# Model warstwowy i architektura sieci komputerowej

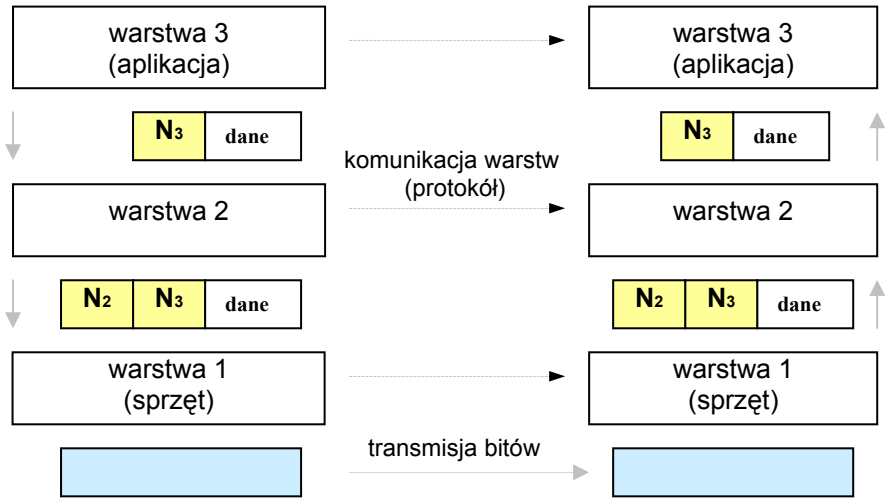
## 1 Wprowadzenie

**Wymagania wstępne: znajomość podstaw modelu warstwowego OSI.**

### 1.1 Podstawowe pojęcia

Ze względu na swą złożoność, oprogramowanie i sprzęt sieciowy podzielone są na niezależne funkcjonalnie **warstwy** (ang. layer). Jedna warstwa odpowiada za realizację odrębnego zbioru funkcji. Każda warstwa (oprócz najwyższej) dostarcza sąsiedniej warstwie wyższej interfejsu do swoich usług. **Model warstwowy** określa liczbę warstw oraz ogólne funkcje każdej z nich.

Komunikacja w sieci zgodnej z modelem warstwowym odbywa się tylko między odpowiadającymi sobie warstwami po obu stronach. Warstwa numer N po stronie nadawczej przesyła więc wiadomość tylko do warstwy numer N po stronie odbiorczej. Taką komunikację umożliwia proces zwany **kapsułkowaniem** (ang. encapsulation). Kapsułkowanie polega na umieszczeniu wiadomości warstwy wyższej wewnątrz wiadomości warstwy niższej. Zanim wiadomość po stronie nadawczej zostanie wysłana, przekazywana jest „w dół" stosu warstw; każda kolejna warstwa po otrzymaniu tej wiadomości od warstwy wyższej, dodaje do niej własny nagłówek i (rzadziej) stopkę. U odbiorcy natomiast zachodzi proces odwrotny. Wiadomość przekazywana jest „w górę" stosu warstw i każda kolejna warstwa interpretuje, a następnie usuwa nagłówek dodany poprzednio przez tę samą warstwę u nadawcy. Tak okrojona wiadomość przekazywana jest następnie warstwie wyższej. Aplikacja u odbiorcy otrzymuje zatem oryginalne dane, wysłane wcześniej przez aplikację po stronie nadawczej. Zasady omówionej tu komunikacji definiuje **protokół sieciowy**. Wiadomość każdego protokołu sieciowego obejmuje dwie części: informacje sterujące (tzw. *nagłówek*) oraz *dane*. Powyższy opis zilustrowano na rysunku 1, zakładającym istnienie trzech warstw.




**Rysunek 1. Kapsułkowanie w modelu 3-warstwowym.**

Warstwowy model sieci komputerowej został zaproponowany w połowie lat 80-tych przez organizację ISO i nosi nazwę modelu OSI (ang. Open Systems Interconnection). Model OSI wyróżnia 7 warstw, od najniższej – fizycznej, przez warstwy: łącza danych, sieciową, transportową, sesji i prezentacji, po warstwę aplikacji, która dostarcza interfejs usług sieciowych programom użytkowym, zwanym aplikacjami.

