Warstwa transportowa

mgr inż. Krzysztof Szałajko

Modele odniesienia

OSI		TCP/IP
7	Aplikacji	Aplikacji
6	Prezentacji	
5	Sesji	
4	Transportowa	Transportowa
3	Sieciowa	Internetowa
2	Łącza danych	Dostępu do sieci
1	Fizyczna	

Warstwa transportowa

- Przygotowuje dane otrzymane od aplikacji do wysłania do sieci
- Przetwarza dane otrzymane z warstwy Internetowej (sieci) do postaci możliwej do przetworzenia przez warstwę aplikacji

Cele warstwy transportowej

Warstwa aplikacji segmentuje dane oraz dba o ich prawidłowy transport poprzez różne ciągi komunikacyjne.

- Śledzi komunikację pomiędzy hostami źródłowym i docelowym
- Segmentuje i oznacza dane
- Łączy odpowiednie fragmenty danych
- Identyfikuje aplikacje

Śledzenie komunikacji

- Dzięki warstwie aplikacji może istnieć wiele strumieni komunikacyjnych pomiędzy aplikacjami
- Wiele aplikacji może się komunikować z jednego hosta

Segmentacja

- Warstwa transportowa dzieli dane na możliwe do przesłania fragmenty
- Dodaje nagłówek umożliwiający poskładanie segmentów odpowiedniego strumienia informacji
- Enkapsulacja w segmenty danych

Scalanie

- Segmenty danych trafiające do warstwy aplikacji wymagają scalenia przed przekazaniem ich do warstwy aplikacji
- Scalanie następuje dzięki informacjom zawartym w nagłówku warstwy transportowej

Identyfikacja aplikacji

Warstwa transportowa musi zidentyfikować aplikację, do której ma przesłać otrzymane dane.

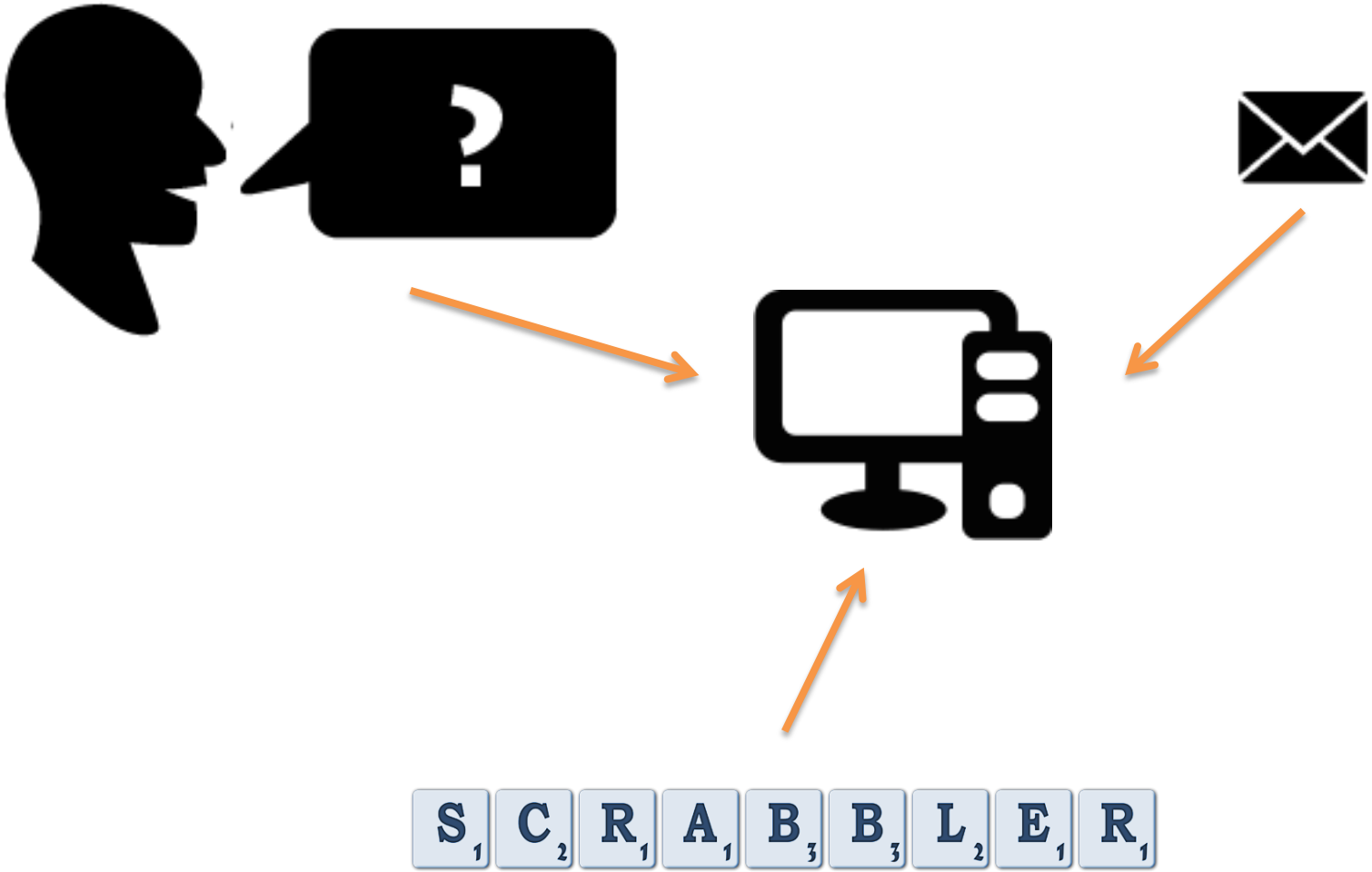
Identyfikacji dokonuje po przydzielonym danej aplikacji identyfikatorze - **numerze portu**.

Numer ten zostaje zapisany w nagłówku warstwy transportowej.

Różnorodność wymagań

- Różne protokoły dostarczają różnych funkcjonalności
  - Minimalizacja opóźnienia: ale nie wszystkie dane muszą dotrzeć do celu
  - Niezawodność dostarczenia danych: ale za to większe wymagania dotyczące sieci

Rozdzielenie strumieni komunikacji



Rozdzielenie strumieni komunikacji

Załóżmy przykładową sytuację, w której użytkownik rozmawia przez Skype`a, odbiera pocztę elektroniczną oraz walczy o uzyskanie najlepszego wyniku na scrabbler.pl.

