如何限定IP访问Oracle数据库

姓名：小麦苗

时间：2017.03.18

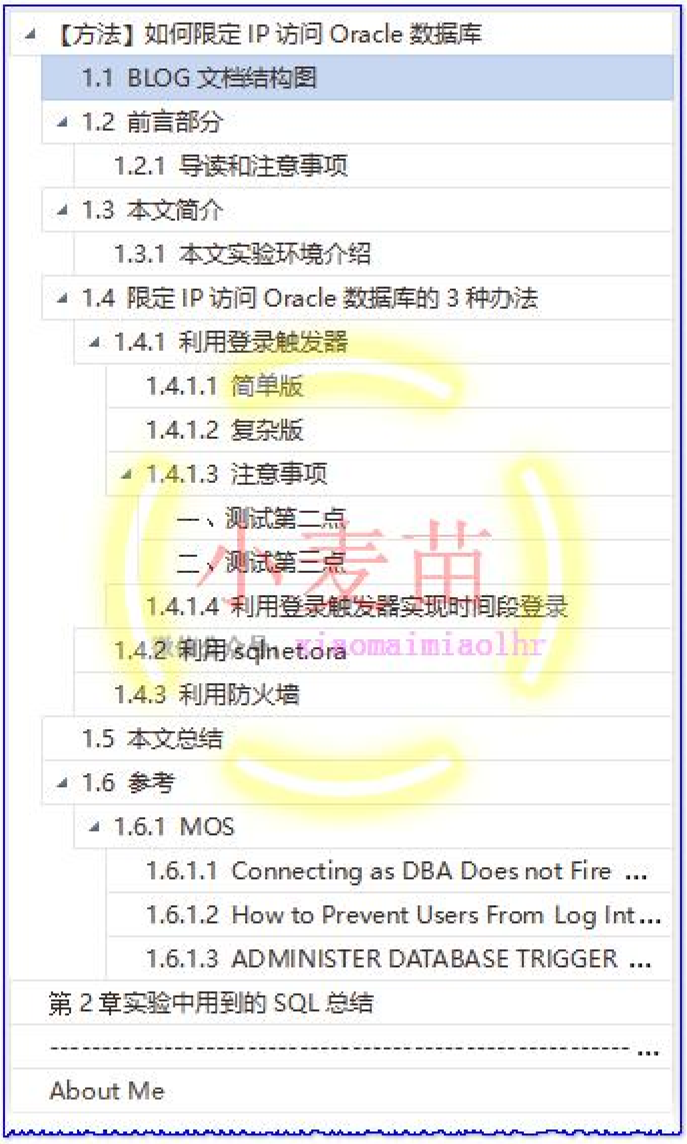
QQ群：230161599

微信公众号：xiaomaimiaolhr

博客地址：http://blog.itpub.net/26736162

【方法】如何限定 IP 访问 Oracle 数据库

1.1 BLOG 文档结构图



1.2 前言部分

1.2.1 导读和注意事项

各位技术爱好者，看完本文后，你可以掌握如下的技能，也可以学到一些其它你所不知道的知识，~O(∩_∩)O~：

① 限定 IP 访问 Oracle 数据库的 3 种方法（重点）

② 如何将信息写入到 Oracle 的告警日志中

③ RAISE_APPLICATION_ERROR 不能抛出错误到客户端环境

④ 系统触发器

⑤ 隐含参数：_system_trig_enabled

Tips:

① 本文在 itpub（http://blog.itpub.net/26736162）、博客园（http://www.cnblogs.com/lhrbest）和微信公众号（xiaomaimiaolhr）上有同步更新。

② 文章中用到的所有代码、相关软件、相关资料及本文的 pdf 版本都请前往小麦苗的云盘下载，小麦苗的云盘地址见：http://blog.itpub.net/26736162/viewspace-1624453/。

③ 若网页文章代码格式有错乱，请下载 pdf 格式的文档来阅读。

④ 在本篇 BLOG 中，代码输出部分一般放在一行一列的表格中。

本文若有错误或不完善的地方请大家多多指正，您的批评指正是我写作的最大动力。

1.3 本文简介

本文详细介绍了 3 种限制 IP 地址登录 Oracle 数据库的办法。

1.3.1 本文实验环境介绍

项目	source db
db 类型	RAC
db version	11.2.0.3.0
db 存储	ASM
OS 版本及 kernel 版本	RHEL 6.5
数据库服务器 IP 地址	192.168.59.130
客户端 IP 地址	192.168.59.1 或 192.168.59.129

1.4 限定 IP 访问 Oracle 数据库的 3 种办法

1.4.1 利用登录触发器

1.4.1.1 简单版

```
SYS@orclasm > CREATE OR REPLACE TRIGGER CHK_IP_LHR
  2    AFTER LOGON ON DATABASE
  3  DECLARE
  4    V_IPADDR    VARCHAR2(30);
  5    V_LOGONUSER VARCHAR2(60);
  6  BEGIN
  7    SELECT SYS_CONTEXT('USERENV', 'IP_ADDRESS'),
  8           SYS_CONTEXT('USERENV', 'SESSION_USER')
  9      INTO V_IPADDR, V_LOGONUSER
 10      FROM DUAL;
 11    IF V_IPADDR LIKE ('192.168.59.%') THEN
 12      RAISE_APPLICATION_ERROR('-20001', 'User '||V_LOGONUSER||' is not allowed to connect from
'||V_IPADDR);
 13    END IF;
 14  END;
 15  /

Trigger created.

SYS@orclasm > create user lhr8 identified by lhr;
```

```
User created.

SYS@orclasm > grant  resource,connect to lhr8;

Grant succeeded.
```

客户端登录：

```
D:\Users\xiaomaimiao>ipconfig
以太网适配器 VMware Network Adapter VMnet8:

   连接特定的 DNS 后缀 . . . . . . . :
   本地链接 IPv6 地址. . . . . . . . : fe80::850a:3293:c7fb:75e1%24
   IPv4 地址 . . . . . . . . . . . . : 192.168.59.1
   子网掩码  . . . . . . . . . . . . : 255.255.255.0
D:\Users\xiaomaimiao>sqlplus lhr8/lhr@orclasm

SQL*Plus: Release 11.2.0.1.0 Production on Sat Mar 18 17:29:27 2017

Copyright (c) 1982, 2010, Oracle.  All rights reserved.

ERROR:
ORA-00604: error occurred at recursive SQL level 1
ORA-20001: User LHR8 is not allowed to connect from 192.168.59.1
ORA-06512: at line 10


Enter user-name:
```

