

13. SERVERUL NTP

13.1. Concepte.....	2
13.1.1. Timpul – determinare si sincronizare.....	2
13.1.2. De ce este nevoie de sincronizare.....	2
13.1.3. Modalitati de determinare a timpului si neconcordante.....	2
13.1.4. Standarde de timp.....	3
13.1.5. Distribuirea timpului oficial.....	3
13.2. Protocolul NTP.....	4
13.2. Serverul NTP.....	5
13.2.1. Fisa serverului.....	5
13.2.2. Configurare.....	5
13.2.2.1. Setari generale.....	5
13.2.2.2. Moduri de operare ale serverului.....	6
13.2.2.3. Stabilirea serverelor ajutatoare folosite pentru sincronizare.....	7
13.2.2.4. Operarea ca client de broadcast.....	7
13.2.2.5. Operarea ca server in modul broadcast.....	8
13.2.2.6. Controlul accesului.....	8
13.2.3. Monitorizarea serverului.....	9
13.2.3.1. Utilitarul ntpq.....	9
13.3. BIBLIOGRAFIE.....	9

13.1. Concepte

13.1.1. Timpul – determinare si sincronizare

A determina si mentine timpul curent presupune doua aspecte distincte:

- determinarea momentului exact in timp – are la baza un set de conventii: momentul considerat ca an 0, durata unei secunde etc. A pornit initial de la alternanta zi-noapte si cea a anotimpurilor; numai stabilind un anumit reper ca an 0 putem spune cu exactitate care este momentul curent
- sincronizarea ceasurilor – odata ce un om/dispozitiv a determinat cu o precizie oarecare momentul exact in timp, este necesara propagarea acestei informatii - “aliniera” altor ceasuri astfel incat ele sa indice corect timpul curent pe o perioada cat mai lunga de timp. Operatia de aliniere a unui ceas la altul poarta denumirea de sincronizare

Sincronizarea a doua ceasuri implica urmatoarele operatii:

- sincronizarea frecventei – cele doua ceasuri trebuie sa “bata” sincron. In realitate, intre orice doua ceasuri va exista o diferenta de frecventa, denumita **clock drift** sau **skew**
- sincronizarea momentului temporal – cele doua ceasuri trebuie sa indice acelasi moment de timp. Deplasamentul temporal intre indicatiile a doua ceasuri poarta de obicei denumirea de **offset**

Rezolvarea unui singur aspect din cele doua nu va rezolva problema:

- degeaba sincronizam frecventa a doua ceasuri, daca unul este in urma fata de celalalt. Frecventa comuna va garanta doar faptul ca, in timp, deplasamentul dintre cele doua ceasuri ramane constant
- degeaba sincronizam doua ceasuri la un moment dat, daca unul dintre ele are tendinta de a ramane in urma celuilalt si, in acest fel, deplasamentul dintre ele marindu-se odata cu trecerea timpului

13.1.2. De ce este nevoie de sincronizare

Multe operatii ale zilelor noastre sunt operatii distribuite in care sunt implicate statii si dispozitive hardware diverse, aflate deseori in arii geografice foarte diferite (si deci subiect de ora locala diferita). Oferim aici cateva exemple:

- tranzactii distribuite – ex: tranzactii bursiere, cumparaturi online etc
- asigurarea criptografica a documentelor, in care intervin si momente exacte de timp (timestamp)
- GPS (Global Positioning System) – pozitia se determina prin masurarea delay-ului in comunicatia cu patru sau mai multi sateliti
- monitorizare retea si logging centralizat – spre exemplu, pentru corelarea informatiilor de logging provenite de pe diferite statii sau echipamente hardware ale unei retele
- controlul traficului si pozitiei in aviatie
- sincronizare pentru conferinte realtime
- jocuri multiplayer prin internet

13.1.3. Modalitati de determinare a timpului si neconcordante

Initial, timpul oficial local se determina dupa soare, in functie de trecerea acestuia peste meridianul local. Fiecare localitate/diviziune administrativa efectua propria determinare a orei locale, ceea ce ducea la diferente de ora intre diferite localitati. Odata cu dezvoltarea mijloacelor de transport rapide acesta a devenit un neajuns, calatorul fiind nevoit sa-si modifice permanent ceasul pentru a se potrivi cu ora locala.