Każda z uruchomionych aplikacji wysyła dane i odbiera je w tym samym momencie.

Rozdzielenie strumieni komunikacji

Wymagania:

- E-mail oraz strona WWW muszą dotrzeć w całości, możliwe są pewne opóźnienia czasowe
- Zagubienie pewnej drobnej części danych w konwersacji jest możliwe, natomiast bardzo ważne jest zapewnienie możliwie najmniejszego opóźnienia

Rozdzielenie strumieni komunikacji

Warstwa aplikacji:

- Rozdziela dane na segmenty
- Zapewnia separację danych z poszczególnych aplikacji

Zalety segmentacji danych

- Wysyłanie ciągłego strumienia danych uniemożliwiałoby inną konwersację oraz utrudniałoby retransmisję
- Dzięki segmentacji danych możliwe jest dokonywanie kilku operacji jednocześnie, np. oglądanie filmu online, odbieranie maila i rozmowa poprzez komunikator

Sterowanie konwersacją

Konwersacja – każdy zbiór danych przepływający pomiędzy hostami

- Segmentacja / scalanie
- Multipleksacja komunikacji
  Na hoście może być uruchomione wiele aplikacji, oznaczonych numerami portów

Sterowanie konwersacją

- Ustanowienie sesji

  W celu dokonania konwersacji zorientowanej połączeniowo warstwa transportowa ustanawia sesję pomiędzy aplikacjami.

Sterowanie konwersacją

- Niezawodność dostarczenia danych

  Warstwa aplikacji może zapewnić niezawodność dostarczenia danych poprzez ich retransmisję w przypadku niepowodzenia dostarczenia

Sterowanie konwersacją

- Kontrola przepływu

  Odbiorca może zażądać zmniejszenia prędkości nadawania w przypadku wyczerpywania się pewnych zasobów, jak np. pamięć czy zbyt ograniczona przepustowość

Wymagane właściwości protokołu

- Szybkość
- Brak potwierdzeń
- Brak retransmisji
- Mały narzut
- Bez układania kolejności

- Niezawodność
- Potwierdzenia
- Retransmisje
- Większy narzut
- Układanie w kolejności

Najpopularniejsze protokoły

- UDP
  - Protokół bezpołączeniowy
  - Opisany w RFC 768
  - Niewielki narzut (8 dodatkowych bajtów nagłówka)
  - Fragmenty danych = datagramy
  - Best effort
  - UDP wykorzystują: DNS, aplikacje przesyłające strumienie wideo, VoIP

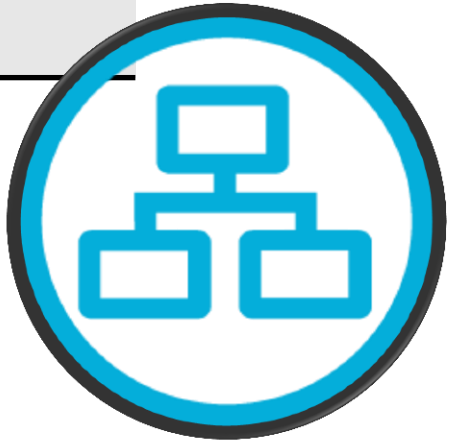
Najpopularniejsze protokoły

- TCP/IP
  - Zorientowany połączeniowo
  - Opisany w RFC 793
  - Dodatkowy narzut ze względu na liczbę dodatkowych funkcji (20 dodatkowych bajtów):
    - Niezawodne dostarczanie
    - Właściwa kolejność
    - Kontrola przepływu
  - Aplikacje wykorzystujące: przeglądarki, e-mail, aplikacje do przesyłania plików

Segment TCP

Port źródłowy (16)	Port docelowy (16)
Numer sekwencyjny (32)	
Numer potwierdzenia (32)	
Długość nagłówka (6) Zarezerwowane (4) Bity kodu (6)	Okno (16)
Suma kontrolna (16)	Wskaźnik pilności (16)
Opcje (0 lub 32)	
Dane warstwy aplikacji	

20 bajtów

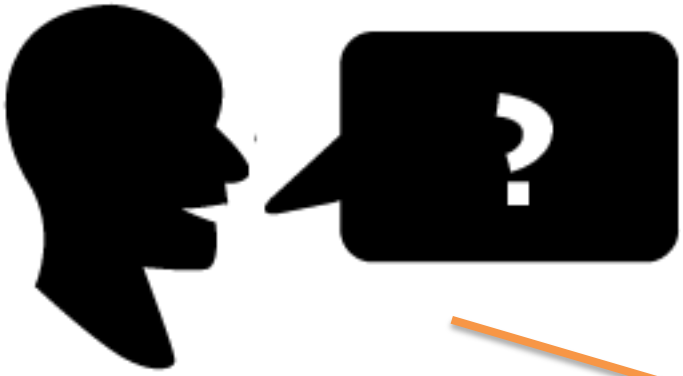


Datagram UDP

Port źródłowy (16)	Port docelowy (16)
Długość (16)	Suma kontrolna (16)
Dane warstwy aplikacji	

8 bajtów

Adresacja portów



Komunikator
port: 8074

E-mail
port: 110

Strona WWW
port: 80

Adresacja portów

- Numer portu + numer IP to konkretny proces na konkretnym urządzeniu
- Aplikacja musi „wiedzieć”, do jakiego portu docelowego wysłać zapytanie
- Port źródłowy jest wybierany losowo, nie może się on powielać z innym, już wykorzystywanym w systemie

Adresacja portów

- Przykład zapytania skierowanego do serwera WWW na port 80, o IPv4: 10.0.0.1:
  - Zapytanie kierowane do gniazda (socket):
    - 10.0.0.1:80
- Losowo wybrany port źródłowy: 65535, a przeglądarkę uruchomiliśmy na hoście: 10.0.0.24
  - Gniazdo dla tej strony:
    - 10.0.0.24:65535

