Rutarea în Internet

Mihai Budiu — mihaib+@cs.cmu.edu
http://www.cs.cmu.edu/~mihaib/

Aprilie 1998

Subiect:

Cunoştinţe necesare:

Cuvinte cheie:

Cuprins

1 Internet-ul: scurt istoric

La mijlocul anilor '60 compartimentul de cercetare al Departamentului Apărării al Statelor Unite (DARPA: Defense Advanced Research Projects Agency), un organism beneficiind de un buget strivitor, a pornit un proiect de cercetare pentru construirea unei reţele strategice pentru transmisiunea datelor între calculatoarele Pentagonului. Reţeaua trebuia să satisfacă două cerinţe:

- Să lege între ele scumpele calculatoare ale agenţiei şi eventual ale altor centre de cercetări;
- Să reziste cu succes în faţa unui atac nuclear.

Reţeaua sponsorizată de DARPA a fost numită ARPAnet (network, net = reţea). Proiectul iniţial a fost făcut în 1966 iar în 1969 primul prototip care lega 4 calculatoare, amplasate la mai multe universităţi şi laboratoare de cercetare, era operaţional.

În acelaşi timp universitatea Stanford construia un alt prototip de reţea, sponsorizat tot de DARPA, care folosea undele radio pentru transmisiunea datelor (ARPAnet transmitea datele prin linii telefonice închiriate). Cercetătorii de la Stanford ar fi dorit să interopereze cu cealaltă reţea. Din cauza asta proiectanţii au adoptat o serie de reguli asemănătoare pentru transmisiunea datelor, dar au construit ansamblul în aşa fel încît cele două reţele componente să poată fi administrate şi dezvoltate independent, fără intervenţii speciale ale unei autorităţi centrale. De asemenea un factor crucial în design a fost decizia ca reţeaua să poată opera în acelaşi fel indiferent de natura mediului de transmisie folosit.

La ARPAnet se puteau conecta universităţile şi centrele de cercetare care primeau bani de la DARPA. Multe alte universităţi ar fi dorit însă să participe la conexiune. Fiecare dintre ele a dezvoltat propria ei reţea peste care a suprapus tehnologiile din ARPAnet. Astfel, mai multe reţele diferite au adoptat standardele propuse în ARPAnet pentru comunicaţie şi au putut interopera. La începutul anilor '80 această federaţie de reţele, din care ARPAnet era doar o părticică, s-a numit “Internet”. Numele înseamnă “inter = între”, “net = reţele”, indicînd faptul că protocoalele permit comunicarea între mai multe reţele.

Internet a crescut cu o viteză uluitoare, legînd din ce în ce mai multe universităţi, centre de cercetare şi alte instituţii de acelaşi gen. Pentru că banii pentru funcţionarea sa veneau de la guvern (iniţial de la armată, apoi, de la mijlocul anilor '80 de la National Science Foundation — NSF, care este o agenţie guvernamentală care sponsorizează cercetarea ştiinţifică de la budget), activităţile comerciale erau strict interzise.

Internet-ul s-a expandat în afara Statelor Unite, cuprinzînd instituţii de pe toate continentele. După căderea cortinei de fier el a devenit accesibil şi în fostele ţări comuniste (fiecare participant acoperea costurile unei anumite părţi din Internet, mod de operare care este valid şi astăzi).

În 1992 a fost ridicată restricţia privind activităţile comerciale pe Internet, ceea ce a adus gradual în scenă o cantitate enormă de participanţi. În 1993 ARPAnet a fost complet desfiinţată. Suportul federal pentru reţea a fost încetat complet pe 1 mai 1995; NSF continuă să sponsorizeze accesul unor universităţi. La ora actuală comerţul pe Internet şi tehnologiile Internet constituie una dintre cele mai prospere activităţi din Statele Unite, şi, pe o scară ceva mai mică, din restul lumii. Companii ca Netscape, care s-a ridicat de la absolut nimic, în 1994, la o valoare la bursă de peste 1 miliard (sic!) de dolari, în 1995, semnifică importanţa economică crescîndă a reţelei.

O adevărată industrie de ISP: Internet Service Providers: firme care oferă servicii de conexiune la Internet au apărut în toată lumea. La ora actuală în Internet se estimează ca sunt conectate peste 30 de milioane de calculatoare de pe tot globul, şi că peste 100 de milioane de oameni beneficiază de serviciile lui în mod regulat. Rata de creştere continuă să fie exponenţială. Se estimează că 30% din populaţia SUA foloseşte Internet-ul; acesta a devenit un dispozitiv de uz casnic (aproape) la fel de obişnuit ca maşina de spălat sau televizorul.

România este legată la Internet cel puţin din 1992. La ora actuală nodul cu cea mai mare importanţă este aflat la Politehnica din Bucureşti; în jur de 80% din traficul din România trece prin acest punct, pentru că aici se află un nod care oferă legătură marilor centre universitare din ţară (informaţiile la zi pot fi obţinute de la (`http://www.roedu.net`)[1]). În multe oraşe au apărut furnizori de servicii Internet.

Iată cum deciziile iniţiale de descentralizare şi inter-operare au permis o dezvoltare uluitoare care întrece imaginaţia celor mai vizionari dintre arhitecţii Internet-ului.

Internet-ul are propriul lui corp de standardizare, numit IETF "Internet Engineering Task Force", care este format din experţi din universităţi, centre de cercetare şi mari firme de calculatoare şi telecomunicaţii. Ritmul de lucru la IETF este uluitor de rapid (comparat cu alte corpuri de standardizare), şi este în mare măsură responsabil pentru evoluţia tehnologică colosală a Internet-ului. (Trebuie spus că deşi anumite decizii au rămas neschimbate de la început, o mulţime de reguli din funcţionarea Internet-ului au fost introduse pe parcurs. Acesta este un fapt colosal: e ca şi cum ai schimba bujiile la maşină în timp ce merge!) IETF îşi publică deciziile în nişte documente numite "Request For Comments (RFC)" (Cerere de comentarii), şi care sunt adoptate prin consensul celor implicaţi în funcţionarea Internet-ului. RFC-urile specifică *totalitatea* regulilor pe care trebuie să le respecte calculatoarele care comunică în Internet (reguli care se numesc *protocoale*). Lista completă a RFC-urilor se poate obţine...de pe Internet (`http://rs.internic.net`)[2]. Dacă vreţi să ştiţi cum funcţionează Internet-ul este suficient să citiţi aceste documente. Dar nu e o treaba prea simplă... (sunt cîteva sute!).

2 Acest articol

Despre reţele şi Internet s-a scris enorm. O carte absolut excelentă (şi foarte fierbinte) a publicat în traducere în romāna editura Agora, a lui Andrew S. Tanenbaum (la un preţ ridicol: ediţia

[1] Cele mai multe cunoştinţe despre Internet le-am căpătat lucrînd ca administrator la acest nod în 1995; în continuare administraţia nodului este făcută în mare parte de studenţi ai Politehnicii.

[2] `internic.net` este reţeaua firmei "Network Solutions", care este responsabilă de administrarea domeniilor rădăcină din Internet. Nu vom vorbi despre domenii în acest articol.

americană costă 80 de dolari).

Acest articol va trata deci foarte sumar o singură problemă legată de funcţionarea Internet-ului: "rutarea" (alegerea unei rute între două calculatoare). Despre rutare numai au fost scrise mai multe cărţi; referinţa cea mai bună este Radia Perlman "Interconnections", Addison-Wesley, 1992. Subiectul este foarte generos, şi deloc simplu; tratamentul pe care îl vom da aici este desigur doar orientativ.

