

Forgalomirányítás tcng és HTB segítségével HOGYAN

1.0 változat

Martin A. Brown

SecurePipe, Inc. (<http://www.securepipe.com>)
Network Administration

mabrown@securepipe.com

Tartalomjegyzék

1. Szerzői jog és licenc.....	2
1.1. Magyar fordítás	2
2. Bevezetés	2
2.1. A forgalomirányítás fogalma és működése	2
2.2. Mi az a HTB?	3
2.3. Mi is az a tcng ?	4
3. Követelmények	4
3.1. Rendszermag követelmények.....	4
3.2. tc követelmények	5
3.3. tcng követelmények	5
4. Beállítási példák	5
4.1. tcng használata csak a letöltések alakításához	6
4.2. A kétsebességű háromszínű mérőóra használata	8
5. Egyéb megjegyzések.....	10
6. További leírások és hivatkozások.....	10

1. Szerzői jog és licenc

© 2003, Martin A. Brown

A dokumentum szabadon másolható, terjeszthető és módosítható, az FSF által közzétett GNU Szabad Dokumentációs Licenc v1.1, vagy annak későbbi változatában szereplő feltételek szerint. A licenc egy másolata megtalálható a www.gnu.org/copyleft/fdl.html (<http://www.gnu.org/licenses/fdl.html>) honlapon.

1.1. Magyar fordítás

A magyar fordítást Völgyi Péter ([mailto:petvolgyi\[kukac\]freemail\[pont\]hu](mailto:petvolgyi[kukac]freemail[pont]hu)) készítette (2004.02.29). A lektorálást Daczi László (mailto:dacas@freemail.hu_NO_SPAM) végezte el (2004.03.11). A dokumentum legfrissebb változata megtalálható a Magyar Linux Dokumentációs Projekt (<http://tldp.fsf.hu/index.html>) honlapján.

2. Bevezetés

Ez a dokumentum rövid áttekintést ad a **tcng** (Traffic Control - Next Generation; következő generációs forgalomirányítás (<http://tcng.sourceforge.net>)) HTB-vel (Hierarchical Token Bucket (<http://luxik.cdi.cz/~devik/qos/htb/>)) való használatáról, mely eszközök segítségével a Linux rendszeren forgalomszabályozást végezhetünk.

Ez a gyakorlatsor elsősorban olyan rendszergazdáknak készült, akik:

- LEGALÁBB alapszintű forgalomirányítási ismeretekkel rendelkeznek,
- forrásból tudják FORDÍTANI az iproute2 és a tcng csomagokat, vagy Ugyanezen csomagokat SRPM-ből telepíteni,
- Az általuk használt rendszer moduláris szerkezetű rendszermagja támogatja a HTB és DSMARK modulokat, VAGY képesek rendszermagot fordítani ezeknek a támogatására.

Megjegyzés: Ez a doksi nem átfogó és nem mértékadó. A szerző várja mind a pozitív, mind a negatív véleményeket a [<mabrown@securepipe.com>](mailto:mabrown@securepipe.com) e-mail címen. A helyreigazításokat, kiegészítéseket is szívesen veszi.

2.1. A forgalomirányítás fogalma és működése

A forgalomirányítás egy hálózaton, vagy hálózati eszközön történő teljes csomag-várakozósor (packet queuing) alrendszer összefoglaló neve. A forgalomirányítás több műveletből tevődik össze. Az osztályozás (classifying) során történik a csomagok azonosítása, majd egyedi várakozósorokba, osztályokba rendezése. Az engedélyezés (policing)

az osztályozásnak megfelelő adatfolyamban a csomagok vagy bájtok számának szabályozását jelenti. A besorolás (scheduling) a döntéshozatali elem, ahol is a csomagokat az adattovábbításhoz sorrendbe, majd újra sorrendbe rakjuk. Az alakítás (shaping) az a folyamat, amikor a csomagokat késleltetjük majd továbbküldjük, ezáltal az átviteli sebesség nagyjából egyenlő ütemű, és előre kiszámítható lesz.