Adresacja portów

- **ICANN**
  (ang. The Internet Corporation for Assigned Names and Numbers - Internetowa Korporacja ds. Nadawania Nazw i Numerów)
  - Zarządzanie nazwami domen
  - Ustalanie struktury domen
  - Nadzór nad działaniem serweów DNS
  - Przydział puli adresów IPv4, IPv6
  - **Rejestracja numerów portów**

Typy numerów portów

- Dobrze znane
  - Numery 0 – 1023
  - Numery zarezerwowane dla usług i aplikacji
- Porty zarejestrowane
  - Numery 1024 - 49151
  - Zarezerwowane dla aplikacji i procesów użytkownika
  - Mogą być używane jako dynamiczne, wybierane przez klienta jako port źródłowy

Typy numerów portów

- Prywatne / dynamiczne numery portów
  - Numery 49152 – 65535
  - Dynamicznie losowane przez aplikacje klienckie jako numer portu źródłowego

Aktualne lista numerów portów:

http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml

Adresacja portów

- Dobrze znane porty TCP/IP
  - 20, 21 FTP
  - 23 Telnet
  - 25 SMTP
  - 80 HTTP
  - 110 POP3
  - 194 IRC
  - 443 HTTPS

Adresacja portów

- Zarejestrowane porty TCP/IP
  - 1863 MSN Messenger
  - 8008 dodatkowy port HTTP
  - 8080 dodatkowy port HTTP

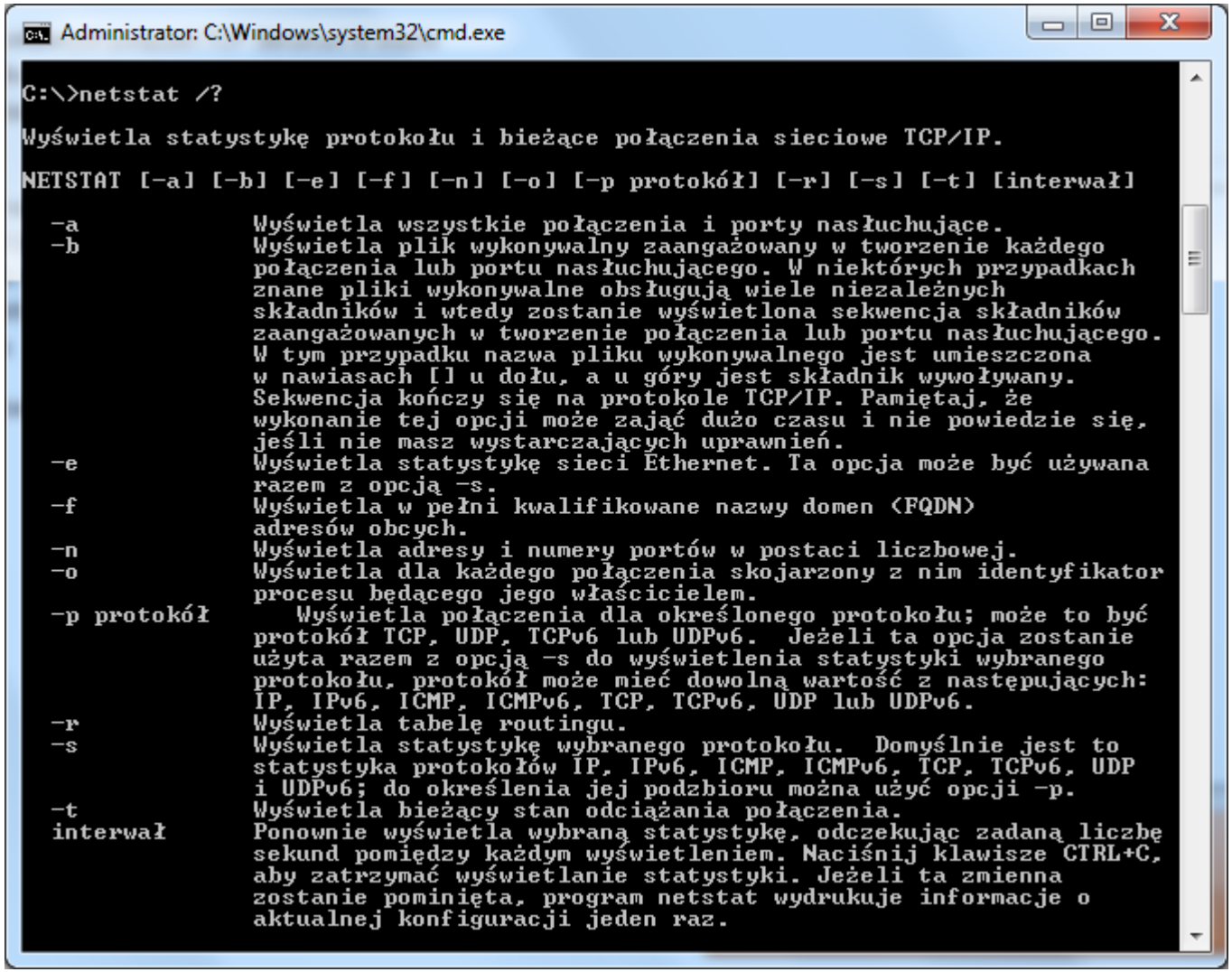
Adresacja portów

- Dobrze znane protokoły UDP
  - 69 TFTP
  - 520 RIP
- Zarejestrowane protokoły UDP
  - 1812 protokół uwierzytelniania RADIUS
  - 2000 Cisco SCCP (VoIP)
  - 5060 SIP (VoIP)

Adresacja portów

- Dobrze znane wspólne protokoły TCP/IP, UDP
  - 53 DNS
  - 161 SNMP
- Porty zarejestrowane
  - 1433 MS SQL
  - 2948 WAP (MMS)

