

Zadania na temat działania protokołu TCP

Założmy, że numer strumienia danych (losowo generowany Initial Sequence Number wysyłany w segmencie z flagą SYN) wysyłanych przez stronę A połączenia TCP jest równy 5000 i że strona A wysłała do strony B 5 segmentów z polem danych o jednakowej długości 1460 B. Przedstawić je na rysunku podając ich numery sekwencyjne. W jaki sposób strona B potwierdzi ich odbiór? Podaj wartość odpowiedniej flagi i pola w nagłówku TCP segmentu z potwierdzeniem.

5001	6461	7921	9381	10841
-----	-----	-----	-----	-----

Strona B potwierdzi odbiór pierwszych 5 segmentów wysyłając w polu AckNo numer sekwencyjny 6 segmentu wynoszący $10841 + 1460 = 12301$

Flaga ACK=1

Pole AckNo=12301

Założmy że strona B wysłała do strony A segment z numerem potwierdzenia (AckNo) równym 12301 i wartością 6000 w polu Window. Ile segmentów (począwszy od pierwszego niepotwierdzonego) z polem danych 1460 B może wysłać A do B przed odebraniem kolejnego potwierdzenia od B? Przedstawić je na rysunku podając ich numery sekwencyjne (SeqNo).

Liczba segmentów = $\lfloor \text{Window} / \text{Długość segmentu} \rfloor = \lfloor 6000 / 1460 \rfloor = 4$

Kolejne segmenty w oknie:

12301	13761	15221	16681
-----	-----	-----	-----

Podstawowym wzorem używanym w powyższych zadaniach jest

$\text{SeqNo}(n+1) = \text{SeqNo}(n) + \text{Liczba_Bajtów}(n)$

Dodawanie własnej aplikacji sieciowej do uruchamianych przez xinetd

1. Na serwerze do katalogu /etc/xinetd.d dodać plik o nazwie np. rozmowa i następującej zawartości:

```
service mytalk
{
    disable = no
    user = root
    wait = no
    socket_type = stream
    server = /usr/bin/tail
    server_args = -f /tmp/tekst
}
```

mytalk jest nazwą usługi, którą trzeba powiązać z numerem portu jej serwera w pliku /etc/services;

Uwaga: nazwa pliku (rozmowa) nie musi być taka sama jak nazwa usługi (mytalk)

2. Na serwerze do pliku /etc/services dopisać na końcu linię:

```
mytalk 65535/tcp
```

np. wydać polecenie `echo mytalk 65535/tcp >> /etc/services`

Uwaga 1: Użyć podwójnego znaku większości! Użycie pojedynczego spowoduje usunięcie dotychczasowej zawartości pliku!

Uwaga 2: Nazwa usługi (mytalk) musi być taka sama jak w jej pliku konfiguracyjnym w katalogu /etc/xinetd.d

3. Na serwerze wydać polecenia:

```
systemctl stop iptables
```

```
systemctl reload xinetd
```

```
tee -a /tmp/tekst 1>/dev/null
```

polecenie tee przepisuje standardowe wejście (deskryptor 0)

do standardowego wyjścia (deskryptor 1)

i dodatkowo do pliku określonego opcją -a

4. Na kliencie (Linux) wydać polecenie:

```
telnet <adres serwera> 65535
```

albo na kliencie (Windows) w oknie PuTTY wpisać IP lub nazwę serwera, Port: 65535

oraz Connection type: Raw

Po wykonaniu tych czynności linia tekstu wpisanego z klawiatury na serwerze

zostanie, po wciśnięciu Enter, zapisana do pliku /tmp/tekst.

Usługa mytalk, korzystając z tego, że polecenie tail z opcją -f /tmp/tekst prezentuje na bieżąco zmiany w pliku /tmp/tekst, wyśle do klienta dopisaną do tego pliku linię tekstu.

Używając aplikacji Wireshark przeanalizuj segmenty przesyłane między klientem a serwerem usługi mytalk. Zastosuj w tym celu następujący filtr przechwytywania:

```
host <IP_klienta> and host <IP_serwera> and tcp port 65535
```

Zaobserwuj jak przebiega etap zamknięcia połączenia kiedy zamyka je klient (Windows – zamknięcie okna, Linux – wciśnięcie Ctrl-J) oraz kiedy zamyka je serwer (systemctl stop xinetd).