5. Počítačové sítě

Osnova

1. Základní pojmy, principy, architektury
2. Spojované a nespojované sítě
3. OSI model, protokoly v prostředí Internetu.
4. Směrování
5. Základní služby počítačových sítí, správa a bezpečnost sítí.

Výklad

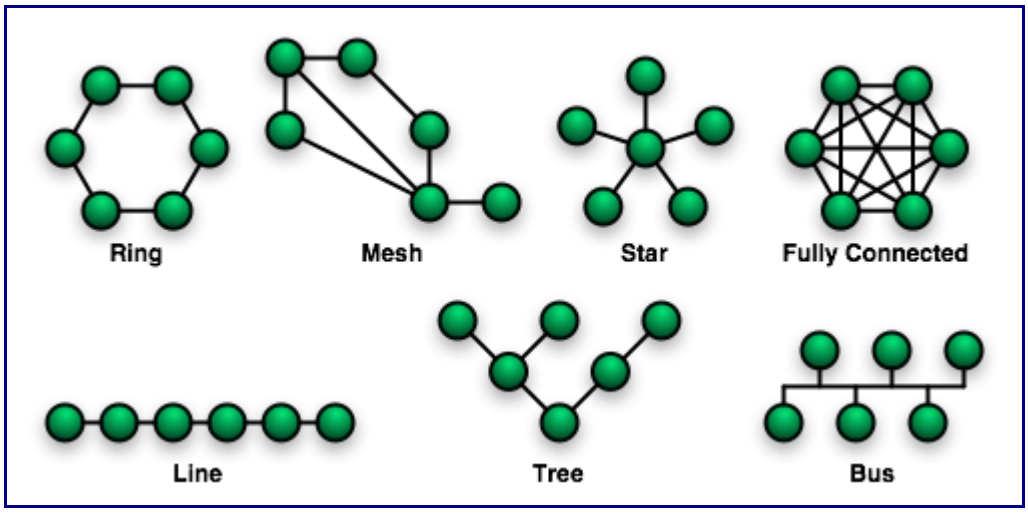
1. Základní pojmy, principy, architektury

- **Počítačová síť** - souhrnné označení HW a SW prostředků pro komunikaci a sdílení zdrojů (probíhá dle protokolů a na vrstvách) mezi počítači.
    - Využití – zjednodušení komunikace, sdílení hw, dat a softwaru.
    - Ideální vs. reálná síť, tj. řešíme problémy struktury (různé vlastnosti a použití), omezená kapacita, možná ztráta dat, zpoždění, jitter (data chodí s různým spožděním, kolísání velikosti spoždění), možné přijmutí paketů v jiném pořadí než odeslaném, poškození dat atd. Ideální síť je:
        - Transparentní struktura pro aplikace (end-to-end) vs. reálná struktura
        - Propustnost (neomezená vs. limitovaná)
        - Zpoždění (0 vs na aktivních prvcích a při přenosu)
        - Rozptyl (jitter) (0 vs vysoký)
        - Pořadí doručení
        - Poškození dat a výpadky
    - Usilujeme o maximální efektivitu, spravedlivost pro datové toky, decentralizaci, multiplexing/demultiplexing, spolehlivost, řízení toku (vyvarování zatížení sítě či účastník)

- **Architektura sítě** - struktura, z jakých částí se skládá a jak spolupracují. Většina sítí je organizována ve vrstvách (s tím pak souvisí ISO/OSI model) a počet vrstev a jejich vlastnosti se liší síť od sítě.
- **Protokol** (!!!) - jedná se o pravidla komunikace (pokud je komunikace mezi dvěma uzly možná, komunikují spolu stejné vrstvy). Protokol definuje *syntax* (struktura/formát dat a pořadí v jakém jsou prezentována - tvar a pořadí signálových prvků), *sémantiku* (význam – jak by měla být každá sekce bytů interpretována) a č*asování* (kdy by měla být data poslána a jak rychle).
- **Multiplexing** - je metoda pro sdílení přenosového média (rozdělení na části) mezi více signály, které jsou zkombinovány do jednoho a šetří tak vytížení drahého zdroje (pomocí **F**requency **D**ivission **M**ultimplexing - rozdělení frekvence a **W**avelength **D**ivission **M**ultiplexing (pro analogové signály) a **T**ime **D**ivission **M**ultiplexing – času (pro digitální signály)).
- **Paket** - data jsou rozdělena na malé části (současná obvyklá velikost **M**aximum **T**ranmission **U**nit je 1500 bajtů tj. maximální délka Ethernetového rámce), avšak vysokorychlostní sítě používají kvůli nižší režii větší délky). Pak můžeme využít sdílení pásma (TDM). Jedná se o nespojovaný přenos.
    - Nese data aplikace. Mají různou délku (nelze garantovat šířku linky)
    - Délka paketu ovlivňuje chování sítě (menší ⇒ férovost, delší ⇒ výkon)
    - Fixní déka ⇒ stabilní kvalita
    - Jiný přistup IPv4 a Ipv6

Topologie

- Sběrnicová topologie - spojení zprostředkovává jediné přenosové médium (sběrnice), ke kterému jsou připojeny všechny uzly sítě (koncové počítače). V počítačových sítích se obvykle používá tzv. systém náhodného přístupu (**C**arrier **S**ense **M**ultiple **A**ccess), který se kolizím snaží předcházet a v případě že nastanou - řeší je.
    - Snadná realizace a snadné rozšíření jíž stávající sítě.
    - Nesnadné odstraňování závad, pokud nastane nějaký problém s kabelem, celá síť přestane fungovat, mohou vznikat kolize.
- Hvězdicová topologie - každý počítač je připojený pomocí kabelu (UTP, STP) k centrálnímu prvku hubu nebo switchi. Mezi každými dvěma stanicemi existuje vždy jen jedna cesta. Jedná se o nejpoužívanější způsob.
    - Pokud selže jeden počítač nebo kabel nebude fungovat spojení pouze pro jednu stanici a ostatní stanice mohou vysílat i přijímat nadále.
      Snadno se nastavuje a rozšiřuje.
    - Potřeba extra hardware v porovnání se sběrnicovou topologií.
      V případě selhání centrálního síťového prvku přestane fungovat celá síť.
- Kruhová topologie - obvyklým způsobem řešení komunikace je implementace tokenu, který si stanice v kruhu postupně předávají a který jeho držiteli umožňuje vysílat, přičemž ostatní stanice pouze naslouchají. Zpráva tak prochází přes všechny mezilehlé počítače v kruhu, přičemž její zpoždění na každém uzlu je jen jeden bit.
    - Přenos dat je relativně jednoduchý.
      Přidání dalšího uzlu má jen malý dopad na šířku pásma, nevznikají kolize, minimální zpoždění (v bitech podle počtu uzlů).
    - Data musí projít přes všechny členy kruhu, což zvyšuje riziko poruchy, při vyřazeni jedné stanice další stanice přestávají pracovat.
      Při přidání nového uzlu je nutné dočasně kruh přerušit.
- Technologie Token ring je kruhovou topologií jen na logické úrovni, fyzicky se jedná o topologii hvězdicovou.



2. Spojované vs. Nespojované sítě (Connection-oriented vs. Connectionless)

Jakým způsobem dostat informaci z místa A do B: na začátku je ustanoveno spojení a jeho vlastnosti (cesta v síti, kvality) nebo se může každý paket cestovat „podle svého“ (problémy s kvalitou).

