**BI-SPOL-24 Protokolová rodina TCP/IP (IPv4, IPv6, TCP, UDP, aplikační protokoly). Řízení datového toku. Princip a využití NAT. Systém DNS**

BI-PSI

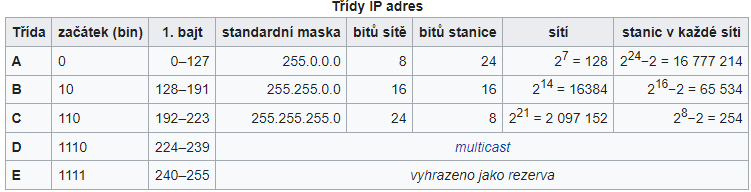
## Protokolová rodina TCP/IP

### IPv4

* IP = Internet Protocol
* Adresa je dlouhá **32 bitů**
* Rozlišuje stanice (zdroj, cíl)
* Síťové adresy jsou použity pro doručování jak v lokální síti, tak i mezi různými sítěmi
* V adresaci jsou zohledněny druhy provozu – Unicast, Broadcast, Multicast
* IP adresy je možné shlukovat do skupin (segmentů)
  + Pro určení velikosti skupiny se používá **síťová maska**, která má stejnou velikost (v bitech) jako IP adresa – značíme IP/PREFIX
  + Maska říká to, jakou část IP adresy mají všichni členové dané skupiny (sítě) společnou
  + Např. když bude prefix 20, znamená to, že prvních 20 bitů adresy bude pro všechny adresy stejný (IP adresu si napíšeme do binárky a podle toho víme) – zbylých 12 bitů umožňuje vytvořit různých adres
  + Maska sítě je potom 20 jedniček a 12 nul (v tomto pořadí)
  + Pro prefix 20 je maska 255.255.240.0
* Je několik schémat IP adresace
  + **CIDR** (classless inter domain routing) – prefix. Notace = IP/PREFIX – libovolná maska
  + Dle třídy (classfull design) – již zastaralé
* Adresy jsou tvaru: [0-255].[ 0-255].[ 0-255].[ 0-255] / maska
* Maska sítě – určuje adresy používané v síti odvozené od IP, schéma CIDR
* TTL – time to live – každý router sníží hodnotu o 1, ochrana proti zacyklení
* Adresa sítě – nejnižší adresa v daném rozsahu
* Broadcast – nejvyšší adresa

**Speciální adresní rozsahy**

* **Privátní adresy** (pouze v místních sítích)
  + 10.0.0.0/8
  + 172.16.0.0/12
  + 192.168.0.0/16
* **Loop-back** (provoz sám sobě, testování síť. Vrstvy)
  + 127.0.0.0/24
* Link-local (lokální, mohou se přiřadit automaticky
  + 169.254.0.0/16
* Multicast (doručování dat ve skupině)
  + 224.0.0.0/4
* Tyto rozsahy lze používat pouze v lokálních sítích. Ostatní adresové rozsahy IPv4 se označují jako veřejné a je možné je používat v internetu. Každý poskytovatel internetu má přidělenou (organizací spravující rozsahy dle příslušného kontinentu) určitou část těchto adresových rozsahů



**Fragmentace**

* Pokud je velikost IP paketu větší než maximální velikost části rámce linkové vrstvy určené pro data, musí se IP přenášet po částech (fragmentech)
* Fragmentaci indikují příznaky v hlavičce IP paketu, offset fragmentu udává jeho pořadí od začátku
* Složení diagramu nastává až na cílové stanici