În funcţionarea unei reţele există două aspecte oarecum complementare: algoritmii şi protocoalele care fac acea reţea să funcţioneze, şi aparatele, sîrmele şi comenzile administratorilor care operează aparatele "reale". Voi încerca în acest articol să le ating pe amîndouă.

Voi folosi adesea analogii cu o altă reţea reală care prezintă oarecare similitudini cu Internet-ul: poşta. La nivelul la care discutăm în acest articol Internet-ul are o singură misiune: îi dai nişte date (grupate într-un *pachet*) şi îi spui *unde* vrei să ajungă, şi Internet-ul face tot ce poate să ducă datele acolo. La fel ca şi poşta: pui la poştă un pacheţel (sau un plic), scrii pe el adresa de destinaţie, şi apoi speri ca sistemul să-şi facă datoria. Nici poşta şi nici Internet-ul nu garantează că ce le-ai dat o să ajungă în partea cealaltă. Dacă vrei să fii sigur trebuie să-ţi iei măsuri suplimentare de precauţie[3].

3 Protocoale

Un protocol este un set de reguli pe care toată lumea trebuie să le respecte. Cînd întîlneşti un cunoscut îi întinzi mîna dreaptă, şi el ţi-o întinde pe a lui; asta e o regulă dintr-un protocol. Protocoalele fac să ne înţelegem unii cu alţii.

Pentru ca două calculatoare să poată comunica unul cu celălalt amîndouă trebuie să respecte aceleaşi reguli. Ingineria programării ne învaţă că pentru a construi sisteme complexe trebuie să le spargem în piese mai simple, pe care apoi le putem construi independent. Din cauza asta, o reţea de calculatoare se specifică mai multe protocoale independente pe care calculatoarele trebuie să le respecte. Pentru fiecare sub-problemă pe care vrem s-o rezolvăm avem un "mic" protocol.

De pildă avem un protocol prin care asociem unei anumite diferenţe de potenţial de pe sîrmă valoarea 0 şi alteia 1; protocolul acesta probabil specifică care sîrmă ce valoare cară, cît timp trebuie să dureze un semnal pentru "1", care este limita între 0 şi 1, etc.

Din cauza interferenţelor electrice semnalele propagate printr-o sîrmă pot fi stricate. Un *alt* protocol specifică cum anume codificăm un mesaj în aşa fel încît atunci cînd o eroare se strecoară în el avem şanse s-o detectăm.

Şi tot aşa, pentru fiecare problemă avem un alt protocol. Protocoalele astea se folosesc împreună; de exemplu cele două de mai sus vor funcţiona în ambele capete ale unei transmisiuni. Protocolul de detecţie a erorilor va funcţiona "deasupra" celuilalt, pentru că după ce transmitem datele le verificăm corectitudinea. Protocolul pentru erori foloseşte rezultatele oferite de protocolul pentru transmisiune: datele transmise.

De aceea în reţele avem de-a face cu o "stivă" de protocoale. De fapt Internet-ul foloseşte mai mult de o duzină de protocoale în mod curent, care fiecare rezolvă o altă problemă.

Protocoalele bine proiectate pot fi văzute ca nişte piese de Lego, care se îmbină unele cu altele. Adesea putem avea mai multe protocoale care rezolvă aceeaşi problemă, şi putem alege pe

[3]De fapt unul dintre protocoalele din Internet, TCP, se ocupă chiar de asta: retransmite copii ale pachetelor care par să se fi pierdut. În articolul de fată însă nu vom ajunge "pînă" la TCP.

oricare dintre ele. Un protocol bine proiectat nu trebuie să depindă de celelalte împreună cu care funcţionează.

Iată un exemplu din sistemul poştal: dacă vrei să-i scrii unui amic compui o scrisoare şi o pui la poştă. Avem (cel puţin) două protocoale aici: conversaţia dintre prieteni, şi sistemul poştal care mişcă scrisorile. Dacă te grăbeşti trimiţi scrisoarea cu altă companie, de pildă DHL. Ai schimbat protocolul prin care informaţia se plimbă, dar conversaţia cu amicul este aceeaşi: conţinutul scrisorii nu s-a schimbat deloc.

Internetul este caracterizat de folosirea anumitor protocoale. Cele mai importante protocoale folosite de calculatoarele din Internet se numesc IP (Internet Protocol) şi TCP (Transport Control Protocol). TCP a fost proiectat înainte de a exista foarte multă experienţă despre reţele, aşa că violează regula expusă anterior: TCP funcţionează numai cu IP.

Protocolul IP este cel care mută pachetele dintr-un loc într-altul, posibil peste multe conexiuni, iar TCP asigură fiabilitatea comunicaţiei.

Iată schema unor protocoale esenţiale din Internet:

```
                                  +-----+
                                  | NFS |
                                  +-----+
                                     |
+----+  +-----+  +------+ +-----+ +-----+
|SMTP|  | BGP |  |Telnet| | FTP | | RPC |
+----+  +-----+  +------+ +-----+ +-----+
    \       \_________  |   |         |
     \_______________ \ |   |         |
                     \+-----+   +-----+    +-----+
                      | TCP |   | UDP | .. | RIP |
                      +-----+   +-----+    +-----+
                         |         |          |
                      +------------------------+------+
                      |           IP           | ICMP |
                      +------------------------+------+
                            ______|___________
                           |                  |
                        +-----+            +------+
                        | ARP |            | RARP |
                        +-----+            +------+
                           \_______ _________/
                                   |
                        +---------------------------+
                        | Protocolul retelei locale |
                        +---------------------------+
```

4 Alte reţele

Nu trebuie să rămînem cu impresia că Internet-ul este singura reţea pentru transmisiunea de date din lume. Poate nici măcar cea mai mare! O reţea enormă dedicată transmisiunii de date este reţeaua telefonică, care există de mai bine de 100 de ani, care foloseşte propriile protocoale.

Pînă în anii '90 nu era deloc clar că suita de protocoale folosite în Internet (TCP/IP) va domina lumea. Marile firme de calculatoare dezvoltaseră protocoalele proprii, care nu erau publice (structura lor era secret comercial, deci nu puteau fi implementate de oricine; de fapt asta a şi dus la pieirea lor). Cele mai faimoase suite de protocoale aparţineau firmei Digital (protocoalele DECnet) şi IBM (SNA: System Network Architecture). Şi la ora actuală reţelele automatelor bancare din întreaga lume folosesc în mod preponderent SNA. Multe din protocoalele din Internet se inspiră puternic din aceste protocoale de firmă.

Pe deasupra mai exista o suită de protocoale (numite OSI: Open System Interconnection) standardizate de organizaţia mondială a standardelor; suita de protocoale X: X25, X400, etc. Multe reţele europene folosesc în continuare aceste protocoale, mai ales reţelele bancare.

O altă suită de protocoale a fost dezvoltată de marile companii de telecomunicaţii, dar nu a fost încă implementată pe scară largă: protocoalele numite generic ATM: Asynchronous Transfer Mode. ATM promitea anumite calităţi care structural scapă Internet-ului, dar nu este prea clar dacă va avea sau nu succes.

Deşi Internet-ul are o mulţime de hibe, este extrem de bine testat: funcţionează încontinuu de aproape 30 de ani. Lumea a învăţat o grămadă de lecţii despre proiectarea reţelelor cu Internet-ul, care au fost înglobate în structura lui. Dificultatea constă în faptul că nimeni nu poate de fapt înţelege funcţionarea unui sistem atît de complex, cu atît mai puţin prezice. Chiar dacă alte protocoale sunt proiectate, nu este deloc limpede că vor funcţiona la fel de bine în practică. De altfel la începutul anilor '90 se preconiza ca Internet-ul să fie desfiinţat şi înlocuit cu o reţea care foloseşte protocoale standardizate internaţional. Acest lucru nu s-a întîmplat, şi a devenit clar că nu se va mai întîmpla niciodată.