- **Spojované** (sítě s přepínáním okruhů)
    - Prostředky jsou **rezervovány** pro spojení (je zabrána celá šířka pásma). Koncové uzly použijí protokol pro ustanovení spojení před vlastním přenosem dat a je jej poté potřeba ukončit.
    - Jde o stavovou komunikaci a přechody musí být koordinované. Pro spojení je nalezena jedna cesta po které komunikace probíhá a mohou ji být vyhrazeny zdroje. Každé spojené má své ID.
    - Lze zaručit přenosovou rychlost (důležité u audia/videa) a zachování pořadí.
    - Pouze unicast (zasílání paketů pouze jednomu cíli. Opakem je broadcast). Používá se multiplexing.
    - např. telefonní spojení (ATM)
    - nikdo nemůže využít tu linku kterou jsem si zamluvil

- **Nespojované** (sítě s přepínáním paketů)
    - Menší overhead. Každý datový stream je rozdělěn na části (pakety).
    - Sdílení síťových prostředků.
    - Komunikující strany o sobě neví. Komunikuje se pomocí zasílání zpráv – datagramů a cesta je hledána pro každý datagram znovu, tj není žádná pevně vytyčena. Každý datagram musí nést plnou adresu příjemce. Je bezstavová. Neřeší se nestandartní situace a pokračuje se v přenosu. Nezachovává pořadí, přenos jednotlivých bloků je vzájemně nezávislý.
    - Umožňuje multicast a broadcast.
    - Nedokáže zaručit přenesovou rychlost. Může vznikat zahlcení (je-li příjem paketů vyšší než zpracovaní), ztráta a zpoždění (zpracováním na směrovači (CRC, zjištění odchozí linky, v IPv4 kráceni dle šířky linky), fronta, přenos).
    - Protože nemáme ustanoveno spojení, je třeba jednotlivým paketům říkat, kam mají „jít“.

- **Srovnejte kvality služeb u spojovaných a nespojovaných sítí.**
    - spojované – pevně daná rychlost dat (někdy pomalejší). Vytváříme spojení mezi dvěma body; nejdříve navázáno spojení, potom se přenášejí data, spojení mohou být permanentní (nemusí). Efektivita není dobrá, až nevhodná. Např.: dříve telefonní systém, každý hovor spojen.
    - nespojované – data rozdělena do paketů bez pevně určené cesty k cíly. Neudržuje se spojení mezi koncovými body. Proměnná rychlost přenosu. Data nejdou přes jedno spojení, můžou být fragmentována (rozdělena na menší). Neznáme stav sítě! Nevíme jak vypadá či jak vypadala. Implementace kvality služby nelze. Pracují ale efektivně.

3. OSI model

Abstraktní model standardizující oblast sítí a síťových protokolů. Byl vypracován organizací ISO. Jedná se tedy o jakousi **abstrakci reálného modelu sítě**. Nespecifikuje implementaci, ale uvádí všeobecné principy sedmivrstvé architektury.
Každá **vrstva má definovanou množinu vlastností a funkcí**, využívá služeb sousední nižší vrstvy a poskytuje služby sousední vyšší vrstvě.

- **Aplikační**
  Poskytuje **aplikacím přístup ke komunikačnímu systému** a umožnit tak jejich spolupráci.
  Protokoly:
  - Telnet, FTP, HTTP, DHCP, DNS, POP3, SMTP

- **Prezentační**
  Transformuje data do tvaru, který používá aplikace (např. převod do kódování a abeced), zabývá se pouze strukturou dat, ne významem. (např. MIME).

- **Relační**
  Správa relací mezi aplikacemi tj. **synchronizuje a organizuje komunikace**. Ustavení, udržení a ukončení relace.
  Protokoly:
  - SIP, SSL

  Služby:
  - Ustanovení a ukončení relace
  - Synchronizace, obnovení spojení a oznámení o výjimečných stavech.

- **Transportní**
  Zajišťuje **přenos dat s požadovanou kvalitou**, řeší spolehlivost a pořadí doručení (TCP, UDP).
  Protokoly:
  - TCP, UDP

  Služby:
  - Adresování – pomocí portů (16 bit, zdrojový a cílový tedy identifikace zdrojové a cílové aplikace)
    Tj. aplikace je jednoznačně identifikována IP adresou a portem.
  - Spolehlivost (TCP) – toku a chyb
  - Ochrana proti zahlcení příjemce a QoS
  - Spojované (seq, ack) vs Nespojované (data jsou odeslána do sítě bez číslování) služby

- **Síťová**
  „Vrstva routerů“, poskytuje **směrování, přepínání a síťové adresování**. Přenos paketů mezi různými počítači, které spolu přímo nesousedí (logicky spojuje LAN sítě). Obsahuje funkce, které odstraňují rozdílné topologické vlastnosti.
  Zařízeni:
  - Router

  Protokoly:
  - IP, ICMP, ARP

  Služby:
  - Packetizing – segmenty jsou transformovány na pakety
  - Fragmeting – existují různorodné MTU na cestě, když datagram je věší než MTU média, kterým paket musí projít (rozdělení na menši rámce tedy pakety). Liší v IPv4 a IPv6!
  - Adresování – IP adresy (unikátní v celé síti), pakety mají adresu příjemce a cíle.
    Adresy – unicast, broadcast (všem na LAN), multicast (jeden zdroj více cílům (odběratelům), jediná kopie dat, pak se musí přeložit na unicast (zmenšení zátěže odesilatele a sítě)

IPv4 – 32 bitů
IPv6 – 128 bitů
- Překlad adres - z IP na MAC (pomocí ARP)
- Směrování/Routing – proces výběru cesty (tj různými sítěmi) od zdroje k cíli
- Přepínání paketu – router přepne paket ze vstup na výstup (výstup dle směrování)
- Podpora kontrolních zpráv – pomocí protokolu ICMP – zprávy co se děje na síti(např. nedoručení paketu)

- **Linková (Spojová)**
  „Vrstva switchů“, poskytuje **spojení mezi sousedními uzly na lokální síti** (identifikátor je MAC adresa). Uspořádává data z fyzické vrstvy do logických celků – **rámce** a opačně rozděluje data z vyšších vrstev na rámce, které jsou sítí odesílány.
  Umí opravovat (kódováním) jednoduché chyby a hlásit neopravitelné. Je schopna l**ogicky oddělit dvě vedlejší sítě**. Zajišťuje přístup k médiu a soupeření o něj.
  Jedné se o LAN síť.
  Zařízeni:
  - Most (spojuje dvě části sítě na druhé (linkové) vrstvě), přepínač (most s více porty), opakovač

