10 GE Paketbeschleuniger

Übung: einen 10 GE Link zwischen 2 Switchen mittels zweier 1 GE Ports auslasten

Wenn man der blinkenden Link-Leuchte am Switch nicht traut, die behauptet einen funktionsfähigen 10 GE-Link anzuzeigen, dann kann man mittels eines Paketbeschleunigers mal testen, ob da wirklich 10GE durch gehen.

Der Trick: 20 VLANs einrichten, per Patchkabel jeweils abwechselend auf einem Switch bridgen und mit den 2 Rechnern in unterschiedlichen VLANs Traffic erzeugen

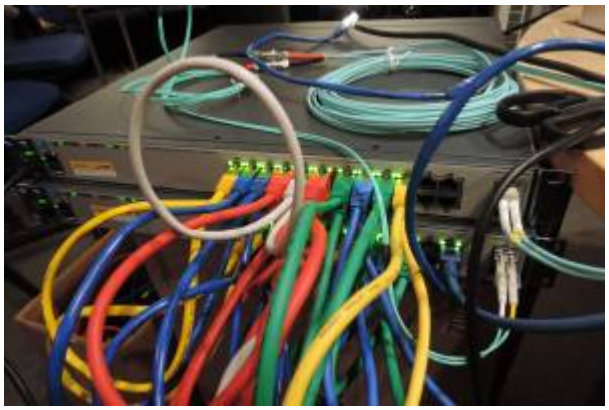
Notwenige Hardware

- 2x HP 3800 oder 5400 mit SFP+ und min. 20 Kupferports
- 2x 10 GE SFP+ GBICs
- 2 Rechner mit Linux, Gbit-fähig
- Glas-Patchkabel und Adapter für einen LC-LC-Direktlink
- 20 kurze RJ45-Patchkabel
- evtl. serielles Kabel + Adapter zum Switch konfigurieren

Aufbau

- 2 HP-Switche mit Strom und 10 GE-GBICs bestücken
- GBICs mit Patchkabel verbinden
- Kupferpatches (Ports auf einem Switch kurzschließen)
    - **Switch A**
        - Port 1-2
        - Port 3-4
        - usw.
        - Port 19-20
    - **Switch B**
        - Port 2-3
        - Port 4-5
        - usw.
        - Port 18-19
- Rechner an **Switch B, Port 20 und 21 (oder höher)** anschließen

Bilder







Switch-Config

- auf **beiden** Switchen
  - VLANs 2-20 anlegen
  - Port 1 → DEFAULT_VLAN untagged
  - Port 2 → VLAN 2 untagged
  - usw.
  - Port 20 → VLAN 20 untagged
  - 10 GE Port → DEFAULT_VLAN untagged, **Alle anderen VLANS tagged**
- auf **Switch B** für die beiden Rechner
  - Port 20 → VLAN 20 untagged
  - Port 21 → DEFAULT_VLAN untagged
  - Mgmt-IP: 10.0.0.3 (Netzmaske: 255.255.255.0) im DEFAULT_VLAN

Traffic erzeugen

1. beiden Rechnern je eine IP anlegen

Rechner 1

```
ip link set dev eth0 up
ip addr add 10.0.0.1/24 dev eth0
```

Rechner 2

```
ip link set dev eth0 up
ip addr add 10.0.0.2/24 dev eth0
```

2. auf Rechner 1 einen Server starten

Daten empfangen

```
nc -l -p 1234 > /dev/null
```

3. auf Rechner 2 einen Client starten und Daten reinkippen

Daten senden

```
cat /dev/zero | nc 10.0.0.1 1234
```

4. Portcounter beim Überlaufen zugucken ;)

Portauslastung anzeigen

- Switch A oder B eine IP im DEFAULT_VLAN geben
  - 10.0.0.3/24
- auf Rechner im DEFAULT_VLAN
  - Script (siehe unten) aufrufen (benötigt SNMP-Tools, unter Debian „apt-get install snmp“)

Portauslastung anzeigen

```
# Usage: -h <ip vom switch> -s <snmp communitystring> -i <interface-nr des
10GE Ports>
./snmp_bandwidth.sh -h 10.0.0.3 -s public -i 25
```

erwartete Ausgabe

Ausgabe der Portauslastung

```
Inbound: 10501 Mbps, Outbound: 10498 Mbps, Speed: 10000 Mbps
Inbound: 10452 Mbps, Outbound: 10450 Mbps, Speed: 10000 Mbps
```



Bashscript

snmp_bandwidth.sh

```
init () {
INTNUMBER=$interface
OUT=$(snmpget -v2c -c $community_string $hostname ifHCOutOctets.$INTNUMBER |
awk '{print $4}')
IN=$(snmpget -v2c -c $community_string $hostname ifHCInOctets.$INTNUMBER |
awk '{print $4}')
#SPEED=$(snmpget -v2c -c $community_string $hostname ifSpeed.$INTNUMBER |
awk '{print $4}')
SPEED=10000000000
TIME=1

        if [ -z "$OUT" ] || [ -z "$IN" ]; then
                msg="Unable to retrieve SNMP info."
                state=CRITICAL
                echo $state $msg
                exit 2
```

```
        fi
}

check () {
                #wait $TIME before running the same check, this way we can
confirm how much the data has changed in two periods.
                sleep $TIME
                OUT2=$(snmpget -v2c -c $community_string $hostname
ifHCOutOctets.$INTNUMBER | awk '{print $4}')
                IN2=$(snmpget -v2c -c $community_string $hostname
ifHCInOctets.$INTNUMBER | awk '{print $4}')
                DELTAOUT=$(( $OUT2 - $OUT))
                DELTAIN=$(( $IN2 - $IN))
                #Value is in octets so will need to be divided by 8 to get
bytes, this is then divided by 1024 to give kilobytes.
                INPUTBW=$((($DELTAIN)/$TIME)*8))
                OUTPUTBW=$((($DELTAOUT)/$TIME)*8))

                #For percentage usage we do 100/(total possible bandwidth –
current bandwidth).
                percentage_use=$(echo "scale=9; $INPUTBW/$SPEED" | bc)
                PRCNTIN=$(echo "scale=0; 100*$percentage_use" | bc)
                percentage_use=$(echo "scale=9; $OUTPUTBW/$SPEED" | bc)
                PRCNTOUT=$(echo "scale=0; 100*$percentage_use" | bc)

                echo "Inbound: " $((INPUTBW/1000000)) "Mbps ($PRCNTIN%
Used), Outbound: " $((OUTPUTBW/1000000)) "Mbps ($PRCNTOUT% Used), Speed: "
$(($SPEED/1000000)) "Mbps"

                IN=$IN2
                OUT=$OUT2
}

while getopts ":i:s:h:" option
do
case $option in
i)interface=$OPTARG
;;
s)community_string=$OPTARG
;;
h)hostname=$OPTARG
;;
*) echo "Syntax is $usage -h <hostname> -s <snmpstring> -i <interface-nr>"
exit 1;;
esac
done

if [ -z "$hostname" ]; then
        echo "-h) IP address required."
        exit 1
elif [ -z "$community_string" ]; then
```

```
        echo "-s) snmp string needs to be specified."
        exit 1
elif [ -z "$interface" ]; then
        echo "-i) interface number required."
        exit 1
else
        init
        while true
        do
            check
        done
fi
```

From:
https://wiki.fem.tu-ilmenau.de/ - **FeM-Wiki**

Permanent link:
https://wiki.fem.tu-ilmenau.de/public/technik/howto/10ge-paketbeschleuniger

Last update: **2015/02/07 12:34**