

[Show pagesource](#)[Old revisions](#)[Recent changes](#) [Search](#)Trace: » [lab1](#) » [lab2](#) » [lab3](#)**Table of Contents**

- Interfețe de rețea și rutare
- Interfețe de rețea
- Bridging
- Rutare
- Autoconfigurare
- NAT
- USB

Interfețe de rețea și rutare

Interfețe de rețea

O interfață de rețea este reprezentarea la nivelul sistemului de operare a unui canal de comunicație, fizic sau virtual, prin care se pot trimite și/sau primi date conform protocoalelor standard folosite în Internet. Interfețele sunt denumite în Linux după protocolul de cel mai jos nivel, urmat de un număr folosit pentru deosebirea interfețelor de același fel. Niște exemple uzuale sunt:

lo	Local Loopback - se primește înapoi ce se trimite
eth0, eth1, ...	Ethernet, cu sau fără fir
wlan0, wifi0	Alias la un eth* pentru interfețele wireless
ppp0	Legătură punct-la-punct, prin modem sau peste Ethernet
usb0	Legătură prin USB, de ex. cu modemurile pentru cablu TV

Fiecare interfață are asociate date pentru protocoalele de nivel legătură de date și rețea. Cele mai importante sunt adresa hardware (MAC, Medium Access Control, pentru nivelul legătură de date) și IP (Internet Protocol, nivelul rețea). Aceste date se pot vizualiza și modifica prin utilitarul `ifconfig`. Rulat fără argumente `ifconfig` afișază interfețele active, parametrii lor și niște statistici. `ifconfig` a afișază și interfețele inactive. Pentru a activa o interfață se folosește `ifconfig eth0 up`, pentru a o opri se folosește `ifconfig eth0 down`, unde `eth0` este interfața. Pentru a modifica adresa MAC a unei interfețe: („hardware is ethernet, has address ...“)

```
ifconfig eth1 hw ether 00:0e:42:c0:ff:ee
```

Pentru a atribui o adresă IP:

```
ifconfig eth1 10.0.0.1 netmask 255.255.255.0
```

sau varianta scurtă:

```
ifconfig eth1 10.0.0.1/24
```

Pentru a adăuga o a doua adresă IP:

```
ifconfig eth1:0 10.0.42.1 netmask 255.255.255.0
```

Se pot adăuga mai multe adrese, `eth1:1`, `eth1:2` ș.a.m.d. Pentru a șterge respectiva adresă:

```
ifconfig eth1:0 down
```

Platforma NGW100 dispune de două interfețe Ethernet 100BASE-TX. Interfața `eth0` este etichetată WAN pe placă și este configurată pentru conectarea la Internet. Interfața `eth1` este etichetată LAN și permite conectarea unei rețele locale. Se conectează pe portul WAN cablul dinspre switch și pe portul LAN stația de lucru folosind un cablu scurt tip patch cord. Astfel, stația de lucru are acces la Internet prin intermediul NGW100, care este configurată în acest sens:

Exercițiu

1. Adăugați adresa IP 10.0.1.1 cu masca 255.255.255.0 interfeței `eth1` de pe NGW100, alături de adresa deja existentă 10.0.0.1.
2. Pe PC adăugați adresa IP 10.0.1.2 alături de cea existentă (10.0.0.x). Folosiți `sudo ifconfig`.
3. Efectuați ping de pe unul din sisteme către fiecare adresă de pe celălalt pentru a verifica funcționarea fiecărei subrețele.

Bridging

Pentru a conecta PC-ul la Internet prin intermediul NGW100, acesta din urmă trebuie să facă o legătură între interfețele sale `eth0` și `eth1`. Acest lucru se poate face atât la nivelul legătură, cât și la nivelul rețea.

Bridging este o tehnică care se folosea atunci când ethernet-ul era un mediu partajat, și se dorea separarea unei rețele fizici în două părți pentru a minimiza coliziunile, comportamentul fiind asemănător cu ceea ce face un switch de nivel 2.