5 Internet-ul: o reţea de reţele

Observaţia cheie pe care trebuie să o facem ca să înţelegem cum lucrează Internet-ul este următoarea: Internet-ul *nu* leagă calculatoare, ci *reţele*. Distincţia este crucială, după cum vom vedea.

O reţea este un număr de calculatoare care pot comunica între ele, fiecare cu fiecare, într-un oarecare mod. Cel mai răspîndit fel de reţea este Ethernet, care leagă mai multe calculatoare între ele folosind un cablu coaxial. Mai avem calculatoarele care vorbesc unele cu altele prin modemuri, folosind felurite protocoale din familia V (de exemplu modemurile de 33,6 kilobiţi pe secundă folosesc V.34bis). Şi aşa mai departe.

Ceea ce face Internet-ul este să ia o colecţie de astfel de reţele, poate foarte diferite arhitectural (primul exemplu istoric de mai sus avea o reţea prin telefon şi una prin radio), şi să pună *peste* mediile de comunicare ale acestor reţele o serie de reguli uniforme. Astfel toate calculatoarele din Internet vor avea nume de acelaşi fel şi vor vorbi acelaşi set de protocoale.

În analogia noastră cu sistemul poştal lucrurile ar sta cam aşa: să ne imaginăm că fiecare ţară din lume are alte reguli şi metode pentru a trimite obiecte între locuitori; în unele locuri obiectele se transportă cu camioane, în alte locuri cu avioane (de exemplu în Hawaii, probabil). Fiecare ţară are propria ei reţea de distribuţie. Dacă toate ţările din lume vor să schimbe între ele scrisori (ceea ce se şi întîmplă), atunci ele cad de acord asupra unui standard comun, independent de modul în care-şi transportă obiectele. De pildă faptul că se folosesc timbre, şi asupra modului în care se scrie o adresă pe plic. Dacă toată lumea face aceste lucruri la fel atunci reţelele poştale pot fi făcute să interopereze.

6 IP

Cel mai important protocol din Internet este protocolul numit IP: Internet Protocol. Scopul principal acestui protocol este să împingă pachetele de la un calculator sursă la o destinaţie din aproape în aproape, folosind legăturile existente între vecini. Dacă problema asta vi se pare simplă atunci încercaţi să o rezolvaţi dumneavoastră: eu vă dau un pachet şi numele unui amic de-al meu; dumneavoastră aveţi misiunea să faceţi acest pachet să-i parvină, fără a avea cea mai mică idee unde stă amicul meu sau cum se ajunge la el. (IP are şi un scop secundar, acela de a sparge pachetele în bucăţele atunci cînd trebuie să traverseze o reţea ale cărei caracteristici nu permit tranzitul pachetelor mari. Dar asta e mai puţin interesant.) IP este descris în detaliu în standardul RFC 791.

IP este un protocol care mută pachetele care i se dau. În interior reţeaua mai foloseşte un protocol special pentru control, care mută pachete generate de reţea însăşi. Acest protocol se numeşte ICMP: Internet Control Message Protocol. Ca să ne facem o idee despre ICMP: dacă un calculator primeşte un pachet care se plimbă de mult prea multă vreme în reţea[4] atunci calculatorul distruge pachetul şi simultan trimite spre sursă un mesaj ICMP care anunţă sursa de pierderea informaţiei. Sursa va trebui să ia acţiuni corective.

Să vedem ce se dă şi ce se cere lui IP.

7 Problema rutării

Fiecare calculator aparţine unei reţele, cu ale cărei componente poate comunica oricînd direct, fără alte intervenţii. Vom vedea imediat cum află un calculator care sunt vecinii lui imediaţi; aceasta este o problemă de configurare care cade în sarcina administratorului de sistem.

Pe de altă parte anumite calculatoare au conexiuni la mai multe reţele simultan; acestea se mai numesc "gateways": "porţi" între reţele. Un pachet care este primit de un astfel de calculator pe *interfaţa* dinspre o reţea poate fi transmis pe cealaltă reţea. Acest proces se numeşte "forwarding" ("înaintare"). Drumul ales de un pachet de la sursă la destinaţie poartă denumirea de "rută" (route). Operaţia de alegerea a unei cărări pentru un pachet se numeşte "rutare" (routing); britanicii ar pronunţa asta aproximativ "ruting", pe cînd americanii ar zice "rauting". Iată un desen care explică fenomenul:

```
 Calculatorul sursa        Ruter intermediar       Calculator dest.
+-------------------+   +---------------------+   +---------------+
| Program sursa     |   |                     |   |   Program dest|
|       \           |   |                     |   |       /       |
|     Protocol IP   |   |     Protocol IP     |   |   Protocol IP |
|           \       |   |      /        \     |   |     /         |
|         Interf R1|   |Interf R1   Interf R2|   |Interf R2       |
+--------------\----+   +--/-------------\----+   +-/ ------------+
               Retea Locala 1          Retea Locala 2
```

Există mai multe feluri de "gateways", dar noi în acest articol o să ne preocupăm cu preponderenţă de porţile care se numesc "rutere" ("router" în engleză). *Un ruter este deci în general un calculator*

[4]Fiecare pachet are un contor care scade spre zero la fiecare pas făcut. Evenimentul descris se petrece atunci cînd contorul ajunge la 0.

conectat simultan la mai multe reţele.

Pentru că avem peste 30 de milioane de calculatoare este clar că nu se poate configura informaţia despre unde se află fiecare în toţi membrii reţelei (de altfel această informaţie este foarte volatilă: în fiecare secundă probabil mii de calculatoare se conectează şi se deconectează). Ştim că calculatoarele sunt foarte bune la prelucrat date, aşa că sarcina calculării rutelor le care lor în sarcină.

Iniţial fiecare calculator are oarecare cunoştinţe despre vecinătatea lui mai mult sau mai puţin imediată; aceste cunoştinţe sunt introduse de administratorul de sistem.

Problema rutării se enunţă astfel:

> Atunci cînd primeşti (tu, un ruter) un pachet pentru o anumită destinaţie trebuie să decizi pe care dintre interfeţele tale să-l trimiţi.

Una dintre interfeţe trebuie să fie întotdeauna mai aproape de destinaţie; dacă fiecare ruter care primeşte un pachet îl trimite apoi pe o interfaţă mai apropiată de destinaţia finală, atunci (prin inducţie matematică) eventual pachetul va ajunge la destinaţie. Simplu, nu?

Problema rutării este (aproape) banală pentru calculatoarele care au o singură interfaţă. Este clar că oricare ar fi destinaţia, dacă ea nu este chiar calculatorul local[5], atunci există o singură interfaţă pe care pachetele pot fi scoase.

Problema rutării este mai complicată pentru calculatoarele care au de-a face două alegeri. Dar înainte de a vedea cum este ea rezolvată, să vedem concret cum se specifică o adresă şi o destinaţie.

8 Adrese

Protocoalele din Internet au fost proiectate cu o grijă deosebită pentru administrare descentralizată. Astfel proiectanţii au construit supape de siguranţă pentru evoluţia protocoalelor. Dacă protocolul evoluează este imposibil să se facă *simultan* în întreaga lume "upgrade". Atunci reţeaua poate funcţiona cu părţi din ea rulînd protocoale mai avansate (care, desigur, trebuie să rămînă compatibile (backwards compatible) cu versiunile anterioare).

