



Proiect Rețele de calculatoare

Transfer de fișiere-fereastră glisantă

Studenti: Dumea Cristian
Tarlev Mateo
Grupa:1308B

Ce este un protocol?

- Un protocol de comunicații este un set de reguli și norme care permite ca două sau mai multe entități dintr-un sistem de comunicații să comunice între ele prin transmiterea de informație printr-un mediu de orice tip prin variația unei mărimi fizice.

Ce ar trebui să facă un protocol?

- Protocele controlează comunicarea datelor
 1. Comprimarea datelor
 2. Cum este construită rețeaua
 3. Verificarea erorilor și manipularea acestora
 4. Conectarea datelor de transmisie.

Exemple de protocol de comunicare:

- I. Protocol Internet(IP)
- II. Protocol de control al transmisiei(TCP)
- III. Protocol de transfer de fișiere(FTP)
- IV. Protocol de transfer simplu prin poștă(SMTP)
- V. Protocol de transfer hipertext(HTTP)
- VI. Protocol simplu de gestionare a rețelei(SNMP)

Controlul fluxului - fereastră glisantă

Protocoalele de tip fereastră glisantă sunt protocoale ale nivelului de legătură de date pentru livrarea sigură a cadrelor de date.

În acest protocol mai multe cadre pot fi trimise de un expeditor simultan înainte de a primi confirmare de la receptor. Termenul de fereastră glisantă se referă la casetele imaginare ce au rolul de a reține cadrele. Metoda ferestrei glisante este cunoscută și sub numele de „fereastră”.

Informațiile trimise printr-o rețea de calculatoare nu sunt trimise într-o transmisie mare, ci mai degrabă în „pachete” mai mici. Pachetele sunt pur și simplu bucăți de informații pe care computerul care îl primește le reasambla pentru a construi produsul finit. Ideea unui puzzle tipic poate fi folosită pentru a ilustra ideea; atunci când puzzle-ul este cumpărat, piesele sunt împrăștiate și dezasamblate, dar pot fi reconstruite pentru a forma o imagine finală. Pachetele funcționează aproape în același mod.

Este necesar un protocol de fereastră glisantă atunci când computerul care primește trebuie să obțină pachetele într-o ordine sigură. Funcționează aproximativ ca un semafor care controlează o intersecție. Când lumina este verde, fereastra glisantă este deschisă, iar informațiile pot circula liber. După ce este trimis un ciclu de informații, „lumina” devine roșie și nu mai pot fi transmise date pentru moment. Odată ce computerul care primește pachetele, trimite înapoi un semnal (ACK) care confirmă că a primit informația; acesta este semnalul pentru a trimite mai multe date de-a lungul liniei.

Atât Protocolul Go-Back-N, cât și Protocolul de repetiție selectivă sunt tipuri de protocoale cu ferestre glisante. Principala diferență dintre aceste două protocoale este că, după găsirea suspectului sau a deteriorării în cadrele trimise, protocolul go-back-n retransmite toate cadrele, în timp ce protocolul de repetare selectivă

retransmite numai cadrul care este deteriorat. Acum, vom vedea diferența dintre ele:

Protocolul Go-Back-N	Protocolul de repetare selectivă
În Protocolul Go-Back-N, dacă cadrul trimis este suspectat, atunci toate cadrele sunt retransmise de la pachetul pierdut la ultimul pachet transmis.	În protocolul de repetare selectivă, sunt retransmise numai acele cadre care sunt considerate suspecte.
Dimensiunea ferestrei expeditorului a protocolului Go-Back-N este N.	Dimensiunea ferestrei expeditorului a protocolului Repeat selectiv este, de asemenea, N.
Dimensiunea ferestrei receptorului Go-Back-N Protocol este 1.	Dimensiunea ferestrei receptorului a protocolului de repetiție selectivă este N.
Protocolul Go-Back-N este mai puțin complex.	Protocolul de repetiție selectivă este mai complex.
În Protocolul Go-Back-N, nici expeditorul, nici destinatarul nu necesită sortare.	În protocolul de repetare selectivă, partea receptorului are nevoie de sortare pentru a sorta cadrele.
În Protocolul Go-Back-N, tipul de confirmare este cumulativ.	În protocolul de repetare selectivă, tipul de confirmare este individual.
În Protocolul Go-Back-N, pachetele în afara comenzii NU sunt acceptate (eliminate) și întreaga fereastră este retransmisă.	În protocolul de repetare selectivă, sunt acceptate pachetele necomandate.
În Protocolul Go-Back-N, dacă primește un pachet corupt, atunci întreaga fereastră este retransmisă.	În protocolul de repetare selectivă, dacă Primește un pachet corupt, acesta trimite imediat o confirmare negativă și, prin urmare, numai pachetul selectiv este retransmis.
Eficiența protocolului Go-Back-N este $N/(1+2*a)$	Eficiența protocolului de repetare selectivă este $N/(1+2*a)$

Protocolul Go Back N

Implementare transport

UDP este un antet fix și simplu de 8 octeți ce preia mesajul de la nivelul aplicație și formează un antet pe care îl va transmite nivelului network. Segmentul conține: portul sursă, portul destinație, lungimea (dimensiunea datagrammei) și suma de control (verifică doar integritatea segmentului).

