

Configurarea plăcii de rețea. Configurarea unei rețele locale

Pentru a putea conecta un calculator în rețea trebuie configurată placa de rețea numită și interfață de rețea (Network Interface Card). Chiar dacă nu suntem conectați la rețea unele aplicații necesită pentru a funcționa configurarea unei rețele interne. Această adresă este numită de loopback și are adresa IP 127.0.0.1. Se poate verifica funcționalitatea interfeței de loopback (și a celorlalte interfețe) cu ajutorul comenzii

```
ifconfig
```

fără parametric. Aceasta va fișa o listă a interfețelor de rețea și a setărilor acestora.. Interfața de loopback va apărea cu eticheta **lo** iar prima placă de rețea Ethernet cu eticheta **eth0**.

Dacă nu apare interfața **lo** în lista de interfețe afișate, atunci se poate adăuga cu comenzile

```
ifconfig lo 127.0.0.1  
route add -net 127.0.0.1
```

1) Configurarea plăcii de rețea (eth0). Setări temporare.

Dacă folosim în loc de eticheta **lo** eticheta **eth0** putem configura interfața de rețea **eth0**. Dacă avem mai multe interfețe de rețea acestea se vor numi succesiv **eth1**, **eth2** și așa mai departe.

Pentru exemplele de mai jos vom folosi adrese nerutabile, care nu se pot folosi direct pentru conectarea la Internet. Astfel avem, următoarele adrese nerutabile la dispoziție:

Clasa rețelei	Masca de rețea (netmask)	Adresa de rețea
A	255.0.0.0	10.0.0.0 – 10.255.255.255
B	255.255.0.0	172.16.0.0 – 172.16.255.255 și 172.31.0.0 – 172.31.255.255
C	255.255.255.0	192.168.0.0 – 192.168.0.255

Dacă ne conectăm la o rețea existentă trebuie să îi cunoaștem adresa sa IP, masca de rețea și adresa de broadcast. De asemenea avem nevoie de adresa routerului și a serverului dns.

În continuare vom prezenta modalitatea de atribuire a unei adrese statice pentru placa de rețea eth1. Conform unor surse documentația consultată această modalitate va avea ca efect o setare **temporară** a interfeței de rețea, care **nu supraviețuiește unei resetări** a calculatorului.

Conform cu alte surse, nu mai este nevoie de nimic în plus pentru ca setările să fie permanente, deoarece sunt create niște scripturi în directorul `/etc/network/network-scripts`, care sunt folosite pentru activarea sau dezactivarea plăcilor de rețea corespunzătoare. Un exemplu de asemenea script numit `/etc/network/network-scripts/ifcfg-eth1` pentru activarea plăcii de rețea **eth1** va fi prezenta mai jos.

Un exemplu de atribuire plăcii de rețea eth1 adresa de clasă C 192.168.1.1, unde masca de rețea este 255.255.255.0 și adresa de broadcast este 192.168.1.255 este

```
ifconfig eth1 192.168.1.1 netmask 255.255.255.0 broadcast 192.168.1.255
```

Rezultatul comenzii de mai sus este crearea automată a unui fișier în directorul /etc/network/network-scripts

Numit **ifcfg-eth1**, care are următorul conținut:

```
DEVICE=eth1  
BOOTPROTO=static  
BROADCAST=192.168.1.255  
IPADDR=192.168.1.1  
NETMASK=255.255.255.0  
NETWORK=192.168.1.0  
ONBOOT=yes
```

Ultima linie **ONBOOT=yes** spune kernelului că interfața **eth1** este activă (sau nu) la startup.

Opțiunea **BOOTPROTO=dhcp** va fi folosită în cazul în care adresa IP este obținută în mod dinamic, de la un server DHCP. În acest caz fișierul ifcfg-eth1 va arăta ca mai jos:

```
DEVICE=eth0  
BOOTPROTO=dhcp  
ONBOOT=yes
```

Pentru configurarea în mod grafic a rețelei se poate folosi comanda **redhat-config-network**

Dacă avem de configurat și alte calculatoare pentru a forma o rețea locală, fără acces la Internet, vom alege adrese successive pentru adresele lor IP, celelalte informații rămân la fel ca la prima. De exemplu folosind comanda

```
ifconfig eth1 192.168.1.2 netmask 255.255.255.0 broadcast 192.168.1.255
```

lansată pe cel de-al doilea calculator din rețea am configurat placa sa de rețea eth1 cu adresa IP 192.168.1.2.

