VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

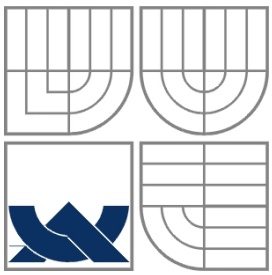
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

MOŽNOSTI NASAZENÍ PROTOKOLU IPV6 VE FIREMNÍ LAN SÍTI

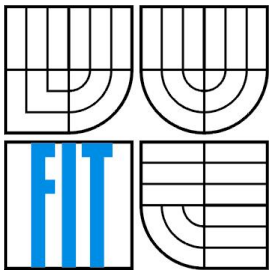
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE MARTIN KŠICA
AUTHOR

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

MOŽNOSTI NASAZENÍ PROTOKOLU IPV6 VE FIREMNÍ LAN SÍTI

HOW TO DEPLOY IPV6 PROTOCOL IN A LAN NETWORK

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MARTIN KŠICA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PETR MATOUŠEK, Ph.D.

BRNO 2008

Abstrakt

Cílem práce bylo vytvořit ucelený přehled o stavu a možnostech nasazení IPv6 protokolu ve firemní LAN síti. Zmapoval jsem možnosti a implementace nejpoužívanějších operačních systémů MS Windows XP, Linux, FreeBSD, směrovačů Cisco, služeb DNS a DHCPv6. Součástí práce je návrh a realizace způsobu rozdělení IPv6 adres pro lokální sít s připojením IPv6 sítě k internetu. Práce zahrnuje průzkum trhu ISP v ČR, schopnost komunikace v novém IPv6 protokolu.

Klíčová slova

Konfigurace IPv6, ICMPv6, objevování sousedů, bezstavová automatická konfigurace, DHCPv6, 6to4.

Abstract

The goal of this thesis is to make the overview in status and possibilities to use IPv6 protocol in companies LAN networks. I maped possibilities to implement most used operational systems MS Windows XP, Linux, FreeBSD, Cisco routers, DNS and DHCPv6 services. The part of this thesis is design and realization way of IPv6 address distribution for local network with connection of IPv6 network to Internet. The thesis includes the ISP market research at CR, further ability and interest to communicate in new IPv6 protocol.

Keywords

Configuration of IPv6, ICMPv6, Neighbor Discovery, Stateless Address Autoconfiguration, DHCPv6, 6to4.

Citace

Kšica Martin: Možnosti nasazení protokolu IPv6 ve firemní LAN síti. Brno, 2008, bakalářská práce, FIT VUT v Brně.

Možnosti nasazení protokolu IPv6 ve firemní LAN síti

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Petra Matouška
Další informace mi poskytl Ing. Tomáš Kašpárek
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

……
Jméno Příjmení
Datum

Poděkování

Děkuji za trpělivost a pomocnou ruku svému vedoucímu práce Ing. Petru Matouškovi a konzultantovi Ing. Tomáši Kašpárkovi

Obsah

1 Úvod

Internetový protokol verze 6 je novým síťovým protokolem, který by měl zcela nahradit stávající protokol IPv4, který je pro dnešní síťové nároky nedostačující. Nedostačující jak z pohledu velkých nároků na množství veřejných IP adres, tak samotného návrhu, který je téměř dvacet let starý a představuje tak určitá omezení, od kterých by se měl nový protokol oprostit. Nízká rozšířenost nového protokolu by se měla prolomit v polovině roku 2011, kdy se předpokládá již neúnosný počet volných IPv4 adres. Do hry tak naplno vstoupí nový protokol s novými možnostmi ve světě sítí.

Tato práce si klade za cíl prozkoumat a popsat možnosti nových vlastností IPv6 protokolu. Vlastnosti, jako obrovský adresový prostor, mechanismus objevování sousedů, bezstavová automatická konfigurace, IP bezpečnost a mobilita, přinášejí nové možnosti do světa síťové komunikace. Hlavním cílem práce je prozkoumat možnosti a nasazení nového protokolu ve firemních LAN sítích. Součástí této studie je průzkum a testování současných implementací IPv6 protokolu operačních systémů a služeb. Práce pojednává o způsobu konfigurace počítačů a aktivních prvků sítě tak, aby byly schopné IPv6 komunikace. Dále vytváří případovou studii návrhu připojení IPv6 sítě k internetu. Navíc obsahuje průzkum trhu ISP v ČR, schopnost komunikace v novém IPv6 protokolu.

První kapitola práce nabízí základní přehled o stěžejních vlastnostech nového protokolu. Za ní následuje kapitola popisující adresaci v novém protokolu. Jejím cílem je podat čtenáři základní pohled na nový protokol a seznámit jej s novým formátem IP adres, zápisem a typy IPv6 adres.

Ve třetí kapitole se seznámíme s protokolem *ICMPv6* (Internet Control Message Protocol Version 6), s jeho funkcí a formátem. Součástí jsou příklady a popis některých ICMPv6 typů zpráv.

Čtvrtá kapitola se věnuje sadě nových mechanismů, která byla označena pod název *objevování sousedů* (Neighbor Discovery). Tento obecný nástroj má na starost např. zjišťování linkových adres počítačů, hledání směrovačů, přesměrování, ověření dosažitelnosti sousedů a další.

Pátá kapitola popisuje automatickou konfiguraci, která spadá pod již zmíněný mechanismus objevování sousedů. Naleznete zde seznámení s novou možností *bezstavové automatické konfigurace* (Stateless Address Autoconfiguration), která bude hrát velkou roli u mobilních síťových technologií.

Následující šestá kapitola se věnuje přechodovým mechanismům nabízející možnost přechodu počítačů a směrovačů k novému protokolu. Naleznete zde výčet nejčastějších řešení přechodu, jako dvojí zásobník, tunelování a translátory.

Sedmá kapitola otevírá praktickou část samotné práce, ve které naleznete možnosti použití IPv6 protokolu v operačních systémech a HW směrovačích společnosti Cisco. Součástí této kapitoly jsou praktické ukázky konfigurace s příklady jednotlivých vlastností nového protokolu. Každá podkapitola systémů obsahuje způsob instalace IPv6 protokolu, konfigurace směrování, nastavení síťových rozhraní a popis zabezpečení počítače.

Osmá kapitola obsahuje návrh rozdělení přiděleného prefixu do podsítí. Cílem je popsat způsob, jakým lze přidělený prefix rozdělit a vytvářet hierarchickou strukturu podsítí.