Pentru eliminarea acestei probleme s-a trecut la asa-numitul “timp standard” - o conventie ce presupune impartirea ariei terestre in zone (meridiane), toate localitatile unei zone avand aceeasi ora oficiala si diferind cu

exact o ora fata de zona invecinata. Ca reper s-a luat meridianul Greenwich unde se afla observatorul astronomic Royal Greenwich Observatory.

Modul initial de determinare a timpului (bazat pe observatii solare) a fost completat ulterior cu doua elemente ce ajuta la cresterea preciziei determinarii:

- in anii '50 au aparut ceasurile atomice (bazate pe cesiu, rubidiu etc), care au o frecventa foarte stabila. Avem astfel o modalitate de a urmari cu acuratete trecerea timpului (dar care necesita un reper pentru a putea determina momentul exact in timp)
- observatiile astronomice au inceput sa se raporteze la alte corpuri ceresti – stele sau quasari indepartati

S-a remarcat insa ca cele doua modalitati de determinare (astronomica si atomica) nu concorda. Acest lucru se intampla deoarece lungimea standard a unei secunde este stabilita conventional de catre SI (Sistemul International) bandu-se pe observatii ale lunii in anii 1700-1800, dar, pe de alta parte, miscarea Pamantului nu este perfect regulata, astfel incat durata zilei creste treptat, intr-un ritm foarte lent, ceea ce duce la o zi reala mai lunga de 86400 de secunde standard.

13.1.4. Standarde de timp

Cele doua modalitati de masurare a timpului au furnizat numeroase standarde, din care in ziua de astazi amintim:

- **UT (Universal Time)** – concept abstract ce desemneaza timpul determinat astronomic, prin raportarea miscarii de rotatie a pamantului la diverse corpuri ceresti (soare, luna, quasari etc.). Exista diferite forme de timp universal, din care exemplificam doua:
 - GMT (initial) – masurat in functie de trecerea soarelui peste meridianul Greenwich
 - UT1 – principala forma de UT actuala, masurat prin raportarea la stele si quasari indepartati
- **TAI (Temps Atomique International)** – timpul mediat ce foloseste ca surse peste 200 de ceasuri atomice din 70 laboratoare la nivel mondial. A fost sincronizat cu UT (Universal Time) in 1958 si de atunci s-a departat de acesta din cauza miscarii neregulate a Pamantului, ajungand la aproximativ 34s diferenta in 2009

Pentru a “impaca” timpul solar cu cel atomic, timpul “civil” este obtinut pe baza celui atomic (profitand de stabilitatea frecventei acestuia) dar adaugandu-i ocazional cate o secunda (numita **leap second**) pentru a-l alinia la cel astronomic. Astfel procedeaza principala referinta pentru timpul civil international, **UTC (Coordinated Universal Time)**, care urmareste TAI pe portiuni extinse (aceeasi frecventa), insa din cand in cand adauga leap seconds astfel incat diferenta fata de UT1 sa fie mentinuta sub 0.9 secunde.

Nota: timpul “civil” (cel distribuit populatiei) respecta impartirea in zone (time zones) si este subiect de daylight saving time (ora de vara/iarna).

13.1.5. Distribuirea timpului oficial

Odata ce laboratoarele oficiale determina timpul standard, el trebuie sa ajunga “in posesia” restului lumii, ceea ce presupune sincronizarea ceasurilor dupa cel “oficial”. Acest lucru se poate realiza pe diferite cai:

- radio – exista statii radio care se sincronizeaza folosind ceasuri atomice si care transmit informatii de timp, cum ar fi WWV (<http://tf.nist.gov/stations/wwv.html>)
- modem telefonic
- internet – protocolul NTP este cel folosit pentru sincronizarea ceasurilor mai multor computere cu un server central

13.2. Protocolul NTP

Protocolul NTP a aparut fiindca ca nu era rentabil ca fiecare dispozitiv dintr-o retea sa aiba atasata o sursa precisa de timp. Rolul unei surse de timp locale, hardware, a fost preluat de unul sau mai multe servere NTP care distribuie statiilor si echipamentelor din retea informatii legate de timp, permitandu-le sa-si sincronizeze ceasurile cu costuri minime. Singura conditie este ca statia sau echipamentul sa poata actiona ca client NTP.

Protocolul NTP transmite timpul UTC, fara a tine cont de fus orar, ora de vara/iarna etc. Este responsabilitatea software-ului client de a face translatiile necesare. Transportul datelor se face folosind protocolul UDP, portul 123. Precizia obtinuta variaza intre zecimi de milisecunda in WAN si LAN, pana la sub o microsecunda in cazul atasarii la server a unei surse de precizie (oscilator cu cesiu, receptor GPS etc).

