Wykład 5
TCP

Sieci Komputerowe 2018

Własności TCP

Jest zorientowany połączeniowo. Zanim zostaną przesłane jakiekolwiek dane, musi zostać nawiązane połączenie

Zapewnia niezawodność (dane przesyłane przez aplikację są dzielone na tzw. segmenty, które wg. TCP mają najlepszy rozmiar, po wysłaniu segmentu jest uruchamiany zegar i rozpoczyna się oczekiwanie na potwierdzenie odebrania segmentu przez drugą stronę). W przypadku nieotrzymania potwierdzenia, segment jest wysyłany ponownie

Sortuje segmenty i w razie potrzeby odrzuca zdublowane

Własności TCP (c.d.)

Stosuje sumę kontrolną nagłówka i danych do kontroli poprawności. Jeśli zostanie wykryty błąd sumy kontrolnej potwierdzenie nie jest wysyłane.

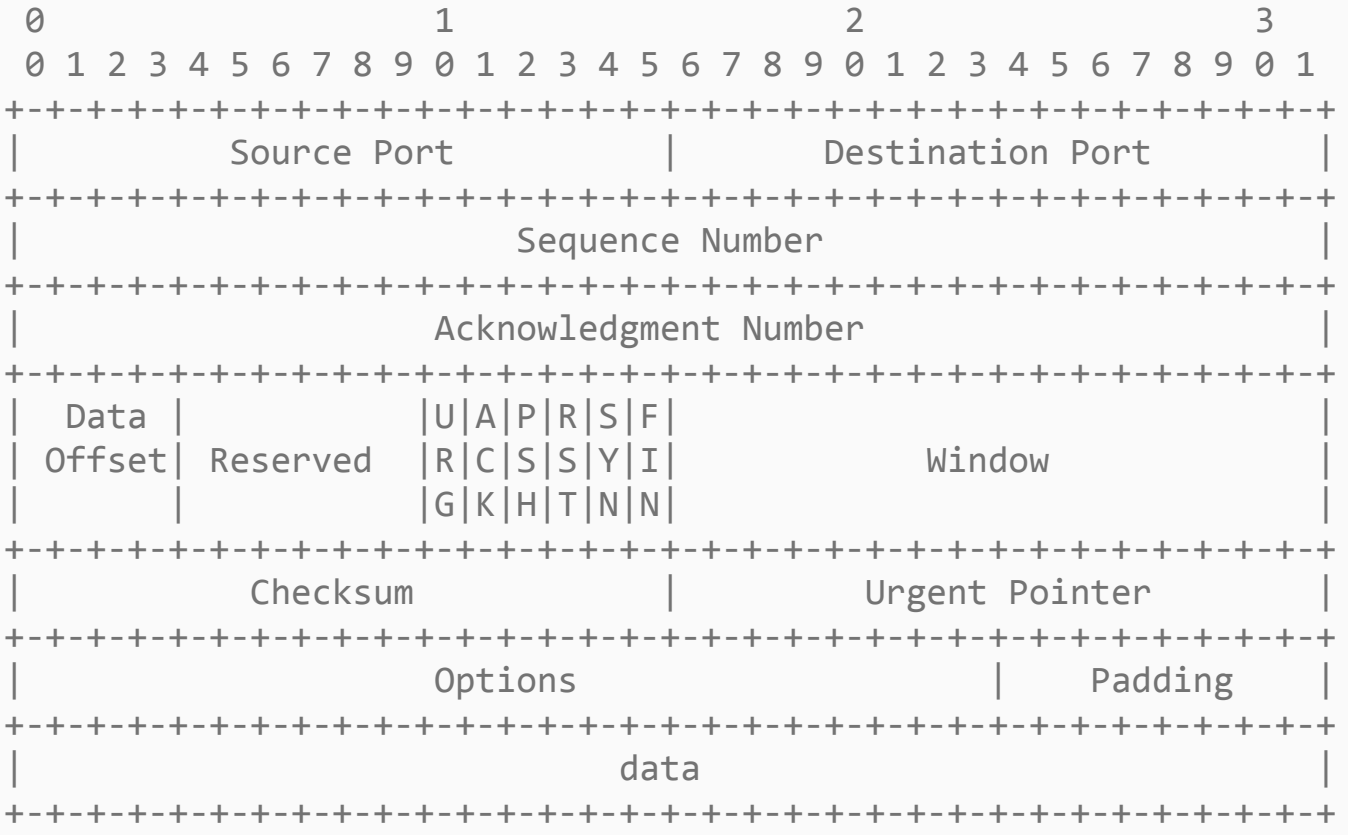
Ponieważ TCP wykorzystuje mechanizm połączeń, nie jest możliwe zastosowanie go do transmisji typu broadcast.

Umożliwia przesyłanie danych w obie strony (tzw. tryb full duplex).

Zapewnia kontrolę przepływu za pomocą mechanizmu okien.

W celu poprawy efektywności, stosuje się algorytmy poprawiające charakterystykę transmisji w często spotykanych scenariuszach.

Nagłówek TCP



Najważniejsze pola nagłówka

Numery sekwencyjny i potwierdzenia:

- Numer sekwencyjny służy do numerowania bajtów
- Numer potwierdzenia jest następnym spodziewanym numerem sekwencyjnym

Pole rozmiar okna służy do kontroli przepływu.

Pole Opcje – np. MSS (ang. Maximum Segment Size).

Pole znaczników – patrz następny slajd.

Pole znaczników

URG – znacznik ważności pola „wskaźnik pilności”

ACK – znacznik ważności pola „numer potwierdzenia”