A forgalomirányítás sokoldalúságát kihasználva komplex rendszereket lehet kialakítani, annak érdekében, hogy egy bizonyos folyamat (vagy alkalmazás) részére sávszélességet biztosítsunk, vagy a rendelkezésre álló sávszélességet korlátozzuk bizonyos folyamatok, vagy alkalmazás előtt.

A forgalomirányítás egyik kulcs-konceptiója a tokenek (jelek) fogalma. Az engedélyezés vagy alakítás folyamata során ki kell számítani az elküldött bájtok vagy csomagok számát, vagyis melyik, milyen sebességgel kerül majd továbbításra. Minden csomag vagy bájt (az alkalmazástól függ, hogy melyik) egy jelnek felel meg, és az engedélyező- vagy alakító-alkalmazás csak akkor továbbítja a csomagot vagy bájtot, ha az rendelkezik a megfelelő jellel. Egy alkalmazás metaforikus jeltárolója a tároló (bucket). Röviden a tároló jelzi, mind az egyszerre használható jelek számát (ez a tároló mérete), mind pedig a jelek újratöltődésének ütemét (milyen gyorsan töltődik újra a tároló).

Bővebben a 2.2 példában találisz a tároló használatára a Linux forgalomirányítási rendszerben.

Linux rendszereken a forgalomirányítás mindig is komoly kihívást jelentett. A **tc** parancssori eszköz kapcsolatot jelent a rendszermag azon részeivel, amelyek az alakítás, besorolás, engedélyezés, és osztályozás folyamatait végzik. A parancs szintaxisa viszont rendkívül bonyolult. A **tcng** projekt azzal, hogy a **tc** parancssori eszköz fölé nyelvi réteget fektet, sokkal barátságosabb környezetet biztosít a felhasználónak. A **tcng**-ben írt forgalomirányítási beállítások ezáltal könnyebben kezelhető, kevésbé bonyolult, és - ami talán a legfontosabb - hordozhatóbb lesz.

2.2. Mi az a HTB?

A Hierarchichal Token Bucket (<http://luxik.cdi.cz/~devik/qos/htb/>) a CBQ-nál egyszerűbb konfigurációs paraméterekkel rendelkező csoportnyi qdisc, melyet Martin Devera alkotott meg. A HTB-ről és használatáról elég sok dokumentum található a szerző honlapján, valamint Stef Coene weboldalán (<http://www.docum.org/>) is. Következzen egy rövid kis ismertető a HTB rendszerről.

Alapvetően a HTB egy tetszőleges számú, hierarchikusan felépített jeltároló (token bucket) (...igen, erre könnyen rájöhettél magadtól is). Tételezzük fel a legegyszerűbb esetet: Bármilyen eszközön az elsődleges kilépési sor a **root** qdisc.

A **root** qdisc egyetlen osztályt fog tartalmazni (komplex esetekben több osztály is csatolódik a **root** qdisc-hez). Az egyszerű HTB osztályt két paraméter jellemzi: a **rate** (mérték) és a **ceil** (plafon). A top-level (legfelsőbb szintű) osztálynál ezek az értékek meg kell, hogy egyezzenek és a kapcsolat teljes sávszélességét mutatják.

A HTB-nél, a **rate** jelenti egy osztály számára a garantált sávszélességet, a **ceil** pedig a maximális sávszélességet. A **rate** és a **ceil** közötti sávszélességet mindig a szülő osztálytól veszi az alkalmazás, vagyis a legfelső szinten lévő osztálynál a **rate** és a **ceil** ugyanannyi lesz.