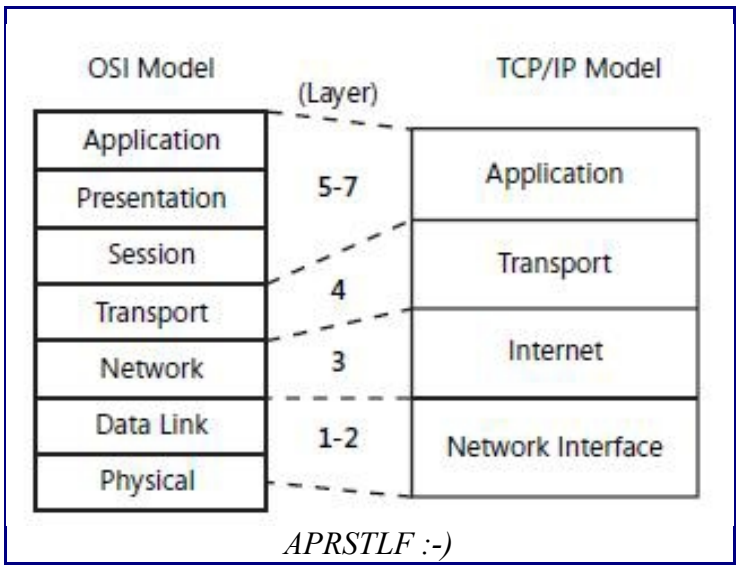
  Protokoly:
  - Ethernet, TDMA, CSMA

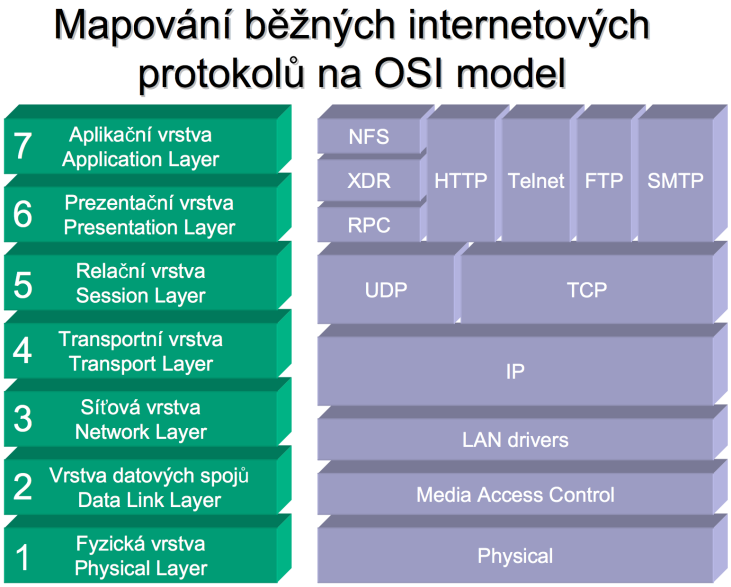
  Služby:
  - Framing – pakety jsou vloženy do rámců
  - Adresování – zařízeních na fyzické vrstvě (pomocí MAC adres, každý interface tj síťově rozhraní má od výrobce MAC adresu)
  - Detekce a oprava chyb (CRC, parita, Hammingův kód)
  - Kontrola toku – aby nedocházelo k zahlcení příjemce (stop-and-wait, sliding-window mechanism)
  - Přístup k médiu – náhodné (soupeření, CSMA/CD a CSMA/CA), kontrolované (rezervace, token) a dle kanálu (multiplexing, FDMA, TDMA)

- **Fyzická**
  Zajišťuje **fyzický přenos dat/bitů** – popisuje elektromagnetické či optické signály tedy způsob přenosu 1 a 0 po médiu (některá média pouze vhodná pro analogové – vzduch, některá pro analogové i digitální – kabely).
  Zařízeni:
  - Opakovače,modemy, síťové adaptéry

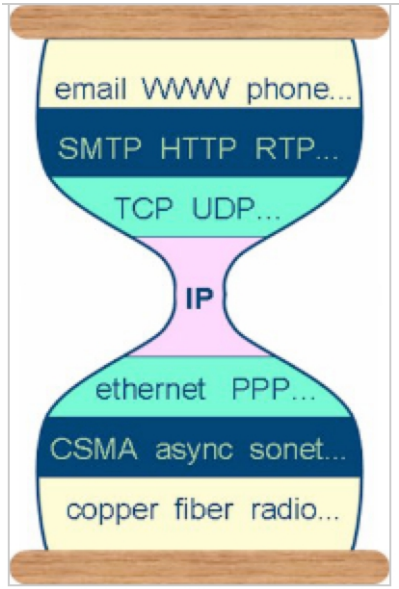
  Standardy:
  - IEEE 802.11, RS232

  Služby:
  - Bit-to-Signal transformace
  - Bit-Rate
  - Bitová synchronizace
  - Multiplexing
  - Přepíná okruhů (naopak přepínání paketů je záležitost od linkové vrstvy)





TCP/IP Hourglass Model – v Internetu se všechno (směrování, adresování … - router je nejvíce zatížen)) staví na IP a ICMP protokolu.



3. Protokoly v prostředí Internetu

Mezi nejpoužívanější protokoly patří:

ARP (**A**ddress **R**esolution **P**rotocol), **RARP** (**R**everse **ARP**)

- Překlad IP adres (L3) na fyzické adresy (L2)

IGMP (**I**nternet **G**roup **M**anagement **P**rotocol**)**

- Slouží k ustavení multicastové skupiny (analogie unicastového ICMP)
- Využívá se pro dynamické přihlašování a odhlašování ze skupiny u multicastového routeru ve své lokální síti.
  IGMP protokol řeší i situaci, kdy jsou v síti připojeny dva a více multicastových routerů, protože pak by mohlo dojít v síti k šíření nadbytečných informací. Routery pracují ve dvou stavech. „dotazovač“, který zasílá dotazy na členství a „posluchač“, který pouze naslouchá a je neaktivní. Aby se stanice přihlásila do skupiny, musí zaslat přes protokol IGMP zprávu „Membership report“ s IP adresou třídy D. Tato zpráva dorazí k routeru lokální sítě a ten si ji zapíše do tabulky.

ICMP (**I**nternet **C**ontrol **M**essage **P**rotocol)

- Řídící protokol, umožňuje oznamování o chybách. Definuje typ zprávy a kód odpovídá operaci nebo chybě (chybové zprávy může vytvářet host nebo směrovač).
- Nepoužívá se síťovými aplikacemi přímo (výjimkou je ping: posílá echo request a očekáva echo reply), ale používají ho operační systémy počítačů v síti pro odesílání chybových zpráv, například pro oznámení, že požadovaná služba není dostupná nebo že potřebný počítač nebo router není dosažitelný.
- Nejpoužívanější ICMP datagramy:
  Echo Request … požadavek na odpověď, každý prvek v síti pracující na IP vrstvě by na tuto výzvu měl reagovat. Často to z různých důvodů není dodržováno.
  Echo Reply … odpověď na požadavek.
  Destination Unreachable … informace o nedostupnosti cíle, obsahuje další upřesňující informaci (Net Unreachable, Host Unreachable, Protocol Unreachable).

IP (**I**nternet **P**rotocol)

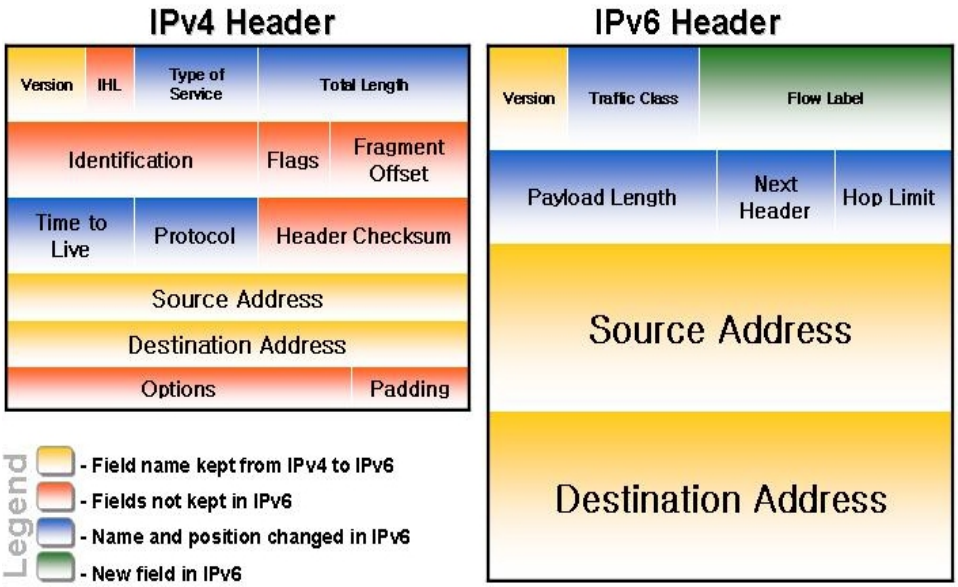
- IPv4
  - Adresy jsou 32 bitové. Možný počet adres: 4 bil. Kdysi se přidělovaly tzv. adresní třídy

	Leading bits	Size of network number bit field	Size of rest bit field	Number of networks	Addresses per network	Start address	End address
A	0	8	24	128	16 777 216	0.0.0.0	172.255.255.255
B	10	16	16	13 384	65 536	128.0.0.0	191.255.255.255
C	110	24	8	2 097 152	256	192.0.0.0	232.255.255.255

  - V roku 1993 však byl systém tříd nahrazen mechanizmem CIDR (Classless Inter-Domain Routing).
    Adresu můžeme zapsat ve tvaru např. 192.0.2.0/24, kde číslo za lomítkem znamená, jak dlouhý prefix adresy (v bitech) identifikuje síť. Zbytek bitů lze použít k adresování v rámci sítě
  - Unicast adresa – identifikace pouze jedinému interface, zaslání paketu pouze jedinému cíli (opak broadcast)

Broadcast adresa – všem v LAN síti (MAC adresa: FF:FF:FF:FF:FF:FF takový datový rámec je přijat všemi síťovými zařízeními ve stejné LAN jako odesilatel).
Multicast adresa – skupině příjemců, kteří požádali o data