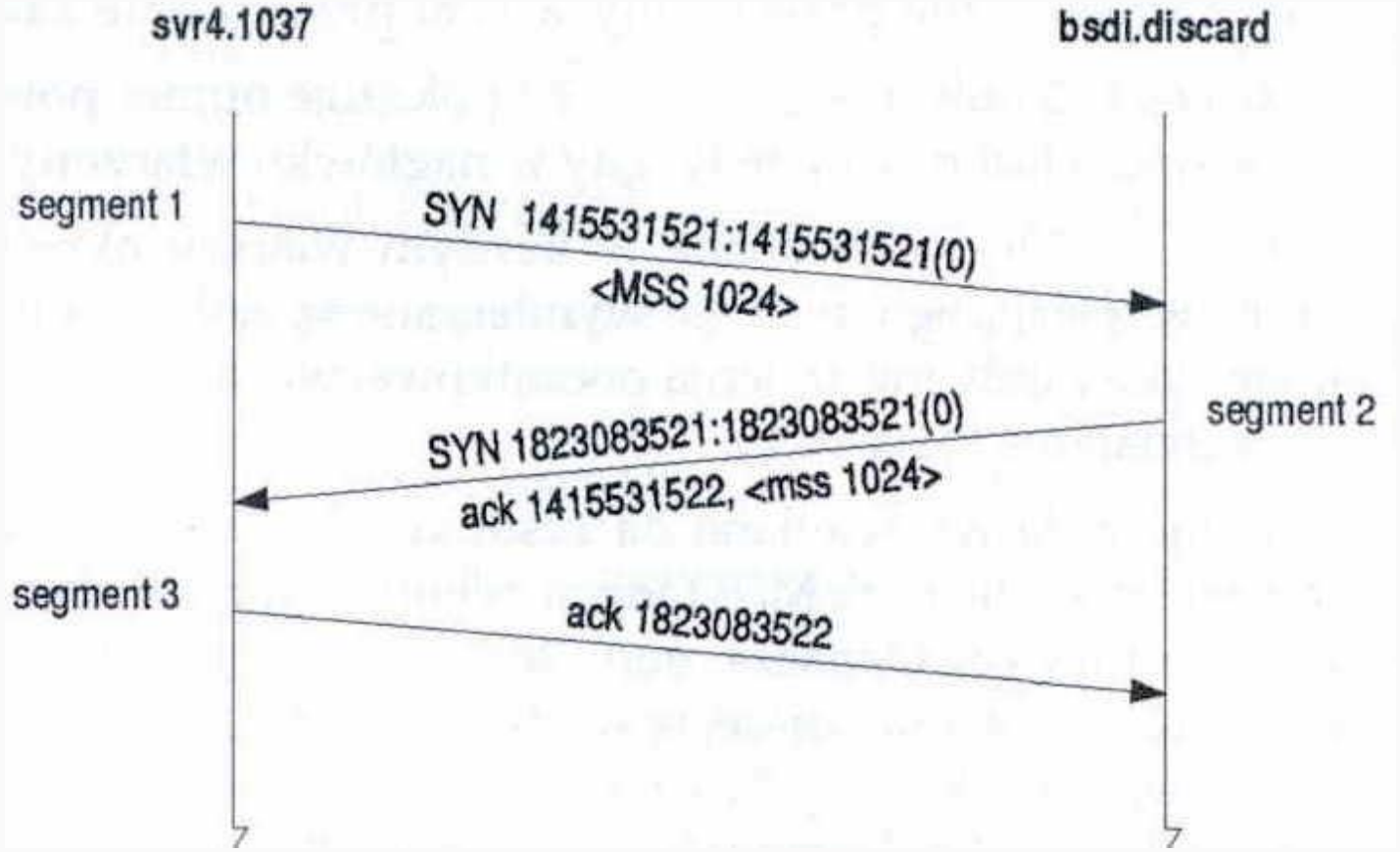
PSH – znacznik ten, jeśli ustawiony, oznacza, że odbiorca powinien przekazać dane do aplikacji tak szybko, jak to możliwe

RST – przerwanie połączenia

SYN – synchronizacja numerów sekwencyjnych w celu inicjalizacji połączenia

FIN – nadawca zakończył wysyłanie danych

Nawiązywanie połączenia



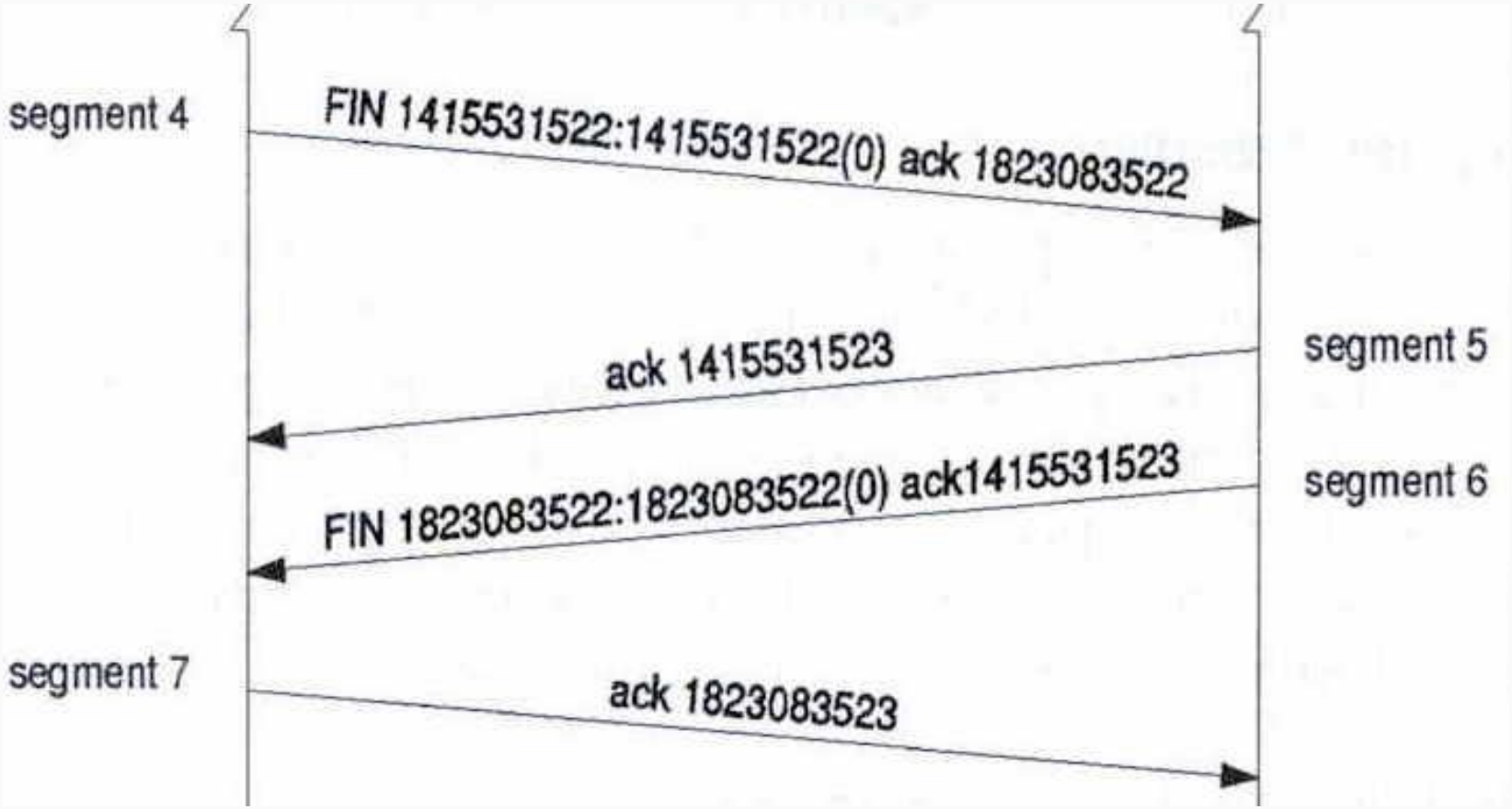
Nawiązywanie połączenia

Proces nawiązywania połączenia polega na synchronizacji numerów sekwencyjnych (trójstopniowy handshake), tak aby było wiadomo jak numerować bajty.

