

MPLS (mérési utasítás)

Mihály Zsigmond dr.



A Multi Protocol Label Switching eljárást Linux környezetben mutatja be a mérés. Az MPLS-ről jó összefoglaló olvasható a [Wikipédián](#), a nyílt forráskódú mpls-linux projekt honlapja a [SourceForge](#)-n található, az onnan letöltött rendszert használjuk a mérésben. A mérés során egy jellegzetes forgalomszabályozási probléma hagyományos, és az MPLS segítségével történő, azaz automatizálható megoldását kell összehasonlítani.

Segédanyagok a felkészüléshez

[MPLS alapok](#)

[mplsadm használata](#)

[iptables + MPLS patch használata](#)

Article Index

[MPLS \(mérési utasítás\)](#)

[Előkészületek](#)

[Feladatok :: IP](#)

[Feladatok :: MPLS](#)

[Kérdések](#)

[All Pages](#)

Előkészületek

A mérést virtuális környezetben fogjuk végezni. A méréshez a drbl rendszer bármelyik kliens gépe megfelel (érdemes "erős" gépet választani), de csak az itt megadott felhasználó tudja futtatni a szükséges programokat. Tehát a bejelentkezéshez:

```
user: mplsmemes, password: mplsmemes
root password: meres
```

A root futtassa a következő két parancsot:

```
mknod -m 0600 /dev/vmmon c 10 165
/etc/init.d/vmware restart
```

(Ha "virtual ethernet failed" üzenetet kapunk, az nem baj.)

Indítsuk el a "vmware workstation" programot a menüből: Applications >> System >> Vmware Workstation. Néhány másodperc múltán jelenik meg az ablak, melyben a méréshez használt 3 gép, és a belőlük képzett hálózat látható.

(Probléma esetén a /home/mplsmemes/vmware/Mpls_meres/Mpls_meres.vmtm konfigurációs fájl kell betölteni.)

A 3 ethernet kapcsolatot úgy konfiguráltuk, hogy az mpls01-mpls02 összeköttetés 10Mbit/sec sebességű, a másik két összeköttetés pedig 100Mbit/sec-os.

A virtuális gépek a "Power on this team" gombbal indíthatók. A három virtuális gépen ugyanaz a konfiguráció: két operációs rendszer van telepítve, a méréshez használt OS automatikusan indul. Login paraméterek:

Mpls_meres OS: meres (meres), root (mpls) <= ezt használjuk! Mivel a mérés során használandó parancsokhoz root jogosultság kell, érdemes mindjárt root-ként bejelentkezni.

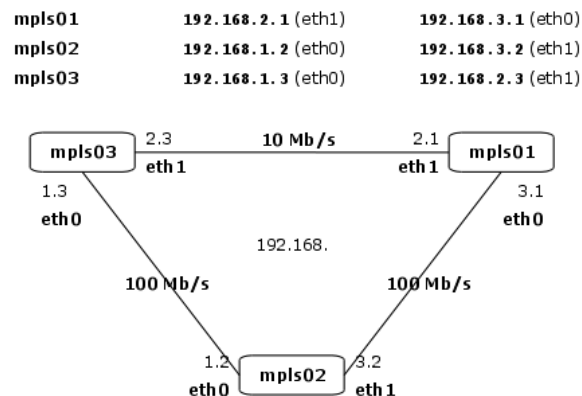
(Debian: server (test), root (server) : ezt nem használjuk.)

A virtuális gépek kezelése először furcsa lesz, de hamar meg lehet szokni. A szokásos terminológiában "host" vagy "host OS" az anya-gép, illetve az azon futó operációs rendszer, "guest" pedig a virtuális gép megnevezése. Az input perifériák (billentyűzet, egér) jelét alapértelmezésben a "host" rendszer kapja. A VMware programban az egérrel lehet kiválasztani, hogy melyik "guest" képernyőjét látjuk. A "guest" képernyőjére klikkelve az inputot a "guest"-re irányítjuk (egér most nem használható). A Ctrl+Alt kombinációval lehet az inputot visszairányítani a "host"-ra.