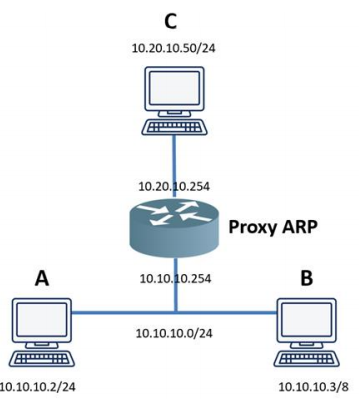
#### ARP protokol

* Address Resolution Protocol
* Důležitý pro doručování IP paketů v lokální síti – je potřeba znalost cílové MAC adresy
* V lokální síti probíhá doručování dat na základě MAC adres – aby bylo možné doručit data mezi počítači v lokální síti jen na zákaldě znalostí IP adres, je nutné umět namapovat tyto IP adresy na odpovídající MAC adresy síťových rozhraní
* ARP dokáže najít pro IP adresu její namapování na MAC adresu
* Proces komunikace pomocí ARP:
  + A odešle žádost na zjištění MAC B přes ARP protokol, který se namapuje do eth. rámce s broadcastovou cílovou adresou (0xFFFFFFFFFFFF)
  + B žádost zachytí a vyplní do dat odpovědi svoji MAC adresu
  + B odešle odpověď eth. rámcem A, ovšem již unicastově
  + A si MAC adresu přečte a zapíše si MAC B do své ARP tabulky.
  + A odešle data B unicastově přes eth. rámec linkovou vrstvou

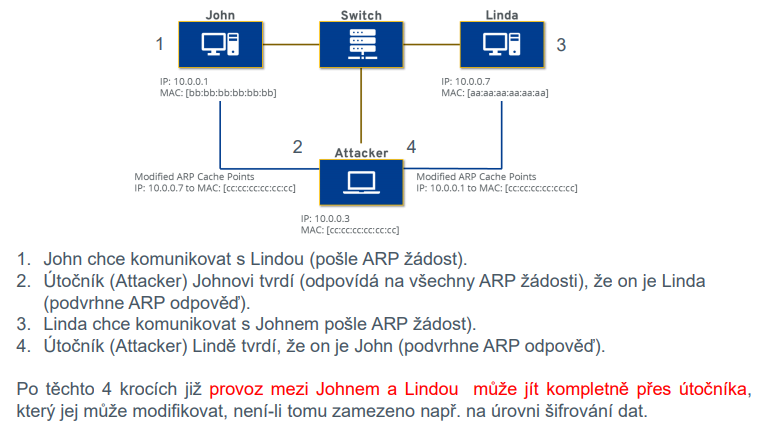
**Proxy ARP**

* Je mechanismus, díky kterému se směrovač pro zdrojovou stanici chová jako přepínač

Obsah obrázku text

Popis byl vytvořen automaticky

**ARP spoofing**



#### Hlavička paketu IPv4

Paket = datagram – základní jednotka přenosu dat na síťové vrstvě

Obsah obrázku text

Popis byl vytvořen automaticky

### IPv6

* 128 bitů adresa (8 skupin po 4 hexa) - adres
* Př. 2001:db8:3333:4444:5555:6666:7777:8888 – nuly můžeme zkracovat
* Náhrada (doplnění) IPv4 adres – větší adresní prostor
* Obsahuje metodu automatické konfigurace
* Překlad IP adres (NAT) není již potřeba – nicméně lze použít kvůli bezpečnosti, resp. oddělení síťového provozu vnější a vnitřní sítě
* Druhy komunikačních operací – unicast, multicast, anycast   
  (broadcast neexistuje – nahrazen specifickými multicasty)
* Hop Limit – obdoba TTL
* Pro řízení provozu používá protokol ICMPv6 – ten zabezpečuje celou sadu funkcí pro řízení provozu, která se označuje jako **objevování sousedů**.

**Druhy adres**

**Obsah obrázku text

Popis byl vytvořen automaticky**

* **Individuální globální adresy (IGA)** = obdoba veřejných adres v IPv4

**Pakety v IPv6**

* Paket v IPv6 = základní hlavička + (rozšiřující hlavička) + data
* *Základní hlavička* = struktura, která obsahuje informace, které souvisí se základními parametry nutnými pro doručení paketu (např. zdrojová + cílová adresa, TTL, velikost dat), je přítomna vždy.
* *Rozšiřující hlavička* = struktura, která obsahuje specifické informace sloužící pro doručení konkrétního paketu. Nemusí být přítomná vždy. Rozšiřující hlavičky se spojují za sebe
* *Data* = hlavičky a data protokolů vyšších vrstev
* Paket se může doručovat celý anebo po částech (fragmentace). Velikost fragmentu paketu na rozdíl od IPv4 určuje zdrojová stanice (u IPv4 směrovač) a po celou dobu doručování se tato velikost nemění (u IPv4 jí mohl měnit každý směrovač).