Kapitola devět obsahuje realizaci bezstavové automatické konfigurace počítačů sítě. Součástí této kapitoly je ukázka konfigurace směrovače a počítače.

Kapitola deset a jedenáct je vyhrazena pro služby DNS a DHCPv6. Zde naleznete příklad použití konkrétních implementací služeb zajišťujících funkcionalitu postavenou pro IPv6 protokol. Každá kapitola obsahuje příklady s nastavením pro realizaci služby.

Dvanáctá kapitola nabízí návrh způsobu připojení lokální sítě k IPv6 internetu. Návrh je postaven na tunelování IPv6 datagramů přes IPv4 internetovou infrastrukturu využitím mechanismu 6to4. Připojení je realizováno softwarovým směrovačem systému FreeBSD. Součástí je schéma sítě s příklady směrovací tabulky, konfigurace síťových rozhranní a prověření IPv6 připojení směrovače.

Předposlední třináctá kapitola popisuje možnosti a způsob realizace IPv6 konektivity ISP v ČR.

Poslední kapitola shrnuje zaměření a dosažené výsledky této práce. Její součástí je výčet možných rozšíření, která se do této práce nedostala.

1.1 Klíčové vlastnosti nového protokolu

Následující výčet shrnuje výhody IPv6 protokolu:

- Větší adresový prostor - IPv4 používá pouze 32-bitový adresový prostor, který nabízí 4 miliardy adres v rámci internetu. Na první pohled se jedná o obrovský rozsah. Nicméně se díky masivnímu rozšíření internetu rozsah volných adres natolik zmenšil, že se v polovině roku 2011 (viz [23]) počítá s nedostatkem adres. IPv6 používá 128-bitový rozsah, kterým nabízí téměř $3{,}4 \times 10^{38}$ jedinečně identifikovatelných adres v rámci internetu. Obrovský adresový prostor umožňuje globální identifikaci počítačů všech sítí světa. Nepočítá se tak s technologiemi typu NAT, které slouží k překladu adres a budování privátních sítí.
- Zlepšení pro nasazení nových technologií:
  - automatická konfigurace - IPv4 nabízí nepovinný DHCP protokol zajištující automatickou konfiguraci počítačů v síti. Problém nastává v situaci připojení k síti, kde takový server zajištující tuto službu není dostupný. IPv6 nabízí bezstavovou konfiguraci počítačů sítě založené na ohlašování informací o síti směrovačem.
  - bezpečnost - U IPv4 se bezpečnost příliš neřešila, proto je IPsec (IP security) nepovinnou výbavou protokolu. Podpora IPsec je u IPv6 povinnou výbavou protokolu. Diky tomu tak můžeme využívat zabezpečenou komunikaci kdykoli si přejeme komunikovat se zařízením podporující IPv6. Více informací o IPsec viz [20].
  - multicast - Oproti IPv4 je u IPv6 multicast povinnou vlastností.
- přechodové mechanismy - jedná se o mechanismy vytvářející dostupnou IPv6 konektivitu.
- a další

2 Adresace

Hlavní změna nového protokolu se týká délky adresy. Délka IPv6 adresy je 128 bitů, která nabízí rozsah $3{,}4 \times 10^{38}$ globálně identifikovatelných adres.

Novinkou u IPv6 adres je existence tzv. prefixu. IPv6 adresa se typicky skládá ze síťového prefixu a identifikátoru, který se obvykle generuje z MAC adresy počítače. Tento prefix určuje typ adresy a zároveň skupinu do které adresa spadá.

Další zajímavostí u IPv6 rozhraní je nutná existence alespoň jedné lokální linkové adresy, která se vytváří z MAC adresy rozhraní a prefixu pro lokální linkové adresy. Slouží pro komunikaci na lokálním segmentu sítě.

2.1 Formát a zápis nových adres

Zápis IPv6 adres umožňuje reprezentaci IPv6 adres třemi způsoby:

1. První a zároveň preferovaný formát má podobu `x:x:x:x:x:x:x:x` kde ‘`x`‘ reprezentuje hodnotu čtyř hexadecimálních čísel.
   Příklad: `ABCD:EF01:2345:6789:ABCD:EF01:2345:6789`
2. Druhý způsob zápisu uvažuje delší řetězec nulových bitů adresy. Tento řetězec se zredukuje na řetězec ‘`::`‘, který představuje jednu či více skupin šestnácti nulových bitů. Tento řetězec lze v adrese použít pouze jednou a to kdekoli v IPv6 adrese.
   Příklad redukce adresy: `FF01:0:0:0:0:0:0:101` na adresu `FF01::101`
3. Třetí alternativa představuje zápis tzv. IPv4 kompatibilních IPv6 adres. Jedná se o adresu přiřazenou zařízení, které pracuje jak s IPv4 tak IPv6 adresami. Tyto adresy mají následující tvar: `x:x:x:x:x:x:d.d.d.d`, kde ‘`x`‘ představuje opět hodnotu čtyř hexadecimálních čísel adresy, a ‘`d`‘ dekadická osmibitová čísla jak je tomu u zápisu IPv4 adres.
   Příklad: `::192.168.1.1`

S posledním typem adres se lze setkat např. u tunelování 6to4, kde se IPv4 kompatibilní adresy používají při směrování síťového provozu.

2.2 Typy adres

U klasické rodiny protokolu IPv4 máme unicastové, broadcastové a multicastové typy adres. Protokol IPv6 používá adresy unicastové, anycastové a multicastové. U protokolu IPv6 je funkce broadcastových adres zastoupena adresami multicastovými.

2.2.1 Individuální adresy (Unicast Addresses)

Individuální adresy jsou stejné jako u protokolu IPv4. Jedna individuální adresa identifikuje jedno síťové rozhraní. Máme čtyři typy individuálních adres.

Globální individuální adresy (Global Unicast Addresses): Tento typ adres je velice důležitý. Nositel této adresy je identifikovatelný v rámci celého internetu. Adresy jsou v IPv6 internetu globálně směrovatelné a dostupné, jsou ekvivalentní veřejným IPv4 adresám. Globálním adresám byl přiřazen prefix, začínající binárně 001.

Lokální linkové adresy (Link Local IPv6 Unicast Addresses): Lokální linkové adresy začínají prefixem FE80::/10. Používají je uzly ke komunikaci se sousedy na stejné lince bez směrovačů. Každý počítač, či směrovač si svoji lokální linkovou adresu přidělí sám na základě identifikátoru rozhraní. Každé rozhraní má svoji lokální linkovou adresu. Využívá se u mechanismu objevování sousedů.

