

## Grafikonok minden mennyiségben – RRDTool (2. rész)

Az RRDTool egyik nagy előnye, hogy a puszta adatokkal képes különféle egyszerűbb műveleteket végezni, illetve ezeket a számolt értékeket a grafikonon kiírni.

© Kiskapu Kft. Minden jog fenntartva

■ Az *RRDTool* által használt matematikai rendszer kicsit érthetetlen a köznapi matematikán nevelkedett szemnek, hiszen inverz lengyel logikán alapul, amelyet *RPN (Reverse Polish Notation)* néven illetnek angol nyelvterületen. Hamar meg lehet szokni, de sok bosszúságot tud okozni, ha nem figyelünk oda. A legjobb tanító a példa, így egy konkrét megvalósításon át mutatom be egy összetett grafikon készítését.

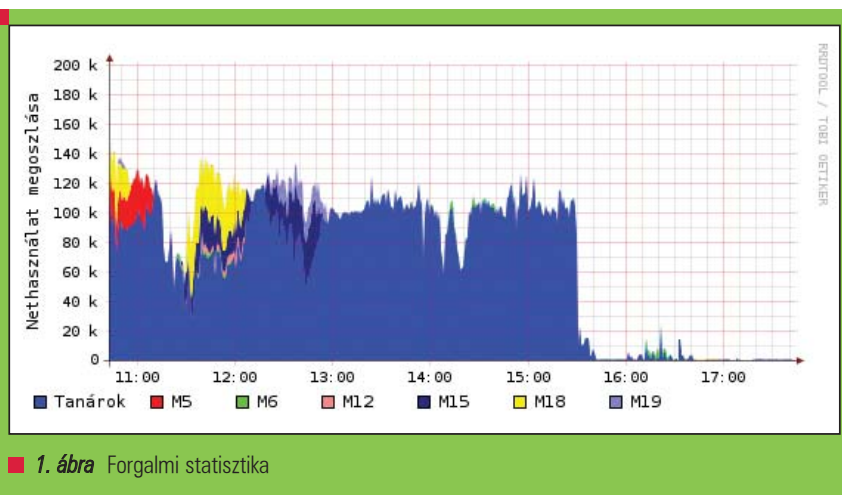
### Hálózati forgalom megoszlása

Érdekes statisztikát tudunk készíteni, ha egy iskolában a számítógéptermekek által okozott forgalmat arányaiban összevetjük. Ehhez egy megfelelő tűzfalszabályt kell létrehoznunk, amely *IP* tartomány szerint számlálja az átmenő forgalmat, majd a mért adatokat rendre beletesszük egy *RRD* fájlba. Első nekifutásra elég, ha egymásra pakoljuk a forgalmi adatokat:

```

/usr/local/bin/rrdtool graph subnet.png \
--imgformat PNG -u 150000 -l 0 --width 464 \
--height 200 \
--end now-3h --start end-7h --vertical-label \
"Nethasználat megoszlása" \
--slope-mode --lazy --color "SHADEA#ffffff" \
--color "SHADEB#ffffff" --color "BACK#ffffff" \
"DEF:Tanar=/var/db/rrd/subnet/subnet.rrd:Tanar: \
AVERAGE" \
"DEF:M5=/var/db/rrd/subnet/subnet.rrd:M5: \
AVERAGE" \
"DEF:M6=/var/db/rrd/subnet/subnet.rrd:M6: \
AVERAGE" \
"DEF:M12=/var/db/rrd/subnet/subnet.rrd:M12: \
AVERAGE" \
"DEF:M15=/var/db/rrd/subnet/subnet.rrd:M15: \
AVERAGE" \
"DEF:M18=/var/db/rrd/subnet/subnet.rrd:M18: \
AVERAGE" \
"DEF:M19=/var/db/rrd/subnet/subnet.rrd:M19: \
AVERAGE" \