#### Hlavička paketu IPv6

Obsah obrázku stůl

Popis byl vytvořen automaticky

Obsah obrázku text, osoba, snímek obrazovky

Popis byl vytvořen automaticky

**Zřetězení hlaviček u IPv6**

* Protokol IPv4 definuje pro paket pouze jednu hlavičku, která obsahuje veškeré informace týkající se jeho doručování (např. fragmentace, offset atd).
* Paket protokolu IPv6 může obsahovat hlaviček více, první hlavička se nazývá základní a další se označují jako rozšiřující.
* Rozšiřující hlavičky jsou umístěné za sebou v definovaném pořadí, každá má specifickou funkci (např. směrování, fragmentace, určení typu přenášených dat atd.), rozlišenou kódem.
* Pro zřetězení hlaviček se používá specifická položka v každé hlavičce – další hlavička.
* Velká výhoda zřetězování hlaviček spočívá v tom, že se přenášejí pouze hlavičky, které jsou skutečně potřeba a oproti hlavičce obsažené v IPv4 mohou být rozšiřující hlavičky podrobnější.
* Počet rozšiřujících hlaviček může být pro každý paket individuální

#### Objevování sousedů (Neighbor Discovery)

* Komplexní způsob pro řízení provozu IPv6.
* Nabízí řadu funkcí jako např.
  + **Automatickou konfiguraci koncových stanic** (obdoba DHCP pro IPv4).
  + **Zjišťování linkových adres v lokální síti** (náhrada ARP pro IPv4).
  + Rychlá aktualizace neplatných položek
    - Detekuje nedosažitelné IPv6 adresy (např. kvůli vypnutí stanice).
  + Hledání směrovačů
    - Automaticky zjišťuje, přes které směrovače je možné přeposílat provoz určený pro stanice umístěné v jiných sítích.
  + Přesměrování
  + Ověřování dostupnosti sousedů (v místní síti, u známých sousedů).
  + Detekce duplicitních adres (hledání a deaktivace chybných adres)

**Automatickou konfiguraci koncových stanic**

Automatická konfigurace = metoda, jejímž účelem je získání individuální globální adresy IPv6 a dalších parametrů, které umožní stanici komunikovat uvnitř místní sítě i Internetu.

* Automatická konfigurace může být bezstavová nebo stavová
* *Bezstavová*
  + založená na myšlence, že v síti existují směrovače, které mají kompletní informace týkající se provozu a dokáží je předat dále
  + směrovač zasílá do sítě zprávu **Ohlášení směrovače** (RA = Router Advertiment), která obsahuje konfiguraci sítě – posílá se periodicky nebo na výzvu
* *Stavová*
  + v síti existuje specializovaný server, který zajišťuje přidělování adres, používá se protokol DHCPv6

**Zjišťování linkových adres v lokální síti**

* Pro komunikaci v lokální síti je nutné mimo IPv6 adres i znát odpovídající MAC adresy rozhraní okolních stanic, na kterých jsou tyto MAC adresy nastaveny.

