Proiect Rețele de calculatoare

Transfer de fișiere – fereastră glisantă

Studenți: Tanasă Ciprian-Ionuț

Heghea Mihail-Cristian

Grupa: 1309A

În protocoalele anterioare, cadrele cu date erau transmise într-o singură direcţie. În cele mai multe situaţii practice, este necesar să se transmită date în ambele direcţii. O modalitate de a realiza transmisia de date full-duplex este de a avea două canale de comunicaţie separate, fiecare dintre ele fiind utilizat pentru traficul de date simplex (în direcţii diferite). Dacă se face aceasta, vom avea două circuite fizice separate, fiecare cu un canal "direct" (eng.: forward) pentru date şi un canal "invers" (eng.: reverse) pentru confirmări. În ambele cazuri lărgimea de bandă a canalului invers este irosită aproape în totalitate.

Următoarele trei protocoale sunt protocoale bidirecţionale care aparţin unei clase de protocoale numite protocoale cu fereastră glisantă (eng.: sliding window). Cele trei diferă între ele în termeni de eficienţă, complexitate şi necesar de tampoane, aşa cum vom vedea mai târziu. În cadrul acestora, ca în toate protocoalele cu fereastră glisantă, fiecare cadru expediat conţine un număr de secvenţă cuprins între 0 şi o valoare maximă. Maximul este de obicei 2n -1, ca numărul de secvenţă să se încadreze exact într-un câmp de n biţi. Protocoalele cu fereastră glisantă pas-cu-pas utilizează n=1, restricţionând numerele de secvenţă la 0 şi 1, dar versiuni mai sofisticate pot utiliza o valoare arbitrară a lui n.

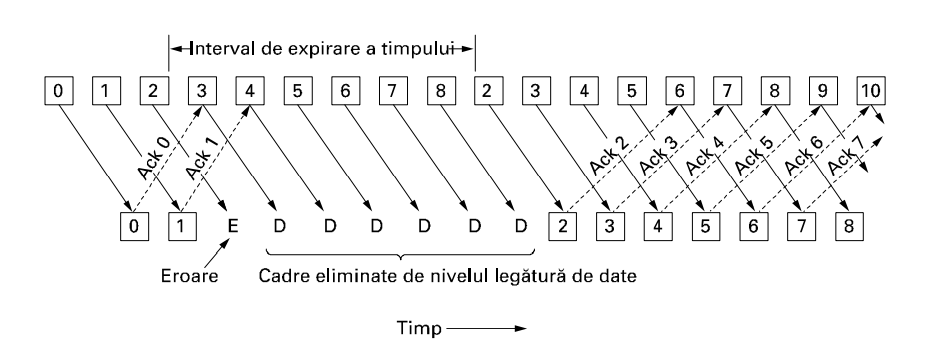
Esenţa protocoalelor cu fereastră glisantă este aceea că, la orice moment de timp, emiţătorul menţine o mulţime de numere de secvenţă care corespund cadrelor pe care are permisiunea să le trimită. Se spune că aceste cadre aparţin ferestrei de transmisie (eng.: sending window). Similar, receptorul menţine de asemenea o fereastră de recepţie (eng.: receiving window), ce corespunde mulţimii de cadre care pot fi acceptate. Fereastra emiţătorului şi fereastra receptorului nu trebuie să aibă aceleaşi limite minime şi maxime şi nici măcar aceeaşi dimensiune. În unele protocoale ele au dimensiune fixă, dar în altele ele pot creşte sau scădea pe măsură ce cadrele sunt emise sau recepţionate.

Câmpul de confirmare conţine numărul ultimului cadru recepţionat fără eroare. Dacă acest număr corespunde cu numărul de secvenţă al cadrului pe care emiţătorul încearcă să-l transmită, emiţătorul ştie că a terminat cu cadrul memorat în tampon şi poate prelua următorul pachet de la nivelul său reţea. Dacă numărul de secvenţă nu corespunde, el trebuie să continue să trimită acelaşi cadru. De fiecare dată când este recepţionat un cadru, un alt cadru este trimis de asemenea înapoi.

Exista doua moduri de baza de tratare a erorilor in prezenta benzii de asamblare. Un mod, numit revenire cu n pasi (go back n), este ca receptorul sa elimine pur si simplu cadrele care urmeaza, netrimitand confirmari pentru cadrele eliminate. Cealalta strategie generala de tratare a erorilor atunci cand este folosita banda de asamblare se numeste repetare selectiva (selective repeat). Cand aceasta este utilizata, un cadru incorect este respins, dar toate cadrele corecte care il urmeaza sunt memorate. Cand contorul de timp al emitatorului expira, cel mai vechi cadru neconfirmat este retransmis.

**Protocol de revenire cu n paşi (Go Back n)**

Până acum am făcut presupunerea tacită că timpul de transmisie necesar pentru ca un cadru să ajungă la receptor plus timpul de transmisie a confirmării este neglijabil. Uneori această presupunere este în mod cert falsă. Pentru a mări eficiența de utilizare a lățimii de bandă soluţia constă în a permite emiţătorului să transmită până la w cadre, în loc de unul singur. Cu o alegere potrivită a lui w emiţătorul va putea să transmită continuu cadre pentru un timp egal cu timpul de tranzit, fără a umple fereastra.

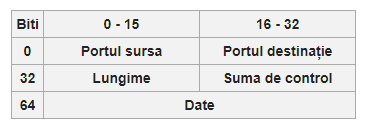
Mai jos este prezentat protocolul de revenire cu n paşi pentru cazul în care fereastra receptorului are dimensiune unu. Cadrele 0 şi 1 sunt primite şi confirmate corect. Cadrul 2, totuşi, este alterat sau pierdut. Emiţătorul, care nu ştie de această problemă, continuă să trimită cadre până când timpul pentru cadrul 2 expiră. Apoi se întoarce la cadrul 2 şi o ia de la început cu el, trimiţând din nou cadrele 2, 3, 4 etc.

**Protocolul Datagramelor Utilizator (UDP)**

User Datagram Protocol este un protocol de comunicație pentru calculatoare ce aparține nivelului Transport (nivelul 4 ) al modelului standard OSI.

Împreună cu Internet Protocol (IP), acesta face posibilă livrarea mesajelor într-o rețea. Spre deosebire de protocolul TCP, UDP constituie modul de comunicație fără conexiune. Este similar cu sistemul poștal, în sensul că pachetele de informații (corespondența) sunt trimise în general fără confirmare de primire, în speranța că ele vor ajunge, fără a exista o legătură efectivă între expeditor și destinatar.

Antetul UDP este alcătuit din 4 câmpuri fiecare având lungimea de 2 octeți.

* Portul sursa - în adresarea bazata pe IPv4 acest câmp este opțional. Daca nu este utilizat acest câmp, are valoarea zero; când reprezinta informație semnificativa, el va indica portul inițiator al procesului de transmisie a datagramelor.
* Portul destinație - spre deosebire de portul sursa, câmpul este obligatoriu și indica portul de recepție
* Lungime - acest câmp indica lungimea în octeți a datagramei: antet plus secțiune de date (valoarea minima a lungimii este 8). Limita teoretica a unui pachet UDP este de 64 KB.
* Suma de control - asigura imunitatea la erori; se calculează ca fiind complementul fata de 1 (pe 16 biți) a pseudo-antetului cu informații extrase din antetul IP, antetului UDP și a câmpului de date, eventual se completează cu zerouri pentru a atinge lungimea stabilita.

**Implementare**

Vom implementa două aplicații scrise în Python care să realizeze transferul de fișiere utilizând metoda Go Back N. Primul pachet trimis către receptor va informații despre transfer precum: numărul de octeți utilizați pentru reprezentarea numarului de pachet, numele fișierului, dimensiunea fișierului. Restul pechetelor vor conține un câmp prin intermediul căruia va fi determinat tipul pachetului (acknowledge, not-acknowledge sau pachet ce conține date), numărul pachetului și datele.