```



1. ábra Forgalmi statisztika

```

"AREA:Tanar#0000FF:Tanárok":STACK \
"AREA:M5#FF0000:M5":STACK \
"AREA:M6#00FF00:M6":STACK \
"AREA:M12#FF8888:M12":STACK \
"AREA:M15#000088:M15":STACK \
"AREA:M18#FFFF00:M18":STACK \
"AREA:M19#8888FF:M19":STACK

```

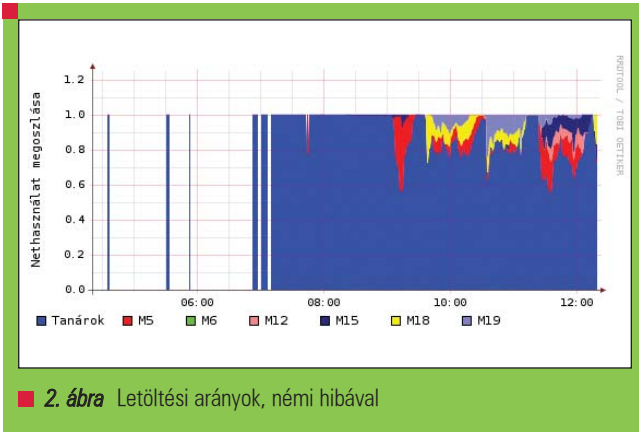
Az 1. ábrán látható grafikon elkészítéséhez csak olyan paramétereket és tulajdonságokat használtunk fel, amelyeket már ismertettem az előző részben, s a grafikon sem azt ábrázolja, amire szükségünk van, ugyanis nem látszik rajta tisztán a hálózat terhelésének százalékos megoszlása. A feladat elvégzéséhez egyszerűen össze kell adnunk az aktuális forgalmakat, majd a mért adatokat egyenként el kell osztanunk a teljes terheléssel, majd ezeket ábrázolni. Az *RRDTool* ehhez egy *CDEF* direktívát ad a kezünkbe, amellyel létre tudunk hozni új változókat:

```

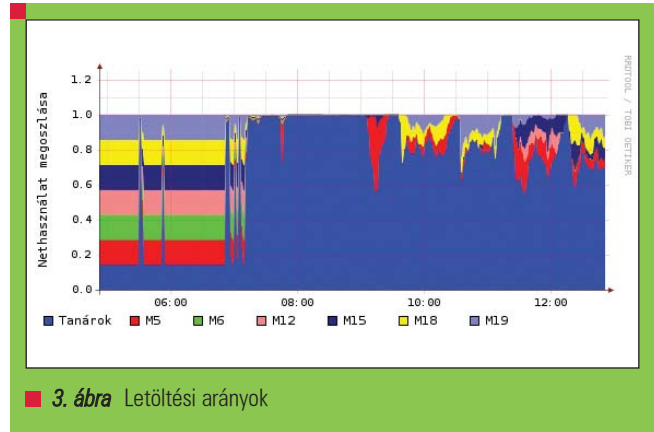
"CDEF:Osszeg=M19,M18,M15,M12,M6,M5,Tanar,+,+,+,+,+ \
+,+"

```

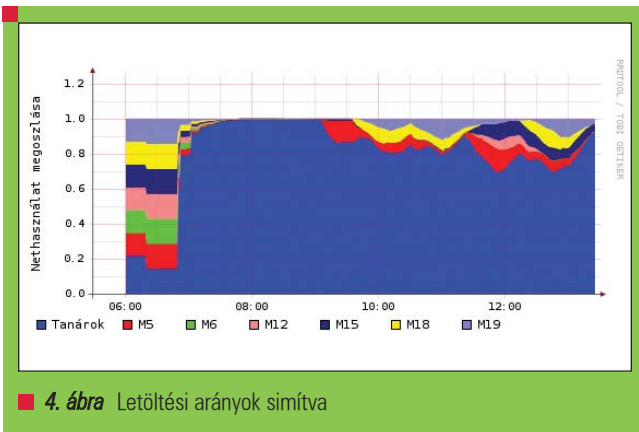
Mint említettem, az összeadás módja messze áll az általános iskolában tanult módszertől. Az inverz lengyel logika olyan ága a matematikának, amelyet a zárójelzés és egyéb problémák kiküszöbölése okán talált ki *Jan Lukasiewicz* lengyel származású matematikus.