告警日志无输出。

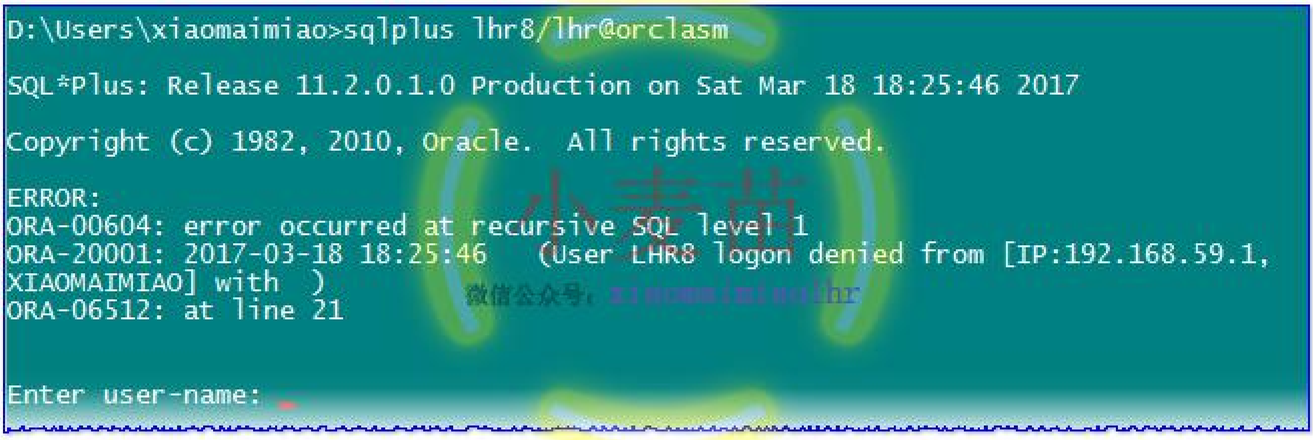
1.4.1.2 复杂版

复杂版就是需要记录登录日志，并把报错信息输出到告警日志中。

```
CREATE TABLE XB_AUDIT_LOGON_LHR(
  ID   NUMBER  PRIMARY KEY,
  INST_ID NUMBER,
  OPER_DATE     DATE,
  OS_USER      VARCHAR2(255),
  CLIENT_IP          VARCHAR2(20),
  CLIENT_HOSTNAME     VARCHAR2(30),
  DB_SCHEMA    VARCHAR2(30),
  SID         NUMBER,
  SERIAL#      NUMBER,
  SPID        NUMBER,
  SESSION_TYPE VARCHAR2(1000),
  DATABASE_NAME VARCHAR2(255)
  ) NOLOGGING
PARTITION BY RANGE(OPER_DATE)  INTERVAL(NUMTOYMINTERVAL(1,'MONTH'))  SUBPARTITION BY HASH(INST_ID)
SUBPARTITION TEMPLATE (
  SUBPARTITION SP1 ,
  SUBPARTITION SP2 )
   (PARTITION P201610  VALUES LESS THAN(TO_DATE('201610','YYYYMM')));

CREATE SEQUENCE S_XB_AUDIT_DDL_LHR START WITH 1 INCREMENT BY 1 CACHE 2000;
SELECT S_XB_AUDIT_DDL_LHR.NEXTVAL FROM DUAL;
CREATE INDEX IND_AUDIT_DDL_OS_USER ON   XB_AUDIT_LOGON_LHR(OS_USER) LOCAL NOLOGGING;
CREATE INDEX IND_AUDIT_DDL_SID ON   XB_AUDIT_LOGON_LHR(SID,SERIAL#) LOCAL NOLOGGING;

GRANT SELECT ON  XB_AUDIT_LOGON_LHR TO PUBLIC;

CREATE OR REPLACE PROCEDURE PRO_TRI_DDL_INSET_LHR AUTHID CURRENT_USER AS
  SP_XB_AUDIT_DDL_LHR XB_AUDIT_LOGON_LHR%ROWTYPE;
  V_COUNT              NUMBER;
  V_TMP                VARCHAR2(255);
  V_MODULE             VARCHAR2(4000);
  V_ACTION             VARCHAR2(4000);
  V_MESSAGE            VARCHAR2(4000);
BEGIN

  BEGIN

    SELECT A.SID,
           A.SERIAL#,
           (SELECT B.SPID
              FROM GV$PROCESS B
             WHERE B.ADDR = A.PADDR
               AND B.INST_ID = USERENV('INSTANCE')) SPID,
           UPPER(A.OSUSER) OSUSER,
           A.MACHINE || '--' || A.PROGRAM || '--' || A.MODULE || '--' ||
           A.ACTION SESSION_TYPE,
           A.USERNAME,
           A.INST_ID
      INTO SP_XB_AUDIT_DDL_LHR.SID,
           SP_XB_AUDIT_DDL_LHR.SERIAL#,
           SP_XB_AUDIT_DDL_LHR.SPID,
           SP_XB_AUDIT_DDL_LHR.OS_USER,
           SP_XB_AUDIT_DDL_LHR.SESSION_TYPE,
           SP_XB_AUDIT_DDL_LHR.DB_SCHEMA,
           SP_XB_AUDIT_DDL_LHR.INST_ID
      FROM GV$SESSION A
     WHERE A.AUDSID = USERENV('SESSIONID')
       AND A.INST_ID = USERENV('INSTANCE');

    --job 信息 不同的数据库这里的os_user需要修改
    IF UPPER(SYS_CONTEXT('USERENV', 'OS_USER')) = 'ORACLE' THEN
      SELECT COUNT(1)
        INTO V_COUNT
        FROM DBA_JOBS_RUNNING A, DBA_JOBS B
       WHERE A.JOB = B.JOB
         AND A.SID = SP_XB_AUDIT_DDL_LHR.SID
         AND A.INSTANCE = USERENV('INSTANCE');
      IF V_COUNT > 0 THEN
        SELECT '【DBA_JOBS：' || B.JOB || '--' || B.WHAT || '】'
          INTO V_TMP
          FROM DBA_JOBS_RUNNING A, DBA_JOBS B
         WHERE A.JOB = B.JOB
           AND A.SID = SP_XB_AUDIT_DDL_LHR.SID
           AND A.INSTANCE = USERENV('INSTANCE');
      ELSE
        SELECT '--' || B.JOB_TYPE || '--' || B.JOB_ACTION
          INTO V_TMP
          FROM DBA_SCHEDULER_RUNNING_JOBS A, DBA_SCHEDULER_JOBS B
         WHERE A.JOB_NAME = B.JOB_NAME
           AND A.SESSION_ID = SP_XB_AUDIT_DDL_LHR.SID
           AND A.RUNNING_INSTANCE = USERENV('INSTANCE');
      END IF;
    END IF;

  EXCEPTION
    WHEN OTHERS THEN
```