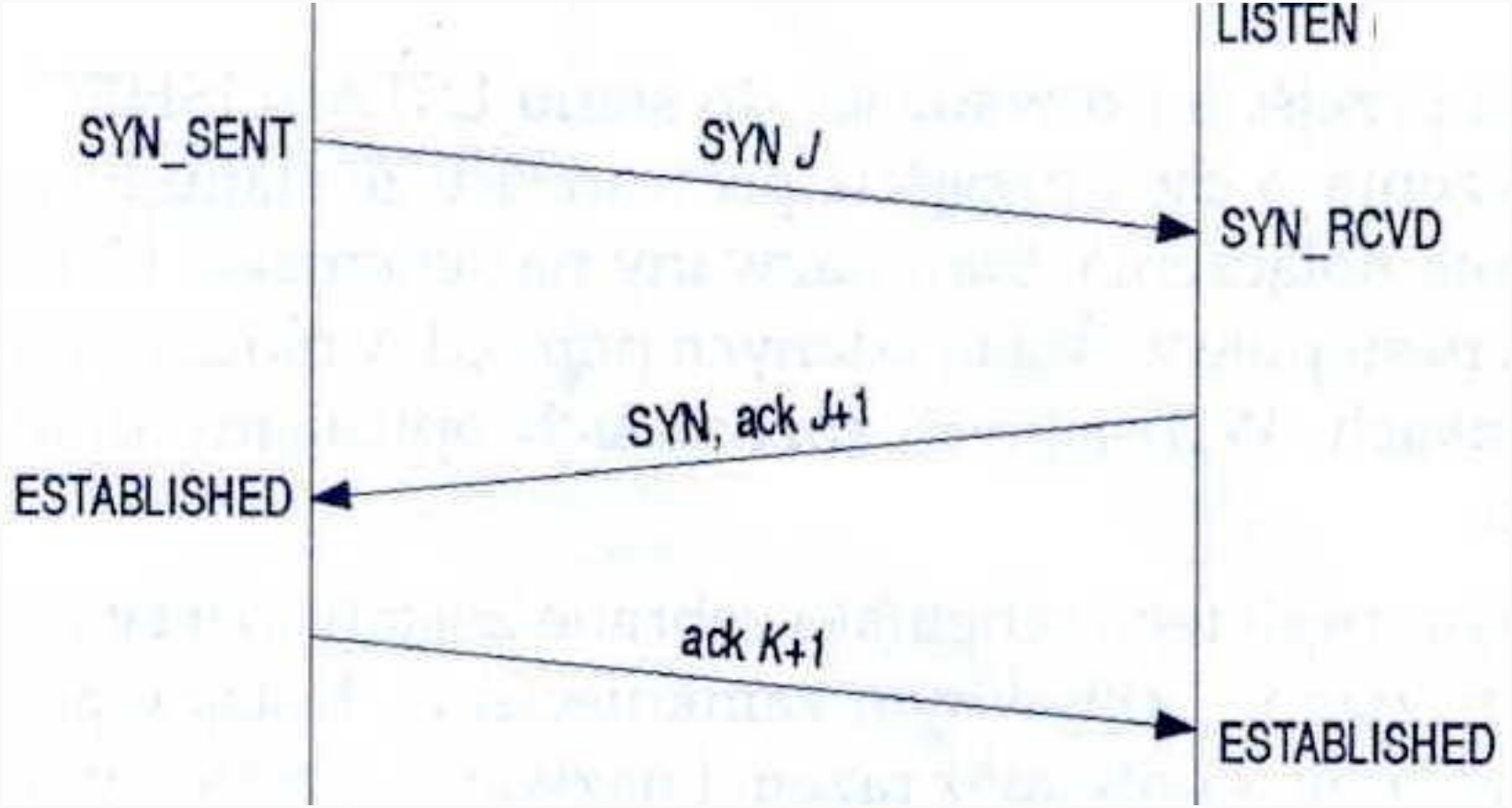
Po nawiązania połączenia możliwa jest komunikacja full-duplex.

Istotny jest wybór początkowego numeru sekwencyjnego (numeru ISN), tak aby był unikalny dla danego połączenia (dba o to implementacja TCP).

