TO CO ZOSTAŁO ZROBIONE NIEKONIECZNIE ZOSTAŁO ZROBIONE DOBRZE. PROSZĘ JAK CZYTACIE TO SPRAWDZAJCIE PRZY OKAZJI

Kto ogląda ten dokument:
https://www.youtube.com/watch?v=JtTaFpM6SsE
https://www.youtube.com/watch?v=fI6Lc1pU1W8

Ważne polecenia i funkcje systemowe:

ip route - pokazuje tablicę routingu

socket() - tworzy gniazdo i zwraca jego deskryptor
recvfrom() - odbiera kolejny pakiet z kolejki związanej z gniazdem. Standardowe wywołanie blokuje, aż w gnieździe będzie pakiet

todo: przejrzeć listy z pracowni i wypisać polecenia

Sieci

Wykład 1: Wstęp

1. **Co to jest protokół komunikacyjny? Dlaczego wprowadza się warstwy protokołów?**

   Protokół komunikacyjny to zbiór reguł i kroków postępowania, które są automatycznie wykonywane przez urządzenia komunikacyjne w celu nawiązania łączności i wymiany danych. Ustalany na stałe lub na czas danej sesji schemat postępowania. Określa on format danych i funkcję zmiany stanu. Na podstawie protokołu powinno dać się **jednoznacznie** skonstruować program komunikujący się.

   Protokół określa jak wygląda przesyłany strumień danych.

   Przesyłanie danych komputerowych to niezwykle trudny proces, dlatego rozdzielono go na kilka "etapów", czyli warstw. Warstwy oznaczają w istocie poszczególne funkcje spełniane przez sieć.

2. **Wymień warstwy internetowego modelu warstwowego. Jakie są zadania każdej z nich?**
   - **Aplikacji** - zajmuje się specyfikacją interfejsu, który wykorzystuje aplikacja do przesyłania danych w sieci. Odpowiada za protokoły użytkowników (FTP, HTTP, SMTP.

   - **Transportowa** - Segmentuje dane oraz składa je w tzw. strumień. Następuje podział danych na części, które są kolejno numerowane i wysyłane do docelowej stacji, Wykorzystywane protokoły: **TCP** i **UDP**. Oba protokoły stosują kontrolę integralności pakietów (checksum), a pakiety zawierające błędy są odrzucane. Zapewnia **globalne** dostarczenie danych między **aplikacjami**.

   - **Sieci** - Tutaj hula IP. Tu dzieje się routing globalny. Warstwa dysponuje topologią całej sieci. Jedynie zadanie - zapewnienie sprawnej łączności między bardzo odległymi punktami sieci. Protokoły sieci to: (**IPv4**, **IPv6**, **ICMP**, …).
     Zapewnia **globalne** dostarczanie danych pomiędzy **komputerami.**

   - **Łącza danych** - Ma ona nadzorować jakość przekazywanych informacji (nadzór dotyczy wyłącznie warstwy niższej). Zajmuje się pakowanie w ramki i wysyłaniem do warstwy fizycznej, tak aby obniżyć liczbę pojawiających się podczas przekaz błędów. Urządzenia działające w tej warstwie to **most** i **przełącznik**.
     Zapewnia **lokalne** dostarczanie danych pomiędzy **komputerami**.

   - **Fizyczna** - Może wysyłać i odbierać pojedynczy bit (bez weryfikacji poprawności danych).

3. **Jakie warstwy są zaimplementowane na komputerach a jakie na routerach?**
    - Routery - od sieciowej w dół:
        - sieciowa (np. IP),
        - łącza danych (np. Ethernet)
    - Komputery - od warstwy aplikacji w dół:
        - aplikacji (np. HTTP)
        - transportowa (np. TCP)
        - sieciowa (np. IP)
        - łącza danych (np. Ethernet)

    Podział ten wynika z zasady end-to-end.

4. **Czym różni się model warstwowy TCP/IP od OSI?**
    - TCP/IP łączy funkcje warstw prezentacji i sesji w warstwie aplikacji.
    - TCP/IP łączy warstwy łącza danych i fizyczną modelu OSI w jednej warstwie (warstwa dostępu do sieci).
    - TCP/IP wydaje się prostszy, ponieważ ma mniej warstw. Model odniesienia OSI jest mniej skomplikowany; ma więcej warstw, a to pozwala na szybszą współpracę i rozwiązywanie problemów.
    - Protokół TCP/IP to standardy, na których oparty jest Internet, dlatego jest on bardziej wiarygodny. Sieci zazwyczaj nie są budowane w oparciu o protokoły modelu OSI, choć wykorzystuje się go jako przewodnika.

5. **Co jest potrzebne do zbudowania dwukierunkowego niezawodnego kanału?**
    Potwierdzanie dostarczenia pakietów, oraz sprawdzanie ich integralności.

6. **Porównaj wady i zalety przełączania obwodów i przełączania pakietów.**
    - przełączanie pakietów
        - brak gwarancji
        - oczekiwanie pakietów w kolejkach
        - narzut czasowy dla każdego pakietu (nagłówek)
        - efektywne wykorzystanie łącza (statystyczny multipleksing)
            - założenie: różne komputery wykorzystują łącze w innych momentach czasu → lepsze wykorzystanie łącza
        - odporne na awarie: wybierana inne ścieżka routingu
        - prostsze
    - przełączanie obwodów
        - gwarancja przepustowości
        - szybkie transfery danych
        - narzut czasowy na nawiązanie połączenia
        - marnowanie łącza jeśli są przerwy w strumieniu danych
        - wolne odtwarzanie w przypadku awarii
        - skomplikowane

7. **Jakie znasz rodzaje multipleksowania? Po co i kiedy się je stosuje?**

Multipleksowanie (multipleksacja, pol. zwielokrotnianie, ang. multiplexing) – w telekomunikacji metody realizacji dwóch lub większej liczby kanałów komunikacyjnych (np.

telefonicznych) w jednym medium transmisyjnym (np. para przewodów światłowód, powietrze itp.). Użytkownicy tych kanałów nie powinni odczuwać, że współdzielą medium transmisyjne. Multipleksowanie pozwala ograniczyć liczbę stosowanych mediów transmisyjnych, zwłaszcza kabli.
Rodzaje:

- Z podziałem czasu TDM - time division multiplexing
- Z podziałem częstotliwości FDM - frequency division multiplexing

8. **Porównaj ze sobą rodzaje routingu.**
    - routing źródłowy
        - nagłówek zawiera całą trasę do celu
    - routing wykorzystujący tablice routingu
        - zbiór reguł typu: “jeśli adres docelowy pasuje do wzorca A, to przekaż pakiet do sąsiedniego routera X
    - wirtualne przełączanie obwodów
        - nadawca najpierw wysyła pakiet kontrolny ustanawiający ścieżkę do celu i konfigurujący routery, czasem rezerwujący część łącza
        - kolejne pakiety przesyłane tą ustaloną ścieżką
    - przełączanie pakietów?

9. **Do czego służy polecenie traceroute? Co pokazuje?**
   Wysyła ICMP stopniowo zwiększając TTL aż uzyska ECHO REPLY od routera docelowego. Wcześniej uzyskuje DESTINATION UNREACHABLE.
   Pokazuje jaką trasę musiał pokonać pakiet do hosta docelowego.

10. **Po co stosuje się bufory w routerach? Co to jest przeciążenie?**
    Bufory stosuje się gdy do routera dochodzi dużo pakietów, np. z 2 interfejsów i muszą zostać wepchnięte w jeden wyjściowy. Wtedy pakiety ustawiane są w kolejne i w danym kanale multipleksowane.
    Może dojść do przeciążenia, gdy pakietów jest bardzo dużo i router wolniej je wysyła niż odbiera. Wtedy dodatkowe są odrzucane.

11. **Jakie są przyczyny opóźnień pakietów?**
    - Fizyczne ograniczenia w sieciach.
    - Oczekiwanie pakietów w kolejkach w routerach.

12. **Co to jest BDP? Co to jest czas propagacji?**
    - **BDP** - bandwidth delay product, iloczyn przepustowości i RTT, “ile danych może pomieścić kanał”.
    - **Czas propagacji** - ile czasu podróżuje sygnał między końcami kanału.
    - **Przepustowość** - ile możemy zapisać do kanału na jednostkę czasu.
    - **RTT** - round trip time, 2x czas propagacji

13. **Wyjaśnij pojęcia: komunikacja simpleksowa, półdupleksowa, pełnodupleksowa.**
    - **simplexowa** - odbiorca i nadawca nie mogą zamienić się funkcjami. Komunikacja w jedną stronę.

- **półduplexowa** - odbiorca i nadawca mogą zamienić się funkcjami, ale nie mogą jednocześnie przesyłać danych.
- **pełnoduplexowa** - informacje przesyłane w obie strony jednocześnie bez stałego podziału na odbiorcę i nadawcę.

14. **Co umożliwia protokół IP? Co to znaczy, że protokół realizuje zasadę *best effort*?**
    Globalne, bezgrawancyjne dostarczenie danych. Best effort, czyli dokłada wszelkich starań żeby pakiet doszedł, ale jak nie dojdzie to nic z tym nie robi.

15. **Jakie są zalety i wady zasady end-to-end?**
    Wszystkie dodatkowe cechy (np. niezawodne przesyłanie danych) implementowane w urządzeniach końcowych (komputerach) → łatwa ewolucja, niski koszt innowacyjności.

16. **Po co wprowadza się porty?**
    Aby na danej maszynie zidentyfikować aplikację i móc jej dostarczyć dane.

17. **Wyjaśnij pojęcie enkapsulacji i dekapsulacji.**
    Enkapsulacja / dekapsulacja: proces dodawania / usuwania nagłówka przy przechodzeniu przez kolejną warstwę.

Wykład 2: Routing (adresowanie)

1. **Z czego wynika hierarchia adresów IP? Jaki ma wpływ na konstrukcję tablic routingu?**
   Każdy prefiks adresu określa nam jakąś sieć. Dzięki takiemu podejściu router nie musi wiedzieć o każdym urządzeniu w sieci, a jedynie o routerze odpowiadającym za jakąś sieć. Zawsze staramy się wysłać do najmniejszej sieci zawierającej jakiś adres, zatem układamy adresy od najdłuższej maski do najkrótszej.

2. **Notacja CIDR.**
   Notacja CIDR (Classless Inter-Domain Routing): opisuje zakres adresów IP posiadających wspólny prefiks za pomocą pary (pierwszy adres z zakresu, długość prefiksu).

3. **Co to jest adres rozgłoszeniowy?**
   Ostatni adres z danej podsieci. Po wysłaniu pakietu na ten adres zostanie on powielony i przekazany do każdego urządzenia w sieci

4. **Co to jest maska podsieci.**
   Maska podsieci to maska bitowa służąca do określenia do jakiej sieci należy adres. W notacji CIDR podaje się ją jako długość prefiksu.

5. **Opisz sieci IP klasy A, B i C.**
   Kiedyś tak było teraz tego nie ma, bo zabrakłoby adresów.
   - A - wielkie sieci - adres zaczyna się od 0 (binarnie) -> maska podsieci /8
   - B - średnie sieci - adres zaczyna się od 10 (binarnie) -> maska podsieci /16
   - C - mniejsze sieci - adres zaczyna się od 110 (binarnie) -> maska podsieci /24

6. **Co to jest pętla lokalna (*loopback*)?**
   - Interfejs sieciowy lo (loopback) = sieć **127.0.0.0/8**
   - Łącząc się z dowolnym adresem z tej sieci łączymy się z lokalnym komputerem
   - używany do testowania aplikacji

7. **Do czego służy pole TTL w pakiecie IP? Do czego służy pole protokół?**
   TTL - time to live. Ile hopów może wykonać pakiet zanim się go ubije. Zapobiega pętleniu się pakietów. Pole protokół służy do wyspecyfikowania enkapsulowanego w IP protokołu.

8. **Jakie reguły zawierają tablice routingu?**
   To nie to samo co tablica przekazywania! Zawiera informacje o tym jaka jest odległość do danego routera.

   Tablica przekazywania zawiera informacje, do którego routera przesłać jaki pakiet.

9. **Na czym polega reguła najdłuższego pasującego prefiksu?**
   Jeśli pasuje więcej niż jedna reguła, to wybierana jest ta, która jest najdłuższym prefiksem (najbardziej konkretna reguła).

10. **Co to jest trasa domyślna?**
    Router do którego wysyłamy pakiety niepasujące do żadnej innej reguły z tablicy przekazywania.

11. **Do czego służy protokół ICMP? Jakie znasz typy komunikatów ICMP?**
    Protokół pomocniczy warstwy trzeciej. Jego pakiety enkapsulowane w IP (stanowią pole danych w pakiecie IP).
    Głównie służy do diagnostyki. Echo request/reply. destination unreachable

12. **Jak działa polecenie ping?**
    Wysyła ICMP ECHO REQUEST. czeka na ECHO REPLY.