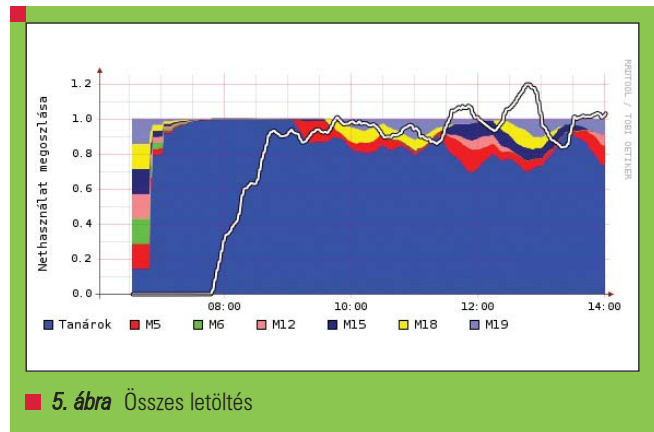
2. ábra Letöltési arányok, némi hibával



3. ábra Letöltési arányok



4. ábra Letöltési arányok simítva



5. ábra Összes letöltés

A módszer abból áll, hogy a nem a változók közé tesszük a műveleti jeleket, hanem eléjük vagy utánuk. A műveletvégzés pedig úgy zajlik, hogy a feldolgozó szál levesz egy műveleti jelet, majd annyi változót, amennyit az adott művelet igényel. A fenti példában a legelső összeadás jel fog először végrehajthatni, és az *RRDTool* műveletvégző eljárása leemeli a tőle balra lévő adatokat – az *M5* és a *Tanar* változókat – a listáról, majd elvégzi rajtuk a műveletet és visszateszi az eredményt a helyükre. Az eljárás addig tart, amíg el nem fogynak a műveleti jelek, ekkor az utolsó változó értékét a megadott nevű változó felveszi. Tulajdonképpen nem értem, hogy miért nem építettek bele az *RRDTool* programba valami „emberibb” matematikai modult, valószínűleg a készítőknél egy *HP* kézi kalkulátor volt az etalon...

```
"CDEF:Osszeg=M19,M18,M15,M12,M6,M5,Tanar,+,+,+,+,+,+,+" \
"+,+,+" \
"CDEF:TanarSzazalek=Tanar,Osszeg,/" \
"CDEF:M5Szazalek=M5,Osszeg,/" \
"CDEF:M6Szazalek=M6,Osszeg,/" \
"CDEF:M12Szazalek=M12,Osszeg,/" \
"CDEF:M15Szazalek=M15,Osszeg,/" \
"CDEF:M18Szazalek=M18,Osszeg,/" \
"CDEF:M19Szazalek=M19,Osszeg,/" \
```