Számos gyerek-osztály létesíthető ez alatt az osztály alatt, kezdetben mindegyikhez némi sávszélességet rendelve a szülő-osztály meglévő készletéből. Ezekben a gyerek-osztályokban már nem kell a **rate** és a **ceil** paramétereknek megegyezniük, mint a szülő osztálynál. Ez lehetővé teszi, előre meghatározott sávszélesség tartalékolását bizonyos osztályoknak. Lehetővé válik továbbá a HTB számára, hogy összehasonlítsa a meglévő sávszélesség eloszlását az osztályok sávszélességeivel. A következő példa talán rávilágít a lényegre:

A Hierarchical Token Bucket magában foglal egy osztálynyi besoroló mechanizmust a Linuxon való forgalomirányításhoz, és ott van még a felhasználónak a **rate** és a **ceil** paraméter az egyes osztályok abszolút

sávszélességének szabályozásához, valamint, hogy jelezze a sávszélesség eloszlását, amikor extra sávszélesség adódik (egészen a `ceil`-ig).

Jegyezzük meg, hogy a top-level osztály sávszélességének a beállításakor a forgalom alakításnak (shaping) csak akkor lesz értelme, ha a szűk keresztmetszet a hálózat és az Internet között az a gép, amin dolgozunk. Tipikusan ilyen eset az otthoni vagy irodai hálózati környezet, ahol az egész helyi hálózatot DSL vagy T1 kapcsolatok kötik össze.

Gyakorlatilag ez annyit jelent, hogy a top-level osztály sávszélességét úgy kell beállítani, hogy vesszük az egész, rendelkezésre álló sávszélességet és kivonunk belőle egy keveset.

2.3. Mi is az a tcng?

A `tcng` (Traffic Control - Next Generation, következő generációs forgalomirányítás) (<http://tcng.sourceforge.net/>) Werner Almesberger projektje, erőteljes, egységes és egyértelmű nyelvezet megteremtésére, amellyel a forgalomirányítási szerkezetek pontosan megfogalmazhatók. A **tcng** csomagban található **tcc** fordító alakítja át a `tcng` nyelvet többféle kimeneti formára. Alapesetben a **tcc** beolvas egy fájlt (amely argumentumként, vagy STDIN-ként adható meg) és a STDOUT kimenetre küld egy sorozat **tc** parancsot, amelyek az elképzelt forgalomirányítási szerkezetet a rendszermagban létrehozzák. (lásd az **iproute2** fejezetet lentebb)

A támogatott rendezési elveket megtalálod a **tcng Queuing discipline parameters** (<http://linux-ip.net/gl/tcng/node159.html>)(Útmutató a **tcng** paramétereikhez) dokumentumban. A `tcng` projekthez a HTB támogatást Jacob Teplitsky írta, aki aktív tag a LARTC levelezési listán (<http://lartc.org/#mailinglist>).

A **tcc** eszköz sokféle kimenetet tud produkálni, ez a dokumentum azonban csak a legáltalánosabb és alapértelmezett kimeneti eredménnyel foglalkozik. A TCNG kézikönyv (<http://linux-ip.net/gl/tcng/>) foglalkozik bővebben a **tcng** parancs használatával.

A **tcsim** eszköz képes szimulálni az adott forgalomirányítási szabályokkal beállított rendszermag viselkedését, mégpedig úgy, hogy `tcng` beállítófájlokat olvas és értelmez. Habár a **tcsim** elég jelentős részét képezi a **tcng** projectnek, itt többet nem foglalkozunk vele.

3. Követelmények

Van néhány dolog, ami szükséges ahhoz, hogy a rendszermag támogassa a HTB-t és DSMARK-ot, a `tc` támogassa a HTB-t és a DSMARK-ot, és, hogy a `tcng` működjön.

Ahhoz, hogy ez a dokumentum használható legyen, feltétlenül kell egyrészt egy olyan rendszermag, amely támogatja a HTB-t, másrészt a `tc` alkalmazás (elég csak a doksi címére utalni). A DSMARK támogatás, a szó szoros értelmében nem feltétel, bár néhány példa (különösen az osztály kiválasztási útvonal, de lehet, hogy más példák is) nem biztos, hogy működik e nélkül.

