Lucrarea 8.

Instalarea şi configurarea unui server de acces PPP *(Dial In)* pe un sistem de operare Linux.

8.1. Etapele instalării şi configurării serverului de acces PPP.

Sistemele de operare Linux oferă resurse importante pentru realizarea unor echipamente de comunicaţie, cu performanţe ridicate. Între acestea se află şi cele care pot să asigure conexiuni prin linii telefonice dedicate sau comutate. În cursul nr. 8 s-a prezentat modul în care trebuie configurate anumite componente pentru a permite **conectarea** sistemului, ca terminal client, la un **server de acces** *(dial-in)* [12, 19, 20].

În această lucrare se prezintă modul în care trebuie configurate anumite componente pentru ca sistemul să devină server de acces prin apel telefonic *(Dial In)* şi să permită transferul datelor între acesta şi terminale-client, aflate la distanţă.

Pentru configurare trebuie parcurse mai multe etape, care depind de cerinţele impuse conexiunii şi de complexitatea acesteia. Să presupunem că se doreşte realizarea unui server care să permită conectarea unor terminale, pe care operează sisteme **Microsoft Windows** şi **Linux**, care trebuie să se autentifice prin **CHAP** şi să primească **automat** adresa **IP**, de nivel **OSI 3** şi adresa serverelor de nume **DNS**.

Conexiunea trebuie să permită transferul datelor comprimate şi criptate prin protocolul **MPPE** Pentru aceasta este nevoie de câteva componente foarte importante: un modem intern sau extern, o aplicaţie de tip `getty` şi pachetul aplicaţiei server-`pppd`. După ce se verifică dacă aceste componente se află la îndemână, se poate trece la instalarea şi configurarea lor, parcurgând următoarele etape:

- Instalarea modemului;
- Configurarea interfeţei seriale;
- Configurarea aplicaţiei `mgetty`;
- Configurarea serviciului `pppd`;
- Configurarea funcţiilor de filtrare şi de manipulare a pachetelor;
- Verificarea funcţionării serverului dial-in.

8.2. Instalarea modemului şi configurarea interfeţei seriale.

Pentru conectarea la o linie telefonică comutată se poate folosi atât un modem extern cât şi unul intern, cu condiţia ca acesta să fie suficient de robust pentru a putea opera într-un regim de funcţionare sever, funcţionare continuă de lungă durată şi apeluri frecvente din centrala telefonică. De aceea, se recomandă utilizarea unui modem extern, pentru care producătorul specifică, în mod explicit, aceste condiţii.

Se verifică mai întâi dacă sistemul de calcul dispune de o interfaţă serială **EIA/TIA 232** accesibilă din exterior, printr-un conector **DB25** sau **DB9**. În general, la instalarea sistemelor de operare Linux, interfeţele seriale sunt recunoscute şi instalate automat, cu opţiuni care le permit funcţionarea la performanţe optime [12, 19]. În caz contrar, se va studia procedura de instalare prezentată în cursul „Interfeţe seriale”. Pentru a verifica dacă aceste interfeţe sunt instalate şi recunoscute de către sistemul de operare, se vor utiliza comenzile:

```
root@masserv:~# more /proc/devices
Character devices:
...
```

```
  4 ttyS
  5 cua
...
root@masserv:~# more /proc/ioports
...
02f8-02ff : serial(auto)
...
03f8-03ff : serial(auto)
...
```

Dacă se va utiliza interfaţa serială `/dev/ttyS1`, atunci aceasta poate fi verificată şi cu comanda:

```
root@masserv:~# setserial /dev/ttyS1
/dev/ttyS1, UART: 16550A, Port: 0x02f8, IRQ: 3, Flags: spd_vhi
root@masserv:~#
```