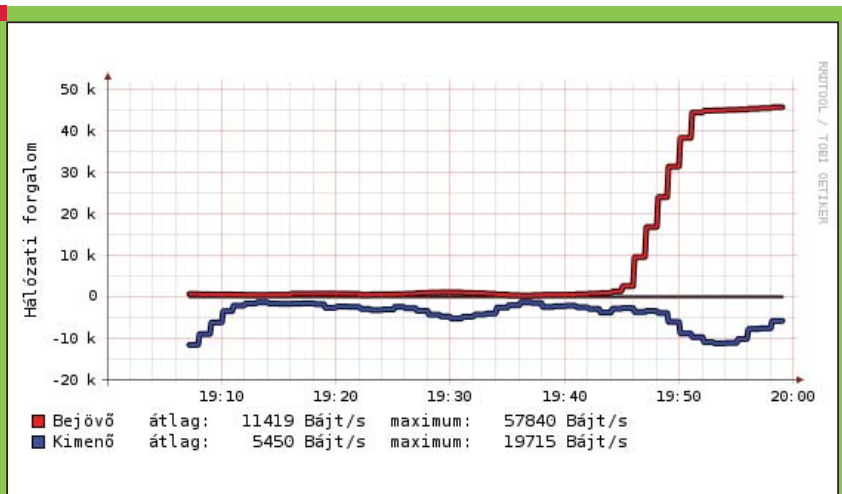
Az összeg kiszámítása után már csak arra kell figyelniük, hogy ne osszunk nullával, hiszen az nem ábrázolható a grafikonon, ilyenkor kihagyásokat fogunk kapni a grafikonon. Minden egyes tartomány terheléséhez

hozzáadunk egy kicsi számot, az összeghez pedig még annyit kell hozzáadnunk, amennyi tartományt ábrázolni szeretnénk:

```
"CDEF:Osszeg=0.7,M19,M18,M15,M12,M6,M5,Tanar,+,+,+,+,+,+,+" \
"+,+,+,+,+,+" \
"CDEF:TanarSzazalek=Tanar,0.1,+,Osszeg,/" \
"CDEF:M5Szazalek=M5,0.1,+,Osszeg,/" \
"CDEF:M6Szazalek=M6,0.1,+,Osszeg,/" \
"CDEF:M12Szazalek=M12,0.1,+,Osszeg,/" \
"CDEF:M15Szazalek=M15,0.1,+,Osszeg,/" \
"CDEF:M18Szazalek=M18,0.1,+,Osszeg,/" \
"CDEF:M19Szazalek=M19,0.1,+,Osszeg,/" \
```

Itt már fokozódik az *RPN* műveletek átláthatatlansága, ugyanis a műveletek sorrendje és a változók sorrendje különbözik! Az összeg kiszámításánál ez nem okoz gondot, hiszen összeadásnál a sorrend felcserélhető. Ha osztani vagy szorozni is akarunk, akkor úgy kell a sorrendet kialakítani, ahogy azt a képlet megkívánja. Gyakorlatilag egy  $(Tanar+0.1)/Osszeg$  műveletet kell végrehajtanunk, amelyet az *RPN* alapján a példa szerint tudunk megoldani. Az *RRDTool* megtalálja az első műveleti jelet, és a tőle balra lévő adatokon végrehajtja, majd a helyükre teszi az eredményt, így a *Tanar* és a  $0.1$  összege fog itt szerepelni. Ezek után ezt elosztjuk az *Osszeg* változó értékével, és elő is áll a kívánt grafikon (3. ábra).

Egy kicsit tudjuk javítani a grafikon kinézetét, ha a durva csúcsokat kicsit lesimítjuk a csúszóablak használatával, így egy kissé hamis, de sokkal informatívabb ábrát tudunk kapni:



6. ábra Írás a jelmagyarázaton

```
"VDEF:outAvg=out,AVERAGE" \
"VDEF:outMax=out,MAXIMUM" \
```

Ezeket akár ki is jelezhetjük a kész grafikonon, viszont sokkal jobb, ha kiírjuk a színek magyarázatához a konkrét értékeket:

```
"LINE2:inT#FF0000:Bejövő " \
"GPRINT:inAvg:átlag\:%7.01f
↳Bájt/s" \
"GPRINT:inMax:maximum\:%7.01f
↳Bájt/s\n" \
"LINE2:outNegT#0000FF:Kimenő " \
"GPRINT:outAvg:átlag\:%7.01f
↳Bájt/s" \
"GPRINT:outMax:maximum\:%7.01f
↳Bájt/s\n" \
```

```
"CDEF:TanarT=Tanar,1800,TREND" \
"CDEF:M5T=M5,1800,TREND" \
"CDEF:M6T=M6,1800,TREND" \
"CDEF:M12T=M12,1800,TREND" \
"CDEF:M15T=M15,1800,TREND" \
"CDEF:M18T=M18,1800,TREND" \
"CDEF:M19T=M19,1800,TREND" \
```