Pentru ca cele două calculatoare să se “vadă” corect între ele mai trebuie să modificăm și fișierele

```
/etc/host.conf  
/etc/hosts  
/etc/resolv.conf
```

2) În primul în /etc/host.conf trebuie să scriem
order hosts, nis, bind

Aceasta indică ordinea în care se face căutarea în momentul rezolvării numelui unui calculator, mai precis: se caută mai întâi în fișierul hosts pentru a rezolva un nume de calculator, apoi harțile NIS, apoi este chestionat serverul de DNS.

3) Fișierul /etc/hosts conține o listă de adrese IP și denumirile de calculatoare asociate. În mod sigur va conține adresa de loopback și numele calculatorului local.

Dacă adresele calculatoarelor din rețea sunt asignate dinamic, de către serverul DHCP, atunci fișierul `/etc/hosts` nu conține altceva.

În caz că există și adrese IP fixe, neatribuite dinamic de către serverul DHCP, atunci în fișierul `/etc/hosts` vom avea linii de forma

IP_address canonical_hostname aliases

unde IP_address este adresa IP a calculatorului descris
canonical_hostname este numele complet
aliases sunt alte nume pentru acest calculator

De exemplu:

```
127.0.0.1    localhost    localhost.localdomain
192.168.1.1  linuxtest1  linuxtest1.info.uvt.ro
194.102.62.46 blue         blue.info.uvt.ro
```

4) Fișierul `/etc/resolv.conf` conține o listă a serverelor DNS din domeniu, de exemplu, pentru calculatorul `linux-test41` din rețeaua noastră, conținutul acestui fișier este `nameserver 194.102.62.1`

Dacă sunt mai multe servere DNS vor exista mai multe linii ca cea de mai sus, câte una pentru fiecare server DNS (`nameserver`).

5) Asignarea a mai multor adrese la o singură placă de rețea.

```
ifconfig eth1:0 192.168.1.4
route add -host 192.168.1.4 dev eth1
```

Prima comandă atribuie adresa `192.168.1.4` interfeței virtuale (logice) `eth1:0`, iar a doua adaugă o rută pentru această adresă la interfața reală `eth1`.

6) Fișierul `/etc/sysconfig/network` conține informații despre configurarea rețelei. Aici se pot preciza numele domeniului, gateway-ul, și hostname-ul. Este importantă prima opțiune (`Networking = yes`) care face ca să fie pornit automat serviciul `network`, lucru care va face să fie folosite scripturile de activare a plăcilor de rețea, scripturi cu numele `ifcfg-eth0`, `ifcfg-eth1`, etc., aflate în directorul `/etc/sysconfig/network-scripts`

```
NETWORKING=yes
HOSTNAME=turtle.mytrek.com
GATEWAY=192.168.0.1
```

7) După setarea de noi configurări pentru placa de rețea pentru ca acestea să devină efective se repornește serviciul `network`

```
/etc/init.d/network restart
```

La laborator vă rog să nu încercați acest lucru, deoarece administrând serverele de la distanță, vom fi cu toții deconectați.

Router și gateway

Dacă avem o rețea locală configurată folosind adrese IP nerutabile (vezi tabelul de mai sus), pentru a avea o conexiune la Internet, avem nevoie de un **router**. Acesta poate fi un dispozitiv special, în general de formă paralelipipedică, sau un calculator cu mai multe interfețe de rețea. Una din cele două interfețe este conectată fizic la prima rețea iar cea de-a doua interfață la cea de-a doua rețea.

