

NFS și NIS

Traducerea realizată de
Tubultoc Ana-Roxana, anul 3, grupa 1B

Sistemul de fișiere al rețelei (NFS) este cunoscut ca fiind dificil de setat și instalat. În realitate, NFS este destul de ușor de implementat pe sistemele Linux, iar dacă mașina ta este una dintre puținele sisteme Linux dintr-o rețea locală (LAN), NFS îți poate oferi o flexibilitate enormă.

În cadrul acestui capitol se vor prezenta avantajele sistemului de fișiere al rețelei (NFS), precum și modul în care sistemul tău poate fi setat în așa fel încât să se comporte atât ca server cât și client pentru celelalte sisteme din rețeaua ta locală. Dacă calculatorul tău rulează sub Linux, în regim independent, NFS nu va avea nici un rol semnificativ pentru tine (decât unul academic). Însă dacă faci parte dintr-o rețea locală(LAN), indiferent de sistemele ce o compun (PC- uri, Mac-uri, Unixuri sau Linux) ar trebui , cel puțin, să afli ce-ți poate oferi NFS-ul.

Cea de a doua parte a acestui capitol face referire la Serviciul de Informație al Rețelei(NIS), o versiune mai veche numită cândva Pagini Aurii(Yellow Pages) și la modul de funcționare la nivel de rețea. Deși probabil nu vei avea nevoie de NIS decât în cadrul unei rețele mari, poți afla cum funcționează acest sistem. Capitolul se referă, de asemenea, și la câteva instrumente de administrare a sistemului pentru administrarea NFS (Sistemul de Fișiere al Rețelei), NIS (Serviciul de Informație al Rețelei) și RPC(Remote Procedure Call =Apelul Procedurilor la Distanță).

Ce este NFS ?

NFS a fost proiectat pentru a rezolva o problemă obișnuită din cadrul unei rețele Unix. Tendința generală cunoscută este cea a proceselor distribuite și a rețelelor client-server. Cu toate acestea, mulți utilizatori au, de fapt, mașini puternice care comunică cu un server. Există aplicații de care au nevoie utilizatorii și care sunt localizate în alte locuri decât pe desktop, și de aceea apare necesitatea unei metode de acces a fișierelor la distanță. Deși servicii precum Telnet-ul permit utilizatorului accesul unor mașini la distanță, acestea nu țin cont de procesorul mașinii, transferând la distanță. Un alt aspect important al schimbului între mașini distribuite este partajarea periferică și nevoia de a furniza acces pentru mulți utilizatori la anumite

resurse. Pentru a facilita integrarea stațiilor într-o rețea locală și pentru a simplifica accesul fișierelor la distanță și partajarea periferică, Sun Microsystems a introdus Sistemul Fișierelor de Rețea (Network File System, NFS). Acesta are la bază un sistem numit RPC (Remote Procedure Calls- Apelul Procedurilor la Distanță).

Cei de la Sun au proiectat NFS astfel încât să permită mașinilor de diferite tipuri să funcționeze la fel, indiferent de sistemul lor de operare. Prin publicarea specificațiilor NFS, Sun a permis celorlalți producători să-și modifice sistemele pentru a fi compatibile cu NFS , rezultând astfel o rețea mult mai omogenă. NFS este acum un standard al mediilor Unix, având un suport puternic în alte sisteme de operare.

NFS se referă, de fapt, la două lucruri diferite: un produs și un protocol. Produsul NFS este un set de protocoale pentru diverse sarcini. Protocolul NFS este unicul protocol din cadrul produsului NFS ce se ocupă cu accesul fișierelor. În prezent, NFS este în strânsă legătură cu sistemul Unix și protocolul TCP/IP. Pentru celelalte sisteme de operare (precum Novell NetWare), NFS este o extensie care este adăugată de către administratorul de sistem. Linux (și majoritatea versiunilor Unix) utilizează procesul NFSD pentru a administra accesul de tip NFS.

NFS-ul permite unei aplicații citirea și scrierea fișierelor aflate pe servere NFS. Accesul la serverul NFS este transparent atât aplicației cât și utilizatorului. Accesul transparent la structura de fișiere a altei mașini este obținut prin legarea logică a serverului NFS la client. Sistemul de fișiere al serverului NFS se poate monta în întregime sau în parte. Montarea este realizată în același fel ca orice montare a sistemului de fișiere(vezi Capitolul 18, "Sistemul de fișiere și Discul", pentru informații referitoare la comanda *mount*), deși comanda are un parametru ce indică folosirea NFS-ului. De exemplu, pentru a monta directorul */usr/database/data* de pe o mașină la distanță (numită *wizard*) în directorul tău */usr/data* vei iniția următoarea comandă:

```
mount -t nfs wizard :/usr/database/data /usr/data
```

La inițierea comenzii , mașina locală verifică dreptul de acces la acest director de pe mașina aflată la distanță. Dacă totul e în ordine, mașina de la distanță trimite un identificator de fișier ce va fi utilizat pentru a redirecționa toate cererile de pe mașina locală, referitoare la acel director. De fiecare dată când utilizatorul unui director montat prin NFS inițiază o cerere, un proces daemon numit NFSD se ocupă de transferuri.