Crearea unui bridge în Linux se reduce la o comandă simplă:

```
brctl addbr br0
```

Apoi trebuie adăugate interfețe la acest bridge, cu:

```
brctl addif br0 eth0
```

Fiecare interfață care participă la bridge nu mai are nevoie de ip, deci trebuie sa scoatem ip-urile asociate interfețelor care participă la bridge

```
ifconfig eth0 down
ifconfig eth0 0.0.0.0 up
```

Pentru a putea accesa și dispozitivul, putem asocia un IP bridge-ului un IP

```
ifconfig br0 10.0.0.1/24
```

Exercițiu

- Conectați-vă la Internet prin intermediul NGW-ului cu un bridge
- ATENȚIE: lucrați din consola serială, odată ce dezactivați interfețele nu o să vă mai puteți conecta la el până nu puneți noul IP pe bridge!

Rutare

Pentru a conecta PC-ul la Internet prin intermediul NGW100, acesta din urmă trebuie să facă o legătură între interfețele sale eth0 și eth1, la nivelul rețea.

Când un pachet sosește pe una din interfețe, dacă el are altă adresă IP destinație decât cea a interfeței, este în mod normal aruncat (dropped). Dacă este pornit așa-numitul forwarding IP, acel pachet este trimis mai departe (sau aruncat) conform tabelii de rutare, de obicei pe altă interfață. Ideea în cazul NGW100 este de a trimite pe eth1 pachetele primite pe eth0 care nu sunt destinate sistemului local și invers.

Utilitarul route afișază tabela de rutare și permite modificarea acesteia. Un exemplu de tabelă de rutare pe NGW100 este următorul:

```
~ # route
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.0.0       *              255.255.255.0  U        0      0      0 eth1
192.168.1.0    *              255.255.255.0  U        0      0      0 eth0
default        192.168.1.1    0.0.0.0        UG       0      0      0 eth0
```

Se observă că NGW100 a fost conectat în rețeaua 192.168.1.0/24. În laborator adresa va fi cel mai probabil alta. Când un pachet este primit pe o interfață și trebuie forward-at (nefiindu-i destinat mașinii locale), sau când este generat pe mașină, kernel-ul se uită la adresa IP destinație a respectivului pachet și începe parcurgerea tabelii de rutare în ordinea listată de route (după câmpul Destination). La primul match se trimite pachetul pe interfața respectivă.

Spre exemplu, în cazul afișat mai sus, să presupunem că un program ce rulează pe NGW100 vrea să inițieze o conexiune spre o stație din rețeaua "LAN", anume 10.0.0.12. Această destinație match-uește prima înregistrare din tabelă: se face ȘI pe biți între masca de subrețea și adresa destinație a pachetului. Dacă e egal cu Destination am match și trimit pachetul pe interfața specificată (eth1). Să ne amintim că LAN este eth1 și WAN este eth0.

Dacă de pe PC aș vrea să mă conectez la o mașină cu adresa 192.168.1.14, care este în aceeași subrețea ca portul WAN al NGW100, se va trimite un pachet de pe PC către portul LAN al NGW-ului (eth1). Forwarding-ul fiind activat, pachetul match-uește a doua înregistrare și este trimis pe eth0 direct către mașina respectivă, adică având adresa MAC destinație a aceleia.

Dacă însă vreau să mă conectez de pe PC sau de pe NGW100 la google.com, care are adresa IP 209.85.171.99, pachetul match-uește ultima intrare din tabelă, default, care este un alias pentru 0.0.0.0. Observați că masca (0.0.0.0) ȘI orice altă adresă care nu a match-uit altă intrare dă 0.0.0.0, deci orice pachet va face match pe ruta default. Pachetul va fi trimis către Gateway (192.168.1.1), dar nu către adresa sa IP (destinația IP este 209.85.171.99), ci MAC.

Legătura între adresele IP și MAC într-o subrețea se face prin protocolul ARP (address resolution protocol). Utilitarul arp listează maparea și permite modificarea ei:

```
~ # arp -a
? (192.168.1.1) at 00:0E:2E:C7:D1:2E [ether] on eth0
? (192.168.1.56) at 00:07:85:92:1D:4C [ether] on eth0
? (192.168.1.42) at 00:0D:87:4D:C1:F0 [ether] on eth0
```

Prima linie reprezintă gateway-ul, a doua și a treia sunt niște stații de pe care m-am conectat prin ssh. La adăugarea unei adrese IP pe o interfață se adaugă automat intrarea corespunzătoare în tabela de rutare pentru subrețeaua respectivă.