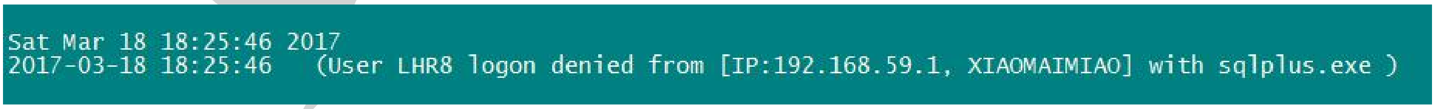
```
      NULL;
  END;

  BEGIN
    --v_module is much useful, "plsqldev.exe"
    DBMS_APPLICATION_INFO.READ_MODULE(V_MODULE, V_ACTION);
    V_MESSAGE := TO_CHAR(SYSDATE, 'yyyy-mm-dd hh24:mi:ss') || '   (User ' ||
                 SYS.LOGIN_USER || ' logon denied from [IP:' ||
                 ORA_CLIENT_IP_ADDRESS || ', ' ||
                 UPPER(SYS_CONTEXT('USERENV', 'OS_USER')) || '] with ' ||
                 V_MODULE || ' ' || V_ACTION || ')';

    --write alert.log
    SYS.DBMS_SYSTEM.KSDWRT(2, V_MESSAGE);
  EXCEPTION
    WHEN OTHERS THEN
      NULL;
  END;

  INSERT INTO XB_AUDIT_LOGON_LHR
    (ID,
     INST_ID,
     OPER_DATE,
     OS_USER,
     CLIENT_IP,
     CLIENT_HOSTNAME,
     DB_SCHEMA,
     SID,
     SERIAL#,
     SPID,
     SESSION_TYPE,
     DATABASE_NAME)
  VALUES
    (S_XB_AUDIT_DDL_LHR.NEXTVAL,
     USERENV('INSTANCE'), -- sp_xb_audit_ddl_lhr.INST_ID  ora_instance_num
     SYSDATE,
     UPPER(SYS_CONTEXT('USERENV', 'OS_USER')), -- sp_xb_audit_ddl_lhr.os_user
     SYS_CONTEXT('userenv', 'ip_address'), --ora_client_ip_address
     SYS_CONTEXT('userenv', 'terminal'), --sys_context('userenv', 'host')
     NVL2(ORA_LOGIN_USER,
          SYS_CONTEXT('USERENV', 'SESSION_USER'),
          SP_XB_AUDIT_DDL_LHR.DB_SCHEMA), -- SYS_CONTEXT('USERENV', 'SESSION_USER') sys.login_user
     SP_XB_AUDIT_DDL_LHR.SID, ---- SYS_CONTEXT('USERENV', 'SID'),
     SP_XB_AUDIT_DDL_LHR.SERIAL#,
     SP_XB_AUDIT_DDL_LHR.SPID,
     SP_XB_AUDIT_DDL_LHR.SESSION_TYPE || V_TMP,
     ORA_DATABASE_NAME --sys_context('USERENV', 'DB_NAME')
     );

  COMMIT;

EXCEPTION
  WHEN OTHERS THEN
    ROLLBACK;
END PRO_TRI_DDL_INSET_LHR;
/


CREATE OR REPLACE TRIGGER CHK_IP_LHR
  AFTER LOGON ON DATABASE
DECLARE
  V_IPADDR    VARCHAR2(30);
  V_LOGONUSER VARCHAR2(60);
  V_MODULE    VARCHAR2(4000);
  V_ACTION    VARCHAR2(4000);
  V_MESSAGE   VARCHAR2(4000);
BEGIN
  SELECT SYS_CONTEXT('USERENV', 'IP_ADDRESS'),
         SYS_CONTEXT('USERENV', 'SESSION_USER')
    INTO V_IPADDR, V_LOGONUSER
    FROM DUAL;

  V_MESSAGE := TO_CHAR(SYSDATE, 'yyyy-mm-dd hh24:mi:ss') || '   (User ' ||
               SYS.LOGIN_USER || ' logon denied from [IP:' ||
               ORA_CLIENT_IP_ADDRESS || ', ' ||
               UPPER(SYS_CONTEXT('USERENV', 'OS_USER')) || '] with ' ||
               V_MODULE || ' ' || V_ACTION || ')';

  IF V_IPADDR LIKE ('192.168.59.%') THEN
    PRO_TRI_DDL_INSET_LHR;
    RAISE_APPLICATION_ERROR('-20001', V_MESSAGE);

  END IF;
END;
/
```

客户端登录：



```
D:\Users\xiaomaimiao>sqlplus lhr8/lhr@orclasm

SQL*Plus: Release 11.2.0.1.0 Production on Sat Mar 18 18:25:46 2017

Copyright (c) 1982, 2010, Oracle.  All rights reserved.

ERROR:
ORA-00604: error occurred at recursive SQL level 1
ORA-20001: 2017-03-18 18:25:46   (User LHR8 logon denied from [IP:192.168.59.1,
XIAOMAIMIAO] with  )
ORA-06512: at line 21


Enter user-name:
```

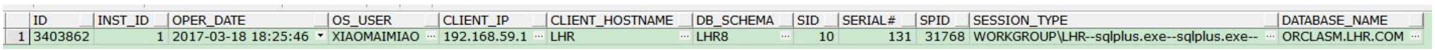
告警日志：



```
Sat Mar 18 18:25:46 2017
2017-03-18 18:25:46   (User LHR8 logon denied from [IP:192.168.59.1, XIAOMAIMIAO] with sqlplus.exe )
```

查询日志表：

```
SELECT * FROM XB_AUDIT_LOGON_LHR;
```



	ID	INST_ID	OPER_DATE	OS_USER	CLIENT_IP	CLIENT_HOSTNAME	DB_SCHEMA	SID	SERIAL#	SPID	SESSION_TYPE	DATABASE_NAME
1	3403862	1	2017-03-18 18:25:46	XIAOMAIMIAO	192.168.59.1	LHR	LHR8	10	131	31768	WORKGROUP\LHR--sqlplus.exe--sqlplus.exe--	ORCLASM.LHR.COM

1.4.1.3 注意事项

需要注意的问题：

① 触发的对象类型可以为 DATABASE，也可以为“用户名.SCHEMA”，如：

```
AFTER LOGON ON DATABASE
AFTER LOGON ON SCOTT.SCHEMA
```

② 当触发的对象类型为 DATABASE 的时候，登录用户不能拥有“ADMINISTER DATABASE TRIGGER”的系统

权限；当触发的对象类型为“用户名.SCHEMA”的时候，登录用户不能拥有“ALTER ANY TRIGGER”的系统权限。否则，这些用户还是会正常登录到数据库，只是将相应的报错信息写入到告警日志中。所以，拥有IMP_FULL_DATABASE 和 DBA 角色的用户以及 SYS 和 EXFSYS 用户将不能通过这种方式限制登录。

③ 隐含参数“_SYSTEM_TRIG_ENABLED”的默认值是 TRUE，即允许 DDL 和系统触发器。当设置隐含参数“_SYSTEM_TRIG_ENABLED”为 FALSE 的时候，将禁用 DDL 和系统触发器。所以，当该值设置为 FALSE 的时候将不能通过这种方式限制登录。

一、 测试第二点

第二点测试如下：

```
SYS@orclasm > grant ADMINISTER DATABASE TRIGGER to lhr8;

Grant succeeded.
```

客户端登录：

```
D:\Users\xiaomaimiao>sqlplus lhr8/lhr@orclasm

SQL*Plus: Release 11.2.0.1.0 Production on Sat Mar 18 18:33:13 2017

Copyright (c) 1982, 2010, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

LHR8@orclasm>
```