NFS utilizează termenul de „client” pentru orice mașină care cere un fișier de pe o altă mașină, care este serverul. Sistemele de operare multitasking pot fi în același timp și client și server. De obicei, se impun restricții asupra fișierelor sau asupra părților unui sistem de fișiere partajabile, atât pentru securitate cât și din considerente de viteză.

O instalare NFS tipică utilizează PC-urile sau stațiile de lucru care nu necesită bootare de pe disc ca fiind clienți ce accesează un sistem server puternic. (Întrucât sistemele de operare instalate pe PC-uri precum MS-DOS sunt monoutilizator (single-tasking), PC-urile se comportă în mod uzual doar ca clienți, cu excepția cazului în care se lucrează pe sisteme de operare multiutilizator (multitasking), precum Windows NT, Windows 95, sau OS/2). Pentru rețelele Linux, pot exista câteva sisteme Linux care își partajează drivere cu alte mașini din cadrul rețelei. E posibil să avem o rețea întreagă de sisteme multiutilizator care își partajează driverele între ele, deși în practică acest lucru este posibil numai în cadrul unei rețele mai mici.

Viteza de transfer în cadrul rețelei devine importantă, datorită cerinței de transferare rapidă a fișierelor cu NFS. Când a fost proiectat, scopul inițial al unui sistem de fișiere montat prin NFS a fost acela de a asigura o performanță echivalentă cu 80% din cea a unui hard disk montat local. Acest scop determină o creștere a performanței atât la nivelul driverului de disc NFS, cât și la nivelul rețelei. În mod obișnuit, driverele de disc NFS de pe un anumit server sunt printre cele mai rapid disponibile, în scopul de a reduce ”gâtuiturile” la celalalt capăt. În practică, pentru majoritatea rețelelor, sistemul NFS utilizează un echipament standard, ceea ce nu constituie, de fapt, o problemă pentru împărțirea unor directoare în cadrul unei rețele mici.

NFS oferă o serie de avantaje în cadrul unei rețele Linux. În primul rând, permite o păstrare a informațiilor și a aplicațiilor mari pe un singur disc al rețelei, la care au acces toate mașinile (deci are loc o economisire a spațiului de disc, pe care ar impune-o copiile independente). Din punct de vedere administrativ, NFS oferă posibilitatea păstrării aplicațiilor într-o singură locație (ba chiar și plasarea tuturor directoarelor utilizatorilor pe o singură mașină) pentru o mai ușoară actualizare, copiere și organizare.

Versiunea Linux a FTP-ului diferă într-o anumită măsură de versiunea standard a Unix-ului, prin faptul că multe din proprietățile sistemului NFS sunt regăsite în codul nivelului nucleu al Sistemului de Fișiere Virtuale (Virtual File System -VFS). Versiuni mai vechi de Linux aveau anumite probleme cu versiunea FTP datorită dimensiunii maxime a datagramelor TCP, care trebuia redusă pentru a funcționa corect. Aceasta a avut un efect drastic de încetinire a performanțelor.

Deoarece NFS-ul se bazează pe Linux, nivelul de securitate oferit este destul de rudimentar. Acesta este motivul pentru care Sun a introdus Secure NFS (NFS Sigur), care se bazează pe un protocol de mesaje criptat pentru o protecție mai bună împotriva unui acces nepermis asupra sistemului de fișiere montat NFS. Această versiune nu este încă disponibilă în cadrul unei implementări Linux.

Instalarea NFS-ului

Primul pas în instalarea NFS-ului pe un sistem Linux este acela de a ne asigura că suportul NFS este compilat în cadrul nucleului. Cele mai noi versiuni de Linux au această caracteristică în mod implicit, dar dacă se lucrează pe o versiune mai veche este necesară o verificare a codului NFS. Versiunile de Linux de după 1.1 pot să confirme faptul că suportă NFS-ul prin examinarea fișierului `/proc/filesystems`. În cadrul acestui fișier trebuie să existe o linie în care `nfs` apare alături de comanda `nodev`. Un fragment din acest fișier este dat mai jos:

`minix`

`ext2`

`umsdos`

`msods`

`nodev proc`

`nodev nfs`

`iso9660`

Penultima linie indică includerea codului NFS în cadrul nucleului. În cazul în care codul NFS nu este inclus, va trebui reconstruit nucleul astfel încât să includă și driverele NFS.