Exercițiu

- Observați ieșirea comenzii route și identificați linia corespunzătoare subrețelei 10.0.1.0/24 pe care ați adăugat-o anterior.
- Ștergeți adresa IP respectivă și observați dispariția intrării din tabela de rutare.
- Observați ieșirea comenzii ifconfig pe NGW100, respectiv pe PC, în particular adresele MAC și IP pentru interfețele prin care sunt conectate (subrețeaua 10.0.0.0/24).
- Observați în fiecare caz pe cealaltă mașină ieșirea comenzii arp -a, verificând că pe una din linii se află adresele IP și MAC ale primii mașini.

Autoconfigurare

Există două posibilități de conectare a unui calculator la o rețea: configurare statică (manuală), respectiv dinamică (autoconfigurare). Configurarea statică implică atribuirea manuală a adreselor IP pe interfețe și specificarea rutei default (sau a mai multor rute în rețele avansate) cu comanda route add. În mod uzual se preferă însă configurarea automată. Aceasta se realizează prin protocolul DHCP (Dynamic Host Configuration Protocol). La pornirea mașinii se rulează un client de DHCP (numit de obicei dhclient pe distribuțiile Linux pentru sisteme mari și udhcpd (micro DHCP client) pe cele embedded). Clientul de DHCP comunică prin mesaje tip broadcast cu un server DHCP (de exemplu un proces dhcpd rulând pe un router Linux), care îi atribuie o adresă IP și o mască de subrețea. De asemenea, serverul DHCP poate oferi una sau mai multe variante de rută default (gateway) și servere DNS (Domain Name System).

NGW100 încearcă să se autoconfigureze pe interfața eth0 („WAN”) și oferă posibilitatea de autoconfigurare calculatoarelor conectate pe interfața eth1 („LAN”), asemenea rutelor embedded din comerț. Când PC-ul este conectat direct în switch, există un server DHCP în altă cameră, de unde se distribuie adrese IP. Când NGW100 este conectat la switch și PC-ul la NGW100, autoconfigurarea PC-ului se face de către NGW100, procesul fiind transparent pentru sistemul de operare de pe PC. În primul caz veți avea o adresă IP în subrețeaua laboratorului, în al doilea caz în 10.0.0.0/24. În primul caz veți avea ca gateway și servere DNS niște mașini, în al doilea caz pe NGW100 (10.0.0.1). Configurarea de pe PC se regăsește în al doilea caz pe NGW100. Programul de pe NGW100 care acționează ca server DNS și DHCP se numește dnsmasq.

Comanda traceroute arată calea urmată de pachete până la un calculator din Internet. Încercați traceroute google.com pe PC, primul nod putând

bserva că este 10.0.0.1.

NAT

După cum se poate ușor observa, cele două interfețe ale NGW100 fac parte din subrețele diferite. În cazul ilustrat mai sus, NGW100 are adresa 192.168.1.102 pe eth0, cu care se conectează la WAN („rețeaua mare”), și 10.0.0.1, prin care acționează ca gateway pentru LAN („rețeaua mică”). Dar adresele din rețeaua 10.0.0.0/8 nu sunt rutabile în Internet, fiind numite și adrese private. Același este și cazul adreselor 192.168.0.0/16, dar vom ignora pe moment acest fapt și vom presupune că adresa interfeței eth0 este direct vizibilă în Internet, iar cea a eth1 nu. Mașinile din LAN vor comunica cu cele din WAN prin intermediul adresei de WAN (eth0) a NGW100, proces numit Network Address Translation (NAT). Acesta este un lucru bun, deoarece providerul de servicii Internet va oferi în multe cazuri o singură adresă IP publică, sau un număr limitat, prin care trebuie conectată la Internet o rețea locală formată din multe mașini. De exemplu, există un router similar cu NGW100 care are adresa publică 86.121.xxx.yyy. Rețeaua locală este 192.168.1.0, care este LAN pentru primul router și WAN pentru NGW100. Înlănțuirea este transparentă.