Místní lokální adresy (Site Local IPv6 Unicast Addresses): Místní adresy začínají prefixem FEC0::/10. Jsou ekvivalentní k IPv4 privátnímu adresovému rozsahu (10.0.0.0/8, 172.16.0.0/12 a 192.168.0.0/16). Tyto adresy se používají v privátním intranetu, do kterého není z internetu přístup. Místní lokální adresy nejsou přístupné z jiné sítě. Více informací o místních lokálních adresách lze nalézt v RFC 4291 [10].

Lokální smyčka (The Loopback Address): Následující adresa ::1 je adresou lokální smyčky, kterou může počítač komunikovat sám se sebou. Jedná se o ekvivalentní adresu k adrese 127.0.0.1 u IPv4.

2.2.2 Skupinové adresy (Multicast Addresses)

Skupinové adresy identifikují sérii rozhraní (patřící typicky různým uzlům). Paket zaslaný na skupinovou adresu je doručen všem ze série rozhraní identifikované takovouto adresou.
Tabulka předdefinovaných multicastových IPv6 adres:

Adresa	Popis významu jednotlivých typů multicastových adres
FFx0::/16 a FF0F::/16	Jedná se o rezervovaný rozsah adres.
FFx1::/16	Rozsah daného rozhraní, jedná se o ekvivalentní rozsah k adrese loopback.
FFx2::/16	Lokální linkový rozsah, rozsah na lokálním segmentu sítě.
FFx4::/16	Malý rozsah, který je nutno nastavit.
FFx5::/16	Místní lokální rozsah, omezen po hranici místní sítě.
FFx8::/16	Rozsah organizace, omezen po sítě dané organizace.
FFxE::/16	Globální rozsah skupinových adres

Více informací o multicastových typech adres lze nalézt v RFC 2375 [26].

2.2.3 Výběrové adresy (Anycast Addresses)

Výběrové adresy identifikují sérii rozhraní (patřící typicky různým uzlům). Paket zaslaný na výběrovou adresu je doručen jednomu ze série adres identifikované touto adresou. Rozhodování, na kterou adresu je paket zaslán, se děje na základě hodnoty parametru metrika ze směrovacích tabulek směrovačů. Použití anycastových adres má význam např. u mechanismu 6to4. Více informací o anycastovém prefixu pro mechanismus 6to4 lze nalézt v RFC 3068 [27].

3 ICMPv6

ICMPv6 (Internet Control Message Protocol Version 6), jehož návrh vychází z dokumentu RFC 4443 [9], je víceúčelový protokol se stejnou funkcí jako ICMP protokol u IPv4. Jeho funkcí je ohlašování chyb, diagnostika přenosu paketů, hledání dalších uzlů sítě, či ohlašování skupinových informací o dané síti. Zastupuje protokoly jako ICMP, IGMP a ARP z protokolu IPv4. Protokol ICMPv6 je nedílnou součástí každé IPv6 implementace, na které stojí např. celý mechanismus objevování sousedů. ICMPv6 zprávy jsou identifikovány hodnotou 58 a jsou přenášeny uvnitř IPv6 datagramů v jejich rozšířené hlavičce.

Formát zpráv se skládá z typu, kódu, kontrolního součtu a těla zprávy. Typ identifikuje typ dané zprávy. Pokud je první bit typu hodnota nula, jedná se o chybovou zprávu, a naopak pokud je typu hodnota jedna, jedná se o zprávu informační. Číslo třídy (kód) určuje typ zprávy. Kontrolní součet slouží k odhalení chyb při přenosu samotné ICMPv6 zprávy.

8 bitů	8 bitů	16 bitů
Typ	Kód	Kontrolní součet
Tělo zprávy		

Příklad ICMPv6 chybových zpráv:

Cíl nedosažitelný (Destination unreachable) typ 1	Tento typ zprávy informuje o nedosažitelném cíli.
Příliš velký paket (Packet Too Big) typ 2	Tato zpráva musí být odeslána směrovačem jako odpověď na paket, který nemohl být předán z důvodu jeho příliš velké velikosti. Tato velikost se identifikuje pod pojmem MTU maximální velikost paketu.
Čas překročen (Time Exceeded) typ 3	Pokud směrovač přijme paket s nulovou hodnotou parametru Hop Limit, nebo směrovač dekrementuje Hop Limit na hodnotu nula, musí paket zahodit a zaslat tento typ zprávy původci zahozeného paketu.

Příklad ICMPv6 informačních zpráv:

Požadavek na odezvu (Echo Request) typ 128	Tato zpráva se využívá k ověřování dostupnosti síťových zařízení.
Odezva (Echo Reply) typ 129	Odpověď na požadavek o odezvu.

Novým typem ICMPv6 zpráv jsou zprávy využívané mechanismem objevování sousedů.

Výzva směrovači (Router Solicitation) typ 133	Tento typ zprávy slouží k výzvě směrovači o zaslání ICMPv6 zprávy typu ohlášení směrovače.
Ohlášení směrovače (Router Advertisement) typ 134	Tímto typem zprávy směrovače periodicky ohlašují svoji přítomnost v síti nebo reagují na výzvu směrovače nějakého počítače. Obsahem této zprávy jsou informace o dané síti, jako adresové prefixy, či hodnota parametru Hop Limit.
Výzva sousedovy (Neighbor Solicitation) typ 135	Slouží k zjišťování linkových adres sousedů, nebo dostupnosti daného souseda využívající tabulku linkových adres sousedů.
Ohlášení souseda (Neighbor Advertisement) typ 136	Touto zprávou soused reaguje na zprávu výzvu souseda. Používá se k oznamování změn linkových adres, či detekce dostupnosti vyzývaného souseda.
Přesměrování (Redirect) typ 137	Využívaný směrovači k informování o počítače o lepší cestě k danému cíli.

Více informací o těchto zprávách lze nalézt v RFC 4861 [7].

4 Objevování sousedů

Objevování sousedů je sada zpráv a procesů, zjišťující vztahy mezi sousedními uzly na stejné lince. Principy objevování sousedů vychází z dokumentu RFC 4861 [7], popisující formát a funkci jednotlivých zpráv a jejich procesů. Obecným problémem IPv4 je absence protokolu, či mechanismu který zajišťuje detekci nedostupnosti sousedních uzlů. Mechanismus objevování sousedů nahrazuje mechanismy z IPv4 protokolu jako ARP, ICMP objevování sousedů a ICMP přesměrování. Nabízí mnohá zlepšení, která jsou u IPv4 protokolu nedostupná.