Versiunea curentă a protocolului IP este 4. Practic toate calculatoarele din Internet rulează versiunea 4, numită şi IPv4. Cîteva calculatoare rulează versiunea experimentală următoare, încă nestandardizată, numită 6 (nimeni nu ştie unde a dispărut versiunea 5). Articolul de faţă discută numai IPv4.

Ca să poţi să indici pe cineva anume trebuie să ai un nume pentru el. Pentru un calculator, mai pregnant decît pentru un om, un obiect care nu are un nume nu există! Numele calculatoarelor din Internet sunt întregi pe 32 de biţi (4 octeţi), numite *adrese IP*. De fapt trebuie să fim precişi: fiecare *interfaţă* (la o reţea) a unui calculator are o altă adresă; un calculator cu 10 interfeţe (un ruter) va avea simultan 10 adrese IP. Adresele IP se scriu convenţional indicînd în baza 10 valorile celor 4 octeţi separate de semnul (.) punct. Iată o adresă IP: "1.2.3.4". Tern, nu? (Un sistem sofisticat există care permite calculatoarelor să aibă nume mai "umane", cum ar fi "apolo.cs.pub.ro", dar asta este cu totul altă poveste, şi este treaba unor protocoale aflate mult deasupra IP.)

Fiecare adresă IP trebuie să apară *o singură dată* în Internet. Pentru a asigura acest lucru adresele IP sunt alocate de o autoritate centrală în grupe unor sub-autorităţi. Fiecare sub-autoritate apoi alocă adresele la rîndul ei. De exemplu Politehnica din Bucureşti a primit 65536

[5] Se întîmplă ca programe de pe acelaşi calculator să comunice între ele folosind protocoalele de reţea; în acest caz sursa şi destinaţia pachetelor vor fi identice.

de adrese, care toate încep cu octeţii 141.85 (deci adresele între 141.85.0.0 şi 141.85.255.255). Politehnica apoi a împărţit adresele fiecărui departament sau laborator, etc. În felul acesta responsabilitatea poate fi relativ clar trasată.

Dacă un calculator într-o anumită parte a Internet-ului îşi alocă o adresă care nu i se cuvine în cel mai bun caz nu va funcţiona, iar în cel mai rău caz va deranja funcţionarea altor calculatoare; cel mai probabil este că administratorii reţelei la care este conectat calculatorul vor trebui să ia măsuri pentru a-l deconecta.

Există un program numit `whois` ("cine e") care interoghează anumite baze de date care conţin informaţii despre administraţia Internet-ului. Din păcate există mai multe astfel de baze de date, acoperind părţi diferite ale reţelei. În principiu puteţi afla informaţii despre un anumit calculator, sau administrator, sau altceva tastînd `whois informatie-de-aflat`.

Există deci teoretic $2^3$2 adrese IP diferite, ceea ce înseamnă peste 4 miliarde. Aparent adresele ar trebui să ajungă, însă în zilele iniţiale ale Internet-ului s-a cam făcut risipă; de pildă Politehnica din Bucureşti nu a folosit toate cele 65000 de adrese, aşa că multe sunt acum inutile, pentru că nu pot fi luate înapoi. Acest fenomen s-a petrecut în mai multe locuri.

Nu este obligatoriu ca adresele să fie administrate atît de grijuliu pe atît de multe planuri; există standarde de reţele în care adresele sunt astfel făcute încît intervenţia administratorului este absolut inutilă; astfel de reţele se numesc "plug-and-play", pentru motive evidente. Ethernet este o astfel de reţea: oricînd iei două calculatoare care vorbesc Ethernet şi le pui laolaltă şi se vor înţelege *fără nici un fel de configurare*! Reţelele din IPv4 necesită administrare pentru că sunt prea puţine; IPv6, care are adrese pe 16 octeţi (128 biţi!) va putea fi şi el plug-and-play.

Atunci cînd un calculator vrea să trimită date spre un altul va pune datele într-un pachet (sau mai multe), şi la începutului fiecărui pachet va pune un *antet* (header) care conţine printre altele adresa IP a sursei şi a destinaţiei. Cea din urmă este folosită de rutere pentru a înainta pachetul.

9 Reţele şi noduri

Pentru a simplifica problema rutării adresele în Internet nu sunt "plate", ci *ierarhice*. Asta înseamnă că o adresă dă şi o oarecare indicaţie despre *traseul* care trebuie urmat pînă la acea adresă.

Ca să înţelegem mai bine diferenţă, numele unui individ este dintr-un spaţiu de nume plat: nu-ţi spune absolut nimic despre locul în care este acel individ (trebuie să foloseşti ceva gen o carte de telefon).

Pe de altă parte o adresă ca cea pusă pe o scrisoare este ierarhică, pentru că îţi indică cu o precizie din ce în ce mai mare (dacă citeşti de sus în jos) *unde* este destinaţia. O adresă de genul:

```
4108 Wean Hall
Carnegie Mellon University
4000 Forbes Avenue
Pittsburgh, PA 15217
USA
```

îţi spune că locul este în USA, din USA apoi trebuie să alegi statul Pennsylvania (PA), oraşul Pittsburgh, şi aşa mai departe. Din ce în ce mai precis. Aproape sigur mai există undeva o stradă numită "Forbes", dar în oraşul Pittsburgh există una singură (asta este treaba primăriei: să asigure această lipsă de ambiguitate). Astfel adresa este foarte precisă.

Adresele din Internet sunt tot ierarhice, avînd însă numai două componente. Prima componentă este numărul *reţelei*, iar a doua componentă este numărul *calculatorului din reţea*. Reamintiţi-vă că Internet-ul este o colecţie de reţele. Toate calculatoarele care fac parte din aceeaşi reţea vor avea deci aceeaşi valoare pentru numărul reţelei.

O adresă Internet trebuie deci să fie însoţită de *o mască*, al cărei folos este să ne indice care din cei 32 de biţi indică reţeaua şi care calculatorul din reţea. (Un calculator într-o reţea se numeşte pe engleză şi "gazdă" — host.) Masca este întotdeauna un şir de biţi "1" consecutivi plecînd de la cel mai important.

Iată un exemplu de o adresă (am generat-o aleator) însoţită de o mască, de data asta scrisă în baza 2:

```
00101101.01011011.11100011.00010110 --- adresa
11111111.11111111.11100000.00000000 --- masca
\___________________/\____________/
  numarul retelei     nr. host-ului

00101101.01011011.11100000.00000000 --- adresa retelei = adresa & masca
00000000.00000000.00000011.00010110 --- host = adresa & ~masca
```

În baza 10 adresa de mai sus ar avea valoarea: 55.91.227.20. Adresa reţelei este 55.91.224.0, iar adresa calculatorului din cadrul reţelei este 0.0.3.20.

Numerotarea asta poate părea confuză, şi în oarecare măsură şi este.

Informaţia purtată de o adresă şi o mască este foarte importantă: un calculator ştie că dacă ajunge la un calculator dintr-o reţea (oricare), atunci ajunge la orice calculator din acea reţea. Cu alte cuvinte, toate calculatoarele care încep cu acelaşi număr de reţea (cu biţii 55.91.224.0) sunt direct conectate unul la altul.

De asemenea, acest lucru simplifică enorm tabelele de rutare ale ruterelor: un ruter nu trebuie să ştie drumul de parcurs pentru fiecare calculator care începe cu 55.91.224.0: ajungă să ştie unul singur, pentru că aceste calculatoare sunt toate la un loc.

În trecut adresele din Internet aveau masca implicită şi se împărţeau în 3 clase mari, numite A, B şi C. Această organizare este acum depăşită, aşa că nu o menţionăm decît în trecere, pentru a nu spori confuzia.