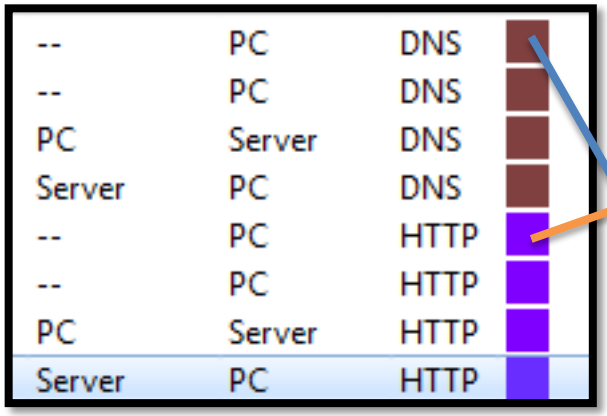
NETSTAT

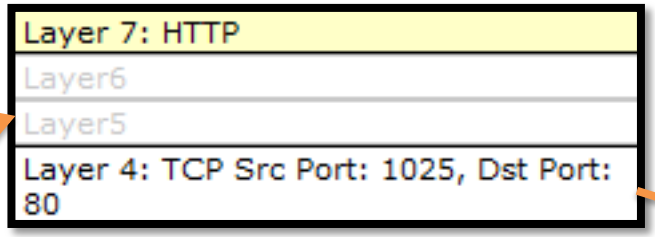


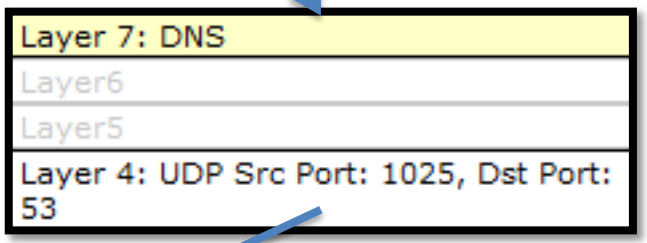
Informacja na temat aktywnych połączeń TCP.

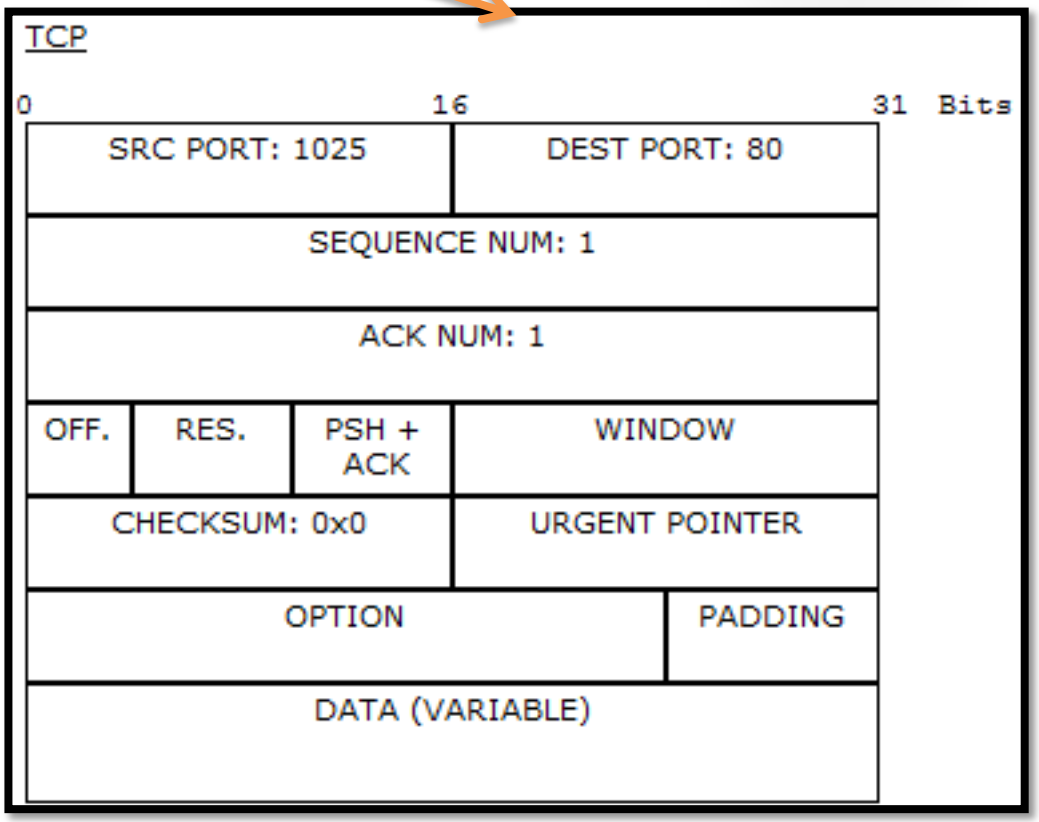
Netstat wyświetla używane protokoły, lokalne i zdalne adresy i numery portów oraz status połączenia.

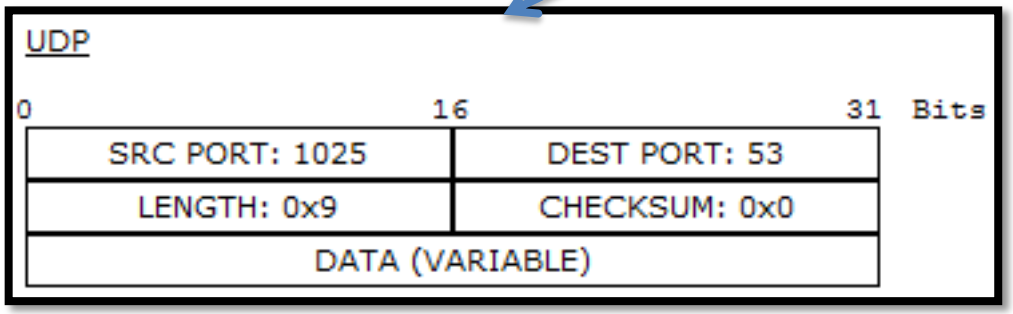
TCP / UDP w Packet Tracer











TCP
Komunikacja niezawwodna

Niezawodność

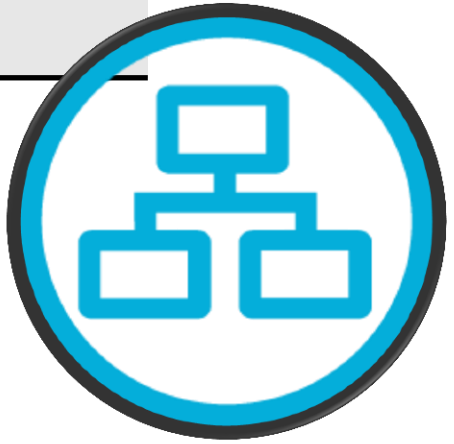
- Uzyskiwania poprzez mechanizm sesji, potwierdzeń oraz retransmisje
  - Komunikacja rozpoczyna się od ustanowienia obustronnej sesji
  - Dostarczenie każdego wysłanego segmentu jest potwierdzane
  - W przypadku braku potwierdzenia dane są retransmitowane

