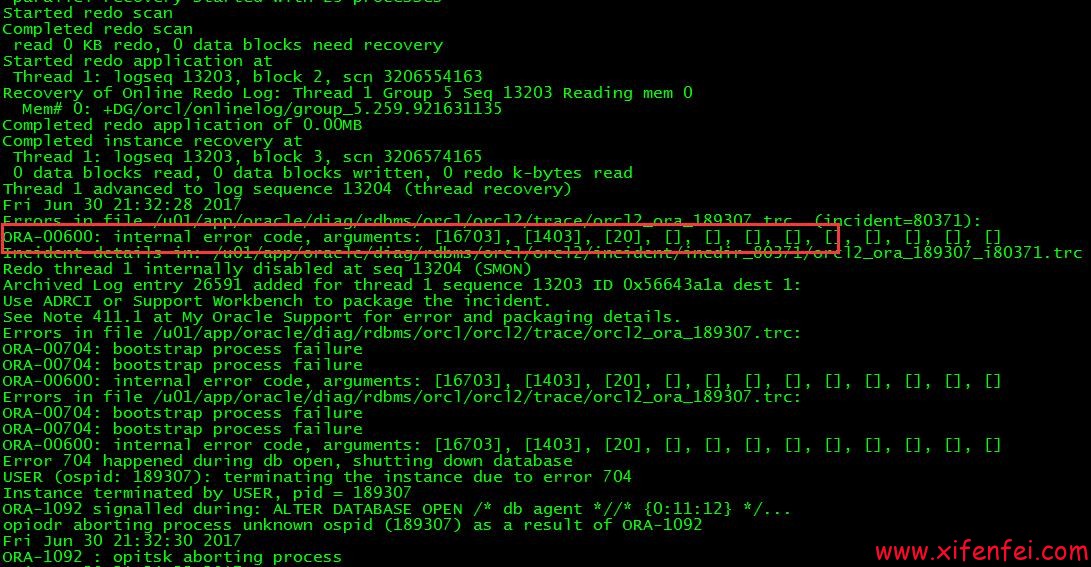
**数据库日志分析**

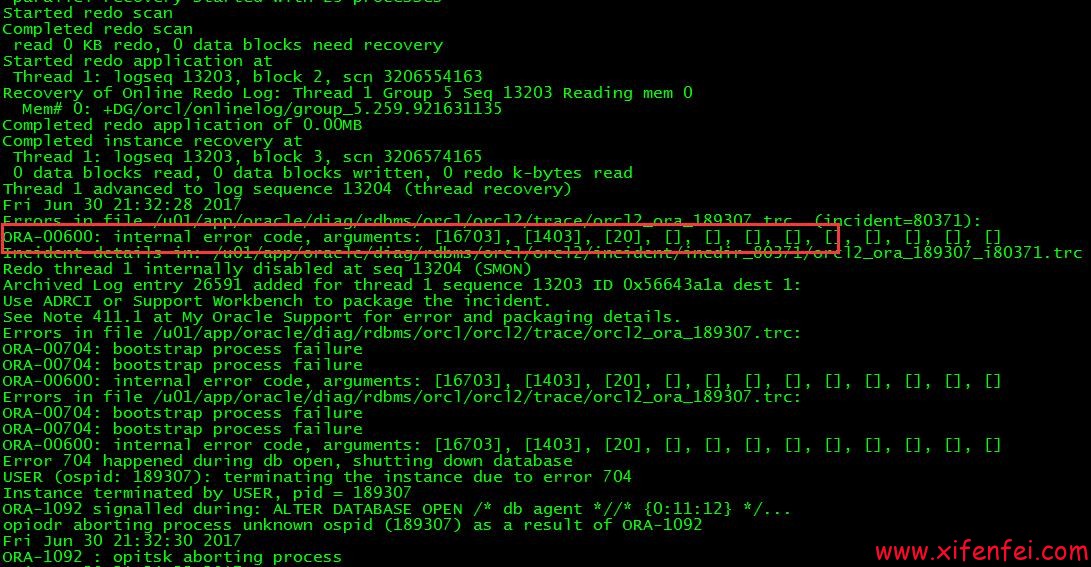
数据库open成功同时报ORA-7445 kokeicbegin和ORA-600 kzrini:!uprofile错误