3.1. Rendszermag követelmények

A rendszermag követelményeknek egyszerű megfelelni. A 2.4.20 és újabb rendszermagok beépített HTB és DSMARK támogatást tartalmaznak. Szóval egyszerűen csak be kell kapcsolni ezeket az opciókat (QoS/FAIR Queing szekció). A rendszermag beállításáról bővebben olvashatsz a DiffServ project (<http://diffserv.sourceforge.net/#24>) honlapján.

2.4.20-nál régebbi rendszermagok esetében ez a link (<http://luxik.cdi.cz/~devik/qos/htb/v3/htb3.6-020525.tgz>) egy folt (patch) a 2.4.17 vagy újabb kernelhez.

3.2. tc követelmények

A **tc** parancs az **iproute2** eszközcsoomag része. Az **iproute2** általános leírása a **iproute2** kézikönyv (<http://linux-ip.net/>) <http://linux-ip.net/</ulink>http://linux-ip.net/gl/ip-cref/> oldalakon található. A csomagot közvetlenül az Alexey Kuznyecov FTP archívuma (<ftp://ftp.inr.ac.ru/ip-routing/>) webhelyről lehet letölteni, de a legtöbb disztribúcióban megtalálható csomagként is. Ha RPM csomagkezelőt használasz, erről a SRPM (<http://linux-ip.net/traffic-control/iproute-2.4.7-7.src.rpm>) webhelyről letöltheted a forrást, majd lefordíthatod a rendszernek megfelelően.

Ha az **iproute2** programot forrásból telepíted, használható Martin Devera HTB (<http://luxik.cdi.cz/~devik/qos/htb/>) webhelyen található **tc** programhoz szükséges folt (<http://luxik.cdi.cz/~devik/qos/htb/v3/htb3.6-020525.tgz>). Erre azért van szükség, hogy a **tc** programban legyen HTB támogatás.

A **tc**-nek támogatnia kell továbbá a dsmark-ot, a diffserv jelölő mechanizmust. Szerencsére ez egyszerűen megoldható, az **iproute2** forrásában található `Config` fájlban az alábbiakat kell megváltoztatni: a `TC_CONFIG_DIFFSERV=n` sort `TC_CONFIG_DIFFSERV=y`-ra kell cserélni, majd a csomagot újra kell fordítani.

Az SRPM (<http://linux-ip.net/traffic-control/iproute-2.4.7-7.src.rpm>) dsmark és HTB támogató **tc** binárist készít, így az alábbi példáknál már nem lesz gond.

3.3. tcng követelmények

Talán a telepítés legkönnyebb része a **tcng** támogatás megoldása. Csak ki kell csomagolni a forrást, és futtatni kell a `./configure --no-tcsim` parancsot fordítás előtt.

Ha RPM-alapú Linux fut a gépen, használhatod a SPEC fájlt `tcng/build/tcng.spec` a fordításhoz, vagy letöltheted és fordíthatod ezt az SRPM-t (<http://linux-ip.net/traffic-control/tcng-9d-1.src.rpm>). Az SRPM két csomagot készít: `tcc` és `tcc-devel`. Csak a `tcc`-re lesz szükség a beállításokhoz.

A `tcc` fordító használatához szükség van még a **c++** csomagra is. Ezt a **tcc** használja.

4. Beállítási példák

Az itt bemutatásra kerülő példák az ebben a könyvtárban (<http://linux-ip.net/code/tcng/>) található, letölthető konfigurációs példák módosított változatai.

Ezek a példák használhatók önálló beállító fájlként a **tcc** fordító segítségével, vagy a példaként szereplő SysV indító szkript (<http://linux-ip.net/code/tcng/tcng.init>) használatával. Az indítószkript a raptor által a LARTC levelezőlistán közzétett (<http://mailman.ds9a.nl/pipermail/lartc/2002q4/005411.html>) szkript egy változata.