Dacă se regăsesc aceste secvenţe, înseamnă că interfeţele sunt instalate şi pot fi configurate, în scopul utilizării lor pentru conectarea modemului. Se va căută în manualul modemului rata **baud maximă** pe care acesta o **suportă -** de exemplu **115.200** baud, şi modul de **control** al **fluxului de date** *(flow control)* - ***hardware***. Aceste opţiuni se vor folosi pentru configurarea interfeţei seriale, la care se va ataşa modemul. Comanda `setserial` se utilizează însoţită de parametri corespunzători modemului folosit, aşa cum se poate vedea în rândurile următoare:

```
root@masserv:~# setserial /dev/ttyS1 spd_vhi
root@masserv:~# setserial -a /dev/ttyS1
/dev/ttyS1, Line 1, UART: 16550A, Port: 0x02f8, IRQ: 3
        Baud_base: 115200, close_delay: 50, divisor: 0
        closing_wait: 3000
        Flags: spd_vhi skip_test

root@masserv:~#
```

Opţiunea `spd_vhi` fixează rata de transfer a interfeţei la o **valoare iniţială** de **115.200** baud, care apoi se poate **modifica,** pentru rate de transfer inferioare, cu opţiunea `divisor=x`, unde `x` este **divizorul** cu care trebuie împărţită valoarea 115.200 pentru a obţine rata dorită (vezi cursul „Interfeţe seriale"). Pentru a automatiza configurarea interfeţei seriale, la aceşti parametri, se va introduce în fişierul `/etc/rc.d /rc.local` următoarea linie de cod:

```
/sbin/setserial /dev/ttyS1 port 0x2F8 irq 3 spd_vhi
```

Notă: În cazul utilizării unui modem intern echipat cu interfaţă serială (modem intern *hardware*) este necesară adăugarea acestei linii, cu opţiunile corespunzătoare acestuia, după verificarea parametrilor la linia de comandă (în urma executării comenzii, sistemul nu trebuie să întoarcă mesaje de eroare).

8.3. Configurarea şi verificarea funcţionării modemului.

După configurarea interfeţei seriale şi conectarea modemului la aceasta, trebuie verificată funcţionarea „lanţului" modem-interfaţă-serială înainte de configurarea următoarelor aplicaţii [18, 19]. Pentru a verificarea modemului se poate folosi utilitarul `minicom` (aplicaţie „emulator de terminal") cu opţiunea `-s`, care permite editarea fişierului `/etc/minirc .dfl`.

```
root@masserv:~# minicom -s
```

Aplicaţia `minicom` are un meniu de configurare cu ferestre din care se va alege `Serial port setup` şi prin apăsarea literei corespunză-toare, se va scrie:

```
A -   Serial Device     /dev/ttyS1
...
E -   Bps/Par/Bits          115200 8N1
F -   Hardware Flow Control Yes
```

Se revine apoi în meniul principal de unde se intră în `Modem and dialing` şi se editează şirul de iniţializare al modemului, astfel:

```
A - Init string
~^M~AT S7=45 S0=0 L1 V1 X4 &C1 E1 Q0 M2 L1^M
```

Acest şir reprezintă un set de opţiuni, de configurare a modemului, trimis spre acesta la pornirea aplicaţiei `minicom`. Se salvează apoi opţiunile `Save as dfl` şi se iese cu `Exit` în fereastra principală a programului. Aici, se va scrie comanda `AT` care trebuie să fie urmată de răspunsul modemului `OK`.

Pentru a verifica dacă linia telefonică este conectată şi funcţionează se va scrie comanda `ATH1`, care trebuie să activeze conectarea modemului la linia telefonică şi apariţia, în difuzorul modemului, a tonului de linie. Dacă acest lucru se verifică, se scrie comanda `ATH` şi se trece la faza următoare de verificare a unui apel.

Utilizând comanda `ATDxxx`, unde `xxx` este un număr de telefon oarecare (de preferinţă al unui prieten care nu se supără dacă primeşte apeluri, în care se aud zgomote ciudate!) modemul se conectează la linie, aşteaptă tonul şi formează numărul specificat. Dacă apelul decurge corect, se apasă orice tastă pentru a închide conexiunea.