Obsah obrázku text, osoba, snímek obrazovky

Popis byl vytvořen automaticky

**Transportní vrstva**

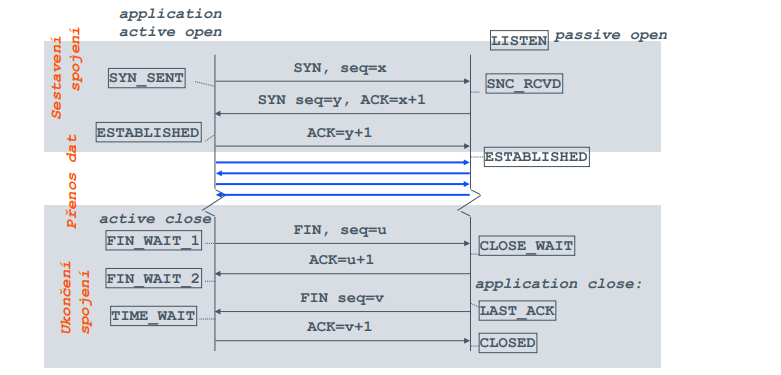
* Transportní vrstva – Hlavním úkolem této vrstvy je doručení dat ke konkrétní službě/aplikaci (procesu) běžícím na stanici dosažitelné prostřednictvím konkrétní IP adresy
* Transportní vrstva má za úkol zajišťovat kontrolu toku dat a opravy chyb (např. ztracených paketů).
* Rozlišení konkrétní zdrojové a cílové aplikace je založeno na použití **portů**.
* Port = 16 bitový identifikátor, který se uplatňuje jako další adresační stupeň

### TCP (Transmission Control Protocol)

* Protokol pro přenos - v transportní vrstvě
* TCP paket = položka data IP protokolu
* Umožňuje řízení toku (klouzavé okno) i eliminaci zahlcení
* Zaručuje spolehlivé dodání dat – zabezpečený oproti chybám, garantuje i pořadí doručených paketů.
* Není šifrovaný
* Detekuje duplicitní pakety
* Je duplexní (odesílání a příjem z obou stran, každá strana otevírá vlastní spojení)
* Používáme tak, kde chceme přesnost a neztrátovost a nevadí nám větší režie
* Nevhodné pro realtime aplikace (streaming apod.)
* Využito v http, FTP, SMTP
* Spojově orientovaný – mezi směrovači se sestavuje a drží virtuální spojení
  + Naváže spojení – odesílá data – ukončí spojení

**Navázání spojení**

* Zahájení – handshake (SYN, SYN+ACK, ACK)
* Ukončení – FIN, FINACK, FIN, ACK



Sekvenční číslo (seq) = pořadové číslo paketu. Počáteční hodnota je domluvena při handshaku (vygenerováno náhodné číslo)

**TCP paket**

* Žlutá barva = hlavička

Obsah obrázku stůl

Popis byl vytvořen automaticky

* Pro řízení toku se používá mechanismus klouzavého okna

### UDP (User Datagram Protocol)

* Protokol transportní vrstvy
* Přenáší se v položce data paketu IP protokolu
* Používá se pro jednoduchou výměnu dat.
* V ideálních podmínkách má lepší propustnost než TCP kvůli nižší režii
* Používá se běžně např. pro přenos hlasu či videa (např. RTP = Real Time Protocol), kde občasná ztráta paketů tolik nevadí
* **Nezabezpečený**
  + Negarantuje doručení paketů a ani pořadí jejich doručení – neposílá znovu poškozená nebo ztracená data
* Žádná kontrola přenášených dat, aby se zabránilo zahlcení cílového zařízení – neřeší
* Prostě jen odešle a nestará se
* Patří mezi nespolehlivé protokoly
  + Paketově orientované, data se posílají po částech a spojení se nevytváří.
  + Jednodušší, snažší implementace oproti spolehlivým.