Przez **architekturę sieci komputerowej** rozumie się zestaw konkretnych rozwiązań protokołów w poszczególnych warstwach modelu. Dla przykładu, w 4-warstwowej architekturze TCP/IP (konkurencyjnej do architektury OSI) warstwę sieciową implementuje protokół IP, warstwę transportową – protokoły TCP oraz UDP, a warstwę aplikacji – cały szereg protokołów aplikacyjnych, np. FTP (transferu plików w sieci Internet), SMTP (wysyłanie poczty elektronicznej) czy SSH (praca w zdalnym systemie). Warto zauważyć, że wbrew swojej nazwie, architektura TCP/IP obejmuje dużo więcej niż tylko dwa protokoły. Ponadto, architektura TCP/IP nie uwzględnia warstw sesji i prezentacji oraz nie definiuje szczegółów dla warstw fizycznej i łącza danych.

Niniejsze ćwiczenie przybliża architekturę TCP/IP oraz proces kapsułkowania. Niektóre pojęcia, związane ze strukturą komunikatów różnych warstw mogą być dla studentów niezrozumiałe, lecz celem ćwiczenia jest przede wszystkim zrozumienie koncepcji kapsułkowania i warstwowej architektury oprogramowania sieciowego. Poznanie dodatkowych szczegółów pozostawia się Studentowi jako pracę własną.

## 2. Organizacja, wymagany sprzęt i oprogramowanie

- zadania wykonywane są indywidualnie;
- sprzęt: 1 komputer PC;
- oprogramowanie: system Linux, analizatory sieciowe (ang. *sniffer*) Wireshark oraz tcpdump.

## 3. Zadania

1. Korzystając z rysunków zamieszczonych na dwóch kolejnych stronach oraz z tabel 1-3, należy stwierdzić, jaka aplikacja wygenerowała poniższą ramkę. **Uwaga:** Ramka pochodzi z sieci technologii Ethernet DIX i nie zawiera preambuły (zaczyna się od pola Destination address).

```
00 50 04 4b 75 82 00 03 47 b0 86 0d 08 00 45 00
00 3b 3f 0d 00 00 80 11 0f 3d 96 fe 11 68 96 fe
ad 03 04 63 00 35 00 27 dd fc 00 02 01 00 00 01
00 00 00 00 00 00 03 77 77 77 06 67 6f 6f 67 6c
65 02 70 6c 00 00 01 00 01
```

2. Używając programów Wireshark oraz tcpdump (obydwa dostępne dla systemów Windows i Linux), dokonaj analizy ramek w lokalnej sieci. Zdefiniuj filtr, który wyselekcjonuje wszystkie te wiadomości, które zostały wysyłane z twojego komputera lub są adresowane do niego.

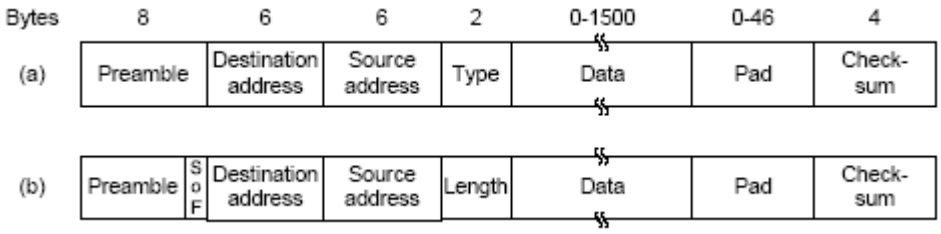
## 4. Pytania sprawdzające

1. Jakie są zalety i wady warstwowej architektury sieci komputerowej?
2. Na czym polega kapsułkowanie wiadomości? Jak przebiega u nadawcy, a jak u odbiorcy?
3. Warstwa wyższa może przekazać wiadomość warstwie niższej, gdyż ma dostęp do interfejsu jej usług. Jak natomiast odbywa się przekazywanie wiadomości „w górę" stosu warstw po stronie odbiorczej?