10 Interfaţa cu sub-reţeaua: ARP, `ifconfig`

Administratorul de sistem configurează un calculator ca la pornire să-şi iniţializeze adresele (şi măştile) pe toate interfeţele. Adresa se poate iniţializa şi inspecta pe o staţie Unix cu comanda `ifconfig` (InterFace CONFIGuration). Iată un exemplu de informaţie pentru o interfaţă aşa cum este el tipărit de `ifconfig`:

```
ifconfig le0
le0: flags=63<UP,BROADCAST,NOTRAILERS,RUNNING>
        inet 132.236.102.10 netmask ffff0000 broadcast 132.236.0.0
```

Numele interfeţei este `le0`, care vine de la Local Ethernet 0 (nu există nici un fel de standardizare pentru aceste nume; pe anumite sisteme Unix puteţi afla toate interfeţele cu comanda `ifconfig -a`). Mai ştiu că adresa IP este 132.236.102.10, şi că masca (tipărită în baza 16) este

255.255.0.0. Cu alte cuvinte adresa reţelei este formată din primii doi octeţi (132.236.0.0), iar adresa calculatorului din această reţea este 0.0.102.10.

Prin convenţie într-o reţea nu poate exista nici un calculator cu o adresa numai din cifre 0 sau numai din cifre 1. Aceste adrese sunt rezervate pentru alte scopuri, pe care le vom ignora aici.

Calculatorul meu chiar dacă ştie numai acest lucru ştie un lucru foarte important: că toate celelalte calculatoare care au adresele de forma 132.236.xxx.xxx sunt accesibile direct prin interfaţă `le0`. Cu alte cuvinte, dacă dau ca destinaţie un astfel de calculator, nu este nevoie de nici un fel de "forwarding": pachetul poate fi pus direct pe interfaţa `le0` şi el va ajunge la destinaţie imediat!

Cum se poate aşa ceva? O mare parte din reţelele locale au o proprietate foarte interesantă: comunicaţia în ele se face prin "difuzare": cînd cineva pune un pachet în reţea absolut toată lumea din reţea vede acel pachet. De altfel este uşor de înţeles: Ethernet-ul seamănă tehnologic cu televiziunea prin cablu, în care un transmiţător pune un semnal pe o sîrmă şi toţi îl aud. Toţi se uită la pachet, dar numai cel căruia îi este destinat îl copiază în memoria proprie. (Destinatarul se auto-identifică comparînd adresa destinaţiei din antetul pachetului cu propria lui destinaţie.)

Era prea frumos ca să fie adevărat: lucrurile sunt în realitate ceva mai complicate (deşi conceptual asta e ideea). Întotdeauna reţeaua locală are la rîndul ei propriile mecanisme de identificare şi numire a calculatoarelor. Deşi este adevărat că fiecare calculator de pe un cablu Ethernet[6] le poate auzi pe toate, comunicaţia se face de obicei tot între perechi de calculatoare astfel: fiecare calculator are o adresă Ethernet, care este unică pe glob, pentru că fiecare fabricant de plăci Ethernet a promis asta unei autorităţi centrale. Cînd un calculator pune un pachet pe un cablu, indică şi adresa Ethernet a destinaţiei.

O interfaţă plasată pe o reţea Ethernet are deci (pe lîngă adresa IP) şi o adresă Ethernet. Cînd calculatorul vrea să comunice cu altul din acelaşi Ethernet, trebuie să indice de fapt adresa Ethernet a destinaţiei, pentru că reţeaua Ethernet habar-n-are de adrese IP. Pentru asta fiecare calculator menţine o tabelă de corespondenţe: pentru fiecare adresă IP din reţeaua locală ştie ce adresă Ethernet corespunde. Tabela asta se construieşte cu ajutorul unui alt protocol, numit ARP (Address Resolution Protocol).

- Pentru cazul unor reţele ca Ethernet-ul se foloseşte proprietatea de "broadcast" a reţelei: cînd vrei să-i trimiţi ceva lui 1.2.3.4 "strigi" pe Ethernet cu ajutorul unui pachet special: "Băi, care ai adresa IP 1.2.3.4?". Cel în seamă vede pachetul tău Ethernet, care conţine adresa ta Ethernet, aşa că poate să-ţi răspundă cu un alt pachet, din care afli corespondenţa între adresele sale IP şi Ethernet.
- Pentru reţele în care nu se poate face broadcast administratorii configurează un server special care ştie toate traducerile;

În Unix puteţi inspecta tabela ARP cu comanda `arp`.

11 Tabelele de rutare

Ce ne facem însă atunci cînd vrem să comunicăm cu un calculator care nu este conectat în aceeaşi reţea locală cu noi? Pachetul nostru va trebui să treacă prin mai multe gateway-uri pînă la destinaţie. Problema care se pune pentru calculatorul meu este: "care este primul gateway"?

Informaţia de acest gen este menţinută de nivelul IP într-o *tabelă de rutare*. Această tabelă descrie pentru fiecare grup de adrese interfaţa care trebuie folosită pentru a ajunge la ele şi primul calculator căruia datele trebuie să-i fie trimise.

Cum arată tabela de rutare şi de unde vin informaţiile din ea?

Răspunsul la această întrebare se poate obţine în trei feluri diferite, depinzînd de importanţa calculatorului care pune întrebarea. Putem distinge trei feluri de calculatoare conectate în Internet, după cum ilustrează figura 1.

[6]Puteţi substitui în loc de Ethernet orice alt tip de reţea locală; lucrurile vor sta la fel.

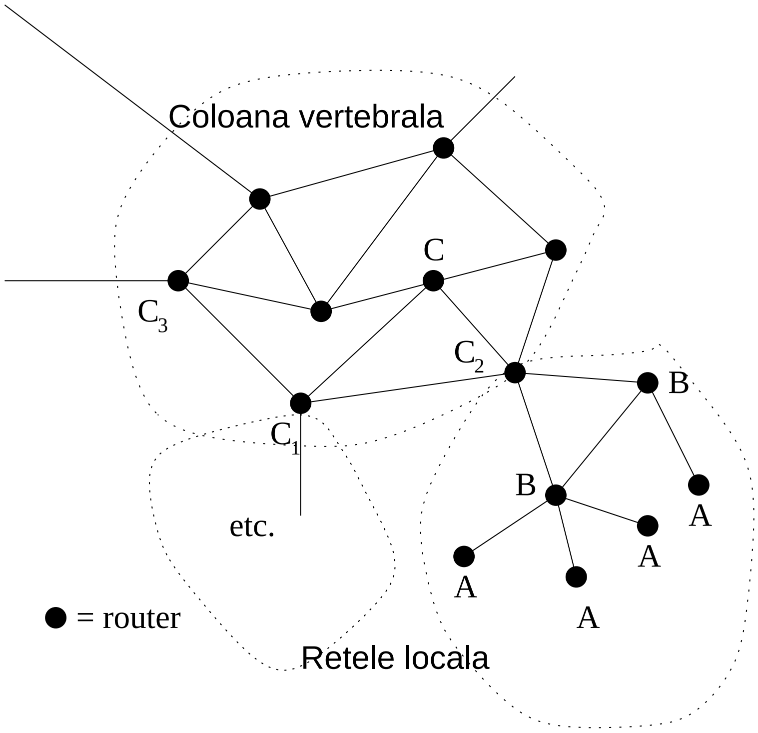

Figura 1: Fragment din Internet

Frunze: orice calculator care are o singură interfaţă este o "frunză" (termen împrumutat din teoria grafurilor) (A în figură) (Atenţie: literele A, B, C nu sunt nicidecum denumiri standard; le-am inventat eu pentru a ilustra acest articol; de exemplu nu trebuie nicidecum confundate cu reţelele de tip A, B, C, care sunt altceva).