Majd a **mérés végén** a virtuális gépeken halt parancsot kell kiadni a kikapcsolás előtt.

Mérési feladatok (IP)

A mérés során egy forgalomszabályozási probléma megoldását kell megvizsgálni tisztán IP és IP over MPLS környezetben. A probléma a következő: egy forgalmi folyam által igényelt sávszélesség megnövekedik, és az eddig e folyamat kiszolgáló útvonal biztosította kapacitás elégtelennek bizonyul. A folyamatot egy elegendő szabad kapacitással rendelkező alternatív útvonalra kell átirányítani.



Az ábrán látható mérési elrendezésen ezt a következőképpen idézhetjük elő: a 10 Mbit/s-os Ethernet kapcsolaton annak kapacitásánál nagyobb volumenű forgalmat próbálunk átvinni az [iperf](#) forgalomgenerátor segítségével (amellyel egyúttal mérni is

tudjuk az átvitt adatmennyiséget). Ezt az alaproblémát kétféleképpen kell a mérés során megoldani: egyrészt az IP routing módosításával, majd az eredeti állapot visszaállítása után MPLS segítségével.

Előkészítésként szükséges, hogy a kernelben kikapcsoljuk a spoof-protection-t az **mpls03** és **mpls01** gépeken. Erre azért van szükség, mert bekapcsolt spoof-protection mellett a kernel egy adott interface-en csak olyan beérkező csomagokat fogad el, amelyek feladójának címzett csomagot a routing tábla alapján az érkezősi interface-en kellene kiküldeni. A spoof-protection állapotáról a `/proc/sys/net/ipv4/conf/{all,default,eth0,eth1}/rp_filter` tartalma tájékoztat, és változtatni is itt lehet rajta (root jogokkal írható/olvasható).

1. Kapcsolja ki az mpls01 és mpls03 gépeken a spoof-protection-t!

Hatékony megoldás:

```
for i in /proc/sys/net/ipv4/conf/*/rp_filter; do echo 0 > $i; done;
```

2. Generáljon 20Mb/s UDP adatforgalmat mpls01 és mpls03 között!

Az iperf program egy kliens-szerver alkalmazás. Először a szerveret indítjuk az mpls03 gépen:

```
mpls03: iperf -s -u (server mode, UDP)
```

Azután a kliens az mpls01 gépen:

```
mpls01: iperf -c 192.168.2.3 -b 20M -u (client mode + server IP, speed, UDP)
```

Meg kell várni, míg a kliens befejezi a tesztet, azután a szerveren lehet megnézni az eredményt.

3. IP szinten irányítsa a forgalmat az mpls02 gépen keresztül!

Mielőtt belefogna, nézze meg a gépek routing tábláját (route vagy ip route : eltérő formátumban, de ugyanazt mutatja a két parancs). A megoldáshoz módosítani fogjuk a gépek routing tábláját.

A végrehajtáshoz a következőket kell tennünk (nem feltétlen ebben a sorrendben!):

Az mpls01 gépen az iperf klienssel a 192.168.1.3 címen kapcsolódunk a szerverre.

Az mpls01 géppel tudatjuk, hogy a 192.168.3.2 címen keresztül lehet elérni a 192.168.1.0 hálózatot.

```
mpls01: ip route add 192.168.1.0/24 via 192.168.3.2
```

Ellentétes irányban az mpls03 routing tábláját is átírjuk.

Az mpls02 gépen be kell kapcsolni az IP forwardingot (`/proc/sys/net/ipv4/ip_forward`).

A ping -R paranccsal ellenőrizze, hogy minden rendben van-e, aztán végezze el a mérést az iperf paranccsal.

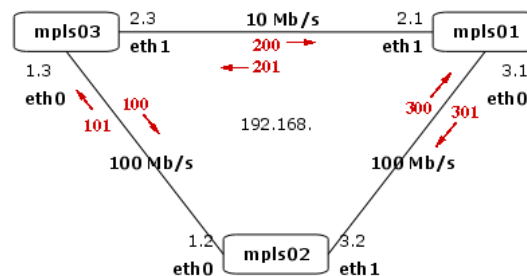
4. Állítsa vissza a routing táblákat, állítsa le a forgalomgenerátort!