Versiunile Linux mai vechi decât 1.1 prezintă o dificultate mai mare în verificarea codului NFS. Cea mai bună metodă pentru realizarea verificării o constituie montarea unui director NFS. În caz de eșec, se admite lipsa codului NFS (în ipoteza în care comanda `mount` este, bineînțeles, corectă). Pentru o verificare rapidă, se poate monta un director local pe

mașina proprie (acest lucru este posibil, pentru toate versiunile NFS, deși uneori poate deveni derutant). Pentru realizarea acestei verificări, se creează un subdirector nou, după care se lansează comanda mount cu un director existent. De exemplu, prin următoarele comenzi se încearcă montarea directorului */usr* în cadrul subdirectorului gol */tmp/nfstest* :

```
mkdir /tmp/nfstest
```

```
mount localhost:/usr /tmp/nfstest
```

Dacă lansarea comenzii se încheie cu succes (se poate merge în directorul */tmp/nfstest*, unde se poate verifica dacă există aceeași listă de fișiere ca și în */usr*), atunci înseamnă că nucleul are codul NFS încorporat. Dacă însă apare un mesaj de eroare similar celui de mai jos:

“fs type nfs not supported” (fișier sistem de tip nfs nesuportat)

atunci codul NFS lipsește și va trebui să reconfigurezi un nou nucleu cu driverele NFS adăugate.

NOTĂ : La verificarea acestui cod NFS pot să apară multe mesaje de eroare. Însă singurul mesaj care contează este cel de tipul „nfs not supported”. Restul mesajelor sunt legate de lipsa configurării NFS-ului.

Procesele care rulează în fundal (numite daemon) ale NFS-ului trebuie să fie setate în sistem. Dacă se dorește un statut de server NFS (permiterea ca directoarele proprii să poată fi montate de către alții), este necesară atât o instalare a *nfsd*-ului cât și a proceselor daemon. Aceste procese daemon (de fundal) își încep activitatea în momentul pornirii calculatorului, prin citirea fișierelor de tip *rc*. Aceste procese necesită programul *rpc.portmap* pentru a funcționa, deoarece amândouă se înregistrează cu utilitatea *portmappper*.

Comenzile de start pentru procesele de fundal se găsesc, de obicei, în fișierul */etc/rc.d/rc.inet2* (sau acolo unde s-a realizat instalarea fișierelor *rc*). Ultimele versiuni de Linux vor avea o secțiune specială în fișierul */etc/rc.d/rc.inet2*. De exemplu, secțiunea referitoare la NFS va arăta în felul următor:

```

# # Start the various SUN RPC servers. ( Pornirea diverselor servere SUN
RPC. )
if [ -f ${NET}/rpc.portmap ]; then

# Start the NFS server daemons. ( Pornirea proceselor server NFS. )

if [ -f ${NET}/rpc.mountd ]; then

echo -n " mountd"

${NET}/rpc.mountd

fi

if [ -f ${NET}/rpc.nfsd ]; then

echo -n " nfsd"

${NET}/rpc.nfsd

fi
# # Fire up the PC-NFS daemon(s). ( Pornește procesele daemon PC-NFS.)

# if [ -f ${NET}/rpc.pcnfsd ]; then

# echo -n " pcnfsd"

# ${NET}/rpc.pcnfsd ${LPSPOOL}

# fi

# if [ -f ${NET}/rpc.bwnfsd ]; then

# echo -n " bwnfsd"

# ${NET}/rpc.bwnfsd ${LPSPOOL}

# fi

```

fi # Done starting various SUN RPC servers. (Am terminat pornirea diferitelor servere SUN RPC.)

Dacă fișierul `inet2` nu are nici o linie similară cu vreuna din cele de mai sus , atunci se va încerca căutarea unei locații prin comanda de pornire `rpc.portmapper`. Secțiunea de pornire a portmapperului va arăta similar cu următoarea:

```
# Start the SUN RPC Portmapper. ( Pornirea portmapperului SUN RPC.)
```

```
if [ -f ${NET}/rpc.portmap ]; then
```

```
    echo -n " portmap"
```

```
    ${NET}/rpc.portmap
```

```
fi
```

Dedesubtul acestor linii, introduceți următoarele comenzi pentru inițierea proceselor `rpcd` și `mountd` (procese daemon):

```
if [ -x /usr/sbin/rpc.mountd ]; then
```

```
    echo -n " mountd"
```

```
    /usr/sbin/rpc.mountd
```

```
fi
```

```
if [ -x /usr/sbin/rpc.nfsd ]; then
```

```
    echo -n " nfsd"
```

```
    /usr/sbin/rpc.nfsd
```

```
fi
```

Dacă procesele `rpc.nfsd` și `rpc.mountd` nu se află în `/usr/sbin`, introduceți corect fiecare cale. Aceste linii nu folosesc o cale deja stabilită anterior. Calea trebuie specificată în mod explicit procesului daemon.

Următorul pas în configurarea sistemului pentru rolul de server NFS îl constituie stabilirea unei liste a tuturor clienților posibili care se pot atașa la sistemul nostru pentru montarea de directoare. Acest lucru se poate realiza prin intermediul fișierului */etc/exports*. Acesta este citit de fiecare dată când procesul daemon *mount* primește o cerere de montare a unui director. Fișierul conține o listă a directoarelor pe care le permitem a fi montate, precum și a sistemelor de la distanță care le pot monta, însoțite de o indicație de permisiune.