Amennyiben a fenti indítószkriptet használod, vess egy pillantást a következő példára: `/etc/sysconfig/tcng`:

Példa 1. `/etc/sysconfig/tcng`

```
# - tcng köztes-konfigurációs fájl
```

```
# (Soha nem használok köztes konfigurációs fájlt, nem szeretem)

#
# -- 2003-03-15 created; -MAB
# -- 2003-03-31 modified to allow ENVAR override; -MAB
#
# -- ez a könyvtár fogja tartalmazni az összes erre a host-ra
# vonatkozó tcng beállítást

TCCONFBASEDIR=${TCCONFBASEDIR:-/etc/sysconfig/tcng-configs}
#
# -- ez az aktív tcng beállítófájl, FIGYELEM! mivel a tcng támogatja a
# az #include szerkezet használatát, a $TCCONFBASEDIR
# változóban tárolt beállítófájlba beépíthetők a konfigurációs
# modulok.

#
TCCONF=${TCCONF:-$TCCONFBASEDIR/global.tcc}

tcstats=${tcstats:-no} # -- statisztikai jelentések kikapcsolása
tcstats=${tcstats:-yes} # -- a tc-t "-s" kapcsolóval indítja

tcdebug=${tcdebug:-0} # -- tipikus indítószkript-használat
tcdebug=${tcdebug:-1} # -- egy csipetnyi információ az eseményekről
tcdebug=${tcdebug:-2} # -- hibakövetési információk
#
#
# még egy lehetőség: felülbírálhatók az alapbeállításként használt tc és tcc eszközök
# az elérési út megadásával, például:
#
# tc=/usr/local/bin/tc
# tcc=/usr/local/tcng/bin/tcc
#
#
```

4.1. tcng használata csak a letöltések alakításához

Ezzel a példával sok általános koncepciót próbálok meg bemutatni. A példát a tc kimenetre a **tcc class-selection-path.tcc** parancs használatával lehet fordítani.

Példa 2. /etc/sysconfig/tcng/class-selection-path.tcc

```
/*
 * Egyszerű, magyarázattal ellátott tcng forgalomirányítási beállítófájl.
 *
 * Martin A. Brown <mabrown@securepipe.com>
 *
 * Példa: A class selection path használata.
 *
 * (Amennyiben HTML formában olvasod a szerkesztett kimenetet, a hívások
 * linkként jelennek meg a szövegben.)
```

```

*
*/

#include "fields.tc"      ❶
#include "ports.tc"

#define INTERFACE eth0  ❷

dev INTERFACE {
    egress { ❸

        /* A class selection path-ban a szűrők jönnek először! Dsmark */ ❹

        class ( <$ssh> )    if tcp_sport == 22 && ip_tos_delay == 1 ;
        class ( <$audio> )  if tcp_sport == 554 || tcp_dport == 7070 ;
        class ( <$bulk> ) \
            if tcp_sport == PORT_SSH || tcp_dport == PORT_HTTP ; ❺
        class ( <$other> )  if 1 ; ❻

        /* Ebben a részben állítjuk be a qdiskeket és az osztályokat */

        htb () { ❼
            class ( rate 600kbps, ceil 600kbps ) { ❽
                $ssh = class ( rate 64kbps, ceil 128kbps ) { sfq; } ;
                ❾ $audio = class ( rate 128kbps, ceil 128kbps ) { sfq; } ;
                $bulk = class ( rate 256kbps, ceil 512kbps ) { sfq; } ;
                $other = class ( rate 128kbps, ceil 384kbps ) { sfq; } ; ❿(10)
            }
        }
    }
}

```

- ❶ A **tcng** nyelvezete lehetővé teszi a C-hez hasonló beszerkeszthető (include) fájlok használatát, amivel bármilyen fájlt a forráskódba illeszthetünk. A beszerkeszthető fájlok elérési útja relatív az aktuális könyvtárhoz, vagy a **tcng** könyvtárhoz képest (alap esetben /usr/lib/tcng/include). Szigorúan véve nem szükséges a ports.tc és a fields.tc fájlok beszerkesztése (#include), mert a **tcc** fordító alpból megteszi ezt.