13. **Jak działa polecenie traceroute?**
    Wysyła ICMP request stopniowo zwiększając TTL aż uzyska ECHO REPLY od routera docelowego. Routery po drodze odsyłają (nie zawsze) informację o śmierci pakietu, co pozwala na prześledzenie trasy którą musiał pokonać pakiet do routera docelowego.

14. **Dlaczego do tworzenia gniazd surowych wymagane są uprawnienia administratora?**
    Jest to zabezpieczenie przed niepowołanym dostępem. Proces, który ma dostęp do gniazd surowych może podsłuchiwać komunikację innych procesów i użytkowników korzystających z tego interfejsu.
    Gniazdo surowe otrzymuje kopię wszystkich pakietów danego protokołu.
+ w gniazdach surowych można podawać dane do umieszczenia bezpośrednio w danych pakietu IP.

15. **Co to jest sieciowa kolejność bajtów?**
    Big-endian. 0xABCD1234 będzie zapisane 0xAB, 0xCD, 0x12, 0x34

16. **Co robią funkcje socket(), recvfrom() i sendto()?**
    - **socket** tworzy gniazdo.
    - **recvfrom** - odbiera kolejny pakiet z kolejki związanej z gniazdem
    - **sendto** wysyła pakiet przez gniazdo

17. **Jakie informacje zawiera struktura adresowa sockaddr_in?**

```
struct sockaddr_in {
    short           sin_family;    // e.g. AF_INET
    unsigned short  sin_port;      // e.g. htons(3490)
    struct in_addr  sin_addr;      // see struct in_addr, below
    char            sin_zero[8];   // zero this if you want to
};

struct in_addr {
    unsigned long s_addr;  // load with inet_aton()
};
```

18. **Co to jest tryb blokujący i nieblokujący?**
    - **tryb blokujący** - Jeśli nie ma pakietu do odebrania to wątek zostanie zablokowany (uśpiony) aż coś przyjdzie.
    - **tryb nieblokujący** - Jeśli nie ma pakietu to metoda zwróci odpowiedni błąd (EWOULDBLOCK). Użyteczne przy wykorzystaniu metody select do odczekiwania co najwyżej X czasu, albo gdy chcemy mieć aktywne czekanie

19. **Jakie jest działanie funkcji select()?**
    Blokuje aż któryś z deskryptorów będzie gotowy do odczytu lub upłynie określony czas

Wykład 3: Routing (tworzenie tablic)

1. **Co to jest cykl w routingu? Co go powoduje?**
   Cykl powoduje, że pakiety krążą w kółko. Powodów może być wiele: informacja o awarii jeszcze się nie rozprzestrzeniła, błąd konfiguracji czy *corner case* w protokole RIP (więcej).

2. **Czym różni się tablica routingu od tablicy przekazywania?**
   Tablica routingu służy do utrzymywania dynamicznego routingu (ilość hopów do danego miejsca). Tablica przekazywania pozwala dopasowywać dany pakiet do danego interfejsu / portu wyjściowego routera (czyli odpowiednio przesłać go dalej).

3. **Dlaczego w algorytmach routingu dynamicznego obliczamy najkrótsze ścieżki?**
   Najkrótsze bo chcemy optymalizować ilość hopów na ścieżce czyli ~czas propagacji. Jeśli mamy najkrótsze ścieżki to nie mamy cykli (w teorii, patrz protokół RIP).

4. **Co to jest metryka? Jakie metryki mają sens?**
   Ustalamy co to znaczy, że jakaś ścieżka jest lepsza/gorsza od innej.
   Wprowadzamy miarę na krawędziach/ścieżkach.
   Np:
   - czas propagacji
   - koszt pieniężny
   - ilość hopów na trasie
   - fizyczna odległość

5. **Czym różnią się algorytmy wektora odległości od algorytmów stanów łączy?**
   - **Algorytm wektora odległości** - rozproszona implementacja algorytmu Bellmana-Forda. Jest prosta w implementacji, ale ma problemy z poprawnością. Komputery przesyłają całą tablicę routingu ale tylko do sąsiadów.
     powiadamiamy sąsiadów o wszystkim. Przykład: RIP, EIGRP
   - **Algorytm stanów łącz** - zalewanie + lokalny Dijkstra. Jest trudniejszy w implementacji, za to matematycznie prosty. Komputery przesyłają informację o stanie swoich łącz do wszystkich wierzchołków w sieci. Nie sprawdza się w dużych sieciach
     powiadamiamy wszystkich o wszystkim. Przykład: OSPF
     wg slajdów powiadamiamy wszystkich o sąsiadach

6. **Jak router może stwierdzić, że sąsiadujący z nim router jest nieosiągalny?**
   Routery co jakiś czas przesyłają sobie tablice routingu. Jeśli sąsiad jej nie przesyła to może znaczyć, że jest nieosiągalny. Jeżeli są połączone kablem, wiedzą to od razu.

7. **Co to znaczy, że stan tablic routingu jest stabilny?**
   Jeśli sieć nie zmienia się przez pewien czas to:
   - każdy router będzie miał ten sam obraz sieci
   - stworzone tablice przekazywania będą bez cykli w routingu

8. **Jak zalewać sieć informacją? Co to są komunikaty LSA?**
    - **LSA** - (Link State Advertisement) stan pojedynczego łącza. OSPF to wysyła.
    - przesyłane na początku + przy zmianie + co jakiś czas
    - LSA zawiera źródło i numer sekwencyjny
    - po 1h otrzymane LSA wyrzucane z pamięci

9. **Co wchodzi w skład wektora odległości?**
    - Okresowo powiadamiaj sąsiadów o całej swojej tablicy przekazywania
    - Aktualizuj swoją tablicę routingu na tej podstawie
    - tablica routingu = tablica przekazywania + informacja o odległościach do celu

10. **W jaki sposób podczas działania algorytmu routingu dynamicznego może powstać cykl w routingu?**
    A --- B --- C --- D Psuje się łącze między C i D, ale B przekazuje swoją tablicę routingu do C szybciej niż C przekazuje swoją do B. Wtedy C mówi, że do D idzie przez B, a B mówi, że do D idzie przez C. I tu zliczanie do nieskończoności wchodzi w grę.

11. **Co to jest problem zliczania do nieskończoności? Kiedy występuje?**
    - routery zwiększają znaną odległość do jakiegoś D średnio o 1 na turę
    - pytanie 10. ^
    - **<prezentacja 3, slajd 50>**

12. **Na czym polega technika zatruwania ścieżek?**
    Chodzi o to, żeby zaradzić zliczaniu do nieskończoności.
    - jeśli X jest wpisany jako następny router na ścieżce do Y to wysyłamy do X informację “mam do Y ścieżkę nieskończoną”
    - **<prezentacja 3, slajd 57>**

13. **Po co w algorytmach wektora odległości definiuje się największą odległość w sieci (16 w protokole RIPv1)?**
    Rozwiązuje to problem zliczania do nieskończoności w przypadku cyklu w routingu, jak doliczymy do największej odległości to mówimy, że brak połączenia (za daleko). A średnicę sieci szacuje się na około 11.
    RIP stosuje również zatruwanie ścieżek.

14. **Po co stosuje się przyspieszone uaktualnienia?**
    Można tak próbować radzić sobie ze zliczaniem do nieskończoności.

15. **Co to jest system autonomiczny (AS)? Jakie znasz typy AS?**
    System autonomiczny (Autonomous System, AS) to zbiór prefiksów (adresów sieci IP) pod wspólną administracyjną kontrolą, w którym utrzymywany jest spójny schemat trasowania (ang. routing policy). Każdy AS to określona klasa adresów IP – whois. AS to:
        - spójne zarządzanie
        - spójna polityka wewnętrznego routingu, zazwyczaj OSPF
        - dla większych AS, OSPF umożliwia tworzenie dodatkowego poziomu hierarchii

    Typy AS:

- Stub (tylko jedno połączenie na zewnątrz
- Multihomed (wiele połączeń z innymi AS)
- Transit (j.w. + pozwala na routing przez siebie)

16. **Czym różnią się połączenia dostawca-klient pomiędzy systemami autonomicznymi od łącz partnerskich (*peering*)?**
    Peering - zgoda na wzajemne (bezpłatne) przesyłanie danych pomiędzy swoimi sieciami AS. Jest to dla nich opłacalne, bo ściągają pieniądze ze swoich klientów.
    Dostawca-klient - za to płaci klient.

17. **Dlaczego w routingu pomiędzy systemami autonomicznymi nie stosuje się najkrótszych ścieżek?**
    ISP:
    - chcą płacić jak najmniej
    - nie chcą udostępniać szczegółów na temat swoich AS
    - nie chcą, żeby ktoś przesyłał dane przez ich AS jeśli nie mają z tego zysku
    - mają na uwadze inne ekonomiczne względy itd.

18. **Które trasy w BGP warto rozgłaszać i komu? A które wybierać?**
    - zawartość naszego AS (prefiksy CIDR) - inaczej nikt do nas nie trafi
    - trasy do naszych klientów
        - klienci nam płacą za ruch
        - szczególnie warto rozgłaszać partnerom, bo za ten ruch nie płacimy
    - trasy do naszych dostawców
        - naszym klientom tak, bo nam za to płacą
        - innym nie - nie chcemy żeby przez nasz AS przechodził tranzyt ZA DARMO do naszego dostawcy
    - trasy do naszych partnerów
        - naszym klientom tak (znowu, pieniądze)
        - innym zazwyczaj nie

19. **Jak BGP współpracuje z algorytmami routingu wewnątrz AS?**

    Routery brzegowe (via BGP):
    - rozgłaszają prefiksy CIDR tego AS
    - dowiadują się o trasach do innych AS

    AS z jednym wyjściem:
    - router brzegowy ustala routing wewnątrz AS
    - dodaje router brzegowy na wszystkich routerach wewnętrznych jako default gateway

    AS z wieloma wyjściami:
    - ustala routing wewnątrz AS
    - każdy router wybiera najbliższy brzegowy jako default gateway
    - informacja o dostępnych trasach synchronizowana pomiędzy routerami brzegowymi

Wykład 4: Routing (wewnątrz routera)

1. Co to są prywatne adresy IP? Jakie pule adresów są zarezerwowane na takie adresy?

Są to adresy przeznaczone do sieci lokalnych. Pakiety z takimi adresami nie są przekazywane przez routery.

10.0.0.0/8 172.16.0.0/12 192.168.0.0/16 czyli takie, jakie są w 109 (i na Twoim routerze)

2. Co robi funkcja bind()?

```
int bind(int sockfd, const struct sockaddr *addr, socklen_t addrlen);
```

bind() przypisuje adres wyszczególniony przez *addr* do gniazda s*ockfd*
Wiąże gniazdo z adresem i portem.

3. Czym różnią się porty o numerach mniejszych niż 1024 od innych? Co to są porty efemeryczne?

Do portów <= 1024 potrzebne są uprawnienia admina. Porty efemeryczne to takie, którym numer został przydzielony automatycznie.

4. Jakie są zadania procesora routingu, portu wejściowego, portu wyjściowego, struktury przełączającej?

Procesor routingu (część sterująca) - tworzy tablicę routingu.
Port wejściowy - odbiera pakiet i zgłasza przerwanie.
Port wyjściowy - zgłasza przerwanie jak jest wolny; jak trzeba to coś wysyła.
Struktura przełączająca - przekazuje pakiet od portu wejściowego do wyjściowego.

5. Na czym polega przełączanie pakietów za pomocą RAM w routerze od przełączania za pomocą sieci?

RAM: port odbiera pakiet, zgłasza przerwanie, pakiet jest kopiowany do RAM
Sieci: procesor otrzymuje niektóre pakiety (RIP, OSPF), tworzy tablicę przekazywania, wysyła ją do portów wejściowych. One odebrawszy pakiet aktualizują jego nagłówek i sprawdzają do jakiego portu wysłać. Struktura przełączająca to np. sieć Benesa-Waksmana.

6. Jakie są pożądane cechy struktury przełączającej w routerze?

Przekazywanie pakietów z prędkością (zbliżoną) do prędkości łącza.
Pytanie z aisdu: mała głębokosc - O(logn) i rozmiar - O(n log n) i niewielka ilość przełączników.

7. Gdzie w routerze stosuje się buforowanie? Po co?

Przy portach wejściowych, jeśli przychodzi na tyle duży ruch, że sieć przełączająca nie wyrabia.
Także, jeśli następuje przepełnienie na porcie wyjściowym.

8. Po co w portach wyjściowych klasyfikuje się pakiety?

Zapobiega to utracie pakietów przy czasowym zwiększeniu ich liczby.