Rutere "mici": un astfel de calculator separă o reţea relativ mică de restul Internet-ului; toate pachetele care trec dintr-o parte într-alta trebuie să treacă prin el (B în figură).

Rutere din backbone: "miezul" Internet-ului arată ca o mare pînză de păianjen; aceasta este "coloana vertebrală" a Internet-ului, formată din trunchiuri de mare viteză care se încrucişează în rutere. Pentru un astfel de ruter nu se poate spune că "taie" Internet-ul în două părţi (C în figură).

Modul în care funcţionează cele trei feluri de calculatoare din punct de vedere al rutării este complet diferit.

12 Rutele implicite (default)

Frunzele trebuie să ştie care este ruter-ul "mic" (terminologia nu este standard, dar nu am o idee mai bună) care leagă propria lor reţea de Internet. În figura anterioară, calculatorul A trebuie să ştie că reţeaua sa locală are ca legătură pe B. Acest lucru este de obicei configurat de administratorul reţelei. De ce?

Pentru că atunci cînd A are de trimis un pacheţel la o adresă algoritmul lui va fi următorul:

- Dacă pachetul este pentru reţeaua locală, atunci pachetul este transmis pe interfaţă proprie indicînd destinaţia dorită;
- Altfel pachetul este înaintat calculatorului B cel mai apropiat.

De ce? Pentru că nu există alt drum spre lumea dinafară decît prin B. B este ceea ce se numeşte "ruter implicit" pentru A (default router); majoritatea pachetelor se vor duce la el.

12.1 `route` şi `netstat`

Dacă frunza rulează sistemul de operare Unix, atunci comanda cu care se adaugă şi şterg rute în tabela de rutare se numeşte `route`. De obicei calculatorul execută un script de iniţializare la boot-are prin care instalează ruterul implicit. Puteţi vedea toate rutele instalate ale unui calculator cu comanda `netstat -r`, de la NETwork STATistics, Routes. Ca în desenul nostru, uneori frunzele pot accesa simultan mai multe porţi diferite spre Internet; de obicei una dintre ele este declarată iniţial poartă implicită.

Iată de exemplu rezultatul acestei comenzi pe un calculator frunză:

```
# netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
128.2.0.0       *               255.255.0.0     U      1500 0          0 eth0
127.0.0.0       *               255.0.0.0       U      3584 0          0 lo
default         GW.CS.CMU.EDU   0.0.0.0         UG     1500 0          0 eth0
```

De aici aflu că:

- Interfaţa `eth0` este conectată direct (adică nu prin intermediul unui gateway: *) la reţeaua cu numărul 128.2.0.0 (masca 255.255.0.0). `U` înseamnă "up", adică "funcţionează";
- Interfaţa `lo` este conectată direct la reţeaua 127.0.0.0 (aceasta este o interfaţă fictivă prezentă în toate calculatoarele; adresele care încep cu 127 sunt rezervate pentru utilizări speciale. De exemplu adresa 127.0.0.1 reprezintă întotdeauna calculatorul local — este echivalentul pronumelui "eu");
- În fine, interfaţa `eth0` ne leagă la o reţea pe care se află ruterul implicit, care este `gw.cs.cmu.edu`. Masca 0.0.0.0 ne arată că acesta este un ruter implicit, deci orice pachet pentru o adresă care nu se potriveşte cu celelalte trebuie trimisă lui.

13 Ruterele şi coloana vertebrală (backbone)

Pentru celelalte tipuri de calculatoare situaţia este diferită. Un router "mic" poate fi încă configurat manual: el trebuie să ştie care sunt toate frunzele pentru care el este "responsabil". Un astfel de calculator va schimba permanent informaţii despre topologia reţelei cu vecinii săi. Protocolul prin care ruterele schimba informaţii se numeşte *protocol de rutare*. La ora actuală sunt standardizate aproape o duzină de astfel de protocoale pentru Internet. În figura noastră B-urile vor schimba informaţii: fiecare îi va spune celuilalt cu ce calculatoare "frunză" este direct conectat, iar ambele îi vor raporta lui C acest lucru.

Caracteristica centrală a unui astfel de ruter este că trebuie să memoreze relativ puţine informaţii în tabela de rutare (zeci sau sute de înregistrări), putîndu-se baza în continuare pe rute implicite, care trec prin cel mai apropiat ruter de tip C.

Astfel de rutere pot fi tot staţii de lucru sau calculatoare personale rulînd Unix, sau calculatoare specializate (cum vom vedea mai jos că sunt folosite întotdeauna pentru ruterele de tip C). Foarte adesea astfel de rutere rulează ceea ce se numesc "ziduri de foc" (firewalls). Acestea sunt programe speciale care inspectează fiecare pachet care trece prin ruter şi interzic transferul unor pachete suspicioase. Aceasta previne o sumedenie de atacuri informatice prin Internet şi restrînge şi controlează fluxul de informaţii din/spre reţeaua locală.

14 Protocoale de rutare

Ruterele de tip B sau C nu pot fi configurate manual; cel mai important motiv este că o reţea mai mare suferă în mod frecvent schimbări, care sunt greu de centralizat şi administrat. Din cauza asta astfel de rutere calculează singure topologia reţelei.

Ruterele de tip C sunt întotdeauna calculatoare special construite pentru acest scop; au sisteme de operare speciale, nu au discuri (folosesc memorii nevolatile flash pentru a ţine sistemul de operare), pot suporta foarte multe plăci de reţea (zeci simultan) şi nu fac tot timpul altceva decît să permită administrarea reţelei şi calculul de tabele de rutare. Cam 80% din ruterele din Internet sunt fabricate de firma Cisco din Statele Unite. Un ruter de mare performanţă poate transfera date de ordinul a 10Gbps (giga biţi pe secundă)!

Sarcina unui ruter de tip C este de cu totul altă natură decît cea a ruterelor de tip A sau B. Spre deosebire de acelea, un ruter de tip C trebuie să ştie *întreaga topologie a Internet-ului*, şi nu numai reţelele care se învecinează direct cu el.

Ca să înţelegem de ce se întîmplă acest lucru să considerăm ruterele C_2 şi C_3 din figura de mai sus. Ce se întîmplă dacă C_2 primeşte un pachet pentru un calculator din reţeaua conectată la C_3? Dacă fiecare C şi-ar cunoaşte numai vecinii, atunci C_2 n-ar avea de unde să ştie că vecinul lui, C_1 este la rîndul lui vecinul lui C_3. Din cauza asta fiecare ruter de tip C va spune tuturor vecinilor săi *toate* informaţiile pe care le are.

Iniţial, la boot-are, un ruter C va şti numai vecinii săi (introduşi tot de administrator). După prima rundă de comunicaţii va afla însă şi vecinii vecinilor săi, după care vecinii vecinilor vecinilor săi, şi aşa mai departe. Cum Internet-ul la ora actuală are un diametru în jur de 30 (adică distanţa dintre oricare două calculatoare trece prin mai puţin de 30 de rutere), după 30 de runde de comunicaţie între vecini orice ruter cunoaşte întreaga topologie a reţelei! Asta înseamnă zeci de megaocteţi de informaţie. Dar fără această informaţie calculatoare îndepărtate n-ar avea nici o idee cum să ajungă unul la altul.