Un exemplu în care se poate vedea cum decurge verificarea, prin înscrierea manuală a comenzilor „AT”, se poate vedea mai jos:

```
    Welcome to minicom 2.1
    OPTIONS: History Buffer, F-key Macros, Search
History Buffer, I18n
    Compiled on Sep 18 2004, 16:54:29.

    Press CTRL-A Z for help on special keys

    AT S7=45 S0=0 L1 V1 X4 &c1 E1 Q0
    OK
    AT
    OK
    ATH1        aici trebuie să se audă tonul de linie
    OK
    ATH         aici trebuie să se oprească tonul de linie
    OK
    ATD402      aici trebuie să se audă tonul de linie urmat de
                formarea numărului şi de tonul de apel propriu-zis
    NO           închiderea conexiunii
CARRIER
```

Se iese din `minicom` cu comanda `Ctrl+A şi apoi X`

8.4. Configurarea utilitarului `mgetty`.

Utilitarul `mgetty` este o aplicaţie, din familia `getty`, care gestionează conexiuni prin interfeţe seriale (vezi cursul „Interfeţe Seriale"). Se pot folosi diferite aplicaţii de acest tip, dar `mgetty` a fost **optimizată** pentru a permite operarea împreună cu **PPP**, permiţând lansarea automată a serviciului `pppd` în cazul detectării unei astfel de cereri, primită de la un terminal, în faza de conectare [25]. Lansarea automată în execuţie a aplicaţiei `mgetty` se poate face dacă în fişierul `/etc/inittab` este introdusă următoarea linie:

```
# Adaugarea unui port serial pentru server de acces Dial-in
# prin modemul conectat la /dev/ttyS1:
#
s2:234:respawn:/sbin/mgetty /dev/ttyS1 -D -s 115200 -n 6 -x 4
#
```

Opţiunea `/dev/ttyS1` reprezintă interfaţa pe care o va utiliza `mgetty`, `-D` limitează utilizarea modemului doar pentru modul date (nu ca fax sau robot de telefon), `-s` defineşte rata de transfer, `-n` numărul de apeluri din centrală până la acceptarea apelului, iar `-x` nivelul de înregistrare a evenimentelor apărute în cursul conectării.

Opţiunile de configurare, ale interfeţei seriale la care este conectat modemul, în cazul în care acestea nu sunt trecute în linia de comandă a fişierului `/etc/inittab`, sunt înscrise în fişierul `/etc/mgetty+ sendfax/mgetty.config`. Acesta trebuie să conţină următoarele linii:

```
# Modem extern conectat la /dev/ttyS1
#
port ttyS1
  data-only y
  speed 115200
  port-owner uucp
  port-group uucp
#
```

Pentru a putea utiliza opţiunea de lansare automată a serviciului `pppd`, pachetul `mgetty` trebuie compilat cu secvenţa „`-DAUTO_PPP`" activă, care permite apoi utilizarea opţiunii `/AutoPPP/`. Pentru modul în care se editează opţiunile de compilare şi procedura de realizare a acesteia, se va consulta documentaţia care însoţeşte fişierele cu codurile sursă ale pachetului `mgetty+sendfax`.

În afară de opţiunile de configurare prezentate în cursul „Interfeţe Seriale", în fişierul `/etc/mgetty+semdfax/login.config` trebuie să fie activată următoarea linie:

```
#
/AutoPPP/   -   a_ppp   /usr/sbin/pppd   file
/etc/ppp/options.ttyS1
#
*       -       -     /bin/login     @
#
```

Prima linie activează lansarea automată în execuţie a serviciului `pppd` după ce `mgetty` detectează secvenţa prin care terminalul, care a făcut apelul, a transmis cererea de utilizare a protocolului **PPP**. Serviciul `pppd` va fi lansat în execuţie cu opţiunile înscrise în fişierul `/etc/ppp /options.ttyS1`.

Cea de a doua linie activează autentificarea terminalului, care cere conectarea, cu ajutorul aplicaţiei `/bin/login` bazată pe numele unui utilizator, înregistrat în sistemul de operare. Acest lucru este valabil doar dacă în fişierul `/etc/ppp/options.ttyS1` conţine linia `login`.