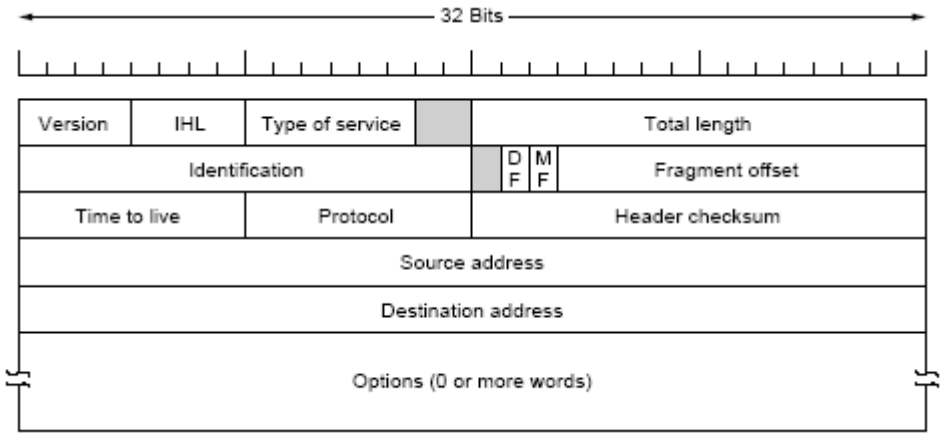
## 5. Literatura


1. Koncepcja modelu warstwowego oraz zarys implementacji: A. S. Tanenbaum „Sieci komputerowe", Helion 2004.
2. Koncepcja modelu warstwowego: J. F. Kurose, F. Ross „Computer Networking – A Top-Down Approach Featuring the Internet", Prentice Hall 2003.
3. Implementacja modelu warstwowego: D. E. Comer i D. L. Stevens „Sieci komputerowe TCP/IP", WNT 1997.

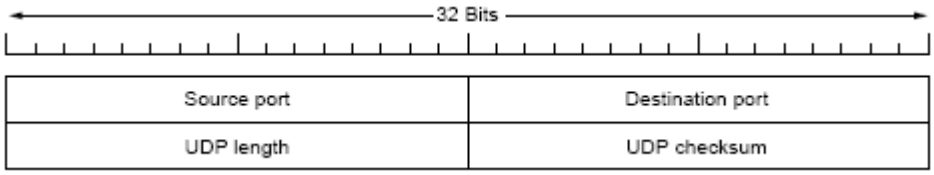
Poniższe rysunki pochodzą z książki A. S. Tanenbaum „Sieci komputerowe”.




**Rysunek 1. Format ramki Ethernet a) Ethernet DIX, b) Ethernet IEEE 802.3**




**Rysunek 2. Format pakietu IPv4**




**Rysunek 3. Format nagłówka UDP**

| Wartość szesnastkowa | Protokół |
|---|---|
| 6005 | DEC |
| 8137 | IPX |
| 0800 | IP |
| 809B | AppleTalk |

**Tabela 1 - przykładowe wartości pola Type w ramce Ethernet DIX**

| Wartość dziesiętna | Protokół |
|---|---|
| 6 | TCP |
| 17 | UDP |
| 88 | EIGRP |
| 89 | OSPF |
| 1 | ICMP |
| 41 | IPv6 |

**Tabela 2 - przykładowe wartości pola Protocol nagłówka IP**

| Aplikacja | Port UDP | Port TCP |
|---|---|---|
| FTP | | 20, 21 |
| SSH | | 22 |
| TELNET | | 23 |
| SMTP | | 25 |
| DNS | 53 | 53 |
| HTTP | | 80 |
| SNMP | 161 | |