- IPv6
  - Adresy jsou 128 bitové. Měl by být dostatek adres, takže by nemělo být potřeba NATu. Příklad IPv6 adresy: fe80:0000:0000:0000:0202:b3ff:fe1e:8329. Zápis jde zkracovat: fe80::202:b3ff:fe1e:8329 (buď sousední nuly na začátku nebo někde v ně adresy – pouze jednou)
  - Unicast adresa – jako v IPv4
    Broadcast adresa – chybí
    Mulicast adresa – jako v IPv4
    Anycast adresa – nově, jedná se o skupinu příjemců jako v multicastu, ale data jsou zaslána pouze jednomu členu ve skupině (nejbližšímu)

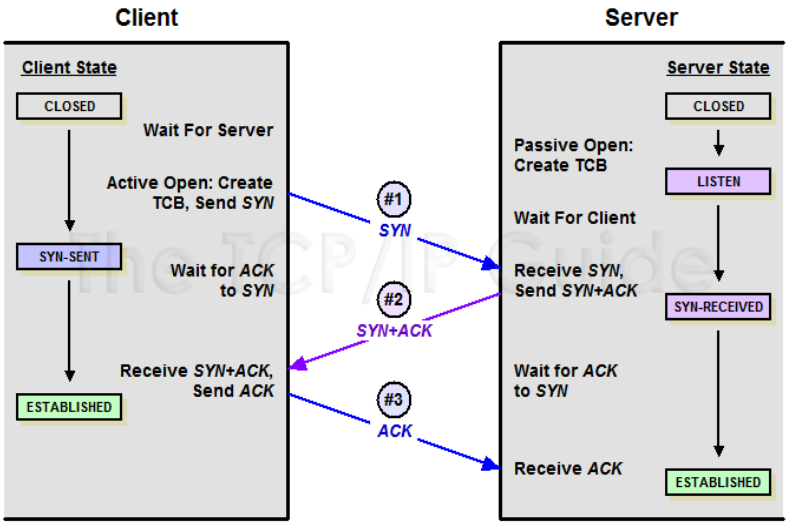


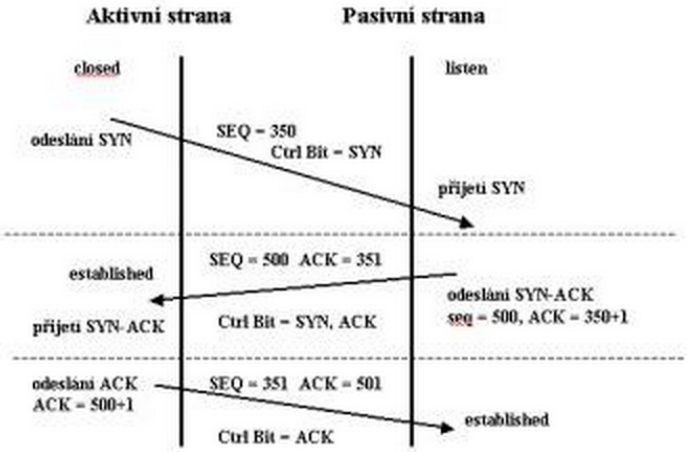
  - Větší adresní prostor + CIDR (IPv6 adresy jsou bez-síťové, jsou difinovaný pomocí CIDRu)
  - MTU discovery – zdroj musí definovat správnou velikost paketu (směrovače už nemohou krátit IPv6 datgram)
  - Jednodušší hlavička + možnost extension hlavičky
  - Podpora real-time komunikace (QoS - deferencované služby – Traffic class) a bezpečnosti (IPSec)
  - Podpora mobilních zařízeních (Correspondent Node, Home Agent)
  - Autokonfigurace zařízení – Neighbour Discovery Protocol (jak zjistit MAC adresu (v IPv4 byl ARP), založen na ICMP zprávách)

TCP (**T**ransmission **C**ontrol **P**rotocol)

- **Spojovaná a spolehlivá služba** transportní vrstvy (pokud to je možné, tak data budou přijata všechna a ve zpraveném pořadí).
- K rozlišení komunikujících aplikací používá TCP protokol čísla portů. Každá strana TCP spojení má přidruženo 16bitové bez-znaménkové číslo portu (existuje 65535 portů) přidělené aplikaci. Porty jsou rozčleněny do třech skupin: dobře známé, registrované a dynamické/privátní.
- Musí být ustanoveno/navázáno spojení (three-way handshake):

  http://www.svetsiti.cz/clanek.asp?cid=TCP-handshake-krok-za-krokem-3122000

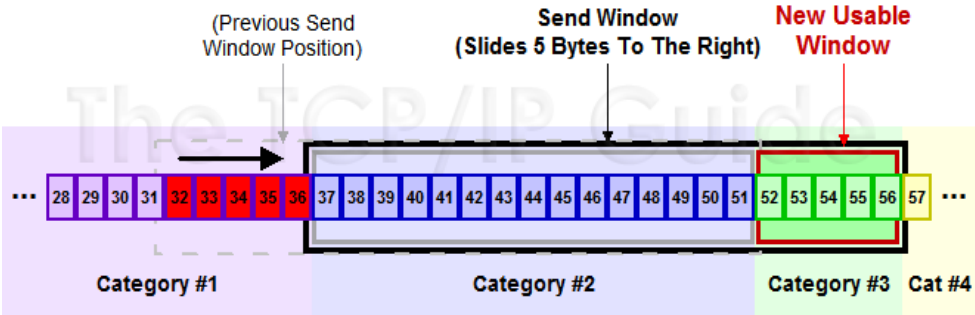




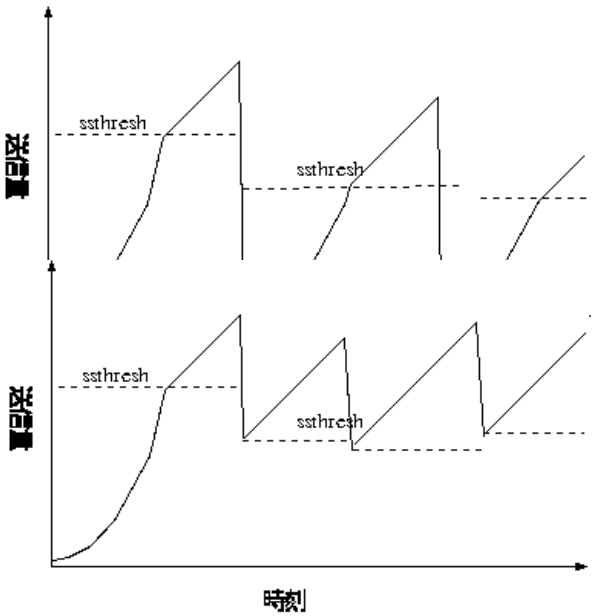
V průběhu navazování spojení se obě strany dohodnou na číslu sekvence a potvrzovacím čísle. Pro navázání spojení se odesílají pakety s nastavenými příznaky SYN a ACK.
Obě strany si pamatují číslo sekvence své i protistrany. Používají se totiž i pro další komunikaci a určují pořadí paketů. Když úspěšně proběhne trojcestný handshaking, je spojení navázáno a zůstane tak až do ukončení spojení. Ukončení spojení probíhá podobně jako jeho navázání. Používá se k tomu příznaků FIN a ACK.

