

KOMUNIKAČNÍ TECHNOLOGIE (BPC-KOM)

Ústav telekomunikací

Fakulta elektrotechniky a komunikačních technologií

VUT v Brně

doc. Ing. Jan Jeřábek, Ph.D.

jerabekj@feec.vutbr.cz

APLIKAČNÍ VRSTVA PŘENOSOVÝCH SYSTÉMŮ



Plán přednášky

3

- Úvod do aplikační vrstvy
- DHCP
- DNS
- TELNET
- FTP
- WWW a HTTP
- E-mail a SMTP
- VoIP protokoly

Úvod do aplikační vrstvy

4

- zahrnuje komunikační funkce specifické pro konkrétní aplikační procesy
- Funkce
 - ▣ mohou realizovat lokální nebo vzdálené programy
 - ▣ vybrané funkce
 - Identifikace účastníků komunikace
 - Zjištění dostupnosti účastníka komunikace
 - Umožnění přístupu k požadovaným zdrojům
 - Stanovení metod pro opravu chyby, potvrzování
 - Bezpečnost dat
 - Management sítě

Úvod do aplikační vrstvy

5

- okno do OSI systému pro uživatelské procesy
- v případě TCP/IP zahrnuje aplikační vrstva i relační a prezentační funkce
- existují stovky protokolů, několik vybraných:
 - ▣ **HTTP** (*Hypertext Transfer Protocol*)
 - základní přenosový protokol ve WWW prostředí
 - ▣ **FTP** (*File Transfer Protocol*)
 - pro přenos souborů
 - ▣ **SMTP** (*Simple Mail Transfer Protocol*)
 - hlavní protokol pro přenos elektronické pošty
 - ▣ **DNS** (*Domain Name System*)
 - pro práci se jmennými názvy (adresami) v celém Internetu
 - ▣ **DHCP** (*Dynamic Host Configuration Protocol*)
 - pro centralizovanou správu IP adres na lokální síti
 - ▣ **TELNET** (*Telecommunication network protocol*)
 - pro vzdálený terminálový přístup k jinému systému
 - ▣ **VoIP** (Voice over IP) protokoly

DYNAMIC HOST CONFIGURATION PROTOCOL (DHCP)



Základní vlastnosti DHCP

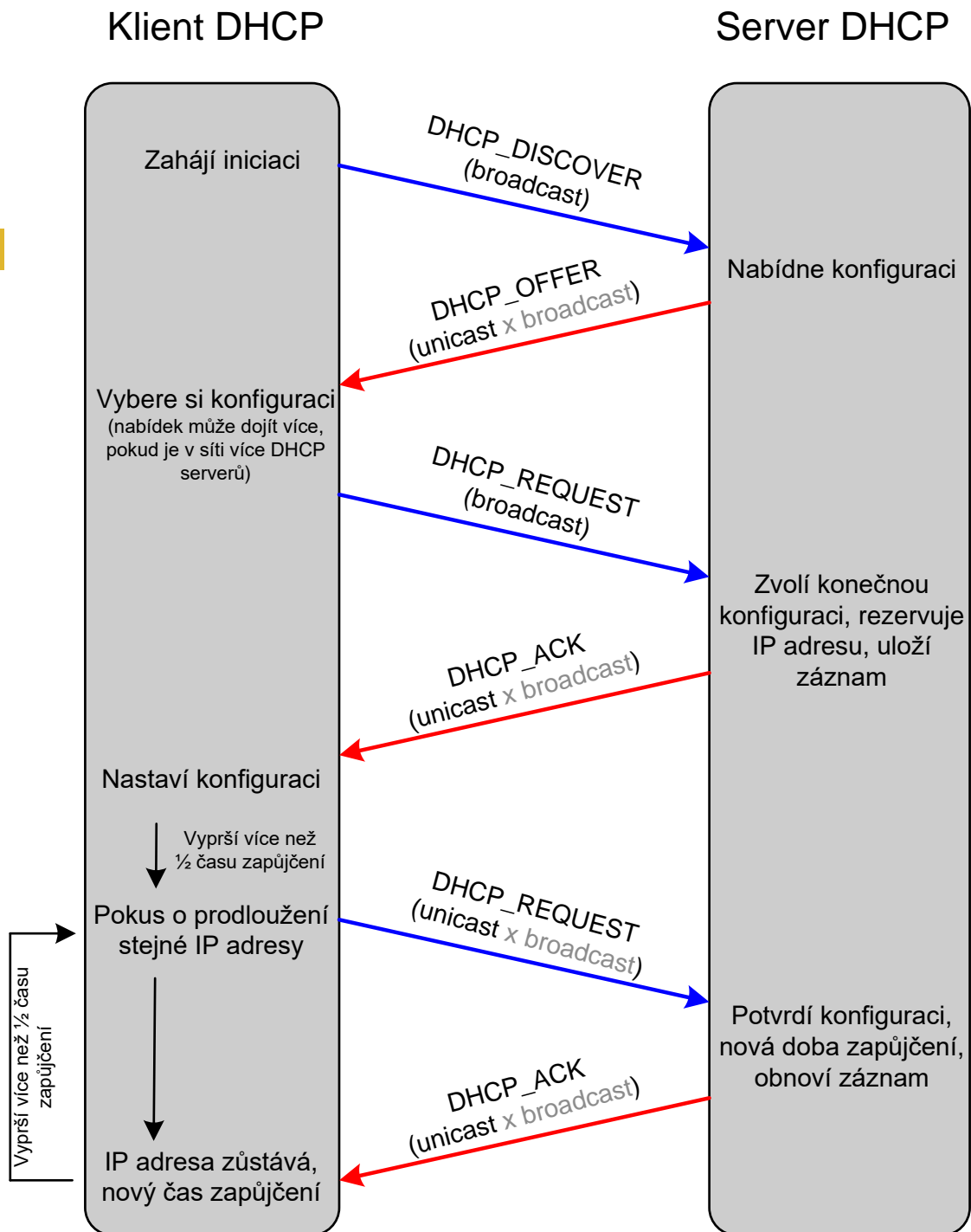
7

- Aplikační protokol typu klient-server
 - ▣ Primárně slouží síťové vrstvě
 - ▣ Jako transportní protokol využívá nespolehlivé UDP
- Pro dynamické nastavení parametrů sítě
 - ▣ **IP adresa**, maska sítě, adresa výchozí brány, ...
 - ▣ Získané parametry mají omezenou časovou platnost
- Rozšířením staršího protokolu BOOTP, vzájemně kompatibilní
- Výhody dynamické konfigurace
 - ▣ Jednodušší správa adresního prostoru
 - ▣ Eliminace možnosti výskytu duplicitních IP adres v síti
 - ▣ Snadnost přečíslování celé sítě
 - ▣ Uživatel nemusí nic nastavovat, *nemusí rozumět problematice IP adres*, musí mít jen povolené DHCP ve svém operačním systému

Princip činnosti DHCP

8

- Stanice neví nic o síti
- Stavová komunikace
- Další zprávy
 - ▣ DHCP_NAK
 - Zamítnutí požadavku
 - ▣ DHCP_RELEASE
 - Vzdání se konfigurace ze strany klienta
 - ▣ DHCP_INFORM
 - Žádost o další informace (lokální konfigurační parametry)
- × DHCPv6