**Tabela 3 - numery portów (dziesiętne) niektórych aplikacji**

**Kod ASCII (ze strony http://www.neurophys.wisc.edu/www/comp/docs/ascii.html)**

| Decimal | Octal | Hex | Binary | Value | |
|---|---|---|---|---|---|
| 000 | 000 | 000 | 00000000 | NUL | (Null char.) |
| 001 | 001 | 001 | 00000001 | SOH | (Start of Header) |
| 002 | 002 | 002 | 00000010 | STX | (Start of Text) |
| 003 | 003 | 003 | 00000011 | ETX | (End of Text) |
| 004 | 004 | 004 | 00000100 | EOT | (End of Transmission) |
| 005 | 005 | 005 | 00000101 | ENQ | (Enquiry) |
| 006 | 006 | 006 | 00000110 | ACK | (Acknowledgment) |
| 007 | 007 | 007 | 00000111 | BEL | (Bell) |
| 008 | 010 | 008 | 00001000 | BS | (Backspace) |
| 009 | 011 | 009 | 00001001 | HT | (Horizontal Tab) |
| 010 | 012 | 00A | 00001010 | LF | (Line Feed) |
| 011 | 013 | 00B | 00001011 | VT | (Vertical Tab) |
| 012 | 014 | 00C | 00001100 | FF | (Form Feed) |
| 013 | 015 | 00D | 00001101 | CR | (Carriage Return) |
| 014 | 016 | 00E | 00001110 | SO | (Shift Out) |
| 015 | 017 | 00F | 00001111 | SI | (Shift In) |
| 016 | 020 | 010 | 00010000 | DLE | (Data Link Escape) |
| 017 | 021 | 011 | 00010001 | DC1 (XON) | (Device Control 1) |
| 018 | 022 | 012 | 00010010 | DC2 | (Device Control 2) |
| 019 | 023 | 013 | 00010011 | DC3 (XOFF) | (Device Control 3) |
| 020 | 024 | 014 | 00010100 | DC4 | (Device Control 4) |
| 021 | 025 | 015 | 00010101 | NAK | (Negative Acknowledgement) |
| 022 | 026 | 016 | 00010110 | SYN | (Synchronous Idle) |
| 023 | 027 | 017 | 00010111 | ETB | (End of Trans. Block) |
| 024 | 030 | 018 | 00011000 | CAN | (Cancel) |
| 025 | 031 | 019 | 00011001 | EM | (End of Medium) |
| 026 | 032 | 01A | 00011010 | SUB | (Substitute) |
| 027 | 033 | 01B | 00011011 | ESC | (Escape) |
| 028 | 034 | 01C | 00011100 | FS | (File Separator) |
| 029 | 035 | 01D | 00011101 | GS | (Group Separator) |
| 030 | 036 | 01E | 00011110 | RS | (Request to Send)(Record Separator) |
| 031 | 037 | 01F | 00011111 | US | (Unit Separator) |
| 032 | 040 | 020 | 00100000 | SP | (Space) |
| 033 | 041 | 021 | 00100001 | ! | (exclamation mark) |
| 034 | 042 | 022 | 00100010 | " | (double quote) |
| 035 | 043 | 023 | 00100011 | # | (number sign) |
| 036 | 044 | 024 | 00100100 | $ | (dollar sign) |
| 037 | 045 | 025 | 00100101 | % | (percent) |
| 038 | 046 | 026 | 00100110 | & | (ampersand) |
| 039 | 047 | 027 | 00100111 | ' | (single quote) |
| 040 | 050 | 028 | 00101000 | ( | (left/opening parenthesis) |
| 041 | 051 | 029 | 00101001 | ) | (right/closing parenthesis) |
| 042 | 052 | 02A | 00101010 | * | (asterisk) |
| 043 | 053 | 02B | 00101011 | + | (plus) |
| 044 | 054 | 02C | 00101100 | , | (comma) |
| 045 | 055 | 02D | 00101101 | - | (minus or dash) |
| 046 | 056 | 02E | 00101110 | . | (dot) |
| 047 | 057 | 02F | 00101111 | / | (forward slash) |
| 048 | 060 | 030 | 00110000 | 0 | |
| 049 | 061 | 031 | 00110001 | 1 | |
| 050 | 062 | 032 | 00110010 | 2 | |
| 051 | 063 | 033 | 00110011 | 3 | |
| 052 | 064 | 034 | 00110100 | 4 | |
| 053 | 065 | 035 | 00110101 | 5 | |
| 054 | 066 | 036 | 00110110 | 6 | |
| 055 | 067 | 037 | 00110111 | 7 | |
| 056 | 070 | 038 | 00111000 | 8 | |
| 057 | 071 | 039 | 00111001 | 9 | |
| 058 | 072 | 03A | 00111010 | : | (colon) |
| 059 | 073 | 03B | 00111011 | ; | (semi-colon) |
| 060 | 074 | 03C | 00111100 | < | (less than) |
| 061 | 075 | 03D | 00111101 | = | (equal sign) |
| 062 | 076 | 03E | 00111110 | > | (greater than) |
| 063 | 077 | 03F | 00111111 | ? | (question mark) |
| 064 | 100 | 040 | 01000000 | @ | (AT symbol) |
| 065 | 101 | 041 | 01000001 | A | |
| 066 | 102 | 042 | 01000010 | B | |
| 067 | 103 | 043 | 01000011 | C | |
| 068 | 104 | 044 | 01000100 | D | |
| 069 | 105 | 045 | 01000101 | E | |
| 070 | 106 | 046 | 01000110 | F | |
| 071 | 107 | 047 | 01000111 | G | |
| 072 | 110 | 048 | 01001000 | H | |
| 073 | 111 | 049 | 01001001 | I | |
| 074 | 112 | 04A | 01001010 | J | |
| 075 | 113 | 04B | 01001011 | K | |
| 076 | 114 | 04C | 01001100 | L | |
| 077 | 115 | 04D | 01001101 | M | |
| 078 | 116 | 04E | 01001110 | N | |
| 079 | 117 | 04F | 01001111 | O | |
| 080 | 120 | 050 | 01010000 | P | |
| 081 | 121 | 051 | 01010001 | Q | |

