- V roce 1983 byly protokoly TCP/IP implementovány v operačním systému Berkeley (BSD) Unix.
- V témže roce byla síť ARPAnet rozdělena na MILNET a část ARPAnet. Termín Internet se používá pro veškerou síť (ARPAnet a MILNET).
- V roce 1985 vzniká nadace National Science Foundation (NSF) a začíná projekt sítě NSFNet propojující akademické a výzkumné organizace v USA k Internetu. Rychlost páteřních linek sítě je 56-kbit/s.
- V roce 1987 vytváří NSF páteřní síť Internetu s linkami T1 (1.5 Mbit/s) a definuje tří-vrstvovou topologii sítí (páteřní síť, regionální sítě, lokální sítě). Od roku 1991 tvoří páteřní spoje linky T3 (45 Mbit/s). V roce 1995 přestává být NSFNet primární páteřní sítí Internetu. Páteřní spoje jsou tvořeny komerčními subjekty např. internetMCI, PSI-Net, SprintLink či ANSNet.

1.2.2.2 Architektura TCP/IP

Architektura TCP/IP je výrazně jednodušší než referenční model OSI. Model TCP/IP spojuje služby prezentační a relační do jediné vrstvy, aplikační. Na úrovni fyzického přenosu bitů spojuje vrstvy fyzickou a linkovou do vrstvy fyzického rozhraní, která je obvykle implementovaná na síťové kartě. Srovnání obou modelů je na obrázku č. 1.5. Model TCP/IP výrazně zjednodušuje komplikovanou strukturu modelu OSI, který nebyl nikdy plně implementován a nasazen.

Vlastní implementace architektury TCP/IP je rozdělena do tří částí. Nejnižší část, vrstva fyzického rozhraní, je implementována v síťové kartě (NIC, Network Interface Card) a ovladači karty (driver). Vyšší vrstvy – internetová a transportní – jsou součástí síťových modulů operačních systémů. Jejich nastavení a ruční instalace je popsána v kapitole 1.4. Nejnižší vrstva síťového rozhraní se někdy nazývá též linková (viz např. RFC 1122 [26]), což je trošku zavádějící, neboť to koliduje se standardní definicí linkové vrstvy (datalink layer) podle standardu OSI. Někteří autoři, například J. F. Kurose [21] či A. S. Tanenbaum [32], používají pěti vrstvový model, kde rozdělují tuto nejnižší vrstvu na fyzickou a linkovou podle modelu OSI. Tento svůj model nazývají internetový model, resp. pěti vrstvový model TCP/IP. Jiní autoři (např. F. Halsall [11]) spojují fyzickou a linkovou vrstvu do jedné vrstvy, což odpovídá původnímu návrhu modelu sítě ARPAnet [22]. Tento model však používá jiné názvosloví pro tyto čtyři vrstvy: nejnižší vrstvu nazývá vrstvou síťového rozhraní (Network Interface), druhou vrstvu koncový uzel – koncový uzel (Host-Host), třetí vrstvu úroveň procesů (Process-Level) a nejvyšší aplikační (Application).

V této publikaci se budeme držet původního čtyřvrstvého návrhu vycházejícího ze standardu RFC 1122 s tím, že pro nejnižší vrstvu budeme používat v literatuře rozšířený název „síťové rozhraní“ (network interface nebo NIC, network interface card). Tento název lépe odpovídá implementaci, neboť služby této vrstvy jsou obvykle implementovány přímo na síťové kartě počítače a jejím ovladači.

Příklad aplikace modelu TCP/IP pro službu WWW (protokol HTTP) lze graficky znázornit na obrázku č. 1.7. Podobně i pro další služby je možné sestavit obdobný model, který pro jednotlivé vrstvy modelu definuje používaný protokol. Protokol nejnižší vrstvy závisí na zvolené přenosové technologii (Ethernet, bezdrát, optika) a pro danou službu je transparentní. Fyzický přenos dat je realizován pomocí datových jednotek PDU, o kterých jsme se zmínili v předchozí kapitole.