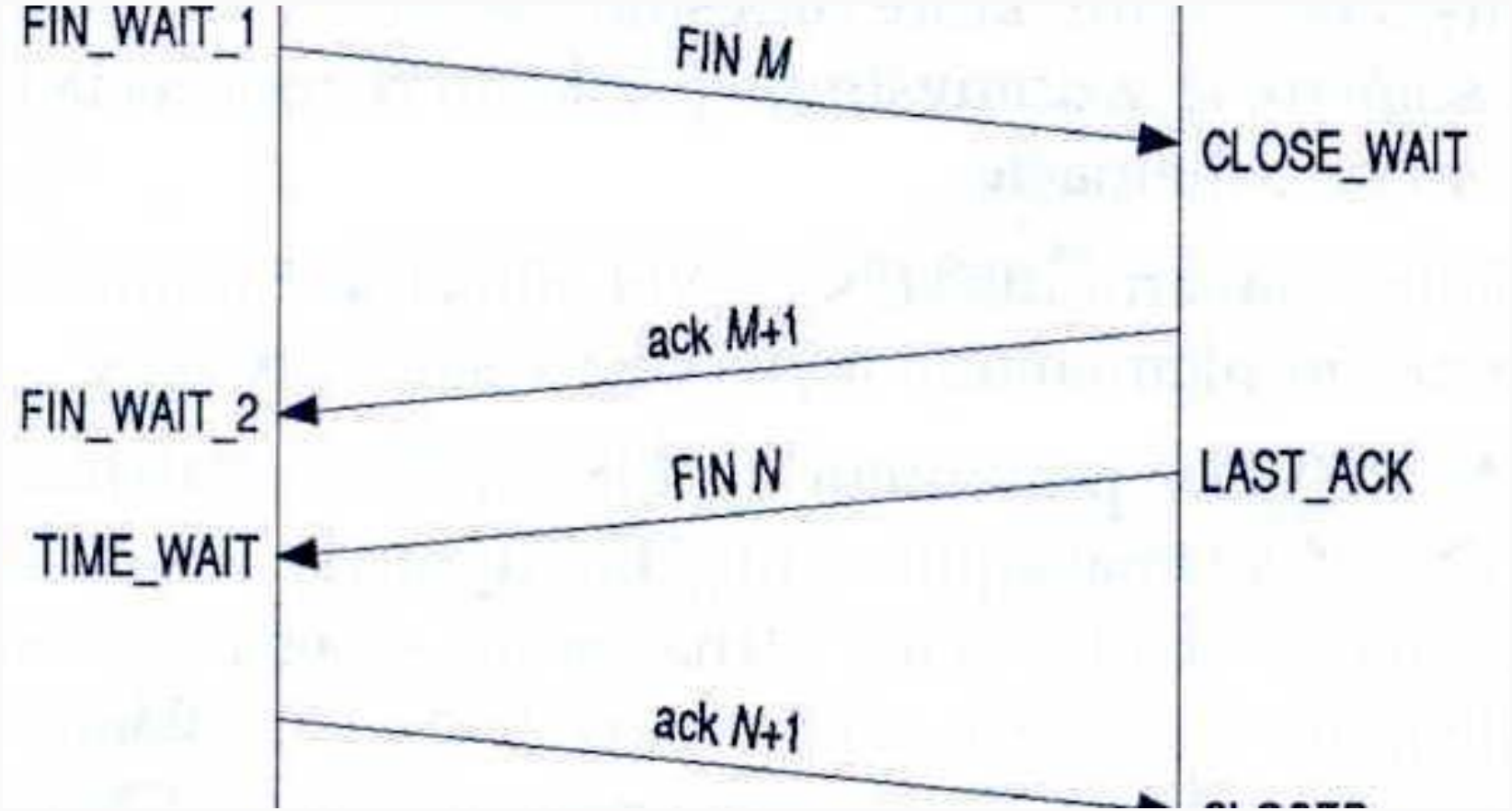
Kończenie połączenia



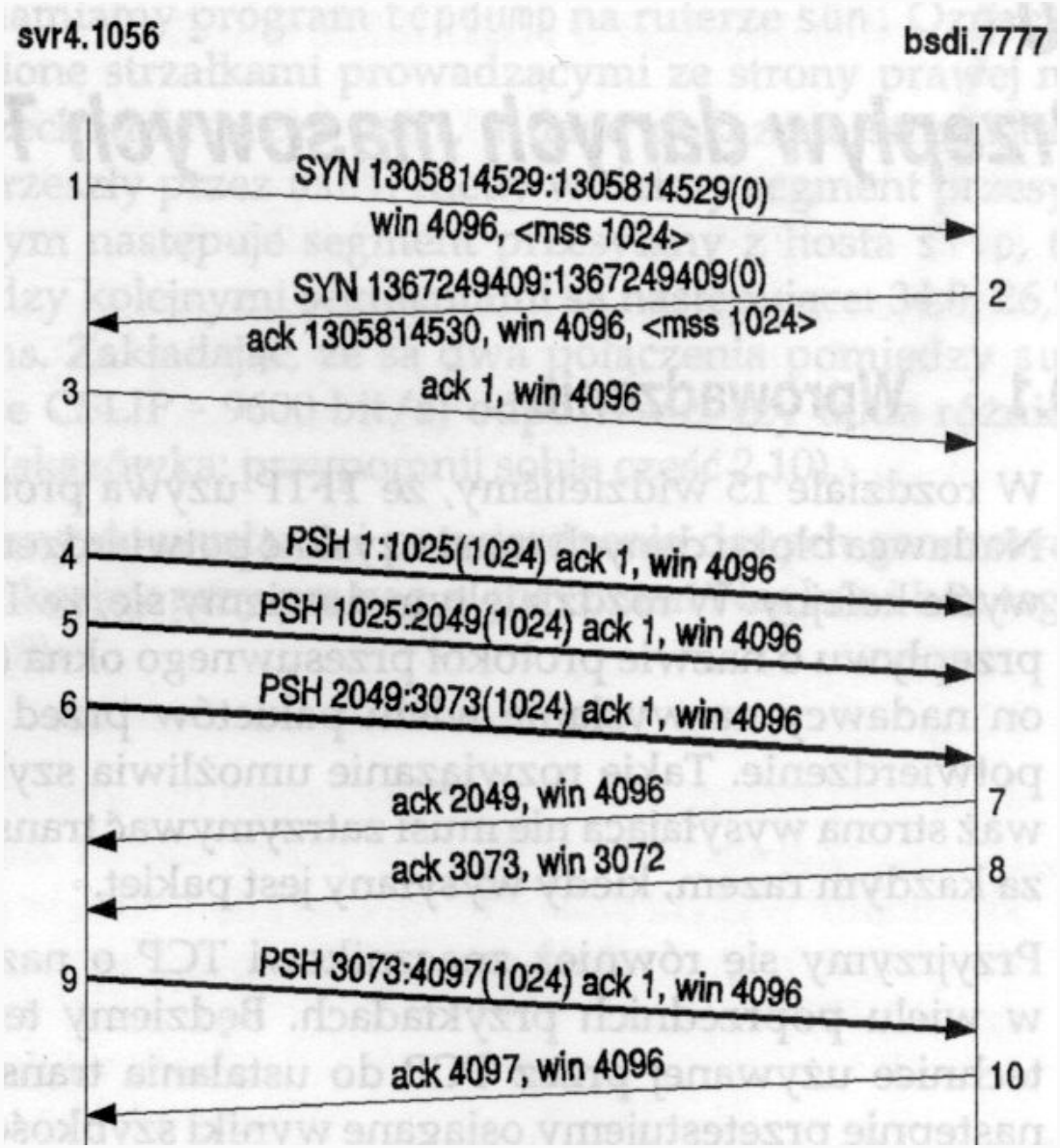
Stany TCP


LISTEN
SYN_SENT
SYN J
SYN_RCVD
SYN, ack J+1
ESTABLISHED
ack K+1
ESTABLISHED

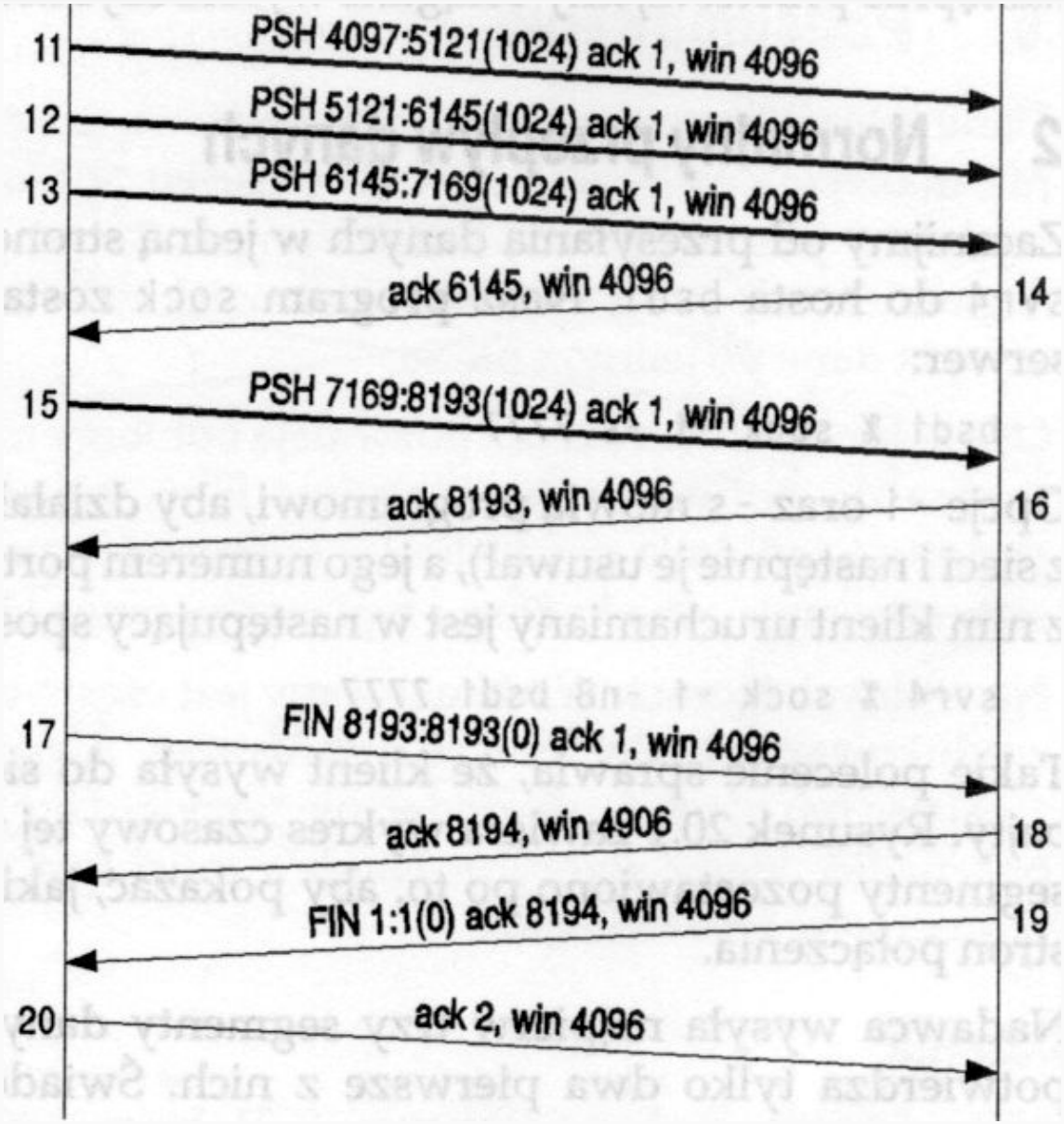
Stany TCP


FIN_WAIT_1
FIN M
CLOSE_WAIT
ack M+1
FIN_WAIT_2
FIN N
LAST_ACK
TIME_WAIT
ack N+1

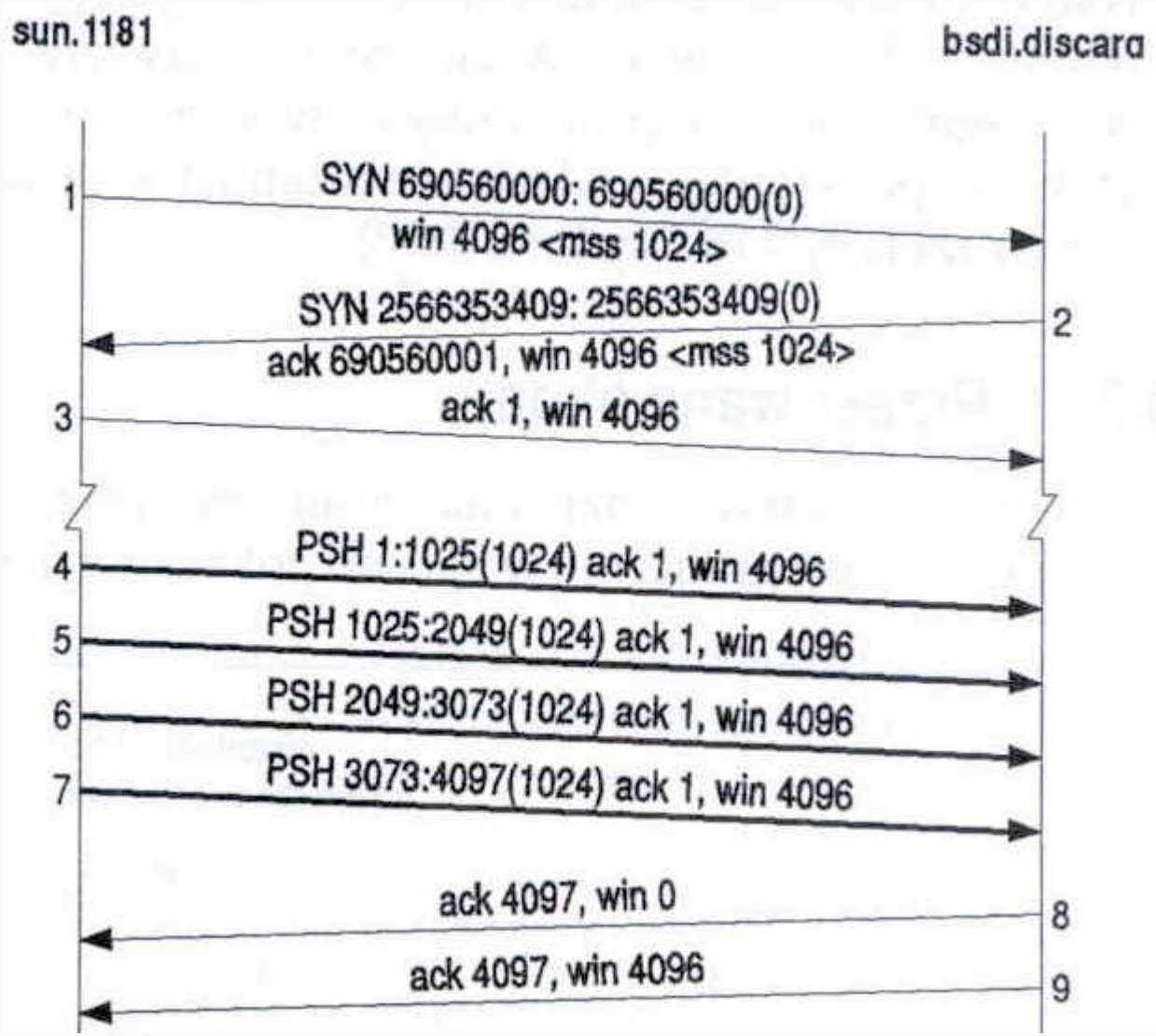
Przepływ danych masowych



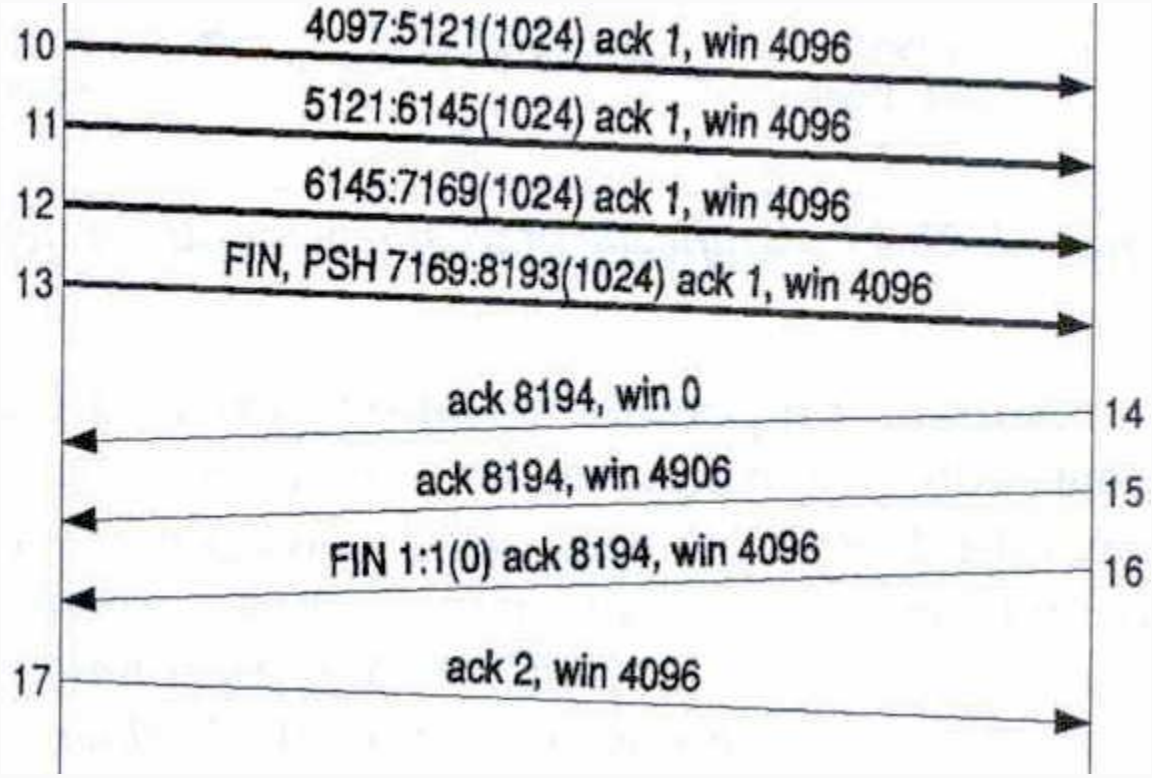
Przepływ danych masowych