Softul de NTP contine de obicei atat un modul de client (pentru cazul in care este necesara contactarea altor servere) cat si unul de server (pentru a oferi servicii de timp unuia sau mai multor clienti).

Nota: in Linux/Unix, softul de NTP este in general numit "serverul NTP" desi el include de fapt ambele functionalitati. Si acest material foloseste termenul de "server NTP" pentru a desemna distributia completa.

In functie de cum a fost configurat, softul de NTP poate juca unul din 3 roluri:

- **server primar** – este un server sincronizat direct de la o sursa hardware (receptor GPS sau radio, modem etc) care sincronizeaza la randul sau alte servere dar nu este sincronizat de catre acestea
- **client** – isi sincronizeaza ceasul folosind informatii de la unul sau mai multe servere NTP, insa nu ofera informatii de timp altor clienti
- **server secundar** – se sincronizeaza folosind informatii de la unul sau mai multe servere parinte si serveste unul sau mai multi clienti

Felul in care serverele NTP comunica unele cu altele sau cu sursele de timp hardware creeaza o structura ierarhica:

- sursele de timp hardware formeaza nivelul 0 (asa-numitul *stratum* 0)
- serverele direct conectate la surse de timp hardware formeaza stratum-ul 1. Acestea functioneaza de obicei exclusiv ca servere primare
- fiecare server care foloseste un altul pentru a-si sincroniza ceasul va face parte dintr-un stratum cu o unitate mai mic decat al serverului "parinte" (exemplu: serverele care obtin informatii de timp de la servere de stratum 1 formeaza stratum-ul 2)

Atunci cand doua softuri de NTP comunica, unul dintre ele joaca rolul de client si celalalt de server. In majoritatea cazurilor clientul este cel care initiaza comunicatia, adresand serverului o cerere, iar serverul raspunde la aceasta cerere (exceptia o reprezinta cazul broadcast/multicast, prezentat mai jos). Serverul este cel care furnizeaza informatia de timp, iar clientul cel care isi sincronizeaza ceasul dupa informatia primita.

*Nota: "Partenerul de dialog" al unui soft NTP (fie el client sau server) este deseori denumit **peer**.*

Serverul nu mentine informatii de stare despre clienti, in schimb clientii mentin astfel de informatii despre serverele cu care comunica, sub forma unor asa-numite **asocieri**. O asociere reprezinta o relatie de scurta sau lunga durata creata intre un client si un server, manifestata printr-o structura de date atasata ei, stocata pe client. O asociere poate fi:

- permanenta - initiata la pornirea serviciului NTP si care ramane activa pe parcursul operarii acestuia (ex: liste de servere definite in fisiere de configurare)
- temporara (efemera) – creata dinamic, in timpul operarii softului de NTP (de ex, la primirea de catre un client a unui mesaj broadcast de la un server NTP) si desfiintata in caz de timeout sau eroare

Protocolul NTP a fost gandit sa ofere redundanta – atunci cand o retea nu are posibilitatea de a obtine informatii de timp externe, serverele sale NTP pot comunica unul cu altul pentru a produce o aproximare cat mai buna a momentului de timp curent bazandu-se pe informatiile in posesia carora se afla in acel moment. De aceea un server NTP isi poate sincroniza ceasul fie in urma comunicatiei fie cu un server de stratum inferior (numar mai mic, deci mai aproape de sursele hardware), fie cu unul sau mai multe de acelasi nivel cu el, in functie de cum a fost configurat. Din acest punct de vedere, softul de NTP poate functiona in diferite variante:

- **multicast/broadcast** – serverul transmite periodic in retea informatii de timp, iar softurile de NTP care le primesc si au fost configurate sa accepte astfel de mesaje isi vor sincroniza ceasurile
- **client** – softul NTP configurat astfel trimite mesaje periodice catre unul sau mai multe peer-uri preconfigurate, aratandu-si disponibilitatea de a fi sincronizat de catre acestea
- **server** – softul NTP astfel configurat ofera sincronizare peer-urilor, dar nu se lasa sincronizat de acestea
- **simetric** – softul NTP isi anunta disponibilitatea de a sincroniza alte servere sau de a fi el insusi sincronizat de catre acestea (actionand astfel ca server sau ca client), in functie de algoritmul de selectie de peer cuprins in specificatia NTP. Clientul creeaza o asociere cu serverul, desfiintata atunci cand comunicatia inceteaza pentru o perioada lunga de timp (timeout)
 - modul **activ** – softul NTP trimite mesaje periodic (indiferent de accesibilitatea sau stratum-ul peer-ului), declarandu-si disponibilitatea de a il sincroniza sau de a fi sincronizat de catre acesta. Mod de lucru recomandabil pentru serverele care actioneaza in stratum-urile de jos, care au seturi de peer-uri preconfigurate
 - modul **pasiv** – o astfel de asociere este creata ca urmare a primirii unui mesaj de la un server care functioneaza in modul simetric activ si al carui stratum este mai mic sau egal cu al serverului care receptioneaza mesajul.

13.2. Serverul NTP

13.2.1. Fisa serverului

Iata cateva informatii de pornire ce trebuie cunoscute despre serverul NTP:

- executabil: **ntpd**
- fisier de configurare: **/etc/ntp.conf**
- pornire in mod debug: **ntpd -d**. Optiunea -d poate fi folosita de mai multe ori pentru a creste gradul de detaliu
- utilitar de monitorizare: **ntpq**
- utilitar de control si monitorizare: **ntpdcc**

13.2.2. Configurare

13.2.2.1. Setari generale

Prezentam cateva elemente de configurare generala, care nu depind de modurile de operare ale serverului:

- locatia fisierului log – se specifica cu comanda *logfile* urmata de calea catre fisierul dorit. Daca serverul nu este pornit cu optiunea -d, informatiile legate de functionarea sa vor fi plasate in acest fisier

```
logfile /var/log/ntpd.log
```

- setul de informatii ce vor fi consemnate in log – se regleaza cu comanda *logconfig* urmata de o lista de categorii de informatie. Fiecare categorie este reprezentata de un cuvânt compus din doua portii, prefixata cu un simbol:
 - prima portiune indica categoria de mesaj si poate fi *clock*, *peer*, *sys*, *sync* sau *all*

- ultima portiune indica tipul de informatie din cadrul categoriei alese si poate fi *info*, *events*, *statistics*, *status* sau *all*.
- simbolul din fata unei categorii poate fi =, + sau -

Exemple:

```
# se consemneaza absolut toate informatiile
logconfig =all
# doar evenimentele legate de sincronizare
logconfig =allsync
```

- felul in care serverul trateaza diversele interfete/adrese din sistem – se specifica cu comanda *interface* urmata de doi parametri:
 - actiunea aplicata adreselor specificate ca parametru secund – poate fi *listen* (serverul asculta pe acea adresa), *ignore* (serverul nu asculta pe adresa specificata) sau *drop* (serverul asculta pe acea adresa dar distruge toate pachetele primite)
 - adresa/setul de adrese ce fac subiectul primului parametru. Printre variante se numara *all*, *ipv4*, *ipv6* sau o adresa in format *adresa* sau *adresa/lungimemasca*

Exemple:

```
# serverul asculta pe toate adresele
interface listen all

# serverul asculta pe adresele ipv6 dar nu proceseaza mesajele (le distruge)
interface drop ipv6
```

- locatia asa-numitului *driftfile* – fisierul in care se inregistreaza “alunecarea” (abaterea) in frecventa a ceasului local fata de cel de referinta

```
driftfile /var/lib/ntp/ntp.drift
```

13.2.2.2. Moduri de operare ale serverului

Serverul NTP poate lucra in multiple moduri. Iata-le pe cele mai importante:

- **client pentru un set precizat de servere**
 - ceasul local este sincronizat cu cel al unuia sau mai multor servere declarate in fisierul de configurare. Relatia nu functioneaza si invers!
 - se declara folosind comanda **server** in *ntp.conf* urmata de IP-ul sau numele serverului
- **simetric**
 - permite o sincronizare a ceasului in oricare sens cu un server pereche (asa-numit “peer”). Softul de NTP al celor doua servere determina in ce sens trebuie sa aiba loc sincronizarea (primul server isi sincronizeaza ceasul cu cel de-al doilea sau invers)
 - se declara folosind comanda **peer** urmata de IP/nume al serverului pereche
- **broadcast**
 - serverul trimite broadcast-uri periodice catre adresa indicata, astfel incat clientii acelei retele sa-si poata sincroniza ceasurile
 - se declara folosind comanda **broadcast** urmata de o adresa de broadcast a unei retele
- **client de broadcast**