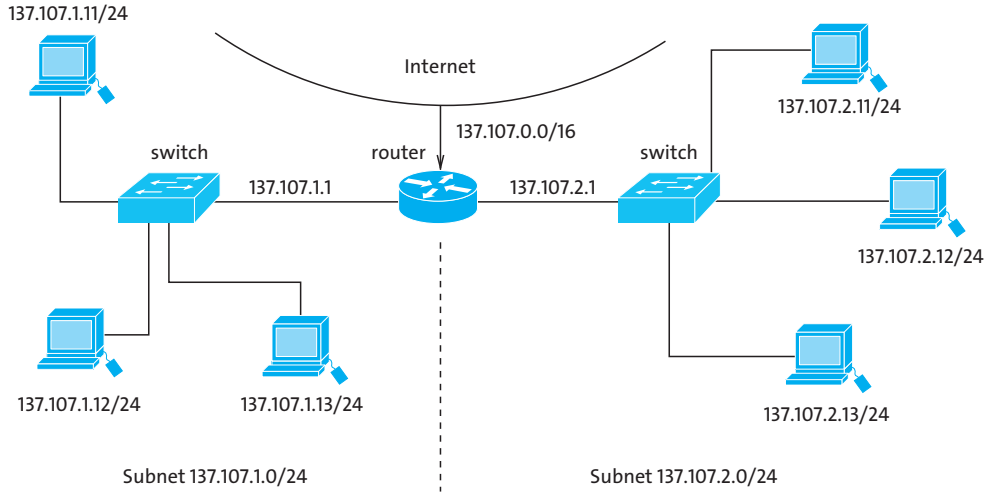
osmi bitů pro vytváření podsítí, by nulová podsíť byla 137.107.0.0/24 a jedničková podsíť 137.107.255.0/24. Standard RFC 950 doporučuje, že tyto adresy, tj. nulová adresa, která znamená „tato síť“, a adresa jedničková, která znamená „všechny“, nemají být použity pro fyzické podsítě.

Technicky lze takové sítě vytvořit a použít, i když ne všechna zařízení a výrobci tyto speciální adresy podporují. Doporučení RFC 1878 [25] toto umožňuje, nicméně se nejedná o standard. Hlavním důvodem, proč se tomu vyhnout je, že může dojít k záměně s jinými adresami.

Například při použití nulové podsítě 137.107.0.0/24 má počítač 137.107.0.10 v této podsíti stejnou adresu sítě jako například počítač 137.107.5.12 v původní síti, tj. 137.107.0.0. Výsledek se liší pouze délkou masky. Při použití jedničkové podsítě pak například broadcastová adresa poslední podsítě, tedy podsítě 137.107.255.0/24, je 137.107.255.255, což je stejná adresa jako broadcast pro celou adresu 137.107.0.0/16. Proto budeme v našich příkladech uvažovat pouze podsítě s minimálně dvěma bity a bez možnosti vytvářet jedničkovou či nulovou podsíť.

Maximálně si můžeme vzít `hostid` – 2 bitů, protože vždy potřebuje alespoň dva bity na adresu počítačů v podsíti. Například pro adresy sítě 137.107.0.0 (třída B) lze použít prvních osm bitů části `hostid` (tzn. třetího bytu adresy) na adresování podsítě. Nové podsítě budou mít adresu 137.107.1.0/24, 137.107.2.0/24 apod. Těchto podsítí lze vytvořit až 254 (2^8-2). V každé podsíti může být 254 počítačů (obr. 1.17). Pokud bychom potřebovali více než 254 počítačů v podsíti, vypůjčíme si z části `hostid` méně bitů, např. tři. Se třemi bity můžeme vytvořit šest podsítí a v každé podsíti mít až 8 190 počítačů. Adresa jedné z podsítí by mohla být např. 137.107.32.0/19. Principy vytváření podsítí v Internetu jsou popsány ve standardu RFC 950 [18].

Při použití této metody se nemění délka adresy sítě `netid`, což je důležité pro směrování. Podle této adresy určí směrovače, kam se datagram přepošle. Poslední směrovač na cestě předá datagram do lokální sítě, kde se doručí cílové stanici. Metoda vytváření podsítí zachovává třídy adres i směrování podle tříd, pouze `netid` se už nevztahuje k jedné síti LAN, ale spíše k lokalitě (např. firma, univerzita apod.). Tato lokalita obvykle obsahuje více podsítí LAN vytvořených pomocí techniky subnetting.