DHCP relay agent

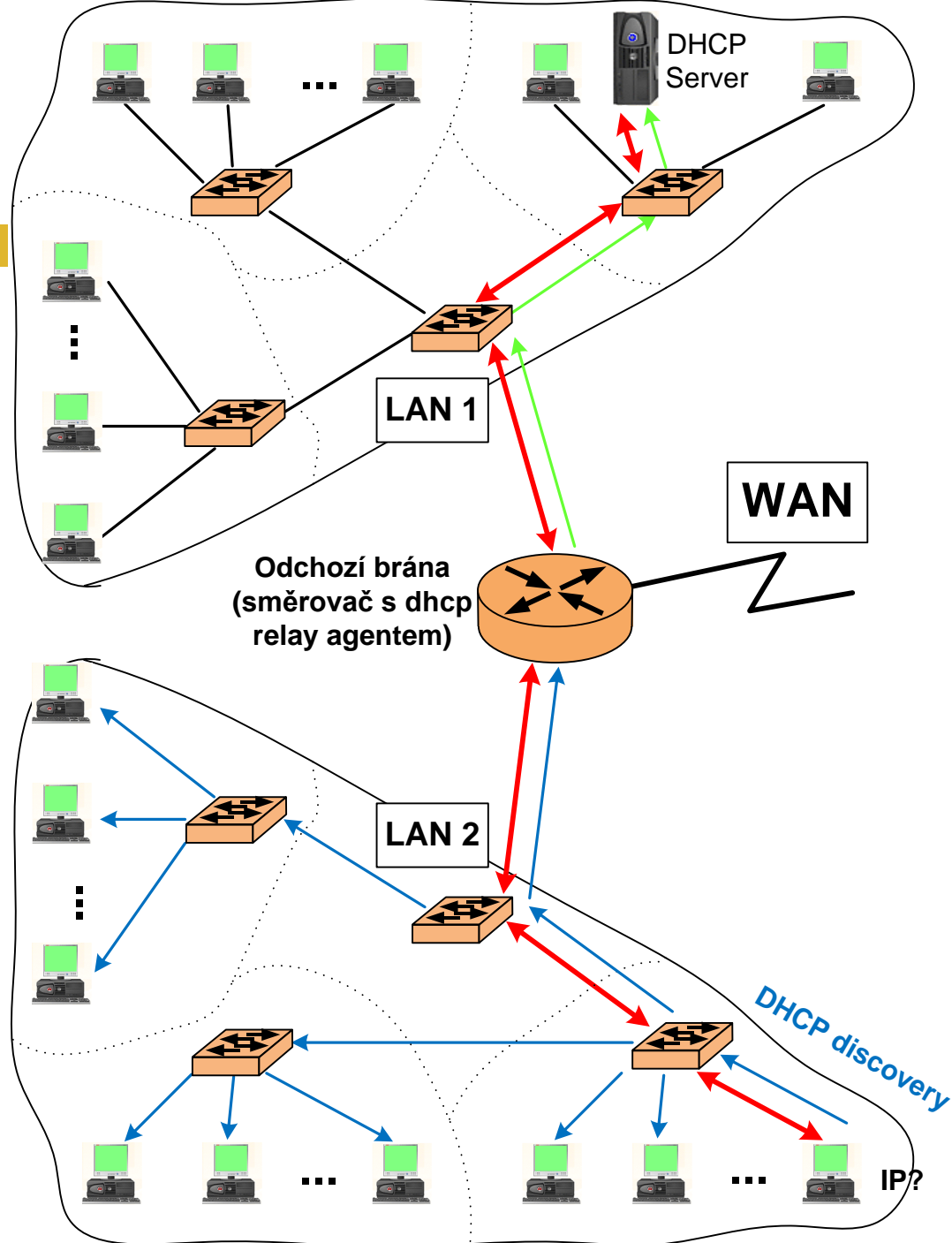
9

Situace:

- Síť s jednotnou správou se skládá z více menších LAN sítí, zbytečné mít v každé DHCP server !?
- Stačí jeden/dva, které ale musí být dostupné pro všechny
- Problém - DHCP_DISCOVERY je broadcast (pouze v rámci konkrétní LAN)

Řešení

- **DHCP relay agent** na směrovači
- Agent přeposílá zprávy DHCP do sítě kde se nachází server DHCP
- Při předání přidává informaci o síti, z které požadavek vzešel



DOMAIN NAME SYSTEM (DNS)



Motivace existence jmenového systému

11

- IP adresy – identifikace koncových/transportních uzlů
- Jejich přímé zadávání problematické
- Změna IP adresy služby, např. serveru
 - ▣ Fyzické přestěhování
 - ▣ Přechíslování IP adres v síti
 - ▣ Změna poskytovatele připojení
- ↓
- jak informovat uživatele/stroje?
- K zajištění dostupnosti stačí změna DNS záznamu (odkaz)
- Spojení pak směřováno na novou IP adresu serveru
- ↓
- DNS osvobozuje uživatele/stroje od problémů ze síťové vrstvy