Cea mai bună metodă de a explica fișierul */etc/exports* este aceea de a urmări un exemplu. În următorul fișier sunt indicate câteva sisteme cărora le este permisă montarea directoarelor pe mașina locală:

```
# /etc/exports for merlin (se montează /etc/exports pe merlin)
```

```
/usr/database/data chatton(rw) big_roy (rw) wizard (rw)
```

```
/usr/book chatton(rw) wizard (ro)
```

```
/usr/bin/bigapp big_roy(rw) wizard (ro)
```

```
/usr/ftp (ro)
```

În fișier se arată că cele trei mașini *chatton*, *big_roy* și *wizard* pot monta directorul local */usr/database/data* în modul citire-scriere (adică pot modifica chiar conținutul). Directorul */usr/book* poate fi montat în mod citire-scriere de către mașina *chatton*, aflată la distanță, și doar în mod citire (nu este permisă scrierea) de către *wizard*. Directorul */usr/ftp* poate fi montat doar în modul citire de către orice mașină se dorește.

Când se specifică numele mașinilor în fișierul */etc/exports* , se pot utiliza nume explicite sau combinații de caractere speciale, precum asterisc sau semnul întrebării, în scopul potrivirii pe mai multe mașini. De exemplu, următoarea intrare

```
/usr/tim/book big_*(rw)
```

permite oricărei mașini al cărei nume începe cu „big_” să monteze directorul în mod citire-scriere. Când nu se specifică nici un nume de gazdă

(ca în cazul directorului */usr/ftp* din exemplul precedent), atunci orice mașină poate monta directorul.

Fișierul */etc/exports* oferă o listă a permisiunilor posibile pentru o mașină aflată la distanță care dorește montarea unui director local. Deși majoritatea sistemelor utilizează doar *rw* și *ro* (pentru citire-scriere și respectiv doar citire), uneori sunt necesare mai multe drepturi. Iată o listă de câteva permisiuni valide:

- *insecure* Permite acces fără identificare asupra mașinii în cauză (suprascrie cererile de identificare).
- *kerberos* Forțează identificarea în mod Kerberos de la distanță (care nu este implementat pentru NFS-ul Linux).
- *link_absolute* Păstrează legăturile simbolice așa cum erau.
- *link_relative* Permite o conversie a legăturilor simbolice absolute în legături relative prin atașarea, în cazul în care este necesar, a lui *../*
- *root_squash* Prin care se interzice utilizatorilor de tip *root* aflați la distanță, dreptul de acces de tip *root* pe mașina locală.
- *secure-rpc* Prin care se forțează identificarea RPC de la distanță (aceasta este în mod implicit, deși nu se regăsește implementată în majoritatea versiunilor NFS).

Dar, cum reușește NFS să lucreze cu fișiere și cu drepturile de acces în contextul montării ? În momentul în care daemonul NFS se ocupă cu transferul de fișiere sau cereri, el va trece și identificatorul utilizatorului, precum și al grupului din care face parte acel utilizator. În cazul în care clientul și serverul au același identificator de utilizator și același identificator de grup (adică împart același spațiu de *uid/gid*), nu există nici o problemă referitoare la drepturi. Când însă UID-ul și GID-ul nu coincid, daemonul se ocupă de transferul dintre ei.

Montarea directoarelor NFS

O dată NFS-ul configurat, el poate fi utilizat pentru montarea directoarelor de la distanță în cadrul propriului sistem local de fișiere. Iar acest lucru se realizează prin intermediul comenzii `mount`. Forma generală a ei, în contextul NFS este următoarea:

```
mount -t nfs director_la_distanță director_local [-o opțiuni]
```

unde `director_la_distanță` este numele mașinii la distanță și a directorului ce urmează a fi montat în manieră NFS, iar `director_local` este locația în care urmează a fi montat directorul aflat la distanță. Parametrul opțiuni poate fi reprezentat de orice flag specific NFS-ului. Parametrul `director_la_distanță` este specificat întotdeauna în formatul:

```
nume_mașină_la_distanță : director_la_distanță,
```

ca în exemplul `wizard:/usr/lib`. Mulți administratori renunță la componenta `-t nfs` a comenzii, întrucât acest format este stabilit în mod unic pentru NFS. Alții consideră, însă că este mai bine să se păstreze opțiunea `-t`, în scopul de a se aminti faptul că mașina la distanță este montată NFS.

Deși există mai multe opțiuni posibile pentru comanda *mount*, în mod NFS, numai câteva sunt utilizate în situațiile reale. În continuare sunt prezentate o serie de opțiuni valabile :

- `hard` Această opțiune stabilește că directorul va fi montat hard. Aceasta este o acțiune implicită.
- `intr` Această opțiune permite întreruperea unei cereri NFS.
- `rsize` Opțiune prin care se poate stabili dimensiunea unei datagrame, utilizate în cadrul unor cereri de citire (dimensiunea implicită este de 1024 de octeți).
- `soft` Specifică o montare soft a directorului (în locul unei montări hard).
- `timeo` Opțiune care specifică timpul rămas până la terminarea unei cereri NFS, în zecimi de secundă (valoarea implicită este de 7/10).