Dacă avem două segmente de rețea sau subrețele conectate între ele printr-un calculator (router) acesta se mai numește și **gateway**. Pentru a putea face rutare între cele două segmente de rețea (subrețele) acest calculator trebuie să aibă **IP forwarding** activate. Pentru a determina dacă este activat se verifică dacă primim rezultatul 1 (unu) la comanda

```
cat /proc/sys/net.ipv4/ip_forward
```

Dacă rezultatul returnat este 0 (zero), atunci activarea IP forwarding se face cu

```
echo "1" > /proc/sys/net.ipv4/ip_forward
```

Să presupunem că avem următoarea situație:

- prima subrețea are adresa de rețea 192.168.1.0 și calculatoare cu adresele din intervalul 192.168.1.1 – 192.168.1.126, masca de rețea (subnet mask) 255.255.255.128
- a doua subrețea are adresa de rețea 192.168.1.128 și calculatoare cu adresele din intervalul 192.168.1.129 – 192.168.1.254, masca de rețea (subnet mask) 255.255.255.128
- pe router avem cele două plăci de rețea configurate astfel

interfață de rețea	adresă IP	netmask
eth0	192.168.1.1	255.255.255.128
eth1	192.168.1.129	255.255.255.128

- calculatoarele din prima subrețea vor avea ca și gateway adresa 192.168.1.1. Acest lucru se realizează cu comanda
route add -net 192.168.1.0 and then
route add default gw 192.168.1.129
pe fiecare stație din rețea
- calculatoarele din a doua subrețea vor avea ca și gateway adresa 192.168.1.129 Acest lucru se realizează cu comanda
route add -net 192.168.1.128 and then
route add default gw 192.168.1.1
pe fiecare stație din rețea

Configurarea unui server DHCP. și a clienților DHCP

Este util un server DHCP dacă dorim ca să avem adresele IP atribuite în mod dinamic calculatoarelor (de către serverul DHCP), nemaifiind astfel nevoie să configurăm fiecare stație individual. Se instalează și se pornește programul dhcpd. Fișierul de configurare a acestuia este

```
/etc/dhcp.conf
```

O mostră a fișierului de mai sus se găsește în directorul

```
/usr/share/doc/dhcp-<version-number>/dhcpd.conf.sample
```

O explicație pe scurt a acestuia urmează mai jos. Important este faptul că trebuie să avem o secțiune **subnet** pentru fiecare placă de rețea a calculatorului pe care rulează serverul DHCP.

```
ddns-update-style interim # Redhat Version 8.0+
subnet 192.168.1.0 netmask 255.255.255.0 {

    # Intervalul de adrese IP pe care serverul DHCP
    # le va atribui clienților care solicită adrese atribuite dinamic
    range 192.168.1.201 192.168.1.220;

    # Intervalul de timp, in secunde pentru care un client poate păstra adresa
    # IP primită
    default-lease-time 86400;
    max-lease-time 86400;

    # Setarea gateway-ului implicit folosit de clienti
    option routers 192.168.1.1;

    # Nu forwarda cererile DHCP primite pe această interfață
    # de rețea către o altă interfață de rețea
    option ip-forwarding off;

    # Setarea adresei de broadcast si a subnet mask-ului
    # pe care il vor folosi clientii DHCP
    option broadcast-address 192.168.1.255;
    option subnet-mask 255.255.255.0;

    # Setarea serverului DNS pe care il vor folosi clientii DHCP
    option domain-name-servers 192.168.1.100;

    # Setarea serverului NTP pe care il vor folosi clientii DHCP
    option nntp-server 192.168.1.100;

    # Dacă folosim un server WINS pentru clientii Windows
    # va trebui inclusa următoarea optiune in fisierul dhcpd.conf option
    netbios-name-servers 192.168.1.100;

}
#
# List an unused interface here
#
subnet 192.168.2.0 netmask 255.255.255.0 {
}
```

```
# Se pot asigna si adrese IP pe baza adresei MAC a clientilor
# ca în exemplul de mai jos, numele calculatorului fiind "smallfry":
host smallfry {
hardware ethernet 08:00:2b:4c:59:23;
fixed-address 192.168.1.212;
}
```

În fișierul mostră

`/usr/share/doc/dhcp-<version-number>/dhcpd.conf.sample`
se găsesc, comentate, mai multe exemple de configurare a serverului de DHCP
Pornirea serverului DHCP se face cu folosind scriptul `dhcp` cu parametrul `start`, ca mai jos