Pe scurt, ideea NAT este maparea adreselor din LAN la o adresă din WAN. De obicei adresele din LAN sunt private, pentru a le putea alocă discreționar, iar cea din WAN este publică, dar nu obligatoriu (se poate face NAT înlănțuit). Există și alte mapări posibile (mai multe adrese în WAN) pe care nu le vom discuta.

Pentru a ilustra procesul, accesați website-ul <http://www.ip-adress.com/> sau unul similar, care vă va raporta adresa publică prin care vă conectați la Internet. Aceeași informație se poate obține și din traceroute (verificați). În orice caz, adresa IP respectivă nu este în rețeaua 10.0.0.0/24 configurată pe NGW100, nu este cea afișată de ifconfig pe PC. S-a realizat deci o translatare a adresei IP.

Acest proces se realizează în Linux cu subsistemul netfilter, același care se ocupă de filtrarea pachetelor, având rol de firewall. Utilitarul pentru configurarea netfilter se numește iptables.

netfilter/iptables folosește mai multe tabele (numite raw, mangle, nat, filter) care definesc reguli de filtrare și de modificare a pachetelor. Tabelele au definite niște lanțuri de prelucrare numite PREROUTING, INPUT, FORWARD, OUTPUT și POSTROUTING (nu toate pentru fiecare tabel). Principial, un pachet parcurge lanțurile din diversele tabele, fiind trecut mai departe, aruncat sau alterat după regulile de match și comenzile specificate în prealabil cu comanda iptables.

Pachetele primite de pe o interfață de rețea intră în lanțurile PREROUTING, unde pot fi modificate și filtrate (acceptate sau aruncate). Mai departe, intră în INPUT dacă au ca destinație adresa IP a interfeței, sau în FORWARD dacă nu. În ambele cazuri pot fi din nou modificate și filtrate. Pe INPUT, dacă există socketji deschiși pe adresa sursă și portul TCP/UDP corespunzător, datele din pachet intră în programul care a deschis socketul. Pe FORWARD, se decide pe baza tabelii de rutare pe ce interfață se va trimite mai departe pachetul. Există și OUTPUT pentru pachetele generate local (prin scrierea de către un program a unor date pe un socket), caz în care se poate face modificarea și filtrarea respectivelor pachete, după care se decide pe ce interfață vor fi trimise. Înainte de a ieși prin interfața respectivă, pachetele (din OUTPUT și FORWARD) intră în POSTROUTING, unde pot fi modificate. Procesul complet este descris în schema de alături:

Regulile de filtrare și prelucrare din lanțurile tabelilor sunt definite prin apelarea iptables manuală, din consolă, sau automată, la bootare, printr-un iniț script.

Pe versiunea curentă a sistemului de operare instalat pe NGW100, script-ul respectiv este /etc/init.d/S22iptables.

Pe scurt, comenzile date la inițializarea sistemului de acest script sunt următoarele:

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

activează forwarding-ul IP.

