**Proiect Retele de Calculatoare**

**Tema:**

Aplicatie pentru transfer de fisiere – implementare printr-un mecanism de control al congestiei

**Realizat de:**

Anitoaei Teodor – 1305B

Stanciu Ioan – 1305B

**Protocolul TCP**

Protocolul TCP (Transmission Control Protocol) este unul dinte protocoalele nivelului Internet al modelului TCP/IPTCP a fost special conceput pentru a oferii un transportde biți de tip capăt-la-căpat, fără erori sau pierderi de date. El a fost definit în standardul RFC 793 (Request For Comment) scris în 1981.

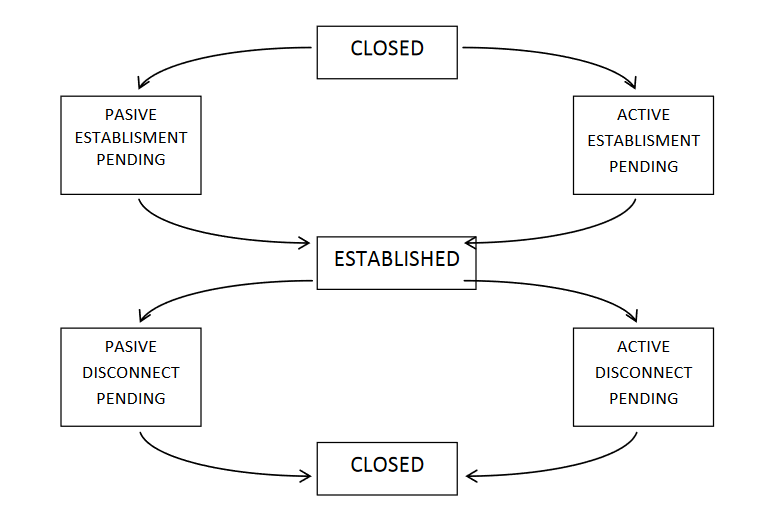
O rețea de internet diferă de una globală deoarece diferite păți pot avea topologii, lațimi de bandă, timpi de întârziere diferiți, dimeniuni ale pachetelor sau alți parametrii diferiți. Astfel TCP a fost astfel conceput să se adapteze la proprietățile diferite ale componentelor folosite în cadrul rețelei.

Protocolul TCP reprezintă unul din compnentele modelul TCP/IP (Protocol de control al transmisiei/ Protocol Internet) care a fost creat de US DoD (Departamentul de aparare al Statelor Unite) din necesitatea unei rețele care ar putea supraviețui în orice condiții. Scopul rețelei TCP/IP era ca orice conexiune s-ar rupe în interiorul rețelei rețeaua în ansamblu să rămână intacta

Protocolul TCP este un protocol orientat pe conexiune care permite ca un flux de octeți transmiși să ajunga la destinație ,fără erori, la orice altă mașină din cadrul aceleiași rețea.

Serviciul TCP trebuie asigurat atat de emitator cat si de receptor deoarece se bazează pe principiul capăt-la-capăt și folosește sockeți. Fiecarui socket îi corespunde un număr cu rol de adresă numit adresa IP adestinatarului si un alt numar alcatuit din 16 biti numit port. Pentru ca serviciul TCP să fie obținut trebuie realizată explicit o conexiune între mașina care transmite mesajul și cea care îl recepționează. Simplificat o conexiune pate fi reprezentatastfel:

Astfel TCP poate fi comparat cu o conexiune telefonica din punct de vedere al modului în care utilizatorul vede această conexiune. Diferența de baza este dată de faptul că mesajele sunt transmise prin intermediul pachetelor fără a bloca o linie telefoincă în cazul netransmiterii de informație.



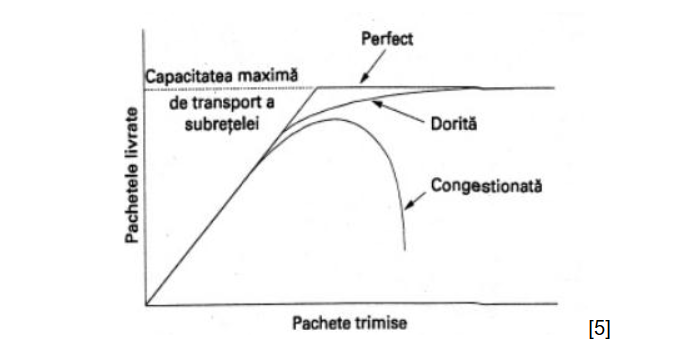
Astfel un anumit port se poate afla în diferite stări, cum ar fi IDLE, ESTABLISHED sau stari intermediare în cadrul cărora se realizează stabilirea conexiunii sau deconectarea. Conexiunea presupune 3 faze: realizarea conexiunii (IDLE->ESTABLISHED), transmiterea informației (starea ramane ESTABLISED) și intreruperea conexiunii (ESTABLISHED->IDLE).

Un socket poate fi folosit pentru mai multe conexiuni simultane. Astfel 2 sau mai multe conexiuni se pot termina la același socket. Acestea sunt identificate prin identificatori de socket aflati la ambele capete.