`/etc/init.d/dhcp start`

Testarea dacă este pornit serverul se face cu același script, parametrul `status`

`/etc/init.d/dhcp status`

Dacă avem mai multe plăci de rețea, dar dorim ca serverul DHCP să pornească pentru una singură dintre ele, de exemplu pentru `eth0`, atunci în fișierul `/etc/sysconfig/dhcpd` Vom scrie

```
# Command line options here
DHCPDARGS=eth0
```

Configurarea unui client DHCP

Pentru clienți DHCP avem 3 variante posibile: `dhcpd`, `dhclient` și `pump`. Un client DHCP asigură următoarele funcții: contactarea serverului DHCP și primirea unei adrese temporare (leased address), reînnoirea acesteia, după ce expiră

Dacă avem un calculator cu următoarele două setări¹, pentru un sistem rulând o distribuție din familia RedHat > 8.0 (RedHat, Fedora, CentOS):

- fișierul `/etc/sysconfig/network` va conține o linie

`NETWORKING=yes`

- placa de rețea `eth1` configurată ca mai jos, având în fișierul `/etc/sysconfig/network-scripts/ifcfg-eth1`:

`DEVICE=eth0`

`BOOTPROTO=dhcp`

`ONBOOT=yes`

Atunci, la pornire este pornit automat serviciul `network` și placa de rețea `eth1` este activată și configurată să primească adresă IP în mod dinamic, de la serverul de DHCP.

Pentru versiuni mai vechi de RedHat, este posibil să fie folosiți clienții `pump` sau `dhcpd`. De exemplu, folosind `dhcpd`, configurarea stațiilor client `dhcp` se face căutând locația programului `dhclient` (cu comanda **`which dhcpd`**) și trecându-l în fișierul

¹ <http://www.redhat.com/docs/manuals/linux/RHL-8.0-Manual/custom-guide/s1-dhcp-configuring-client.html>

/etc/rc.local în care se găsesc scripturi care sunt rulate după procesul **init**. Un exemplu de cum ar putea arăta acest script²:

```
#!/bin/sh
#
# rc.inet1      This shell script boots up the base INET system.

HOSTNAME=`cat /etc/HOSTNAME` #This is probably not necessary but I
                             #will leave it in anyways

# Attach the loopback device.
/sbin/ifconfig lo 127.0.0.1
/sbin/route add -net 127.0.0.0 netmask 255.0.0.0 lo

# IF YOU HAVE AN ETHERNET CONNECTION, use these lines below to configure the
# eth0 interface. If you're only using loopback or SLIP, don't include the
# rest of the lines in this file.

/usr/sbin/dhccpd
```

Conform cu site-ul¹ nu este nevoie de configurațiile de mai sus pentru Redhat > 8.0, deci nici de modificarea scriptului rc.local ca mai sus. În mod normal, în acest script lipsește ultima linie.

Agentul DHCP Relay

Agentul DHCP Relay (`dhcrelay`) permite redirectarea cererilor DHCP și BOOTP din subrețele pe care nu există server de DHCP la unul sau mai multe servere de pe alte subrețele¹.

Când un client DHCP solicită o informație, agentul DHCP Relay retrimite cererea la lista de servere DHCP specificate când a fost pornit agentul DHCP Relay. Când un server DHCP returnează un răspuns, acesta este transmis în modul broadcast sau unicast în rețeaua din care provine semnalul original.

Dacă nu specificăm restricții în fișierul `/etc/sysconfig/dhcrelay` prin directiva `INTERFACES` atunci agentul DHCP Relay ascultă pe toate interfețele.