- *wsize* Aceasta specifică dimensiunea unei datagrame utilizată pentru o cerere de scriere (în mod implicit, valoarea ei este de 1024 de octeți).

Opțiunile *rsize*, *timeo* și *wsize* sunt urmate de un semn egal și de valoarea care le este asignată. Opțiunile *rsize* și *wsize* sunt folosite, în primul rând, pentru a schimba dimensiunea datagramei de pe mașina aflată la distanță (în cazul în care se utilizează o dimensiune mai mare decât cea cu care lucrează sistemul Linux). Toate opțiunile NFS trebuie să urmeze tiparul dat de parametrul *-o* din linia de comandă, dacă după el sunt fixate anumite opțiuni. De exemplu, pentru a seta timpul rămas la 2 secunde, în cadrul unei comenzi de montare la distanță a unui director și pentru a permite, în plus, întreruperile, se poate iniția următoarea comandă:

```
mount -t nfs wizard:/usr/data /usr/data -o timeo=20,intr
```

Ca o alternativă, dacă nu se dorește specificarea în linia de comandă a opțiunilor pentru directoarele care sunt mai des montate, se poate utiliza fișierul */etc/fstab* pentru a le furniza. Aceeași comandă ca mai sus poate fi plasată în fișierul */etc/fstab* astfel:

```
wizard:/usr/data /usr/data nfs timeo=20,intr
```

Când se folosește fișierul */etc/fstab* pentru precizarea opțiunilor și a punctelor de montare, se poate monta conținutul de la distanță mult mai ușor, inițiind comanda:

```
mount wizard:/usr/data
```

Comanda *mount* verifică fișierul */etc/fstab* pentru determinarea punctului de montare și a opțiunilor utilizate și în același timp recunoaște comanda drept comandă *mount NFS*. În cazul directoarelor aflate la distanță, care vor fi mai des folosite , este recomandabilă această formă decât scrierea ei întregă, de fiecare dată.

Două dintre opțiunile de montare NFS făceau referire la o montare soft sau hard. În mod implicit, se consideră că montarea este de tip hard. Aceasta înseamnă că, dacă NFS-ul nu este în stare să monteze un anumit director, la terminarea acestei încercări se va genera un mesaj de eroare , după care va încerca din nou, dublând timpul rămas până la terminarea

comenzii. Și astfel se reia procedura până când directorul aflat la distanță este montat (generându-se câte un mesaj de eroare de fiecare dată când expiră timpul alocat). O astfel de montare a unor directoare care se reia până la finalizarea cu succes, poartă denumirea de montare hard („hard mount”). O montare soft este cea care se comportă similar, însă care generează mesaje de eroare numai după o perioadă mai mare de timp, adică la fiecare 60 de secunde. Mesajele de eroare nu sunt afișate, (din moment ce se găsesc în coada I/O), însă se poate câștiga control asupra sistemului mult mai ușor prin intermediul unei montări soft după un timp mai mare.

Administrarea NFS-ului

NFS (și RPC, de care depinde NFS) are două instrumente principale de administrare disponibile pentru furnizarea actualizărilor stărilor și a indicațiilor pentru problemele ce pot surveni în cadrul sistemului. Rularea unui singur instrument din acesta nu este suficientă pentru rezolvarea problemei. Deseori se poate întâmpla ca un instrument să semnaleze o problemă la un anumit port, însă la o analiză mai atentă se poate descoperi că acel port funcționează și că procesul aflat la celălalt capăt și-a încetat activitatea. De aceea, aceste instrumente au fost proiectate în scopul de a fi complementare până când se poate stabili cu exactitate o cauză a problemei survenite.

rpcinfo

Programul *rpcinfo* monitorizează mapperul portului mașinei pe care rulează și mapperele porturilor serverelor de la nivelul întregii rețele. Deoarece mapperul portului este programul care controlează accesul la RPC, acest tip de informație prezintă o importanță deosebită în urmărirea problemelor. Programul *rpcinfo* poate afișa conținutul tabelor de mapare, arătând portul și numerele programelor corespunzătoare fiecărei conexiuni și este capabil să activeze servere la distanță pentru testarea unei conexiuni.