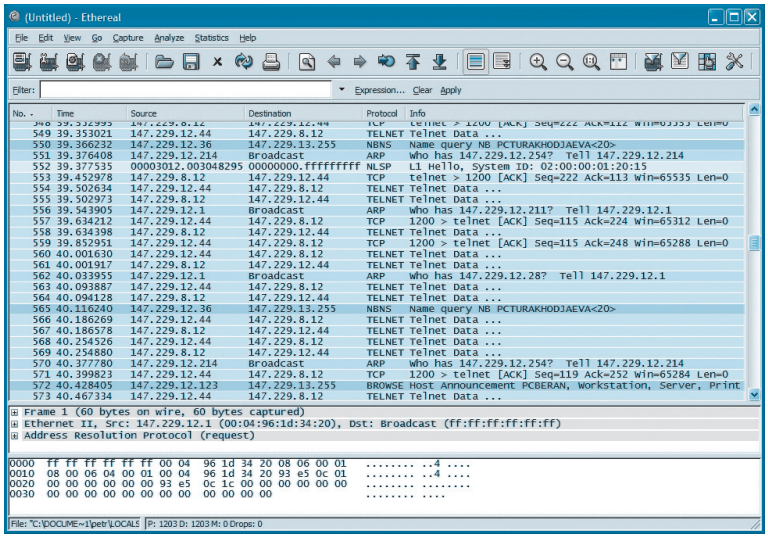
Obrázek 1.17: Vytvoření podsítě pro adresy třídy B

Sledování toku dat na síťovém rozhraní

Sledování toku a rozbor dat představuje náročnější způsob analýzy. Na druhou stranu nám tento přístup nabízí ucelený a jednoznačný pohled na aktuálně probíhající komunikaci. Při analýze vidíme obsah příchozích i odchozích paketů. Protože se jedná o citlivé aktivity, může tyto prostředky pro sledování používat pouze uživatel s právy administrátora systému. V prostředí unixových operačních systému se používá program `tcpdump`, který sleduje přenos na zadaném rozhraní.

```
tcpdump -s0 -X -i fxp0
  13:53:02.136977 IP pcmatousek.fit.vutbr.cz.62047 > kazi.fit.vutbr.cz.domain:
   8123+PTR? 254.12.229.147.in-addr.arpa. (45)
     0x0000: 4500 0049 6233 0000 4011 dc3f 93e5 0c5b E..Ib3..@..?...[
     0x0010: 93e5 080c f25f 0035 0035 23ea 1fbb 0100 .....5.5#.....
     0x0020: 0001 0000 0000 0000 0332 3534 0231 3203 .........254.12.
     0x0030: 3232 3903 3134 3707 696e 2d61 6464 7204 229.147.in-addr.
13:53:04.458742 IP pckolar.fit.vutbr.cz.ipp>147.229.13.255.ipp: UDP,length
     0x0000: 4500 00cb dec6 4000 4011 1935 93e5 0c5d E.....@.@..5...]
     0x0010: 93e5 0dff 0277 0277 00b7 e27a 3330 3136 .....w.w...z3016
     0x0020: 2033 2069 7070 3a2f 2f70 636b 6f6c 6172 .3.ipp://pckolar
     0x0030: 3a36 3331 2f70 7269 6e74 6572 732f 4850 :631/printers/HP
     0x0040: 3433 3030 5053 2022 4c61 7365 724a 6574 4300PS."LaserJet
```

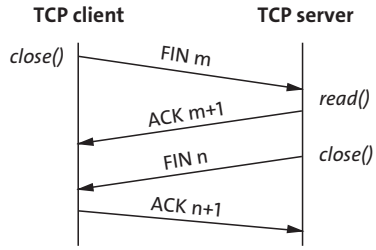
V prostředí Windows i Unix je velmi oblíbený grafický protokolový analyzátor Wireshark (dříve Ethereal), obr. 1.23, případně programy Network Monitor, Network Diagnostik a další.



Obrázek 1.23: Protokolový analyzátor Wireshark

Blíže si o způsobu analýzy dat povíme v kapitole 10 Správa sítě.

je dvě minuty [6]. Výpočet hodnoty závisí na více parametrech a může se pohybovat od jedné do čtyř minut podle typu spojení. Korektní ukončení TCP spojení zahrnuje výměnu čtyř paketů TCP s příznaky FIN m, ACK m+1,FIN n, ACK n+1, viz obrázek 2.9.



Obrázek 2.9: Korektní ukoncení spojení TCP

`close()` – uzavření schránky

```
int close(int sockfd)
```

Funkce `close()` uzavře schránku a ukončí spojení TCP. Ukončení spojení může trvat delší dobu, protože na cestě mohou být ještě nějaká data. Teprve po standardní uzavírací proceduře (sekvenci čtyř paketů TCP s výše uvedenými příznaky) je spojení skutečně ukončeno. Pokud počet referencí na schránku je větší než nula, spojení se fyzicky neuzavře, pouze se sníží počet referencí.

```
#include <unistd.h>
close(s);
```

Server uzavírá spojení funkcí také funkcí close(). Poslední paket s příznakem FIN se předá aplikaci jako znak EOF (End Of File), po němž funkce read() vrátí hodnotu 0.