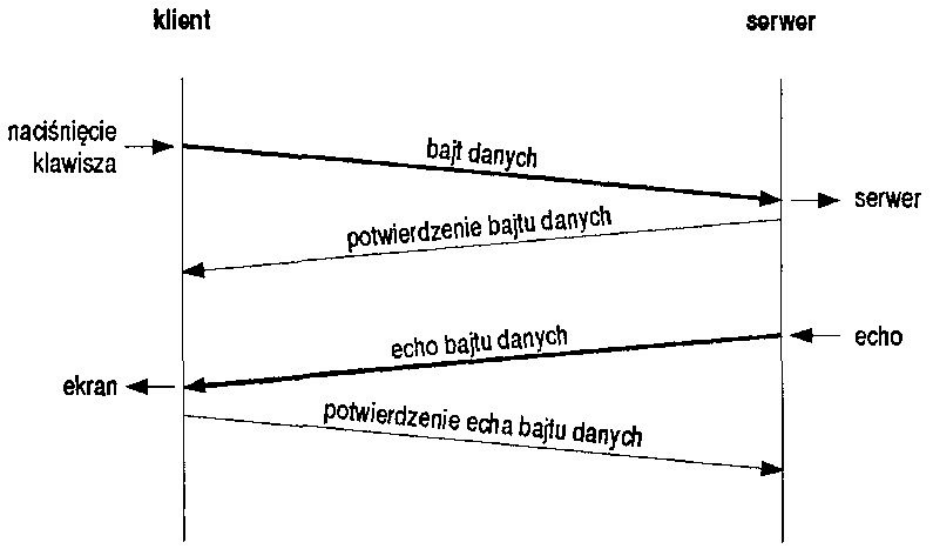
Szybki nadawca, wolny odbiorca



Szybki nadawca, wolny odbiorca



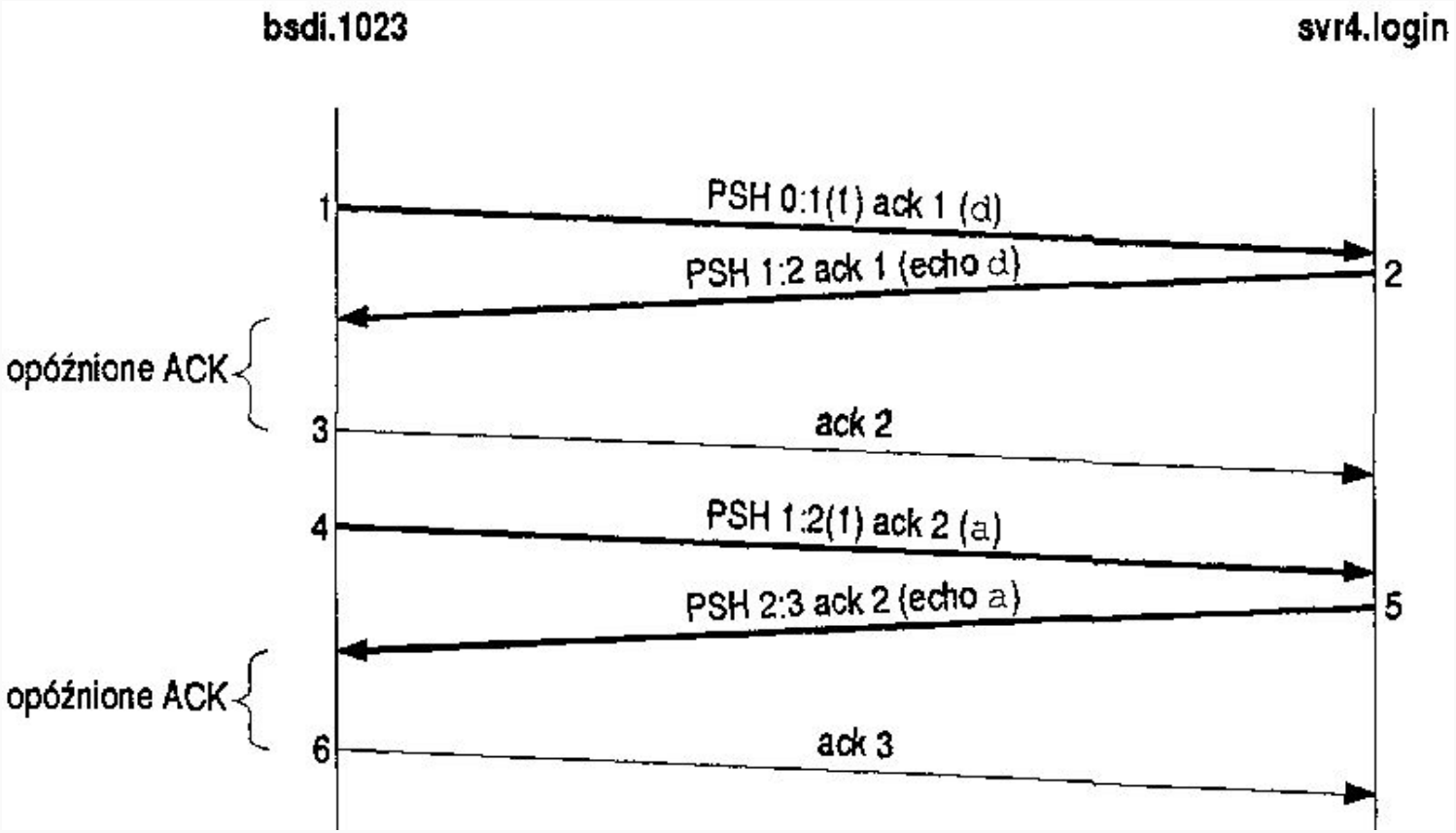
Przepływ danych interaktywnych



Przesyłane są pojedyncze znaki, pakiet IP dla przesłania 1 znaku ma długość 41 bajtów!

Dla jednego przesłanego znaku tworzone są zwykle 4 segmenty.

Opóźnione potwierdzenia w przesyłaniu interaktywnym



Algorytm Nagle’a

Służy do minimalizacji liczby wysyłanych segmentów