8.5. Configurarea serviciului pppd.

Serviciul `pppd` se configurează prin editare unor fişiere aflate în directorul `/etc/ppp`. Fişierul care conţine opţiunile de configurare, a modului de funcţionare, al acestui serviciu este `/etc/ppp/options .ttyS1` [20].

Pentru a permite conectarea unor terminale, pe care operează sisteme din familia MS Windows, cu opţiuni de autentificare prin **CHAP** şi criptare **MPPE**, acest fişier trebuie să conţină opţiunile care se pot vedea în exemplul de mai jos:

```
# Optiuni Generale:
lock
# Setari specifice liniei seriale si ale Modemului:
asyncmap 0
crtscts
modem
# Optiuni de autentificare obligatorie prin CHAP:
auth
login
require-chap
# Opţiunea de criptare cu protocolul MPPE:
require-mppe
# Optiune de operare necesară operării pppd ca proces activ
nodetach
# Serverul transfera în retea cererile arp
proxyarp
# Opţiuni de configurare a serverului de evenimente, syslog
kdebug 4
debug
# Adresa Locala IP: Adresa Terminal Client IP
192.168.121.1:192.168.121.2
# Servere (DNS) Domain Name System:
ms-dns 193.226.5.33
ms-dns 193.226.5.35
#
```

Aceste opţiuni permit conectarea de la distanţă prin linie telefonică comutată, utilizând protocol de nivel **OSI 2 PPP**, protocol de reţea **IP**: şi cu autentificare prin **CHAP**, pe baza unui nume de utilizator înregistrat pe sistem.

În cazul în care nu se doreşte autentificarea prin aplicaţia `/bin/ login`, se pot folosi fişierele `/etc/ppp/pap-secrets`, pentru autentificarea prin **PAP**, sau `/etc/ppp/chap-secrets` pentru cea prin **CHAP**. Conţinutul acestor fişiere este asemănător şi conţine informaţii, aşa cum se poate vedea în exemplul următor:

```
#
# Nume-client   Server   Parola   Adresa-IP
     *            *       ""         *
  florind         *   Pc_CuRs2007    *
#
```

În cazul utilizării aplicaţiei `/bin/login` pentru autentificarea utilizatorilor, înregistraţi în sistemul pe care operează serverul de acces, sunt valabile opţiunile de pe prima linie care nu începe cu #, caracterul * are rolul de a înlocui orice nume înregistrat, împreună cu parola sa de acces reprezentată prin caracterele ””.

Se poate restricţiona şi adresa terminalului de pe care se face cererea, în cazul când aceasta se fixează de către client, în caz contrar când aceasta este furnizată de către server, se foloseşte caracterul de înlocuire *.

Accesul altor utilizatori, care nu sunt înregistraţi în sistem, este permis doar dacă numele lor şi parola se adaugă, pentru fiecare dintre aceştia, în câte o linie suplimentară (cea de a doua care nu începe cu #).

8.6. Configurarea funcţiilor de filtrare şi manipulare a pachetelor *(firewall)*.

În cazul în care sistemul, pe care s-a configurat serverul de acces, are o conexiune spre alte resurse de reţea, sau Internet şi se doreşte ca acestea să fie împărtăşite clienţilor care se conectează de la distanţă, trebuie configurate tabele de rutare sau de translatare a adreselor (NAT).

Pentru aceasta trebuie scris un fişier care se execută automat la pornirea sistemului în care să fie trecute regulile de manipulare a pachetelor. Acest fişier se poate numi oricum, dar trebuie inclus în directorul `/etc/rc.d/` unde sunt aşezate aceste tipuri de fişiere, de exemplu `/etc/rc.d/rc.dial-nat`. În acest fişier se scriu instrucţiuni prin care se apelează aplicaţia `iptables` care permite modificarea regulilor de manipulare a pachetelor de reţea. Un astfel de fişier trebuie să conţină următoarele linii:

```
# Fisier de configurare a pentru accesul la reţea prin
# dial-in şi ieşire prin interfaţa eth0:
#
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
#
# Acceptarea transferului pachetelor iniţiate din
# interior de pe interfaţa dial-in
iptables -A FORWARD -m state --state NEW
 -s 192.168.121.0/30 -j ACCEPT
#
# şi a conexiunilor deja stabilite cu acestea:
#
iptables -A INPUT -m state --state ESTABLISHED,RELATED
 -j ACCEPT
iptables -A OUTPUT -m state --state NEW,ESTABLISHED,
 RELATED -j ACCEPT
#
# Listarea tebelelor de manipulare a pachetelor
iptables -L -n
```