Obecně se objevování sousedů využívá pro:

- rozeznávání linkových adres sousedních uzlů
- zjišťování změn linkových adres sousedních uzlů
- zjišťování dostupnosti sousedů
- přesměrování

Počítače a směrovače jej využívají pro:

- objevování sousedních směrovačů
- objevování adres, adresových prefixů a dalších konfiguračních parametrů

Směrovače jej využívají pro:

- oznamování své přítomnosti v síti, konfiguračních parametrů a prefixů

Příklad IPv6 procesů objevování sousedů:

Objevování směrovačů (Router Discovery)	Proces, kterým počítače a směrovače zjišťují přítomnost směrovačů na lokální síti.
Objevování prefixů sítě (Prefix Discovery)	Proces, kterým počítače zjišťují síťové prefixy pro lokální podsítě.
Objevování parametrů (Parameter Discovery)	Proces zjišťování dodatečných parametrů sítě, jako MTU, výchozí hodnota parametru Hop Limit pro odchozí pakety.
Automatická konfigurace adres (Address Autoconfiguration)	Proces automatické konfigurace adres síťových rozhraní. Popisuje způsob a princip stavové (stateful) a bezstavové (stateless) automatické konfigurace adres počítačů.
Rozpoznávání adres (Address resolution)	Proces, kterým počítače zjišťují překlady IPv6 adres sousedů na linkové adresy. Nahrazuje tak protokol ARP u IPv4 protokolu.
Detekce duplicitních adres (Duplicate Address Detection)	Proces ověřování jedinečnosti adresy v dané síti.
Detekce dostupnosti (Neighbor Unreachability Detection)	Proces neustálého sledování stavu dosažitelnosti sousedů, se kterými daný počítač komunikuje.

5 Automatická konfigurace

Jednou z důležitých vlastností IPv6 je automatická konfigurace. Tento mechanismus zajišťuje automatické přidělování IP adres všem prvkům v síti požadující adresu. Na rozdíl od klasického IPv4 přináší nová verze vedle stavové (Stateful Address Autoconfiguration) také tzv. bezstavovou automatickou konfiguraci (Stateless Address Autoconfiguration), zajišťující přidělování adres, aniž je nutno na nově připojeném počítači cokoli ručně konfigurovat.

5.1 Bezstavová automatická konfigurace

Nová verze protokolu IPv6 přináší nový bezstavový mechanismus automatické konfigurace IP adres počítačů dané sítě (RFC 4862 viz [8]). Základem této automatické konfigurace je směrovač udržující informace o síti. Samotné ohlašování informací směrovačem probíhá v podobě periodického rozesílání ICMPv6 zpráv typu ohlášení směrovače (Router Advertisement) na skupinovou adresu určenou pro všechny uzly v rámci téže linky.

Jiným způsobem získání těchto informací je odeslání žádosti (Router Solicitation) o bezstavovou konfiguraci. Ta probíhá v následujících krocích:

1. Na základě lokální linkové adresy žadatel vyšle žádost směrovači o informace dané sítě.
2. Pokud tuto žádost směrovač přijme, zašle ohlášení směrovače, ve kterém žadateli sdělí např. prefixy IP adres, které se v síti používají, dobu platnosti implicitního směrovače, maximální hodnotu parametru Hop Limit odesílaných paketů a použití stavové či bezstavové konfigurace.
3. Z těchto informací a linkové adresy se vytvoří další individuální adresa, identifikující počítač buď v dané síti, či internetu. Počítač si poté ověří jedinečnost takové adresy (Duplicate Address Detection) pokusem kontaktovat vytvořenou adresu a zjistit její existenci v síti.
4. Pokud se neprokáže duplikát takové adresy, je adresa prohlášena za platnou. Následně může plně komunikovat v IPv6 síti.

Hlavní výhodou tohoto mechanismu je opuštění od klasického stavového mechanismu DHCP serveru, zajišťující alternativní přístup k automatické konfiguraci IP adres a vyžadující tak server, který se stará o samotné přidělování jednotlivých adres. Nevýhodou bezstavového mechanismu je absence šíření informací o dostupných DNS serverech. Tento nedostatek doplňuje dokument RFC 4339 [13] popisující možnosti konfigurace DNS serverů hostitelských počítačů, ve kterém je definována doplňující volba pro šíření DNS informací.

5.2 Stavová automatická konfigurace (DHCPv6)

DHCPv6 (Dynamic Host Configuration Protocol for IPv6) je protokol využívaný k stavovému přidělování IPv6 adres. Jeho další funkcí je distribuce informací, které není možné jinak zjistit z dané sítě, jako DNS servery či NIS servery. DHCPv6, je protipólem k bezstavové konfiguraci. Servery využívají rezervovanou multicastovou adresu FF05::1:3 a FF02::1:2.

Hlavní rozdíly mezi DHCPv4 a DHCPv6:

- na rozdíl od DHCPv4, je přidělování IPv6 adres řízeno parametry dané zprávy
- jsou odebrány zprávy typu DHCPDISCOVERY a DHCPOFFER. Místo nich je k dispozici zpráva SOLICIT (žádost) pro klienty a ADVERTISE (ohlášení) pro servery
- u DHCPv6 má klient možnost požádat o více IPv6 adres

Výhodou DHCPv6 oproti bezstavové konfiguraci je centralizované řízení přidělených IPv6 adres, dynamická aktualizace DNS záznamů a vyšší zabezpečení.

Další změnou je použití tzv. DUID (DHCP Unique Identifier) identifikátoru, který slouží serveru k identifikaci klientů, vyhledávajících konfigurační parametry sítě a současně k asociaci s IA (Identity Association), což je soubor adres přiřazených klientovi. Hlavní požadavky DUID jsou:

- jedinečnost v rámci všech klientů a serverů
- neměl by se v průběhu času měnit

Více informací o DHCPv6 lze nalézt v RFC 3315.