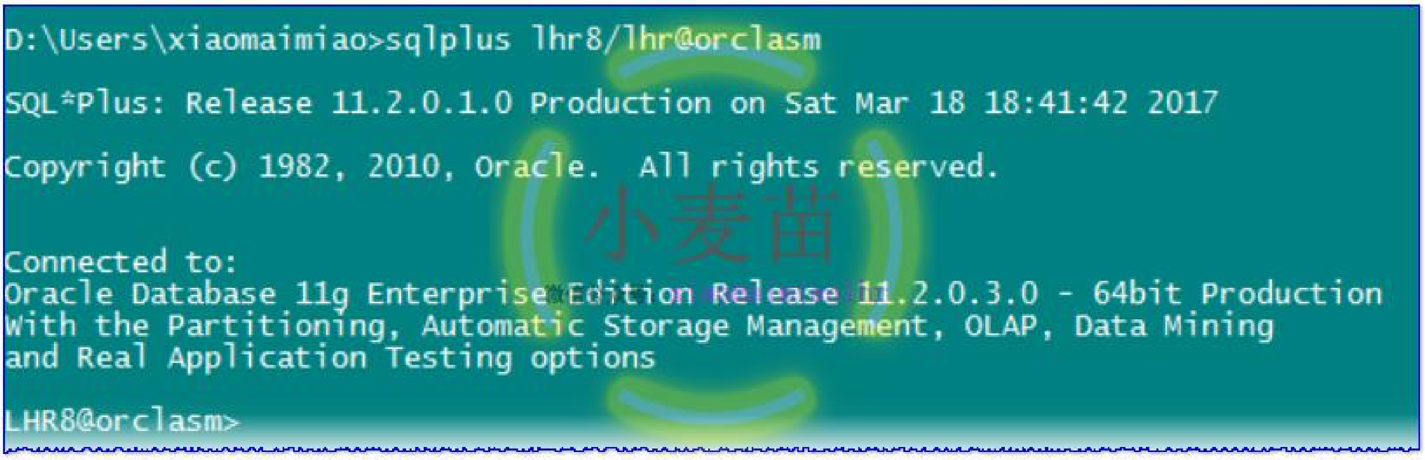
告警日志：

```
Sat Mar 18 18:33:13 2017
2017-03-18 18:33:13   (User LHR8 logon denied from [IP:192.168.59.1, XIAOMAIMIAO] with sqlplus.exe )
Errors in file /u01/app/oracle/diag/rdbms/orclasm/orclasm/trace/orclasm_ora_33505.trc:
ORA-00604: error occurred at recursive SQL level 1
ORA-20001: 2017-03-18 18:33:13   (User LHR8 logon denied from [IP:192.168.59.1, XIAOMAIMIAO] with  )
ORA-06512: at line 21
```

继续测试：

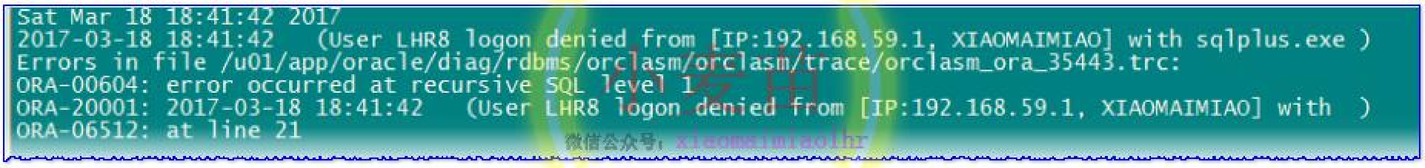
```
SYS@orclasm > revoke  ADMINISTER DATABASE TRIGGER from lhr8;

Revoke succeeded.

SYS@orclasm > GRANT ALTER ANY TRIGGER TO LHR8;

Grant succeeded.

SYS@orclasm >
```

客户端继续登录，发现不能正常登录。将触发器中的 AFTER LOGON ON DATABASE 修改为 AFTER LOGON ON LHR8.SCHEMA，其他不变，继续测试：



```
D:\Users\xiaomaimiao>sqlplus lhr8/lhr@orclasm

SQL*Plus: Release 11.2.0.1.0 Production on Sat Mar 18 18:41:42 2017

Copyright (c) 1982, 2010, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

LHR8@orclasm>
```

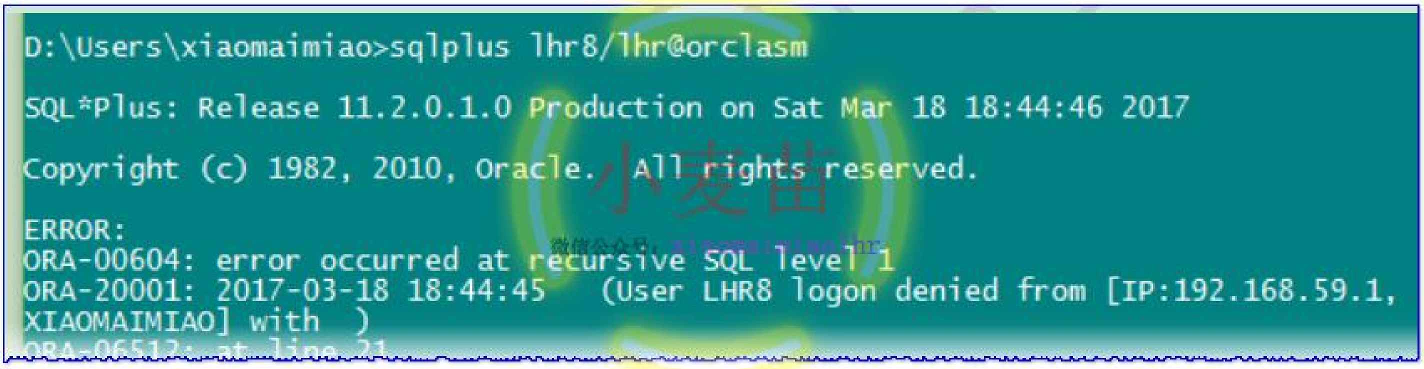
发现可以正常登录了，告警日志：



```
Sat Mar 18 18:41:42 2017
2017-03-18 18:41:42    (User LHR8 logon denied from [IP:192.168.59.1, XIAOMAIMIAO] with sqlplus.exe )
Errors in file /u01/app/oracle/diag/rdbms/orclasm/orclasm/trace/orclasm_ora_35443.trc:
ORA-00604: error occurred at recursive SQL level 1
ORA-20001: 2017-03-18 18:41:42    (User LHR8 logon denied from [IP:192.168.59.1, XIAOMAIMIAO] with  )
ORA-06512: at line 21
```

二、　　测试第三点

将触发器中的 AFTER LOGON ON LHR8.SCHEMA 修改为 AFTER LOGON ON DATABASE，其他不变，继续测试：



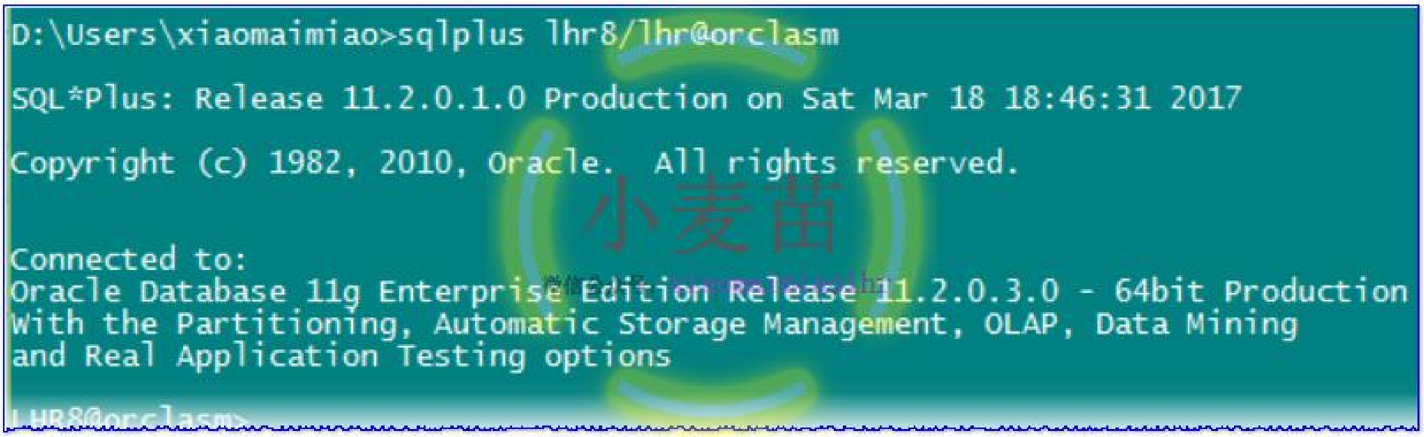
```
D:\Users\xiaomaimiao>sqlplus lhr8/lhr@orclasm

SQL*Plus: Release 11.2.0.1.0 Production on Sat Mar 18 18:44:46 2017

Copyright (c) 1982, 2010, Oracle.  All rights reserved.

ERROR:
ORA-00604: error occurred at recursive SQL level 1
ORA-20001: 2017-03-18 18:44:45    (User LHR8 logon denied from [IP:192.168.59.1,
XIAOMAIMIAO] with  )
ORA-06512: at line 21
```