Fi	0 - 15	16 - 32
0	Portul sursa	Portul destinație
32	Lungime	Checksum
64	Dată	

Portul sursa - în adresarea bazată pe IPv4 acest câmp este opțional. Dacă nu este utilizat acest câmp, are valoarea zero; când reprezintă informație semnificativă, el va indica portul inițiator al procesului de transmisie a datagramelor.

Portul destinație - spre deosebire de portul sursa, câmpul este obligatoriu și indică portul de recepție

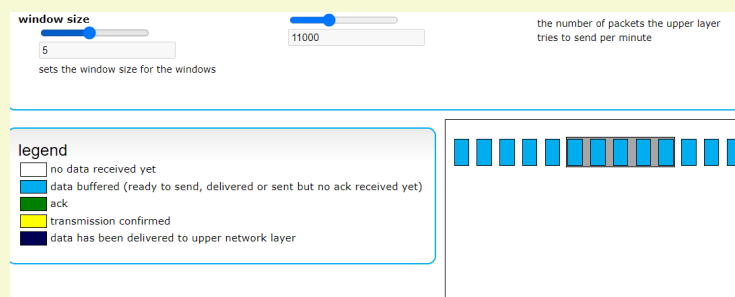
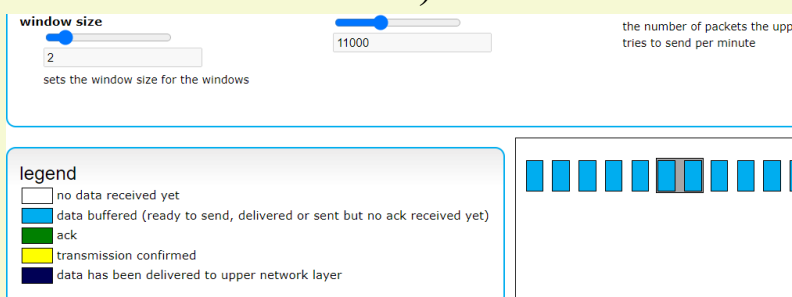
Lungime - acest câmp indică lungimea în octeți a datagrammei: antet plus secțiune de date (valoarea minimă a lungimii este 8).

Suma de control - asigură imunitatea la erori; se calculează ca fiind complementul față 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 stabilită.

Implementarea fluxului de date

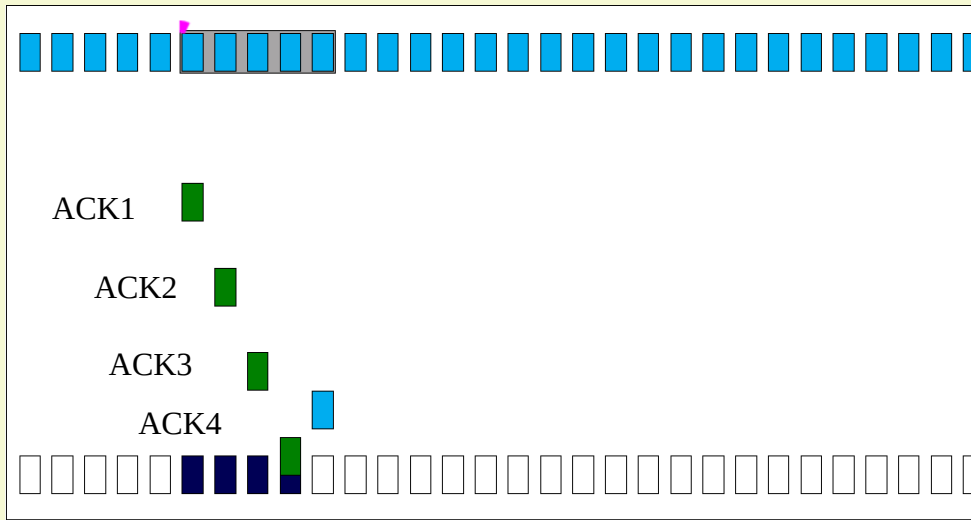
Implementarea ferestrei glisante se va face prin protocolul Go back N ARQ. ARQ se bazează pe: detecția erorilor (sumă de control), confirmarea primirii datelor printr-un ACK și retransmiterea dacă pachetul nu a fost primit corect.

Emitătorul va selecta dimensiunea ferestrei, notată N (numărul maxim de cadre).

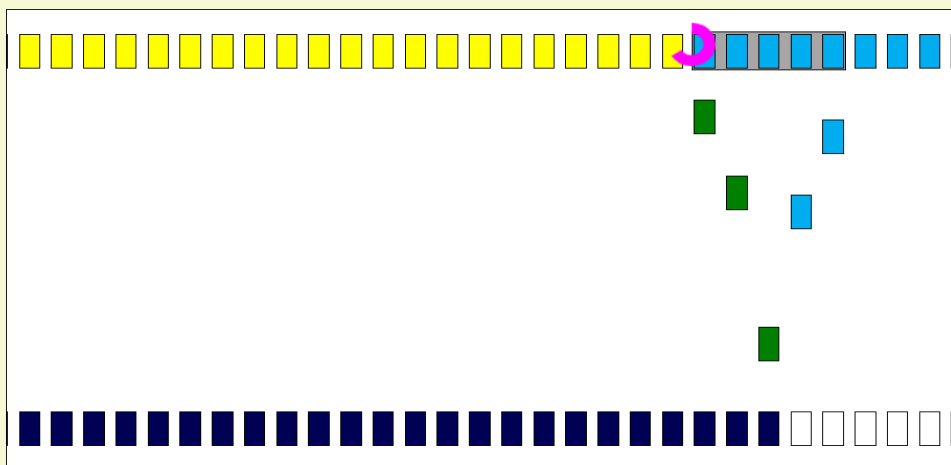


Receptorul nu are nevoie să rețină date inutile, el doar va reține numărul cadrului următor așteptat.

În primă fază emițătorul trimite pachetele către receptor așteptând confirmarea, ACK.

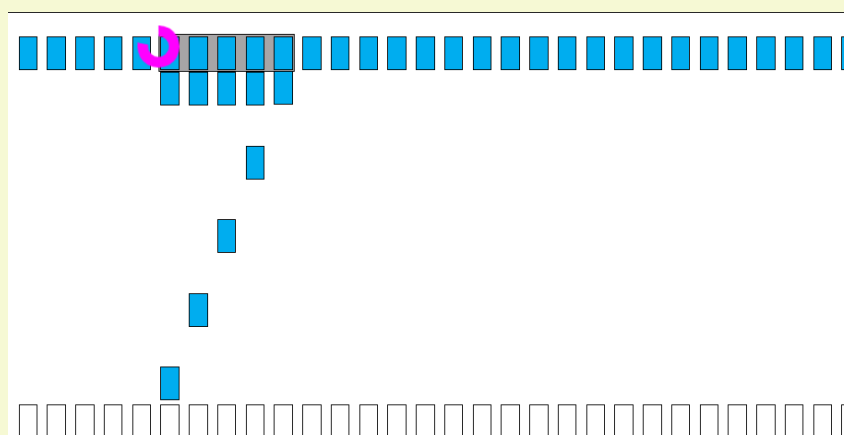


Odată ce un ACK este recepționat de către emițător, fereastra se glisează cu o poziție, trimițând următorul pachet.



Vom

introduce în timp limită de așteptare pentru fiecare pachet. În caz de pachetul nu va reuși să ajungă la emițător cu ACK în timpul estimat, va retransmite încă o dată acel pachet.



Implementare aplicație

Vom implementa două aplicații scrise în Python care să realizeze transferul de fișiere utilizând metoda Go Back N. Prima aplicație este realizarea serverului unde vom recepționa pachetele. A doua aplicație o reprezintă clientul care trimite pachetele către server. Clientul are 3 metode de trimitere printre care se numără:

- selectarea și citirea fișierului pe care dorește să îl trimită și determină numărul de pachete în care este împărțit fișierul
- trimiterea de informații serverului sub formă codificată, fiecare bit este ales în funcție de tipul fișierului, numărul de pachete și numele fișierului
- trimiterea de date către server care se realizează astfel: se trimite un număr de pachete egal cu dimensiunea ferestrei așteptând confirmare. Dacă timpul de așteptare s-a scurs și nu s-a primit confirmare, atunci se retrimite datele

Fiecare aplicație are o interfață proprie realizată în Python cu ajutorul librăriei Tkinter.

Bibliografie

1. https://en.wikipedia.org/wiki/User_Datagram_Protocol
2. <https://www.geeksforgeeks.org/difference-between-go-back-n-and-selective-repeat-protocol/>
3. https://www2.tkn.tu-berlin.de/teaching/rn/animations/gbn_sr/
4. <https://www.javatpoint.com/go-back-n-arq>