PRTG Monitoring Script

Add Service Account

```
useradd -m prtg
passwd prtg
```

GRANT SQL Access to prtg User

```
psql -U postgres -d sbcedb
CREATE USER prtg WITH PASSWORD 'xxxxxxxxxxxxxxx';
GRANT SELECT ON ALL TABLES IN SCHEMA public TO prtg;
GRANT EXECUTE ON ALL FUNCTIONS IN SCHEMA public TO prtg;
GRANT CONNECT ON DATABASE sbcedb to prtg;
```

Create Script on SBC Server

```
vi /sbin/check_sbc.sh
```

check_sbc.sh

```
#!/bin/bash
localnodeip=`ifconfig M2 |grep "inet "|sed -re 's/inet ([1].*)
netmask.*/\1/g'`
hainfo=`psql -d sbcedb -c "SELECT status FROM ha_info WHERE ha_ip =
'$(echo $localnodeip)';"|sed -re  '3! d'|sed 's/^.//'`
if [ $1 = "sipstatus" ]; then
        if [[ $hainfo == "Primary" || -z $localnodeip ]]; then
                #echo "Is Primary"
                sipstatus=`psql -d sbcedb -c "select * from
sip_server_status_info WHERE NOT hb_status = 'UP';"|tail -n +3|head -n
-2`
                #echo $sipstatus
                if [[ -z "$sipstatus" ]]; then
                        echo "All SIP Endpoints UP"
                        exit 0
                else
                        echo "SIP Status Error:"
                        psql -d sbcedb -c "select
ip_address,transport_type, port,hb_status from sip_server_status_info
WHERE NOT hb_status = 'UP';"|tail -n +3|he ad -n -2
                        exit 2
                fi

        else
                echo "Is not Primary SBC"
                exit 0
        fi
```

```
elif [ $1 = "alarmstatus" ]; then
        alarms=`psql -d sbcedb -c "select * from alarms;"|tail -n
+3|head -n -2`
        if [[ -z "$alarms" ]]; then
                echo "no pending Alarms"
                exit 0

        else
                echo "pending Alarms:"
                psql -d sbcedb -c "select
trap_severity_level,alarm_notes,alarm_timestampsecs from alarms;"|tail
-n +3|head -n -2|sed -re 's/.(.*).*\|.(.*).\|.*([ 0-9]{10}).*/echo
"Serverity: \1 Alarmtext: \2 Date: `date -d @\3`"/e'
                exit 2
        fi

elif [ $1 = "hastatus" ]; then
        if [[ $hainfo == "Primary" || $hainfo == "Secondary" ]]; then
                echo "$hainfo SBC"
                exit 0
        elif [ -z "$nodeip" ]; then
                echo "Single SBC"
                exit 0
        else
                echo "HA-Status: SBC $hainfo"
                exit 2
        fi

else
        echo "No or wrong command suffix used. Possible options:
sipstatus, alarmstatus, hastatus"
fi
```

```
mkdir -p /var/prtg/scripts
ln -s /sbin/check_sbc.sh /var/prtg/scripts/check_sbc.sh
chown -R prtg /var/prtg
chown -R prtg.prtg /sbin/check_sbc.sh
chmod 755 /sbin/check_sbc.sh
```

From:
https://wiki.buergi.org/ - **Reto's WIKI**

Permanent link:
https://wiki.buergi.org/doku.php?id=avaya:sbce:monitoring&rev=1775036007

Last update: **2026/04/01 11:33**