9. Co to jest blokowanie początku kolejki? Gdzie występuje? Jak się go rozwiązuje?

Przepustowość struktury przełączającej jest zbyt mała → pakiety kierowane do zajętych łącz wyjściowych → łącza są blokowane → pakiety za zablokowanym pakietem czekają.
Rozwiązanie: **wirtualne kolejki**, czyli jedna kolejka do jednego portu wyjściowego.

10. Rozwiń skrót LPM.

Longest Prefix Match. W tablicy routingu: jeżeli więcej niż jedna reguła pasuje, to wybierz tę, która ma najdłuższy prefiks.

11. Jakie znasz struktury danych implementujące LPM? Porównaj je.

	Lista prefiksów	**Tab haszujące**	**Drzewa trie**	**TCAM**
Pamięć	O(n)	O(n)	O(n * w)	
Lookup	O(n)	O(w) oczekujących	O(w)	
Ins/del	O(1)	O(1) oczekujących	O(w)	
		mamy w+1 tablic dla każdej długości prefiksu		

12. Co to jest pamięć TCAM? Jak można ją zastosować do implementacji LPM?

TCAM = ternary content-addressable memory. Stosowane w nowszych (i droższych) routerach. Mamy pary (p, m) i dla adresu ‘w’ równolegle szukamy takich par, że w&m = p&m, sprzętowo wybieramy najdłuższy.

13. Na czym polega fragmentacja IP? Gdzie się ją stosuje i dlaczego? Gdzie łączy się fragmenty?

Jeżeli rozmiar pakietu jest większy niż MTU łącza wyjściowego, to pakiet dzielony jest na fragmenty. Pakiet dzieli się na dowolnym routerze na trasie, a łączy dopiero na urządzeniu docelowym.

Fragmenty dostają:

- identyczny identyfikator
- MF = czy jest więcej fragmentów?
- offset = numer pierwszego bajtu w oryginalnym pakiecie

Oczywiście fragmentacja jest nieefektywna.

14. Co to jest MTU? Na czym polega technika wykrywania wartości MTU dla ścieżki?

Maximum Transmission Unit. Technika ta polega na wykrywaniu najmniejszego łącza na trasie:
a) ustaw bit DF (don’t fragment w nagłówku IP)
b) jeżeli w routerze jest konieczna fragmentacja, to odrzuca pakiet i odsyła ICMP *destination unreachable, cant fragment* z rozmiarem MTU kolejnego łącza.

15. Jak działa szeregowanie pakietów w buforze wyjściowym routera?

Kolejka FIFO. Szeregowanie pakietów: przypisujemy pakiety do strumieni na podstawie adresu i portu źródłowego i docelowego. Pakiety są szeregowane wg. strumienia:

a) względem priorytetów strumieni
b) round robin - po tyle samo pakietów z każdego strumienia

16. Jakie są różnice w nagłówkach IPv4 i IPv6?

IPv6:

a) 8 bloków po 4 cyfry w hex (rozdzielone dwukropkiem)
b) mają nagłówki stałej długości
c) brak fragmentacji
d) brak sumy kontrolnej
e) mają etykietę strumienia - nie trzeba patrzeć na porty

17. Zapisz adres IPv6 0321:0000:0000:0123:0000:0000:0000:0001 w najkrótszej możliwej postaci.

8 bloków po 4 16-stkowe cyfry.

Uproszczenia zapisu:

1) można opuszczać wiodące zera w każdym bloku
2) **Jeden** ciąg zerowych bloków można zamienić na **::**

321:0:0:123::1

18. Co to jest tunelowanie 6in4?

Mechanizm migracji do IPv6: pakiety IPv6 przesyłane są jako dane pakietów IPv4 tam, gdzie IPv6 jest nieobsługiwane. *Bonus*: broker to urządzenie, które wyciąga/pakuje IPv6 w IPv4

19. Na czym polega NAT i po co się go stosuje? Jakie są jego zalety i wady?

Adresy IPv4 się wyczerpują i są one drogie. Robimy sieć i łączymy ją z Internetem routerem NAT. Internet widzi tę sieć jako pojedynczy komputer z adresem IP routera. Jak komunikujemy się ze światem? Komputer z sieci wysyła pakiet, który przechodzi przez router NAT. W routerze adres źródłowy i port pakietu zostają podmienione na te z routera. W powracającym pakiecie adres docelowy jest podmieniany na ten komputera wysyłającego pierwszy pakiet; pakiet wędruje w głąb sieci.

20. **Jaki stan musi przechowywać router z funkcją NAT**?

Przechowuje *tablicę NAT*, a w niej przypisanie:

(adres źródłowy, port źródłowy, adres docelowy, port docelowy) → port w routerze NAT

Jak router NAT odbierze pakiet z jakimś portem, to szuka w tej tablicy i odpowiednio podmienia adres i port docelowy.

Wykład 5: Niższe warstwy

1. **Jakie są zadania warstwy łącza danych a jakie warstwy fizycznej?**
    a. warstwa łącza danych
        i. umożliwia komunikację między dwoma bezpośrednio połączonymi urządzeniami
        ii. zapewnia zawodną usługę wysyłania ramek
        iii. kanał może być współdzielony przez wiele urządzeń
        iv. musi radzić sobie z błędami transmisji
    b. warstwa fizyczna
        i. określa szczegóły przesyłania pojedynczych bitów
        ii. kodowanie za pomocą sygnałów elektrycznych, fal radiowych, światła itd.

2. **Rozwiń pojęcia LAN i WAN.**
    a. **LAN** - (Local Area Network) - sieć lokalna. Ethernet, WLAN
    b. **WAN** - (Wide Area Network) - sieć rozległa

3. **Czym różni się koncentrator od przełącznika sieciowego?**
    a. koncentrator nic nie wie o tym co przesyła, on tylko rozprasza sygnał na wszystkie swoje wyjścia
    b. przełącznik już wie co przesyła (ramki) kieruje je tam gdzie trzeba

4. **Co to jest komunikacja pełnodupleksowa, półdupleksowa i simpleksowa?**
   Było wcześniej.

5. **Jak działa algorytm rundowy i bezrundowy ALOHA?**
    a. rundowy
        i. czas podzielony na rundy; runda wystarcza do nadania jednej ramki
        ii. sukces (ramka słyszana) = dokładnie jeden komputer nadaje
        iii. jeśli komputer ma ramkę danych do wysłania to wysyła ją z ppb. p
        iv. dla p = 1/n gdzie n to #komputerów, sukces średnio co e tur.
    b. bezrundowy
        i. jak poprzednio, ale nie ma globalnego zegara, każdy ma własne rundy

6. **Jak działa algorytm oczekiwania wykładniczego?**
    - Dynamiczne dopasowywanie wartości p.
    - Na początku p = 1
    - Po kolizji p = p/2

7. **Wyjaśnij skróty CSMA/CD i CSMA/CA, opisz te algorytmy**
   TU WARTO PRZECZYTAĆ SLAJDY <prezentacja 5, slajdy od 14>
   Carrier Sense Multiple Access with Collision Detection
    - na początku sprawdź czy kanał jest wolny
    - jeśli kolizja to przestań nadawać
    - reguluj dynamicznie wartość p

- początkowo p = 1
- po kolizji p = p/2

Schemat dla każdej ramki do wysłania:

1. m = 1
2. poczekaj aż kanał będzie pusty i zacznij nadawać
3. jeśli podczas nadawania usłyszysz kolizje to:
    a. skończ nadawać
    b. wylosuj k ze zbioru {0,...,2^(m-1)} i odczekaj k rund
    c. m = m+1
    d. wróć do kroku 2

Czas nadawania ramki powinien być równy co najmniej 2*czas propagacji. Wtedy albo dotrze, albo o kolizji dowiemy się podczas nadawania.

Carrier Sense Multiple Access with Collision Avoidance.

- w bezprzewodowych gdy nie można jednocześnie nadawać i słuchać
- potwierdzanie ramek
- ramki zawsze nadawane do końca
- odczekujemy pewien czas nawet jeśli kanał właśnie się zwolnił

8. Opisz budowę ramki ethernetowej.

a. adres docelowy MAC
b. adres źródłowy MAC
c. typ - definiuje protokół w danych, np. IP
d. dane - MTU - maximum transmission unit
e. suma kontrolna (CRC)

Ethernet definiuje maksymalną odległość w sieci i minimalną długość ramki.

9. Co to jest adres MAC?

Ciąg 6 bajtów przypisany (teoretycznie) na stałe do karty sieciowej.
Pierwsze 3 bajty o producent a reszta to numer karty.

10. Do czego służy tryb nasłuchu (*promiscuous mode*)?

Przestawienie karty sieciowej w ten tryb powoduje, że odbiera ona wszystkie widzialne ramki (tak działa Wireshark)

11. Dlaczego w Ethernecie definiuje się minimalną długość ramki?

- łatwiej odróżnić ją od śmieci
- wysyłanie trwa co najmniej 2*czas propagacji
- żeby wykrywać dobrze kolizje

12. Jak dobierać długość rundy (“odczekiwania”) w protokole CSMA/CD?

<prezentacja 5, slajdy od 19>

Runda musi być dłuższa od 2*czas propagacji

13. Do czego służą protokoły ARP, RARP, DHCP i APIPA?

- **ARP** - (Address Resolution Protocol)

    - rozgłasza zapytania “kto ma dany adres IP”
    - enkapsulowany w ramkach broadcastowych FF:FF:...:FF
    - z reguły jeden komputer odpowiada
    - wszyscy słyszą i zapisują odpowiedź
- **RARP** - (Reverse ARP)
    - MAC -> IP
- **DHCP** - (Dynamic Host Configuration Protocol) pobiera całą konfigurację sieci
- **APIPA** - (Automatic Private IP Addressing) - konfiguracja automatyczna

14. Czym różni się łączenie dwóch sieci za pomocą mostu od łączenia ich za pomocą routera?

Routery przesyłają informacje obwodami międzysieciowymi znacznie szybciej (routery usuwają zewnętrzne warstwy danych, zanim wyślą pakiet z jednej sieci lokalnej do drugiej) i skuteczniej niż mosty. Jeżeli dwie sieci używają tych samych segmentów sieci i protokołów kontroli dostępu np. Ethernet to można połączyć mostami każdą z nich. Jeśli jednak sieci są różne np. jedna wykorzystuje Ethernet, a druga token ring - najlepszym rozwiązaniem będą routery, ponieważ usuną one pakiety sformułowane dla IPX lub IP z ramki niższego poziomu.

15. Jak warstwa łącza danych realizuje rozgłaszanie?

FF:FF:FF:FF:FF:FF to adres rozgłoszeniowy MAC warstwy łącza danych.

16. Na czym polega tryb uczenia się w przełączniku sieciowym?

Przełącznik uczy się który port prowadzi do którego adresu MAC, dzięki czemu może kierować ramki do właściwych portów. Nauczanie skojarzeń następuje algorytmem Transparent Bridging (lub Backward Learning).

Switch ma tablicę par adres MAC–port w której zapamiętuje adresy źródłowe wyciągnięte z ramek. Jeśli docelowy adres MAC nie jest zapisany, ramka jest wysyłana na wszystkie porty (oprócz źródłowego) pod adres rozgłoszeniowy (FF-FF-FF-FF-FF). Przeterminowane adresy są usuwane.

17. Po co w przełączanym Ethernecie stosuje się algorytm drzewa spinającego?

Żeby nie było cykli, jeśli wysyłamy ramkę na adres rozgłoszeniowy.

18. Wyjaśnij zjawisko ukrytej stacji.

Zjawisko ukrytej stacji (ang. *hidden terminal*) może wystąpić, kiedy nie wszystkie stacje sieci mają bezpośrednią łączność. Stacja jest ukryta, jeżeli znajduje się w zasięgu stacji odbierającej dane, ale jest poza zasięgiem stacji nadającej. Stacja A nadaje do stacji B. Ponieważ stacje A i C znajdują się poza swoim zasięgiem, transmisja ta nie zostanie wykryta w stacji C, która wobec tego przyjmuje, że łącze jest wolne i może rozpocząć transmisję do stacji B lub D. Transmisja ta powoduje w stacji B kolizję z danymi ze stacji A, co powoduje spadek ogólnej przepustowości łącza wskutek konieczności retransmisji.

19. Na czym polega rezerwowanie łącza za pomocą RTS i CTS?

Request to send, clear to send.

- B chce wysłać do C
- B wysyła RTS (Request to send), słyszy to też A

- C wysyła CTS (Clear to send), słyszy to też D
- A i D będą milczeć (NAV - Network Allocation Vector)
- B wysyła dane, C potwierdza

20. Jakie znasz problemy z warstwą fizyczną w sieciach przewodowych i bezprzewodowych?

Bezprzewodowe:

- interferencje - wzajemne zakłócenia urządzeń pracujących z takimi samymi częstotliwościami (inne karty sieciowe, kuchenki mikrofalowe, ...)
- malejąca siła sygnału - sygnał rozpraszany, słabnie przy przechodzeniu przez ściany
- propagacja wielościeżkowa - ten sam sygnał wędruje do celu ścieżkami różnej długości
- półduplex - nie można jednocześnie nadawać i słuchać (tak jak w Ethernecie), nie wiemy czy wystąpiła kolizja (CSMA/CD bezużyteczne).