6 Přechodové mechanismy

Důležitým problémem, který bylo nutné vyřešit, je samotný přechod z protokolu IPv4 na nový protokol IPv6. Tento problém vyplývá ze samotné infrastruktury internetu, postavené na stávajícím protokolu IPv4. Rozšíření IPv6 infrastruktury závisí na ukončení přidělování IPv4 adres a spuštění globálního přidělování IPv6 adres ISP. Je evidentní, že samotný přechod bude po několik let znamenat současnou koexistenci obou protokolů v internetu. Do doby, než se začne globálně používat nový protokol, musel vzniknout návrh, který zajistil průchod IPv6 paketů přes IPv4 síťovou infrastrukturu. Pro tuto koexistenci vzniklo několik návrhů vycházejících z RFC 4213 [6].

Dvojí zásobník (Dual Stack)	V tomto režimu pracuje směrovač s oběma protokoly. Směrovač je schopen komunikovat s oběma světy, a to v závislosti na schopnostech protějšího konce.
Tunelování (Tunneling)	Principem je průchod IPv6 paketů přes IPv4 infrastrukturu. Tento způsob komunikace využívá zapouzdření IPv6 paketu do IPv4 paketu, který se označí speciálním typem tunelovaného paketu. Ten se následně odešle přes IPv4 infrastrukturu na druhý konec tunelu. Příjemce paket rozbalí a na základě své směrovací tabulky paket doručí.
Translátory (Translation)	Translátory našly své využití již u IPv4 ve formě NATu. Cílem NATu je překlad zdrojových, cílových adres a vytváření privátních sítí. Důsledkem je navíc zpomalení zabírání volných veřejných IPv4 adres. Translátory mají u IPv6 funkci překladu adres mezi oběma protokoly. Jejich cílem je umožnit komunikaci mezi zařízeními, jestliže každý podporuje jiný typ protokolu. Příkladem je počítač podporující pouze IPv6, který žádá data od počítače podporující pouze IPv4 protokol. V těchto příkladech tunelované spojení nepomůže a jsme odkázáni na translátory.

7 Praktické nasazení nového protokolu

7.1 Microsoft Windows XP

Prvním operačním systémem řady Windows, který byl schopen komunikovat s protokolem IPv6, byl OS řady Windows XP, jak ve verzi Professional, tak ve verzi Home Edition. Avšak samotná implementace byla označována za vývojářskou a v žádném případě se nedoporučovalo ji používat ke komerčním účelům.

Vývoj této implementace pokračoval i po uvedení systému na trh. Protokol se oficiální podpory dočkal až s aktualizací Service Pack 1. Díky tomu se implementace dostala do stavu použitelného pro širší veřejnost.

Nedostatkem je absence stavového mechanismu automatické konfigurace (DHCPv6 klient) hostitelských počítačů ze sady procesů Objevování sousedů. Tento nedostatek do jisté míry kompenzuje dostupná bezstavová konfigurace, která by měla hrát klíčovou roli u nového protokolu.

V tomto dokumentu uvažujeme Windows XP s nainstalovanou aktualizací Service Pack 2, který obsahuje další rozšíření jako dostupnost IPv6 Firewallu, filtrující nevyžádaný příchozí provoz a přechodový mechanismus Teredo. Principem mechanismu Teredo jsou globální internetové síťové servery Teredo, nabízející na požádání unicastovou adresu, přes kterou hostitelský počítač vstupuje do IPv6 internetu. Výhodou tohoto mechanismu je jeho schopnost komunikace s počítači oddělené jedním, či více NAT mechanismy. Jeho nevýhodou jsou neefektivní nároky na přeposílání dat.

Kvalitní nápověda systému popisuje nejen samotnou instalaci, konfiguraci protokolu, ale také sadu nástrojů, bez kterých by další ladění systému nebylo možné.

7.1.1 Instalace protokolu

Po klasické instalaci systému je podpora pro IPv6 vypnuta. Z důvodu chybějícího konfiguračního GUI rozhraní IPv6 se konfiguruje pouze přes příkazový řádek.

1. Otevřete okno příkazového řádku
2. Do příkazového řádku zadejte: `ipv6 install`

Po instalaci si všechna síťová rozhraní vytvoří z linkových adres povinnou lokální linkovou adresu, kterou využívají ke komunikaci na lokální lince. Pokud se v síti nachází nějaký směrovač rozesílající ohlášení směrovače, systém si z těchto informací vytvoří další adresu, včetně nastavení implicitního směrování na tento směrovač.

Veškerá další konfigurace se děje přes nástroj `netsh` pokrývající dostatečnou množinu nastavení, od zobrazování informací o síťových rozhraních, po zobrazení tabulky sousedů dané sítě.

Zajímavostí je automatická konfigurace 6to4 tunelu a použití implicitního směrovače na adrese 6to4.ipv6.microsoft.com. Nutností je zde veřejná IPv4 adresa, ze které se na základě mechanismu 6to4 vytvoří IPv6 adresa, poskytující připojení počítače do světa IPv6 internetu. To vše automaticky ihned po instalaci samotného protokolu.

7.1.2 Konfigurace síťových rozhraní

Adresy síťových rozhraní lze konfigurovat pouze z příkazové řádky systému, např. nástrojem `netsh`. Nastavení probíhá v přepnutí do kontextu konfigurace IPv6 rozhraní.

```
netsh interface ipv6
```

V tomto kontextu lze konfigurovat, či zobrazit nastavení IPv6 adres rozhraní, směrování, obecné parametry globální konfigurace, stav služby Teredo a další.

Konfigurace IPv6 adresy rozhraní vypadá následovně:

```
netsh interface ipv6 set address "internet" 2001::1
```

Kde řetězec `“internet“` zastupuje název daného rozhraní, kterému chceme přiřadit novou adresu `2001::1`. Dodatečnými parametry lze specifikovat např. persistentní nastavení, typ adresy apod.

7.1.3 Konfigurace DNS serveru

Vzhledem k absenci stavové konfigurace a absenci konfigurace dostupných DNS serverů u bezstavové konfigurace systému, je nutno přistoupit k manuální konfiguraci dostupných DNS serverů, poskytujících překlad IP adres na doménová jména. Navíc jsou všechny dotazy na DNS adresy a příslušné odpovědi posílány prostřednictvím IPv4 protokolu. Systém Windows XP je tak zcela závislý na protokolu IPv4, bez kterého není schopen získat jakýkoli překlad doménového jména. I přes tuto skutečnost nabízí systém možnost konfigurace IPv6 adres DNS serverů.