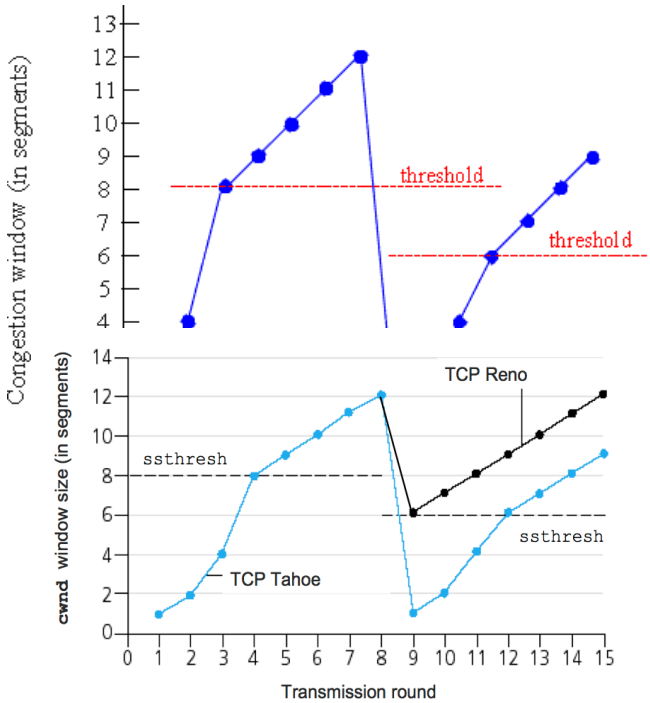
- Existuje několik různých implementací protokolu (Tahoe, Reno, Vegas). Všechno to jsou TCP protokoly, ale liší se způsobem řešení řízení toku a zahlcení.
- Zachovává pořadí paketů (čísluje), pokud nedorazí paket, příjemce vyšle potvrzení o nedoručení a všechny následující pakety zahazuje.
- **Řešení řízení toku** (rychlý zdroj a síť, pomalý cíl – **aby se nezahltil příjemce**)
  Standardní TCP algoritmus: Slidding window (stop-and-wait, Go-Back-N, Selective repeat), kdy velikost okna se nastavuje dle příjemce
  Bytes Sent And Acknowledged: Bytes 1 to 36.
  Bytes Sent But Not Yet Acknowledged: Bytes 37 to 51.
  Bytes Not Yet Sent For Which Recipient Is Ready: Bytes 52 to 56.
  Bytes Not Yet Sent For Which Recipient Is Not Ready: Bytes 57 to 95.
  - sliding window → když by odesílatel generoval data rychleji, než by je druhá strana byla schopna odebírat – došlo by pak nutně k přeplnění vyrovnávací paměti na straně příjmu. Součástí hlavičky TCP je proto šestnáctibitová položka velikost okna (window size). Přijímající strana do ní průběžně zapisuje, kolik bajtů datového proudu ještě může do vyrovnávací paměti přijmout. Odesílatel pak nemusí čekat na potvrzení každého segmentu,

ale může poslat – počínaje od nejvyššího potvrzeného sekvenčního čísla – ještě tolik bajtů, kolik udává velikost okna.



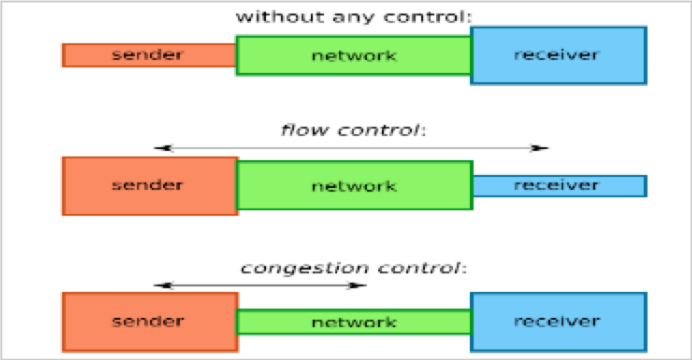
- Pro řízení zahlcení (**řízení zahlcení sítě**, využívá se cwnd (congestion window) tedy limituje počet not ack paketů, které mohou být poslány). Implementace využívá čtyři následující algoritmy:
  - **Algoritmus: Pomalý start**
    1. Congestion window (cwnd), vysílající odhaduje objem dat (průtok), který smí být vyslán, který se pak zvětšuje až do první chyby
    2. Receiver advertised window (rwnd), příjemce definuje objem dat, který akceptuje
    3. Po každém potvrzení se zvyšuje cwnd (congestion window) o svoji předchozí délku (cwnd = cwnd + 1), dochází k exponenciálnímu navyšování objemu posílaných dat až do ssthresh (slow start threshhold), pak pokračuje lineárně algoritmem Zábrana zahlcení.
  - **Algoritmus: Zábrana zahlcení**
    1. Nárůst pouze lineární, cwnd se zvyšuje o segment pouze aditivně za RTT (Round Trip Time – čas, jenž data potřebují na cestu mezi vysílajícím a přijímajícím a zpět) až po první chybu.
    Algoritmus: Rychlá retransmise
    1. Reakce na duplikované potvrzení (příjem 3 duplikovaných potvrzení detekuje ztrátu segmentu)
    2. Následuje jeho opětovné zaslání
  - **Algoritmus: Rychlé zotavení**
    Tahoe – nemá, znovy jde od začátku (od 1 MSS) pomalým startem, ale snížila se sstresh (na polovinu cwnd)
    Implementuje RENO:
    1. Jedná se o způsob, jak předejít návratu do fáze pomalý start po ztrátě paketů.
    2. Nepadá na 1 ale na nový sstresh.



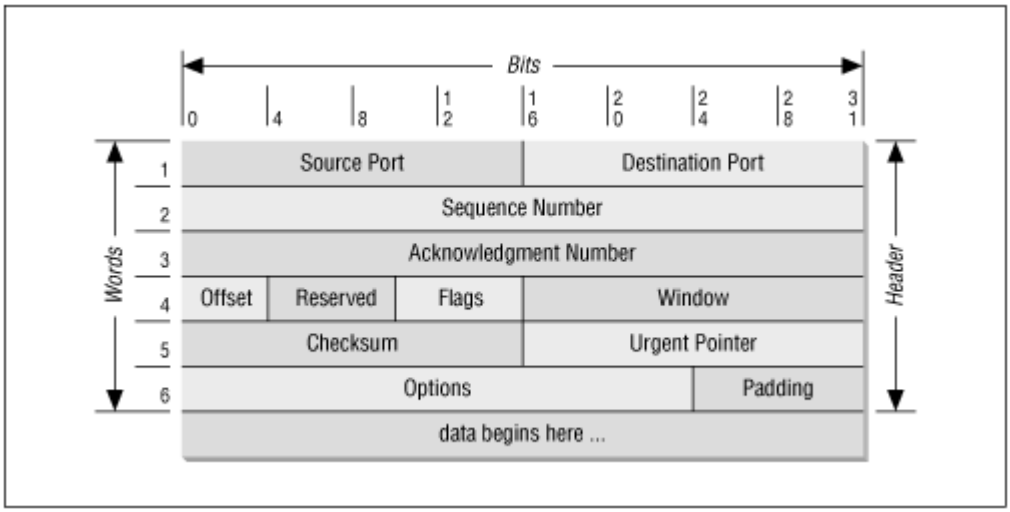


Flow control vs. Congestion control

- Ochrana príjemcu pred zahltením = flow control
- Ochrana siete pred zahltením = congestion control

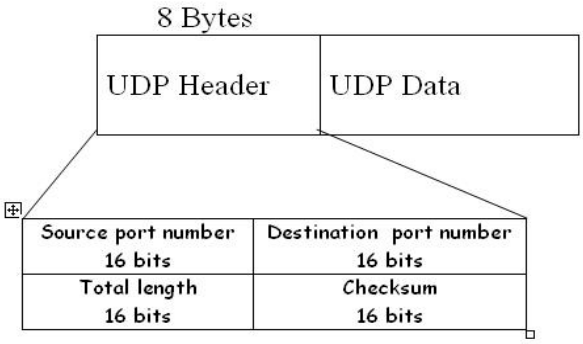


- Využívá jej přes 90 % dat. Blokově (bloky dat) orientován. FTP (port 21 a 20), SMTP (port 25), DNS (port 53) a HTTP (port 80).
- Pro mnoho aplikací není TCP vhodné. Velkým problémem je (alespoň u normálních implementací), že aplikace po ztrátě jednoho paketu nemůže dostat následující pakety do té doby, dokud není ztracený paket znovu poslán a úspěšně přijat. To způsobuje problémy realtimeovým aplikacím jako streamovaná média (např. internetové rádio), realtimeové multiplayerové hry a VoIP, kde je často užitečnější dostávat data včas, než je dostávat ve správném pořadí a kompletní.