`shutdown()` – okamžité uzavření schránky

```
int shutdown(int sockfd, int howto)
```

Narozdíl od funkce `close()` iniciuje funkce `shutdown()` normální uzavření spojení TCP (příkazem FIN) bez ohledu na počet referencí na schránku. Funkce `close()` uzavírá obě strany spojení TCP, které je plně duplexní (tj. souběžně probíhá čtení i zápis v rámci jednoho spojení). Funkce `shutdown()` dovoluje uzavřít pouze jednu stranu komunikace. Závisí to na parametru `howto`, který může obsahovat následující hodnoty:

SHUT_RD: Uzavření schránky pro čtení. Žádná další data nemohou být přijata do schránky.
SHUT_WR: Uzavření schránky pro zápis. Data určená k poslání jsou odeslána spolu s ukončovací sekvencí (FIN).
SHUT_RDWR: Uzavření schránky pro čtení i zápis.

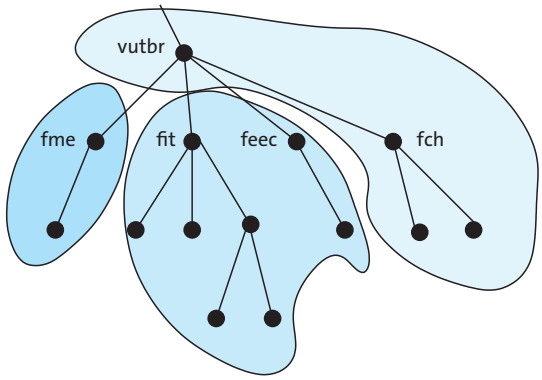
aplikace. Při práci s DNS, jako je například konfigurace serveru DNS či klienta DNS, je ukončující tečka nezbytná.

Databáze DNS obsahuje jednotlivá doménová jména uspořádaná do podstromů (domén). Správa domén (přidávání koncových uzlů, rušení uzlů, vytváření subdomén), je decentralizovaná a deleguje se na další organizace. Například doménová jména počítačů `pc1.fit.vutbr.cz`, `pc2.fit.vutbr.cz`, `pc3.fit.vutbr.cz`, `www.fit.vutbr.cz`, `ns.fit.vutbr.cz` atd., patří do domény `fit.vutbr.cz.`, kterou spravuje Fakulta informačních technologií VUT v Brně na svém lokálním serveru DNS `kazi.fit.vutbr.cz`.

Uspořádání prostoru doménových adres

Prozatím jsme se bavili o uspořádání prostoru doménových jmen, který má strukturu hierarchického stromu. Tento strom však není uložen na jednom místě v jedné databázi. DNS je systém hierarchický a decentralizovaný. Jednotlivé části podstromu celého prostoru doménových adres jsou fyzicky uloženy na lokálních serverech DNS, které dohromady tvoří systém DNS. Data o objektech v prostoru DNS (obecně zdrojích, resources) netvoří pouze doménová jména. Záznamy obsahují také informace o primárních a sekundárních serverech DNS, správcích domén, poštovních serverech a podobně. Veškeré tyto informace jsou zapisovány v textovém formátu, který definují RFC 1034 [27] a RFC 1035 [28]. Ukládají se do takzvaných *záznamů DNS* (RR, resource records). Přehled typů záznamů DNS a podrobnější popis jejich formátu je uveden v kapitole 3.3.

Fyzické části prostoru DNS, které jsou pod jednou správou, se nazývají *zóny*. Zóna není totožná s doménou. Například poskytovatel síťového připojení může spravovat více domén (a subdomén) tvořící jednu zónu. Nebo naopak velká doména (např. `edu`) je rozdělena na více částí a umístěna na více serverech DNS. Například doména `vutbr.cz` může být rozdělena do více zón, které pokrývají různé části doménového prostoru a které spravují různé subjekty, viz obrázek 3.2.