Az #include használata rugalmas változó definiálást, és a forgalomirányítás általános elemeinek gyors beszerkeszthetőségét teszi lehetővé.

További részletek találhatók a tcng kézikönyv Include files

(<http://linux-ip.net/gl/tcng/node35.html>)(Beszerkesztett fájlok) című fejezetében.

- ❷ Ezek CPP direktívák. A #define makrók és konstansok létrehozására szolgál. A használatáról többet a **tcng** kézikönyv Variables (<http://linux-ip.net/gl/tcng/node111.html>) (A Változók) fejezetében olvashatsz.
- ❸ Az egress kulcsszó a dsmark szinonimája. Az itt szereplő példa a class selection path (<http://linux-ip.net/gl/tcng/node32.html>) módszert használja. Ez az egress kulcsszó használata ebben a konfigurációban, szükséges hozzá a rendszer mag dsmark támogatása és a **tc**.
- ❹ A class selection path egyfajta megközelítési módja a forgalom alakításnak. A class selection path-ban a csomag a routerbe kerülésekor jelölést kap (DiffServ jel). Ennek a kezdeti osztályozásnak az eredményeként a router bármennyi műveletet végrehajthat, bármennyi szabályt (engedélyezés, besorolás, alakítás) alkalmazhat a csomagon.

További részletek találhatóak a **tcng** kézikönyvben az on class selection path (<http://linux-ip.net/gl/tcng/node32.html>) (A class selection path-ról) fejezetben.

- ⑤ Ez a példa a kapuk (portok) jelölésénél a nevek használatát mutatja be a számok használata helyett. Ez a **tcng** egyik kényelmi funkciója (a `ports.tc` automatikus beszerkesztésén keresztül válik elérhetővé). A kapuk elnevezésénél az IANA által használt kapu-nevek az irányadók. Lásd az IANA regisztrált portok (<http://www.iana.org/assignments/port-numbers>) oldalt a szabványos nevekért, vagy vizsgálj meg a `ports.tc` fájlt.

Mind a nevek, mind a számok használata elfogadott és érvényes.

- ⑥ Figyeljük meg ezt a különös konstrukciót, amely az összes eddig nem osztályozott csomagot osztályba sorolja. Minden, eddig nem osztályozott csomag a "\$other" osztályba kerül. Az `if 1` szerkezet a nem osztályozott forgalom maradványának osztályozására szolgál.
- ⑦ Itt történik a root qdisc létrehozása, ami jelen esetben az `eth0` eszközre van csatlakoztatva. A **tcng** kézikönyv referenciaanyagai között az appendix on queuing discipline parameters (<http://linux-ip.net/gl/tcng/node159.html>) (Függelék: sorba rendezés paraméterezése) fejezetben minden qdisc-hez találhatunk paramétereket. Bármilyen qdisc paraméter beilleszthető a zárójelek közé, ugyanúgy, ahogy egy későbbi példánál látni fogjuk. Paraméter nélküli használat esetén a zárójeleket nem kötelező kitenni.
- ⑧ Ebben a példában a legfelső-szintű osztályhoz a maximális sávszélességet rendeljük, ami ezen osztályon keresztül mehet. Tételezzük fel, hogy az `eth0` a belső hálózati csatlóója a gépnek. Ez a teljes sávszélességet 600 kilobit/s-ra korlátozza a belső hálózat felé.