Základní popis protokolu DNS

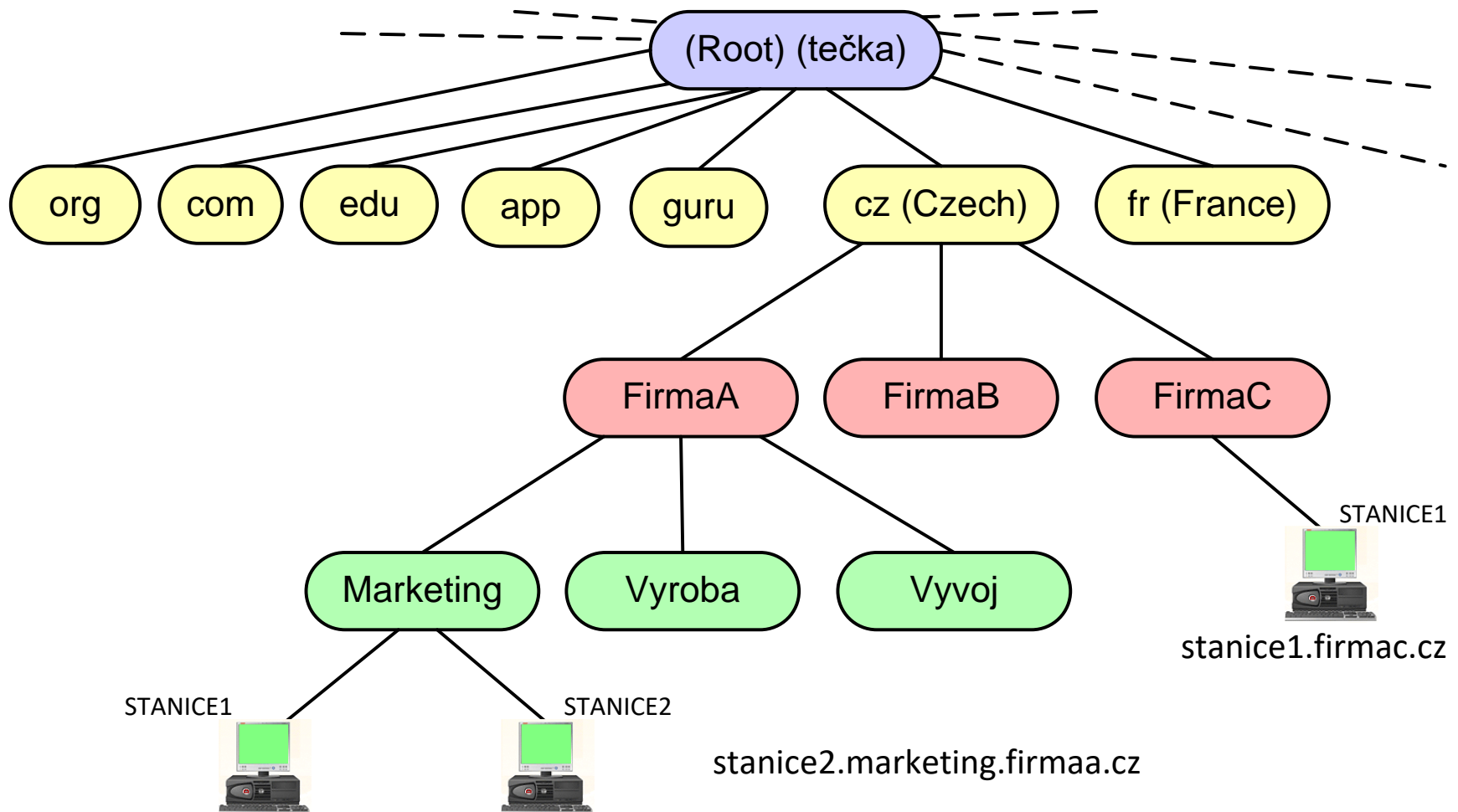
12

- Aplikační protokol typu klient-server
- Nahrazuje abstrakci na úrovni síťové vrstvy (IP adresy) ještě vyšší úrovní abstrakce – jmenné názvy (DNS jména) -> **mapování**
- Příklady
 - 77.75.76.3 -> www.seznam.cz (WWW)
 - 147.229.72.16 -> fest.stud.feec.vutbr.cz (E-mail)
- *Decentralizovaný* a hierarchický systém
- Jako transportní protokol využívá jak UDP tak TCP
- DNS server (jmenný server)
 - Musí být zadán IP adresou
 - Překlad (mapování)
 - Reverzní mapování

Základní popis protokolu DNS

13

- Hierarchický systém – hierarchie domén – definované úrovně



Domény a doménová jména

14

□ Hierarchický systém

- ▣ Domény, subdomény (domény druhého a vyššího řádu)
- ▣ Vyhodnocování zprava doleva

□ Distribuovaný systém

- ▣ DNS server není jeden, ale je jich velké množství, rozmístěny v rámci Internetu
- ▣ Žádný server neobsahuje kompletní mapovací tabulku, ale pouze část
- ▣ Každá větší síť by měla mít vlastní (rekurzivní) DNS server
- ▣ Každý DNS server by měl být schopen kontaktovat jiný DNS server, pokud nenajde ve svých záznamech požadovanou položku – viz dále

Domény a doménová jména

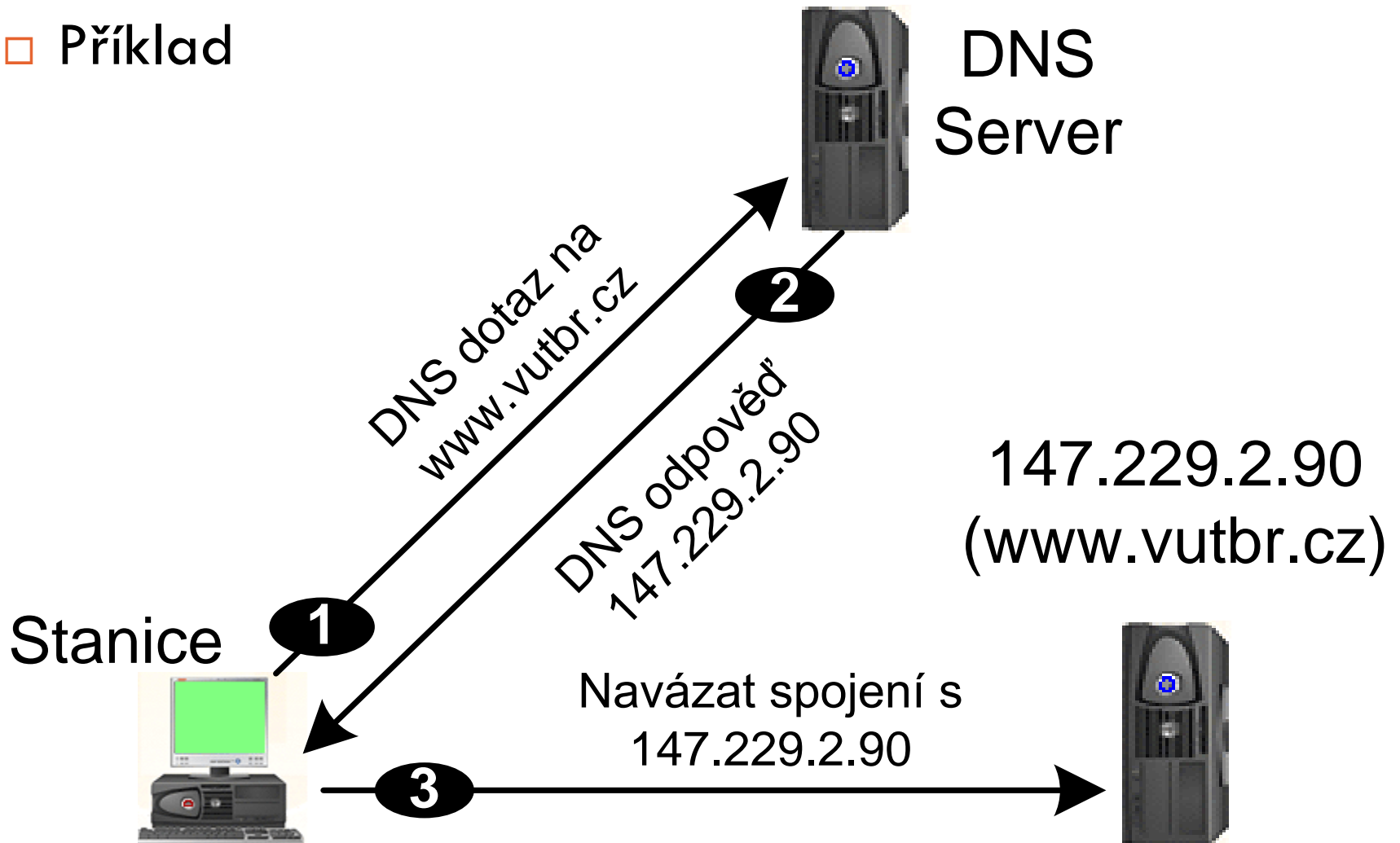
15

- Historická pravidla pro tvorbu DNS názvu
 - ▣ Celkem maximálně 255 znaků
 - ▣ Jednotlivé řetězce odděleny tečkami (každý max. 63 znaků)
 - ▣ Pouze písmena, číslice a pomlčky
- Současná pravidla pro tvorbu DNS názvu
 - ▣ Dobrá implementace -> libovolné 8-bit znaky
 - ▣ Existují rozšíření (diakritika, další znaky) – viz dále
- Bezpečnost DNS
 - ▣ Velmi nízká vzhledem ke kritické důležitosti služby
 - ▣ Nezabezpečený přenos dat -> důraz na rychlost
 - ▣ Záměrné x neúmyslné poškození databáze
 - ▣ DNSSec

Základní princip komunikace v DNS

16

□ Příklad



Resolver

17

- V rámci OS se o DNS stará klient – **stub resolver** (Win x Linux)
 - Zprostředkovává komunikaci s rekurzivním serverem DNS
 - Udržuje lokální dočasnou databázi překladů
 - Pracuje s tabulkou statických překladů (pokud jsou definovány)
 - Dotazování zpravidla vícenásobné pokud počítač členem určité domény
 - Příklad: dotaz na www.mit.edu v doméně utko.feec.vutbr.cz
 1. www.mit.edu[.utko.feec.vutbr.cz](http://utko.feec.vutbr.cz) -> neexistuje
 2. www.mit.edu[.feec.vutbr.cz](http://feec.vutbr.cz) -> neexistuje
 3. www.mit.edu[.vutbr.cz](http://vutbr.cz) -> neexistuje
 - (4. www.mit.edu.cz) -> nedotazováno
 5. www.mit.edu -> úspěch