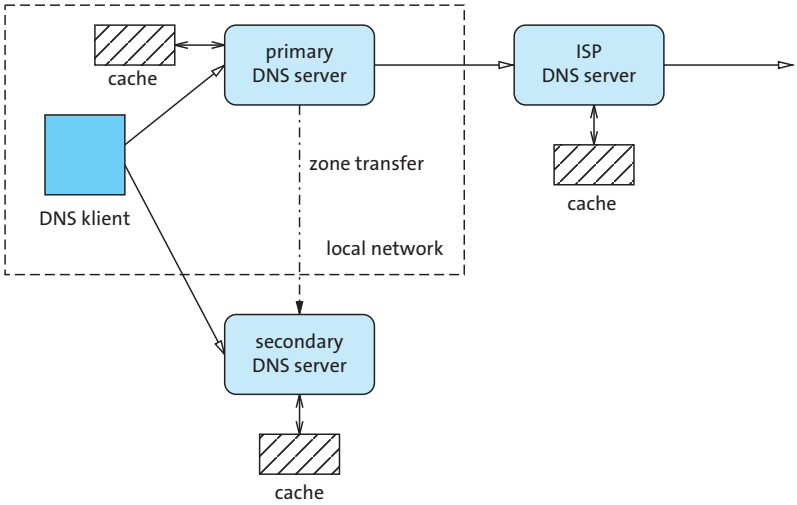
Obrázek 3.2: Příklad uspořádání DNS prostoru do zón

Zatímco doména je část prostoru adres, který má společný suffix (např. doména vutbr.cz), zóna je tvořena částmi prostoru uloženými na konkrétním serveru. Zónu mohou tvořit například subdomény fit.vutbr.cz a feec.vutbr.cz pod jednou správou. Zóna může obsahovat celou doménu nebo jen část domény. Existují také dva speciální typy zón:

- **Sekundární server DNS** (slave, secondary nameserver)
  Sekundární server získává data od primárního serveru. Soubor, který obsahuje databázi konkrétní domény (či subdomény), se nazývá *zónový soubor*. Proces přenosu zónových souborů z primárního serveru na sekundární se nazývá *přenos zón* (zone transfer). Přenos zón je podrobněji popsán v kapitole týkající se komunikace DNS. Sekundární server musí zajistit pravidelný přenos zónových dat a aktuálnost dat. Sekundární server je také autoritativní server pro danou doménu.

- **Záložní server DNS** (caching-only nameserver)
  Záložní server pracuje jako proxy server. Přijímá dotazy od klientů a přeposílá je dalším serverům DNS. Když záložní server dostane odpověď na svůj dotaz, uchová si ji a použije ji v budoucnosti. Záložní servery poskytují neautoritativní odpovědi, tj. odpovědi, které mohou být neúplné a neaktuální. Zrychlují však proces rezoluce doménového jména.

Zóna DNS je souvislá část jmenného prostoru DNS, pro kterou je daný server DNS autoritativní. Server může obsahovat více zón. Každá zóna obsahuje záznamy DNS pro danou část jmenného prostoru.

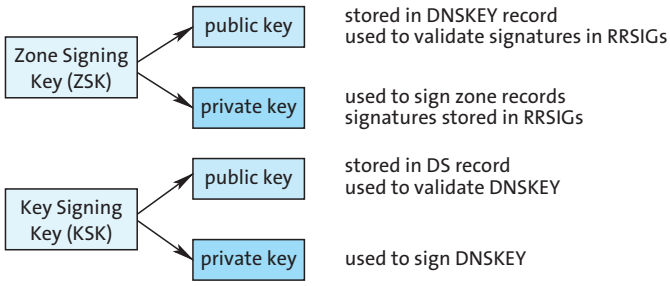


Obrázek 3.4: Typy DNS serverů

Platnost záznamů DNS na sekundárním a záložním serveru je časově omezena. Hodnota expirace je uvedena u každého záznamu. Pokud dojde k expiraci záznamu, musí server daný záznam smazat ze své databáze, případně si ho znovu načíst z primárního serveru. Pokud by neprovedl aktualizaci, může dojít k situaci, kdy dva různé servery DNS odpoví na stejný dotaz různým způsobem, což vede k nekonzistenci dat.

Zejména při změně v záznamech DNS (změna IP adresy, přidání nového záznamu apod.) je potřeba počítat s tím, že rozšíření změn trvá v řádu hodin, což je doba, kdy dojde k expiraci původních záznamů a načtení nových. Na druhou stranu by nastavení krátké doby expirace vedlo k častým dotazům na obnovení záznamu a přetížení autoritativních serverů. Protože změny se šíří v síti DNS pomalu, je nutné v případě, kdy potřebujeme zjistit aktuální hodnotu nějakého záznamu v DNS, kontaktovat přímo primární nebo sekundární server DNS.