[](http://www.xifenfei.com/?p=7156)

[](http://www.xifenfei.com/?p=7156)

再次启动数据库直接报ORA-600 16703错误

[](http://www.xifenfei.com/?p=7156)

[](http://www.xifenfei.com/?p=7156)

这里有一个其他现象,就是数据库在open成功的同时(同一秒内),开始报异常.重启之后直接报

ORA-00704: bootstrap process failure

ORA-00704: bootstrap process failure

ORA-00600: internal error code, arguments: [16703], [1403], [20], [], [], [], [], [], [], [], [], []

根据10046分析结果

=====================

select rowcnt,blkcnt,empcnt,avgspc,chncnt,avgrln,nvl(degree,1), nvl(instances,1) from tab$ where obj# = :1

END OF STMT

PARSE #140048443935120:c=0,e=390,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905161433

=====================

select blevel, leafcnt, distkey, lblkkey, dblkkey, clufac,        nvl(degree,1), nvl(instances,1) from ind$ where bo# = :1 and obj# = :2

END OF STMT

PARSE #140048443934176:c=1000,e=601,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905162088

=====================

PARSING IN CURSOR #140048443933232 len=70 dep=1 uid=0 oct=3 lid=0 tim=1499185905162444 hv=3377894161 ad='84f13d70' sqlid='32d4jrb4pd4sj'

select charsetid, charsetform from col$  where obj# = :1 and col# = :2

END OF STMT

PARSE #140048443933232:c=0,e=294,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905162443

=====================

PARSING IN CURSOR #140048443932288 len=52 dep=1 uid=0 oct=3 lid=0 tim=1499185905247020 hv=429618617 ad='84f0f2d0' sqlid='4krwuz0ctqxdt'

select ctime, mtime, stime from obj$ where obj# = :1

END OF STMT

PARSE #140048443932288:c=0,e=549,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905247019

[BI](http://www.itpub.net/tree/index_339/)NDS #140048443932288:

select blevel, leafcnt, distkey, lblkkey, dblkkey, clufac,        nvl(degree,1), nvl(instances,1) from ind$ where bo# = :1 and obj# = :2

END OF STMT

PARSE #140048443934176:c=1000,e=601,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905162088

=====================

PARSING IN CURSOR #140048443933232 len=70 dep=1 uid=0 oct=3 lid=0 tim=1499185905162444 hv=3377894161 ad='84f13d70' sqlid='32d4jrb4pd4sj'

select charsetid, charsetform from col$  where obj# = :1 and col# = :2

END OF STMT

PARSE #140048443933232:c=0,e=294,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905162443

=====================

PARSING IN CURSOR #140048443932288 len=52 dep=1 uid=0 oct=3 lid=0 tim=1499185905247020 hv=429618617 ad='84f0f2d0' sqlid='4krwuz0ctqxdt'

select ctime, mtime, stime from obj$ where obj# = :1

END OF STMT

PARSE #140048443932288:c=0,e=549,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1499185905247019

BINDS #140048443932288:

Bind#0

  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00

  oacflg=00 fl2=0001 frm=00 csi=00 siz=24 off=0

  kxsbbbfp=7f5f91b87bd0  bln=22  avl=02  flg=05

  value=20

EXEC #140048443932288:c=2000,e=2686,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=1218588913,tim=1499185905249810

WAIT #140048443932288: nam='db file sequential read' ela= 6205 file#=1 block#=337 blocks=1 obj#=36 tim=1499185905256132

