

## 10046 using oradebug

Trace from OS using PID

```
sqlplus "/ as sysdba"  
SQLPLUS> oradebug setospid 1356  
SQLPLUS> oradebug event 10046 trace name context forever, level 8
```

--- DO your work and once done stop the trace.

```
sqlplus "/ as sysdba"  
sqlplus> oradebug event 10046 trace name context off
```

Waits



SURENDER  
SARA

## 10046 using dbms\_system

Trace using serial# and sid and at various trace levels

```
execute dbms_system.set_sql_trace_in_session(sid,serial#,TRUE);  
or,  
execute dbms_system.set_ev(sid, serial#, 10046, 8, "");
```

The latter statement is better because it sets event 10046 at level 8, which will include details of all wait states. Other levels for the 10046 event are

level 1 SQL statements (equivalent to alter session set sql\_trace true;)  
level 4 include details of bind variables  
level 8 include wait events  
level 12 include both bind variables and wait events

SURENDER  
SARA

## 10046 for a given users sessions

An piece of PL/SQL code that enables tracing for all sessions belonging to a specific username:

```
Begin  
For ii in (select sid,serial# from v$session where username like upper('&user_name'))  
loop  
sys.dbms_system.set_sql_trace_in_session  
(ii.sid,ii.serial#,TRUE);  
end loop;  
end;
```

## Tracing all user connections and their activities at the user trigger level

```
CREATE OR REPLACE TRIGGER TRC_APPS  
AFTER LOGON ON APPS.schema  
DECLARE  
event_stmt varchar2(100) := 'ALTER SESSION SET EVENTS ''' 10046 TRACE NAME CONTEXT FOREVER, LEVEL 12 ''' ;  
BEGIN  
EXECUTE IMMEDIATE event_stmt ;  
END;
```