Przewodowe:

- Jeżeli kilka komputerów wpiętych w jeden kabel -> kolizje
- Jeżeli kable nieekranowane -> oddziaływania elektromagnetyczne i przekłamania bitów
- Uszkodzenie fizyczne kabli -> przesuwanie biurek po kablach

21. Jakie znasz standardy szyfrowania w sieciach bezprzewodowych?

- WEP
- WPA2/PSK
- WPA2/ENTERPRISE (np. eduroam)

22. Wymień popularne standardy Ethernetu i sieci WLAN

802.3 – Ethernet
802.11 – WLAN

Wykład 6: Transport (niezawodny transport)

1. **Co może stać się z przesyłanym ciągiem pakietów IP podczas zawodnego i niezawodnego transportu?**

   Zawodny:
   - uszkodzone
   - zgubione
   - opóźnione
   - zamienione w kolejności
   - zduplikowane (przez niższe lub wyższe warstwy)

   Niezawodny: to samo ale wykrywamy problemy i je poprawiamy lub ignorujemy.

2. **Co to jest kontrola przepływu?**

   Dostosowanie prędkości wysyłania do prędkości odczytu przez odbiorcę, poprzez zmianę wielkości okna.

3. **Czym różnią się protokoły UDP i TCP? Podaj zastosowania każdego z nich.**

   UDP wysyłamy datagram, zawodny w sensie transportu, ale znana, stała wielkość.
   Stosowane przy:
   - przesyłanie małych ilości danych (DNS, DHCP)
   - Proste, ograniczone obliczeniowo urządzenia (np. TFTP wykorzystywany do aktualizacji firmware).
   - Konieczna jest szybka reakcja (gry).
   - Utrata pojedynczych datagramów nieistotna (transmisje multimedialne).
   - Chcemy pełnej kontroli nad przesyłanymi danymi (NFS).

   TCP wysyłamy strumień danych (w segmentach). Protokół zapewnia, że dane będą wysłane poprawnie. Zastosowania:
   - Przesyłane są duże ilości danych (np. HTTP, FTP, sieci P2P).
   - Istotne potwierdzanie danych: praca zdalna (np. SSH)

4. **Co to jest segmentacja? Dlaczego segmenty mają ograniczoną wielkość?**

Podział pakietu na mniejsze części; górnym ograniczeniem jest MTU. Mniejsze pakiety powodują mniejsze opóźnienia.
MSS = MTU - rozmiar nagłówka IP - rozmiar nagłówka TCP
Większe pakiety mają problemy z zakłóceniami przy transmisji bezprzewodowej

5. **Jak nazywają się jednostki danych przesyłane w kolejnych warstwach?**

Warstwa fizyczna: strumień bitów
Warstwa łącza danych: ramki
Warstwa sieci: pakiety
Warstwa transportowa: Segmenty (TCP), datagramy (UDP)

6. **Rozwiń i wytłumacz skrót MSS.**

   maximum segment size

7. **Jak małe pakiety zmniejszają opóźnienie przesyłania danych?**

<prezentacja 6, slajd 16>

8. **Wytłumacz znaczenie skrótów RTT i RTO. Na jakiej podstawie ustalana jest wartość RTO?**

RTT - Round Trip Time - czas jaki zajmuje pakietowi dojście do końca sieci i zpowrotem
RTO - Retransmission Timeout - określa jak długo TCP czeka na odpowiedź ACK

9. **Jak protokoły niezawodnego transportu wykrywają duplikaty pakietów i potwierdzeń?**

Poprzez numery sekwencyjne???

10. **Opisz algorytm Stop-and-Wait. Jakie są jego wady i zalety?**

Odbiorca czeka na pakiety. Jeżeli dostanie następny w kolejności segment, to wysyła potwierdzenie i przekazuje dane do aplikacji.
Nadawca: jeśli chce coś wysłać, to wysyła segment i oczekuje na potwierdzenie ACK. Jeżeli go nie otrzyma, to po pewnym czasie wysyła segment ponownie.
Zalety: banalny w implementacji.
Wady: algorytm jest bardzo wolny; jeżeli ACK zostanie utracone lub opóźnione, to nadawca ponownie wyśle segment - odbiorca nie ma jak stwierdzić, że jest to duplikat. (Slajdy 22-23)

11. **Do czego służa numery sekwencyjne w niezawodnym protokole transportowym?**

Każdy segment jest numerowany i każde ACK zawiera numer potwierdzonego segmentu. Jak ACK zostanie utracone i nadawca ponownie wyśle segment, to odbiorca wie, że już go ma; odsyła ponowne jego potwierdzenie, aby nadawca nie spamował już tym segmentem.

12. **Opisz algorytm okna przesuwnego.**

SWS (Sender Window Size) - maksymalna liczba wysłanych i niepotwierdzonych segmentów. Wysyłamy do skutku wszystkie pakiety znajdujące się w oknie rozmiaru SWS. Jeżeli otrzymamy ACK dla pakietu LAR+1 (Last ACK Received), to przesuwamy okno.

13. **Jaki jest związek między rozmiarem okna a BDP (bandwidth-delay product)?**

BDP - ile danych mieści się na łączu. Nadawca wysyła przez czas RTT (dopóki nie dostanie odpowiedzi żadnej to wysyła z okna na pałę) maksymalnie tyle danych, ile wynosi BDP.
Czyli im większe BDP to większe okno.

14. **Opisz i porównaj następujące mechanizmy potwierdzania: Go-Back-N, potwierdzanie selektywne, potwierdzanie skumulowane.**

Go-Back-N:
Odbieramy i wysyłamy ACK tylko jeśli przyszedł kolejny pakiet w kolejności.
Selektywne:
Od razu wysyłamy ACK do segmentów należących do okna. Dla okna rozmiaru 1 jest równoważne Go-Back-N.
Skumulowane:
Nie od razu wysyłamy ACK, czekamy, bo może uda się wysłać z innymi danymi. To ogranicza ilość wysłanych potwierdzeń.

15. **Dlaczego istotne jest potwierdzanie odbioru duplikatów segmentów?**

Ponieważ nadawca nie wie, że wysyła duplikat, więc mówimy mu by przestał.

16. **Co to jest okno oferowane? Jak pomaga w kontroli przepływu?**

Oferowane okno to ilość wolnego miejsca w buforze odbiorcy, wysyłana do nadawcy. Dzięki temu wie on ile danych jest sens wysłać. Rozmiar okna nadawcy jest równy oferowanemu oknu odbiorcy.

17. **Jakie mechanizmy niezawodnego transportu i kontroli przepływu implementowane są w protokole TCP?**

Potwierdzanie skumulowane. Numerujemy bajty a nie segmenty.
Potwierdzanie bajtów, a nie segmentów. (ACK(n) -> mam wszystko do n-1 bajtu włącznie).
Pozwala na osobne wysłanie oferowanego okna.

18. **Na czym polega opóźnione wysyłania ACK w protokole TCP?**

Jako, że ACK wysyłamy razem z danymi w drugą stronę to, jeżeli nie mamy pakietów do wysłania, to wysłanie ACK będzie opóźnione. Jest to potwierdzanie skumulowane.

19. **Na czym polega mechanizm Nagle'a? Kiedy nie należy go stosować?**

Kiedy mamy do wysłania mniej niż MSS to czekamy aż wszystkie poprzednie pakiety zostaną potwierdzone. Jednak przy aplikacjach interaktywnych, powodowane przez to opóźnienia nie są pożądane.

20. **Co oznaczają pola ,,numer sekwencyjny'' i ,,numer potwierdzenia'' w nagłówku TCP?**

Numer sekwencyjny to numer pierwszego bajtu w wysłanym pakiecie.
Numer potwierdzenia to numer ostatniego bajtu z prefiksu otrzymanych danych.

21. **Czy warstwa transportowa implementowana jest na routerach? Dlaczego?**

Niby nie, ponieważ nawet jeśli zachowamy niezawodny transport pomiędzy routerami to i tak błąd może zajść wewnątrz nich, więc trzeba sprawdzać poprawność na końcach. Ale przy komunikacji bezprzewodowej traci się 20-80% pakietów, więc TCP działałoby bardzo słabo.

22. **Sformułuj słabą i silną zasadę end-to-end.**

Słaba: Końce zawsze sprawdzają, ale pośrednicy w warstwach niższych też mogą.
Silna: Końce zawsze sprawdzają, a pośrednicy w warstwach niższych nie powinny.

Wykład 7: Transport (protokół TCP)

1. **Co to jest gniazdo?**

Reprezentuje dwukierunkowy punkt końcowy połączenia. Dwukierunkowość oznacza możliwość wysyłania i przyjmowania danych. Gniazdo posiada trzy główne właściwości: typ gniazda identyfikujący protokół wymiany danych, lokalny adres (np. adres IP), opcjonalny lokalny numer portu identyfikujący proces, który wymienia dane przez to gniazdo. Na czas trwania komunikacji może posiadać dodatkowe dwa atrybuty: adres zdalny (ponownie np. adres IP), opcjonalny numer portu identyfikujący zdalny proces (jeśli typ gniazda pozwala używać portów).
Adres IP wyznacza węzeł w sieci, numer portu określa proces w węźle, a typ gniazda determinuje sposób wymiany danych.

2. **Czym różni się gniazdo nasłuchujące od gniazda połączonego? Czy w protokole UDP mamy gniazda połączone?**

Wydaje mi się, że to co poniżej to ma zastosowanie do TCP, niekoniecznie do UDP.
Gniazda UDP są bezpołączeniowe i bezstanowe. Nie ma różnicy między klientem a serwerem.

> Gniazdo nasłuchujące nie jest końcówką żadnego połączenia. Nie można przez nie przesyłać danych. Służą one do przyjmowania żądań połączenia, dlatego gniazdko takie nazywa się pasywnym (biernym) - nie robi ono nic, poza oczekiwaniem, aby zestawić połączenie.
> Gniazdo połączone - w momencie gdy przychodzi żądanie połączenia, na gnieździe nasłuchującym przeprowadzana jest operacja, która tworzy NOWE gniazdo połączone (może ono wysyłać i odbierać komunikaty), reprezentujące połączenie z klientem. Warto zauważyć, że gniazdo nasłuchujące nadal istnieje i oczekuje na połączenia.

3. **Co robią funkcję jądra bind(), listen(), accept(), connect()?**
   - **Bind()**: służy do przypisania adresu (adresu węzła i numeru portu) do podanego gniazda. Funkcja ta musi być wywołana przez serwer zarówno w trybie połączeniowym jak i bezpołączeniowym. Może ją także wywołać klient jeśli chce używać do komunikacji konkretnego portu, a nie portu przydzielonego automatycznie przez system.
   - **Listen()**: sygnalizuje gotowość do przyjmowania żądań nawiązania połączenia, wysyłanych przez klientów.
   - **Accept()**: wywoływana w celu przyjęcia żądania nawiązania połączenia, zgłoszonego wcześniej (po wywołaniu funkcji listen()) i oczekującego w kolejce. Jeżeli żadne żądanie nie dotarło, serwer jest blokowany do momentu otrzymania żądania. Po przyjęciu żądania funkcja tworzy nowy deskryptor dla danego gniazda, który może być następnie wykorzystywany przez proces obsługi zgłoszenia.
   - **Connect()**: wywołana dla gniazda obsługiwanego przez połączeniowy protokół transportowy, podejmuje automatycznie próbę nawiązania połączenia, zgodnie z dowiązanym adresem. Może być ona jednak użyta również w trybie bezpołączeniowym. Wówczas, gdy dzięki wywołaniu connect() do gniazda zostanie dowiązana struktura adresowa; przy wysyłaniu datagramu nie jest

potrzebne podawanie adresu i można używać funkcji write() lub send() zamiast sendto().

4. **Czym różni się komunikacja bezpołączeniowa od połączeniowej?**
   - Połączeniowa np. TCP, tworzony jest kanał komunikacyjny, dzięki czemu kolejne wiadomości łatwiej jest przesyłać. Na koniec trzeba zamknąć połączenie
   - Bezpołączeniowa np. UDP każda wiadomość jest niezależna, nie ma związania między stronami komunikacji, nie utrzymują one stanu.