UDP (**U**ser **D**atagram **P**rotocol)

- **Nespojovaný a nespolehlivá služba** (odpovědnost za pakety nese aplikace) transportní vrstvy. Poskytuje best-efford service. Nezachovává pořadí – Při odeslání dvou zpráv jednomu příjemci nelze předvídat, v jakém pořadí budou doručeny.
- Jednoduchost, minimální overhead, žádná režie (žádné ustanovené spojení, bezstavová komunikace).
  Určená pro prostý přenos paketů. De facto je to rozšíření IP protokolu o informaci o portech.
- Využívá jej přes pod 10 % dat (DNS, DHCP, real-time data, multicast). Byt stream orientován.


8 Bytes
UDP Header
UDP Data
Source port number
16 bits
Destination port number
16 bits
Total length
16 bits
Checksum
16 bits

4. Směrování

- Směrování je proces nalezení optimální cesty v síti od zdrojové stanice k cílové stanici. Tento proces využívá k nalezení optimální cesty informace obsažené v cílové síťové adrese koncové stanice. Směrovací proces patří do **třetí vrstvy** OSI modelu.
- Problém nalezení cesty mezi dvěma uzly, která splňuje zadané omezující podmínky (pakety prochází řadou směrovačů).
- Síť je reprezentována jako graf. **Uzly** = prvky sítě, **Hrany** = komunikační linky. **Ohodnocení hran** = cena komunikace.

Hierarchie směrování v Internetu: V lokálních sítích a podsítích, V autonomních systémech (AS, jsou číslované identifikátorem přidělovaným ICANN), Mezi autonomními systémy, Páteřní směrování.

- Směrování v lokální sítí
  - Na základě cílové IP adresy, IP adresy odesílatele a masky sítě odesílatele se prohledá routovací tabulka a najde se nejdelší prefix a packet se pošle na adresu příslušné brány.
- Směrování uvnitř AS
  - Autonomní systém (AS) je množina IP sítí a routerů pod společnou technickou správou, která reprezentuje vůči Internetu společnou routovací politiku (např. síť jednoho poskytovatele internetuú
  - Na základě nějakého vnitřního/**interního** routovacího protokolu (**I**nterior **G**ateway **P**rotocols) znají vždy všechny routery v autonomním systému nejkratší cesty do všech míst AS.
  - Internet by teoreticky mohl tvořit jediný autonomní systém, ale routovací tabulky by pak byly příliš veliké a technicky těžko zvládnutelné. Proto je rozdělen na více autonomních systémů, mezi kterými se routuje pomocí nějakého vnějšího/**externího** routovacího protokolu (**E**xterior **G**ateway **P**rotocols).
  - IGP → Distance Vector: **R**outing **I**nformation **P**rotocol, Link State: **O**pen **S**hortest **P**ath **F**irst
- Směrování mezi AS
  - EGP → Path vector: **B**order **G**ateway **P**rotocol

- Směrovací schémata mohou být:
  - Distribuovaná vs. centralizovaná
  - „Krok za krokem“ vs. Zdrojová
  - Deterministická vs. Stochastická
  - Jednocestná vs. Vícecestná
  - Dynamická vs. statický výběr cest
  - (Internet)

- Směrovací schémata dle sémantiky doručení (komu)
  - Unicast – specifický uzel (dominantní způsob doručení, jinými se zde ani nezabýváme)
  - Broadcast – Všechny uzly v síti
  - Multicast – Skupina uzlů, která projevila zájem o vysílání,
  - Anycast – nějaký uzel (nejbližší),
  - Geocast – uzly v geografické oblasti.

- Statické směrování
  - Známe topologii ⇒ lze dobře optimalizovat
  - Existuje centrální směrovací tabulka, bývá zpracovávána „ručně“
  - Jednoduché, ale citlivější na výpadky v síti a zátěž

- Dynamické směrování
  - Adaptabilní na výpadky, flexibilní a robustní

- Realizováno složitými algoritmy, které aktualizují tabulky
- Dynamické periodické výměny tabulek (občas dochází k nekonzistenci)
- Dělí se na:
  - Centralizované – stav se posílá do centra, které distribuuje tabulky, není robustní
  - Distribuované – uzly spolu vzájemně spolupracují (BGP, OSPF, RIP alg.), informace se postupně předává mezi uzly
  - Izolované – řešeno metodou náhodné procházky, vyžaduje zpětnou vazbu, šíření tabulek broadcastem ⇒ vysoká zátěž sítě (záplavový alg.)

Směrovací algoritmy (dynamické směrování)
Vybírají cestu na základě daných kritérií (propustnost, zpoždění, minimalizace počtu skoků (hops) přes aktivní prvky, minimalizace „ceny“ cesty).
Vlastnosti: Správnost, Jednoduchost, Robustnost, Stabilita, Spravedlivost, Efektivita, Optimálnost

- **Distance Vector** (rodina protokolů)
  - „Všechny informace o sítí jen svým sousedům“ [☝]
  - Směrování podle délky vektoru
  - Každý směrovač se snaží získat cenu (počet hopů) ke každému cíli. Na začátku zná jen sousedy, poté tabulku doplňuje. Na základě přijatých změn aktualizuje router svou směrovací tabulku.
  - DV je dvojice <Cíl, Cena>
  - Směrovač periodicky informuje sousedy o změnách v topologii (rozesílá DV toho co zná ~ jeho představa o síti)
  - Ak sa zmení stav linky, zmena je oznámená len najbližšiemu susedovi, vzdialenejším susedom je to oznámené len v prípade, ak tento zmenený stav zmení súčasnú najkratšiu cestu v strome
  - Výměna kompletní kopie směrovacích tabulek [☝]
  - Problém zacyklení (při výpadku hrany) řešen dělením horizontu, směrovač nikdy nesděluje cestu zpět k uzlu, od nějž se ji dozvěděl
  - Primárním algoritmem je Bellman-Fordův alg.
  - **RIP**
    - Využívá jedinou metriku, kterou je počet přeskoků.
    - Pokud je více cest do cíle, RIP zvolí tu s nejmenším počtem přeskoků a to i v případě, že je nejpomalejší.
    - Nemůže směrovat paket do větší vzdálenosti, než 15 přeskoků => použití pro malé sítě

- **Link State** (rodina protokolů)
  - „Informace o svých sousedech všem“ [☝]
  - Směrovače **znají na začátku pouze cestu a cenu k sousedům**, ostatní uzly nekonečná cena, **snažíme se aby všichni věděli vše** {stejné}
  - Směrovače informují všechny uzly v síti o změnách v topologii
  - Periodická výměna linek se kterými jsou spojeny **[☝]**
  - Vhodné pro všechny sítě
  - Každý uzol v sieti musí poznať náklady na každú cestu v sieti, akonáhle sa stav spojenia zmení, musia sa zmeniť všetky ostatné uzly
  - Idea: šíří se topologie, cesty si směrovače počítají samy
  - Pracuje ve dvou krocích:
    - Šíření topologie (broadcast)
    - Výpočet nejkratší cesty (Dijkstra)
  - **OSPF**
    - Na hranicích OSPF area jsou zpravidla routery náležející jak do vnitřní sítě, kde používají OSPF (nebo I-BGP) tak ven, kde používají zpravidla BGP-4.
    - Používá Dijkstrův algoritmus pro nalezení cesty
    - Momentálne najpoužívanejší LS protokol (Informácie o mojich susedoch zasielaná všetkým uzlom v sieti). Zhromažďuje informácie z dostupných routrov a konštruuje topologickú mapu siete.

