script, route, fping, oracle, cman

# Check Routes

```
#!/bin/bash

#VAR_DEBUG="yes";
VAR_NUMOFPINGS="5";
VAR_MWIPS="";
VAR_DB_HOSTS="dbhost1 dbhost2";
VAR_MW_HOSTS="mwhost1 mwhost2";
VAR_CMAN_HOSTS="cmanhost1 cmanhost2";
VAR_BATCH_HOSTS="batchhost1 batchhost2 batchhost3 batch4host";
#------------------------------------------------------------------------------
# Funktion:     f_fping_host()
# Beschreibung: pingt Hosts und liefert Informationen über deren Erreichbarkeit
# Return code:  0 - if all the hosts are reachable
#               1 - if some hosts were unreachable
#               2 - if any IP addresses were not found
#               3 - for invalid command line arguments
#               4 - for a system call failure
#
# Parameter:    1: remote hostname to connect to via ssh
#               2: remote hostname to ping
#------------------------------------------------------------------------------
f_fping_host() {
    # Variablen
    local L_HOSTNAME="$1";                              # Der Variable L_HOSTNAME wird der Wert des
ersten Arguments zugewiesen
    local L_REMOTEHOSTS="$2";                           # Der Variable L_REMOTEHOSTS wird der Wert des
ersten Arguments zugewiesen
    local L_REPLY="";                                   # Die Variable L_REPLY wird initialisiert#
    local L_NUMOFPING="$VAR_NUMOFPINGS";
    local L_RCONLY="";
    # Die Ausgabe von ssh wird unterdrückt. Hostkey- und Host-IP-Checking werden deaktiviert.
    # Auf dem Remote-System wird der Befehl "fping" ausgeführt.
    L_REPLY=$( ssh -q -o StrictHostKeyChecking=no -o CheckHostIP=no $L_HOSTNAME "sudo /usr/sbin/fping -c
$L_NUMOFPING -u $L_REMOTEHOSTS"; echo $?);
    L_RCONLY="$( echo -e "$L_REPLY" | tail -n 1 )";
    case "$L_RCONLY" in
        0)
            echo "$(tput setab 2)$(tput setaf 7)$(tput bold)      $(tput sgr0)";
                        echo "$(tput setab 2)$(tput setaf 7)$(tput bold)  OK  $(tput sgr0)";
                        echo "$(tput setab 2)$(tput setaf 7)$(tput bold)      $(tput sgr0)";
            ;;
        1)
            # sudo problem und aehnliches
                        # if some hosts were unreachable
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)                              $(tput
sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)  some hosts were unreachable  $(tput
sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)                              $(tput
sgr0)";
            ;;
        2)
            # if any IP addresses were not found
                        echo "$(tput setab 1)$(tput setaf 7)$(tput bold)
$(tput sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)  IP addresses were not found  $(tput
sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)                              $(tput
sgr0)";
            ;;
        3)
            # for invalid command line arguments
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)                                 $(tput
```

```
sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)  invalid command line arguments  $(tput
sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)                                    $(tput
sgr0)";
            ;;
        4)
            # for a system call failure
                        echo "$(tput setab 1)$(tput setaf 7)$(tput bold)                      $(tput
sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)  system call failure  $(tput sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)                       $(tput sgr0)";
            ;;
        255)
            # SSH Fehler
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)             $(tput sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)  SSH error  $(tput sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)             $(tput sgr0)";
            ;;
        *)
                        # Something else went wrong
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)                              $(tput sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)  Something else went wrong  $(tput sgr0)";
            echo "$(tput setab 1)$(tput setaf 7)$(tput bold)                              $(tput sgr0)";
            echo "$L_REPLY";
            ;;
    esac
}
echo
"=============================================================================================================
=======";
echo "Ermitteln der IPs der DB-Interfaces auf den Middlewaresystemen";
echo
"=============================================================================================================
=======";
for VAR_CNTR1 in $VAR_MW_HOSTS;
do
        echo "Ermittle DB-LAN-IP von ${VAR_CNTR1}"; VAR_REPLY1=$(ssh $VAR_CNTR1 "ip a"); VAR_MW_IPS+=$(
grep -oh "inet 10\.10[2-5]\.0\.[0-9]\{1,3\}" <<< $VAR_REPLY1 ); VAR_MW_IPS+=" ";
done;

VAR_MW_IPS=$(echo "$VAR_MW_IPS"|tr " " "\n"|sort|uniq|tr "\n" " ");
VAR_MW_IPS=$(printf '%s\n' "${VAR_MW_IPS//inet/}");
if [ "$VAR_DEBUG" == "yes" ];
then
        echo $VAR_MW_IPS;
fi



echo
"=============================================================================================================
=======";
echo "Ermitteln der IPs der DB-Interfaces auf den DB-Hosts";
echo
"=============================================================================================================
=======";
for VAR_CNTR2 in $VAR_DB_HOSTS;
do
    echo "Ermittle DB-LAN-IP von ${VAR_CNTR2}"; VAR_REPLY2=$(ssh $VAR_CNTR2 "ip a"); VAR_DB_IPS+=$( grep
-oh "inet 192\.168\.1[0-5]\.[0-9]\{1,3\}" <<< $VAR_REPLY2 ); VAR_DB_IPS+=" ";
done;

VAR_DB_IPS=$(echo "$VAR_DB_IPS"|tr " " "\n"|sort|uniq|tr "\n" " ");
VAR_DB_IPS=$(printf '%s\n' "${VAR_DB_IPS//inet/}");
if [ "$VAR_DEBUG" == "yes" ];
then
        echo $VAR_DB_IPS;
fi
```

```
echo
"=============================================================================================================
=======";
echo "Ermitteln der IPs der DB-Interfaces auf den BATCH-Hosts";
echo
"=============================================================================================================
=======";
for VAR_CNTR3 in $VAR_BATCH_HOSTS;
do
    echo "Ermittle DB-LAN-IP von ${VAR_CNTR3}"; VAR_REPLY3=$(ssh $VAR_CNTR3 "ip a"); VAR_BATCH_IPS+=$(
grep -oh "inet 192\.168\.5[0-6]\.[0-9]\{1,3\}" <<< $VAR_REPLY3 ); VAR_BATCH_IPS+=" ";
done;

VAR_BATCH_IPS=$(echo "$VAR_BATCH_IPS"|tr " " "\n"|sort|uniq|tr "\n" " ");
VAR_BATCH_IPS=$(printf '%s\n' "${VAR_BATCH_IPS//inet/}");
if [ "$VAR_DEBUG" == "yes" ];
then
        echo $VAR_BATCH_IPS;
fi



echo
"=============================================================================================================
=======";
echo "Ermitteln der IP des Public-Interfaces auf dem CMAN-Hosts";
echo
"=============================================================================================================
=======";
for VAR_CNTR4 in $VAR_CMAN_HOSTS;
do
    echo "Ermittle Public-LAN-IP von ${VAR_CNTR4}"; VAR_REPLY4=$(dig +noall +short +answer
${VAR_CNTR4}); VAR_CMAN_IPS+=$( grep -oh "[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}" <<<
$VAR_REPLY4 ); VAR_CMAN_IPS+=" ";
done;

VAR_CMAN_IPS=$(echo "$VAR_CMAN_IPS"|tr " " "\n"|sort|uniq|tr "\n" " ");
if [ "$VAR_DEBUG" == "yes" ];
then
        echo $VAR_CMAN_IPS;
fi



echo
"=============================================================================================================
=======";
echo "PING-Test DB-Server zu CMAN-Hosts";
echo
"=============================================================================================================
=======";

for DB_POINTER in $VAR_DB_HOSTS
do
        echo -e "${DB_POINTER}:";
        echo "----------------";
        f_fping_host "$DB_POINTER" "$VAR_CMAN_IPS";
        echo "";
done



echo
"=============================================================================================================
=======";
echo "PING-Test DB-Server zu BATCH-Hosts";
echo
"=============================================================================================================
=======";
```

```
for DB_POINTER in $VAR_DB_HOSTS
do
        echo -e "${DB_POINTER}:";
        echo "----------------";
        f_fping_host "$DB_POINTER" "$VAR_BATCH_IPS";
        echo "";
done



echo
"=============================================================================================================
=======";
echo "PING-Test DB-Server zu MIDDLEWARE-Hosts";
echo
"=============================================================================================================
=======";

for DB_POINTER in $VAR_DB_HOSTS
do
        echo -e "${DB_POINTER}:";
        echo "----------------";
        f_fping_host "$DB_POINTER" "$VAR_MW_IPS";
        echo "";
done
```