După aceasta se lansează în execuţie instrucţiunile înscrise în fişierul `/etc/rc.d/rc.dial-nat` astfel:

```
root@masserv:~# /etc/rc.d/rc.dial-nat
Try `iptables -h' or 'iptables --help' for more information.

Chain INPUT (policy DROP)
target   prot opt source     destination     state
ACCEPT   all  --  0.0.0.0/0  0.0.0.0/0    RELATED,ESTAB LISHED
ACCEPT   all  -- 192.168.121.0/30 0.0.0.0/0 NEW
...
Chain OUTPUT (policy DROP)
ACCEPT   all  -- 0.0.0.0/0     0.0.0.0/0    NEW,RELATED,
ESTABLISHED
```

Aceste instrucțiuni pot fi trecute într-un fişier mai complex care defineşte toate regulile de acces şi face obiectul funcții de control a accesului, numite uzual „ziduri de foc" *(firewall)*.

8.7. Lansarea şi verificarea funcționării serverului *Dial In*.

După ce toate aceste fişiere au fost editate se porneşte din nou sistemul, sau se folosesc comenzi de inițializare a sistemului cu noile opțiuni. Astfel, pentru re-inițializarea instrucțiunilor înscrise în `/etc/inittab` se foloseşte comanda:

```
root@masserv:~# init q
```

Această comandă citeşte conținutul fişierului `/etc/inittab` şi re-inițializează sistemul cu noile opțiuni. După inițializare se verifică dacă este pornită aplicația mgetty, folosind comanda de listare a proceselor, astfel:

```
root@masserv:~# ps -ax
...
4047 ?        Ss     0:00 /sbin/mgetty /dev/ttyS2 -D -s 115200 -n
6 -x 4
...
```

Această comandă, însoțită de opțiunile `-ax` (toate procesele), întreoghează sistemul de operare şitipăreşte un raport care conține lista proceselor pornite şi a unor parametri ai acestora, în care trebuie să se regăsească procesul corespunzător aplicației `mgetty`. Această aplicație înregistrează în fişierul `/var/log/mgetty.log.ttyS1` evenimentele apărute în timpul activității sale şi ocazia conectării unui terminal, prin linia serială pe care o controlează. Conținutul acestui fişier depinde nivelul de înregistrare şi de situațiile întâlnite, putând înregistra tot traficul în cazul utilizării opțiunii `-x 7`. O sesiune de conectare, înregistrată cu nivelul 4, poate să arate ca în exemplul de mai jos:

```
...
(Reinițializarea modemului):
02/07 13:58:30 yS1  checking if modem is still alive
02/07 13:58:30 yS1  mdm_send: 'AT' -> OK
02/07 13:58:30 yS1  waiting...

(Primirea apelului însoțit de o cerere de conectare PPP):
02/07 14:57:21 yS1  waiting for ``RING'' ** found **
02/07 14:57:21 yS1  waiting for ``RING'' ** found **
02/07 14:57:23 yS1  waiting for ``RING'' ** found **
02/07 14:57:28 yS1  waiting for ``RING'' ** found **
02/07 14:57:33 yS1  waiting for ``RING'' ** found **
02/07 14:57:38 yS1  waiting for ``RING'' ** found **
02/07 14:57:43 yS1  send: ATA[0d]
02/07 14:57:43 yS1  waiting for ``CONNECT'' ** found **
02/07 14:58:04 yS1  send:
02/07 14:58:04 yS1  waiting for ``_'' ** found **
02/07  14:58:07  #####  data  dev=ttyS1,  pid=4047,  caller='none',
conn='14400/14400

(Lansarea în execuție a componentei PPP):
 NONE', name='', cmd='/usr/sbin/pppd', user='/AutoPPP/'
--
02/07 14:59:25 yS1  mgetty: experimental test release 1.1.14
```

```
(Închiderea conexiunii şi reiniţializarea modemului):
02/07 14:59:26 yS1  check for lockfiles
02/07 14:59:26 yS1  locking the line
02/07 14:59:26 yS1  lowering DTR to reset Modem
02/07 14:59:27 yS1  send: \dATQ0V1H0[0d]
02/07 14:59:27 yS1  waiting for ``OK'' ** found **
02/07 14:59:30 yS1  send: ATS0=0Q0&D3&C1[0d]
02/07 14:59:30 yS1  waiting for ``OK'' ** found **
02/07 14:59:31 yS1  waiting.
```