Pentru a porni agentul DHCP Relay, se folosește comanda

```
service dhcrelay start.
```

Baza de date pentru închirieri de adrese DHCP (Lease Database)

Toți clienții care primesc adrese dinamice de la un server DHCP trebuie reținuți, serverul formând o bază de date cu aceste informații. Aceasta se găsește în fișierul `/var/lib/dhcp/dhcpd.leases`

² <http://tldp.org/HOWTO/DHCP/x74.html>

S-ar putea ca la prima rulare a serverul DHCP, acesta să nu pornească din cauza lipsei acestei baze de date. Putem să creem, în acest caz una vidă cu comanda

```
touch /var/lib/dhcp/dhcpd.leases
```

Nu este indicat să facem totuși acest lucru, dacă sistemul, din cauza unei opriri bruște (pană de curent, de exemplu) a rămas fără acest fișier. Serverul salvează periodic fișierul `dhcpd.leases` deoarece informațiile din el se pot perima (clienți care s-au deconectat, a expirat o închiriere, etc.). La o asemenea salvare, cel vechi va avea o copie de backup în fișierul `dhcpd.leases~`. Putem să rezolvăm problema situației în care am rămas fără fișierul amintit prin redenumirea copiei de backup.

Pentru mai multe informații despre sintaxa fișierului `dhcp.conf` se poate consulta documentația de la adresa citată 1.

Rutarea. Comanda route. Optiuni de rutare

- 1) dacă rețeaua nu conține subrețele, nu este necesară nici o rutare explicită
- 2) rutări statice, folosite pentru rețele de dimensiuni mici sau medii, având calea origine-destinație relativ simplă
- 3) rutări dinamice, în care caile spre destinație sunt determinate la momentul trimiterii pachetelor, prin protocoale de rutare, care folosesc demonii `routed` și `gated`

Rutările statice pot fi gestionate folosind comanda `route`

- a) `route [-v] [-n]` afișează tabela de rutare, opțiunea `-v` da și informații suplimentare, `-n` afișează adrese numerice în loc de nume

Același efect se poate obține și cu comanda

```
netstat -nr
```

- b) `route [-v] [-A familie] add [-net | -host] dest [optiuni]`
adauga rutări statice

- c) `route [-v] [-A familie] del [-net | -host] dest [optiuni]`
sterge rutări statice

optiunile:

- | | |
|-----------------------------|---|
| <code>-v</code> | afișează informații suplimentare |
| <code>-A familie</code> | specifică o familie de adrese |
| <code>-net</code> | specifică faptul ce dest este o rețea |
| <code>-host</code> | dest este un calculator |
| <code>dest</code> | adresa calculatorului sau rețelei destinație, fie numerică, fie prin nume |
| <code>netmask adresa</code> | utilizează această mască atunci când se adaugă o rutare |
| <code>default</code> | stabilește rutarea implicită care va fi utilizată, |

atunci cand nu exista o alta ruta catre destinatie
gw gateway ruteaza pachetele printr-un gateway. Calea catre acesta
trebuie definita in prealabil printr-o rutare statica
dev dispozitiv asociaza rutarea cu dispozitivul specificat (acest parametru
nu e necesar)

Ex:

a) route add -net 192.130.21.0 netmask 255.255.255.0 dev eth1
adauga o rutare catre reseaua 192.130.21.0 prin placa de retea eth1

b) route add default gw 194.102.62.8
adauga o rutare implicita (care va fi utilizata cand nu exista o alta rutare)
prin masina 194.102.62.8 (goliat.info.uvt.ro)

Testarea accesibilitatii unei alte masini

Se poate face cu comanda ping, care are sintaxa
ping host

unde host este numele masinii destinatie, sau IP-ul său. Comanda ping trimite un pachet de date în mod repetat către host-ul destinație, afișând timpii de răspuns și numărul de pachete pierdute. Dacă nu specificăm câte pachete să fie trimise, aceasta se va repeta de un număr mare de ori (infinite), astfel că putem întrerupe aceasta cu combinația de taste Ctrl-C.

Comanda traceroute afiseaza si ruta pe care circula pachetul catre destinatie.
Sintaxa ei este
traceroute destinatie

Exemple:

ping goliat.info.uvt.ro
traceroute www.cinemagia.ro

Activarea interfetei de retea se face cu comanda

ifup nume_interfata

(nu incercati aceasta comanda, nici pe urmatoarea, pentru ca veti fi deconectati toti si repararea problemei, adică activarea din nou a placii de retea, sau altfel spus a interfetei de retea, nu se poate face decat de la consola statiei respective)

Comanda

ifdown nume_interfata
dezactiveaza interfata de retea specificata.