A csúszóablak lényege, hogy a grafikonon a megadott időtartam átlaga fog megjelenni, így elsimítjuk a kiugró csúcsokat, az arányok változását szépen követhető vonalakká alakíthatjuk. Az egyetlen probléma, hogy más-más ablakméretet kell használnunk az utolsó egy óra és az utolsó egy év ábrázolásához, de ezt hamar meg tudjuk oldani. Ha meg tudjuk határozni a vonalunk maximális áteresztő képességét, akkor érdemes erre a grafikonra egy vonallal rárajzolni az aktuális százalékos terhelést is. Ebből egy szám rendelkezésünkre áll, mégpedig az összes áteresztett forgalom, ezt el kell osztani azzal a számmal, amennyi a bejövő vonalunk teljes kapacitása. Ezt esetleg korrigálnunk kell a *HTTP proxy* által kiszolgált tartalommal, amely a 100% fölé tudja emelni a letöltött adatmennyiséget.

```
"CDEF:TeljesSzazalek=Osszeg,131072,/" \
"LINE4:TeljesSzazalek#000000:" \
"LINE2:TeljesSzazalek#FFFFFF:" \
```

### Statisztika, függvények

A grafikonok alapján néhány mért adatot ki tudunk írni, így pontos képet kaphatunk a mért és tárolt adatokról. A CDEF direktíva mellett van egy VDEF is, amely nem az aktuális időpont értékét kezeli, hanem a megjelenített tartományon képes műveleteket végezni. Általában három műveletet szoktunk használni a rendelkezésre álló tíz közül: az átlag, a minimum és a maximum meghatározása. A VDEF használata nagyon egyszerű, nézzük meg egy hálózati csatlakozón átfolyó adatmennyiséget, és határozzuk meg az átlagot és a maximumot:

```
"VDEF:inAvg=in,AVERAGE" \
"VDEF:inMax=in,MAXIMUM" \
```

A kiírást úgy kell elvégeznünk, hogy az *RRDTool* a grafikon alján egy grafikus kurzort mozgat a kiírt karakterek szerint. Így a jelmagyarázat után közvetlenül ki tudjuk írni, hogy mennyi az átlagos és mennyi a maximális forgalom, a C nyelvből örökölt formátumszövegeket használva. Munkánk eredménye egy grafikon, amelynek a jelmagyarázata tartalmazza a kívánt értékeket.

### Összegzés

Az *RRDTool* ideális eszköz, hogy különféle tendenciákat és jellegzetességeket figyelhessünk meg a rendszeresen keletkező adatainkon, legyenek ezek számítógépekhez közel álló adatok (terhelés, forgalom, hőmérséklet, stb), vagy egyéb mérések eredményei (időjárás, játék, tőzsdei információk, stb).



**Auth Gábor**  
(auth.gabor@enaplo.hu)

Egy pécsi középiskolában informatikát és programozást oktat. Tíz éve botlott először a UNIX rendszerekbe, 7 év Linux használat után kapta el a FreeBSD látat, amiből máig nem tudott kigyógyulni.

### KAPCSOLÓDÓ CÍMEK

- Az RRDTool honlapja  
➔ <http://www.rrdtool.org>
- A cikkben említett példák forrása  
➔ <http://user.enaplo.hu/~auth.gabor/rrd/>
- FreeCIV játék RRDTool követése  
➔ <http://user.enaplo.hu/~auth.gabor/freeciv/>
- Egy való életből vett példa  
➔ <http://www.enaplo.hu/index.jsp?page=visitor.loadStat>