- Jeśli nadawca miałby wysłać dwa niepełne segmenty TCP pod rząd, to przed wysłaniem drugiego oczekuje na ACK pierwszego
- Może więc występować wstrzymanie wysyłania danych, tak, aby później wysłać więcej w jednym segmencie
- Istnieje możliwość wyłączenia algorytmu Nagle'a, za pomocą odpowiedniej opcji API gniazd

Retransmisja

Retransmisja następuje w momencie, gdy TCP nie otrzymał potwierdzenia dla któregoś z segmentów.

Konieczne jest wyznaczenie czasu oczekiwania, po którym ma nastąpić retransmisja.

Aby wyznaczyć czas oczekiwania (RTO, Retransmission Timeout),
TCP musi mierzyć czas podróży segmentów (RTT, Round Trip Time)

- Czas podróży: $R = \alpha R_p + (1 - \alpha)M, \ \alpha=0.9$
- Czas oczekiwania: $RTO = R\beta, \ \beta=2$

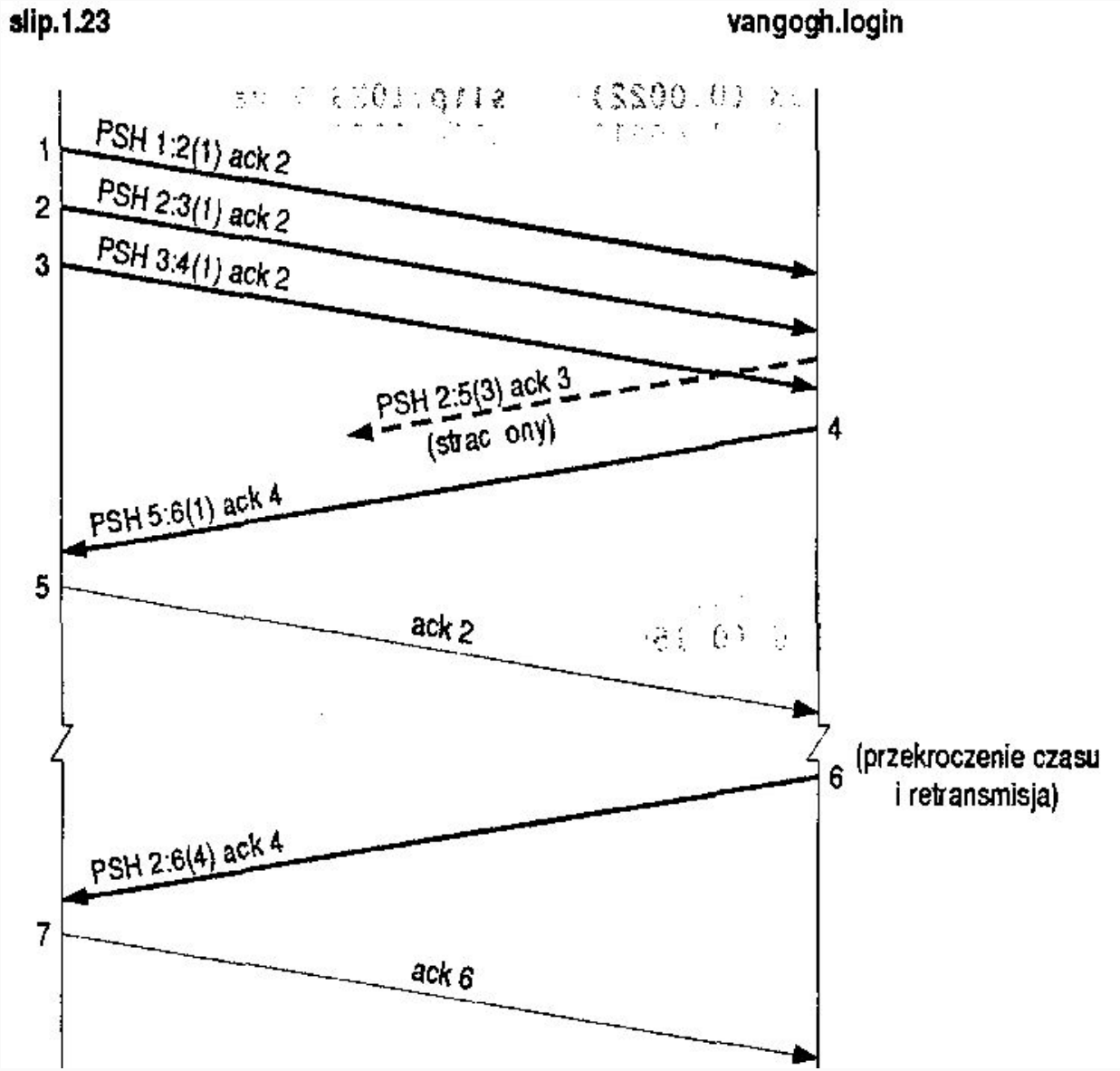
Szybka retransmisja

Jeśli odbiorca otrzyma segment, który nie jest kolejnym przez niego oczekiwanym, ma prawo bezzwłocznie wysłać segment ACK.