Nakład

- Niezawodność generuje wymagany dodatkowy nakład
  - Ustanowienie sesji
  - Kontrola stanu sesji
  - Potwierdzenia
  - Retransmisje

Pola segmentu

Port źródłowy (16)	Port docelowy (16)
Numer sekwencyjny (32)	
Numer potwierdzenia (32)	
Długość nagłówka (6) Zarezerwowane (4) Bity kodu (6)	Okno (16)
Suma kontrolna (16)	Wskaźnik pilności (16)
Opcje (0 lub 32)	
Dane warstwy aplikacji	



Pola segmentu

- Numer portu źródłowego
  - Numer portu urządzenia, które utworzyło sesję, zazwyczaj wartość losowa powyżej 1023
- Numer portu docelowego
  - Protokół warstwy wyższej
- Numer sekwencyjny
  - 32-bitowy identyfikator określający miejsce pakietu danych w pliku przed fragmentacją

Pola segmentu

- Numer potwierdzenia
  - 32-bitowy numer będący potwierdzeniem otrzymania pakietu przez odbiorcę
- Długość nagłówka
  - Wielkość nagłówka segmentu liczony w ilości 32 bajtowych linii

Pola segmentu

- Zarezerwowane
  - Miejsce przeznaczone do ewentualnego wykorzystania w przyszłości, obecnie wypełnione zerami
- Flagi
  - Określenie sposobu traktowania bieżącego segmentu

Flagi

NS – (ang. Nonce Sum) jednobitowa suma wartości flag ECN (ECN Echo, Congestion Window Reduced, Nonce Sum) weryfikująca ich integralność

CWR – (ang. Congestion Window Reduced) flaga potwierdzająca odebranie powiadomienia przez nadawcę, umożliwia odbiorcy zaprzestanie wysyłania echa.

ECE – (ang. ECN-Echo) flaga ustawiana przez odbiorcę w momencie otrzymania pakietu z ustawioną flagą CE

URG – informuje o istotności pola "Priorytet"

ACK – informuje o istotności pola "Numer potwierdzenia"

PSH – wymusza przesłanie pakietu

RST – resetuje połączenie (wymagane ponowne uzgodnienie sekwencji)

SYN – synchronizuje kolejne numery sekwencyjne

FIN – oznacza zakończenie przekazu danych

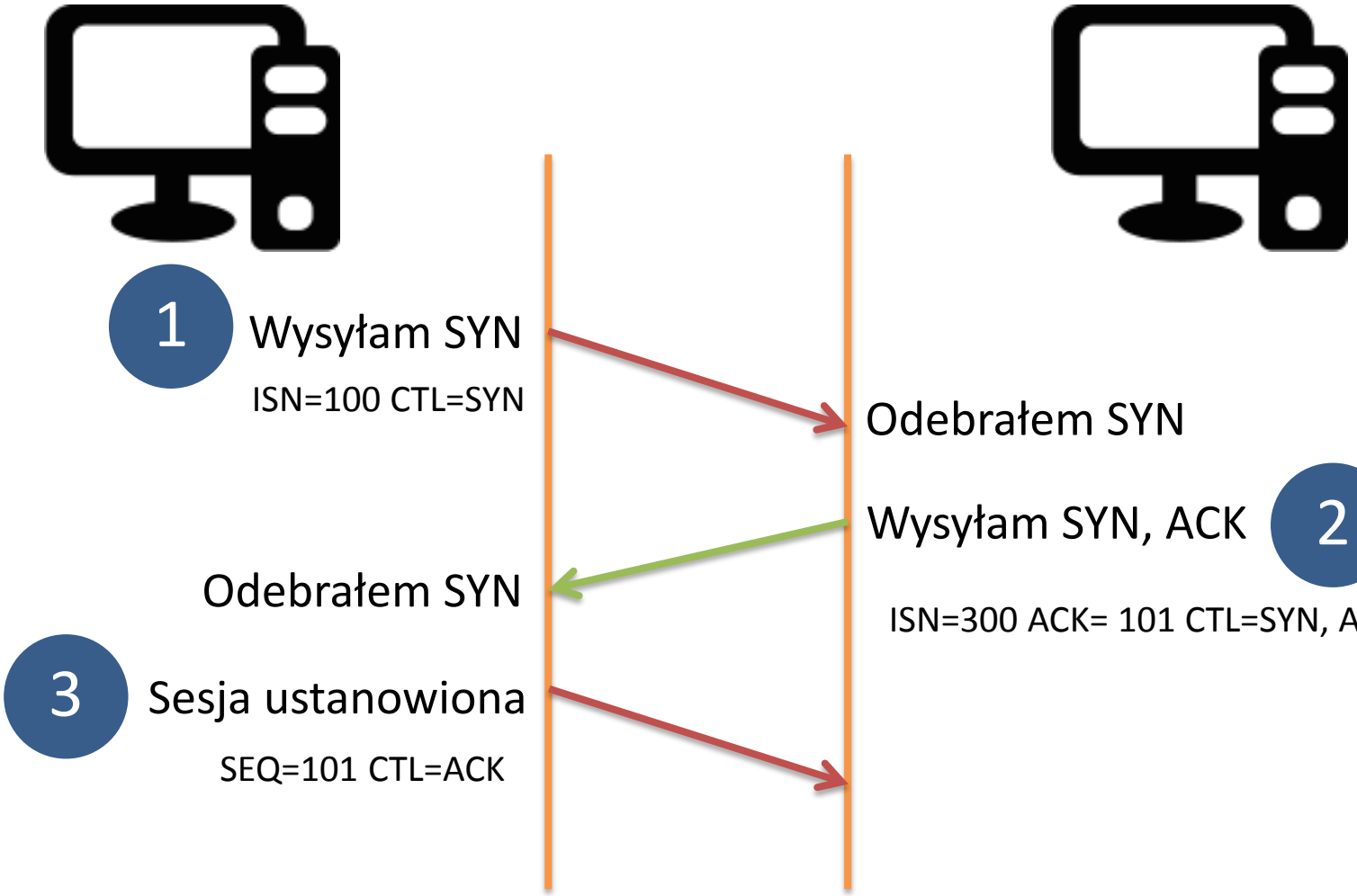
Pola segmentu

- Rozmiar okna
  - Ilość bajtów, jaka może być przesłana, zanim nadawca będzie oczekiwał na potwierdzenie
- Suma kontrolna
  - Pozwala na sprawdzenie pakietu pod względem poprawności przesłania danych

