

Instrukcja laboratoryjna z przedmiotu: Sieci komputerowe

Ćwiczenie 7: Warstwa transportowa. Działanie protokołu TCP i UDP

Marta Szarmach
Zakład Telekomunikacji Morskiej
Wydział Elektryczny
Uniwersytet Morski w Gdyni

04.2022

I. Wprowadzenie

Warstwa 4. modelu OSI — transportowa — ma za zadanie utworzyć całościowe połączenie pomiędzy usługami na konkretnych urządzeniach. Do rozpoznawania konkretnej usługi na danym urządzeniu używane są numery **portów sieciowych** (np. usługa HTTP służąca do przesyłania zawartości stron internetowych identyfikowana jest przez port o numerze 80, a **gniazdo** 172.16.1.1:80 oznacza usługę HTTP na komputerze o adresie IP 172.16.1.1).

Na warstwie transportowej wyróżniamy 2 najpopularniejsze protokoły TCP oraz UDP.

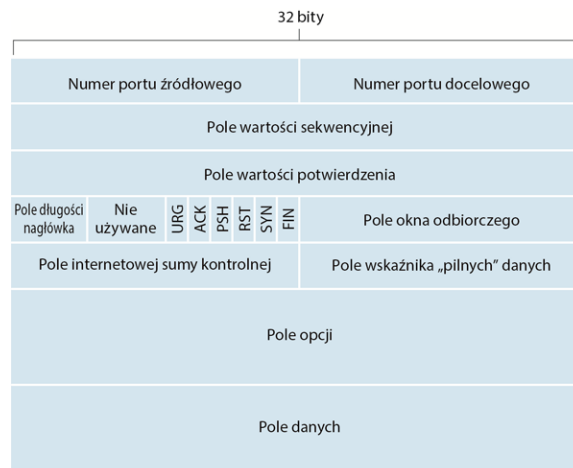
TCP (ang. *Transmission Control Protocol*)

TCP jest protokołem połączeniowym (tworzy sesję pomiędzy komunikującymi się urządzeniami) i nadzoruje ruch w ramach sesji: stosuje mechanikę potwierdzeń (ACK), aby być pewnym, że wysłane dane zostały poprawnie odebrane (i we właściwej kolejności), w razie potrzeby dokonuje retransmisji utraconych danych, umożliwia też sterowanie przepływem, tj. regulację ilości przesyłanych danych w czasie, aby nie doszło do przeciążenia. TCP najczęściej przenosi informacje pochodzące z usług nieakceptujących błędów w transmisji: przesył zawartości stron internetowych (HTTP), poczty elektronicznej (POP3, IMAP, SMTP), plików (FTP), telnet oraz SSH.

TCP umieszcza w nagłówku dane umożliwiające sterowanie sesją i identyfikację usług na komunikujących się urządzeniach:

- **Port źródłowy** — identyfikator usługi (aplikacji) źródłowej,

- **Port docelowy** — identyfikator usługi (aplikacji) docelowej,
- **Numer sekwencyjny** — numer identyfikujący konkretny segment w całym strumieniu danych
- **Numer potwierdzenia** — numer używany do potwierdzenia odebrania segmentu; informuje, którego bajtu danych oczekuje teraz odbiorca,
- **Flagi** — wskazują cel segmentu, przykładowo:
 - ACK — segment stanowi potwierdzenie odebrania porcji danych,
 - SYN — sygnalizuje chęć rozpoczęcia sesji i synchronizuje numery sekwencyjne w jej ramach,
 - FIN — sygnalizuje chęć zakończenia sesji.
- **Rozmiar okna odbiorczego** — zawiera informacje o tym, ile bajtów odbiorca jest w stanie przyjąć, wykorzystywane przy sterowaniu przepływem.

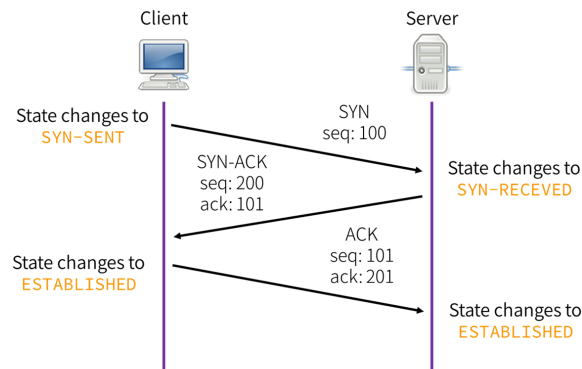


Grafika: strefainzyniera.pl

Aby móc utworzyć sesję, serwer oczekuje na chętnych do otwarcia połączenia w trybie LISTEN. Rozpoczynanie sesji realizowane jest w 3 krokach (tzw. *three way handshaking*):

1. Stacja wyrażająca chęć nawiązania sesji (klient) wysyła do serwera segment TCP z flagą SYN.
Klient w stanie SYN-SENT
2. Jeśli serwer godzi się na rozpoczęcie sesji, odsyła segment z flagą SYN i ACK. Oczekuje na potwierdzenie od klienta.
Serwer w stanie SYN-RECEIVED

3. Klient odsyła ACK. Sesja zostaje ustanowiona.
Klient i serwer w stanie ESTABLISHED



Grafika: python.astrotech.io

W trakcie trwania sesji komunikacja jest nadzorowana: stacja odbierająca potwierdza odebranie tych danych, wysyłając segment z ustawioną flagą ACK z numerem potwierdzenia informującym, którego bajtu danych teraz spodziewa się odebrać. Jeśli stacja wysyłająca nie otrzyma potwierdzenia w określonym czasie, dokonuje retransmisji tego fragmentu danych.

Po wyczerpaniu się danych do wysłania, stacje przechodzą 4 kroki, aby zakończyć sesję:

1. Stacja inicjująca zakończenie sesji wysyła segment TCP z ustawioną flagą FIN.

Stacja inicjująca w stanie FIN-WAIT1

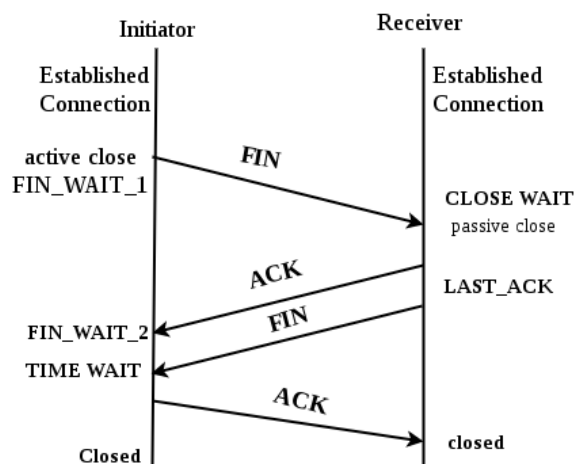
2. Stacja odbierająca odpowiada na FIN, wysyłając ACK.

Stacja odbierająca w stanie CLOSE-WAIT, stacja inicjująca w FIN-WAIT2

3. Stacja odbierająca potwierdza chęć zakończenia sesji, wysyłając swój własny FIN.

Stacja odbierająca w stanie LAST-ACK, stacja inicjująca w TIME-WAIT.

4. Stacja inicjująca potwierdza (ACK) otrzymanie FIN od drugiej strony. Sesja jest zakończona po upływie czasu trwania stanu TIME-WAIT (max. 4 min).



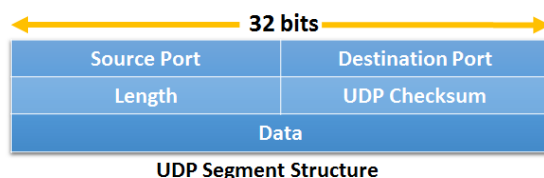
Grafika: geeksforgeeks.org

UDP (ang. *User Datagram Protocol*)

UDP zapewnia dostarczenie danych do konkretnej usługi, lecz nie tworzy sesji w trakcie komunikacji, nie zapewnia niezawodności transmisji (nie obejmuje mechanizmu potwierdzeń i retransmisji) ani kontroli nad przepływem, lecz dzięki temu nie obciąża łącza dodatkowymi informacjami kontrolno-sterującymi. UDP sprawdza się:

- tam, gdzie ważniejsza jest szybkość transmisji, a utrata części danych może zostać nawet niezauważona przez użytkownika (VoIP, streaming),
- w komunikacji typu zapytanie-odpowiedź, gdzie brak odpowiedzi samoczynnie odbierane jest jako zachęta do retransmisji (np. DNS, DHCP),
- tam, gdzie zapewnienie potwierdzenia odbioru jest zaimplementowane na wyższych warstwach (TFTP).

Nagłówek UDP też jest uproszczony w stosunku do nagłówka TCP: zawiera jedynie informacje o docelowej i źródłowej usłudze (numery portów), sumę kontrolną służącą do sprawdzenia poprawności danych w nagłówku oraz informację o jego długości.



Grafika: ipwithease.com

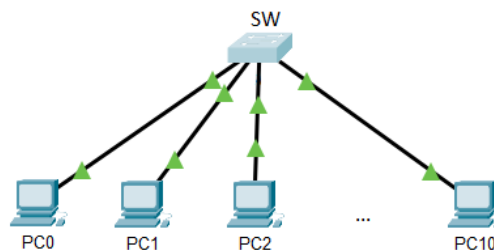
II. Cel ćwiczenia

Celem niniejszego ćwiczenia jest zapoznanie się z funkcjonowaniem warstwy transportowej modelu OSI w sieciach komputerowych, to znaczy działaniem protokołów TCP i UDP, poprzez:

- przechwycenie w programie Wireshark ruchu TCP i zaobserwowanie procesu rozpoczynania sesji, potwierdzania danych i kończenia sesji,
- przechwycenie w programie Wireshark ruchu UDP i zaobserwowanie różnic w działaniu protokołów TCP i UDP,
- użycie komendy *netstat* do analizy połączeń aktualnie otwartych na komputerze.

III. Stanowisko laboratoryjne

Do wykonania ćwiczenia niezbędne jest stanowisko laboratoryjne składające się z komputerów klasy PC z zainstalowanym systemem Windows oraz oprogramowaniem Wireshark, połączonych w sieć za pomocą przełącznika sieciowego Cisco.



Przed przystąpieniem do ćwiczenia:

- Włącz komputer do lokalnej sieci laboratoryjnej, uruchamiając na nim kartę sieciową o nazwie *LAB*. Kliknij *Start* ⇒ *Ustawienia* ⇒ *Połączenia sieciowe*. Prawym klawiszem wybierz kartę sieciową *LAB* i kliknij *Włącz*, podobnie wybierz kartę sieciową *Internet* i wybierz *Wyłącz* (od tego momentu komputer straci połączenie z internetem na rzecz sieci laboratoryjnej).
- Ustaw statycznie adres IP według schematu:
IP: 172.16.1.*numer_Twojego_stanowiska*
Maska podsieci: 255.255.255.0

IV. Przebieg ćwiczenia

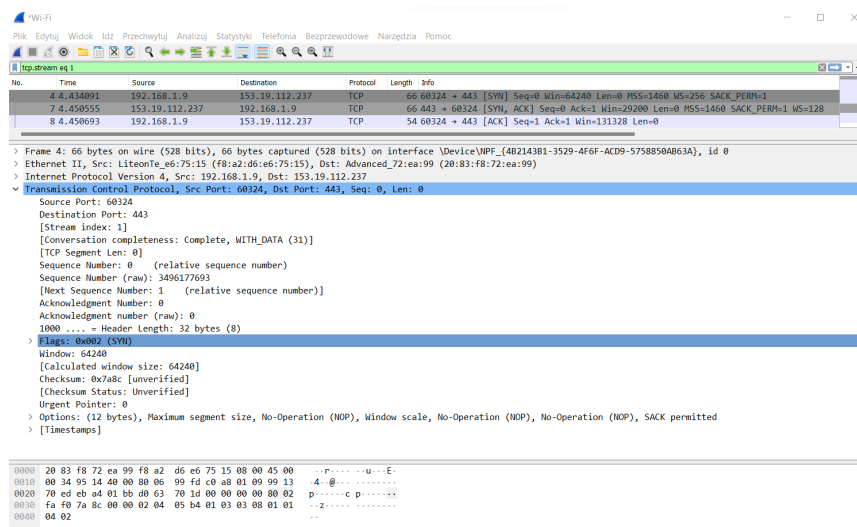
1 Analiza ruchu TCP

1.1 Przechwyć ruch TCP wygenerowany podczas otwierania strony internetowej.

- a) Otwórz program Wireshark i rozpocznij przechwytywanie ruchu sieciowego na karcie sieciowej Realtek. Włącz filtrowanie przechwyconego ruchu, tak, by widoczne były tylko segmenty TCP.
- b) Wygeneruj ruch sieciowy: poproś sąsiada o adres IP jego komputera i wpisz go w pasku adresu w przeglądarce internetowej. Przejdiesz w ten sposób na prostą stronę WWW, która jest przechowywana na komputerze sąsiada: **It works!** (przypomnij sobie, że na każdym z komputerów w laboratorium zainstalowane jest oprogramowanie Apache, dzięki któremu każdy z komputerów staje się serwerem WWW).
- c) Sprawdź, czy w programie Wireshark pojawiły się przechwycone segmenty. Jako że do przekazywania ruchu HTTP wykorzystywany jest protokół TCP na warstwie transportowej, powinien być widoczny cały przebieg sesji — od jej rozpoczęcia po zamknięcie. Zatrzymaj przechwytywanie danych.

1.2 Przyjrzyj się budowie nagłówka segmentu TCP.

- a) Zaznacz pierwszy przechwycony segment (zawierającą wysłany od Ciebie segment SYN — upewnij się, kontrolując źródłowy adres IP w nagłówku protokołu warstwy trzeciej) i rozwiń w środkowej części okna czwartą sekcję — odpowiadającą nagłówkowi protokołu TCP.
- b) Przyjrzyj się niektórym polom nagłówka:
 - Port docelowy — powinien wskazywać na to, że próbowałeś komunikować się z usługą HTTP na komputerze sąsiada (wartość 80),
 - Port źródłowy — powinien zawierać dużą wartość (port prywatny),
 - Numer sekwencyjny — jako że jest to pierwszy segment w ramach sesji, prawdopodobnie przyjmuje wartość 0,
 - Numer potwierdzenia — prawdopodobnie przyjmuje wartość 0, gdyż dany segment nie ma ustawionej flagi ACK,
 - Spośród wszystkich flag ustawiona jest flaga SYN, oznaczająca, że dany segment jest zaproszeniem do nawiązania sesji,
 - Rozmiar okna odbiorczego — może przyjmować dużą wartość, gdyż jest to dopiero początek sesji i bufor odbiorczy może być jeszcze w miarę pusty.



1.3 Zaobserwuj proces nawiązywania sesji TCP i potwierdzenia odbioru fragmentu danych.

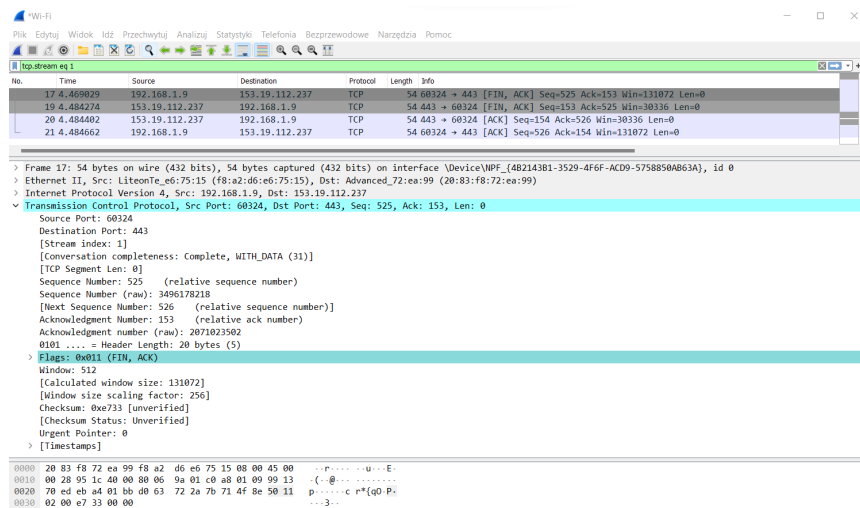
a) Przyjrzałeś się już pierwszemu segmentowi TCP w ramach sesji — wysłanemu przez Ciebie segmentowi SYN, skierowanego od Ciebie (inicjatora) do serwera WWW. Kliknij zatem na drugi segment — wysłany przez serwer SYN+ACK, stanowiący kolejny krok w *three way handshake*.

- Na warstwie 3. powinienś zauważyć, że nadawcą jest komputer sąsiada, a odbiorcą — Twój komputer.
- W nagłówku warstwy 4. zaobserwuj, że porty docelowe i źródłowe zamieniły się — teraz źródłowym portem jest port 80 (segment pochodzi z usługi HTTP), a docelowym Twój prywatny port, otwarty wyłącznie na potrzeby tej jednej transmisji.
- Spójrz na ustawione flagi — tym razem ustawione są flagi SYN (serwer wyraża chęć na nawiązanie sesji z Twoim komputerem) oraz ACK (segment jest jednocześnie potwierdzeniem otrzymania przez serwer poprzedniego segmentu SYN).
- Numer sekwencyjny znów ma wartość 0 — jest to pierwszy segment wysłany przez serwer w ramach tej sesji.
- Numer potwierdzenia przyjmuje wartość 1 — serwer otrzymał „zera” segment i informuje, że oczekuje na przyjęcie „pierwszego”.
- Rozmiar okna odbiorczego mógł zostać zmodyfikowany przez serwer, w zależności od jego obciążenia.

b) Przyjrzyj się segmentowi TCP stanowiącemu trzeci — ostatni krok nawiązywania sesji. Twój komputer, zgadzając się na nawiązanie sesji, odpowiedział serwerowi WWW segmentem ACK.

- Na warstwie 3. źródłowym adresem jest adres IP Twojego komputera, docelowym — adres IP komputera sąsiada.
- Na warstwie 4. źródłowym portem jest Twój prywatny (wysoki) port, docelowym — port 80 (HTTP).
- Numer sekwencyjny — jako że jest to kolejny wysłany przez Twój komputer segment TCP w ramach tej sesji, przyjmuje wartość o 1 większą niż ostatnio (a więc 1).
- Numer potwierdzenia — ten segment stanowi potwierdzenie na SYN+ACK wysłane przez serwer, a więc Twój komputer informuje, że oczekuje bajtu nr 1.
- Jedyna ustawiona flaga to flaga ACK.

1.4 Zaobserwuj proces zamykania sesji TCP.



- Pierwszym z segmentów sygnalizujących chęć zakończenia sesji jest wysłany przez Twój komputer segment z ustawioną flagą FIN — zlokalizuj go na liście przechwyconych segmentów. Być może ustawiona jest na nim też flaga ACK, co oznacza, że jest potwierdzeniem otrzymania wcześniejszego segmentu w ramach tej sesji.
- Znajdź segment będący potwierdzeniem wysłanym przez serwer na Twoją chęć zakończenia sesji. Ustawioną ma tylko flagę ACK. Zwróć uwagę, że numer potwierdzenia jest o 1 większy niż numer sekwencyjny potwierdzanego segmentu FIN.
- Znajdź segment z ustawioną flagą FIN wysłany przez serwer, potwierdzający jego chęć zakończenia sesji.

- d) Znajdź segment będący potwierdzeniem wysłanym przez Twój komputer na segment FIN otrzymany od serwera. Porównaj numer potwierdzenia w tym segmencie oraz numer sekwencyjny potwierdzanego segmentu FIN. Po otrzymaniu przez serwer tego segmentu ACK oraz odczekania przez Twój komputer pewnego czasu w stanie TIME-WAIT, sesja TCP zostaje zakończona.

2 Analiza statystyk ruchu sieciowego za pomocą komendy netstat

2.1 Zapoznaj się z różnymi opcjami komendy netstat.

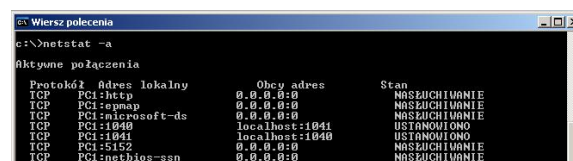
- a) W Wierszu polecenia systemu Windows wydaj polecenie *netstat /?*, dzięki któremu wyświetlisz wszystkie dostępne opcje komendy *netstat* na Twoim komputerze.

```
netstat /?
```

Zauważ, że możesz:

- wyświetlić listę wszystkich aktualnie otwartych i nasłuchujących portów (*netstat -a*),
 - wyświetlić listę wszystkich aktualnych połączeń dla danego protokołu: TCP lub UDP (*netstat -p nazwa_protokołu*),
 - wyświetlić statystyki sieciowe (*netstat -s*),
 - odświeżać wybrane statystyki co X sekund (*netstat X*).
- b) Wyświetl informację o wszystkich otwartych portach na Twoim komputerze, wydając polecenie *netstat -a*.

```
netstat -a
```



Znajdź wpis informujący o tym, że Twój komputer nasłuchuje na porcie 80, związanym z usługą HTTP, a więc jest serwerem WWW.

- c) Wyświetl informację o wszystkich sesjach TCP na Twoim komputerze, wydając polecenie *netstat -p tcp*.

```
netstat -p tcp
```

Jeśli nie widzisz żadnego wpisu, spróbuj wywołać ruch sieciowy, wchodząc np. na stronę internetową sąsiada.

Aktywne połączenia			
Protokół	Adres lokalny	Obcy adres	Stan
TCP	PCI:1040	localhost:1041	USTANOWIŁO
TCP	PCI:1041	localhost:1040	USTANOWIŁO
TCP	PCI:2176	172.16.1.253:telnet	USTANOWIŁO

TCP	192.168.1.9:49991	20.199.120.182:https	ESTABLISHED
TCP	192.168.1.9:58589	52.97.223.66:https	ESTABLISHED
TCP	192.168.1.9:58594	a23-73-140-82:https	CLOSE_WAIT
TCP	192.168.1.9:61697	rev-30:imaps	ESTABLISHED
TCP	192.168.1.9:61705	52.143.87.28:https	ESTABLISHED
TCP	192.168.1.9:61706	20.42.65.85:https	ESTABLISHED

Zwróć uwagę na stany, w jakich znajdują się sesje. Przez większość czasu jest to stan ESTABLISHED, związany z poprawnym zestawieniem sesji, ale może uda Ci się wychwycić moment zamykania sesji (CLOSE-WAIT lub TIME-WAIT) albo jej otwierania (SYN-SENT, SYN-RECEIVED).

- d) Odświeżaj co 5 sekund informację o wszystkich sesjach TCP na Twoim komputerze, wydając polecenie *netstat -p tcp 5*.

```
netstat -p tcp 5
```

Zatrzymaj odświeżanie, wciskając Ctrl+C.

- e) Wyświetl statystyki ruchu TCP na Twoim komputerze, wydając polecenie *netstat -s*.

```
netstat -s
```

```
TCP Statistics for IPv4
Active Opens                = 5589
Passive Opens               = 290
Failed Connection Attempts  = 35
Reset Connections           = 2643
Current Connections         = 88
Segments Received           = 447369
Segments Sent                = 111232
Segments Retransmitted      = 0
```

Przyjrzyj się informacjom na temat ilości dotychczas otwartych sesji (zainicjowanych albo przez Ciebie, albo z zewnątrz), nieudanych i wznowionych prób nawiązania połączenia, obecnie otwartych sesji, czy też otrzymanych/wysłanych/retransmitowanych segmentów.

2.2 Wykorzystaj komendę *netstat* do obserwacji stanów sesji TCP.

- W Wierszu polecenia systemu Windows wydaj polecenie *netstat -p tcp 5*, aby odświeżać co 5 sekund informacje o aktywnych sesjach TCP na Twoim komputerze. Nie martw się, jeśli na początku nie wyświetli się żaden wpis.
- Uruchom nowe okno Wiersza polecenia. Zaloguj się na laboratoryjny switch (komendą *telnet 172.16.1.253*, podaj hasło *cisco*).
- W pierwszym oknie Wiersza polecenia powinieneś zaobserwować pojawienie się nowej sesji — z urządzeniem 172.16.1.253 na porcie 23 (telnet),

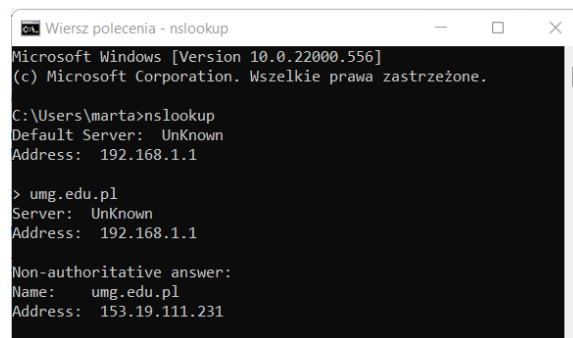
prawdopodobnie w stanie ESTABLISHED (stany występujące podczas zawiązywania sesji, tj. SYN-SENT oraz SYN-RECEIVED trwają bardzo krótko i jest nikła szansa, że odświeżenie statystyk wystąpi akurat podczas ich trwania).

- d) Możesz wylogować się ze switcha, wpisując komendę *exit*. Być może uda Ci się zaobserwować jeszcze stan związany z kończeniem sesji TCP pomiędzy Twoim komputerem a switchem (byłby to stan TIME-WAIT, jako że to Twój komputer jest inicjalizatorem zakończenia sesji. Ponownie, stany takie jak FIN-WAIT1 czy FIN-WAIT2 trwają zazwyczaj bardzo krótko).

3 Analiza ruchu UDP

3.1 Przechwyć ruch UDP wygenerowany podczas wysyłania zapytania i odbierania odpowiedzi DNS.

- a) Wyłącz kartę sieciową LAB, włącz kartę o nazwie Internet, aby odzyskać dostęp do internetu. Upewnij się (za pomocą polecenia *ipconfig*), czy Twój komputer otrzymał adres IP od serwera DHCP (z puli 192.168.133.0/24).
- b) Otwórz program Wireshark i rozpocznij przechwytywanie ruchu sieciowego na karcie sieciowej Realtek. Włącz filtrowanie przechwyconego ruchu, tak, by widoczne były tylko datagramy UDP.
- c) W Wierszu polecenia systemu Windows wydaj polecenie *nslookup*. Zadaaj serwerowi DNS zapytanie o adres IP jakiegokolwiek strony internetowej.



```
Wiersz polecenia - nslookup
Microsoft Windows [Version 10.0.22000.556]
(c) Microsoft Corporation. Wszelkie prawa zastrzeżone.

C:\Users\marta>nslookup
Default Server: UnKnown
Address: 192.168.1.1

> umg.edu.pl
Server: UnKnown
Address: 192.168.1.1

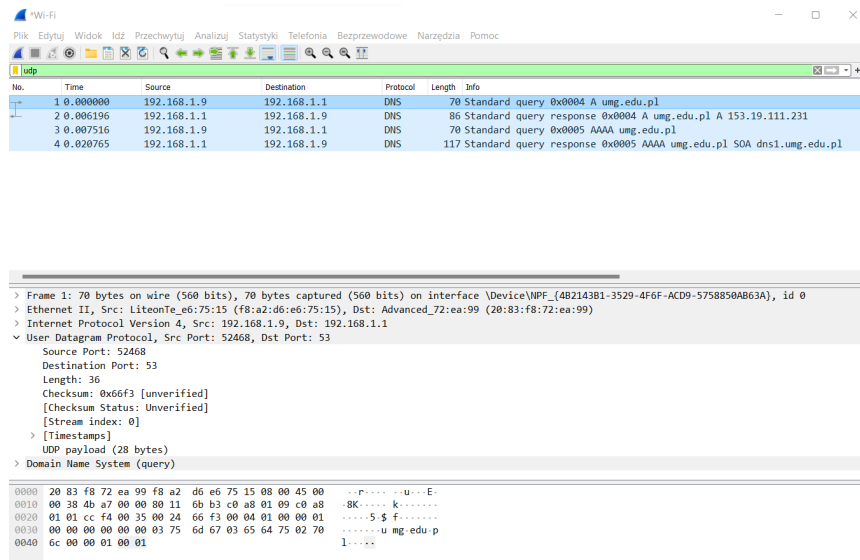
Non-authoritative answer:
Name: umg.edu.pl
Address: 153.19.111.231
```

- d) Sprawdź, czy w programie Wireshark pojawiły się przechwycone datagramy, po czym zatrzymaj przechwytywanie danych.

3.2 Przeanalizuj przechwycony ruch UDP.

- a) Przyjrzyj się przechwyconym datagramom. Zwróć uwagę, jak bardzo prosta jest komunikacja z wykorzystaniem UDP: nie zawiera informacji

kontrolno-sterujących służących do zawiązania sesji, potwierdzenia odebrania danych czy zakończenia sesji. W przechwyconym ruchu powinno być widać jedynie zapytanie-odpowiedź protokołu DNS.



- b) Kliknij w pierwszy przechwycony datagram (Twoje zapytanie DNS, *Standard query A*). Rozwiń czwartą sekcję w środkowej części ekranu — tę dotyczącą nagłówka protokołu warstwy transportowej, czyli UDP. Przekonaj się, jak uproszczony jest ten nagłówek w porównaniu z nagłówkiem TCP:

- Port źródłowy powinien zawierać dużą wartość (port prywatny),
- Port docelowy powinien wskazywać na usługę DNS (53). Nie ma żadnego pola dotyczącego flag, numeru sekwencji czy potwierdzenia.

- c) Zastanów się, w jakich sytuacjach bardziej opłaca się używać UDP, a nie TCP? Innymi słowy, kiedy bardziej zależy nam na prostej, nieobciążającej łączy komunikacji i możemy zaakceptować fakt, że część danych może zostać utraconych?

3.3 Zadanie dodatkowe — obserwacja ruchu TFTP.

TFTP (ang. *Trivial File Transfer Protocol*) jest prostym protokołem umożliwiającym przesyłanie plików, wykorzystującym na warstwie transportowej protokół UDP. Jako że UDP nie zapewnia mechanizmu potwierdzenia poprawnego odbioru danych, za utrzymanie niezawodności przesyłanych plików odpowiada sam TFTP.

Na Twoim komputerze zainstalowany jest program `tftpd23`, za pomocą którego Twój komputer staje się serwerem TFTP. Spróbuj wysłać na

ten serwer plik tekstowy z konfiguracją laboratoryjnego przełącznika, a w programie Wireshark zaobserwuj, jak wygląda ruch UDP podczas takiego przesyłu plików.

- a) Przełącz się z powrotem na kartę sieciową LAB.
- b) Uruchom program tftpd32 (Start \Rightarrow Programy \Rightarrow tftpd32).
- c) Zaloguj się na laboratoryjny switch (*telnet* 172.16.1.253 w cmd, hasło *cisco*). Przejdź komendą *enable* do trybu uprzywilejowanego (ponownie podaj hasło *cisco*).
- d) Uruchom przechwytywanie ruchu sieciowego w programie Wireshark, filtruj po UDP.
- e) Wydadź przełącznikowi polecenie przekopiowania pliku z konfiguracją bieżącą (*running-config*) na serwer TFTP:

```
Switch# copy running-config tftp:
```

Doprecyzuj adres serwera TFTP, na który chcesz wysłać plik (podaj adres IP swojego komputera) oraz zostaw domyślną nazwę pliku po wysłaniu. Powinieneś zaobserwować, że plik pojawił się na pulpicie Twojego komputera.

- f) Przyjrzyj się w Wiresharku, jak wygląda ruch TFTP. Zwróć uwagę na mechanizm potwierdzania odbioru stosowany w tym protokole.

V. Pytania kontrolne

1. Jaka jest rola warstwy transportowej według modelu OSI?
2. Jak wygląda proces rozpoczynania sesji TCP?
3. Jak wygląda proces kończenia sesji TCP?
4. W jaki sposób protokół TCP zapewnia niezawodność transmisji?
5. Jakie są różnice pomiędzy TCP a UDP?