Z tohoto důvodu nám nestačí jenom klíč pro podpis zóny ZSK (Zone Signing Key). Je potřeba ještě klíč pro ověření těchto klíčů, takzvaný KSK (Key Signing Key). Pro podepisování klíčů opět použijeme asymetrickou kryptografii. Máme tedy dva páry klíčů ZSK a KSK, které se použijí pro vybudování důvěry mezi servery DNS. Jejich vztah a uložení v DNS ukazuje obrázek 3.17.

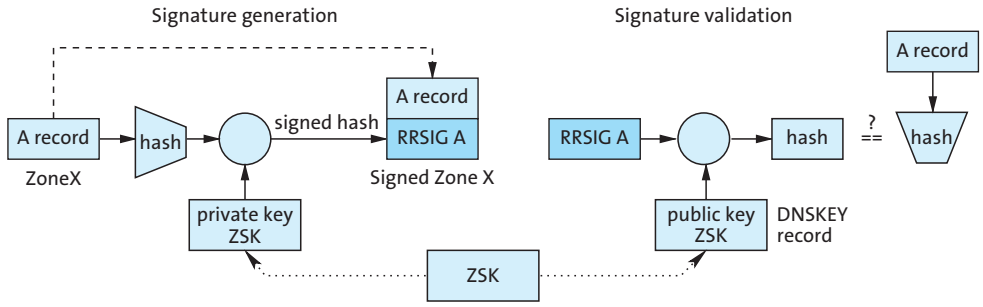


Obrázek 3.17: Klíč pro podpis zóny (ZSK) a klíč pro podepisování klíčů (KSK)

Tyto dva páry klíčů, ZSK a KSK, tvoří základ systému zabezpečení DNSSEC. Používají se k podepisování a validaci zón a k podpisování a validaci klíčů pro podpis zón. Klíče KSK vytváří (přesněji řečeno veřejný klíč KSK) vytváří tzv. *důvěryhodný vstupní bod SEP (Security Entry Point)*, viz [17]. Společně vytváří propojení KSK a ZSK tzv. *řetězec důvěry (chain of trust)*.

3.5.4 Řetězec důvěry

Podepsaná zóna obsahuje veřejný klíč DNS (uložený v záznamu DNSKEY), podpisy záznamů (uložené v záznamech RRSIG), odkazy na další záznamy (záznamy NSEC), případně záznam DS ověřující klíč zóny v DNSKEY. Pokud zóna neobsahuje tyto záznamy, jedná se o nepodepsanou zónu. Příklad podepsání a ověření záznamu v DNS je na obrázku 3.18.



Obrázek 3.18: Podepsání záznamu typu A a ověření podpisu

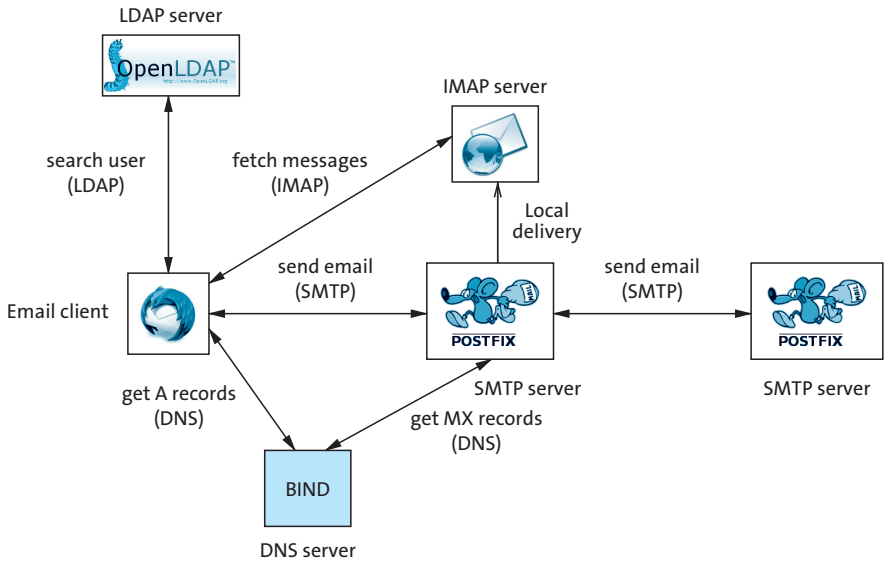
Jak jsme si již řekli, záznam typu DNSKEY obsahuje veřejný klíč pro podepisování záznamů v dané zóně. Tento klíč se používá k ověření podpisu záznamů uložených v RRSIG. Podpisy se připojují k záznamům typu A, CNAME, MX, NS a podobně v dané