**Formát hlavičky (žlutá) + data**



### Aplikační protokoly

* Úkolem aplikační vrstvy je interakce uživatele a vzdálených síťových služeb

Obsah obrázku stůl

Popis byl vytvořen automaticky

Obsah obrázku stůl

Popis byl vytvořen automaticky

* Popis protokolů v přednášce č. 9

## Řízení datového toku

### Algoritmy pro doručování dat

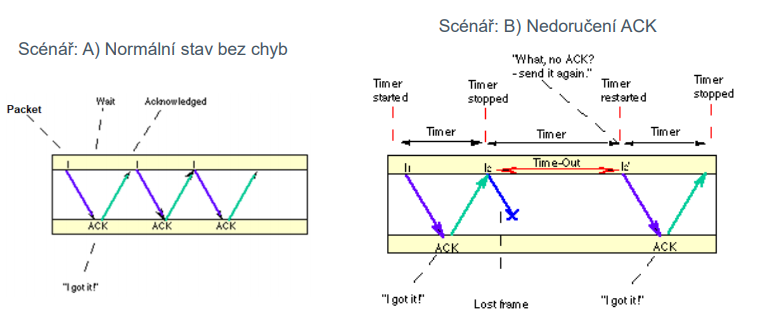
* (Na transportní vrstvě)
* **Stop & Wait** – spolehlivé doručování dat
* **Stop & Go** – pro řízení toku
* **Klouzavé okénko** – pro spolehlivé doručování dat i řízení toku

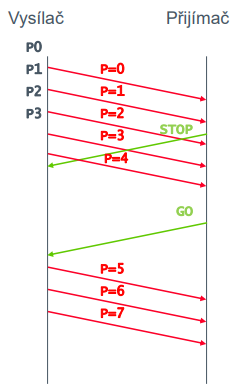
#### Stop & Wait

* Jednoduchý, ale málo efektivní
* Doručuje data přesně v tom pořadí, jak jdou za sebou

**Scénáře**

* Normální stav bez chyb
  + Vysílač pošle paket a čeká do doby, než přijde potvrzení. Pak se pošle další paket
* Nedoručení ACK
  + Vysílač pošle paket, na který nedostane potvrzení. Jakmile na vysílači vyprší timeout indikující chybějící potvrzení, pošle vysílač znovu tentýž paket



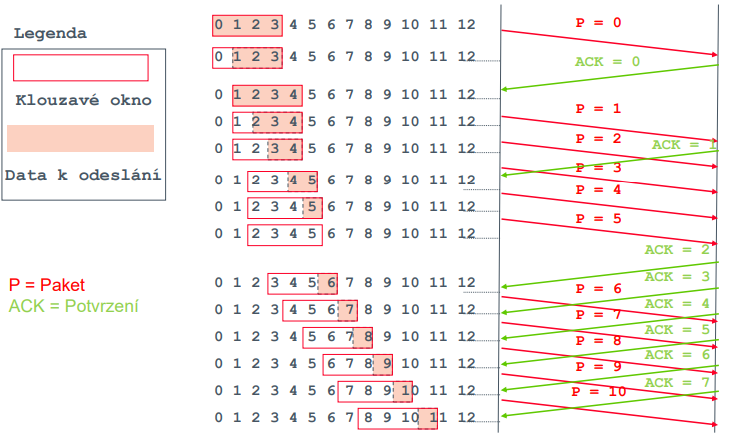


#### Stop & Go

* Dokáže ovlivnit množství paketů, které přichází od vysílače.
* Nedokáže vyřešit problém ztráty paketů
* Vysílač posílá pakety přijímači (posílá posloupnost paketů) dokud mu přijímač nepošle STOP – přestane
* Vysílač čeká na zprávu GO – poté zase vysílá

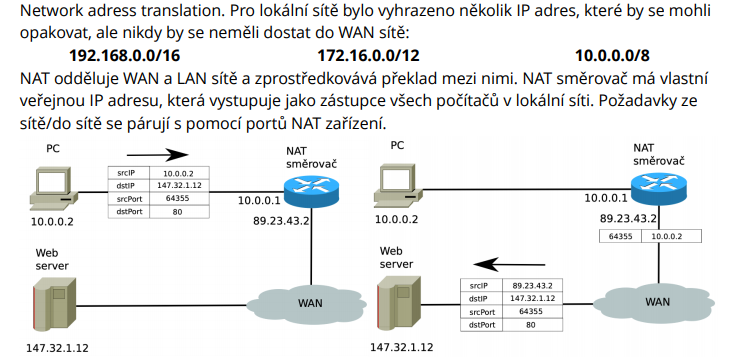
#### Klouzavé okénko (Sliding Window)

* Složitější, avšak efektivnější než Stop & Wait.
* Dovoluje řízení toku úpravou velikosti okna.
* Úprava velikosti okna se různými mechanismy optimalizuje na základě parametrů přenosu (delay, RTT, packet loss)
* Můžeme říct odesílateli rychlost, kterou by měl odesílat
* Pokud se začne příliš ztrácet – zpomalíme – udržujeme optimální rychlost
  + Snažíme se najít optimální řešení – nejvyšší rychlost při malé ztrátovosti
* Chodí potvrzení, ale třeba i zpožděně



## NAT

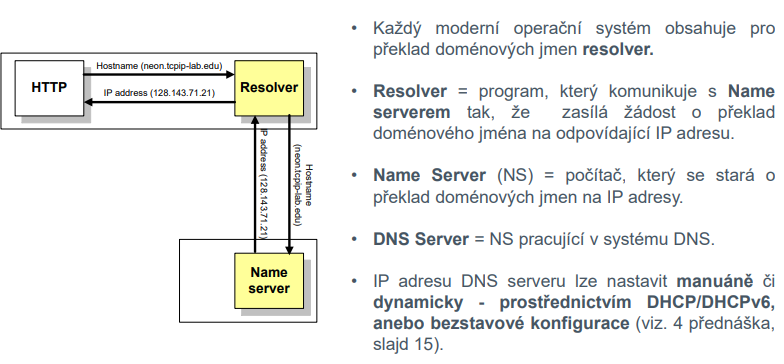
* IPv4 adres je nedostatek – nelze každému PC přiřadit veřejnou adresu v internetu
* Řešením je, že všechny privátní adresy se při průchodu do vnější sítě tváří jako jedna veřejná adresa
* Směrovač při průchodu vyměňuje zdrojovou adresu
* NAT = Network Address Translation – převádí privátní adresy na veřejné
* Pro lokální sítě jsou vyhrazeny 3 rozsahy IP adres
* *Masquerade (maškaráda):* všechny zařízení vystupují jako jedno do WAN sítě
  + Rozdíl oproti NAT je v tom, že *Masquerade* pracuje s dynamicky přiřazenými IP adresami
  + NAT pracuje pouze se statickými
* *Port-Forwarding:* pokud by se chtělo přistoupit do konkrétního počítače z WAN do LAN, tak by zařízení muselo forwardovat jistý port na jednu konkrétní IP adresu v LAN – např. aby měl někdo z internetu přístup k serveru, který je uvnitř lokální sítě



## Systém DNS

* systém primárně určený pro překlad doménových jmen na IP adresy a naopak
* DNS je hierarchický
* V původní verzi nezabezpečený
* DNSSEC = zabezpečené rozšíření DNS tak, aby bylo možné zabránit zfalšování odpovědi
* DNS může kromě překladu doménových jmen provádět i další funkce (např. rozdělování zátěže či omezení přístupu).
* DNS běží standardně na portu 53 a většina jeho implementací využívá UDP protokol.