WAIT #140048443932288: nam='db file sequential read' ela= 3739 file#=1 block#=338 blocks=1 obj#=36 tim=1499185905266704

WAIT #140048443932288: nam='db file sequential read' ela= 4966 file#=1 block#=241 blocks=1 obj#=18 tim=1499185905271761

FETCH #140048443932288:c=0,e=21976,p=3,cr=3,cu=0,mis=0,r=1,dep=1,og=4,plh=1218588913,tim=1499185905271820

STAT #140048443932288 id=1 cnt=1 pid=0 pos=1 obj=18 op='TABLE ACCESS BY INDEX ROWID OBJ$ (cr=3 pr=3 pw=0 time=21993 us)'

STAT #140048443932288 id=2 cnt=1 pid=1 pos=1 obj=36 op='INDEX RANGE SCAN I\_OBJ1 (cr=2 pr=2 pw=0 time=16923 us)'

CLOSE #140048443932288:c=0,e=63,dep=1,type=0,tim=1499185905271941

BINDS #140048443935120:

Bind#0

  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00

  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0

  kxsbbbfp=7f5f91c07de8  bln=22  avl=02  flg=05

  value=20

EXEC #140048443935120:c=1000,e=795,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=2970138452,tim=1499185905272802

WAIT #140048443935120: nam='db file sequential read' ela= 3197 file#=1 block#=169 blocks=1 obj#=3 tim=1499185905276069

WAIT #140048443935120: nam='db file sequential read' ela= 3459 file#=1 block#=170 blocks=1 obj#=3 tim=1499185905404084

WAIT #140048443935120: nam='db file sequential read' ela= 6358 file#=1 block#=145 blocks=1 obj#=4 tim=1499185905410548

FETCH #140048443935120:c=999,e=137805,p=3,cr=3,cu=0,mis=0,r=0,dep=1,og=4,plh=2970138452,tim=1499185905410635

STAT #140048443935120 id=1 cnt=0 pid=0 pos=1 obj=4 op='TABLE ACCESS CLUSTER TAB$ (cr=3 pr=3 pw=0 time=137810 us)'

STAT #140048443935120 id=2 cnt=1 pid=1 pos=1 obj=3 op='INDEX UNIQUE SCAN I\_OBJ# (cr=2 pr=2 pw=0 time=131330 us)'

\*\*\* 2017-07-05 00:31:46.094

Incident 176395 created, dump file: /[oracle](http://www.itpub.net/tree/index_1/)/diag/rdbms/orcl/orcl2/incident/incdir\_176395/orcl\_ora\_51261\_i176395.trc

ORA-00600: internal error code, arguments: [16703], [1403], [20], [], [], [], [], [], [], [], [], []

以及以往恢复经验和mos,基本上可以确定是由于tab$和obj$记录不匹配导致该问题.而且是obj#=20的记录在tab$和obj$中不匹配.

**分析tab$和obj$记录**

Data UnLoader: 11.2.0.1.5 - Internal Only - on Wed Jul 05 01:28:53 2017

with 64-[bi](http://www.itpub.net/tree/index_339/)t io functions and the decompression option

Copyright (c) 1994 2017 Bernard van Duijnen All rights reserved.

Strictly [Oracle](http://www.itpub.net/tree/index_1/) Internal Use Only

Found db\_id = 1334610369

Found db\_name = orcl

DUL> unload table TAB$( OBJ# number, DATAOBJ# number,

  2      TS# number, FILE# number, BLOCK# number,

  3      BOBJ# number, TAB# number, COLS number, CLUCOLS number,

  4      PCTFREE$ ignore, PCTUSED$ ignore, INITRANS ignore, MAXTRANS ignore,

  5      FLAGS ignore, AUDIT$ ignore, ROWCNT ignore, BLKCNT ignore,

  6      EMPCNT ignore, AVGSPC ignore, CHNCNT ignore, AVGRLN ignore,

  7      AVGSPC\_FLB ignore, FLBCNT ignore,

  8      ANALYZETIME ignore, SAMPLESIZE ignore,

  9      DEGREE ignore, INSTANCES ignore,

10      INTCOLS ignore, KERNELCOLS number, PROPERTY number)

11      cluster  C\_OBJ#(OBJ#)