Následující příkaz ukazuje konfiguraci dostupných DNS serverů, poskytujících překlad IP adres na doménová jména.

```
netsh interface ipv6 add dns název_připojeného_rozhraní IPv6_adresa
```

příklad:

```
netsh interface ipv6 add dns “internet“ fec0:0:0:ffff::1
```

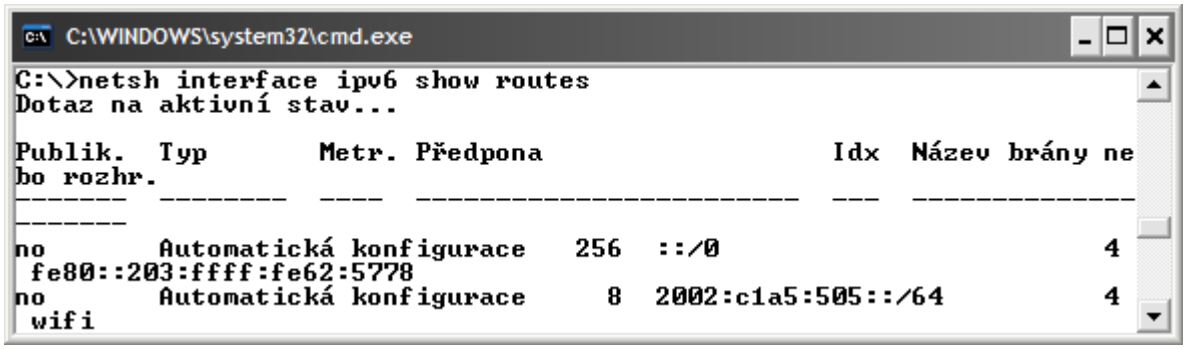
Užitečným nástrojem je mapování IPv6 adres na názvy počítačů. Toto mapování se nastavuje v souboru `C:\Windows\system32\drivers\etc\hosts`.

7.1.4 Konfigurace směrování

Další důležitou vlastností systému je možnost konfigurace směrovací tabulky, díky které lze směrovat síťovou komunikaci na síťových rozhraních. Systém v tomto směru nabízí aplikaci, zajišťující dostatečnou konfiguraci nebo pouhou kontrolu nastavení.

Možností konfigurace je několik, ale zůstaneme u novějšího nástroje `netsh` a starší nástroj `ipv6` necháme být.

Výpis směrovací tabulky:



Vytvoření statického směrování:

```
netsh interface ipv6 add route route_na ifindex route_přes
```

Příklad použití:

```
netsh interface ipv6 set route 3ffe::/16 "internet" fe80::1
```

Pokud se v síti nachází směrovač rozesílající ohlášení směrovače, nemusíte provádět následující nastavení, jelikož si systém nastaví směrovač jako výchozí bránu pro IPv6 síťový provoz směřující mimo lokální síť.

Nastavení výchozí brány směrování:

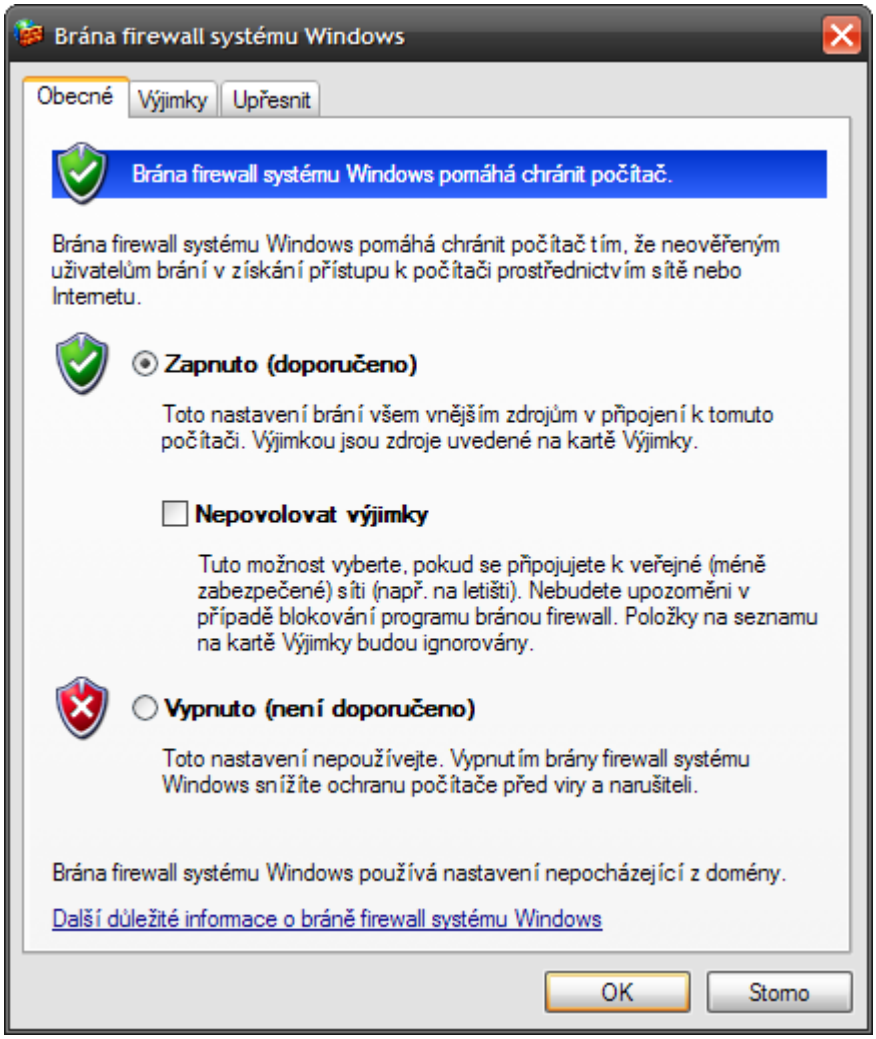
```
netsh interface ipv6 add route ::/0 ifindex ipv6_adresa_brány
```

Výchozí bránu zde zastupuje adresa `::/0` , která směruje veškerý provoz na nastavené IPv6 rozhraní. Vhodnějším nastavení by byla adresa `2000::/3`, která se chová jako implicitní brána pro všechny globální individuální adresy světa IPv6 internetu. Možností je zde opravdu mnoho.

7.1.5 Bezpečnost

Bezpečnost zajišťuje Brána Windows Firewall, jehož nastavení lze provádět z grafického rozhraní. Otevření panelu Firewallu provedeme spuštěním panelu Brána Windows Firewall v Ovládacích panelech systému. Je důležité zmínit společné nastavení Firewallu, jak pro protokol IPv4, tak IPv6, které činí nastavení jaksi nepružné pro oddělený provoz obou protokolů.