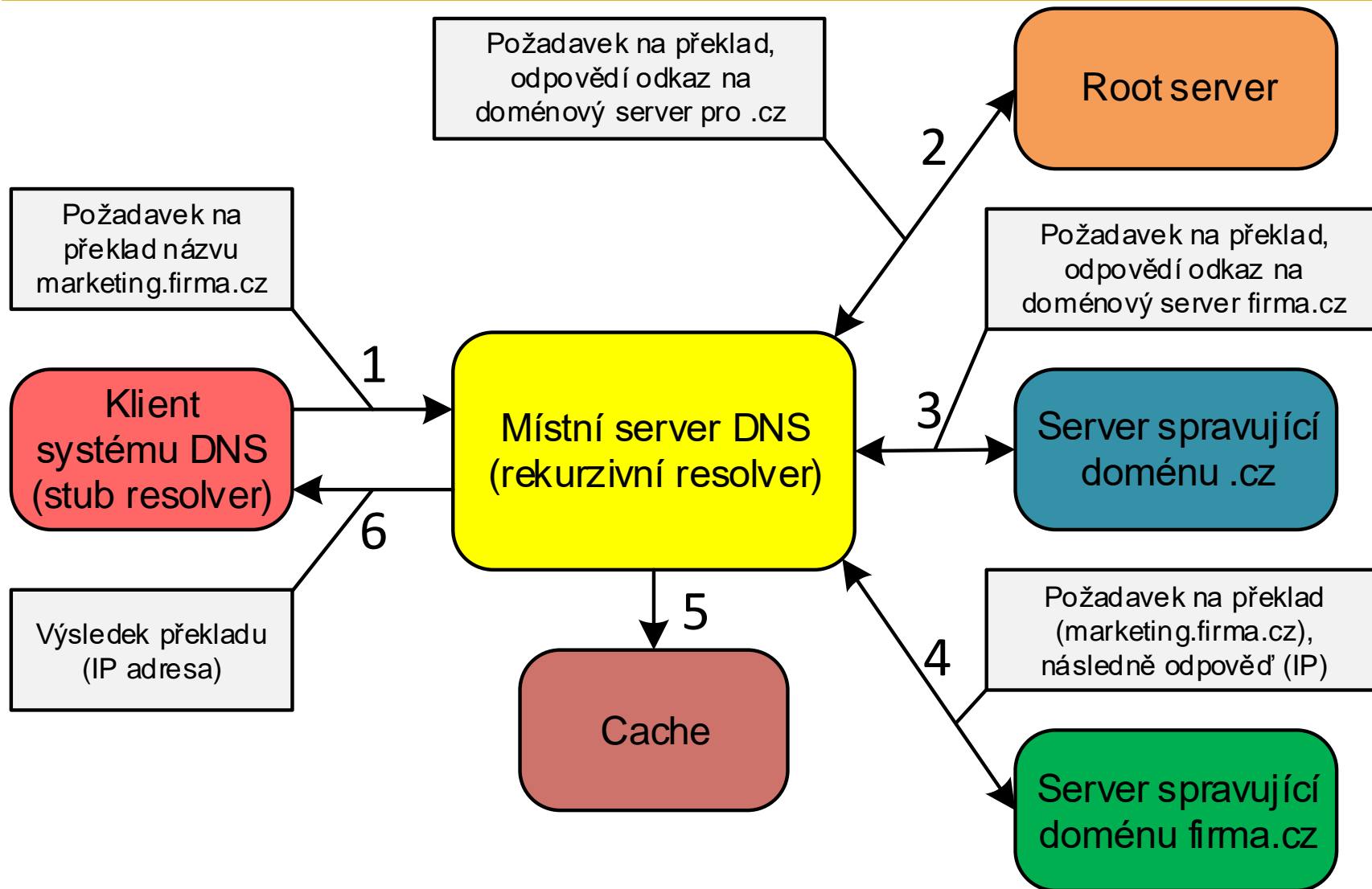
Resolver

18

- Př.: uživatel zadá **adela** v doméně **utko.feec.vutbr.cz**
 1. **adela.utko.feec.vutbr.cz** -> úspěch
- Takto nastavený stub resolver umožňuje usnadnění lokální komunikace, stačí zadávat lokální názvy, platné v rámci domény
- Vychází se z priority rychlé a hlavně snadné vnitřní komunikace
- Na DNS serveru se také nachází resolver (**rekurzivní**)
 - Pracuje na stejném principu
 - Při vyřizování žádosti pracuje s lokální mapovací tabulkou, dočasnou pamětí, vzdálenými DNS servery

Hierarchie DNS serverů

19



Hierarchie DNS serverů

20

- 13 serverů (reálně 1 100+ lokalit) <http://www.root-servers.org/>
- 12 organizací
- obsluha root zóny
- anycast -> distribuovaná služba
- X.root-servers.net, X-> {A, B, , ..., M}
- Na každý z nich >1M dotazů/sec
 - ▣ cca 1/2 užitečné dotazy

Typy DNS záznamů

21

- DNS záznamy různého typu – uloženy ve větách RR, přenos
- Typy
 - ▣ **A** (*A host address*) – 32-bitová IPv4 adresa, slouží jako výsledek překladu
 - ▣ **NS** (*Authoritative name server*) – doménové jméno DNS serveru, který je autoritou pro danou doménu, používá se k přesměrování na jiný DNS server
 - ▣ **MX** (*Mail eXchanger*) – speciální záznam pro mail servery domény
 - ▣ **AAAA** (*IPv6 address*) – 128-bitová IPv6 adresa, slouží jako výsledek překladu
 - ▣ **CNAME** (*Canonical name for an alias*) – doménové aliasy (další možné názvy domény, které odkazují na stejný stroj)
- Možné poslat dotazy na více typů, nemusí existovat všechny

Registrace domén

22

- ❑ Registrace nutná – dostupnost serverů
- ❑ Specifická pravidla, zákony, technické okolnosti
- ❑ Celosvětově organizuje IANA, u nás spravuje CZ.NIC (doména .cz), resp. cca 45 registrátorů
- ❑ Registrovaných domén .cz – 1,1 M (počátek 2014)
- ❑ Registrace „volné“ domény
 - ▣ Kontaktní osoby s různými rolemi
 - ▣ Sada jmenných serverů
- ❑ Ověření registrace – služba WHOIS