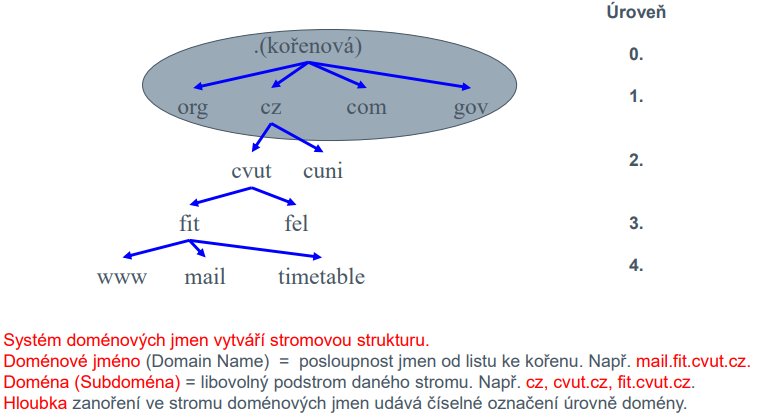
**DNS v rámci OS**



Obsah obrázku text

Popis byl vytvořen automaticky

**Hierarchie domén a doménových jmen v DNS**



TopLevelDomain - .cz, .com spravuje sdružení cz.nic

Obsah obrázku text

Popis byl vytvořen automaticky

#### DNS protokol

Obsah obrázku text

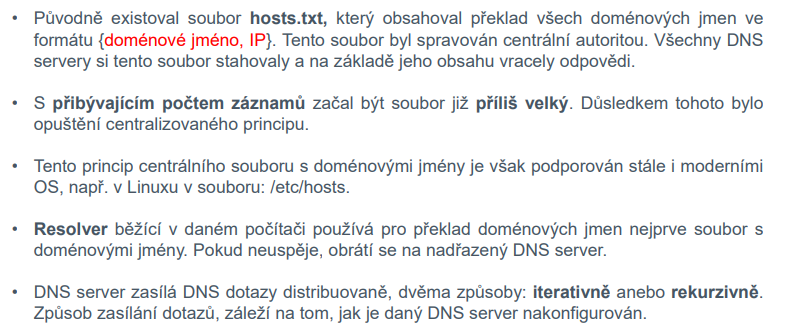
Popis byl vytvořen automaticky

* Obecný formát DNS dotazu/odpovědi

Obsah obrázku stůl

Popis byl vytvořen automaticky

**Princip překladu jmen v DNS**



**Iterativní způsob dotazování v DNS**

Obsah obrázku text

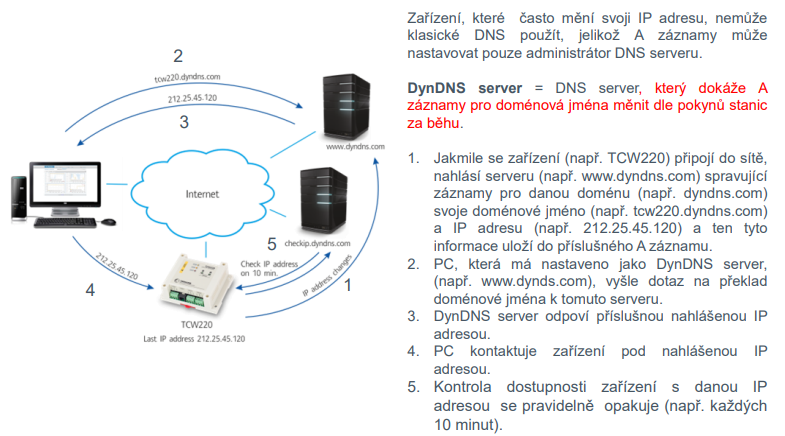
Popis byl vytvořen automaticky

**Rekurzivní způsob dotazování v DNS**

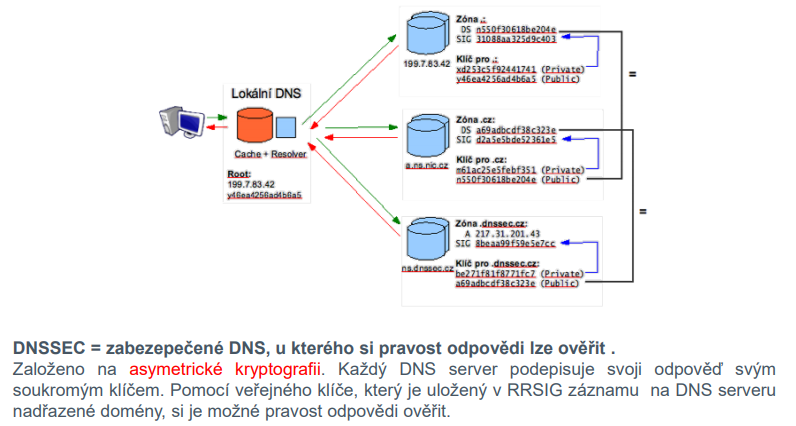
**Obsah obrázku text

Popis byl vytvořen automaticky**

#### Dynamické DNS (DynDNS)

****

#### DNSSEC – bezpečnější DNS



#### Útok na DNS – Cache Poisoning