Alături de acest fişier, se mai poate consulta şi raportul trimis de componenta `pppd` serverului `syslog`, fişier a cărui locaţie depinde de opţiunile de configurare a sistemului, în general în `/var/log/debug`.

Pe un terminal pe care operează un sistem MS Windows, echipat cu modem, se instalează şi configurează o aplicaţie-client de conectare prin linie telefonică *(dialer)*, cu opţiunile prezentate în enunţul temei.

Pentru Windows XP, se alege din meniul *Start>Settings>Network Connections NewConnection Wizard* şi se urmează instrucţiunile pentru crearea unei conexiuni la Internet. Se alege configurarea manuală, folosind modem. La numele furnizorului de servicii se scrie Linux şi numărul de telefon în căsuţa destinată acestuia. După înscriere numelui de utilizator şi a parolei. După terminarea configurării şi salvarea opţiunilor, se porneşte aplicaţia de conectare. Din butonul pentru proprietăţi se alege *Security* unde se activează opţiunile: *Require secured Password* şi *Require data encryption*. Din meniul *Networking*, la tipul serverului, se alege PPP, iar la proprietăţile *Protocolului Internet* se alege *Obtain an Ip address autoamatically* şi *Obtain DNS server address automatically*.

Se revine apoi în fereastra de conectare şi seapasă **Connect**. Procedura de conectare trebuie să decurgă normal şi să se încheie cu un mesaj afirmativ de notificare. Pentru a vedea dacă această conexiune este activă, se deschide un terminal text, apăsând **StartYRun** unde se scrie **cmd** şi se apasă **OK**. În această fereastră se scriu următoarele comenzi-linie:

```
C:\Documents and Settings\Administrator>route print
===========================================================
Interface List
0x1 ........................... MS TCP Loopback interface
0x2 ...00 30 05 3d f9 3e ...... Intel(R) PRO/1000 VE Network
Connection - Packet
Scheduler Miniport
0x20004 ...00 53 45 00 00 00 ...... WAN (PPP/SLIP) Interface
===========================================================
Active Routes:
Network Dest.    Netmask       Gateway       Interface Metric
...
192.168.121.0   255.255.255.0  192.168.121.2  192.168.122.3 1
192.168.122.255 255.255.255.255 192.168.122.2 192.168.122.2 5
...
C:\Documents and Settings\Administrator>ping 192.168.121.1

Pinging 192.168.122.1 with 32 bytes of data:

Reply from 192.168.122.1: bytes=32 time=16ms TTL=64
Reply from 192.168.122.1: bytes=32 time=18ms TTL=64
Reply from 192.168.122.1: bytes=32 time=25ms TTL=64
Reply from 192.168.122.1: bytes=32 time=17ms TTL=64
```

```
Ping statistics for 192.168.122.1:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 16ms, Maximum = 25ms, Average = 19ms

C:\Documents and Settings\Administrator>
```

Aceste mesaje confirmă funcționarea conexiunii dintre terminalul client şi serverul de acces *dial-in*. Semnificația informațiilor oferite de acestea şi modul în care se configurează alte opțiuni care fac posibil transferul datelor, între servicii aparținând celorlalte nivele **OSI**, se vor consulta cursurile următoare..