Pola segmentu

- Wskaźnik ważności
  - Ważność (priorytet) pakietu, wykorzystywany jeśli flaga URG jest włączona
- Opcje
  - Ewentualne dodatkowe informacje i polecenia

Uzgodnienie trójetapowe



Uzgodnienie trójetapowe

- Sprawdza czy urządzenie docelowe jest dostępne w sieci
- Sprawdza czy urządzenie docelowe ma aktywną usługę i akceptuje połączenie na wskazanym porcie
- Informuje urządzenie docelowe o chęci nawiązania połączenia

Uzgodnienie trójetapowe

- Etap 1
  Host wysyła segment SYN z początkową wartością synchronizacyjną ISN
  – żądanie rozpoczęcia sesji

Klient przechodzi w stan SYN-SENT.

Uzgodnienie trójetapowe

- Etap 2
  Serwer odpowiada wysyłając segment SYN, ACK. Ustanawia wartość potwierdzenia (ACK) na o 1 większą od otrzymanej wartości ISN oraz własną wartość ISN.

Serwer przechodzi w stan SYN-RECEIVED

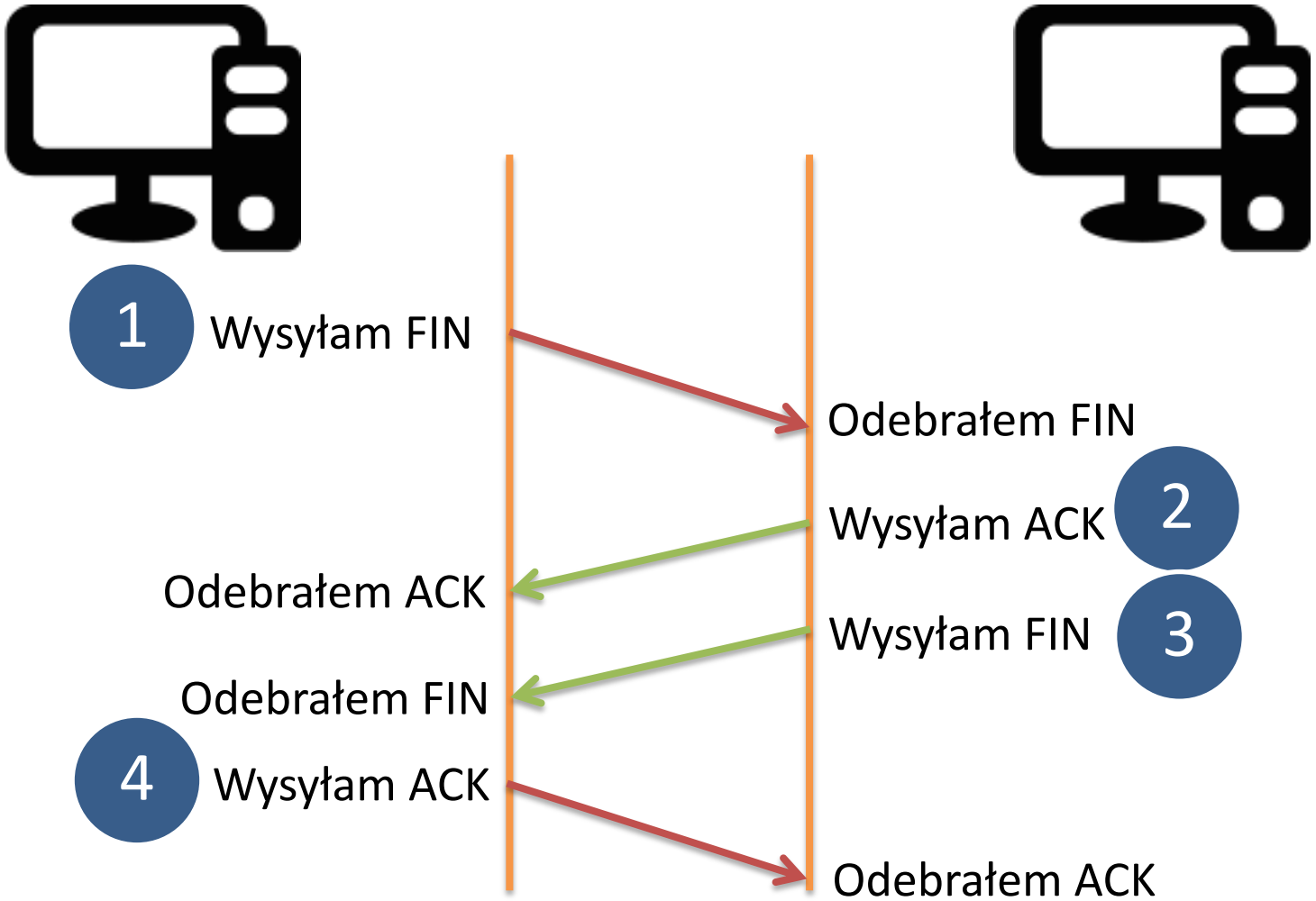
Uzgodnienie trójetapowe

- Etap 3
  Klient potwierdza nawiązanie połączenia wartością o 1 większą od otrzymanej wartości synchronizacji.