5. **Czym różni się otwarcie bierne od otwarcia aktywnego? Czy serwer może wykonać otwarcie aktywne?**

Otwarcie bierne:

- serwer wykonuje otwarcie bierne tworząc gniazdo nasłuchujące

Otwarcie aktywne:

- klient wykonuje otwarcie aktywne wysyłając segment SYN zawierający numer początkowy
- serwer potwierdza przyjęcie segmentu SYN i wysyła własny segment SYN zawierający początkowy numer danych, które będzie wysyłał przez to połączenie, wraz z segmentem ACK - segment SYN/ACK
- klient sygnalizuje odebranie odpowiedzi wysyłając segment ACK

6. **Do czego służą flagi SYN, ACK, FIN i RST stosowane w protokole TCP?**

SYN - używana przy nawiązywaniu połączenia, ustala początkowy numer sekwencyjny (synchronizuje kolejne numery sekwencyjne)
ACK - flaga mająca na celu potwierdzić odebranie odpowiednich danych (np. pakietu z flagą SYN lub FIN)
FIN - flaga używana przy kończeniu połączenia
RST - oznacza wystąpienie błędu, kończy połączenie. Wysyłana np. przy próbie połączenia z portem na którym nikt nie nasłuchuje (resetuje połączenie, wymagane ponowne uzgodnienie sekwencji)

7. **Opisz trójstopniowe nawiązywanie połączenia w TCP. Jakie informacje są przesyłane w trakcie takiego połączenia?**

K1: strona chcąca nawiązać połączenie wysyła segment SYN
K2: druga strona odsyła pakiet z ustawionymi flagami SYN oraz ACK
K3: strona inicjująca połączenie wysyła pakiet z ustawioną flagą ACK

8. **Dlaczego przesyłanych bajtów nie numeruje się od zera?**

(niepewne) Żeby nie można było “przechwycić” otwarcia, klient losuje pierwszy numer sekwencji (wiki).

(wykład 6): Bo ***ACK n*** oznacza: “mam wszystko do bajtu n-1 włącznie” (potwierdzenie skumulowane)

9. **Jakie segmenty są wymieniane podczas zamykania połączenia w protokole TCP?**

K1: dowolna ze stron wykonuje zamknięcie aktywne wysyłając segment **FIN**

K2: druga ze stron potwierdza odebranie tego komunikatu wysyłając segment **ACK** (zamknięcie bierne) teraz mogą przepływać komunikaty od strony wykonującej zamknięcie bierne do strony wykonującej zamknięcie aktywne
K3: strona wykonująca zamknięcie bierne wysyła segment **FIN**
K4: druga strona wysyła segment **ACK**

10. **Co zwraca funkcja recv() wywołana na gnieździe w blokującym i nieblokującym trybie?**
    recv(_,_,_,0) = read(_,_,_). Przy UDP było recvfrom, przy TCP jest recv()
    send(_,_,_,0) = write(_,_,_). Przy UDP było sendto, przy TCP jest send()
    Recv() – zwraca ilość otrzymanych bajtów (może być mniej, niż podane w argumencie len wywołania), -1 w przypadku błędu, 0 gdy połączenie po drugiej stronie zostało zamknięte

11. **Czy do stanu TIME_WAIT przechodzi strona, która wykonuje zamknięcie aktywne czy bierne? Po co wprowadzono taki stan?**

Strona aktywna po wysłaniu swojego ACK (czyli ostatniego pakietu w połączeniu) nie wie czy ten ACK dotarł, bo druga strona już go o tym nie powiadomi. Dlatego po wysłaniu go wchodzi się na 1-4min w stan TIME_WAIT i czeka ponowne przyjście FIN, kiedy nasze ACK nie dotarło. Do tego służy to usuwaniu starych duplikatów segmentów z sieci. (?)

12. **Na podstawie diagramu stanów TCP opisz możliwe scenariusze nawiązywania i kończenia połączenia.**

Wykład 8: Transport (kontrola przeciążenia)

1. **Czym różni się kontrola przepływu od kontroli przeciążenia**

Kontrola przepływu - nadawca powinien dostosować prędkość transmisji do szybkości z jaką odbiorca może przetwarzać dane. Nie chcemy zalać odbiorcy danymi.

Kontrola przeciążenia - nie chcemy przesyłać za mało, ale też nie za dużo. Chcemy przesyłać tyle, że prawie występuje przeciążenie. Nie chcemy zalać sieci danymi.

2. **Co to jest przeciążenie?**

Sytuacja, w której bufor routera na trasie się przepełni. W tym wypadku nowe pakiety są odrzucane.

3. **Na czym polega mechanizm opóźnionych potwierdzeń?**

- Potwierdzanie każdego segmentu osobno jest nieefektywne
- Potwierdzenia można wysyłać ,,przy okazji“ razem z danymi.
- Jeśli nie ma danych do wysłania, to mechanizm opóźnionych potwierdzeń wymusza upłynięcie pewnego czasu (np. 200 ms) między kolejnymi potwierdzeniami

4. **Jaka jest zależność między rozmiarem okna nadawcy a prędkością transmisji?**

SWS - Sender Window Size - wysyłane przez odbiorcę z każdym ACK. Z regułu większe okno = większa prędkość transmisji. Jeżeli okno jest mniejsze od BDP → nadawca nie jest w stanie wykorzystać całego łącza.

5. **Czy nieskończone bufory rozwiązałyby problem przeciążenia?**

Nie. Opóźnienie jest liniową funkcją rozmiaru kolejki.

6. **Jak zależy średni rozmiar kolejki od średniej prędkości nadchodzenia pakietów?**

Rozmiar kolejki dąży do nieskończoności gdy zbliżamy się do maksymalnej przepustowości (patrz wykresik na slajdzie 14)

7. **Jakie są cele kontroli przeciążenia?**

- Wysokie wykorzystanie łączy
- sprawiedliwy podział łącza
- rozproszony algorytm szybko reagujący na zmieniające się warunki

8. **Jak można definiować sprawiedliwy podział łącza? Co to jest max-min fairness?**

Przypisanie jest max-min fairness jeżeli nie można zwiększyć szybkości żadnego ze strumieni bez spowolnienia innego strumienia, który jest wolniejszy lub tak samo szybki.

9. **Na jakiej podstawie zmienia się rozmiar okna przeciążenia?**

10. **Kiedy TCP wnioskuje, że pakiet zaginął?**

Timeout dla pakietu (nie otrzymano ACK).

11. **Opisz algorytm ustalania rozmiaru okna przeciążenia**

SWS = min {oferowane okno, cwnd}, gdzie:
cwnd = congestion window.
początkowo oferowane okno = inf

12. Rozwiń skrót AIMD. Czego dotyczy?

Additive Increase, Multiplicative Decrease. Dotyczy kontroli przeciążenia.

13. W jaki sposób AIMD gwarantuje sprawiedliwy podział łącza?

Wysłaliśmy pakiet poprawnie? (otrzymaliśmy ACK?): **cwnd←cwnd + 1/cwnd**
Pakiet zgubiony lub opóźniony? (ACK nie dociera przed RTO?): **cwnd ← cwnd/2**
AMID nie kontroluje szybkości wysyłania.
AMID kontroluje liczbę pakietów danego strumienia, która jednocześnie może być w sieci
Przy odpowiednio dużych buforach najbardziej krytyczne łącze jest wykorzystywane w 100%
→ łącze jest dzielone sprawiedliwie

14. Opisz fazy unikania przeciążenia i wolnego startu w TCP.

Wolny start:

a) cwnd = 1
b) po każdym ACK zwiększamy cwnd o MSS (cwnd zwiększy się dwukrotnie co RTT)
c) faza trwa do utraty pierwszego pakietu

W dowolnej fazie: gdy stracimy pakiet:

a) ssthresh ← cwnd /2
b) uruchom fazę wolnego startu gdy cwnd > sstresh

Szczegóły na slajd 35

15. Opisz mechanizm szybkiej retransmisji i szybkiego przywracania.

Szybka retransmisja (wysyłamy brakujący segment bez czekania na timeout).
Szybkie przywracanie (pomijamy fazę krótkiego startu): ssthresh = cwnd / 2; cwnd = ssthresh.

16. Na czym polega mechanizm RED?

Random Early Detection

- ruter na trasie losowo wyrzuca pakiety. Prawdopodobieństwo wyrzucenia jest ustalane jako rosnąca funkcja **średniej** długości kolejki. Nie reaguje na krótkotrwałe zwiększenia kolejki
- krótsze kolejki = mniejsze opóźnienia
- desynchronizacja strumieni - zmniejszenie prędkości w różnych momentach.

17. Opisz działanie mechanizmu ECN (*explicit congestion notification*).

Jeżeli prawdopodobne jest przeciążenie, to router ustawia bity ECN w nagłówku IP.
Odbiorca otrzymawszy takie pakiet ustawia bity ECN w nagłówku TCP potwierdzenia ACK.
Nadawca reaguje tak, jak na utratę pakietu.

18. Jaka jest relacja w AIMD między przepustowością a traconymi pakietami?

Przepustowość jest proporcjonalna do 1/sqrt(p), gdzie p jest frakcją traconych pakietów

19. Jakie modyfikacje wprowadza FastTCP do AIMD? Dlaczego?

Powyżej pewnej wartości **cwnd** zwiększane szybciej i zmniejszane wolniej.

Wykład 9: Zastosowania (część 1)

1. **Jaki jest cel systemu nazw DNS?**
   System nazw DNS istnieje aby ułatwić zapamiętywanie poszczególnych adresów. Ludziom łatwiej jest zapamiętać adres strony pl.wikipedia.org, niż jej adres IP: 145.97.39.135.

2. **Do czego służy plik /etc/hosts?**
   Plik hosts (/etc/hosts – ścieżka do pliku w systemach UNIXowych) jest jednym z modułów wielu systemów operacyjnych, który wspomaga adresowanie w sieciach komputerowych. Jego zadaniem jest tłumaczenie przyjaznych użytkownikom nazw domenowych (kanonicznych) na ich numeryczne odpowiedniki (adresy IP).

3. **Rozwiń skrót TLD (kontekst: DNS), podaj parę przykładów.**
   TLD (Top Level Domain) - domena internetowa powyżej której nie istnieją żadne inne domeny w systemie DNS. Są one tworzone i zarządzane przez IANA i ICANN.
   Każda domena w Internecie składa się z pewnej liczby nazw, oddzielonych kropkami. Ostatnia z tych nazw jest domeną najwyższego poziomu. Na przykład w "pl.wikipedia.org" domeną najwyższego poziomu jest "org".

4. **Czym są strefy i delegacje DNS?**
   Strefa jest najmniejszą jednostką administracyjną DNS. Strefa to nazwa nadana hostom wewnątrz danej domeny z pominięciem wszystkich domen podrzędnych (np. w strefie uni.lodz.pl znajduje się host www.uni.lodz.pl, ale nie ftp.math.uni.lodz.pl). Za daną strefę odpowiada co najmniej jeden serwer nazw (w przypadku „.” Istnieje 13 serwerów głównych). Serwerami dla „pl” rządzi NASK.
   Jest to spójny fragment poddrzewa, identyfikowany przez swój korzeń. Serwer nazw odpowiadający za daną strefę zna zawartość strefy oraz serwery nazw odpowiedzialne za strefy podrzędne (dzięki delegacjom).
   Delegacje to krawędzie między poszczególnymi strefami.

5. **Czym różni się rekurencyjne odpytywanie serwerów DNS od iteracyjnego?**
   Iteracyjne: klient przechodzi drzewo DNS zaczynając od korzenia.
   Rekurencyjne: klient odpytuje serwer DNS, a on w naszym imieniu wykonuje odpytywanie.
   - Windowsowy klient DNS i część systemów uniksowych wymaga takiego serwera
   - Dla poprawy wydajności, serwer zapisuje sobie zwracane wyniki w pamięci podręcznej (odpowiedzi wyświetlane z pamięci podręcznej wyświetlane są jako non-authoritative)

6. **Jak działa odwrotny DNS? Jaki typ rekordów i jaką domenę wykorzystuje?**
   Polega na konwersji odwrotnej: adres IP → nazwa domeny. Zamiast tworzyć kolejny protokół do tego celu, wykorzystuje się możliwości DNS, rekord **PTR**. Wykorzystuje on domenę in-addr.arpa , której poddomenami są klasy lub adresy IP. Przykładowo 222.111.in-addr.arpa opisuje adresy sieci 111.222.0.0/16 .