TELNET

Telnet

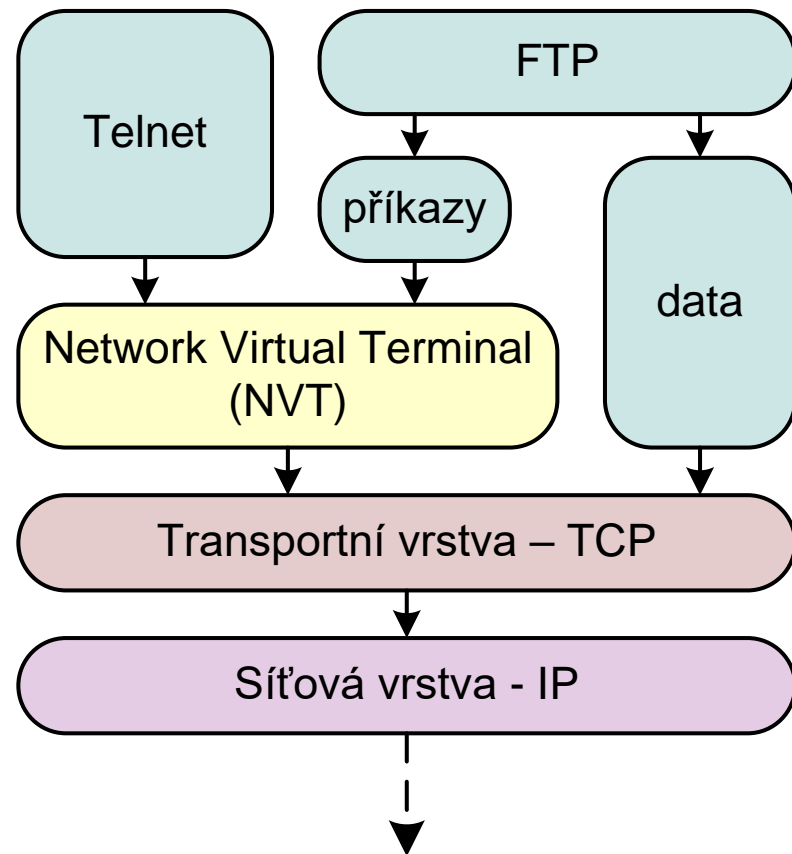
24

- *Telecommunication network protocol*
- Protokol typu klient-server
- Platný standard z roku 1983
- Jako transportní využívá spolehlivý protokol TCP
- Umožňuje
 - Přihlášení na vzdálený stroj (server) ze vzdáleného počítače (klient)
 - Přihlášení na základě kombinace login/heslo (nezabezpečené !)
- ↓
- Často blokován administrátory (port TCP 23)
- Nahrazeno SSH

Network Virtual Terminal (NVT)

25

- ▣ Prezentační vrstva – prezentace znaků, řídicích příkazů
- ▣ Aby „A“ na klientovi bylo „A“ na serveru (obecně různé znakové sady), různé reprezentace
- ▣ Zavádí „síťovou“ standardní reprezentaci
 - Problematické mohou být zejména znaky:
 - Vymazání znaku (backspace); nový řádek (CR + LF, CR, LF)
 - Použitá reprezentace: ASCII kódy (8 bit) s přiřazeným významem u speciálních znaků
- ▣ NVT se používá obdobně i u FTP, SMTP, HTTP a dalších



Komunikace prostřednictvím Telnetu

26

Uživatelský terminál

Uživatelův vstup
v reprezentaci
klientského operačního
systému

Operační
systém
klienta

Telnet
klient

Uživatelův výstup
v reprezentaci
klientského operačního
systému

Uživatelův vstup
a příkazy klienta v NVT
reprezentaci

Internet /
Intranet

Uživatelův výstup
a příkazy serveru v NVT
reprezentaci

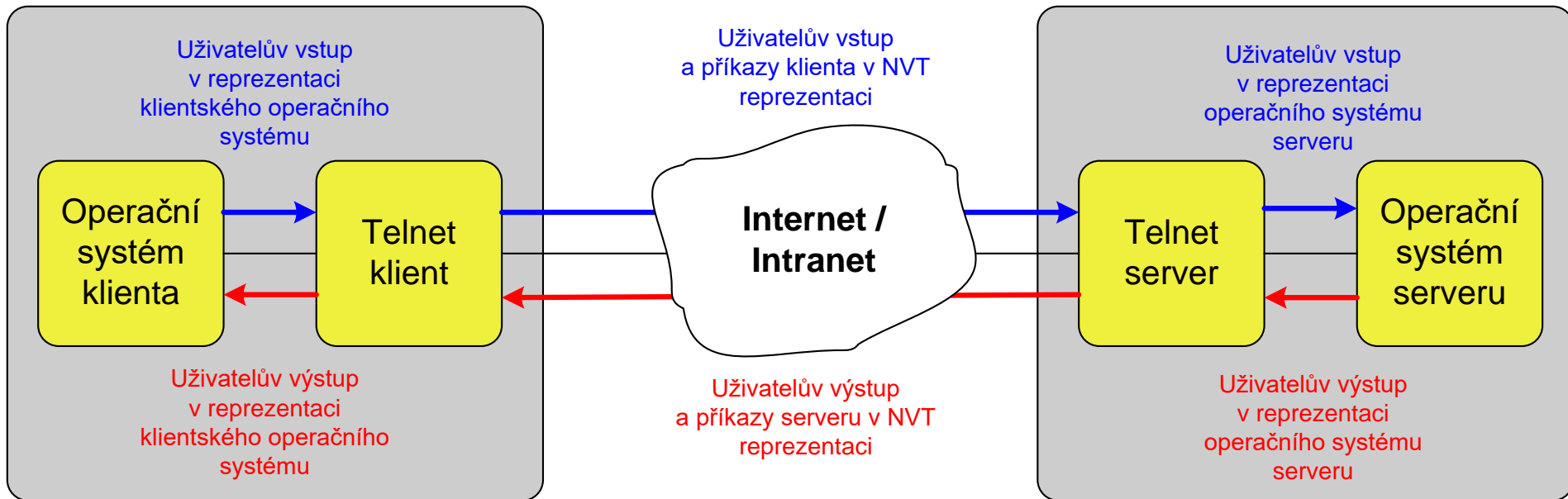
Vzdálený host

Uživatelův vstup
v reprezentaci
operačního systému
serveru

Telnet
server

Operační
systém
serveru

Uživatelův výstup
v reprezentaci
operačního systému
serveru



PŘENOS SOUBORŮ A PROTOKOL FTP



Základní popis protokolu

28

- Aplikační protokol, standard pro přenos souborů (RFC959 z roku 1985)
- Založen na modelu klient-server
- Přenos mezi počítači rozmístěnými libovolně v Internetu, libovolný OS
- Využívá transportní protokol TCP
- Unikátnost FTP: dvě oddělené spojení pro „jeden“ přenos
 - TCP/20 (na straně serveru) pro přenos dat
 - TCP/21 (na straně serveru) pro přenos řídicích zpráv
- Přenos dat v otevřené formě
- Spojení iniciuje klient, protokol je **stavový**
- Server klienta registruje a má o něm uloženy některé údaje
 - Ve kterém adresáři se momentálně nachází
 - Nastavený režim přenosu

