

Probleme mit der Internetverbindung automatisch protokollieren mit dem kabelbw-isp-logger.sh script

Ich bin seit April 2009 Kunde bei KabelBW und habe immer wieder MASSIVE Probleme mit meiner Internetverbindung. Zum Teil habe ich mal einige Stunden Ruhe, aber dann gibt es auch wieder ganze Tage oder Wochenenden an denen meine Verbindung alle 2-5 Minuten für min 60 Sekunden oder gar mehrere Minuten zusammenbricht. Während dieser Zeit ist auch kein telefonieren möglich, da KabelBW auf VoIP setzt. d.H. verliert mein Kabelmodem die Verbindung zum ISP sind immer mein Internet sowie meine Telefonleitung weg!

Nach einem extremen Wochenende in 2009 mit unzähligen massiven Ausfällen habe ich das protokollieren per Hand eingestellt und statt dessen dieses kleine shell Script entwickelt, mit dessen Hilfe eine beliebige Internetverbindung permanent rund um die Uhr überprüft und sämtliche Ausfälle protokolliert werden.

Es ist sehr einfach gestrickt. Ich prüfe einfach nur per Ping eine kleine Liste unterschiedlicher Hosts im Internet ab. Genauer DREI Hosts bei Kabel BW selbst. Warum? Ganz einfach, weil zumindest diese KabelBW Hosts bei einer stabilen Internet-Verbindung aus Sicht des Kunden praktisch IMMER online d.h. erreichbar sein müssten. Sind nun mind. drei Hosts gleichzeitig nicht erreichbar, so muss ein Problem mit der Internet Verbindung zum ISP vorliegen, da es sehr unwahrscheinlich ist, dass Zeitgleich DREI wichtige Hosts des ISPs ausfallen.

Es werden in einem Logfile die Vorkommnisse mit Datum und Uhrzeit, sowie mit der Dauer des Ausfalls (bzw der vorherigen Uptime), protokolliert.

Da ich das Script nur "bei Bedarf" und nicht immer laufen lassen möchte ist es so ausgelegt, das man es einfach ein mal als Root auf einer beliebigen Console ausführt und laufen lässt.

Hier die aktuellste Version des Scriptes:

[kabelbw-isp-logger.sh](#)

```
#!/bin/bash
#
# KabelBW Internet Ausfall Protokollierer V 2.0 (2013-05-17)
# by Axel Werner [mail@awerner.myhome-server.de]
#
#
# History:
#
#      2009-05-18      Public Release 1.1 - it now only logs CHANGES
of the last State      and does not log every bad linktest anymore.
#
Also added more
#
Hosts so there is a better fault tolerance if
one host may
```

```
# be not alive at some time. But its pretty rare
that all three
# Hosts are dead at the same time. So if all
three are dead,
# there must be a problem with the internet link
to the ISP.
#
# 2009-05-19 Added a Line for better Visibility in logfile
#
# 2009-05-20 Change: Logging Date now is been determined at
the start of
# the Sampling, not at the End. Since Pings can
take some
# Time on failing hosts the Date in the logs is
often too
# late. so in logs it seems the interupted link
was only
# seconds, while in real the interupt been often
+10secs
# longer.
#
# 2009-05-25 Fix/add: added hosts to the status display.
# Syslog now also reports which Hosts been not
available while in "good" condition.
# Raised Ping TIMEOUT from 2secs to 4secs and
made it configurable
#
# 2010-04-20 change: added automatic date to name of
logfile.
#
# 2013-05-17 change: full script rewrite - New Version 2.0
#
#
# User Config goes here
logfile=/root/kabelbw-ausfaelle-ab-`date +%F`.log
pingIntervall=5
hostsToTest="smtp.kabelbw.de dns01.kabelbw.de dns02.kabelbw.de"

# Defining bash functions..."
now() {
    date '+%F %T'
}

unixTime() {
    date '+%S'
}
```

```
getIp() {
    # function requires ONE PLAIN IP ADDRESS OR HOSTNAME for
    argument
    # it then tests if its an ip adress or a hostname and "prints
    out"
    # the IP address. If a hostname was given it resolves for the
    ip
    # and prints the ip instead.
    ipAddr=`dig +short "$1"`
    # dig +short returns empty string if argument was plain ip
    adress
    if [ -z "$ipAddr" ] ; then
        echo "$1"
    else
        echo "$ipAddr"
    fi
}

pinger() {
    # function requires ONE PLAIN IP ADDRESS + FQDN/Hostname for
    arguments
    echo "pinging: @$@"
    hostIp="$1"
    hostName="$2"

    pingStartTime=`now`

    TIMEOUT=2

    if ping -c1 -W "$TIMEOUT" "$hostIp" 2>&1 > /dev/nul ; then
        return 0
    else
        SCORE=`expr $SCORE + 1`
        echo "WARNING: $hostName ($hostIp) was not pingable at
        $pingStartTime. (SCORE=$SCORE)" | tee -a "$tmpFailedHosts"
        return 666
    fi
}

echo "$0 started on: `now` , writing log to: $logfile"

# load pc speaker kernel driver module if not loaded by default
modprobe pcspkr

# set default state
LINK="up"

# reseting up/downTimers
```

```
wentDownTimeSecs=`unixTime`
wentUpTimeSecs=`unixTime`
upTime=0;

tmpFailedHosts=`mktemp`

#while loop
while true ; do

    # resetting score
    SCORE=0

    # resetting/clearing tmp file with failed hosts
    echo > $tmpFailedHosts

    # test all Hosts listed in $hostsToTest
    for H in $hostsToTest; do
        pinger `getIp $H` $H
    done

    if [ "$SCORE" -ge 3 -a "$LINK" = "up" ]; then
        LINK="down"
        wentDownTimeSecs=`unixTime`
        upTime=`expr $wentDownTimeSecs - $wentUpTimeSecs`
        echo "KABEL BW LINK DOWN at `now`. No remote Host were
reachable. Last UPTIME WAS: $upTime sec." | tee -a "$logfile"
        cat "$tmpFailedHosts" | tee -a "$logfile"
        beep -f 2000 -n -f 1500 -n -f 1000

    elif [ "$SCORE" -eq 0 -a "$LINK" = "down" ]; then
        LINK="up"
        wentUpTimeSecs=`unixTime`
        downTime=`expr $wentUpTimeSecs - $wentDownTimeSecs`
        echo "KABEL BW LINK UP AGAIN at `now`. DOWNTIME:
$downTime sec. All remote Host are reachable. " | tee -a "$logfile"
        beep -f 1000 -n -f 1500 -n -f 2000
    elif [ "$SCORE" -ge 3 -a "$LINK" = "down" ]; then
        # just audio feedback via pc speaker that link is still
down
        beep -l 200 -f 300 -n -l 400 -f 260
    fi

    sleep "$pingIntervall"
done

exit
```