4.1.1 Architektura elektronické pošty

Původní emailový systém přenášel jednoduché textové zprávy, kde na prvním řádku byla adresa příjemce. Přenos probíhal pomocí kopírování souborů službou FTP. Později byl návrh rozšířen. V roce 1982 byl definován protokol SMTP (Simple Mail Transfer Protocol) pro přenos emailových zpráv [1, 3] a standard pro tvorbu emailů [1, 3]. V roce 1984 organizace CCITT vytvořila standard X.400 pro elektronickou poštu. Tento komplexní a robustní systém se neujal a dnes se již téměř nepoužívá.

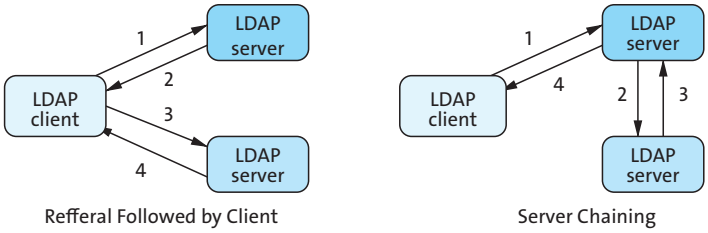
Základní architekturu systému elektronické pošty tvoří dvě entity – *uživatelský agent UA* (User Agent) a *agent pro přenos zpráv MTA* (Message Transfer Agent). Uživatelský agent je klientská aplikace, která slouží k vytváření, odesílání a čtení elektronické pošty. Agent pro přenos zpráv předává zprávu mezi jednotlivými uzly až na místo doručení. Agentem pro přenos zpráv může být poštovní server (mail server), např. sendmail, postfix, MS Exchange.

Systém elektronické pošty nepracuje pouze s přenosovým protokolem SMTP. Tvoří ho přístupové protokoly pro čtení doručených zpráv ze schránek (IMAP a POP3). Pro správné doručení pošty jsou také nezbytné služby DNS, konkrétně záznamy typu MX. Tyto záznamy k dané doméně (např. fit.vutbr.cz) přiřazují poštovní servery, které pro tuto doménu přijímají elektronickou poštu. Pokud tyto záznamy chybí, odmítne SMTP server poštu doručit a vrátí ji adresátovi. Podrobnosti o záznamech MX lze najít v části 3.3.



Obrázek 4.1: Systém elektronické pošty a další služby

Systém elektronické pošty rozšířený o další potřebné služby je zobrazen na obrázku 4.1. Na obrázku je zobrazen emailový klient, který slouží k vytváření a odesílání elektronické pošty. Při vytváření pošty lze pro vyhledání adresy příjemce použít buď lokální adresář, který bývá součástí emailového klienta, nebo adresářovou službu LDAP. Ta umí podle vlastního jména osoby vyhledat emailovou adresu příslušného uživatele. Více o adresářových službách se dočtete v kapitole 5.



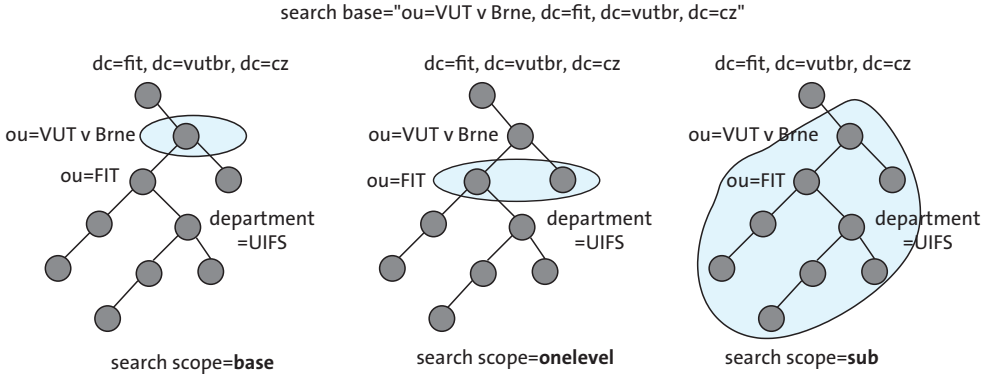
Obrázek 5.10: Zpracování odkazů typu referral v LDAP

Zpracování odkazů na straně klienta může být buď (1) automatické – klient po obdržení odpovědi typu referral snaží spojit s odkazovaným serverem a získat potřebnou informaci – nebo (2) manuální, kdy klient vrátí uživateli odpověď s odkazem a uživatel musí sám iniciovat spojení s dalším serverem.