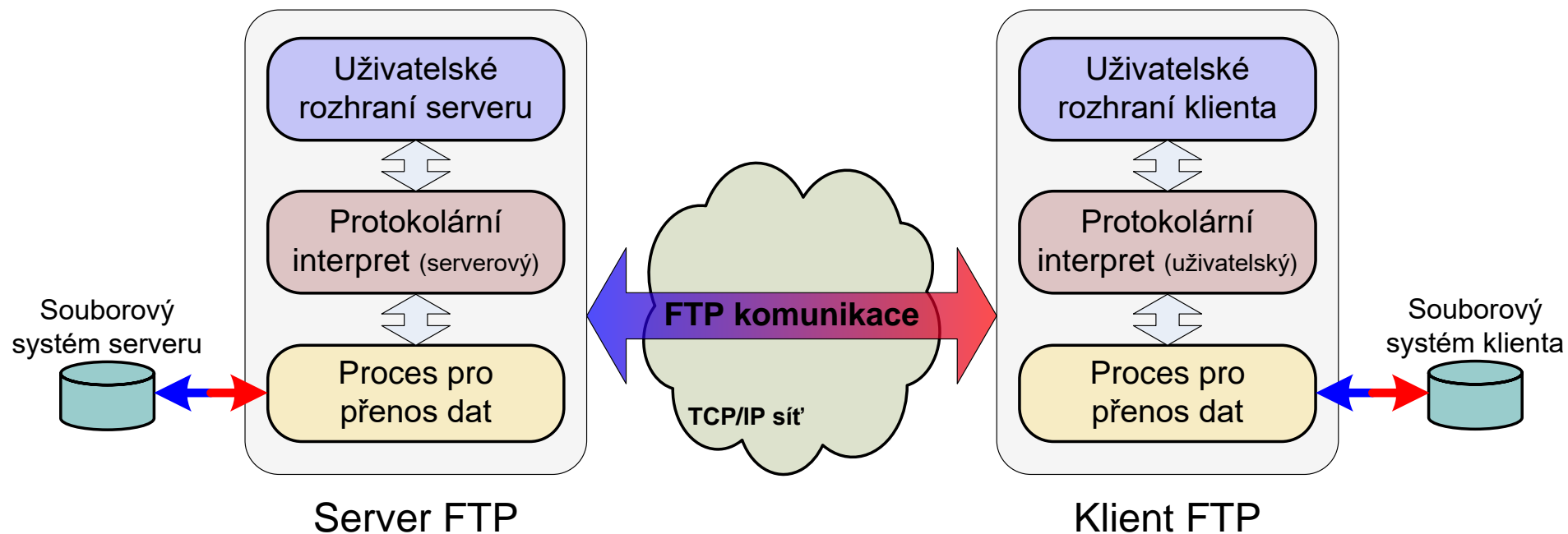
Průběh spojení FTP klienta se serverem

29

- Příklad průběhu spojení FTP klienta se serverem
 - ▣ Klient naváže řídicí TCP spojení na port 21 serveru
 - Spojení je **udržováno po celé trvání relace**
 - Jednoúčelové spojení pro výměnu textových příkazů a odpovědí (kódů)
 - ▣ Přihlášení klienta na server
 - Anonymně
 - Zadání dvojice údajů [login & heslo]
 - ▣ Klient posílá příkazy, server odpovídá (viz dále)
 - ▣ Přenos dat – nové **jednorázové** spojení TCP
 - ▣ Klient posílá další příkazy
 - ...
 - ▣ Ukončení spojení

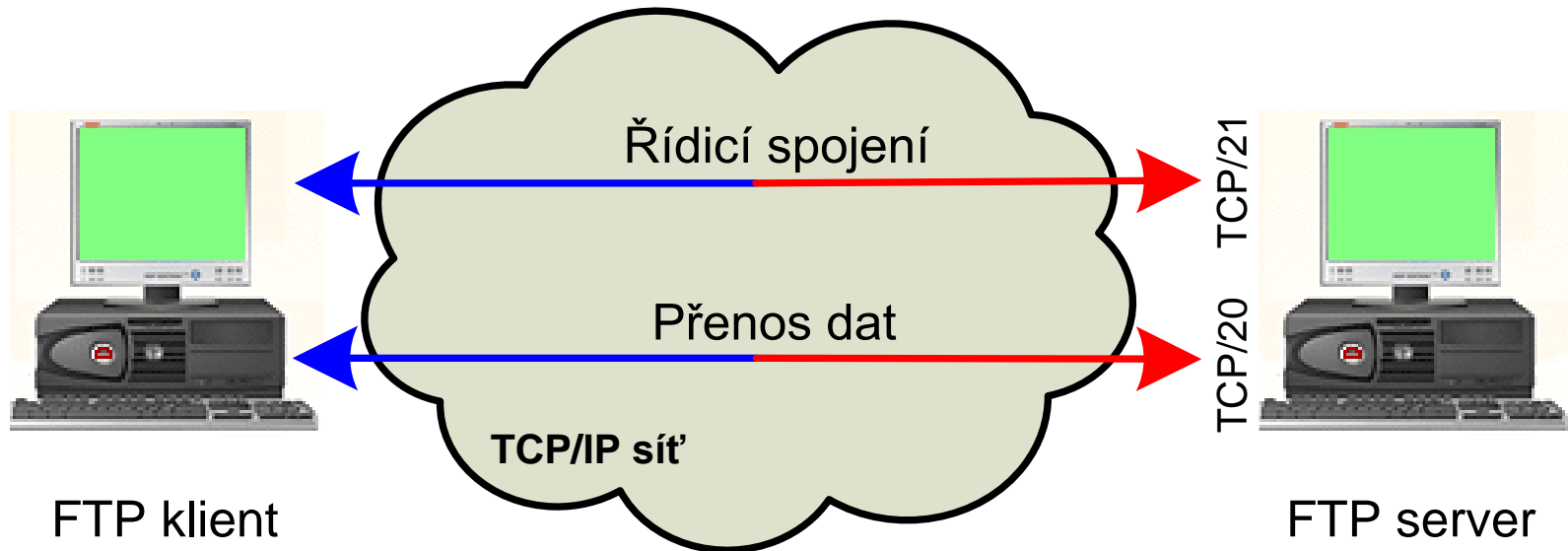
Základní schéma systému FTP

30



Nejčastější způsob komunikace v FTP

31



Výměna informací mezi klientem a serverem

32

- Komunikace klient -> server – *FTP commands*
- Syntaxe
 - ▣ PŘÍKAZ nebo PŘÍKAZ parametry
 - ▣ Příkaz je textový, délka 3 nebo 4 znaky, parametry i delší
 - ▣ Příklady
 - Kontrola přístupu
 - ▣ USER xjmeno00 (identifikace uživatele) – *username*
 - ▣ PASS heslo (následuje po USER, zadání hesla uživatele) – *password*
 - Přenosové parametry
 - ▣ PASV (přechod do pasivního režimu) – *passive*
 - Příkazy související s přenosem
 - ▣ RETR cesta_k_souboru (přenos souboru ze serveru) – *retrieve*

Výměna informací mezi klientem a serverem

33

- Komunikace server > klient – *FTP replies*
- Server odpovídá na příkazy klienta jednou nebo i více odpověďmi
 - ▣ Př: **příkaz:** „přenes soubor“, **odpověď:** „přenos zahájen“, [vlastní přenos], odpověď „přenos dokončen“
- Odpovědi -> tříčíselné výsledkové kódy
- První číslice odpovědi určuje zda se požadovaná akce zdařila nebo ne
- Druhá číslice odpovědi přibližuje víc druh odpovědi (chyby)
- Třetí číslice odpovědi doplňuje obsah zprávy
 - ▣ Př: kód 231 značí, že uživatel byl korektně odhlášen a služba ftp byla ukončena
 - ▣ Př: kód 425 značí, že nelze otevřít datové spojení pro přenos souboru

Výměna informací mezi klientem a serverem

34

První číslice	Zpráva
1	Požadovaná akce byla úspěšně započata, čekej další odpověď
2	Příkaz úspěšně proveden, může být zadán další
3	Příkaz byl přijat, server očekává další (používá se o u sekvence souvisejících příkazů, např. přihlášení na server)
4	Akce se nezdařila, ale existuje možnost pokusit se provést příkaz znovu (např. dočasná nedostupnost)
5	Příkaz nebyl proveden, např. není podporován, uživatel není oprávněn apod.