Nachtrag 15.12.2010

Bin kürzlich in einen anderen Stadtteil umgezogen, werde aber noch immer über KabelBW mit Internet und Telefonie versorgt (ging nicht anders, dank deren Knebelverträge). ABER... ich muss sagen, seit dem Umzug (bei dem ich übrigens ein neues/anderes Kabel-Modem erhalten habe) funktioniert die Verbindung bis auf ganz wenige und nur kurze Unterbrechungen SEHR GUT! Wie es scheint hat KabelBW in manchen Stadtteilen oder Straßenzügen Mega Probleme mit ihrer Intrastruktur. Traurig nur das man bei der Störungsstelle lieber die Kunden maltretiert und irgend welche "dummen" Analyse-Prozeduren immer und immer wieder durchlaufen lässt um die gemeldete Störung in dem Moment wieder fallen zu lassen, sobald mal für 5 Minuten die Verbindung wieder da ist. Hier wäre es doch sehr wünschenswert wenn sich diese Leute einfach mal DAS LOG der Kundenbeschwerden/Störungsmeldungen anschauen würden. Sie könnten leicht erkennen das hier doch ernstzunehmende Fehler auf der Leitungsstrecke vorliegen und vielleicht mal ZWEI MAL MEHR einen Techniker schicken um die Intrastruktur richtig auszumessen und zu untersuchen. Daher vertrete ich inzw. folgende Meinung: Ist die Leitung von KabelBW in Ordnung ist Internet und Telefonie über TV Kabel eine feine Sache und wirklich empfehlenswert! Hat man jedoch Pech und die Leitungen sind nicht besonders beständig, so hat es sich bei mir gezeigt, bringt es bei KabelBW wohl leider nicht wirklich etwas sich andauernd zu beschwehren und Faxe zu schreiben. Das einzige was diese Leute wirklich verstehen ist KÜNDIGEN. Und das möglichst sofort. Hat sich später die Lage geklärt kann man den Vertrag ja auch kurzfristig weiterlaufen lassen.

— [Axel Werner](#) 2010-12-15 17:39

Nachtrag 17.05.2013

So nun habe ich auch in der neuen Wohnung im neuen Stadtteil wieder massive Probleme mit dem KabelBW Internet+Telefon. Habe dazu mein altes Script noch mal überarbeitet und gestartet um ein aktuelles Protokoll zu erhalten. Es sieht so aus als würde alle 5-10 Minuten die FritzBox von KabelBW automatisch einen reboot machen. Das dauert dann meist knapp 2 Minuten bis eine Internet-Verbindung wiederhergestellt werden konnte. Morgen werde ich das entstandene Protokoll an KabelBW einsenden.

— [Axel Werner](#) 2013-05-17 19:56

[bash](#), [shell](#), [linux](#), [isp](#), [kabelbw](#), [scripting](#), [log](#), [ping](#), [link](#), [internet](#), [vorsicht](#), [kunde](#), [provider](#), [service](#)

From: <https://www.awerner.myhome-server.de/> - Axel Werner's OPEN SOURCE Knowledge Base

Permanent link: <https://www.awerner.myhome-server.de/doku.php?id=it-artikel:linux:probleme-mit-der-internetverbindung-automatisch-protokollieren-mit-dem-kabelbw-isp-logger.sh-script>

Last update: 2019-05-18 14:39