不能正常登录，下面禁用系统触发器：

```
SYS@orclasm > set pagesize 9999
SYS@orclasm > set line 9999
SYS@orclasm > col NAME format a40
SYS@orclasm > col KSPPDESC format a50
SYS@orclasm > col KSPPSTVL format a20
SYS@orclasm > SELECT a.INDX,
  2         a.KSPPINM NAME,
  3         a.KSPPDESC,
  4         b.KSPPSTVL
  5  FROM   x$ksppi  a,
  6         x$ksppcv b
  7  WHERE  a.INDX = b.INDX
  8  and lower(a.KSPPINM)  like  lower('%&parameter%');
Enter value for parameter: _system_trig_enabled
old   8: and lower(a.KSPPINM)  like  lower('%&parameter%')
new   8: and lower(a.KSPPINM)  like  lower('%_system_trig_enabled%')

     INDX NAME                                     KSPPDESC                                           KSPPSTVL
```

```
---------- ---------------------------------------- ---------------------------------------------------
--------------------
     1750 _system_trig_enabled                 are system triggers enabled                    TRUE

SYS@orclasm > alter system  set "_system_trig_enabled"=false;

System altered.

SYS@orclasm >
```

进行登录：



```
D:\Users\xiaomaimiao>sqlplus lhr8/lhr@orclasm

SQL*Plus: Release 11.2.0.1.0 Production on Sat Mar 18 18:46:31 2017

Copyright (c) 1982, 2010, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

LHR8@orclasm>
```

发现可以正常登录了。将参数"_system_trig_enabled"修改回原值。

```
SYS@orclasm > alter system  set "_system_trig_enabled"=true;

System altered.

SYS@orclasm > alter system reset "_system_trig_enabled" scope=spfile sid='*';

System altered.

SYS@orclasm >
```

1.4.1.4 利用登录触发器实现时间段登录

Use Event Triggers

-------------------

If you allow the users to log in the database only from Monday to Friday included, and from 8AM to 6PM, create an event trigger that checks after logon on database for each user (except the DBA users) that the connection occurs only within this timeframe.

Example 1

-------

1. No check set up yet: any ordinary user can log into the database:

```
SQL> connect test_trigger/test_trigger
```

```
  Connected.
```

2. The DBA creates an event trigger that checks if the connection occurs between Monday and Friday , and within working hours: 8AM to 6PM.

```
  SQL> connect system/manager
  Connected.
  SQL> create or replace trigger logon_trg after logon on database
      begin
       if (to_char(sysdate,'D') not between '2' and '6')
          or (to_char(sysdate, 'HH24') not between '08' and '18') then
          RAISE_APPLICATION_ERROR(-20001, 'You are not allowed to log into
                                   database now.');
    end if;
   end;
   /
```

Trigger created.

3. It is Friday 5PM : an ordinary user can log into the database:

```
  SQL> connect test_trigger/test_trigger
  Connected.

  It is Monday  7AM : an ordinary user cannot log into the database
  It is Saturday 9AM : an ordinary user cannot log into the database:

  SQL> connect test_trigger/test_trigger
  ERROR:
  ORA-00604: error occurred at recursive SQL level 1
  ORA-20001: You are not allowed to log into database now.
  ORA-06512: at line 3


  Warning: You are no longer connected to ORACLE.
  SQL>
```

Example 2

Another example to restrict the logon periods for a users so that they can only access the database betrween the periods to 17:00 - 24:00 daily.

If the user attempts to logon during a period outside of this range his logon attempt will fail:

```
  SQL> CREATE OR REPLACE TRIGGER ScottLoginTrigger after logon on scott.schema
      declare
        temp varchar2(50);
        v_time varchar2(50);
        begin
         temp := 'select to_char(sysdate,''HH24:MI'') from dual';
         EXECUTE IMMEDIATE temp into v_time;
         if (to_date(v_time,'HH24:MI') < to_date('17:00','HH24:MI')) then
           raise_application_error (-20001,'SCOTT access is denied until 17:00. The current time is
'||v_time,true);
         end if;
```

```
          if (to_date(v_time,'HH24:MI') > to_date('23:59','HH24:MI')) then
            raise_application_error (-20001,'SCOTT access is denied because the time is past 23:59. The
current time is '||v_time,true);
          end if;
        end;
       /
```

However, users with ADMINISTER DATABASE TRIGGER system privilege can log into the database any time.

1.4.2 利用 sqlnet.ora

第二种是修改$ORACLE_HOME/network/admin/sqlnet.ora 文件，增加如下内容：

```
TCP.VALIDNODE_CHECKING=YES  #开启 IP 限制功能
TCP.INVITED_NODES=(127.0.0.1,IP1,IP2,……)  #允许访问数据库的 IP 地址列表，多个 IP 地址使用逗号分开
TCP.EXCLUDED_NODES=(IP1,IP2,……)  #禁止访问数据库的 IP 地址列表，多个 IP 地址使用逗号分开
```

之后重新启动监听器即可。这样客户端在登录的时候会报“ORA-12537: TNS:connection closed”的错误。

需要注意的问题：

① 需要设置参数 TCP.VALIDNODE_CHECKING 为 YES 才能激活该特性。

② 一定要许可或不要禁止数据库服务器本机的 IP 地址，否则通过 lsnrctl 将不能启动或停止监听，因为该过程监听程序会通过本机的 IP 访问监听器，而该 IP 被禁止了，但是通过服务启动或关闭则不影响。

③ 当参数 TCP.INVITED_NODES 和 TCP.EXCLUDED_NODES 设置的地址相同的时候以 TCP.INVITED_NODES 的配置为主。

④ 修改之后，一定要重起监听才能生效，而不需要重新启动数据库。

⑤ 这个方式只是适合 TCP/IP 协议。

⑥ 这个配置适用于 Oracle 9i 以上版本。在 Oracle 9i 之前的版本使用文件 protocol.ora。

⑦ 在服务器上直接连接数据库不受影响。

⑧ 这种限制方式是通过监听器来限制的。

⑨ 这个限制只是针对 IP 检测，对于用户名检测是不支持的。

删除之前创建的触发器，继续测试。

```
[grid@rhel6lhr ~]$ more $ORACLE_HOME/network/admin/sqlnet.ora
# sqlnet.ora Network Configuration File: /u01/app/grid/11.2.0/network/admin/sqlnet.ora
# Generated by Oracle configuration tools.

NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)

ADR_BASE = /u01/app/grid
TCP.VALIDNODE_CHECKING=YES
TCP.INVITED_NODES=(127.0.0.1,192.168.59.130,192.168.59.1,192.168.59.2)
TCP.EXCLUDED_NODES=(172.168.*)
[grid@rhel6lhr ~]$
```

重启监听：

```
[grid@rhel6lhr ~]$ lsnrctl reload

LSNRCTL for Linux: Version 11.2.0.3.0 - Production on 18-MAR-2017 18:55:54

Copyright (c) 1991, 2011, Oracle.  All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.59.130)(PORT=1521)))
The command completed successfully
[grid@rhel6lhr ~]$
```

客户端连接：

```
[oracle@orcltest ~]$ ip a | grep eth0
4: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    inet 192.168.59.129/24 brd 192.168.59.255 scope global eth0
[oracle@orcltest ~]$ sqlplus lhr8/lhr@192.168.59.130/orclasm.lhr.com

SQL*Plus: Release 11.2.0.3.0 Production on Sat Mar 18 18:57:43 2017

Copyright (c) 1982, 2011, Oracle.  All rights reserved.

ERROR:
ORA-12537: TNS:connection closed


Enter user-name:
```

监听报错：

```

Sat Mar 18 18:58:44 2017
18-MAR-2017 18:58:44 * 12546
TNS-12546: TNS:permission denied
 TNS-12560: TNS:protocol adapter error
  TNS-00516: Permission denied
```

使用 192.168.59.1 客户端进行登录：

```
D:\Users\xiaomaimiao>sqlplus lhr8/lhr@192.168.59.130/orclasm.lhr.com

SQL*Plus: Release 11.2.0.1.0 Production on Sat Mar 18 19:00:15 2017

Copyright (c) 1982, 2010, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

LHR8@192.168.59.130/orclasm.lhr.com>
```

发现可以正常登录。将 TCP.INVITED_NODES 的 IP 里加入 192.168 网段，则可以正常登录：

```
[grid@rhel6lhr ~]$  more $ORACLE_HOME/network/admin/sqlnet.ora
# sqlnet.ora Network Configuration File: /u01/app/grid/11.2.0/network/admin/sqlnet.ora
# Generated by Oracle configuration tools.

NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)
```

```
ADR_BASE = /u01/app/grid
TCP.VALIDNODE_CHECKING=YES
TCP.INVITED_NODES=(127.0.0.1,192.168.59.130,192.168.59.1,192.168.59.2,192.168.*)
TCP.EXCLUDED_NODES=(172.168.*)
```

客户端登录：

```
[oracle@orcltest ~]$ sqlplus lhr8/lhr@192.168.59.130/orclasm.lhr.com

SQL*Plus: Release 11.2.0.3.0 Production on Sat Mar 18 19:03:27 2017

Copyright (c) 1982, 2011, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

LHR8@192.168.59.130/orclasm.lhr.com>
```

1.4.3 利用防火墙

第 3 种是修改数据库服务器的 IPTABLES（配置文件：/etc/sysconfig/iptables）来限制某些 IP 登录数据库服务器。如下：

```
iptables -I INPUT -s 192.168.59.129 -j DROP
service iptables save
```

则，192.168.59.129 这台主机将不能连接到数据库服务器了，会报“ORA-12170: TNS:Connect timeout occurred”的错误。

测试：

```
[oracle@orcltest ~]$ sqlplus lhr8/lhr@192.168.59.130/orclasm.lhr.com

SQL*Plus: Release 11.2.0.3.0 Production on Sat Mar 18 19:19:23 2017

Copyright (c) 1982, 2011, Oracle.  All rights reserved.

ERROR:
ORA-12170: TNS:Connect timeout occurred


Enter user-name:

[oracle@orcltest ~]$ tnsping 192.168.59.130/orclasm.lhr.com

TNS Ping Utility for Linux: Version 11.2.0.3.0 - Production on 18-MAR-2017 19:18:16

Copyright (c) 1997, 2011, Oracle.  All rights reserved.

Used parameter files:
/u02/app/oracle/product/11.2.0/dbhome_1/network/admin/sqlnet.ora

Used EZCONNECT adapter to resolve the alias
Attempting to contact
(DESCRIPTION=(CONNECT_DATA=(SERVICE_NAME=orclasm.lhr.com))(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.59.130)(
PORT=1521)))
```

```
^C
[oracle@orcltest ~]$ ping 192.168.59.130
PING 192.168.59.130 (192.168.59.130) 56(84) bytes of data.
^C
--- 192.168.59.130 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2136ms

[oracle@orcltest ~]$
```

该部分可以参考网络配置，小麦苗从网上找了很多。

我们可以通过以下的 iptables 的设置来限制用户访问 oracle 所在 linux 操作系统的安全。

1、清楚操作系统默认的 iptables 策略

我本机安装的是 centos6.0,安装之后系统会提供 iptables 默认的 policy 策略,我们首先要清楚默认的策略

```
    iptables -F
```

2、开发 22 和 1521 端口对局域网的某个 IP，在本例中客户端 ip 是 192.168.1.125,oracle 所在机器的 IP 是 192.168.1.144,在这里，设置仅有该客户端可以访问 22 和 1521 端口，局域网内的其他 IP 都不允许访问，

```
  iptables -A INPUT -s 192.168.1.125/32 -i eth0 -p tcp  --dport 22 -j ACCEPT
  iptables -A INPUT -s 192.168.1.125/32 -i eth0 -p tcp  --dport 1521 -j ACCEPT
  iptables -A INPUT -s 192.168.1.0/24 -p tcp  --dport 22 -j DROP
  iptables -A INPUT -s 192.168.1.0/24 -p tcp  --dport 1521 -j DROP
```

这样同一网段内除 192.168.1.125 之外其他 IP 都不能访问数据库服务器，即使 ping 命令也不可以

3、开发 22 和 1521 的 OUTPUT 链给 192.168.1.125，否则已经启动的 oracle instance 的 pmon 进程无法动态注册到 1521 端口中

```
   iptables -A OUTPUT -d 192.168.1.125/32 -p tcp  --sport 22 -j ACCEPT
   iptables -A OUTPUT -d 192.168.1.125/32 -p tcp --sport 1521 -j ACCEPT
```

4、保存当前设置的 iptables 规则

```
    service iptables save
```

这时系统会将已经设置的规则保存到/etc/sysconfig/iptables 文件中

否则重启之后之前设置的规则都会失效

先关闭所有的 80 端口

开启 ip 段 192.168.1.0/24 端的 80 口

开启 ip 段 211.123.16.123/24 端 ip 段的 80 口

```
# iptables -I INPUT -p tcp --dport 80 -j DROP
# iptables -I INPUT -s 192.168.1.0/24 -p tcp --dport 80 -j ACCEPT
# iptables -I INPUT -s 211.123.16.123/24 -p tcp --dport 80 -j ACCEPT
```

以上是临时设置。

1.先备份 iptables

```
# cp /etc/sysconfig/iptables /var/tmp
```

2.然后保存 iptables

```
# service iptables save
```

3.重启防火墙

```
#service iptables restart
```

以下是端口，先全部封再开某些的 IP

```
iptables -I INPUT -p tcp --dport 9889 -j DROP
iptables -I INPUT -s 192.168.1.0/24 -p tcp --dport 9889 -j ACCEPT
```

如果用了 NAT 转发记得配合以下才能生效

```
iptables -I FORWARD -p tcp --dport 80 -j DROP
iptables -I FORWARD -s 192.168.1.0/24 -p tcp --dport 80 -j ACCEPT
```

常用的 IPTABLES 规则如下：

只能收发邮件，别的都关闭

```
iptables -I Filter -m mac --mac-source 00:0F:EA:25:51:37 -j DROP
iptables -I Filter -m mac --mac-source 00:0F:EA:25:51:37 -p udp --dport 53 -j ACCEPT
iptables -I Filter -m mac --mac-source 00:0F:EA:25:51:37 -p tcp --dport 25 -j ACCEPT
iptables -I Filter -m mac --mac-source 00:0F:EA:25:51:37 -p tcp --dport 110 -j ACCEPT
```

IPSEC NAT 策略

```
iptables -I PFWanPriv -d 192.168.100.2 -j ACCEPT
iptables -t nat -A PREROUTING -p tcp --dport 80 -d $INTERNET_ADDR -j DNAT --to-destination 192.168.100.2:80
iptables -t nat -A PREROUTING -p tcp --dport 1723 -d $INTERNET_ADDR -j DNAT --to-destination 192.168.100.2:1723
iptables -t nat -A PREROUTING -p udp --dport 1723 -d $INTERNET_ADDR -j DNAT --to-destination 192.168.100.2:1723
iptables -t nat -A PREROUTING -p udp --dport 500 -d $INTERNET_ADDR -j DNAT --to-destination 192.168.100.2:500
iptables -t nat -A PREROUTING -p udp --dport 4500 -d $INTERNET_ADDR -j DNAT --to-destination 192.168.100.2:4500
```

FTP 服务器的 NAT

```
iptables -I PFWanPriv -p tcp --dport 21 -d 192.168.100.200 -j ACCEPT
iptables -t nat -A PREROUTING -p tcp --dport 21 -d $INTERNET_ADDR -j DNAT --to-destination 192.168.100.200:21
```

只允许访问指定网址

```
iptables -A Filter -p udp --dport 53 -j ACCEPT
iptables -A Filter -p tcp --dport 53 -j ACCEPT
iptables -A Filter -d www.3322.org -j ACCEPT
iptables -A Filter -d img.cn99.com -j ACCEPT
iptables -A Filter -j DROP
```

开放一个 IP 的一些端口，其它都封闭

```
iptables -A Filter -p tcp --dport 80 -s 192.168.100.200 -d www.pconline.com.cn -j ACCEPT
iptables -A Filter -p tcp --dport 25 -s 192.168.100.200 -j ACCEPT
iptables -A Filter -p tcp --dport 109 -s 192.168.100.200 -j ACCEPT
iptables -A Filter -p tcp --dport 110 -s 192.168.100.200 -j ACCEPT
iptables -A Filter -p tcp --dport 53 -j ACCEPT
iptables -A Filter -p udp --dport 53 -j ACCEPT
iptables -A Filter -j DROP
```

多个端口

```
iptables -A Filter -p tcp -m multiport --destination-port 22,53,80,110 -s 192.168.20.3 -j REJECT
```

连续端口

```
iptables -A Filter -p tcp -m multiport --source-port 22,53,80,110 -s 192.168.20.3 -j REJECT iptables -A Filter -p tcp --source-port 2:80 -s 192.168.20.3 -j REJECT
```

指定时间上网

```
iptables -A Filter -s 10.10.10.253 -m time --timestart 6:00 --timestop 11:00 --days Mon,Tue,Wed,Thu,Fri,Sat,Sun -j DROP
iptables -A Filter -m time --timestart 12:00 --timestop 13:00 --days Mon,Tue,Wed,Thu,Fri,Sat,Sun -j ACCEPT
iptables -A Filter -m time --timestart 17:30 --timestop 8:30 --days Mon,Tue,Wed,Thu,Fri,Sat,Sun -j ACCEPT
```

禁止多个端口服务

```
iptables -A Filter -m multiport -p tcp --dport 21,23,80 -j ACCEPT
```

将 WAN 口 NAT 到 PC

```
iptables -t nat -A PREROUTING -i $INTERNET_IF -d $INTERNET_ADDR -j DNAT --to-destination 192.168.0.1
```

将 WAN 口 8000 端口 NAT 到 192。168。100。200 的 80 端口

```
iptables -t nat -A PREROUTING -p tcp --dport 8000 -d $INTERNET_ADDR -j DNAT --to-destination 192.168.100.200:80
```

MAIL 服务器要转的端口

```
iptables -t nat -A PREROUTING -p tcp --dport 110 -d $INTERNET_ADDR -j DNAT --to-destination 192.168.100.200:110
iptables -t nat -A PREROUTING -p tcp --dport 25 -d $INTERNET_ADDR -j DNAT --to-destination 192.168.100.200:25
```

只允许 PING 202。96。134。133,别的服务都禁止

```
iptables -A Filter -p icmp -s 192.168.100.200 -d 202.96.134.133 -j ACCEPT
iptables -A Filter -j DROP
```

禁用 BT 配置

```
iptables –A Filter –p tcp –dport 6000:20000 –j DROP
```

禁用 QQ 防火墙配置

```
iptables -A Filter -p udp --dport ! 53 -j DROP
iptables -A Filter -d 218.17.209.0/24 -j DROP
iptables -A Filter -d 218.18.95.0/24 -j DROP
iptables -A Filter -d 219.133.40.177 -j DROP
```

基于 MAC，只能收发邮件，其它都拒绝

```
iptables -I Filter -m mac --mac-source 00:0A:EB:97:79:A1 -j DROP
iptables -I Filter -m mac --mac-source 00:0A:EB:97:79:A1 -p tcp --dport 25 -j ACCEPT
iptables -I Filter -m mac --mac-source 00:0A:EB:97:79:A1 -p tcp --dport 110 -j ACCEPT
```

禁用 MSN 配置

```
iptables -A Filter -p udp --dport 9 -j DROP
iptables -A Filter -p tcp --dport 1863 -j DROP
iptables -A Filter -p tcp --dport 80 -d 207.68.178.238 -j DROP
iptables -A Filter -p tcp --dport 80 -d 207.46.110.0/24 -j DROP
```

只允许 PING 202。96。134。133 其它公网 IP 都不许 PING

```
iptables -A Filter -p icmp -s 192.168.100.200 -d 202.96.134.133 -j ACCEPT
iptables -A Filter -p icmp -j DROP
```

禁止某个 MAC 地址访问 internet:

```
iptables -I Filter -m mac --mac-source 00:20:18:8F:72:F8 -j DROP
```

禁止某个 IP 地址的 PING:

```
iptables –A Filter –p icmp –s 192.168.0.1 –j DROP
```

禁止某个 IP 地址服务:

```
iptables –A Filter -p tcp -s 192.168.0.1 --dport 80 -j DROP
iptables –A Filter -p udp -s 192.168.0.1 --dport 53 -j DROP
```

只允许某些服务，其他都拒绝(2 条规则)

```
iptables -A Filter -p tcp -s 192.168.0.1 --dport 1000 -j ACCEPT
iptables -A Filter -j DROP
```

禁止某个 IP 地址的某个端口服务

```
iptables -A Filter -p tcp -s 10.10.10.253 --dport 80 -j ACCEPT
iptables -A Filter -p tcp -s 10.10.10.253 --dport 80 -j DROP
```

禁止某个 MAC 地址的某个端口服务

```
iptables -I Filter -p tcp -m mac --mac-source 00:20:18:8F:72:F8 --dport 80 -j DROP
```

禁止某个 MAC 地址访问 internet:

```
iptables -I Filter -m mac --mac-source 00:11:22:33:44:55 -j DROP
```

禁止某个 IP 地址的 PING:

```
iptables –A Filter –p icmp –s 192.168.0.1 –j DROP
```

1.5 本文总结

在 Oracle 中，有 3 种办法可以限定特定 IP 访问数据库。第一种是利用登录触发器，如下：

```
CREATE OR REPLACE TRIGGER CHK_IP_LHR
  AFTER LOGON ON DATABASE
DECLARE
  V_IPADDR    VARCHAR2(30);
  V_LOGONUSER VARCHAR2(60);
BEGIN
  SELECT SYS_CONTEXT('USERENV', 'IP_ADDRESS'),
         SYS_CONTEXT('USERENV', 'SESSION_USER')
    INTO V_IPADDR, V_LOGONUSER
    FROM DUAL;
  IF V_IPADDR LIKE ('192.168.59.%') THEN
    RAISE_APPLICATION_ERROR('-20001', 'User '||V_LOGONUSER||' is not allowed to connect from '||V_IPADDR);
  END IF;
END;
/
```

需要注意的问题：

① 触发的对象类型可以为 DATABASE，也可以为“用户名.SCHEMA”，如：

```
AFTER LOGON ON DATABASE
AFTER LOGON ON SCOTT.SCHEMA
```

② 当触发的对象类型为 DATABASE 的时候，登录用户不能拥有“ADMINISTER DATABASE TRIGGER”的系统权限；当触发的对象类型为“用户名.SCHEMA”的时候，登录用户不能拥有“ALTER ANY TIGGER”的系统权限。否则，这些用户还是会正常登录到数据库，只是将相应的报错信息写入到告警日志中。所以，拥有 IMP_FULL_DATABASE 和 DBA 角色的用户以及 SYS 和 EXFSYS 用户将不能通过这种方式限制登录。

③ 隐含参数“_SYSTEM_TRIG_ENABLED”的默认值是 TRUE，即允许 DDL 和系统触发器。当设置隐含参数“_SYSTEM_TRIG_ENABLED”为 FALSE 的时候，将禁用 DDL 和系统触发器。所以，当该值设置为 FALSE 的时候将不能通过这种方式限制登录。

第二种是修改$ORACLE_HOME/network/admin/sqlnet.ora 文件，增加如下内容：

```
TCP.VALIDNODE_CHECKING=YES   #开启 IP 限制功能
TCP.INVITED_NODES=(127.0.0.1,IP1,IP2,……)   #允许访问数据库的 IP 地址列表，多个 IP 地址使用逗号分开
TCP.EXCLUDED_NODES=(IP1,IP2,……)   #禁止访问数据库的 IP 地址列表，多个 IP 地址使用逗号分开
```

之后重新启动监听器即可。这样客户端在登录的时候会报“ORA-12537: TNS:connection closed”的错误。

需要注意的问题：

① 需要设置参数 TCP.VALIDNODE_CHECKING 为 YES 才能激活该特性。

② 一定要许可或不要禁止数据库服务器本机的 IP 地址，否则通过 lsnrctl 将不能启动或停止监听，因为该过程监听程序会通过本机的 IP 访问监听器，而该 IP 被禁止了，但是通过服务启动或关闭则不影响。

③ 当参数 TCP.INVITED_NODES 和 TCP.EXCLUDED_NODES 设置的地址相同的时候以 TCP.INVITED_NODES 的配置为主。

④ 修改之后，一定要重起监听才能生效，而不需要重新启动数据库。

⑤ 这个方式只是适合 TCP/IP 协议。

⑥ 这个配置适用于 Oracle 9i 以上版本。在 Oracle 9i 之前的版本使用文件 protocol.ora。

⑦ 在服务器上直接连接数据库不受影响。

⑧ 这种限制方式是通过监听器来限制的。

⑨ 这个限制只是针对 IP 检测，对于用户名检测是不支持的。

第 3 种是修改数据库服务器的 IPTABLES（配置文件：/etc/sysconfig/iptables）来限制某些 IP 登录数据库服务器。如下：

```
iptables -A INPUT -s 192.168.59.1/32 -i eth0 -p tcp  --dport 1521 -j DROP
service iptables save
```

则，192.168.59.1 这台主机将不能通过 1521 端口连接到数据库服务器了，会报“ORA-12170: TNS:Connect timeout occurred”的错误。

1.6 参考

1.6.1 MOS

1.6.1.1 Connecting as DBA Does not Fire RAISE_APPLICATION_ERROR in a AFTER LOGON ON DATABASE TRIGGER (文档 ID 226058.1)

Connecting as DBA Does not Fire RAISE_APPLICATION_ERROR in a AFTER LOGON ON DATABASE TRIGGER (文档 ID 226058.1).mhtml

1.6.1.2 How to Prevent Users From Log Into a Database Within Defined Periods (文档 ID 220491.1)

How to Prevent Users From Log Into a Database Within Defined Periods (文档 ID 220491.1).mhtml

1.6.1.3 ADMINISTER DATABASE TRIGGER Privilege Causes Logon Trigger to Skip Errors (文档 ID 265012.1)

ADMINISTER DATABASE TRIGGER Privilege Causes Logon Trigger to Skip Errors (文档 ID 265012.1).mhtml

第 2 章　实验中用到的 SQL 总结

```
grant ADMINISTER DATABASE TRIGGER to lhr8;
GRANT ALTER ANY TRIGGER TO LHR8;

CREATE OR REPLACE TRIGGER CHK_IP_LHR
  AFTER LOGON ON DATABASE
DECLARE
  V_IPADDR    VARCHAR2(30);
  V_LOGONUSER VARCHAR2(60);
BEGIN
  SELECT SYS_CONTEXT('USERENV', 'IP_ADDRESS'),
         SYS_CONTEXT('USERENV', 'SESSION_USER')
    INTO V_IPADDR, V_LOGONUSER
    FROM DUAL;
  IF V_IPADDR LIKE ('192.168.59.%') THEN
    RAISE_APPLICATION_ERROR('-20001', 'User '||V_LOGONUSER||' is not allowed to connect from '||V_IPADDR);
  END IF;
END;
/

set pagesize 9999
set line 9999
col NAME format a40
col KSPPDESC format a50
col KSPPSTVL format a20
SELECT a.INDX,
       a.KSPPINM NAME,
       a.KSPPDESC,
       b.KSPPSTVL
FROM   x$ksppi  a,
       x$ksppcv b
WHERE  a.INDX = b.INDX
and lower(a.KSPPINM) like  lower('%&parameter%');
alter system  set "_system_trig_enabled"=true;
alter system reset "_system_trig_enabled" scope=spfile sid='*';
iptables -I INPUT -s 192.168.59.129 -j DROP
service iptables save
```

About Me

....

- 本文作者：小麦苗，只专注于数据库的技术，更注重技术的运用
- 本文在 itpub（http://blog.itpub.net/26736162）、博客园（http://www.cnblogs.com/lhrbest）和个人微信公众号（xiaomaimiaolhr）上有同步更新
- 本文 itpub 地址：http://blog.itpub.net/26736162/viewspace-2135609/
- 本文博客园地址：http://www.cnblogs.com/lhrbest/p/6575975.html
- 本文 pdf 版及小麦苗云盘地址：http://blog.itpub.net/26736162/viewspace-1624453/
- QQ 群：**230161599**　　　微信群：私聊

- 联系我请加 QQ 好友(642808185)，注明添加缘由
- 于 2017-03-18 08:00 ~ 2017-03-18 22:00 在泰兴公寓完成
- 文章内容来源于小麦苗的学习笔记，部分整理自网络，若有侵权或不当之处还请谅解
- 版权所有，欢迎分享本文，转载请保留出处

拿起手机使用**微信客户端**扫描下边的**左边**图片来关注小麦苗的微信公众号：xiaomaimiaolhr，扫描**右边**的二维码加入小麦苗的 QQ 群，学习最实用的数据库技术。