Grafické rozhraní pro správu paketového filtru:



Správu Firewallu lze provádět z příkazového řádku v kontextu `firewall` nástroje `netsh`.

Příklad, který zakáže veškerý ICMP/ICMPv6 příchozí síťový provoz:

```
netsh firewall set icmpsetting all disable
```

Více možností příkazů Firewallu zobrazíme příkazem:

```
netsh firewall help
```

7.1.6 Shrnutí implementace

I přes to, že se v některý publikacích uvádí schopnost systému plně pracovat na IPv6, podle mého názoru jeho implementace není bez problému, což naznačuje následující výčet omezení systému:

- absence stavové automatické konfigurace DHCPv6 (řešením je použití alternativní implementace DHCPv6 klienta např. Dibbler. Více informací o projektu Dibbler viz [21].
- všechny DNS dotazy jsou odesílány prostřednictvím IPv4 protokolu (řešením je aktivní IPv4 nebo instalace DNS služby ISC BIND 9 zajišťující překlad IPv6 doménových jmen)
- prohlížeč Internet Explorer pro Windows XP nepodporuje formát pro IPv6 literál (popsaný v dokumentu RFC 2732) URL adresy (řešením je použití alternativního prohlížeče Firefox)

- společná konfigurace Firewallu pro protokol IPv4/IPv6 (řešením je použití jiné implementace filtrující síťový provoz na IPv6 protokolu)
- absence IPv6 podpory pro sdílení souborů a tiskáren (viz [18])

7.2 Linux

Linux ve světe IPv6 není žádným nováčkem, o čem svědčí již mnohaletá stabilní podpora ve starších jádrech systému. První vydání IPv6 kódu bylo přidáno do linuxového jádra 2.1.8 v roce 1996.

7.2.1 Instalace protokolu

Podpora pro IPv6 je dostupná buď v samotném jádře systému nebo ve formě modulu, který se zavádí při startu systému. Vzhledem k neustálému vývoji samotného protokolu se zaměříme na novější jádra verze 2.6.x, nabízející kvalitní a stabilní funkcionalitu systému.

U některých linuxových distribucí se můžeme setkat s vypnutou podporou IPv6. Důvodem je nízká globální rozšířenost nového protokolu. Než tedy začneme využívat IPv6 protokol, musíme zkontrolovat připravenost jádra pracovat s tímto protokolem.

Ke kontrole podpory IPv6 v jádře OS slouží následující příkaz:

```
$cat /proc/net/if_inet6
```

Výstup by měl být podobný tomuto:

```
00000000000000000000000000000001 01 80 10 80       lo
fe8000000000000023048fffe5b411a 02 40 20 80     eth0
fe8000000000000023048fffe5b411b 03 40 20 80     eth1
```

Jedná se o výpis IPv6 adres na vašich síťových rozhraních, kde první řádek příkladu identifikuje tzv. loopback smyčku místního počítače s adresou ::1/128. Druhý a třetí řádek identifikuje lokální linkové adresy rozhraní `eth0` a `eth1`, odvozené z hardwarových MAC adres rozhraní.

Pokud příkaz selhal a tento soubor ve vašem systému neexistuje, s největší pravděpodobností nemá jádro systému načtenou podporou IPv6 síťové komunikace.

Načtení modulu se provede s právy super uživatele následujícím příkazem:

```
$modprobe ipv6
```

Úspěšné načtení lze ověřit mnoha způsoby, nejlépe je lze ověřit nástrojem `lsmod`, který zobrazuje stav načtených modulů jádra systému:

```
$lsmod | grep ipv6
```

Výstup by měl být podobný následujícímu:

```
ipv6                  287584  20
```

Pokud i tento příkaz selhal, doporučuji se poohlédnout po jiné linuxové distribuci, která již podporu IPv6 obsahuje. Kvalitní distribucí operačního systému Linux je CentOS [25].

7.2.2 Konfigurace síťových rozhraní

Linux nabízí nepřeberné možnosti nastavení síťových rozhraní, od editace konfiguračních souborů, po robustní konfigurační nástroje jako `ip`, `route`, `ifconfig`, či `traceroute6`. Konfigurační nástroj `ip`, se vyskytuje téměř ve všech dostupných distribucích.

Přidání IPv6 adresy síťovému rozhraní:

```
$ip addr add 2001:918:fffc:12:1::2/64 dev eth0
```

Zrušení IPv6 adresy:

```
$ip addr del 2001:918:fffc:12:1::2/64 dev eth0
```

Další důležitou součástí nastavení je povolení bezstavové konfigurace, založené na ohlašování směrovače. Následující příkazy povolí automatikou konfigurace hostitelského počítače.

```
$echo “1“ > /proc/sys/net/ipv6/conf/eth0/accept_ra
$echo “1“ > /proc/sys/net/ipv6/conf/eth0/autoconf
```

Naopak, vypnutí automatické konfigurace lze provést pouhou záměnou z hodnoty jedna na hodnotu nula. U všech těchto příkazů nastavuji pro ilustraci síťové rozhraní `eth0` .

Pokud je nasazen systém do role směrovače, je nutné nastavit přeposílání paketů mezi jednotlivým rozhraními. Toto nastavení lze provést nástrojem `sysctl`, který sloužící ke konfiguraci parametrů jádra systému za běhu.

```
$sysctl net.ipv6.conf.all.forwarding=1
```

7.2.3 Konfigurace DNS serveru

Nastavení DNS serveru se provádí v souboru `/etc/resolv.conf`. Jedná se konfigurační soubor systémového překladače doménových jmen, zajišťující překlad doménových jmen na IP adresy. Soubor by měl obsahovat řádky s IPv4/IPv6 adresami serverů poskytujících službu DNS.

Formát zápisu je následující:

```
nameserver ipv4/ipv6_adresa_dns_serveru
```

Např.

```
nameserver 2001:718:1001:149::9
```

Vyhledávání probíhá v tom pořadí, v jakém jsou adresy dostupných serverů zapsány v konfiguračním souboru. Tento konfigurační soubor nabízí další konfigurační rozšíření jako je nastavení lokálního doménového jména pro snazší vyhledávání v lokální síti.