În general, *rpcinfo* este apelat cu opțiunea `-p` în vederea afișării programelor RPC ce sunt urmărite în mod curent de către mapperul de port. Opțional, se poate adăuga numele unei mașini pentru a afișa eventualele conexiuni doar cu o singură mașină. Un rezultat tipic al programului *rpcinfo* este arătat mai jos:

```
$ rpcinfo -p
```

```
program vers proto port
```

100000 2 tcp 111 portmapper
100000 2 udp 111 portmapper
100008 1 udp 1026 walld
150001 1 udp 1027 pcnfsd
150001 2 udp 1027 pcnfsd
100002 1 udp 1028 rusersd
100002 2 udp 1028 rusersd
100024 1 udp 1029 status
100024 1 tcp 1024 status
100020 1 udp 1034 llockmgr
100020 1 tcp 1025 llockmgr
100021 2 tcp 1026 nlockmgr
100021 1 tcp 1027 nlockmgr
100021 1 udp 1038 nlockmgr
100021 3 tcp 1028 nlockmgr
100021 3 udp 1039 nlockmgr

Dacă apare vreo problemă în timpul contactării mapperului de port, *rpcinfo* va returna un mesaj de eroare. Într-un astfel de caz, mapperul de port nu funcționează corect și apare riscul imposibilității contactării altor mașini. O modalitate de verificare este folosirea comenzii ping. Un exemplu al unui astfel de mesaj de eroare de tipul „fatal error” este dat mai jos:

```
$ rpcinfo -p
```

rpcinfo: can't contact port mapper (mapperul de port nu poate fi contactat)

RFC: Remote system error -125

Anumite conexiuni pot fi testate cu *rpcinfo* utilizând numele mașinii și al procesului, așa după cum se poate vedea în exemplul următor:

```
$ rpcinfo -u merlin walld
```

```
program 100008 version 1 is ready and waiting  
( programul 100008, versiunea 1 este pregătit și se află în stare de așteptare)
```

Observați că opțiunea *-u* este utilizată în cazul conexiunilor UDP, în timp ce opțiunea *-t* trebuie să fie utilizată în cazul celor TCP. În acest exemplu, clientul *rpcinfo* a trimis o cerere către programul specificat și așteaptă o replică. Un răspuns cu succes rezultă din mesajul arătat mai sus. Dacă nu se primește un răspuns până la expirarea unui anumit timp, se afișează un mesaj de eroare.

În exemplul de ieșire arătat mai sus, există un proces numit *pcnfsd*, care este un server RPC dezvoltat pentru utilizarea în cadrul mașinilor DOS. Se ocupă cu drepturile de acces, precum și cu rularea serviciilor pentru partea de DOS; în același timp simplifică accesul mașinii DOS la serviciile NFS.

nfsstat

Programul *nfsstat*, așa după cum și numele ne sugerează, furnizează statistici legate de numărul și tipul cererilor RPC ce sunt făcute. Deși această comandă se apelează fără opțiuni, există totuși o serie de opțiuni (care depind de implementare și de versiune) care indică statistici sau exemplifică numai anumite părți ale unei conexiuni. Programul *nfsstat* nu face parte din majoritatea distribuțiilor Linux, însă poate fi găsit pe câteva siteuri Linux FTP și BBS și ca parte a unor pachete utilitare a administrării sistem. Pentru o rețea mai mică, rezultatul comenzii *nfsstat* este arătat mai jos:

```
Server rpc:  
calls badcalls nullrecv badlen xdrcall  
10465 0 0 0 0
```

Server nfs:
calls badcalls
10432 0
null getattr setattr root lookup readlink read

1 0% 24 0% 1 0% 0 0% 10123 0% 0 0% 5 0%

wrcache write create remove rename link symlink

0 0% 2 0% 0 0% 1 0% 0 0% 1 0% 0 0%

Client rpc:
calls badcalls retrans badxid timeout wait newcred
8273 2 0 0 0 0 0

Client nfs:
calls badcalls
8263 0

null getattr setattr root lookup readlink read

1 0% 24 0% 1 0% 0 0% 10123 0% 0 0% 5 0%

wrcache write create remove rename link symlink

0 0% 2 0% 0 0% 1 0% 0 0% 1 0% 0 0%

Rezultatul este util pentru depistarea problemelor legate de conexiune. Numărul care apare ca *badcalls* ne arată numărul de mesaje RPC defecte procesate de sistem. Numerele pentru *nullrecv* și *badlen* ne indică numărul de mesaje goale sau incomplete. *Xdr call* ne arată numărul erorilor ce au survenit în interceptarea mesajelor.

Privitor la client, *badxid* indică numărul mesajelor primite care nu se potrivesc cu o cerere trimisă (se bazează pe un număr de identificare). Timpul rămas și *retrans* ne arată de câte ori a trebuit să fie retrimis un mesaj. Dacă aceste numere sunt mari, în general va însemna că acea conexiune este prea înceată sau că este o greșeală la nivel de UDP. Numărul *wait* ne indică de câte ori a fost amânat un proces din cauza lipsei de porturi disponibile.

Aceste date statistice sunt utile pentru o configurare adecvată a RPC-ului. Administratorii de sistem pot modifica valorile pentru sistemul NFS și pot, de asemenea, urmări efectul lor asupra desfășurării de-a lungul timpului.

Ce sunt NIS și YP ?

Protocolul Paginile Aurii(Yellow Pages) este un serviciu RPC (așa cum este și NFS) care oferă un director de servicii. În conformitate cu cererile dreptului de autor, Yellow Pages a fost redenumit Network Information Service (Serviciul de Informații al Rețelei- NIS), deși ambii termeni sunt utilizați în mod curent și semnifică aproape același lucru. YP a fost dezvoltat cu anumite scopuri, însă cel care vizează în mod deosebit utilizatorii este cel legat de drepturile de acces.