În realitate tabelele de rutare ar fi foarte mari, aşa încît pentru întregul Internet se foloseşte în mod normal o adresare ierarhică pe 3 nivele. Nivelul cel mai de jos este calculatorul, apoi reţeaua (pe astea deja le-am văzut), iar deasupra vine *sistemul autonom* (Autonomous System, AS). În general un sistem autonom este o unitate administrativă. De exemplu întreaga reţea educaţională a României este un sistem autonom. Informaţia este apoi compactată astfel în ruterele de tip C:

- Un ruter cunoaşte topologia completă a întregului sistem autonom din care face parte;
- Pentru celelalte sisteme autonome un ruter ştie numai două feluri de informaţii:
  - Ce reţele sunt în fiecare;
  - Ce şir de sisteme autonome trebuie traversat pentru a ajunge la un sistem autonom anume.

Ca să înţelegem de ce asta comprimă foarte tare informaţia, să ne gîndim din nou la sistemul poştal, în care o ţară ar fi un sistem autonom şi un oraş ar fi o reţea. Să presupunem că toate scrisorile circulă terestru. Atunci informaţia completă de rutare între Bucureşti şi Londra ar trebui să fie lista tuturor oraşelor care trebuie traversate pe drum. Pe de altă parte lista sistemelor autonome ar fi numai lista *ţărilor* care trebuie traversate. Ideea este că, dacă destinaţia este înafara propriului sistem autonom, dai pachetul primei ţări pe drum (îl duci în capitală = ruter de margine al sistemului autonom), care se descurcă.

Ruterele vorbesc în Internet două feluri diferite de protocoale de rutare, schimbînd feluri diferite de informaţie. Anumite protocoale sunt vorbite *între* ruterele din acelaşi sistem autonom, şi astfel de protocoale se numesc *protocoale interne*. Cele mai folosite sunt RIP (Routing Internet Protocol) şi OSPF (Open Shortest Path First).

Ruterele vorbesc cu vecinii lor din alte sisteme autonome protocoale *externe*. Cele mai faimoase sunt EGP (External Gateway Protocol), care este acum treptat înlocuit cu BGP (Border Gateway Protocol).

15 Administrarea reţelei

Administrarea şi configurarea unei reţele de calculatoare este o slujbă foarte interesantă, mai ales a unei reţele de mare întindere. Problema este că adesea operatorul se află la foarte mare distanţă de echipamentele supravegheate, aşa că administraţia reţelei se face...folosind reţeaua. Administratorul are conturi pe ruterele din domeniile administrate şi le configurează de la distanţă (presupunînd că funcţionează...).

Există un protocol special numit SNMP (Simple Network Management Protocol), dar care nu este deloc simplu, prin care ruterele comunică evenimente deosebite şi prin care administratorul de reţea configurează parametrii de funcţionare ai ruterelor.

Dar cele mai folositoare scule ale administratorului sunt două mici programe: `ping` şi `traceroute`.

15.1 `ping`

`ping` vine de la Packet INternet Gropper (pachet care bîjbîie prin Internet). Programul `ping` trimite la o destinaţie un pachet ICMP cu rugămintea destinaţiei de a întoarce pachetul de îndată ce-l primeşte. Este metoda cea mai bună de a verifica dacă rutele între sursă şi destinaţie sunt bine configurate (în ambele sensuri!), şi de a măsura încărcarea reţelei (din timpul de întîrziere între transmisie şi recepţie). Iată un exemplu de folosire a lui `ping`:

```
# ping pub.pub.ro
PING pub.pub.ro (141.85.128.1): 56 data bytes
64 bytes from 141.85.128.1: icmp_seq=0 ttl=238 time=722.7 ms
64 bytes from 141.85.128.1: icmp_seq=1 ttl=238 time=730.6 ms
64 bytes from 141.85.128.1: icmp_seq=2 ttl=238 time=825.5 ms
64 bytes from 141.85.128.1: icmp_seq=3 ttl=238 time=991.5 ms
```

Asta arată cum fiecare pachet se întoarce de la adresa 141.85.128.1 după un timp cuprins între 722 şi 991 milisecunde.

15.2 `traceroute`

`traceroute` afişează întregul traseu al unui set de pachete spre o anumită destinaţie (fiecare pas făcut). Cu opţiunea `-n` adresele sunt afişate sub formă numerică (IP) numai. Iată un exemplu:

```
# traceroute pub.pub.ro
traceroute to pub.pub.ro (141.85.128.1), 30 hops max, 40 byte packets
 1  GW.CS.CMU.EDU (128.2.254.36)  0.757 ms  0.738 ms  1.029 ms
 2  RTRBONE.NET.CMU.EDU (128.2.1.2)  1.378 ms  1.271 ms  1.292 ms
 3  nss5.psc.net (192.88.114.254)  3.46 ms  4.278 ms  3.543 ms
 4  atm0-0-3.pit-bb1.cerf.net (134.24.137.21)  3.513 ms *  37.252 ms
 5  atm8-0.bos-bb1.cerf.net (134.24.46.93)  15.473 ms  16.077 ms  15.966 ms
 6  atm10-0.nyc-bb2.cerf.net (134.24.46.5)  20.593 ms  20.937 ms  20.022 ms
 7  atm0-0-0.sr1.nyc1.alter.net (134.24.32.50)  21.252 ms  23.044 ms  21.642 ms
 8  105.ATM2-0-0.XR2.NYC1.ALTER.NET (146.188.177.158)  21.338 ms  20.904 ms  20.39 ms
 9  194.ATM9-0-0.GW1.NYC5.ALTER.NET (146.188.177.229)  120.15 ms  23.331 ms  22.024 ms
10  421.ATM9-0-0.BR1.NYC5.Alter.Net (137.39.30.118)  21.395 ms  22.958 ms  20.692 ms
11  225.ATM1-0-0.CR1.AMS2.Alter.Net (146.188.5.10)  104.605 ms  108.978 ms  105.051 ms
12  213.Hssi1-0.CR1.STK2.Alter.Net (146.188.2.194)  127.12 ms  128.184 ms  128.107 ms
13  Fddi0-0.GW1.STK2.Alter.Net (146.188.31.99)  143.882 ms  141.473 ms  139.756 ms
14  Taide-gw.customer.ALTER.NET (146.188.48.10)  150.409 ms  144.739 ms  142.156 ms
15  SE-STO-MFS-0.taide.net (193.219.194.145)  145.877 ms  164.948 ms  144.165 ms
16  NO-NIT-TN-0.taide.net (193.219.193.93)  164.505 ms  162.902 ms  159.266 ms
17  RO-BUC-PUB-1.taide.net (193.219.193.170)  746.259 ms  720.339 ms  791.329 ms
18  waiting.pub.ro (141.85.130.16)  706.019 ms  696.28 ms  753.865 ms
19  pub.pub.ro (141.85.128.1)  730.402 ms  718.742 ms  725.471 ms
```

Se văd cei 19 paşi pe care trebuie să-i facă un pachet plecînd de la calculatorul de pe biroul meu pînă la un calculator (`pub.pub.ro`) din Politehnica din Bucureşti. Numele calculatoarelor sunt afişate atît în formă textuală cît şi în formă IP. Pentru fiecare destinaţie sunt trimise 3 pachete, şi timpii de propagare dus-întors sunt afişaţi.

Putem afla multe informaţii interesante din acest tabel. Cea mai mare diferenţă de timp este între paşii 15 şi 16: pînă la 15 pot ajunge în 150ms, dar pînă la 16 în 750ms. Asta pentru că legătura între 16 şi 17 se face printr-un satelit geosincron, care zboară la o altitudine de cam 36,000km. Semnalului electromagnetic îi trebuie cam jumătate de secundă pentru a traversa distanţa asta în sus şi în jos.