        - Metrikou je cost (cena) - Je to nejaké číslo, ktoré nemusí odpovedať poctu hopov

- [**Path Vector** (rodina protokolů)]$_{\text{vsuvka}}$
    - **BGP-4**
        - Základními operacemi BGP routeru jsou: Vysílání oznámení, přijímání a filtrování oznámení, výběr cesty.
        - Základem protokolu je zasílání oznámení („advertisements“), která obsahují adresu cílové sítě, atributy cesty a identifikaci next-hop routeru.
          Předpokládá se, že BGP peerové nelžou o tom co posílají – případně lze filtrovat, pokud by posílali nesmysly. Filtrují se také vlastní cesty pro případ vzniku cyklů.
        - Zasílání oznamů definuje administrátor systému a umožňuje mu řídit kam potečou data.
        - Peerování se vesměs realizuje nepsanými dohodami či smlouvami. Příkladem je NIX.CZ.
        - BGP peering se realizuje ustaveným TCP spojením na portu 179, může být i autentizované.
        - Routovací tabulka vnějšího BGP směrovače v současné době roste k 200 tis. záznamů.

Statické vs. dynamické směrování - jaký je rozdíl a co používá v Internetu. Proč?

- Statické směrování – směrovací tabulky nastaveny ,je dána konfigurací počítače a změny je nutno provádět ručně.
- Dynamické směrování – průběžně reaguje na změny v síti a pomocí směrovacích algoritmů upravuje směrovací tabulky. V Internetu se díky neustálým změnám a obrovské robustnosti celé sítě používá distribuovaných dynamických směrovacích algoritmů, tedy dynamické směrování.

5. Základní služby počítačových sítí

Jakým způsobem dostat informaci z místa A do B: na začátku je ustanoveno spojení a jeho vlastnosti (cesta v síti, kvality) nebo se může každý paket cestovat „podle svého“ (problémy s kvalitou).

- Informace
  Pull model - uživatel si data explicitně vyžádá
  Push model - data jsou rozeslána na základě znalosti uživatele
- Komunikace (textové, hlasové, video, sdílení plochy a dat, …)
- Distribuované výpočty a gridy. (buzzword „Cloud“, sdílení výkonu a prostoru)
- Zábava

Protokoly:

- DNS – překlad IP adres na human-readable
- DHCP – automatická konfigurace sítě
- (s)FTP, SCP – sdílení souborů v síti
- SSH – Vzdálený přístup
- Kerberos – autentizace
- NTP (Network Time Protocol) – synchronizace času
- SMTP, POP3, IMAP – email

6. Správa sítě

Monitoring jednotlivých prvků a analýza výsledků.

- Dle ISO se správa sítě dělí na:
  - Správa Výkonu (SNMP)
  - Správa Chyb (SNMP traps)
  - Správa Konfigurací (často proprietární)
  - Správa Účtování (detekce uživatelů)
  - Správa Bezpečnosti (autorizace, ochrana před zneužitím)

- Přístupy ke správě
  - Reaktivní řízení – reakce na problémy
  - Proaktivní – detekce možnosti vzniku problémů a předcházení jim

SNMP (Simple Network Management Protocol)

  - Je součástí balíku protokolů „Internet Protocol Suite“
  - Je určen pro správu zařízení na IP sítích, typicky routery, switche, servery, pracovní stanice, tiskárny, atd.
  - Vystavuje konfigurační proměnné monitorovaného zařízení funguje na principu request – response:
    Řídící subjekt vydá požadavek
    Ten je přenesen agentu spravované entity
    Následně je zpět přenesena odpověď.

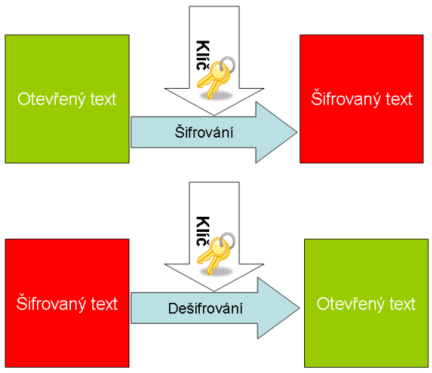
7. Bezpečnost

Klasickou kryptografickou úlohou je bezpečná komunikace po nezabezpečených kanálech. Podrobnější pohled požaduje od kryptografických funkcí zabezpečení:

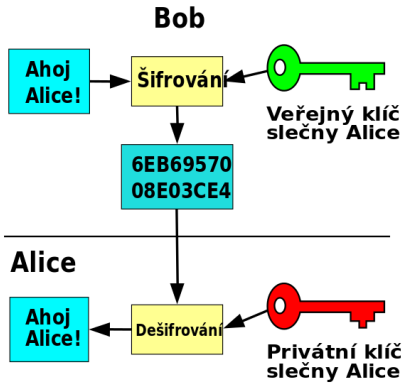
- **Utajení (Confidentality)** – kdy přenášeným informacím rozumí jen příjímající a odesílající subjekt.
- **Integrita (Integrity)** – kdy přenášená informace nemůže být nepozorovaně změněna.
- **Nepopiratelnost (Non-repudability)** – kdy odesílající subjekt nemůže popřít odeslání informace.
- **Autentizace** – zjištění a prokázání identity buď příjemce nebo odesílatele informace.
- **Anonymita** – nemožnost spojit si činnost se subjektem, který ji provádí.
- **Nesledovatelnost** – nemožnost spojit si dvě různé akce jednoho subjektu.

Obecně jsou kryptografické funkce matematické funkce, které lze rozdělit na skupiny:

- **Symetrická kryptografie** (konvenční)
  - Používá se stejný klíč pro šifrování i dešifrování.
  - Problém: distribuce klíčů (odesilatel a příjemce tajné zprávy se musí předem domluvit – bezpečně – na tajném klíči)
  - Výhody: vysoká rychlost, nízká výpočetní náročnost, malá délka klíčů
  - Typickými zástupci jsou AES, IDEA, 3DES. Dříve zejména DES.



- **Asymetrická kryptografie** (kryptografie s veřejným klíčem)
  - Pro šifrování a dešifrování se používají odlišné klíče, 1. klíč pro šifrování (příjemce ho nemusí znát) a 2. klíč pro dešifrování (odesilatel ho zpravidla nezná)
  - Nejběžnější verzí asymetrické kryptografie je využívání tzv. veřejného a soukromého klíče: šifrovací klíč je veřejný, majitel klíče ho volně uveřejní, a kdokoli jím může šifrovat jemu určené zprávy; dešifrovací klíč je privátní (tj. soukromý), majitel jej drží v tajnosti a pomocí něj může tyto zprávy dešifrovat (existují i další metody asymetrické kryptografie, ve kterých je třeba i šifrovací klíč udržovat v tajnosti)



  - Šifrovací klíč a dešifrovací klíč spolu musí být matematicky svázány, avšak nezbytnou podmínkou pro užitečnost šifry je praktická nemožnost ze znalosti šifrovacího klíče spočítat dešifrovací.
  - Problémy:
    Jak se přesvědčit, že veřejný klíč skutečně patří danému subjektu?
    Pomalost asymetrické kryptografie (netriviální matematiké výpočty, 100.000 x pomalejší)
    Délka klíčů
  - Výhody:
    Množství klíčů.
  - Typickým zástupcem je šifra RSA, dále podpisové schéma DSA, ECDSA nebo El-Gamal.