5.2.3 Funkční model

Funkční model popisuje operace, které lze provádět nad adresářem, zejména dotazy (prohledávání adresáře), změny dat (přidávání, aktualizace, rušení záznamu) a řízení přístupu k datům (identifikace klienta).

Funkční model popisuje jednak příkazy protokolu LDAP [3], dále pak způsob a rozsahy vyhledávání informací v adresářovém stromě. Při prohledávání zejména rozsáhlých databází je vhodné omezit podstrom vyhledávacích informací. LDAP definuje tři typy úrovně vyhledávání v adresáři: vyhledávání pouze v zadaném bázovém objektu (base), vyhledávání pouze v bezprostředních následnících zadaného objektu (one-level) nebo vyhledávání v celém podstromu adresáře (subtree), viz obr. 5.11.



Obrázek 5.11: Rozsahy vyhledávání v adresáři LDAP

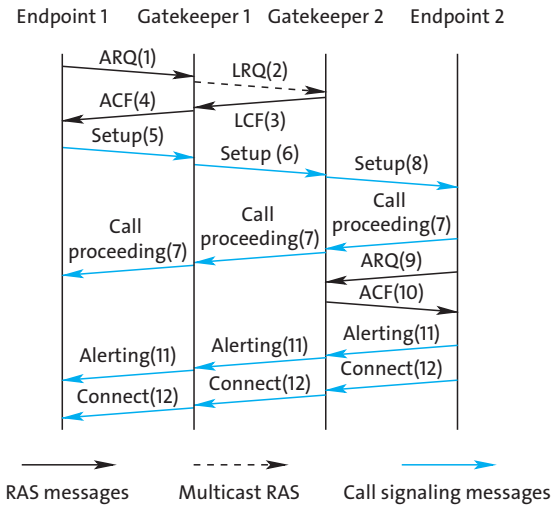
Podle typu dat lze při vyhledávání zvolit různé způsoby porovnání dat. Ne pro všechny typy dat jsou definovány veškeré typy porovnání. Pokud je nad daným typem definováno řazení (např. jména, čísla, apod.), lze použít při porovnání relace typu menší, větší, apod., případně vyhledávání v řetězci na rovnost, podřetězec či přibližnou shodu.

Vytvoření spojení prostřednictvím ústředny (gatekeeper)

Před vytvářením spojení mezi koncovými uzly prostřednictvím ústředny dochází předtím k registraci koncových bodů na ústředně. Proto, aby se mohl koncový bod registrovat, musí znát IP adresu ústředny. Ta může být buď manuálně nastavená v konfiguraci koncové stanice, nebo může koncový uzel vyslat multicastový příkaz GRQ (Gatekeeper Request) na adrese 224.0.1.41. Pokud koncová stanice obdrží odpověď GCF (Gatekeeper Confirmation), použije zaslanou IP adresu. Vlastní registrace na ústředně probíhá pomocí komunikace RAS (obr. 7.10). Pro identifikaci může stanice použít své ID a IP adresu. Poté následuje vytvoření spojení:

1. Vysílající stanice pošle žádost ARQ (Admission Request) na ústřednu.
2. Ústředna žádost potvrdí (ARC, Admission Confirmation) a pošle IP adresu volané stanice.
3. Vysílající stanice začne vytvářet spojení s volanou stanicí (příkaz Setup protokolu Q.931).
4. Volaná stanice se zaregistruje na ústředně (ARQ přes RAS) a zkontroluje svá práva.
5. Pokud proběhne úspěšně registrace volané stanice na ústředně (ACF), volaná stanice odpoví na žádost o vytvoření spojení přes Q.931.
6. Pomocí kanálu H.245 se vymění logické parametry spojení a otevře se transportní kanál RTP mezi koncovými stanicemi.
7. Přes RTP dochází k výměně multimediálních dat.

Příklad vytvoření spojení mezi koncovými zařízeními H.323 prostřednictvím ústředny je zakreslen na obrázku 7.12.



Obrázek 7.12: Vytvoření spojení pomocí signalizačního protokolu H.323

Základní komponenty systému H.323

Základní prvky systému H.323 tvoří terminál, brána (gateway), ústředna (gatekeeper) a jednotka MCU: