

本TIPsは、以下の場面で活用します。

リスナーの停止を素早く検知し通知する

※ システム稼働状況によって本手順の適切値と運用影響が異なりますので、本番環境への適用時には、事前に十分な検証を実施してください。

■ リスナーの死活監視スクリプト (listener_alive_check.sh)

リスナープロセス (tnslsnr「リスナー名」) をカウントし、該当プロセスの存在有無をチェックする

引数: 1. DB名 2. リスナー名

戻り値: '0' (正常) '1' (異常)

```
$ vi listener_alive_check.sh
#!/bin/bash
#-----
# 第1引数まで同じプロセスは重複して起動されないよう制御
#-----
SCRIPT_NAME=`basename $0`
PROCESS_CNT=`ps -ef | grep ${SCRIPT_NAME} | grep ${1} | grep -v grep | grep -v 'sh -c' | grep -v ${$} | wc -l`
if [ $PROCESS_CNT -ne 0 ]; then
    echo "同じプロセスが起動中のため中断"
    echo 0;
    exit;
fi
ALERTLOG=ALERT`date '+%y%m%d'`.log
TIME=`date '+%H:%M:%S'`
DB_NAME=$1
UPPER_DB_NAME=`echo ${DB_NAME} | tr "a-z" "A-Z"`
LISTENER_NAME=$2
INDICATOR_NAME="listener down"
TARGET_STRING="tnslsnr ${LISTENER_NAME}"
INDICATOR_VALUE=`ps -ef | grep -v grep | grep -i "${TARGET_STRING}" | wc -l`
if [ $INDICATOR_VALUE -eq 0 ];
then
    echo $TIME$'\t'$INDICATOR_NAME$'\t'$INDICATOR_VALUE$'\t'2 >> $MXG_HOME/conf/${UPPER_DB_NAME}/ALERTLOG/$ALERTLOG
    echo 1;
    exit;
fi
echo 0;
sleep 1
```

■ alert.conf へのリスナーの死活監視の設定追加

リスナーの死活監視用のスクリプト(listener_alive_check.sh)を 60 秒間隔で実行しながら、戻り値が '0' でない場合 R.T.M に 'listener down' を通知する

□構文

Script<n>=<アラートの文字列>、<スクリプトの実行間隔(秒)>、<正常の戻り値>、<スクリプトファイルのフルパス> <DB名> <リスナー名>

例)「ORCL」DB の「LISTENER」リスナーの場合

```
$ cd $MXG_HOME/conf/ORCL
$ vi alert.conf

script1=listener down,60,0,$MXG_HOME/conf/ORCL/script/listener_alive_check.sh ORCL LISTENER

$ mxgctl stop ORCL RTAD
$ mxgctl start ORCL RTAD
```

 引き続き、メール設定を行う場合は MaxGauge TIPs 「[DBの死活監視を行う](#)」の4ページからご参照ください。



今回のTIPからは、以下の事を確認しました。

- ✓ リスナーの死活監視方法



MaxGauge TIPsはこちら

http://www.ex-em.co.jp/exem_lab0/exemlabo_maxgauge_tips_index.html

<お問い合わせ>

日本エクセム株式会社

TEL: 03-4530-9598

e-mail: service@ex-em.co.jp

DATABASE **ARTIST** GROUP

<http://www.ex-em.co.jp>