

Sprawozdanie z listy 4.

Arkadiusz Dębski

1. Problem zawarty w zadaniu.

Przesyłanie spójnych, kompletnych danych przez sieć, ze względu na złożoność i jakość tych sieci może być obarczone problemami z niezawodnym transferem pakietów na poziomie warstwy transportowej. Pakiety wielokrotnie przekazywane przez różne urządzenia np. bramy czy routery mogą być tracone lub dublowane. Również, ze względu na wiele możliwości routingu pakietów, kolejność ich dostarczenia może być inna niż kolejność nadawania.

2. Rozwiązanie za pomocą namiastki protokołu TCP.

W celu zaradzenia wyżej opisanym problemom możemy wprowadzić system numeracji pakietów, potwierdzania ich dostarczenia. Podstawowymi założeniami takiego systemu jest stosowanie numeracji pakietów przez nadawcę oraz potwierdzanie odebrania pakietów przez odbiorcę. Kiedy zostanie otrzymana informacja o otrzymaniu niepełnej puli pakietów lub gdy Klient nie otrzyma potwierdzenia ich otrzymania nadawca wysyła utracone pakiety ponownie, żeby upewnić się, że przesłane dane są kompletne. Duplikaty natomiast są ignorowane, a otrzymane pakiety są buforowane z rzeczywistą numeracją co zapewnia spójność i brak powtórzeń w danych. Implementacją takiego podejścia stosowaną na co dzień jest protokół warstwy transportowej sieci **TCP** (*Transmission Control Protocol*).

Segment TCP [\[edytuj\]](#) [\[edytuj kod\]](#)

TCP																																																									
Offset	Oktet	0								1								2								3																															
Oktet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																								
0	0	Port nadawcy																Port odbiorcy																																							
4	32	Numer sekwencyjny																																																							
8	64	Numer potwierdzenia (jeżeli flaga ACK jest ustawiona)																																																							
12	96	Długość nagłówka				Zarezerwowane				N S	C	E	U	A	P	R	S	F	Szerokość okna																																						
16	128	Suma kontrolna																Wskaźnik priorytetu (jeżeli flaga URG jest ustawiona)																																							
20	160	Opcje (jeżeli długość nagłówka > 5, to pole jest uzupełniane "0")																																																							
...																																																							

Do zadania w zupełności wystarczy użycie numeru sekwencyjnego określającego miejsce pakietu w transmisji oraz numeru potwierdzenia, który wysyła numer pakietu potwierdzonego.

3. Implementacja poprawek do kodu.

Przed edycją kodu źródłowego program **Z2Forwarder** przesyła zniekształcone dane, co powoduje, że wiadomość jest niemożliwa do odczytania. Jest to spowodowane problemami opisanymi w punkcie pierwszym.

Problem zostaje rozwiązany poprzez:

- Sprawdzanie kolejności otrzymywania pakietów oraz pomijanie duplikatów – po stronie odbiorcy.
- Potwierdzenia odebrania pakietów.
- Retransmisję i odczyt potwierdzeń po stronie nadawcy.

Przed wprowadzeniem zmian program **Z2Sender** otrzymywał komunikat o treści:

„Ala ma kota.
Ola nie ma kota.
Ela ma dwa koty.”

Natomiast program **Z2Receiver** otrzymywał następującą wiadomość (spacje zastąpione ‘_’, koniec linii ‘&’):

„la__Am_&
lan__tke__&
aOta_&
ak__o.l_Em__ky_d.wa_oyt(tutaj brak końca linii)”

Znając pierwotną wiadomość można zauważyć, że podczas transmisji niektóre ze znaków zostały pominięte, a niektóre zdublowane. Natomiast po wykonaniu poprawek w kodzie źródłowym wiadomość przesyłana jest zauważalnie dłużej, jednakże wiadomość zostaje przesłana w dobrej kolejności i jest taka sama jak oryginał.

4. Podsumowanie i wnioski.

Uproszczona wersja protokołu TCP spełnia zadanie spójności transmisji danych. Niestety wzrost czasu potrzebnego na transmisję sprawia, że protokoły podobne do TCP odbiegają od ideału. W niektórych przypadkach, gdzie nie jest wymagana stuprocentowa zgodność danych w porównaniu z oryginałem stosuje się różne inne protokoły jak chociażby UDP, który nie posiada tylu zabezpieczeń co TCP, ale także nie posiada tak dużego opóźnienia. Jest on często wykorzystywany przy transmisjach obrazu i/lub dźwięku (np. streaming) oraz grach sieciowych