12      storage ( tablespace 0 segobjno 2 tabno 1 file 1 block 144);

. unloading table                      TAB$       0 rows unloaded

DUL> unload table OBJ$( OBJ# number, DATAOBJ# number, OWNER# number,

  2      NAME clean varchar2(30), NAMESPACE ignore, SUBNAME clean varchar2(30),

  3      TYPE# number, CTIME ignore, MTIME ignore, STIME ignore,

  4      STATUS ignore, REMOTEOWNER ignore, LINKNAME ignore,

  5      FLAGS ignore, OID$ hexraw)

  6      storage ( tablespace 0 segobjno 18 file 1 block 240);

. unloading table                      OBJ$   89804 rows unloaded

DUL>

这里可以明确表示tab$无记录,obj$有记录,从而启动的过程出现ORA-600 16703错误可以很好解释.由于数据库启动成功和报错几乎同时进行，以及上面看到的tab$记录不存在的情况，初步怀疑是有startup触发器清空tab$表记录

**工具分析触发器表trigger$**

[](http://www.xifenfei.com/?p=7156)

[](http://www.xifenfei.com/?p=7156)

这里果然看到一个after startup on database的触发器,名字为DBMS\_SUPPORT\_DBMONITOR,而它调用的是DBMS\_SUPPORT\_DBMONITORP存储.

**工具分析pl/sql表source$**

[](http://www.xifenfei.com/?p=7156)

[](http://www.xifenfei.com/?p=7156)

对wraped加密的程序进行解密

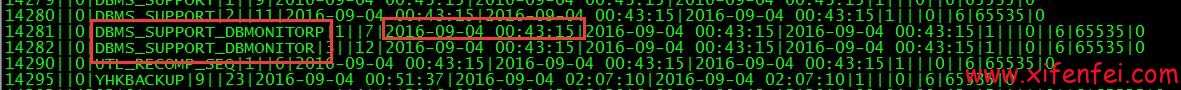
[](http://www.xifenfei.com/?p=7156)

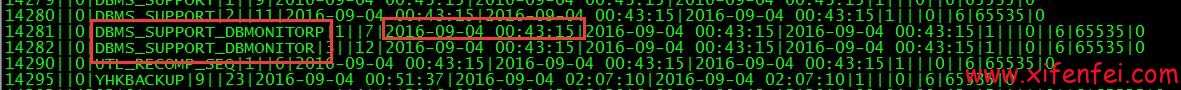
[](http://www.xifenfei.com/?p=7156)

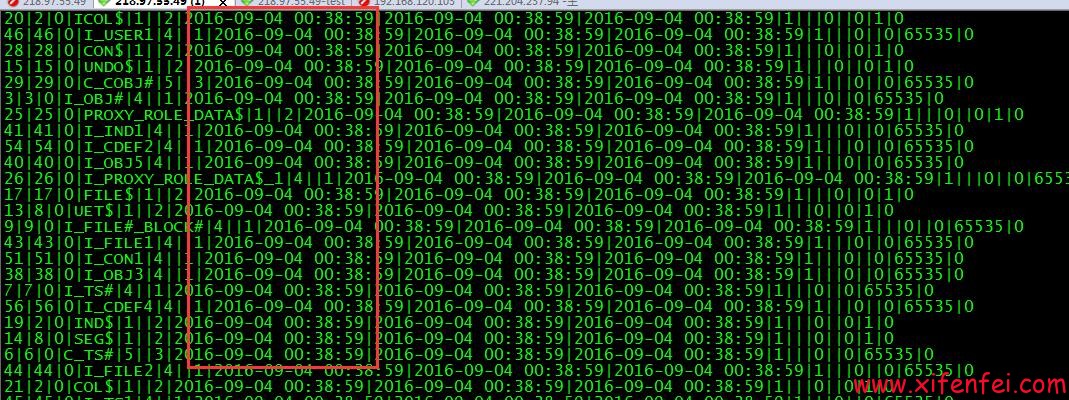
这里可以明确的看到DBMS\_SUPPORT\_DBMONITORP存储过程备份tab$表到orachk中然后delete tab$表,现在已经明确是由于DBMS\_SUPPORT\_DBMONITOR触发器在数据库重启之后开始执行调用DBMS\_SUPPORT\_DBMONITORP程序,如果判断数据库创建时间大于等于300天,便干掉tab$表,实现数据库破坏.

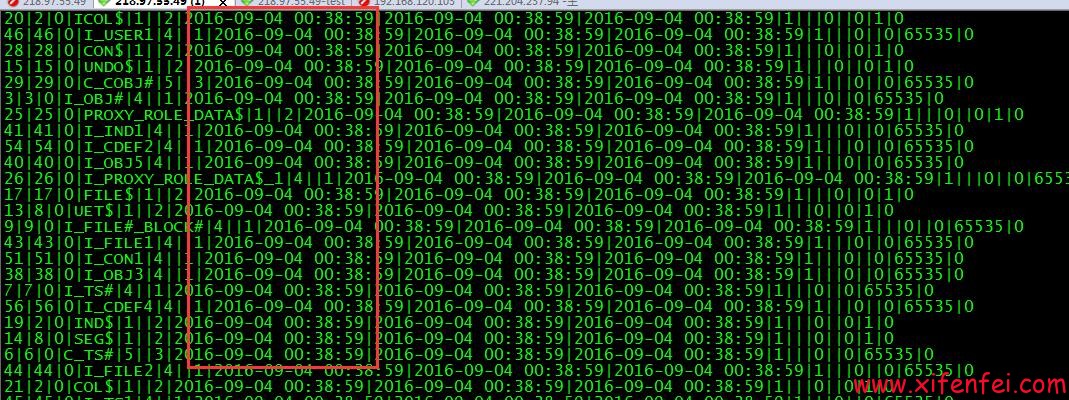
**查找DBMS\_SUPPORT\_DBMONITOR等程序源头**

分析相关程序创建时间,通过obj$表的ctime和name来判断

[](http://www.xifenfei.com/?p=7156)

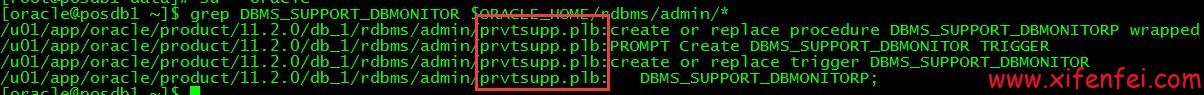
[](http://www.xifenfei.com/?p=7156)

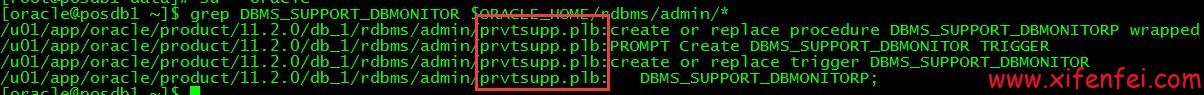
[](http://www.xifenfei.com/?p=7156)

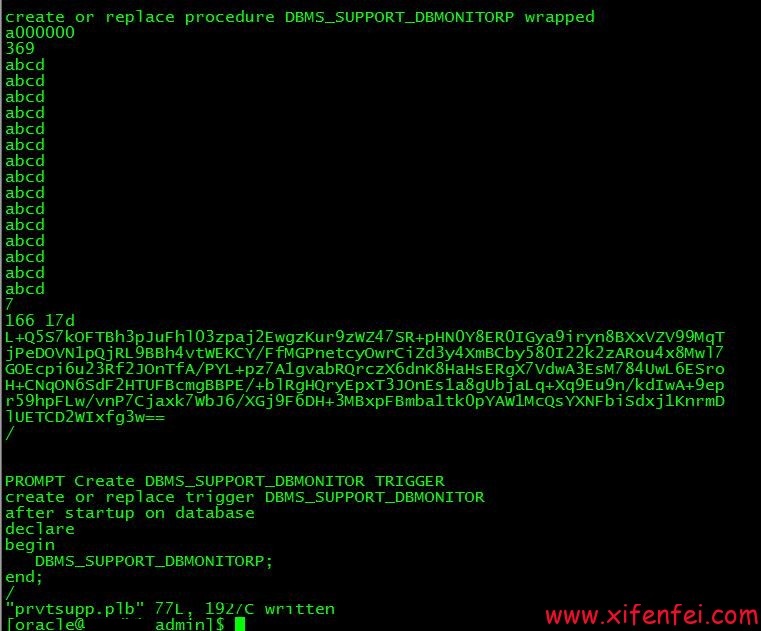
[](http://www.xifenfei.com/?p=7156)

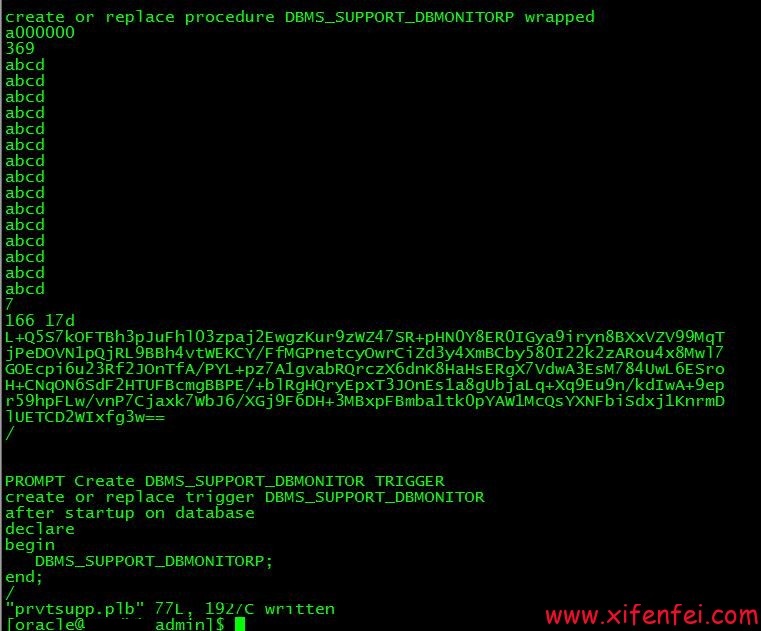
这里可以看出来DBMS\_SUPPORT\_DBMONITOR和DBMS\_SUPPORT\_DBMONITORP的创建时间基本上和数据库核心对象的创建时间相差无几,我们可以大概排除掉,是由于pl sql等工具连接数据库导致该发问题(类似:[plsql dev引起的数据库被黑勒索比特币实现原理分析和解决方案](http://www.xifenfei.com/2016/11/plsql-dev-hacker-bitcoin.html)),很可能是在dbca创建库的过程中就已经带有了DBMS\_SUPPORT\_DBMONITOR等程序,如果这样那很可能是由于数据库的安装介质被破坏导致该问题.

**分析DBMS\_SUPPORT\_DBMONITOR来源**

[](http://www.xifenfei.com/?p=7156)

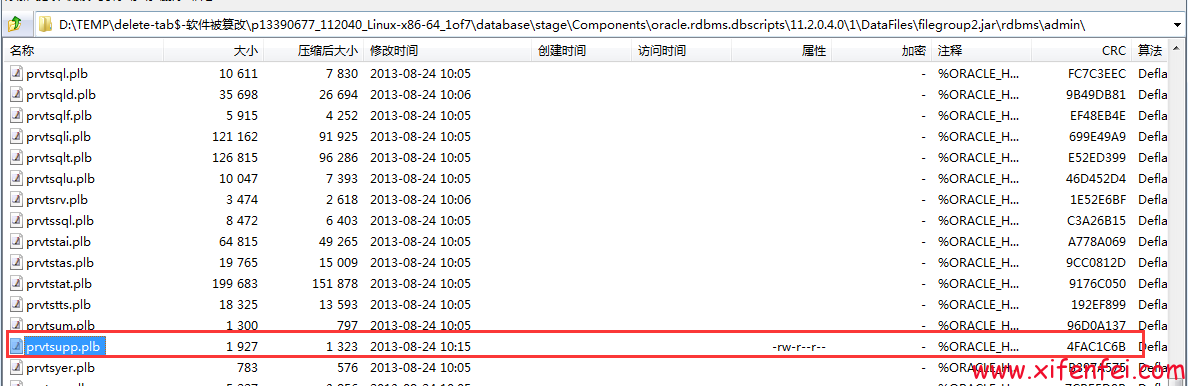
[](http://www.xifenfei.com/?p=7156)

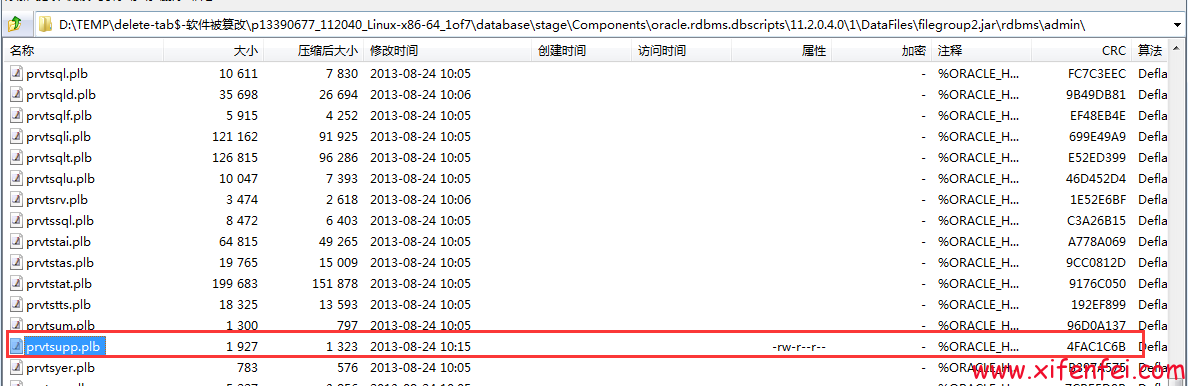
[](http://www.xifenfei.com/?p=7156)

[](http://www.xifenfei.com/?p=7156)

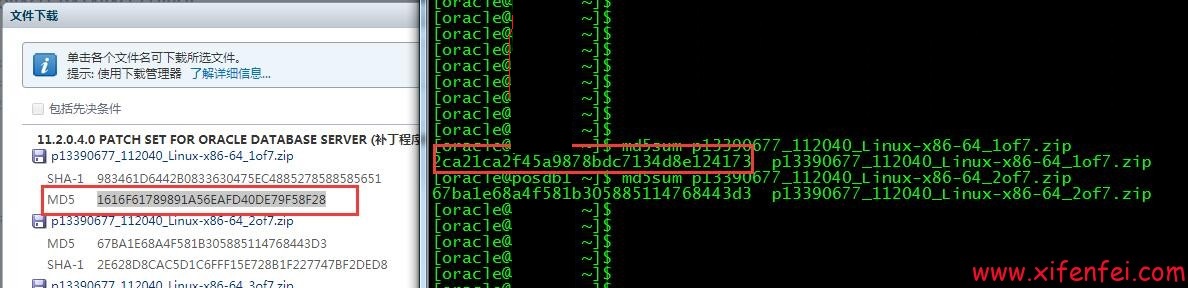
这里已经很清晰,由于prvtsupp.plb被人注入了恶意脚本从而使得数据库被创建了DBMS\_SUPPORT\_DBMONITOR的触发器和DBMS\_SUPPORT\_DBMONITORP的存储过程,从而实现了对数据库的而易破坏.

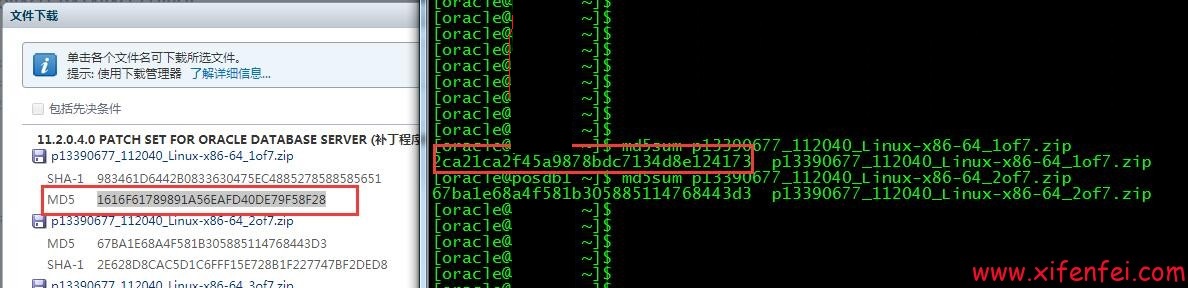
**确定破坏文件prvtsupp.plb来源于介质**

[](http://www.xifenfei.com/?p=7156)

[](http://www.xifenfei.com/?p=7156)

这里比较明显,文件就是来源database\stage\Components\oracle.rdbms.dbscripts\11.2.0.4.0\1\DataFiles\filegroup2.jar\rdbms\admin\prvtsupp.plb文件被修改导致

[](http://www.xifenfei.com/?p=7156)

[](http://www.xifenfei.com/?p=7156)

通过md5码对比,可以确定是有人对软件的安装介质进行了破坏,从而实现了恶意代码的注入,从而实现了数据库300天之后重启之后无法正常启动而是出现类似ORA-00600: internal error code, arguments: [16703], [1403], [20], [], [], [], [], [], [], [], [], []的错误.

**温馨提示**

各位一定要从官方途径下载oracle安装介质,如果是从其他互联网途径下载一定要验证md5,确保文件没有被人恶意篡改,造成无可挽回的损坏.如果真的不幸遇到这类问题,请保护现场联系我们

**<Tel:13429648788>    Q Q:107644445**

[IMG_285](http://wpa.qq.com/msgrd?v=3%26uin=107644445%26site=qq%26menu=yes)

[IMG_286](http://wpa.qq.com/msgrd?v=3%26uin=107644445%26site=qq%26menu=yes)

**E-Mail:[dba@xifenfei.com](mailto:dba@xifenfei.com)**

可以实现业务数据0丢失恢复,最大限度抢救您的数据

参考：[http://www.xifenfei.com/2017/07/oracle-software-malicious-injection.html](http://www.itpub.net/%E6%83%9C%E5%88%86%E9%A3%9E---%E8%AD%A6%E5%91%8A%EF%BC%9A%E4%BA%92%E8%81%94%E7%BD%91%E4%B8%AD%E6%9C%89oracle%E4%BB%8B%E8%B4%A8%E8%A2%AB%E6%B3%A8%E5%85%A5%E6%81%B6%E6%84%8F%E7%A8%8B%E5%BA%8F%E5%AF%BC%E8%87%B4%E2%80%94ORA-600 16703)

来自 <<http://www.itpub.net/thread-2091896-1-1.html>>