Mérési feladatok (MPLS)

5. Hozzon létre mpls tunnelt az mpls02 gépen keresztül!

A feladatot úgy fogjuk megoldani, hogy az mpls01 és mpls03 gépek változatlan IP címen fogják egymást látni, de a csomagok a nagyobb sebességű összeköttetésen fognak közlekedni.. A címke kiosztás önkényes, az ábrán pirossal jelöltük. A parancsokat az [mplsadm használata](#) leírás alapján állítottuk össze.

mpls01	192.168.2.1 (eth1)	192.168.3.1 (eth0)
mpls02	192.168.1.2 (eth0)	192.168.3.2 (eth1)
mpls03	192.168.1.3 (eth0)	192.168.2.3 (eth1)



mpls01:

Lekapcsoljuk az eredeti Ethernet interfészt:

```
ifconfig eth1 down
```

A routing táblából automatikusan törölődnek az interfészhez tartozó bejegyzések.

Az mpls02 felé menő interfészt a nullás címterhez rendeljük:

```
mplsadm -L eth0:0
```

Létrehozuk a kimenő csatornát mpls02 felé:

```
mplsadm -A -O 0 -o push:gen:301:set:eth0:ipv4:192.168.3.2
```

A parancs egy hexa kóddal tér vissza, ami a létrehozott csatorna azonosítója (estünkben 0x2).

Létrehozuk a tunnelt:

```
mplsadm -A -T mpls0
```

Az így létrehozott mpls0 interfész ugyanúgy használható, mint például eth0.

A létrehozott interfészhez rendeljük az IP címet:

```
ifconfig mpls0 192.168.2.1 netmask 255.255.255.255
```

A létrehozott interfész kimenő forgalmát a csatornába irányítjuk:

```
mplsadm -B -O 0x2 -T mpls0
```

```
A 192.168.2.0 alhálózat az új interfészen lesz elérhető:
route add -net 192.168.2.0/24 mpls0
A 300-as címmel várunk érkező csomagokat
mplsadm -A -I gen:300:0
```

Az mpls aktuális beállításait a /proc/net/mpls_ * fájlokban lehet megnézni.

Az ellen-állomáson végrehajtandó parancsok:
mpls03:

```
ifconfig eth1 down
mplsadm -L eth0:0
mplsadm -A -O 0 -o push:gen:100:set:eth0:ipv4:192.168.1.2 (0x2)
mplsadm -A -T mpls0
ifconfig mpls0 192.168.2.3 netmask 255.255.255.255
mplsadm -B -O 0x2 -T mpls0
route add -net 192.168.2.0/24 mpls0
mplsadm -A -I gen:101:0
```

Az átjátszó állomáson:
mpls02:

```
mplsadm -L eth0:0
mplsadm -L eth1:0
mplsadm -A -O 0 -o push:gen:101:set:eth0:ipv4:192.168.1.3 (0x2)
mplsadm -A -O 0 -o push:gen:300:set:eth1:ipv4:192.168.3.1 (0x3)
mplsadm -A -I gen:100:0
mplsadm -A -I gen:301:0
```

A következő két parancs végzi az átírányítást:

```
mplsadm -B -I gen:100:0 -O 0x3
mplsadm -B -I gen:301:0 -O 0x2
```

6. Generáljon 20Mb/s adatforgalmat a két gép között!

Ilyet már csináltunk, nem okozhat gondot. Mérje meg a csatorna sebességét.

Nem fog működni a dolog: valami még hiányzik. Tessék próbálkozni a ping -R paranccsal, a tcpdump-pal vizsgálható a forgalom.

7. Építse ki a 200-201 címekkel működő ágot!

Kérdések

Mi az MPLS működésének a lényege?
Mi az LSP?
Mi az LSR/LER, mi a különbség köztük és hogyan működnek?
Mit jelent az MPLS forwarding instruction? Miért van rá szükség?

A mérést kidolgozta: Király Csaba, Kálmán Barnabás és Pándi Zsolt