- serverul receptioneaza mesajele broadcast trimise de servere NTP din retea locala. Se poate impune ca aceste mesaje sa fie autentificate, pentru a nu risca “otravirea” serverului nostru cu informatii invalide
- se declara folosind comanda **broadcastclient**

Aceste moduri nu se exclud reciproc, ci pot fi cumulate. Spre exemplu, putem configura serverul NTP sa se sincronizeze (ca client) cu unul sau mai multe servere NTP, si in plus sa faca broadcast catre una sau mai multe retele locale direct conectate, care la randul lor vor actiona ca clienti broadcast pentru serverul nostru.

13.2.2.3. Stabilirea serverelor ajutatoare folosite pentru sincronizare

Atunci cand intentionam ca serverul nostru NTP sa se foloseasca de alte servere pentru a-si sincroniza ceasul, este necesar sa specificam fiecare dintre aceste servere printr-o comanda **server**, care are sintaxa urmatoare:

```
server adresa/nume optiuni
```

Primul parametru poate fi adresa IP a serverului ajutor sau numele sau DNS. Optiunile pot lipsi integral sau pot consta dintr-o insiruire de unul sau mai multi parametri separati prin spatiu. Iata cateva optiuni utile:

- **iburst** – cand serverul nu poate fi contactat, in loc de a trimite un singur pachet catre el va fi trimisa o serie de 8, spatiate la 2 secunde. Este util avand in vedere ca NTP foloseste UDP ca transport si ca un singur pachet se poate pierde
- **minpoll N** si **maxpoll N** (unde N este un numar) – specifica intervalul de timp minim/maxim intre doua mesaje NTP trimise de serverul nostru. N reprezinta o putere a lui 2 – cu alte cuvinte, numarul de secunde rezultat va fi egal cu 2^N . Minimul implicit este 4 (adica 16 s) iar maximul implicit este 10 (adica 1024 de secunde)
- **dynamic** – serverul este luat in calcul chiar daca nu este disponibil momentan. Optiunea este utila pentru clientii mobili – ex: laptop-uri care migreaza dintr-o retea in alta, serverele definite in fisierul ntp.conf nefiindu-le permanent disponibile
- **noselect** – marcheaza serverul ca fiind nefolosit. Este o modalitate de a dezactiva temporar un server dar pastrandu-l inca in fisierul de configurare

Exemple:

```
# specificare folosind nume DNS, interval minim de polling 32s
server 0.pool.ntp.org minpoll 4 iburst

# specificare folosind adresa IP; server temporar dezactivat
server 10.0.0.5 noselect

# specificare server folosind adresa; serverul nu este permanent disponibil
server 192.168.0.100 iburst dynamic
```

13.2.2.4. Operarea ca client de broadcast

Pentru a activa aceasta facilitate a serverului NTP este necesara folosirea comenzii **broadcastclient**. In acest mod de lucru, serverul nostru va lua in considerare mesajele broadcast primite din retelele la care este conectat. Setarile implicite impun ca mesajele de broadcast primite sa nu fie luate in seama decat daca sunt autentificate (folosind diversi algoritmi criptografici); putem elimina aceasta cerinta folosind comanda **disable auth**:

```
# activarea prelucrarii mesajelor NTP de tip broadcast primite
broadcastclient
# mesajele broadcast vor fi luate in considerare chiar daca nu sunt autentificate corect
```

```
# (presupune sa avem incredere deplina in restul retelei locale)
disable auth
```

13.2.2.5. Operarea ca server in modul broadcast

Atunci cand dorim ca serverul nostru sa ofere servicii de timp altor clienti din reseaua locala putem activa modul broadcast, folosind comanda cu acelasi nume urmata de adresa de broadcast a retelei. Daca intentionam sa servim mai multe retele vom folosi mai multe comenzi **broadcast**:

```
# facem broadcast in subretea 10.0.0.0/28
broadcast 10.0.0.16
# servim si retea 192.168.0.0.25
broadcast 192.168.0.128
```

13.2.2.6. Controlul accesului

Putem restrictiona accesul altor statii la serverul nostru folosind comanda **restrict**. Trebuie inteles de la bun inceput ca ea afecteaza toate categoriile de trafic initiat sau receptionat de serverul nostru, cum ar fi:

- mesaje NTP primite de la clienti (cereri NTP)
- mesaje NTP primite de la alte servere (raspunsuri NTP)
- mesaje de control primite de la utilitare care ruleaza pe alte masini (ex: ntpq, ntpdc)

Sintaxa generala a lui restrict este:

```
# pentru a ne referi la o singura adresa
restrict adresa optiuni

# pentru a ne referi la un intreg subnet
restrict adresa mask masca optiuni
```

Optiuni:

- **ignore** – nu ia in considerare niciun fel de pachet venit de la statia/retea specificata (nici macar mesaje de control)
- **noquery** – interzice mesajele de control (provenite de la ntpq sau ntpdc), dar permite restul de mesaje
- **nomodify** – interzice acea parte a mesajelor de control care produc modificari ale configurarii serverului. Mesajele de control care doar solicita informatie sunt permise; la fel si mesajele care tin de serviciul de timp
- **noserve** – interzice toate mesajele care tin de serviciul de timp, permitand exclusiv accesul mesajelor de control
- **nopeer** – interzice mesajele care ar crea o asociere noua cu un alt server (ex: mesaje simetric activ, mesaje broadcast etc)
- **notrust** – interzice pachetele neautentificate care ar crea o asociere noua

In fisierul de configurare vor exista de obicei mai multe comenzi *restrict*. Prima regula *restrict* stabileste politica din oficiu, aplicabila tuturor mesajelor care nu corespund altor reguli. Ea trebuie sa aiba adresa 0.0.0.0 si masca 0.0.0.0 (sau, alternativ, se poate folosi cuvantul cheie *default*):

```
restrict 0.0.0.0 mask 0.0.0.0 ignore

# ...alternativ:
restrict default ignore
```

Iata mai jos un exemplu comentat de serie de directive restrict:

```
# implicit, serverul va schimba mesaje de timp cu oricine, inasa va interzice mesajele de control  
restrict default noquery nopeer  
  
# statia locala poate controla complet serverul (lipsesc optiunile care restrictioneaza)  
restrict 127.0.0.1  
  
# statia 10.0.0.100 poate obtine informatii de stare dar nu poate configura serverul de la  
distanta  
restrict 10.0.0.100 nomodify  
  
# clientii din reseaua 192.168.1.0/24 au acces total, inasa numai daca sunt autentificati  
restrict 192.168.1.0 mask 255.255.255.0 notrust
```

13.2.3. Monitorizarea serverului

13.2.3.1. Utilitarul ntpq

Utilitarul ntpq (NTP query) permite extragerea de informatie de la un server NTP. El poate fi folosit in doua moduri:

- modul non-interactiv, in care categoria de informatie dorita este specificata sub forma de optiuni in linia de comanda la apelarea lui *ntpq*
- modul interactiv (implicit), in care *ntpq* ofera o interfata text unde utilizatorul poate solicita serverului informatii sub forma unor comenzi *ntpq*. Iata cateva exemple:
 - **peers** (prescurtata **pe**) – afiseaza lista de peer-uri ale serverului interogat
 - **associations** (prescurtata **as**) – afiseaza lista de asocieri curente. Fiecare asociere este identificata printr-un association ID
 - **rv** (de la remote variables) – urmata de un id de asociere, comanda afiseaza toate informatiile de stare ale acelei asocieri

13.3. BIBLIOGRAFIE

- Moduri de operare ale softului de NTP:
http://www.bytefusion.com/products/ntm/ptnt/3_3modesofoperation.htm
- <http://www.meinberg.de/english/info/ntp.htm>
- Protocolul NTP v3: <http://www.rfc-editor.org/rfc/rfc1305.txt>
- Protocolul NTP v4: <http://www.eecis.udel.edu/~mills/database/reports/ntp4/ntp4.pdf>
- Lista directivelor din ntp.conf: <http://www.eecis.udel.edu/~mills/ntp/html/comdex.html>
- Documentatia distributiei NTP: <http://www.eecis.udel.edu/~mills/ntp/html/index.html>
- NTP FAQ: <http://www.ntp.org/ntpfaq/NTP-s-def.htm>
- Introduction to NTP: http://www.akadia.com/services/ntp_synchronize.html
- Exemplu de configurare complexa: http://docsrv.sco.com/NET_tcpip/ntpC.complete_scenarios.html
- <http://www.ijs.si/time/>