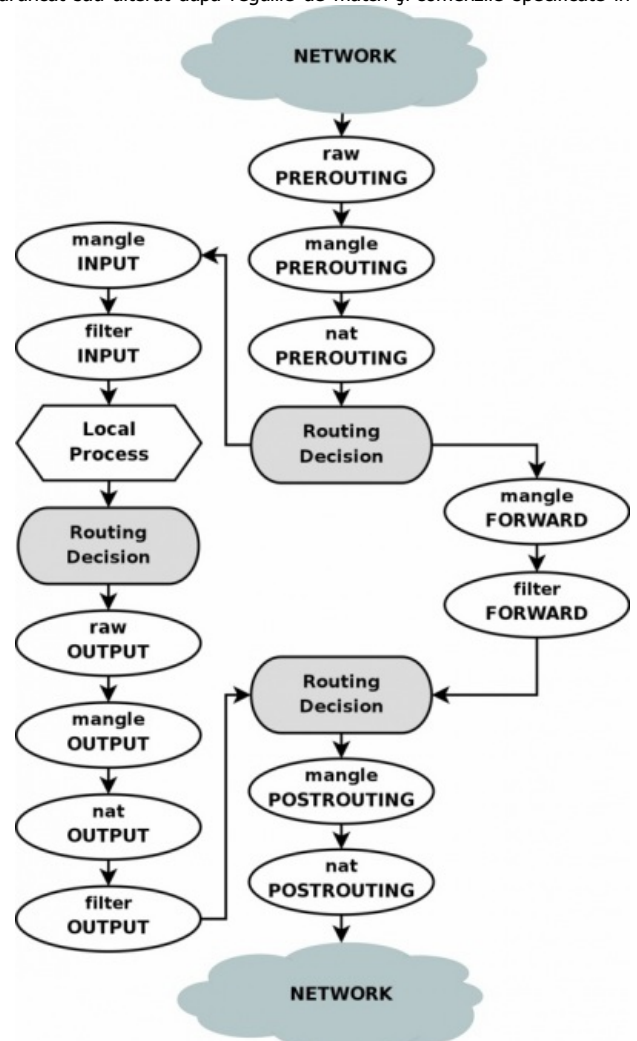
```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

adaugă în lanțul de prelucrare post-rutare al tabelului nat o regulă ce specifică următorul comportament: „pachetele ce trebuie să iasă prin interfața eth0 vor fi mascaradate”. Mascaradarea este un concept NAT ce implică înlocuirea adresei IP sursă a pachetelor ce vin din LAN (de tipul 10.0.0.0 în cazul nostru) cu adresa IP de WAN (a interfeței pe care ies, 192.168.1.102 în cazul meu). Astfel, ruterele următoare și mașina destinație vor crede că pachetele vin de la NGW100, nu de la un calculator din spatele lui.

```
iptables -A FORWARD -i eth0 -o eth1
-m state --state RELATED,ESTABLISHED
-j ACCEPT
```

(pe o singură linie) forwardează de pe eth0 pe eth1 (dinspre WAN spre LAN) acele pachete care au legătură cu o conexiune inițiată dinspre LAN spre WAN. Cu alte cuvinte, dacă încerc să accesez un website de pe un calculator din LAN, pachetele întoarse de server trebuie să se întoarcă la mine. Alte pachete nu au motiv să ajungă în rețeaua locală.

Pachetele primite din WAN în numele unui calculator din LAN vor avea ca adresă IP destinație adresa de pe eth0 a NGW100, ele fiind răspunsuri la pachete cu adresa sursă mascaradată. Când vor fi trimise pe eth1, adresa destinație va fi automat înlocuită cu cea a mașinii care a inițiat conexiunea, pentru ca întregul proces să fie transparent pentru aceasta. Kernelul memorează conexiunile inițiate dinspre eth1 și face înlocuirea adresei pe baza porturilor TCP sau UDP pe care le alocă pe interfața eth0. Acest lucru este asigurat de comanda MASQUERADE dată anterior. iptables A FORWARD i eth1 o eth0 j ACCEPT forwardează de pe eth1 pe eth0 (din LAN spre WAN) toate pachetele.



USB

Diverse echipamente de comunicație (modemuri de cablu TV, modemuri ADSL) permit conectarea unui PC la Internet printr-un port Ethernet

standard sau printr-un port USB. Este și cazul NGW100, care dispune de un port USB. Astfel, păstrând eth0 ca interfață WAN, conectarea unui al doilea LAN se poate face prin interfața usb0. Bineînțeles sunt posibile și alte topologii de rețea, spre exemplu conectarea la doi provideri de servicii Internet prin interfețele eth0 și eth1 și a unei rețele locale prin usb0 cu distribuirea traficului între cele două conexiuni, dar acest lucru necesită pași suplimentari complecși. Crearea interfeței usb0 se face prin încărcarea unui driver, cu comanda `modprobe g_ether`. Aceasta inserează în kernel un modul numit `g_ether.ko`, cu încărcarea prealabilă a eventualelor alte module de care acesta depinde. Modulul este unul standard, distribuit cu kernelul, și se află în directorul `/lib/modules/`uname -r`/kernel/drivers/usb/gadget`. Nu trebuie specificată manual calea, `modprobe` o calculează singur.

Comanda `uname -r` afișează versiunea de kernel care rulează, iar construcția ``comandă`` returnează ieșirea comenzii. În cazul sistemului curent kernelul (și directorul corespunzător) este 2.6.25.10.atmel.2. Când sunt mai multe versiuni de kernel instalate, din care una rulează, această metodă permite încărcarea modulului potrivit (nu se poate încărca un modul compilat pentru altă versiune de kernel).

Dacă `modprobe` nu funcționează, încercați `insmod` cu calea completă. O dată încărcat modulul, puteți vedea mesajele date de acesta cu comanda