| | | | | | |
|---|---|---|---|---|---|
| 082 | 122 | 052 | 01010010 | R | |
| 083 | 123 | 053 | 01010011 | S | |
| 084 | 124 | 054 | 01010100 | T | |
| 085 | 125 | 055 | 01010101 | U | |
| 086 | 126 | 056 | 01010110 | V | |
| 087 | 127 | 057 | 01010111 | W | |
| 088 | 130 | 058 | 01011000 | X | |
| 089 | 131 | 059 | 01011001 | Y | |
| 090 | 132 | 05A | 01011010 | Z | |
| 091 | 133 | 05B | 01011011 | [ | (left/opening bracket) |
| 092 | 134 | 05C | 01011100 | \ | (back slash) |
| 093 | 135 | 05D | 01011101 | ] | (right/closing bracket) |
| 094 | 136 | 05E | 01011110 | ^ | (caret/cirumflex) |
| 095 | 137 | 05F | 01011111 | _ | (underscore) |
| 096 | 140 | 060 | 01100000 | ` | |
| 097 | 141 | 061 | 01100001 | a | |
| 098 | 142 | 062 | 01100010 | b | |
| 099 | 143 | 063 | 01100011 | c | |
| 100 | 144 | 064 | 01100100 | d | |
| 101 | 145 | 065 | 01100101 | e | |
| 102 | 146 | 066 | 01100110 | f | |
| 103 | 147 | 067 | 01100111 | g | |
| 104 | 150 | 068 | 01101000 | h | |
| 105 | 151 | 069 | 01101001 | i | |
| 106 | 152 | 06A | 01101010 | j | |
| 107 | 153 | 06B | 01101011 | k | |
| 108 | 154 | 06C | 01101100 | l | |
| 109 | 155 | 06D | 01101101 | m | |
| 110 | 156 | 06E | 01101110 | n | |
| 111 | 157 | 06F | 01101111 | o | |
| 112 | 160 | 070 | 01110000 | p | |
| 113 | 161 | 071 | 01110001 | q | |
| 114 | 162 | 072 | 01110010 | r | |
| 115 | 163 | 073 | 01110011 | s | |
| 116 | 164 | 074 | 01110100 | t | |
| 117 | 165 | 075 | 01110101 | u | |
| 118 | 166 | 076 | 01110110 | v | |
| 119 | 167 | 077 | 01110111 | w | |
| 120 | 170 | 078 | 01111000 | x | |
| 121 | 171 | 079 | 01111001 | y | |
| 122 | 172 | 07A | 01111010 | z | |
| 123 | 173 | 07B | 01111011 | { | (left/opening brace) |
| 124 | 174 | 07C | 01111100 | \| | (vertical bar) |
| 125 | 175 | 07D | 01111101 | } | (right/closing brace) |
| 126 | 176 | 07E | 01111110 | ~ | (tilde) |
| 127 | 177 | 07F | 01111111 | DEL | (delete) |