Pracovní režimy vzniku datového spojení

35

□ Aktivní režim

- Standardní režim (není potřeba nějak zapínat)
- Klient otevře náhodný port
- Naslouchá a očekává navázání spojení od serveru
- Server se připojí a probíhá přenos dat – **Server navazuje datové spojení**
- Problémy s NAT a firewally

□ Pasivní režim

- Nestandardní režim (musí předcházet domluva klienta a serveru)
- Server otevře náhodný port
- Naslouchá a vybídne klienta aby se připojil
- Klient se připojí a probíhá přenos dat – **Klient navazuje datové spojení**
- Nevadí NAT a firewally

WWW A PROTOKOL HTTP



Stručná historie a současnost WWW

37

- 1989 v Ženevě definován hypertextový systém pro CERN
- O rok později vznikl program pro tvorbu primitivních hypertextových stránek
- Systém běžící na jediném počítači nazván „**World-Wide Web**“
- 1992 na světě okolo padesáti webových serverů
- Postupně vznikají první grafické prohlížeče
- Ustaveno W3C (WWW Consortium), které má dohlížet a schvalovat standardy Webu
- 1995 skoro 100 000 webových serverů, nyní desítky miliónů
- Uživatelé vyžadují interaktivní stránky pro přístup k informacím
- Web byl původně vyvinut jako pomůcka pro sdílení výsledků vědeckého výzkumu po celém světě
- Dnes nová platforma, na které mohou běžet nejrůznější aplikace, které nejsou závislé na konkrétním operačním systému

Technologie kolem WWW

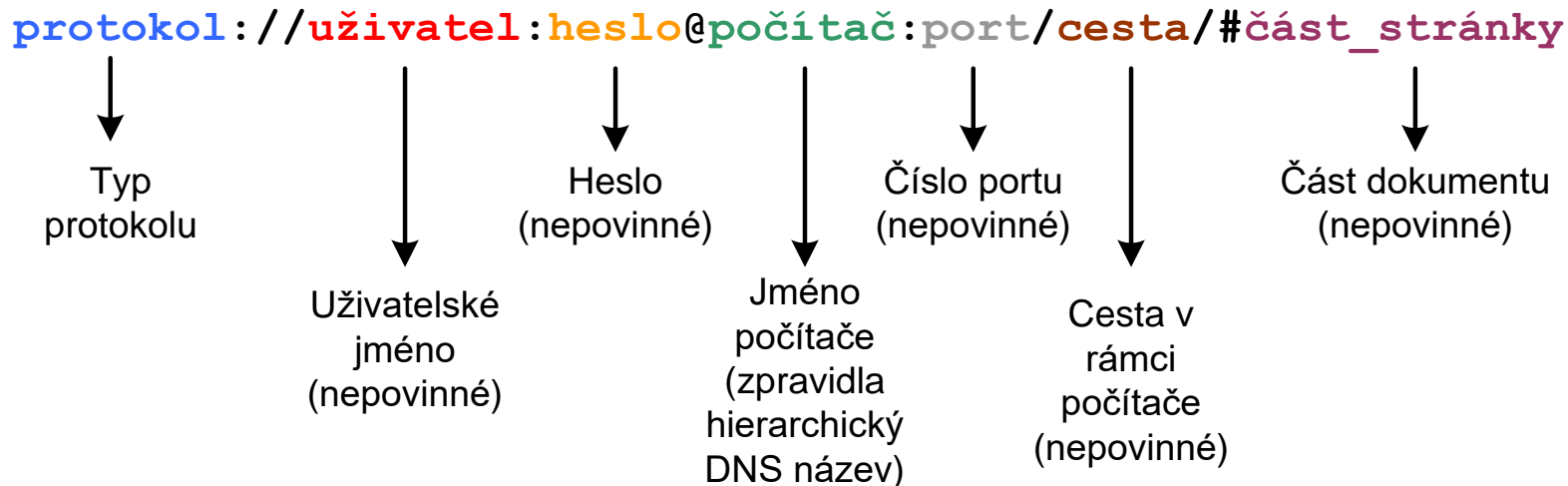
38

- **Jazyk HTML** (*HyperText Markup Language*)
 - ▣ Značkový jazyk sloužící k zápisu webových stránek
 - ▣ Všechny verze tohoto jazyka jsou zpětně kompatibilní
- **Protokol HTTP** (*HyperText Transfer Protocol*)
 - ▣ Zajišťuje přenos HTML stránek mezi WWW serverem a prohlížečem na straně uživatele
 - ▣ Dnes relevantní verze: HTTP/1.1, HTTP/2 a připravované HTTP/3, odlišné chování
- **URL** (*Uniform Resource Locator*)
 - ▣ Každý objekt na webu má jedinečnou adresu sloužící k jeho jednoznačné identifikaci
 - ▣ Umožňuje vytvoření odkazů mezi objekty webu
- Z dnešního pohledu spojení těchto tří technik již nabízí málo
 - ▣ prohlížení statických elektronických dokumentů propojených odkazy
 - ▣ požaduje se interaktivita a dynamika stránek (zpřístupnění měnících se informací)
 - ▣ požadavek zobrazení stránky vyvolá spuštění skriptu (kódu)

Uniform Resource Locator (URL)

39

- Řetězec má přesně definovanou strukturu, některé části jsou volitelné
- Vybrané položky spolu souvisí
- Adresa počítače vzniká hierarchicky podle umístění počítače v doméně (DNS)
- Položka port je nepovinná a prohlížeč její hodnotu doplní podle typu protokolu, (http 80, https 443)



Obecný popis protokolu HTTP

40

- Internetový ASCII orientovaný protokol
- Typ klient-server
- Původně určen pouze pro výměnu hypertextových dokumentů v HTML
- Komunikace mezi prohlížečem (klient) a www serverem
- Definuje tvar přenášených dat a formát dotazů a odpovědí
- Využívá port 80 na straně serveru (TCP, UDP, ...)
- Standard HTTP obsahuje definici číselných výsledkových kódů (odezvy na podnět klienta zasílané serverem)
- Rozšíření MIME (*Multipurpose Internet Mail Extension*) umožňuje přenos libovolného typu souboru
- Dále přes HTTP lze spouštět vzdálené aplikace – Webové služby
- Aplikační brány – zpřístupnění dalších protokolů – FTP, SMTP

Činnost protokolu HTTP

41

- Funguje způsobem dotaz – odpověď
- Uživatel pošle serveru dotaz ve formě čistého textu
 - ▣ Obsahuje označení požadovaného dokumentu
 - ▣ Informace o schopnostech prohlížeče apod.
- Server odpoví pomocí několika řádků textu popisujících výsledek dotazu
 - ▣ Zda se dokument podařilo najít, jakého typu dokument je, ...
- Následují data požadovaného dokumentu
- HTTP 1.0 a nižší
 - ▣ Každý objekt přenášen samostatně, bezestavově
 - ▣ Případně předchází navazování TCP spojení, následuje ukončení spojení
- V HTTP 1.1 přidána podpora perzistence spojení
 - ▣ Možnost přenášet více objektu v jednom spojení (sekvenčně)
 - ▣ Po určité době nečinnosti spojení ukončeno
 - ▣ Zrychlení komunikace klient-server
- HTTP/2 a HTTP/3 – novější verze protokolu s upravenými vlastnostmi
- Protokol často využíván i odlišným způsobem k přenosu mezi aplikacemi

Vybrané metody protokolu HTTP

42

□ Metoda

- Způsob, pomocí kterého se specifikuje požadavek klienta na serveru

Vybrané metody

□ **OPTIONS**

- Představuje dotaz na možnosti komunikace spojené s uvedeným URL

□ **GET**

- Požadavek na poslání dokumentu specifikovaného pomocí URL

□ **HEAD**

- požadavek na čtení záhlaví www stránky - doplňkových informací o dokumentu

□ **PUT**

- Požadavek na uložení posílaných dat pod specifikované URL na server
- Takto uložená data budou dostupná např. následnými dotazy GET

ELEKTRONICKÁ POŠTA A PROTOKOL SMTP



Schéma klasického přenosu e-pošty

44

- Klient – server model
- Obr.: Obě strany disponují vlastním poštovním serverem
- Klienti disponují programem MUA (*mail user agent*) - poštovní klient, schopný komunikovat se serverem