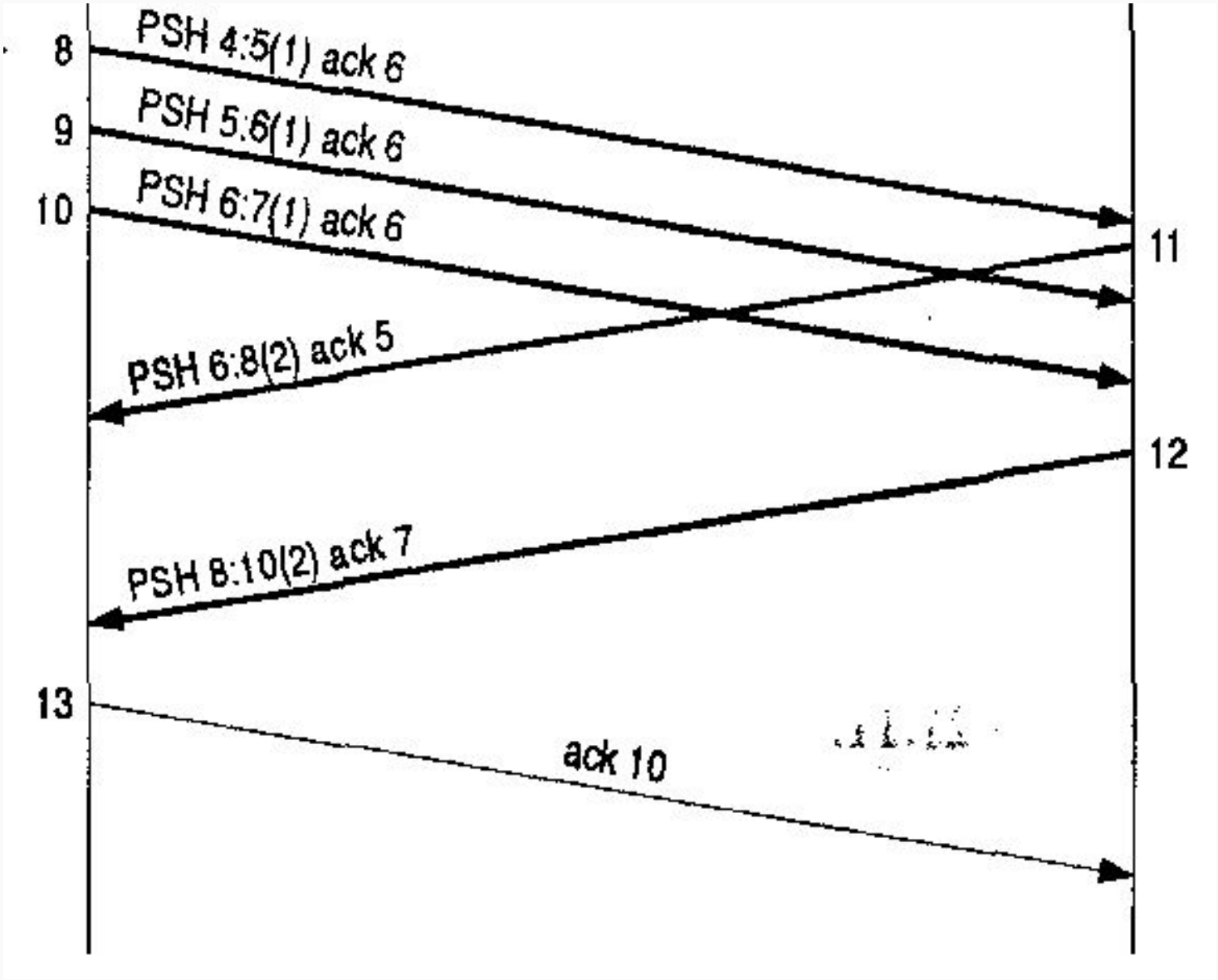
Taki zduplikowany segment ACK jest dla pierwotnego nadawcy wskazówką, iż któryś z wcześniej wysłanych segmentów nie dotarł.

Szybka retransmisja, bez oczekiwania na upłynięcie RTO, jest wykonywana w przypadku otrzymania trzech takich samych zduplikowanych ACK pod rząd.

Retransmisja



Retransmisja



Repakietyzacja

Po zakończeniu odliczania czasu oczekiwania następuje retransmisja, ale TCP nie musi retransmitować identycznych segmentów.

Może wykonać repakietyzację, czyli wysłać segment większy.

Algorytm powolnego startu

Służy do kontroli przepływu związanego z obciążeniem sieci
(inaczej niż w przypadku ogłaszania wielkości okna).

Problem jest wykrywany, gdy pojawiają się straty segmentów.

Początkowo transmisja nie wypełnia okna ogłaszanego przez odbiorcę.
Liczba wysyłanych segmentów bez potwierdzenia zaczyna się od 1
i jest stopniowo zwiększana wraz z otrzymywaniem kolejnych ACK.

W tym celu dla każdego połączenia utrzymywana jest zmienna *cwnd*
(ang. congestion window) – okno przeciążenia – w następujący sposób:

- wartość cwnd jest ustawiona na początku na rozmiar jednego segmentu (MSS) i stopniowo zwiększana
- wysyłana liczba bajtów nie może przekroczyć wartości cwnd i wartości okna ogłaszanego przez odbiorcę
- za każdym potwierdzonym segmentem, wielkość cwnd wzrasta o MSS, powoduje to wykładniczy wzrost liczby wysyłanych segmentów (1,2,4,8...)

Algorytm powolnego startu – zapobieganie zatorom

Dodatkowo wprowadzamy dla połączenia zmienną *ssthresh* (próg powolnego startu),

Gdy wystąpi zator (wykryty poprzez przekroczenie czasu):

- *ssthresh* := *cwnd* / 2
- *cwnd* := MSS (powolny start)

Gdy *cwnd* >= *ssthresh*, zaczyna działać algorytm zapobiegania zatorom, który aktualizuje wartość *cwnd*, utrzymując ją możliwie dużą, jednakże nie powodującą częstej utraty segmentów.

Kilka konkretnych algorytmów:

- TCP Tahoe, Reno, Vegas
- BIC (Linux Kernel 2.6.8)
- CUBIC (Linux Kernel 2.6.19)

Łącza o dużych przepustowościach

Pojemność potoku można zdefiniować jako:

pojemność (bity) = szerokość pasma (b/s) * czas podróży (s)

Potok powinien być wypełniony, aby uzyskać oczekiwaną przepustowość.

Ćwiczenie: Jakie jest potrzebne minimalne okno, aby wypełnić potok 10Gbps o czasie podróży 1ms?

Opcja skalowania okna

Jak widzieliśmy, okno opisane liczbą 16 bitową może być zbyt małe.

Stosowana jest opcja TCP skalowania okna

- Wartość jednobajtowa *scale*, między 0 a 14.
- Rozmiar okna = wartość w nagłówku * 2^{scale}

Opcja skalowania może się pojawiać jedynie w segmentach SYN, więc jest stała dla danego połączenia.

Jeśli strona, która wykonuje aktywne otwarcie umieści tę opcję w segmencie SYN, ale nie dostanie jej w segmencie SYN przesłanym przez drugą stronę, to opcja ta nie może być używana, gdyż druga strona jej nie obsługuje.

Demo

Dziękuję

Za tydzień kilka słów
o programowaniu usług
sieciowych.

Szymon Acedański
WMIM UW
accek@mimuw.edu.pl