7. **Jakie znasz typy rekordów DNS? Co to jest rekord CNAME?**
   - **rekord A lub rekord adresu (ang. address record) -** mapuje nazwę domeny DNS na jej 32-bitowy adres IPv4.
   - **rekord AAAA lub rekord adresu IPv6 (ang. IPv6 address record)** - mapuje nazwę domeny DNS na jej 128-bitowy adres IPv6.
   - **rekord CNAME** lub **rekord nazwy kanonicznej (ang. canonical name record)** - ustanawia alias nazwy domeny. Wszystkie wpisy DNS oraz poddomeny są poprawne także dla aliasu.
   - **rekord MX** lub **rekord wymiany poczty (ang. mail exchange record) -** mapuje nazwę domeny DNS na nazwę serwera poczty oraz jego priorytet.
   - **rekord PTR** lub **rekord wskaźnika (ang. pointer record)** - mapuje adres IPv4 lub IPv6 na nazwę kanoniczną hosta.
   - **rekord NS** lub **rekord serwera nazw (ang. name server record)** - mapuje nazwę domenową na listę serwerów DNS dla tej domeny.
   - **rekord SOA** lub **rekord adresu startowego uwierzytelnienia (ang. start of authority record) -** ustala serwer DNS dostarczający autorytatywne informacje o domenie internetowej, łącznie z jej parametrami (np. TTL).
   - **rekord SRV** lub **rekord usługi (ang. service record)** - pozwala na zawarcie dodatkowych informacji dotyczących lokalizacji danej usługi, którą udostępnia serwer wskazywany przez adres DNS.
   - **TXT** – rekord ten pozwala dołączyć dowolny tekst do rekordu DNS. Rekord ten może być użyty np. do implementacji specyfikacji Sender Policy Framework.

8. **Po co są wpisy sklejające w opisie delegacji DNS?**
   Serwery najwyższego poziomu z reguły posiadają tylko odwołania do odpowiednich serwerów DNS odpowiedzialnych za domeny niższego rzędu, np. serwery główne (obsługujące między innymi TLD .com) wiedzą, które serwery DNS odpowiedzialne są za domenę example.com. Serwery DNS zwracają nazwę serwerów odpowiedzialnych za domeny niższego rzędu. Możliwa jest sytuacja, że serwer główny odpowiada, że dane o domenie example.com posiada serwer dns.example.com. W celu uniknięcia zapętlenia w takiej sytuacji serwer główny do odpowiedzi dołącza specjalny rekord (tak zwany glue record) zawierający także adres IP serwera niższego rzędu (w tym przypadku dns.example.com).

9. **Co robi funkcja getaddrinfo()**
   http://man7.org/linux/man-pages/man3/getaddrinfo.3.html

10. **Opisz budowę adresu URL. Opisz budowę adresu URL w przypadku schematu http.**
    Adres URL składa się z dwóch część oddzielonych dwukropkiem:
    - Schemat (http, ftp, mailto, file ...)
    - Część zależna od rodzaju zasobu

    URL w przypadku schematu http składa się z (część po dwukropku):
    - //,
    - Nazwa DNS serwera,
    - Opcjonalnie :port,
    - /,

- Identyfikator zasobu wewnątrz serwera,
- Przykład: http://www.ii.uni.wroc.pl/~mbi/dyd/sieci_13s/.

11. **W jakim celu serwer WWW ustawia typ MIME dla wysyłanej zawartości? Podaj kilka przykładów typów MIME.**

    Internet media type, zwany także **typem MIME** oraz czasem **Content-Type** (po nazwie nagłówka kilku protokołów, którego wartość jest tego typu) jest dwuczęściowym identyfikatorem formatu plików w Internecie. Serwer www ustawia odpowiedni typ MIME w celu określenia rodzaju zasobu, który jest przesyłany.

    Przykłady:
    - text/plain – plik tekstowy
    - text/html – strona HTML
    - image/jpeg – obrazek JPEG
    - video/mpeg – film MPEG
    - application/msword – dokument DOC
    - application/pdf – dokument PDF
    - application/octet-stream – ciąg bajtów bez interpretacji

12. **Wymień parę możliwych odpowiedzi HTTP wraz z ich znaczeniem.**
    - 200 OK – jest ok
    - 301 Moved Permanently - zasób został trwale przeniesiony w inne miejsce
    - 302 Found - plik został znaleziony, ale tymczasowo znajduje się w innej lokalizacji. Żądanie o zasób kończy się więc przekierowaniem.
    - 304 Not Modified - zasób nie uległ zmianie patrząc pod kątem danych, które przekazano w żądaniu o ten zasób.
    - 401 Unauthorized - strumień danych przesłanych przez klienta (np. przeglądarkę internetową) jest prawidłowy i serwer odczytał go poprawnie, lecz źródło URL wymaga autoryzacji danych użytkownika.
    - 403 Forbidden – serwer został znaleziony, lecz jest brak dostępu
    - 404 Not Found – zasobu nie znaleziono
    - 418 I’m a teapot - easter egg (dzięki mazur za info)
    - 500 Internal Server Error – wewnętrzny błąd serwera

1xx: hold on

2xx: here you go

3xx: go away

4xx: you fucked up

5xx: I fucked up

13. **Po co w nagłówku żądania HTTP/1.1 podaje się pole Host?**

    Nagłówek host: służy do rozpoznania hosta, jeśli serwer na jednym IP obsługuje kilka VirtualHostów. Dla przykładowego żądania: <http://www.w3.org/pub/WWW/> pole host będzie zawierało: www.w3.org

14. **Do czego służą pola Accept, Accept-Language, User-Agent, Server, Content-Length, Content-Type w nagłówku HTTP?**
    - **Accept** - Służy do określenia listy akceptowalnych przez **przeglądarkę** typów MIME dokumentu, oraz opcjonalnie hierarchii każdego typu. Liczby podane po ;q= powinny mieć wartości od 0 do 1, co 0.1. Jako separator miejsc dziesiętnych użyta musi być kropka. W przypadku braku zdefiniowanej wartości ;q= przyjmowana jest wartość: ;q=1(równoznaczna z ;q=1.0).
    - **Accept-Language** - Określa w jakim języku użytkownik **przeglądarki** życzy sobie czytać strony
    - **User-Agent** - identyfikuje przeglądarkę
    - **Content-Length** - Długość w bajtach przesyłanej zawartości (wyłączając część nagłówkową). Nagłówek obowiązkowy dla danych wysyłanych z **serwera**, oraz przy korzystaniu z metody POST.
    - **Content-Type** - Tym nagłówkiem serwer informuje przeglądarkę, w jakim formacie i stronie kodowej wysyłany jest dokument.

15. **Jak wygląda warunkowe zapytanie GET protokołu HTTP?**
    A *conditional GET* is an HTTP GET request that may return an HTTP 304 response (instead of HTTP 200). An HTTP 304 response indicates that the resource has not been modified since the previous GET, and so the resource is not returned to the client in such a response.

16. **Jakie znasz kody odpowiedzi protokołu HTTP?**
    Patrz 12

17. **Na czym polegają połączenia trwałe w HTTP/1.1? Do czego służy opcja Connection: close w nagłówku HTTP?**
    Połączenie trwałe - połączenie, które obsługuje kilka zapytań/odpowiedzi i nie zamyka się po każdym pojedynczym zapytaniu.
    Connection: close służy do zamknięcia połączenia.

18. **Do czego służą arkusze stylów CSS?**
    Arkusz stylów CSS to lista dyrektyw (tzw. reguł) ustalających w jaki sposób ma zostać wyświetlana przez przeglądarkę internetową zawartość wybranego elementu (lub elementów) (X)HTML lub XML. Można w ten sposób opisać *prawie* wszystkie pojęcia odpowiedzialne za prezentację elementów dokumentów internetowych, takie jak rodzina czcionek, kolor tekstu, marginesy, odstęp międzywierszowy lub nawet pozycja danego elementu względem innych elementów bądź okna przeglądarki.

19. **Wymień parę możliwości uzyskiwania dynamicznych stron WWW.**
    Dynamika po stronie klienta
    - Javascript: prosty obiektowy interpretowany język, kod programu jest wbudowany w HTML.
    - Aplety Javy, aplikacje Flash, Silverlight (wykonanie realizowane przez odpowiednie wtyczki do przeglądarki)

    Dynamika po stronie serwera

- URI może wskazywać na program, którego wynikiem działania jest HTML (+ewentualnie nagłówek HTTP)
- CGI (Common Gateway Interface): standard umożliwiający wykonanie dowolnego zewnętrznego programu Mechanizmy zintegrowane z serwerem WWW (PHP, JSP, ASP, mod_perl, ...)
- Formularze, przekazywanie parametrów (metody GET i POST)
- Cookies = utrzymywanie stanu sesji

20. Co to jest CGI?

CGI (Common Gateway Interface): standard umożliwiający wykonanie dowolnego zewnętrznego programu. Znormalizowany interfejs, umożliwiający komunikację pomiędzy oprogramowaniem serwera WWW a innymi programami znajdującymi się na serwerze. Zazwyczaj program serwera WWW wysyła do przeglądarki statyczne dokumenty HTML. Za pomocą programów CGI można dynamicznie (na żądanie klienta) generować dokumenty HTML uzupełniając je np. treścią pobieraną z bazy danych. Programy CGI są często pisane w językach interpretowanych takich jak Perl, przez co nazywa się je także skryptami CGI. Tak działa np. (open) Spotify.

21. Po co stosuje się metodę POST?

Z metodą POST mamy do czynienia, gdy w URI nie widać żadnych parametrów. Dane metodą POST przesyłane są w obszarze danych pakietu i umieszczane (uwaga, PHP) w superglobalnej tablicy $_POST (tzn. można się od niej odwołać z dowolnego miejsca skryptu). Jako że użytkownik nie może podejrzeć przesłanych danych, tą metodą przesyłamy np. dane uwierzytelniające. Stosuje się ją także przy wgrywaniu plików.

22. Co to jest technologia REST?

REST (Representational State Transfer) - tworzenie usługi sieciowej wykorzystując metody (GET, PUT, POST, DELETE) protokołu HTTP. REST nie jest standardem, raczej filozofią. Łatwy do zautomatyzowania, czytelny dla człowieka.
Zautomatyzowany dostęp do niektórych serwisów WWW (przykładowo do: eBay, Amazon, Twitter, Flickr, ...).
Jest wykorzystywany przez wiele frameworków aplikacji internetowych np. Ruby on Rails, Sinatra, Django, RESTlet, RESTeasy i wiele innych. Charakterystycznym elementem REST jest "restowy" (RESTful) interfejs usług webowych, w którym parametry wywołania danej usługi są umieszczane w ścieżce adresu URL, a nie w części przeznaczonej na parametry, jak w klasycznych wywołaniach GET lub POST.
Wywołanie klasyczne:
http://example.com/article?id=1234&format=print
Wywołanie RESTful
http://example.com/article/1234/print

23. Jaka jest rola trackera w sieci Bittorrent?

Jest to komputer (serwer) będący nadzorcą działania sieci BitTorrent, czuwa niejako nad całym procesem wymiany plików, podłączają się do niego poszczególni klienci (programy które umożliwiają nam wymianę plików) i ma on za zadanie między innymi przekazywać dane tj. adresy IP pomiędzy osobami pobierającymi plik. Sprawność jego

działania poniekąd wpływa na szybkość z jaką możemy współdzielić pliki. Jego obecność jest konieczna do nawiązania połączenia z inną osobą i możliwości pobrania od tejże osoby jakiegokolwiek pliku.

24. Po co w plikach .torrent stosuje się funkcje skrótu?

Funkcja skrótu, inaczej: funkcja mieszająca lub funkcja haszująca – jest to funkcja, która przyporządkowuje dowolnie dużej liczbie krótką, zwykle posiadającą stały rozmiar, nie specyficzną, quasi-losową wartość, tzw. skrót nieodwracalny.
Info hash - 160-bitowa wartość pochodząca z funkcji skrótu SHA1. Funkcji tej jest podawana część metapliku .torrent zawierająca nazwy plików oraz hasze udostępnianych danych. Możliwa jest zmiana trackera oraz komentarza w pliku .torrent bez zmiany Info hasha.

25. Jakie są różnice w postępowaniu *seedera* i *leechera* w sieci BitTorrent?

Seeder - pobrał cały plik z torrenta i udostępnia całość danych dalej.
Leecher - posiada jedynie fragmenty pliku, które może udostępnić pod warunkiem, że dostanie coś w zamian.

Wykład 10: Zastosowania (część 2)

1. **Na czym polegają połączenia odwrócone? Jak stosuje się je w protokole FTP?**

Klient prosi serwer o przesłanie pliku na jakiś port X (różny od tego, z którego przyszła prośba). Router NAT odrzuci połączenie, bo nie ma portu X w tablicy NAT.
Tryb pasywny: klient zaczyna słuchać na porcie Y i wysyła komunikat “słucham na porcie Y”. A łączy się z serwerem i pobiera plik.

2. **Opisz podobieństwa i różnice asymetrycznych (*cone*) NAT (pełnego, ograniczonego i ograniczonego portowo) i symetrycznych NAT.**

Pełny asymetryczny NAT - przekazuje wszystkie pakiety do kompów w sieci
Ograniczony asymetryczny NAT - przekazuje pakiety tylko z listy odbiorców
Ograniczony portowowo asymetryczny NAT - przekazuje tylko pakiety od par (IP,port) z listy

NAT asymetryczny - port nadany przychodzącemu z wewnętrznej sieci pakietowy zależy tylko od adresu i portu nadawcy.
NAT symetryczny - ten port zależy od adresu i portu nadawcy i odbiorcy. Tu nie działa wybijanie dziur.