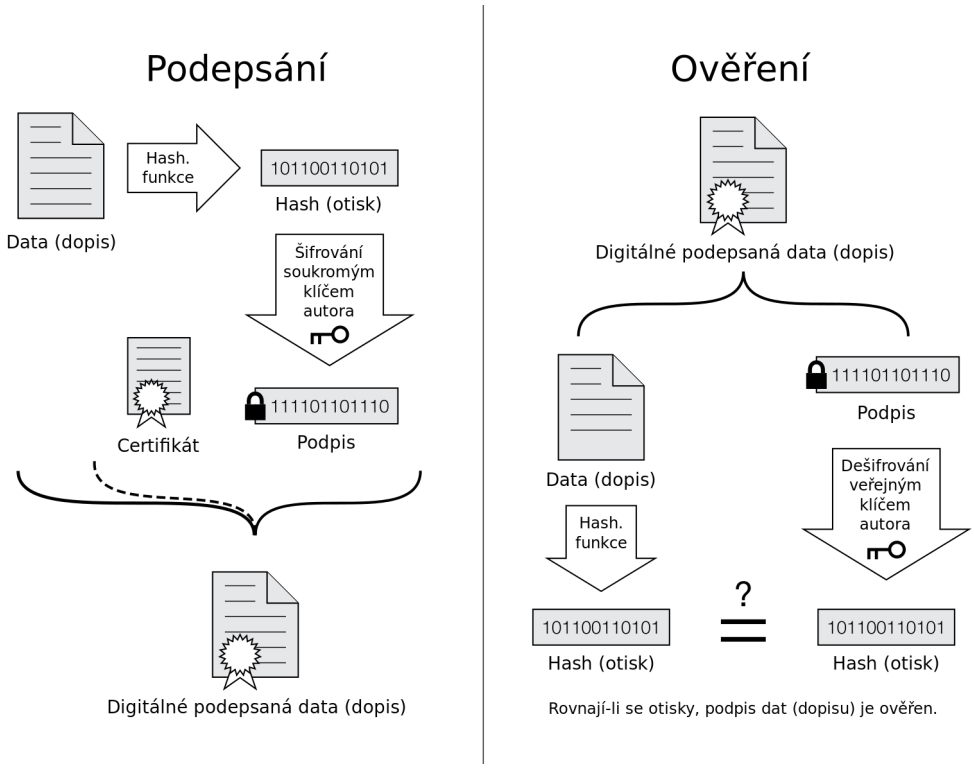
- **Hybridní šifrování**
  - Hybridní systém spojuje výhody obou řešení tím, že nejprve náhodně vygeneruje klíč pro symetrickou šifru a zašifruje jím zprávu, poté klíč samotný zašifruje asymetricky a spolu se

šifrovanou zprávou ho odešle příjemci. Ten si pomocí asymetrické šifry klíč dešifruje a pak pomocí klíče k symetrické šifře dešifruje i zprávu samotnou.
    - Pomocí pomalé asymetrické šifry se tak šifruje pouze krátký klíč, zatímco samotná zpráva, která může být velmi dlouhá, je šifrována rychlou šifrou symetrickou.

- **Hašovací funkce**
    - = funkce, které pro určitý vstup generují vždy stejný otisk.
      V ideálním případě mají vlastnosti: Jednosměrnost (z otisku nelze dopočítat vstup), Bezkoliznost (nelze najít dvě zprávy se stejným otiskem).
    - Funkce se používají v elektronickém podpisu jednak pro zajištění integrity a také pro obcházení problému pomalosti asymetrické kryptografie (podepisuje se jen krátký otisk zprávy).
    - Mezi nejpoužívanější funkce v současné době patří SHA-1 (objeven algoritmus pro získání kolizí, nicméně je ho výpčoetní náročnost je mimo současnou techniku) u které se očekává nahrazení funkcemi SHA-2. Z dřívějších jde zejména o MD5 (objeven postup pro získání kolizí).
    - Použítí např. v DB když ukládáme heslo → neuložíme původní heslo, ale použijeme SHA-2 fci

Digitální podpis
Nebo také elektronický podpis dle direktivy evropské komise o elektronickém podpisu je informace připojená k nějaké zprávě, která zajišťuje právě zmíněné funkce:
    - Integrita
    - Nepopiratelnost
    - Autentizace (a také identifikace, pokud není použit např. pseudonym)

V současné době realizován převážně pomocí prostředků asymetrické kryptografie (s veřejným klíčem) ve spojení s hašovacími funkcemi.



Nejprve je vypočten otisk (tzv. hash) dokumentu. Výsledný hash je poté zašifrován autorovým privátním klíčem (tj. utajeným klíčem), čímž vznikne elektronický podpis. Při ověření podpisu je nejprve znovu vypočten hash zprávy. Poté je pomocí veřejného klíče autora podpisu dešifrován obsah elektronického podpisu a výsledek je porovnán s vypočteným hashem zprávy. Pokud jsou obě hodnoty hashe stejné, je podpis z matematického hlediska platný.

Digitální certifikát vydaný certifikační autoritou je elektronicky podepsaný veřejný klíč, ke kterému jsou připojeny identifikační údaje jeho majitele.

Protože asymetrické šifrování je výpočetně náročné a proto by šifrování delších zpráv trvalo neúnosně dlouho → v praxi se nešifruje celá zpráva, ale pouze její tzv. otisk. Otisk můžeme chápat jako jakési „zhuštění“ původní zprávy do malého objemu pomocí matematické funkce, která je zvolena tak, že při jakékoliv změně v původní zprávě dostaneme zcela odlišný otisk.

Autentizace – potvrzujeme, že data pocházejí od určitého subjektu (autentikace, autentifikace jsou patvary)

Autentizační protokoly

- **Kerberos**
  - Autentizační protokol (má v současnosti 5 implementací) založený na principu autentizace dvou stran (A, B) přes důvěryhodného prostředníka (autentizační server T). A i B sdílí tajemství s T, mezi sebou nikoliv.
  - Po úvodní autentizaci klienta/terminálu vůči autentizačnímu serveru je klientovi přidělen tzv „ticket“ pro další autentizaci vůči (unix) serverům poskytujícím nějaké služby
  - KDC (key distribution center): server sdílí klíč s každým klientem; (klienti však mezi sebou klíče nesdílí); distribuuje klíče, které generuje.
  - KTC (key translation center): generování klíčů může zajišťovat tento dedikovaný server
- **Protokol SSL/TLS**
  - Autentizace pomocí certifikátů a protokolu typu výzva–odpověď
  - Po úvodní autentizaci je ustanoven symetrický klíč, kterým je šifrována všechna následující komunikace
  - Mezi aplikační vrstvou a protokolem TCP
  - Složen z komponent:
    - Record Layer Protocol – umožňuje integraci s protokoly vyšší úrovně (http, ftp, …)
    - Handshake Protocol – úvodní domluva parametrů a autentizace
    - Change Cipher Specification Protocol – použití nových parametrů šifrování
    - Alert protocol – informace o chybách a varováních
- **IPsec (Internet Protocol Security)**
  - IPv4 není dostatečně bezpečný ⇒ vývoj IPsec, který je součástí IPv6
  - Řeší:
    - Autentizaci původu dat – každý datagram je ověřován, zda byl odeslán uvedeným odesilatelem
    - Integrita dat – ověřuje se, zda data nebyla při přenosu změněna
    - Důvěrnost dat – data jsou před přenosem šifrována
    - Ochrana před útokem přehráním – útočník nemůže zneužít odposlechnutou komunikaci k útoku přehráním
    - Správa klíčů
- **PAP (Password authentication protocol)**
  - Autentizační protokol, kde hesla cestují v otevřené formě po sítí, ale zas mohou být uložená u poskytovatele jako haše.
- **CHAP (Challenge-Handshake Authentication Protocol)**
  - Challenge/response protokol,
  - Obě strany znají heslo, neposílá se přes síť
- **RADIUS a TACACS+**
  - Služby pro autentizaci a následnou autorizaci přístupu v UNIX sítích.
  - RADIUS je složitější, používá challenge response, může být hiearchický a umí i záložní server.
  - Protokol je rozšiřitelný.

- WiFi sítě
    - WEP – Wired Equivalent Privacy
      Starší, zavrhnutý protokol, dneska je ho možné nabourat za pár minut
      Používal stream šifru RC4 a CRC součet pro integritu, klíč o velikosti 40 až 128 bitů
    - WPA – Wi-Fi Protected Access – momentálně používaný
      Ve verzích 1 a 2
      Generuje 128 bitový klíč pro každý paket

Typy síťových útoků

- Odposlouchávání (Eavesdropping, Wiretapping)
- Identity spoofing (IP spoofing) – vydávání se za někoho jiného
- Man-in-the-Middle – narušení komunikace dvou a více entit vydáváním se za druhou stranu
- Denial of Service (DoS) – útok přetížením, službu nadměrnými požadavky vyřadíme
- Skenování TCP/UDP portů
- Phishing – sociálně-inženýrský útok, od oběti získaváme heslo podvodnými stránkami a maily

Závěr

http://statnice.dqd.cz/mgr-szz:in-ins:5-ins

Slidy k PA159

Slidy k IPK (FIT)