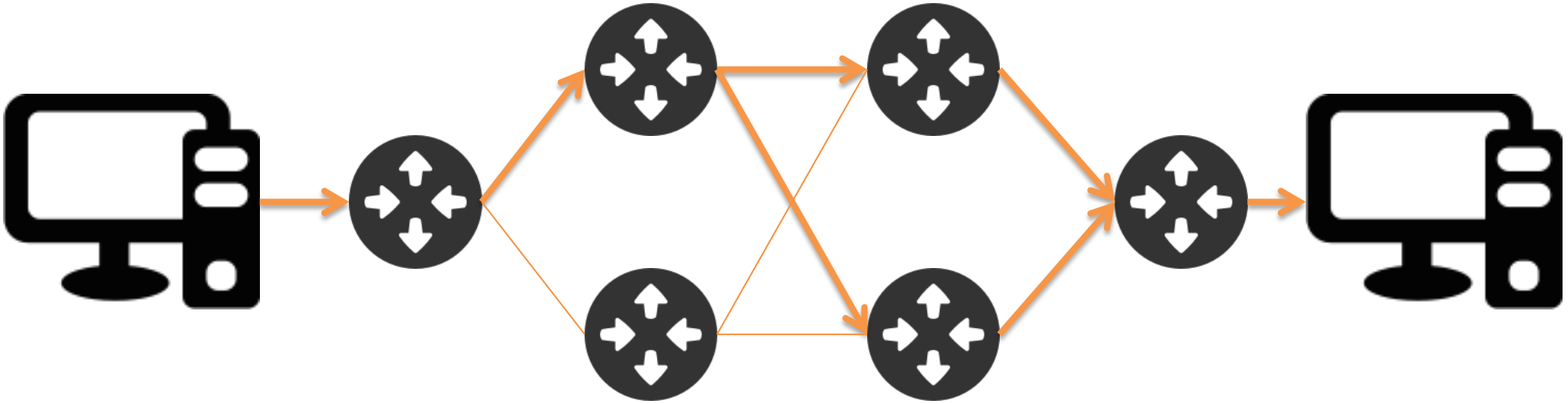
Klient przechodzi w stan SYN-ESTABLISHED

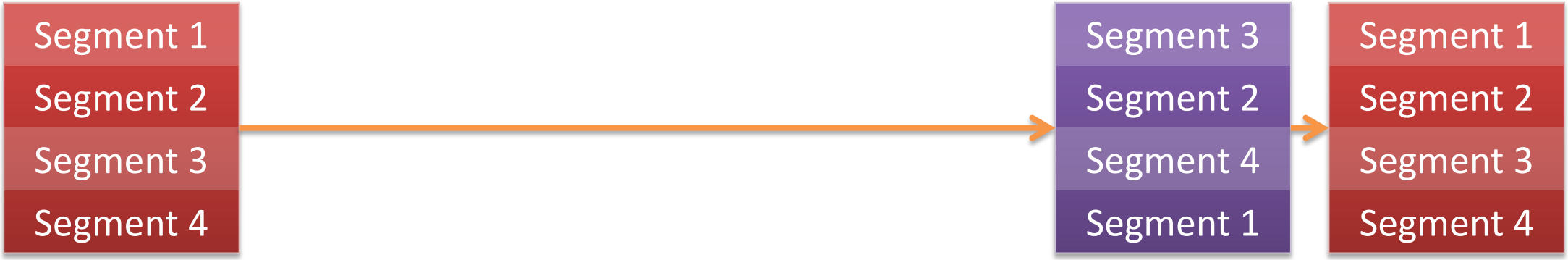
Serwer po otrzymaniu potwierdzenia przechodzi w stan SYN-ESTABLISHED

Finalizowanie połączenia



Scalanie segmentów





Scalanie segmentów

- Segmenty mogą podróżować różnymi drogami i dotrzeć w zmienionej kolejności
- Każdy segment ma większy numer ISN od poprzedniego o jego ilość bitów
- Proces odbierający umieszcza dane w buforze odbiorczym
- Po ustawieniu segmentów są one przekazywane do warstwy aplikacji

Potwierdzenia segmentów

Źródło	Cel	Nr sekw.	Potw.	
1088	23	1	1	...

10 bajtów

Źródło	Cel	Nr sekw.	Potw.	
23	1088	1	11	...

Źródło	Cel	Nr sekw.	Potw.	
1088	23	11	1	...

Kolejne bajty począwszy od 11

Potwierdzenia segmentów

- Wiele segmentów może zostać wysłanych zanim nadawca otrzyma potwierdzenia
- Ilość danych jaką nadawca może przesłać zanim otrzyma potwierdzenia nazywamy **rozmiarem okna**

Retransmisja utraconych segmentów

- Host docelowy potwierdza grupę otrzymanych segmentów
- Wysłana została grupa segmentów o numerach od 1000 do 5000
  - Sytuacja 1: wszystkie segmenty dotarły do adresata
    Zostaje wysłane potwierdzenie o numerze 5001
  - Sytuacja 2: nie dotarły segmenty od 3000 do 3500
    Zostaje wysłane potwierdzenie o numerze 3001

Retransmisja utraconych segmentów

- W przypadku braku otrzymania potwierdzenia otrzymania całej grupy segmentów jest ona w całości retransmitowana
- W przypadku otrzymania potwierdzenia częściowego dostarczenia segmentów, retransmitowana zostaje grupa segmentów, których potwierdzenia nie otrzymano