Akik használtak már HTB-t, azoknak a `rate` és a `ceil` paraméterek ismerősek lesznek. Ezek HTB specifikus paraméterek, és a **tcc** eszköz segítségével fordíthatók le pontosan. Nézzük meg a **tcng** ütem és sebesség specifikáció részben található táblázatot!.

- ⑨ Ezen osztály hozzárendelése a változóhoz. Ezt általában a class selection path részeként végezzük el.
- (10) Ahogyan azt Martin Devera is leírta a HTB honlapján, egy beágyazott SFQ minden osztálynak ad egy sorba rendező algoritmust a konkurens folyamatoknak az adott osztályon keresztüli csomagküldés erőforrásainak megfelelő elosztásához. Figyeljünk a paraméterek hiányára a beágyazott sorba rendezés elvénél.

Ha nem határozzuk meg sorba rendezési szabályokat a végső osztályoknál, akkor az alapbeállítás érvényesül, vagyis a `pfifo_fast` qdisc. Egy sztochasztikus tiszta sorba rendező qdisc beszerkesztése a vég-osztályokban meggátolja azt, hogy egyetlen kapcsolat uralja a sávszélességet az adott osztályban.

4.2. A kétsebességű háromszínű mérőóra használata

Példa 3. `/etc/sysconfig/tcng/two-rate-three-color-meter.tcc`

```
/*
 * Egyszerű, magyarázattal ellátott tcng forgalomirányítási beállítófájl.
 *
 * Martin A. Brown <mabrown@securepipe.com>
 *
 * Példa: Mérőóra használata
```



```

*
* (Amennyiben HTML formában olvasod a szerkesztett kimenetet, a hívások
* hivatkozásként jelennek meg a szövegben.)
*
*/

#define EXCEPTION      192.168.137.50
#define INTERFACE      eth0

$meter = trTCM( cir 128kbps, cbs 10kB, pir 256kbps, pbs 10kB ); ❶

dev eth0 {
    egress {
        class ( <$full> )      if ip_src == EXCEPTION          ; ❷
        class ( <$fast> )      if trTCM_green( $meter )         ; ❸
        class ( <$slow> )      if trTCM_yellow( $meter )        ; ❹
        drop                  if trTCM_red( $meter )            ; ❺
        htb {
            class ( rate 600kbps, ceil 600kbps ) {
                $fast = class ( rate 256kbps, ceil 256kbps ) { sfq; } ;
                $slow = class ( rate 128kbps, ceil 128kbps ) { sfq; } ;
                $full = class ( rate 600kbps, ceil 600kbps ) { sfq; } ;
            }
        }
    }
}

```

- ❶ Ez az osztályozáshoz használt mérőóra deklarációja. A háttértechnológia a mérés kivitelezésére az engedélyezés (policing). További információk a tcng kézikönyv (<http://linux-ip.net/gl/tcng/node53.html>) "Meters" (Mérések) fejezetben találhatók.

Ez a kétsebességes-háromszínű mérő a legösszetettebb mérőórátípus a **tcng** nyelvben. Ez a mérőóra a zöld, sárga és piros színeket ad vissza, a tárolók foglaltságától és csúcserőtekeitől (peak) függően. Ha a mért sebesség nagyobb, mint a foglaltságot jelző érték, akkor sárga színű lesz a jelzés, ha a mért sebesség túllépi a csúcserőteket (a csúcserőteket itt nem a maximális, hanem kiemelkedő értéket jelent - a lektor), akkor piros színű lesz a jelzés.

A \$meter változó a mérőóra típusának megfelelő függvényekkel működtethető. Ebben az esetben három függvény jöhet szóba a \$meter állapotának tesztelése céljából, ezek: trTCM_green, trTCM_yellow, és trTCM_red. A hatékonyság növelése érdekében lapozz bele a Accelerating three color meters (<http://linux-ip.net/gl/tcng/node58.html>) (gyorsított háromszínű mérők) fejezetbe.