3. **Opisz technikę wybijania dziur (*hole punching*) w NAT. Po co konieczny jest serwer pośredniczący?**

jakieś gówno, slajd 23

4. **Do czego służą serwery proxy?**
    - Ograniczanie ruchu do/z zewnętrznych stron WWW i przechowywanie zawartości stron w pamięci proxy.
    - Kontrolowanie dostępu do zasobów WWW.
    - Uwaga: serwer proxy zazwyczaj oznacza serwer proxy WWW, ale można wyobrazić sobie proxy dla wielu innych usług (ARP, DNS, DHCP)

5. **Co to jest odwrotne proxy? Co to jest CDN?**

Odwrotne proxy wykorzystywane jest przez dostawców treści: serwer taki jest na drodze serwera www do klienta. Klient łącząc się z serwerem www natrafia na serwer proxy, on odsyła to co ma w cache → zmniejsza obciążenia dla serwera www. Wada: duże opóźnienie.
CDN (Content Distribution Networks), np CloudFlare - serwer proxy obsługiwany przez osobną organizację.

6. **Jak skłonić klienta, żeby łączył się z serwerem proxy a nie bezpośrednio ze stroną WWW?**

Skonfigurować go. (?)

7. **Jakie informacje dołączane są przez serwer proxy do zapytania?**

„Zwykły” serwer proxy dodaje do naszego żądania HTTP dodatkowe pola w nagłówku, m.in.
X-Forwarded-For: (nasz adres IP)
Via: (adres IP proxy)

(Istnieją tzw. anonimowe serwery proxy, które nie dodają tych nagłówków

8. **Co to są anonimowe serwery proxy?**
   Serwer pośredniczący, który funkcjonuje jako przekaźnik pomiędzy użytkownikiem i serwisem internetowym oraz którego zadaniem jest ukrywanie adresu IP maszyny użytkownika, usuwanie niektórych elementów pozwalających na identyfikację użytkownika (ciasteczka, identyfikator używanej przeglądarki, itp.) i ewentualne szyfrowanie komunikacji, co ma na celu uczynienie użytkownika anonimowym. Anonimowy proxy nie dodaje takich nagłówków jak: X-Forwarded-For, Via.

9. **Do czego służy protokół SMTP a do czego POP3?**
   SMTP (Simple Mail Transfer Protocol) jest prostym tekstowym protokołem służącym do wysyłania wiadomości email. Jeśli wysyłamy wiadomość na adres email w domenie obsługiwanej przez dany serwer to zostaje ona zapisana na dysk. W przeciwnym przypadku serwer może przekazać ją dalej, stając się na chwilę klientem SMTP.
   POP3 (Post Office Protocol version 3) to protokół internetowy z warstwy aplikacji pozwalający na odbiór poczty elektronicznej ze zdalnego serwera do lokalnego komputera poprzez połączenie TCP/IP.

10. **Co to jest przekazywanie poczty (*relaying*)? Co to jest *smarthost*?**
    Jeśli wysyłamy wiadomość na adres email w domenie nie obsługiwanej przez dany serwer to może [nie musi] przekazać ją dalej. Czynność to relaying [denied]. Kiedyś był to domyślny sposób pracy serwerów pocztowych. Obecnie: aby serwer był skłonny do takiej operacji, email musi pochodzić z zaufanego źródła (np. klient SMTP musi się uprzednio autoryzować).
    Podczas przekazywania wiadomości innemu serwerowi, przekazujący serwer SMTP staje się w danej transakcji zwykłym klientem SMTP. Przekazać może do docelowego serwera SMTP (rekord MX w DNS) lub do serwera SMTP (smarthost), który skłonny będzie przekazać ją dalej.

11. **Jaki rekord DNS jest sprawdzany przed wysłaniem poczty do danej domeny?**
    Rekord MX

12. **Wymień parę popularnych pól w nagłówku maila. Do czego służą pola Received i Bcc?**

- From
- To
- Cc (kopia)
- Bcc („ślepa” kopia)
- Date (data)
- Subject (temat)
- Received - tracking information generated by mail servers that have previously handled a message
- Bcc - umożliwia wysłanie wiadomości poczty elektronicznej do wielu na raz odbiorców w taki sposób, że odbiorcy nie widzą wzajemnie swoich adresów.

13. **Jakie pola w nagłówku są używane do tworzenia wątków z wiadomościami?**

References
In-Reply-To

14. Co umożliwia standard MIME?

Możliwość określenia Content-Type: (tak jak w HTTP) a w nim również kodowania.
Pole Content-Transfer-Encoding: (np. 8bit, base64).
Content-Type: multipart/*: możliwość wysyłania załączników (przykładowo wysyłania wiadomości jednocześnie w txt i html)

15. Co to jest spam? Jakie znasz metody walki ze spamem?

Są to niechciane wiadomości elektroniczne. Pierwszy raz wysłany prawdopodobnie 1 maja 1978 (reklama komputerów DEC).
Sposoby walki ze spamem:

- Blokowanie konkretnych tematów (wyrażenia regularne)
- Metody statystyczne (filtry bayesowskie)
- Blokowanie adresów (greylisting)
- SPF - Sender Policy Framework - definiuje jakie komputery są uprawnione do wysyłania poczty z polem From:

Ciekawostka historyczna: “spam” pochodzi ze skeczu Monthy Pythona. Tak, ta godzina jest już tak późna, że powinieneś zrobić przerwę (i obejrzeć wideło)

16. Na czym polega greylisting?

Początkowo wszyscy są na szarej liście, jeśli ktoś chce wysłać email na adres w danej domenie to proszony jest o ponowienie próby wysłania maila za jakiś czas, jeśli ponowi, to trafia na listę białą (zakładamy że to nie spamer).

17. Na czym polega mechanizm SPF?

SPF (Sender Policy Framework) – niekomercyjny projekt mający na celu wprowadzenie zabezpieczenia serwerów SMTP przed przyjmowaniem poczty z niedozwolonych źródeł. Ma to pozytywnie wpłynąć na ograniczenie ilości spamu oraz zmniejszenie ilości rozsyłających się wirusów.
Rekord SPF w DNS dla danej domeny:
ii.uni.wroc.pl. TXT "v=spf1 ip4:156.17.4.0/24 mx:ii.uni.wroc.pl mx:gmail.com mx:google.com -all"
definiuje jakie komputery są uprawnione do wysyłania poczty z polem From:
pochodzącym z domeny ii.uni.wroc.pl :
komputery z adresów 156.17.4.0/24
komputery obsługujące pocztę dla domen ii.uni.wroc.pl, gmail.com i google.com
rekord sprawdzany przez odbiorcę
Problemy przy przekazywaniu poczty (komputer przekazujący nie jest oryginalnym nadawcą wiadomości)
Serwer B zabezpieczony przez SPF sprawdza w DNS-ie, czy wysyłana do niego poczta pochodzi z serwera posiadającego „uprawnienia” do wysyłania poczty z danej domeny. Jeżeli tak, to poczta jest przyjmowana. Natomiast jeśli adres IP nie pasuje do danej domeny – połączenie jest odrzucane. Dzięki temu wiadomości wysyłane przez spamerów podszywających się pod cudze adresy e-mail lub przez wirusy typu Mydoom zostaną odrzucone.

Wykład 11: Kodowanie i szyfrowanie

1. **Jakie znasz typy kodów detekcyjnych? Do czego służą i jakie są między nimi różnice?**
   a. **Sumy kontrolne**
      Najprostszy wariant kodów detekcyjnych: dodajemy do siebie (16/32-bitowe) słowa w przesyłanej wiadomości. Warianty: przeniesienia, negowanie bitów, … Nie wykrywają zamian słów. Efektywnie obliczane przez CPU. Stosowane w warstwie sieciowej (IP) i transportowej (TCP/UDP).
      i. Wariant: bit parzystości:
         Do wiadomości dodajemy bit, który ustawiamy tak, żeby liczba ustawionych bitów w całości była parzysta. Wykrywa przekłamania nieparzystej liczby bitów.
   b. **Kody CRC (Cyclic Redundancy Check)**
      Sumy kontrolne CRC bazują na dzieleniu w pierścieniu wielomianów nad ciałem F_2 (zbiór {0,1} z działaniami modulo 2). W prostszych słowach podstawą CRC są działania wykonywane na wielomianach, których współczynniki są ze zbioru {0,1}, a działania na tych współczynnikach są wykonywane modulo 2.
      Efektywnie obliczane sprzętowo. Stosowane w warstwie łącza danych.
      Przykładowo ethernet definiuje konkretny wielomian stopnia 32. Stosowane wielomiany stopnia n wykrywają najczęściej:
      - pojedyncze błędy bitów,
      - nieparzystą liczbę pojedynczych błędów bitów,
      - dwa błędy bitów oddalonych o co najwyżej 2^n-1
      - przekłamania ciągu bitów nie dłuższego od n
   c. **Kody MAC (Message Authentication Code)**
      Kod uwierzytelniający. Cel: zapewnienie integralności wiadomości: trudno ją zmodyfikować tak, żeby uzyskać taki sam MAC.
      Kryptograficzne funkcje haszujące:
      - Funkcja h: funkcja haszująca, szybko obliczalna,
      - h: ciąg bitów dowolnej długości →ciąg bitów długości d.
      - Przykładowo dla MD5 d = 160, dla SHA-256 d = 256.
      - Dla dowolnego x znalezienie y, takiego że h(x) = h(y) jest obliczeniowo trudne.

      Funkcję h można użyć do wykrycia błędów w transmisji (MD5 podawane wraz z plikiem na stronie).

2. **Jakie rodzaje błędów mają wykrywać kody detekcyjne? Z czego biorą się błędy przy przesyłaniu danych?**
   a. up ↑
   b. Zakłócenia warstwy fizycznej, błędy w sterownikach/routerach, atak

3. **Jak działa algorytm obliczania sum kontrolnych CRC?**
   Traktujemy wiadomość jako wielomian w Z_2, dzielimy go z resztą przez jakiś określony wielomian. Następnie doklejamy tą resztę do końca wiadomości. Przy odebraniu

wiadomości wystarczy sprawdzić, czy wszystko co zostało odebrane dzieli się przez ten ustalony wielomian.

4. **Do czego służą kody MAC? Co to jest HMAC?**
   Patrz zadanie 1,
   HMAC: wyślij m, h(s ‖ h(s ‖ m)), s: sekret, m: wiadomość, h: funkcja szyfrująca

5. **Jakie własności powinna mieć kryptograficzna funkcja skrótu?**
   a. łatwo obliczalna
   b. dla ustalonego x trudno jest znaleźć y t.że h(x) = h(y).
   c. prawdopodobieństwo kolizji jest małe (tzn p-stwo że x != y, ale h(x) = h(y) dla dowolnych x, y

6. **Jakie znasz metody korygowania błędów w transmisji?**
   a. Kody detekcyjne + ARQ (wysyłanie do skutku)
   b. Kodowanie naiwne (3, 1) każdy bit powtarzamy 3 razy, pozwala na skorygowanie 1 przekłamanego bitu.
   c. Kodowanie hamminga (7, 4) na 7 bitach kodujemy 4, również pozwala na skorygowanie jednego bitu ale dużo lepsza efektywność

7. **Co to jest (a,b)-kod? Podaj przykład.**
   Na a bitach kodujemy b bitów informacji. Np. Hamming(7, 4)

8. **Co to jest odległość Hamminga? Jak wpływa na możliwość detekcji i korekcji błędów?**
   Odległość Hamminga to minimalna liczba zmian potrzebna by przekształcić jeden ciąg w drugi. Modyfikacje to dodanie jednego bitu, usunięcie jednego bitu oraz zmiana jednego bitu.
   Jeżeli odległość hamminga między dowolnymi dwoma kodami jest większa niż k to:
   - Można wykryć do k - 1 błędów pojedynczych bitów
   - Można naprawić do floor((k - 1) / 2) błędów - patrzymy jaki kod jest najbliżej

9. **Czym różni się poufność od integralności?**
   a. Poufność - nikt poza mną i osobą z którą się komunikuję nie może podejrzeć wiadomości
   b. Integralność - Jeśli nastąpiła zmiana treści wiadomości to się o tym dowiaduję

10. **Co to są szyfry monoalfabetyczne? Dlaczego łatwo je złamać?**
    Szyfry które zmieniają każdą literę na jakąś inną, np (x + 5) mod 26 przykład ROT13
    Łatwo je złamać, można brute-forcem. Jest tylko 26 możliwości. W przypadku szyfru Vigenere'a atak słownikowy.