Dacă ești un utilizator în cadrul unei rețele mari și te conectezi la alte mașini (prin Telnet sau FTP, de exemplu), trebuie să menții o situație a tuturor mașinilor la care te conectezi. Vei avea astfel nevoie de conturile utilizatorilor tuturor mașinilor pe care dorești să le accesezi. Este deci dificilă menținerea de parole pe un număr mare de mașini, deoarece este necesar să te înregistrezi la fiecare pentru a efectua schimbările de parolă. Yellow Pages a fost dezvoltat pentru a permite existența unui singur fișier central de parole, care va fi partajat pe tot cuprinsul rețelei.

NIS este un sistem de acces distribuit deoarece fiecare mașină din rețea care utilizează NIS accesează un server central, numit master NIS sau ypmaster, pentru drepturi de acces. În cadrul rețelelor mai mari, există unele mașini care sunt desemnate ca slave sau ypslave , care mențin informația de acces actualizată. În cazul unei căderi a serverului master , un calculator de tip slave preia controlul asupra funcțiilor de validare.

Notă: Există două versiuni de YP sau NIS care sunt utilizate, în general. Prima versiune(Versiunea 1) prezenta anumite probleme în anumite situații, așa încât Versiunea 2 a fost imediat produsă. Cu toate acestea, unele sisteme încă mai utilizează vechea versiune.

YP sau protocolul NIS (ambele denumiri sunt valabile, deși NIS ar fi preferabil de folosit) are un set de proceduri care permit o căutare a serverelor master, acces la fișierele utilizatorilor și funcții de organizare a sistemului. O altă procedură este folosită pentru a transfera copii ale fișierului de acces al masterului. Cu NIS, se pot grupa un număr de mașini într-o subrețea NIS, numită „domeniu” (a nu se confunda cu noțiunea de domeniu Internet). Fiecare astfel de domeniu are mașini master și slave proprii.

NIS păstrează informațiile de acces într-o mulțime de hărți, fiecare astfel de hartă corespunzând la o zonă sau la un domeniu anume din rețea. Acest lucru permite mai multor grupuri să folosească același master NIS, dar să aibă diverse drepturi de acces. Hărțile NIS nu trebuie să corespundă domeniilor DNS, ceea ce conferă o flexibilitate mai mare în configurație. Aceste hărți conțin o mulțime de înregistrări în format ACSII, fiecare cu un index pentru o căutare mai rapidă(acest index este, de obicei, numele utilizatorului). Înregistrările au aceeași structură ca un fișier `/etc/passwd` obișnuit), pentru compatibilitate și simplitate.

Notă: Utilizarea NIS-ului nu elimină necesitatea unei mulțimi complete de fișiere de acces pe fiecare mașină, din moment ce NIS sau YP este încărcat după ce mașina a fost pornită. Asupra fișierelor de sine stătătoare ar trebui să aibă acces cel puțin administratorul de sistem, deși este mai practic să fie incluși și cei mai frecvenți utilizatori, în cazul unei căderi a rețelei, împiedicând accesul la directoarele NIS.

NIS nu este restricționat doar la utilizatori. Orice fișier poate fi setat pentru a utiliza NIS, ca și lista mașinilor din cadrul unei rețele (fișierul `/etc/hosts`). Astfel este necesar să fie făcută doar o singură schimbare asupra acestor fișiere în orice rețea. O mulțime de aliasuri poate fi gestionată de către NIS sau YP.

Unele comenzi specifice YP/NIS sunt legate de protocol, deși majoritatea administratorilor de sistem fixează aliasuri pentru a minimiza impactul asupra utilizatorilor. Pentru marea parte a utilizatorilor, este necesară doar o singură comandă : `yppasswd` pentru a schimba o parolă.

Aceasta este de obicei folosită sub denumirea de `passwd`, care este comanda obișnuită pentru schimbarea parolei.

Unele implementări NIS pentru Linux sunt mai bune decât altele. O nouă implementare este pe cale să apară, numită NYS, ce oferă cea mai mare flexibilitate. NYS (sau o versiune mai veche a uneia dintre versiunile NIS de Linux) este inclusă pe majoritatea distribuțiilor de Linux, pe CD.

Instalarea NIS

NIS are două componente: serverul și clientul. Dacă există deja un server NIS în cadrul rețelei tale, va trebui să mai instalezi doar părțile ce țin de client. Oricum, pentru setarea un sistem de server Linux, vei avea nevoie de ambele componente.

Poți alege între două produse server NIS, care există în mod curent în distribuțiile de Linux: `ypser` și `yps`. Alegerea serverului ce urmează a fi folosit nu este importantă, din moment ce ambele furnizează servicii complete. Totuși, sistemul `ypserv` are o securitate ceva mai mare decât `yps`. Poți obține soft NIS prin siteurile Linux FTP și BBS.