- ❷ A példában a 192.168.137.50 IP címet kizártuk az eth0 eszközről induló engedélyezési listából.
- ❸ A "foglalt" információ-sebesség eléréséig (cir) a csomagok áthaladnak ezen az osztályon A tokenek eltávolításra kerülnek a cir/cbs tárolóból.

A mérőóra zöld.

- ❹ Itt lesznek osztályozva azok a tárolók, ahol a forgalom sebessége átlépi a cir/cbs értéket. A pir/pbs bucket (pir a "peak information rate"="csúcs információ-sebesség"), pbs a "peak burst size"="csúcs kitörési sebesség"). Ez teszi lehetővé egy bizonyos adatfolyam számára, hogy garantált sávszélességet kapjon egy osztályon belül, majd újra osztályozásra kerüljön.

A mérőóra sárga.

- ⑤ Azok a tárolók lesznek osztályozva itt, ahol a forgalom sebessége meghaladja a `pir/pbs` értéket. Az általános beállítások azt eredményezik, hogy a forgalom a csúcssebesség feletti tartományba kerül, habár a forgalmat újra osztályozva a garantált osztályból a legjobb-teljesítmény osztályba kerülne.

A mérőóra piros.

5. Egyéb megjegyzések

Hál Istennek a **tcng** elkerüli a **tc** kisebb kényelmetlenségeit. A következő táblázat ábrázolja ezen eszközök szintaxisát és mértékegységeit, angol megfelelőikkel együtt.

Táblázat 1. Sebesség/Ütem szintaxis: **tcng** vs. **tc**

tcng	Angol megfelelő	tc
bps	bit(s) per second	bit
Bps	byte(s) per second	bps (argh!)
kbps	kilobit(s) per second	kbit
kBps	kilobyte(s) per second	kbps
Mbps	megabit(s) per second	mbit or Mbit
MBps	megabyte(s) per second	mbps or Mbps
pps	packet per second	??

Ez csak minimális igazítást igényel az ősrégi **tc** felhasználóktól, de sokkal jobb választás az angolul beszélőknek ha a megérzéseikre szeretnének hagyatkozni.

Például a **tcng** konfigurációjában használhatjuk a sebesség jelölésére a konvencionális kifejezéseket: **100Mbps**, **128kbps**, sőt **2Gpps**. Bővebben a **tcng** kézikönyv Units (<http://linux-ip.net/gl/tcng/node21.html>) (a mértékegységekről) fejezetében olvashatunk.

A forgalomirányítás hatékonysága érdekében meg kell keresni a hálózat szűk keresztmetszeti pontjait. Legtöbb esetben ezeken a helyeken kell elvégezni a műveleteket.

6. További leírások és hivatkozások

- the linux DiffServ project (<http://diffserv.sourceforge.net/>)
- HTB site (Martin “devik” Devera) (<http://luxik.cdi.cz/~devik/qos/htb/>) (HTB honlap)
- Traffic Control Next Generation (**tcng**) (<http://tcng.sourceforge.net/>) (következő generációs forgalomirányítás)
- TCNG manual (Werner Almesberger) (<http://linux-ip.net/gl/tcng/>) (TCNG kézikönyv)

- iproute2 (*Alexey Kuznetsov*) (<ftp://ftp.inr.ac.ru/ip-routing/>)
iproute2 manual (*Alexey Kuznetsov*) (<http://linux-ip.net/gl/ip-cref/>) (iproute2 kézikönyv)
- Research and documentation on traffic control under linux (*Stef Coene*) (<http://www.docum.org/>) (Forrásanyagok és dokumentációk a forgalomirányításról Linux alatt)
- LARTC HOWTO (*bert hubert, et. al.*) (<http://lartc.org/howto/>)
- guide to IP networking with linux (*Martin A. Brown*) (<http://linux-ip.net/>) (Az IP hálózati protokoll használata Linuxszal)