```
dmesg|tail
```

(display messages, take output and display its tail):

```
ether gadget: using random self ethernet address
ether gadget: using random host ethernet address
usb0: Ethernet Gadget, version: May Day 2005
usb0: using atmel_usba_udc, OUT ep2 IN ep1 STATUS ep3int
usb0: MAC 36:c9:a8:f2:46:0c
usb0: HOST MAC 32:0a:d6:a7:10:5f
usb0: RNDIS ready
usb0: high speed config #1: 100 mA, Ethernet Gadget, using CDC Ethernet
```

Modulul crează un dispozitiv USB cu clasa CDC (Communications Device Class), pentru care există un driver standard independent de producătorul de hardware (ca și pentru HID - Human Interface Device ș.a.). Se va crea o interfață `usb0` pe NGW100 și una pe PC, inițial dezactivate. Dacă nu sunt inițial down, opriți-le. Nu uitați să conectați dispozitivul USB.

Exercițiu

Replicați funcționalitatea oferită de NGW100 prin interfața `eth1` pe interfața `usb0`. Pentru aceasta, vor fi parcurși următorii pași pe NGW100:

- Atribuiți adresa IP 10.0.1.1/24 (netmask 255.255.255.0) pe interfața `usb0`, ceea ce o va și activa.
- Editați fișierul de configurare `/etc/dnsmasq.conf` pentru serverul DHCP și DNS.

Editarea se face cu `vi`. Găsiți linia `interface=eth1` tastând `/eth1.` / face căutare. Cu tasta Insert activați modul de editare. Adăugați dedesubt linia `interface=usb0`, dacă nu este deja adăugată. Căutați 1-2 ecrane mai jos linia `dhcprange=10.0.0.20,10.0.0.254,72h`. Aceasta reprezintă domeniul de adrese IP asignat automat calculatoarelor din LAN (conectate prin `eth1`). Nu o confundați cu liniile-exemplu ce încep cu `#`. Adăugați o linie corespunzătoare pentru rețeaua 10.0.1.1, spre exemplu `dhcprange=10.0.1.20,10.0.1.254,72h`. Apăsăți tasta Esc, apoi dați comanda `:wq` care va salva modificările și va ieși din editorul `vi`.

- Restartați `dnsmasq` cu comenzile:
 1. `killall dnsmasq`
 2. `/etc/init.d/S21dnsmasq`

(scriptul nu este deocamdată corect scris pentru a interpreta argumentele `start`, `stop`, `restart`).

- Pe PC, autoconfigurați interfața `usb0` cu comanda `sudo dhclient usb0`. În acest punct, ar trebui să aveți conectivitate între PC și NGW prin USB. Verificați cu `ping`.
- NGW100 activați NAT cu comenzile descrise mai devreme, modificate prin înlocuirea `eth1` cu `usb0`:

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

(nu mai e necesară în cazul nostrul, mascaradarea pe `eth0` e deja activată de la configurarea `eth1`)

```
iptables -A FORWARD -i eth0 -o usb0
-m state --state RELATED,ESTABLISHED
-j ACCEPT (pe o singură linie)
iptables -A FORWARD -i usb0 -o eth0 -j ACCEPT
```

- Scoateți cablul din mufa LAN a NGW100 (`eth1`) și verificați că puteți accesa un site Web prin conexiunea USB. Reintroduceți cablul. Dezactivați interfața `usb0` pe PC și autoconfigurați interfața Ethernet sau restartați subsistemul de networking.

si/lab/lab3.txt · Last modified: 2009/10/27 15:00 by dan

Show pagesource Old revisions

Login Index Back to top

Except where otherwise noted, content on this wiki is licensed under the following license: CC Attribution-Noncommercial-Share Alike 3.0 Unported