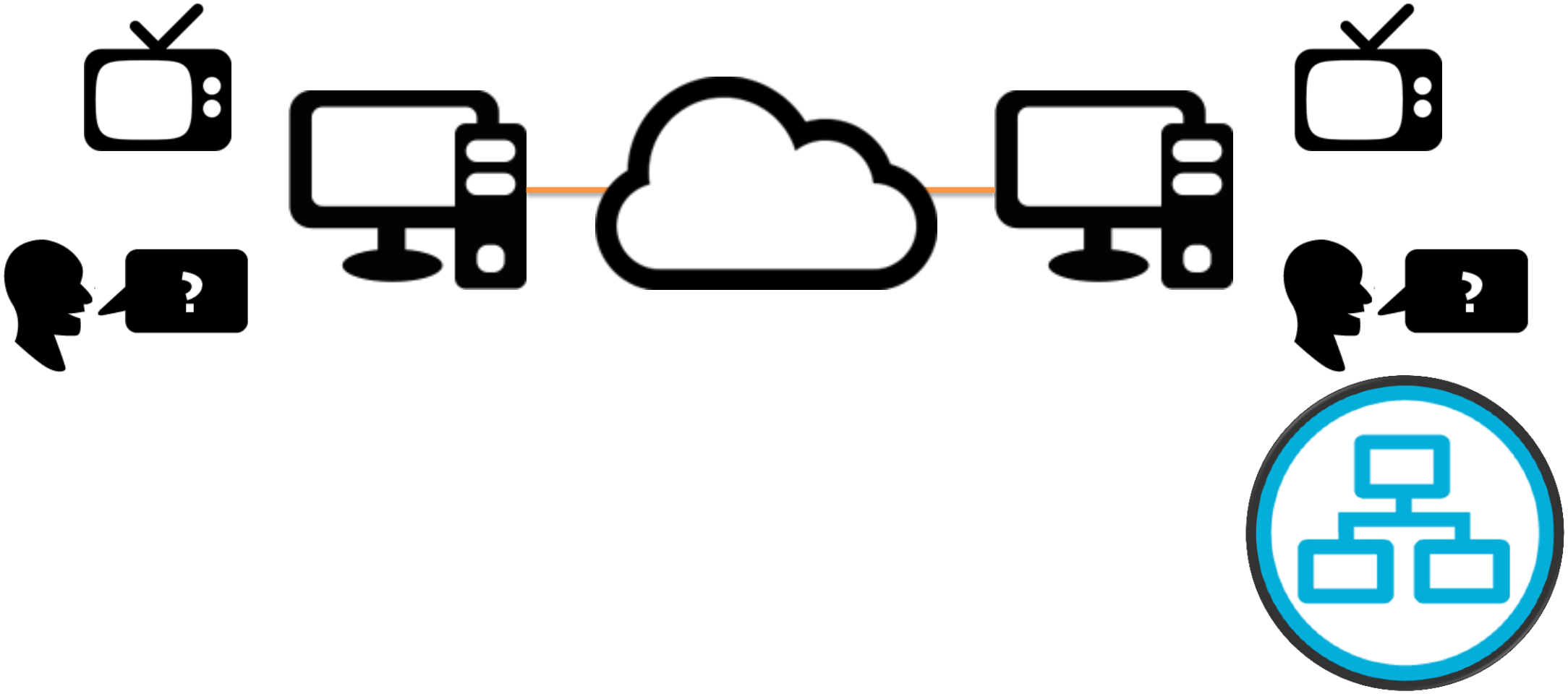
Dynamiczny rozmiar okna

- Sposób kontroli przeciążenia
- Kiedy sieć jest przeciążona, zdarzają się częste retransmisje rozmiar okna zostaje zmniejszony, a co za tym idzie:
  - Wymagane jest częstsze otrzymywanie potwierdzeń
  - Następuje czas oczekiwania na kolejne transmisje
  - Zmniejsza się ilość przesyłanych danych

Dynamiczny rozmiar okna

- Następnie rozmiar okna jest stopniowo zwiększany, aż do momentu zwiększenia liczby retransmisji, po czym jest zmniejszany…
- Jest to proces ciągły

UDP



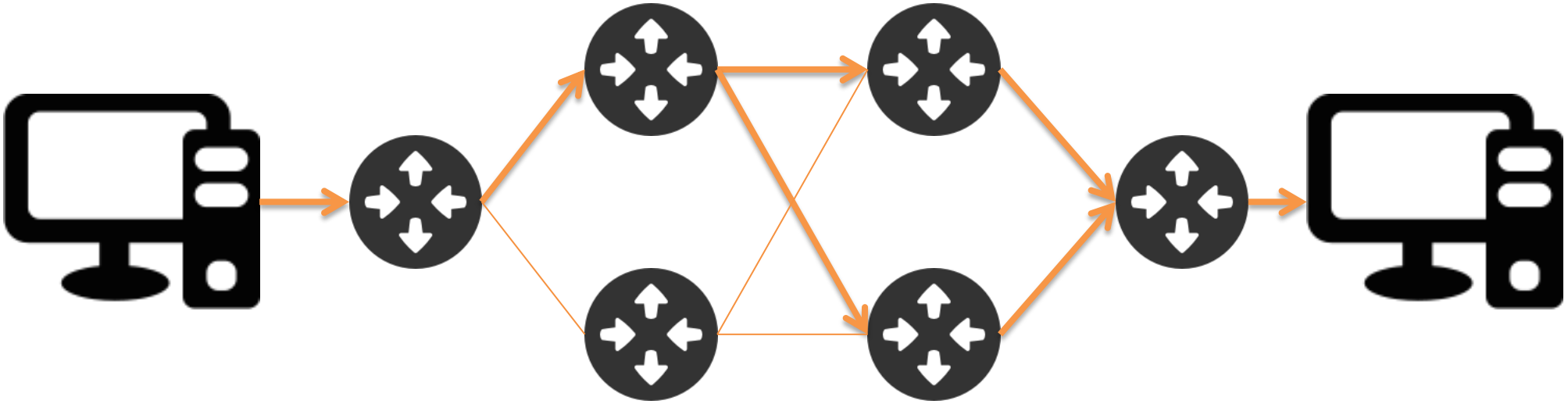
UDP

- Zorientowany bezpołączeniowo
- Mały narzut
- Niewielki nagłówek
- Brak konieczności zarządzania ruchem

Aplikacje korzystające z UDP

- protokół DNS
- protokół SNMP
- protokół DHCP
- protokół RIP
- protokół TFTP
- gry online

Kolejność datagramów



Kolejność datagramów

- Zagubione datagramy nie są retransmitowane
- Datagramy dostarczone w niewłaściwej kolejności nie są układane
- Datagramy są scalane w kolejności otrzymania i przekazywane do warstwy aplikacji
- To warstwa aplikacji „martwi się” o właściwą kolejność

TechnikInformatyk.pl

Wszelkie Uwagi i Spostrzeżenia:

kontakt@technikinformatyk.pl