Pentru a instala și programul server, acesta trebuie copiat în `/usr/sbin` (sau în altă locație a unui fișier binar ce este accesat de obicei). Apoi, se creează un director special care va reține harta fișierelor pentru domeniul nostru (de reamintit că acesta este un domeniu NIS și nu unul Internet). De obicei, aceste fișiere se pun într-un director ca `/var/yp/tpci` (ultima componentă a căii este numele domeniului nostru).

Serverul NIS poate suporta câteva fișiere de mapare. În general, fișierele reflectă o imagine a fișierelor Linux standard, dar sunt numite astfel încât să arate dacă sunt accesate după nume sau după alte criterii (ca adrese IP sau nume utilizator). De exemplu, există două copii ale fișierului `/etc/passwd` care sunt menținute de către NIS: `passwd.byname` și `passwd.byuid`. Următoarele fișiere sunt utilizate de NIS și de către hărțile lor corespunzătoare:

```
/etc/group group.byname, group.bygid  
/etc/hosts hosts.byname, hosts.byaddr  
/etc/networks networks.byname, networks.byaddr
```

`/etc/passwd passwd.byname, passwd.byuid`
`/etc/protocols protocols.byname, protocols.bynumber`
`/etc/rpc rpc.byname, rpc.bynumber`
`/etc/services service.byname, services.bynumber`

Toate aceste fişiere de mapare sunt păstrate într-un format numit DBM (un program de baze de date simplu). Sistemul Linux include adesea şi o versiune GNU pentru DBM, numită GDBM.

Dacă se foloseşte ypserv, se recomandă utilizarea utilitarului ypMakefile pentru construirea fişierelor de baze de date pentru NIS. Copiaţi fişierul în directorul ce conţine fişierele de mapare, redenumiţi-l ca Makefile, după care editaţi-l în conformitate cu hărţile pe care le doriţi în domeniul vostru. Acest lucru este realizat de una dintre primele linii, care arată astfel:

```
all: hosts networks protocols rpc services passwd group
```

Îndepărtaţi intrările pentru care nu doriţi fişiere de mapare. Dacă alegeţi utilizarea serverului yps, va fi necesară folosirea programului makedbm pentru construirea indexului din aceste fişiere.

Pentru a seta softul client pe un sistem Linux (permiţându-i să se conecteze prin ypmaster la un alt server), va trebui pregătit nucleul pentru folosirea sistemului NIS. Se începe cu setarea numelui pentru ypmaster în cadrul fişierului `/etc/yp.conf`. În fişier trebuie să apară o linie ca mai jos:

```
ypserver wizard.tpci.com
```

Această linie indică maşina locală la care trebuie să ajungă ypserver. (Unele versiuni de Linux folosesc termenul de server care poate fi înlocuit prin ypserver în fişierul `yp.conf`). Unele sisteme de Linux mai vechi folosesc un fişier de două linii `/etc/yp.conf` care listează numele domeniului şi serverul pe linii separate, ca mai jos:

```
domainname tpci.com
```

```
server wizard
```

Setați fișierul *yp.conf* în mod read pentru a putea fi citit de către utilizatori, grup și alții. Apoi testați instalarea NIS utilizând comanda *ypcat*:

```
ypcat passwd.byname
```

Această comandă ar trebui să afișeze harta *passwd.byname* a serverului master. Dacă se obțin mesaje de eroare, atunci este posibil ca mașina locală să nu fi contactat serverul la distanță în mod corect. Dacă apare mesajul:

Can't bind to server which serves domain

atunci înseamnă că fie avem de a face cu un server deficitar, fie s-a trecut un nume greșit în fișierul */etc/ypconf*. Pentru verificarea serverului folosiți comanda *ping*, pentru a vedea dacă conexiunea rețea este intactă.

În momentul în care ne-am asigurat că această conexiune NIS funcționează corect, se poate decide care fișiere să fie primite de la *ypmaster* și care să fie păstrate local. În majoritatea cazurilor, se dorește primirea fișierelor *passwd* și a celor de grup de la server, însă restul fișierelor pot fi păstrate local. Ordinea în care mașina locală și serverul NIS sunt verificați pentru fiecare tip de fișier de mapare este controlată de către fișierul */etc/nsswitch.conf*. Acest fișier arată astfel:

```
hosts: nis files
```

```
networks: nis files
```

```
services: files
```

```
rpc: files
```

```
protocols: files
```

Fiecare linie începe cu numele unui fișier, urmat de cuvinte cheie care controlează locul în care sistemul Linux caută fișierul. Iată câteva valori valide (care sunt citite și prelucrate în ordine):

dbm Folosește un fișier în cadrul fișierelor DBM sub */var/dbm*

dns Folosește serverul de nume de domeniu

files Folosește fișiere locale

nis Folosește serverul NIS

Sunt valabile și multe alte opțiuni în unele versiuni îmbogățite în proprietăți ale NIS, însă acestea sunt primele alegeri (și ar trebui să fie îndeajuns pentru majoritatea sistemelor Linux).