Ce e amuzant este că ambele capete ale legăturii prin satelit se află în Europa. Prefixul "NO" din nume mă îndeamnă să ghicesc că legătura este între Norvegia şi România. Zic "amuzant" pentru că a doua diferenţă ca mărime este între 10 şi 11. Legătura asta este transcontinentală, între New York şi Amsterdam, dar întîrzierea de propagare este de numai 80ms! Asta pentru că foloseşte un cablu submarin, probabil de fibră optică.

Putem afla şi care sunt companiile ale căror reţele sunt tranzitate. În ordine pachetul trece prin:

- `NET.CMU.EDU`: reţeaua universităţii mele, Carnegie Mellon;
- `psc.net`: Pittsburgh Supercomputing Center;
- `cerf.net`;
- `ALTER.NET`: o reţea a firmei UUNET Technologies;
- `taide.net`: o reţea a ţărilor baltice;

- `pub.ro`: reţeaua politehnicii din Bucureşti.

Am aflat informaţii despre aceste reţele folosind `whois`, descrisă mai sus, interogînd baza de date de la `internic.net`

```
# whois -h rs.internic.net taide.net
Upnet Baltic Taide Network AB (TAIDE-DOM)
   Radiskis Farm
   Vidiskiai, 4126
   LT

   Domain Name: TAIDE.NET

   Administrative Contact:
      Bjerkeland, Vidar  (VB50)  vidar.bjerkeland@ARCTICMAIL.COM
      +47 9002 74 66 (FAX) +47 9027 74 66
   Technical Contact, Zone Contact:
      Vosylius, Saulius  (SV104)  saulius@TAIDE.LT
      +370-2-223541
   Billing Contact:
      Bjerkeland, Vidar  (VB50)  vidar.bjerkeland@ARCTICMAIL.COM
      +47 9002 74 66 (FAX) +47 9027 74 66

   Record last updated on 04-Mar-98.
   Record created on 24-Aug-95.
   Database last updated on 12-Apr-98 03:37:05 EDT.

   Domain servers in listed order:

   PARABOL.TAIDE.NET            193.219.197.130
   MIDUS.TAIDE.NET              193.219.195.130


The InterNIC Registration Services Host contains ONLY non-military Domains and
POC's.  Other associated whois servers:
  American Registry for Internet Numbers whois server - whois.arin.net
  European IP Address Allocations whois server - whois.ripe.net
  Asia Pacific IP Address Allocations whois server - whois.apnic.net
  US Military whois server - whois.nic.mil
```

Putem afla şi alte lucruri interesante din numele ruterelor tranzitate, cum ar fi tipul reţelei locale: unde apare "ATM" este vorba de o reţea cu protocoale ATM, unde apare "Hssi" e o reţea de tip "High Speed Serial Interface", unde apare "Fddi" e o reţea de tip "Fiber Distributed Data Interface", o reţea cu fibra optică. Ş.a.m.d.

16 Alte probleme

Voi încheia aici această discuţie despre Internet şi protocoalele de rutare. În clipa asta am o listă mai mare de lucruri pe care aş fi dorit să le spun decît de lucruri pe care am reuşit să le povestesc.

O să menţionez în treacăt cîteva probleme fascinante ale rutării, care influenţează enorm arhitectura şi protocoalele reţelei, dar despre care va trebui poate să revin altădată.

16.1 Stabilitate

Discuţiile de mai sus presupun adesea că reţeaua funcţionează impecabil. Ce ne facem cînd avem de-a face cu rutere defecte, sau mai rău, cu atacuri informatice.

De exemplu, în primele zile ale ARPAnet-ului, un ruter a avut o cădere de memorie, care a şters parţial tabelele de rutare, dar nu l-a scos complet din funcţiune. Ruterul a ajuns la concluzia că se află la distanţă 0 de oricare alt ruter din Internet. Cînd a schimbat informaţii cu vecinii săi, aceşti s-au bucurat că au găsit o ruta mai rapidă pentru pachetele lor şi au început să-i dea toate pachetele acestui ruter. Apoi vestea cea bună s-a dus mai departe. În cîteva ore toate pachetele din Internet veneau direct la ruterul defect. Asta a produs o congestie enormă, care a oprit literalmente reţeaua.

Lecţia este: nu poţi construi un sistem cu 30 de milioane de bucăţi dacă nici una nu se poate strica. De atunci protocoalele de rutare au devenit din ce în ce mai robuste, fiind capabile să tolereze malfuncţii şi configurări greşite în părţi ale reţelei. În plus ruterele sunt configurate să se autentifice reciproc (folosind parole) şi să nu primească informaţii de rutare decît de la calculatoare de încredere. "Încredere" înseamnă practic sub administraţia unei organizaţii care poate fi trasă la răspundere pentru erori, şi eventual pedepsită cu deconectarea totală.

Un al doilea tip de stabilitate este următorul: dacă un ruter reboot-ează sau o nouă legătură este introdusă sau scoasă din reţea, după un timp finit ajung toate ruterele de acord asupra topologiei? Deşi poate părea evident că răspunsul este "da", lucrurile nu stau chiar aşa; multe protocoale din Internet de fapt au mari probleme să incorporeze astfel de tranziţii.

Un al treilea tip de problemă este cauzat de faptul că informaţiile despre modificări se propagă în timp, aşa că anumite părţi ale reţelei funcţionează cu idei despre o topologie care nu mai există! Se pot astfel dezvolta calamităţi cum ar fi bucle de rutare, în care un pachet se plimbă în cerc între două (sau mai multe rutere), pentru că fiecare din ele crede că celălalt este mai aproape de destinaţie. Astfel de pachete sunt distruse într-o "gaură neagră" după ce le expiră timpul de viaţă. Inutil de spus că astfel de probleme sunt adevărate coşmare pentru administratorii de reţea (care le pot crea atunci cînd introduc rute greşite), şi sunt extrem de greu de depistat şi depanat.

În general, reţeaua poate dezvolta tot felul de anomalii, ca în cazul în care protocoale de rutare au ales o rută între două oraşe americane aflate la 100km unul de altul prin Japonia, pentru că un segment de reţea pe acolo era mai "ieftin". La ora actuală o mare parte din cercetarea Internet-ului este de natură empirică: dai drumul la nişte pachete şi după aia te miri ce se întîmplă cu ele şi încerci să explici. Să observăm că nu există nici o garanţie că două pachete ale aceleiaşi transmisiuni vor merge pe acelaşi drum! Sistemul este mult prea mare.

16.2 Multicast

O problemă absolut fascinantă care a fost rezolvată cu succes este transmisiunea spre multiple destinaţii simultan. Anumite rutere, care vorbesc un protocol special, sunt grupate într-o "reţea virtuală” numită MBONE (Multicast Bone), pe care ceea ce un calculator trimite primesc toate cele interesate. Am văzut astfel în direct pe monitorul meu aterizarea navetei spaţiale, transmisă în direct de la NASA. Un fel de televiziune prin cablu, dacă vreţi, dar cu potenţiali milioane de receptori oriunde în lume şi a cărei topologie se schimbă permanent.

16.3 Rutare mobilă

O altă problemă colosală a fost rezolvată: cum poţi comunica cu un calculator care umblă tot timpul? Presupunerile noastre despre faptul că toate calculatoarele cu adrese în aceeaşi reţea sunt la un loc nu mai sunt valabile. Scheme deosebit de ingenioase permit acum unui individ să călătorească oriunde în lume şi să folosească permanent Internet-ul. Schema este analoagă telefoniei celulare: calculatorul mobil îşi schimbă tot timpul ruterul implicit, dar detaliile sunt mult mai complicate.