**Congestia retelei**

Congestia apare in reţelele de calculatoare atunci când incărcarea depăşeşte posibilităţile acesteia de a transfera datele. Incărcarea unei reţele la un moment dat nu depinde numai de capacitatea ei de transmitere ci şi de erorile care apar în mediul de transmisie, de viteza de prelucrare in noduri şi de mecanismele de confirmare folosite de receptor.

Controlul acestui fenomen intrǎ in responsabilitatea nivelelor de transport și de reţea. Apare ca rezultat al traficului intens la nivelul transport, propagȃndu-se la nivelul reţea, performanţele totale ale sistemului degradȃndu-se prin intȃrzierea și pierderea pachetelor de date.



Congestia apare ca rezultat al mai multor factori. In situaţia in care la sosirea unui numǎr mare de pachete provenind de pe mai multe linii de intrare intr-o singurǎ linie de ieșire, atunci se va forma o coada. Pentru a le pǎstra pe toate, sistemul necesitǎ memorie, iar creșterea acestei capacitǎţi de memorare duce la inrǎutǎţirea congestiei și nu la ameliorarea ei.

Un alt factor care cauzeazǎ congestia este reprezentat de viteza procesoarelor. Dacǎ unitatea centralǎ a ruterelor este lentǎ in execuţia funcţiilor sale, cozile pot crește. Și liniile cu lǎţime de bandǎ scǎzuta pot provoca congestia. Schimbarea liniilor cu unele mai performante și pǎstrarea aceluiași procesor sau invers ajutǎ puţin, deoarece problema ţine de incompatibilitatea intre pǎrţile sistemului și aceasta va persista pȃnǎla aducerea la echilibru a tuturor componentelor.

Controlul congestiei trebuie sǎ asigure capabilitatea reţelei de a transporta intreg traficul implicat. Este o problemǎ globala care implicǎ comportamentul tuturor calculatoarelor gazdǎ și al ruterelor.

**Algortimul BITCP**

Binary Increase TCP este un algoritm de control al congestiei care adapteazǎ controlul ferestrei in funcţie de dimensiunea acesteia. Acesta constǎ in douǎ pǎrţi: cǎutare prin creștere binarǎ (binary search increase) și creșterea aditivǎ a dimensiunii ferestrei (additive increase).

*Binary search increase*

In aceastǎ etapǎ privim controlul congestiei ca pe o problemǎ de cǎutare in care sistemul poate rǎspunde prin da sau nu atunci cȃnd vrem sǎ stim dacǎ rata de transmisiune curentǎ depǎșește sau nu capacitatea reţelei. Fereastra minimǎ curentǎ poate fi estimatǎ ca dimensiunea ferestrei atunci cȃnd nu avem pierderi.

Dacǎ se știe dimensiunea maximǎ a ferestrei, putem sǎ aplicǎm o tehnicǎ de cǎutare binarǎ pentru a seta dimensiunea ferestrei curente la dimensiunea medie dintre minimul estimat și maximul cunoscut. Dacǎ aceastǎ dimensiune dǎ pierderi,noua fereastrǎ poate fi tratatǎ ca un nou maxim, iar dimensiunea ferestrei dupǎ pierderea pachetelor devine noul minim. Media acestor valori devine noul optim.

Acest proces de recalculare a minimului și maximului are loc pȃnǎ cȃnd diferenţa dintre dimensiunea maximǎ și cea minimǎ scade sub un nivel prestabilit numit incrementare minimǎ Smin.

*Additive increase*

Pentru a asigura o convergenţǎ rapidǎ a algoritmului și corectitudinea, se combinǎ cǎutarea binarǎ cu o strategie de creștere aditivǎ. Cȃnd distanţa dintre dimensiunea medie și minimul curent este foarte mare, creșterea dimensiunii ferestrei direct la cea medie este un factor de risc in reţea. Astfel, cȃnd distanţa dintre dimensiunea ferestrei curente și cea medie este mai mare decȃt o valoare presetatǎ, numitǎ increment maxim Smax, in loc sǎ creștem dimensiunea ferestrei direct la acea valoare medie in urmǎtorul RTT, o sǎ o creștem cu Smax pȃnǎ cȃnd distanţa va deveni mai mica decat Smax, moment in care dimensiunea va fi setata direct la valoarea medie.

*Slow start*

Atunci cȃnd dimensiunea ferestrei curente devine mai mare decat valoarea maximǎ, nu cunoaștem noul maxim. In acest moment se va aplica tehnica de cǎutare binarǎ care va seta un maxim egal cu o valoare presetatǎ și dimensiunea ferestrei curente la o valoare minimǎ. Vom folosi o strategie de tip „slow start” pentru a cǎuta o valoare a maximului mai mica decȃt Smax. Dacǎ considerǎm cwnd dimensiunea curenta a ferestrei și incrementul maxim Smax, la fiecare RTT dimensinea ferestrei va crește astfel: cwnd+1, cwnd+2....cwnd+Smax. In acest mod se cautǎ banda disponibilǎ pȃnǎ cȃnd se asigurǎ faptul cǎ o creștere cu Smax este nedǎunǎtoare. Dupǎ aceastǎ etapa se va crește dimensiunea la fiecare RTT cu Smax.