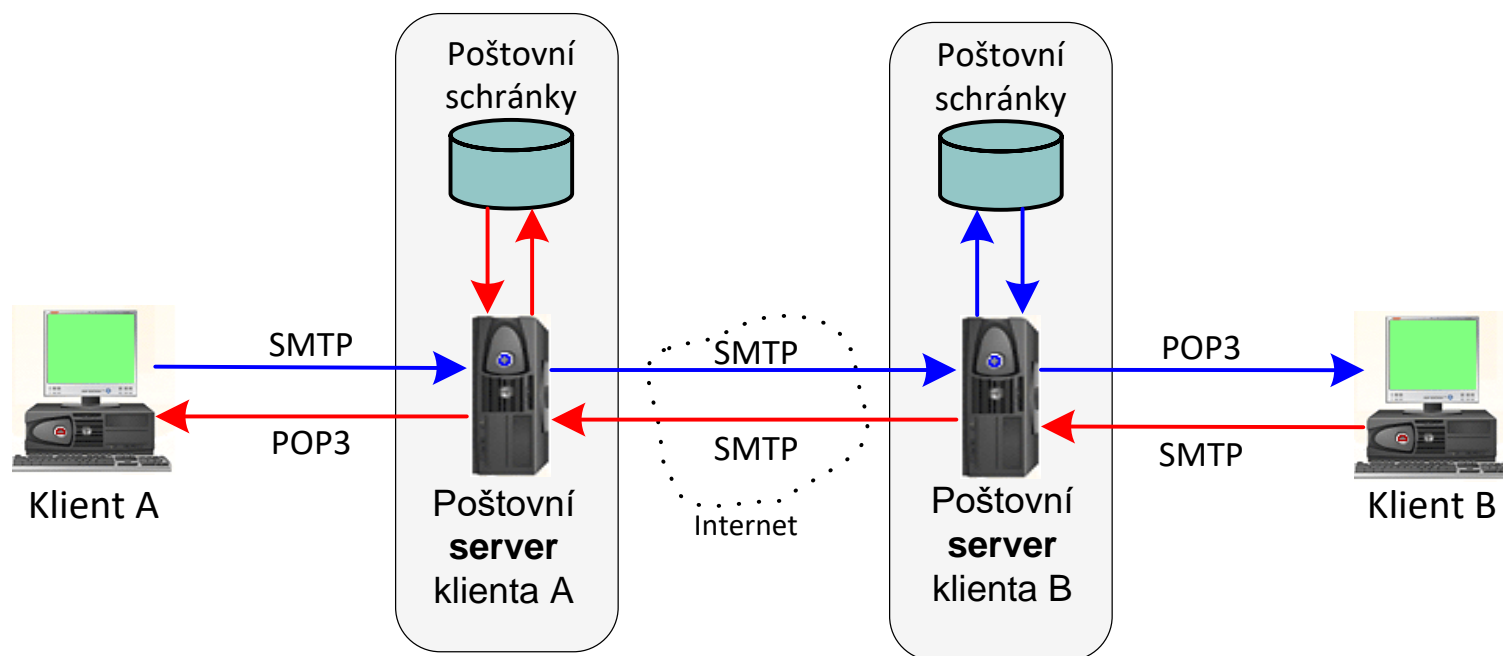


Schéma klasického přenosu e-pošty

45

- Klient A odešle zprávu na svůj poštovní server (Server A)
- Server A na základě cílové adresy odešle zprávu na server B
- Server B zprávu uloží do schránky uživatele
- Klient B si může po přihlášení na server B zprávu stáhnout
- Server má dvě části
 - MTA (*Mail Transport Agent*) – přenos zpráv mezi servery
 - MDA (*Mail Delivery Agent*) – lokální doručení zprávy na serveru do konkrétní schránky (dvousložková emailová adresa) nebo i přeposílání
- Dva základní druhy poštovních serverů – privátní a veřejné

Formát zprávy e-pošty

46

- Dvě základní části (obě textové)
 - ▣ **Záhlaví zprávy**
 - Struktura obsahující mnoho položek
 - Řídicí informace systému elektronické pošty
 - ▣ **Tělo zprávy**
 - Volná struktura, přenáší uživatelský vstup
 - Obsah interpretuje až poštovní klient (MUA)
 - Od záhlaví odděleno volným řádkem

Vybrané položky záhlaví emailu

47

Položka	Význam
Received	Toto záhlaví připisuje na počátek e-mailu každý emailový server, kterým zpráva projde (může jich být po cestě i více). Při čtení od spodu nahoru lze zjistit celou cestu, kterou zpráva prošla
From	Adresa odesílatele zprávy
Sender	kdo vyřizuje, např. sekretářka (nebo zde bývá i informace o konferenci, přes kterou zpráva přišla, pokud to tak je)
Reply-To	Odpověď zasílejte na (emailová adresa)
In-Reply-To	Odpověď na konkrétní zprávu
To	Adresát zprávy
Cc	Kopie zprávy (carbon copy)
Bcc	Utajená kopie, před odesláním (adresátovi v To či Cc) se toto záhlaví smaže (background carbon copy)
Subject	Krátká charakteristika zprávy (předmět)

Simple Mail Transfer Protocol (SMTP)

48

- Standard pro přenos elektronické pošty Internetem (RFC 2821 z roku 2001, původní standard RFC 821 – rok 1982)
- Využívá transportní protokol TCP (port 25 na straně serveru)
- SMTP **využíváno** na přenos od klienta k serveru **a mezi servery**
- Komunikace na principu klient-server
- Poštovní server musí obsahovat jak klientskou část, tak serverovskou
- SMTP je textový protokol (příkazy v US-ASCII) na aplikační úrovni
 - Klient zadává čtyřznakové příkazy
 - Server odpovídá stavovými kódy s textovým popisem (× HTTP, FTP)
 - Definované typy zpráv
 - Inicializace klienta (HELO, EHLO – verze SMTP)

Simple Mail Transfer Protocol (SMTP)

49

- Nyní rozšířená verze **ESMTP** (*Extended SMTP*)
 - ▣ umožňuje např. přenos potvrzování o doručení emailové zprávy
- **SMTP server – server odchozí pošty v MUA**
 - ▣ doručuje zprávy v zastoupení za uživatele
 - ▣ Uživatelé jsou často nuceni mít nastaven v rámci sítě konkrétní server SMTP, přístup na jiné je záměrně blokován (*spam*)

VoIP (Voice over IP)

50

- Obecně nejen přenos hlasu v reálném čase, ale multimediálních dat (zvuk, obraz), zaměříme se na hlas
- Celá řada standardizovaných protokolů aplikační vrstvy
 - ▣ **SIP** (Session Initiation Protocol)
 - ▣ **RTP** (Real-time Transport Protocol)
 - ▣ **RTCP** (RTP Control Protocol)
 - ▣ ...
 - ▣ Všechny využívají primárně jednoduché UDP, TCP nevhodné
- Infrastrukturní prvky (servery, brány)
 - ▣ Zprostředkování komunikace, konverze kodeků
 - ▣ Účtování hovorů
 - ▣ Ukládání údajů do databáze

Protokoly pro VoIP

51

□ SIP

- ▣ Kontrolní protokol, vytváření a ukončování relací
- ▣ Dohoda na kódování a způsobu komunikace (kodek)
- ▣ Řada zpráv (např. INVITE, 200 OK)

□ RTP

- ▣ Přenos multimediálních dat v reálném čase
- ▣ Paket obsahuje pořadové číslo a časovou značku
- ▣ 12 bajtů záhlaví, 160 bajtů dat (*chunk*) -> několik ms hlasu

□ RTCP

- ▣ Servisní kanál komunikace při RTP přenosu
- ▣ Řídí RTP přenos, umožňuje vyhodnocování parametrů, napomáhá dosažení požadované QoS (*Quality of Service*)
- ▣ Malý datový tok