11. **Na czym polegają ataki z wybranym tekstem jawnym, znanym tekstem jawnym i znanym szyfrogramem?**
    a. Wybrany tekst jawny - atakujący może zmusić do wysłania jakiejś konkretnej wiadomości
    b. Znany tekst jawny - atakujący zna wysyłany tekst jawny oraz szyfrogram.

c. Znany szyfrogram - atakujący widzi tylko szyfrogramy.

12. Czym szyfrowanie symetryczne różni się od asymetrycznego?

a. Symetryczne - tym samym kluczem szyfrujemy oraz rozszyfrowujemy wiadomości, jest to współdzielony sekret.
b. Asymetryczne - są dwa klucze A oraz a, każdym z nich można zaszyfrować lub odszyfrować, ale operację odwrotną trzeba wykonać drugim. Np szyfruję A odszyfrowuję a.

13. Co to jest szyfrowanie *one-time pad*?

Szyfrowanie całej wiadomości poprzez XOR z losowym ciągiem bitów tej samej długości. Szyfr perfekcyjny ale każdego ciągu można użyć tylko raz i odbiorca też musi go mieć

14. Na czym polega szyfrowanie blokowe? Czym różni się tryb ECB od CBC?

Szyfrowanie blokowe - całą wiadomość dzielimy na bloki jednakowej długości (np 80 bit) i w razie potrzeby wypełniamy na końcu. Następnie każdy blok szyfrujemy.
ECB - Każdy blok szyfrujemy osobno, jeśli jakieś dwa bloki były takie same, to kody też będą.
CBC - Losujemy jakiś wektor początkowy IV, po czym xorujemy z pierwszym blokiem, szyfrujemy, wynik xorujemy z drugim blokiem i szyfrujemy itd (str 38). W ten sposób dwa bloki o tej samej zawartości będą miały inny szyfrogram. Znając klucz odszyfrujemy wszystko poza pierwszym blokiem.

Wykład 12: Podstawy kryptografii

1. **Czym szyfrowanie symetryczne różni się od asymetrycznego?**
   a. symetryczne - jeden klucz do szyfrowania i odszyfrowywania
   b. asymetryczne - osobne klucze do dekryptażu i szyfrowania

2. **Na czym polega bezpieczeństwo przy szyfrowaniu asymetrycznym?**
   Tylko kluczem prywatnym można odszyfrować wiadomość zaszyfrowaną kluczem publicznym.

3. **Opisz algorytm RSA**

4. **Czy różni się szyfrowanie od uwierzytelniania?**
   a. szyfrowanie - zabezpiecza dane
   b. uwierzytelnianie - potwierdzenie tożsamości

5. **Co to jest atak powtórzeniowy?**

6. **Czy w szyfrowaniu asymetrycznym szyfrujemy kluczem publicznym czy prywatnym?**
   Szyfrujemy kluczem publicznym.

7. **Na czym polega podpisywanie wiadomości? Jakim kluczem to robimy?**
   Strona uwierzytelniająca wylicza skrót (ang. hash) podpisywanej wiadomości. Następnie szyfruje ten skrót swoim kluczem prywatnym i jako podpis cyfrowy dołącza do oryginalnej wiadomości. Dowolna osoba posiadająca klucz publiczny może sprawdzić autentyczność podpisu, poprzez odszyfrowanie skrótu za pomocą klucza publicznego nadawcy i porównanie go z własnoręcznie wyliczonym na podstawie wiadomości.

8. **Jak można wykorzystać podpisy cyfrowe do uwierzytelniania?**
9. **Czy HMAC można wykorzystać do uwierzytelniania? Czy HMAC jest podpisem cyfrowym?**
10. **Dlaczego lepiej podpisywać funkcję skrótu wiadomości niż samą wiadomość? Z jakim ryzykiem się to wiąże?**
11. **Co to są certyfikaty? Co to jest ścieżka certyfikacji?**
12. **Co to jest urząd certyfikacji (CA)?**
13. **Jak SSL/TLS zapewnia bezpieczeństwo połączenia?**
14. **W jaki sposób w SSL następuje uwierzytelnienie serwera, z którym się łączymy?**
15. **Czym różnią się certyfikaty zwykłe od rozszerzonych?**
    - ceną
    - rozszerzone zawierają dodatkowe informacje jak np. nazwa firmy

16. **Co to są klucze sesji? Po co się je stosuje?**
    M. in. służą do zapamiętywania uwierzytelnienia użytkownika.

17. **Co to są kolizje kryptograficznej funkcji skrótu?**
    x,y t.że x != y ale h(x) = h(y)

18. **Na czym polega atak urodzinowy?**

    Celem **ataku urodzinowego** jest znalezienie kolizji funkcji haszującej. Jest to atak siłowy. U jego podstaw leży jednak paradoks dnia urodzin, który pozwala oczekiwać, że kolizja zostanie znaleziona znacznie szybciej niż sugerowałby to rozmiar przeciwdziedziny funkcji haszującej. Liczba potrzebnych do tego sprawdzeń rośnie bowiem proporcjonalnie do pierwiastka z liczby wszystkich możliwych wyników funkcji haszującej.

    Przykład: Algorytm haszujący MD5 generuje 128-bitowe skróty. Daje nam to 2^{128} różnych skrótów. Aby jednak trafić na dwa identyczne skróty z 50% prawdopodobieństwem, wystarczy wygenerować ok. $1{,}1774 * 2^{64}$ skrótów.

    Szczegółowe wyprowadzenie znajduje się w artykule paradoks dnia urodzin.

19. **Na jaki atak narażone jest podejście, w którym wiadomość najpierw szyfrujemy a potem podpisujemy?**

    Ktoś może ją przechwycić, usunąć nasz podpis, po czym dodać inny.

20. **Na jaki atak narażone jest podejście, w którym wiadomość najpierw podpisujemy a potem szyfrujemy?**

    Osoba która otrzymuje naszą wiadomość może ją odszyfrować, po czym podpisać się i wysłać dalej.

Wykład 13: Bezpieczeństwo sieci

1. **Co to jest pamięć CAM i jak stosuje się ją w przełącznikach? Jak można ją przepełnić?**
   Jest to pamięć skojarzeniowa, używana między innymi w przełącznikach do przechowywania tablicy przełączania. Jest to rodzaj pamięci o krótkim czasie dostępu. Przełącznik ma sprzętową tablicę haszującą (CAM = content addressable memory) z wpisami „adres MAC - port".
   Zmieniając adres MAC można zalać CAM nowymi wpisami - przełącznik przejdzie w tryb uczenia się.

2. **Opisz atak typu *ARP spoofing*; jak można go wykorzystać do podsłuchiwania komunikacji między dwoma komputerami podłączonymi do przełącznika sieciowego?**
   ARP spoofing to atak sieciowy w sieci Ethernet pozwalający atakującemu przechwytywać dane przesyłane w obrębie segmentu sieci lokalnej. Przeprowadzony tą metodą atak polega na rozsyłaniu w sieci LAN odpowiednio spreparowanych pakietów ARP zawierających fałszywe adresy MAC. W efekcie pakiety danych wysyłane przez inne komputery w sieci zamiast do adresata trafiają do osoby atakującej pozwalając jej na podsłuchiwanie komunikacji.

3. **Co oznacza termin *IP spoofing*? Na czym polega metoda weryfikacji tak zmodyfikowanych pakietów (*ingress filtering*)?**
   IP spoofing - czyli fałszowanie adresu IP nadawcy.
   Ingress Filtering (weryfikacja) - sprawdzanie czy przychodzące do nas pakiety rzeczywiście mogą pochodzić z sieci z którą dany interfejs jest połączony. Przykładowo z interfejsu sieciowego podłączonego do sieci 192.168.0.0/24 nie powinien nadejść pakiet ze źródłowym IP 200.200.200.200.

4. **Na czym polega atak *RIP spoofing*?**
5. **Opisz, jak wygląda uwierzytelnianie serwera SSH.**
6. **Na czym polega uwierzytelnianie użytkownika przez SSH z wykorzystaniem kluczy RSA?**

7. **Przedstaw przykładowe ataki wykorzystujące brak sprawdzania poprawności wprowadzanych danych.**

- Wprowadzenie dodatkowego, nowego polecenia, które zmodyfikuje zapytanie do bazy danych. (SQL injection),
- Wykorzystanie programu do wypisania zawartości plików systemu, które nie są przeznaczone dla zwykłego usera (../), przykład: http://jakas.domena/skrypt?plik=../../etc/passwd?
- Podanie na wejściu zbyt dużej ilości danych, przez co można nadpisać pamięć, które nie jest dla nas przeznaczona przykładowo adres powrotu (przepełnienie bufora

8. **Wyjaśnij pojęcia: robak internetowy, exploit, botnet**
   - **robak** - złośliwe, samoreplikujące się oprogramowanie
   - **exploit** - program wykorzystujący jakąś lukę bezpieczeństwa
   - **botnet** - sieć zdalnie sterowanych, zainfekowanych komputerów

9. **Na czym polega *phishing*?**
   Wyłudzanie informacji poprzez podszywanie się pod godną zaufania osobę, lub instytucję (stronę internetową). Atak oparty na inżynierii społecznej.

10. Co to jest skanowanie portów? Po co się je wykonuje?

Skanowanie portów to wysyłanie zapytań różnych protokołów na różne porty i nasłuchiwanie odpowiedzi. Na jej podstawie możemy zidentyfikować działającą na danym porcie usługę. Używane np. przy testach penetracyjnych.

11. Co to są ataki DoS i DDoS?

- **Denial of Service** - celem ataku jest wyczerpanie zasobów celu. Możemy wyczerpać np. moc obliczeniową, czy limit jednoczesnych połączeń.
- **DDoS** - rozproszony DoS. Wiele komputerów wykonuje atak DoS. Np. botnet.

12. Na czym polega atak typu odbity (*reflected*) DoS?

Atak polega na spreparowaniu pakietów z adresem źródłowym ustawionym na cel ataku. Wysyłamy do komputera pośredniego mały pakiet, który generuje dużą odpowiedź na adres celu ataku.

13. Jak działa i do czego jest wykorzystywany ICMP Traceback?

Przed atakiem DDoS można się bronić tylko, jeśli atak pochodzi z jednego geograficznego obszaru. Problemem jest ustalenie źródła ataków (źródłowe adresy IP są podrobione. Po ustaleniu źródła, można zadzwonić do administratora.
ICMP Traceback: każdy router z małym prawdopodobieństwem (ok. 1/20000) dla przesyłanego pakietu wysyła również do odbiorcy dodatkowo komunikat ICMP, który zawiera informację o przesyłanym właśnie przez router pakiecie, informacje o routerze, itp.

14. Podaj przykłady tunelowania.

a. HTTP w SSL = HTTPS
b. VPN
c. TCP w SSH

15. Rozwiń skrót VPN. Do czego służy?

- **Virtual Private Network**
  - do tunelowania.
  - pozwala zachować prywatność (w obrębie tunelu)
  - pozwala połączyć dwie niepołączone sieci w jedną logiczną sieć

16. Porównaj wady i zalety filtrów pakietów: prostych, stanowych i działających w warstwie aplikacji.

- **proste**
  - analizują tylko nagłówki IP
  - szybkie, bardzo nieprecyzyjne
- **stanowe**
  - analizują nagłówki IP i TCP
  - śledzą trójstanowe uzgodnienie, pamiętają stan połączenia
  - “rozumieją” numery sekwencyjne - lepsza odporność na fałszowanie nagłówków
- **w warstwie aplikacji**
  - analizują zawartość segmentów i datagramów
  - np. w przypadku FTP “rozumieją” że trzeba otworzyć odpowiedni port na dane
  - to coś innego niż zapory aplikacji (które analizują wywołania systemowe aplikacji)

17. Do czego służą moduły (*chains*) INPUT OUTPUT i FORWARD w zaporze Linuksa?

Realizują firewalla.

- **INPUT** - pakiety przychodzące z zewnątrz i kończące trasę w naszym komputerze
- **FORWARD** - pakiety przechodzące przez nasz komputer
- **OUTPUT** - pakiety tworzone lokalnie i opuszczające nasz komputer

18. W jakich łańcuchach zapory Linuksa wykonywany jest źródłowy a w jakich docelowy NAT?

WARTO WIEDZIEĆ

1. **Paradygmaty przesyłania danych**
    a. przełączanie obwodów
        i. koncepcja jak w centralach telefonicznych
        ii. ścieżka dla strumienia danych ustalana raz na cały czas komunikacji
        iii. rezerwowane zasoby
        iv. gwarantowana stała szybkość
    b. przełączanie pakietów
        i. Multipleksowanie: rezerwujemy tylko fragment łącza dla jednego strumienia danych
        ii. gwarantowana stała szybkość
        iii. marnotrawstwo łącza jeśli nic nie wysyłamy
        iv. Idea routerów: każdy tylko przekazuje o jeden dalej.

Paczka z egzaminami:
https://drive.google.com/file/d/0Bx8IvpMic3PbNmUxNndHYjNWR2s/view