Funkci systémového překladače doménových jmen lze prověřit nástroji `nslookup` nebo `hosts`. Jejich funkcí je překlad doménového jméno na odpovídající IPv6 adresu. Mapování IP adres na názvy počítačů je v souboru `/etc/hosts` .

7.2.4 Konfigurace směrování

Veškerá nastavení, týkající se směrování lze nastavit nástrojem `route` .

Směrovací tabulku jádra získáme příkazem:

```
$route -6
```

Nejčastější případem konfigurace směrování je nastavení výchozí brány. Výchozí brána zde slouží jako implicitní brána, na kterou se přeposílají všechny síťové pakety, ke kterým nebyla nalezena cesta.

Nastavení výchozí brány:

```
$route -A inet6 add 2000::/3 gw <adresa_brány> dev <síťové_rozhraní>
```

Např.:

```
$route -A inet6 add 2000::/3 gw 3ffe:ffff:1234:5678::1 dev eth0
```

Nastavení zkontrolujte výpisem směrovací tabulky.

7.2.5 Tunelování

Linux nabízí více typů IPv6 tunelování, jako je tunelování point-to-point, automatické tunelování, či mechanismus tunelování 6to4. Slouží k zajištění IPv6 konektivity sítí, které jsou přístupné pouze přes IPv4 protokol. Tento princip využívaly organizace jako 6bone, zajišťující IPv6 konektivitu lidem z celého světa.

Příklad vytvoření manuálního tunelování IPv6 nad IPv4 protokolem:

```
$ip tunnel add T mode sit remote 10.0.0.1 local 10.0.0.2 ttl 255
$ip link set dev T up
$ip addr add fec0:0:0:0::2/64 dev T
$ip route add fec0:0:0:0::/64 dev T
```

Jedná se o vytvoření tunelu s názvem T, na který se tuneluje veškerý provoz s prefixem `fec0:0:0:0::/64`. Tunelování probíhá přes rozhraní `10.0.0.2` na uzel s adresou `10.0.0.1`.

Vytvořená tunelování lze zobrazit příkazem:

```
$ip tunnel show
```

Výpis by v tomto příkladě vypadal následovně:

```
T: ipv6/ip  remote 10.0.0.1  local 10.0.0.2  ttl 255
```

7.2.6 Bezpečnost

Filtrování síťové komunikace zajišťuje pro protokol IPv4 nástroj `iptables`. Součástí balíku `iptables` je nástroj `ip6tables` filtrující IPv6 síťovou komunikaci.

Následující příklady demonstrují použití a možnosti nástroje `ip6tables`.

```
$ip6tables –-flush # vymaže veškerá pravidla paketového filtru
$ip6tables –-list  # zobrazí pravidla paketového filtru
```

Povolení příchozího ICMPv6 provozu:

```
$ip6tables –A INPUT –p icmpv6 –j ACCEPT
```

Zakázání ICMPv6 zpráv typu odezva:

```
$ip6tables –A INPUT –p icmpv6 -–icmpv6-type echo-request –j DROP
```

Tento nástroj nabízí velké množství přepínačů, jejichž popis lze nalézt v manuálových stránkách nástroje.

7.3 FreeBSD

Hledáte-li stabilní a výkonný síťový operační systém podporující IPv6, vaše pozornost by se měla obrátit na operační systémy řady BSD. Podporu IPv6 zajišťuje projekt KAME (http://www.kame.net). Z mého pohledu jde o nejpovedenější implementaci IPv6 pro operační systémy řady BSD (FreeBSD, OpenBSD, NetBSD, BSD/OS).

Tvůrci systému si uvědomují budoucnost postavení IPv6 protokolu, a proto je ve výchozí instalaci systému podpora nového protokolu automaticky zapnuta. Další nespornou výhodou těchto systémů je udržování aktuálnosti implementace různých částí systému, a proto obsahují mechanismy, zajišťující aplikaci nově vydaných aktualizací.

Systém FreeBSD nabízí robustní řešení síťové komunikace, a hraje tak důležitou roli na poli síťového výzkumu po celém světě.

7.3.1 Instalace protokolu

Jak jsem již uvedl, ve výchozí instalaci je podpora pro IPv6 automaticky načtena při spuštění systému. Aktivita protokolu se zkontroluje příkazem `ifconfig`, který vypisuje vytvořenou lokální linkovou adresu síťových rozhraní.

Mnoho konfiguračních nastavení se načítá ze souboru `/etc/rc.conf` nebo se získá automaticky ze sítě. Následující příklady popisují možnosti konfigurace systému načtené ze souboru `/etc/rc.conf`.

K zapnutí či vypnutí IPv6 protokolu slouží volba `ipv6_enable`.

```
ipv6_enable="YES" # povolí použití protokolu ipv6
                  # výchozí hodnota je "NO"
```

Volbou `ipv6_network_interfaces` lze povolit IPv6 pouze na některých rozhraních. Ve výchozím nastavení je nastavena na `"auto"` a zapíná IPv6 na všech síťových rozhraních.

```
ipv6_network_interfaces="de0" # zapne ipv6 pouze na rozhraní de0
                              # výchozí hodnota je "auto"
```

Systém může plnit jednu ze dvou různých rolí, funkci počítače nebo směrovače. Prvky, zajišťující směrování paketů sítě jsou nazývány směrovači. Ostatní jsou hostitelské stanice s jedním síťovým rozhraním. Díky rozsáhlým možnostem konfigurace systému FreeBSD lze systém nasadit do obou těchto rolí.

7.3.2 Konfigurace síťových rozhraní

Nastavení síťových rozhraní je součástí konfiguračního souboru `/etc/rc.conf`. Ke statickému nastavení IPv6 adres hostitelské stanice jsou vyhrazeny dvě volby `ipv6_prefix_<interface>` a `ipv6_ifconfig_<interface>`. U těchto voleb se namísto řetězce `<interface>` dosadí název síťového rozhraní. Pokud se tedy rozhodneme ke konfiguraci síťového rozhraní s názvem `de0`, volby budou vypadat takto: `ipv6_prefix_de0` a `ipv6_ifconfig_de0`.

- První volbou `ipv6_prefix_<interface>` lze specifikovat prefixy nových IPv6 adres, které se mají přiřadit danému rozhraní. Tyto prefixy udávají počáteční 64-bitovou polovinu adresy. Zbylých 64 bitů se obvykle vytvoří z linkové MAC adresy rozhraní.

  Následující příklad demonstruje použití volby `ipv6_prefix_<interface>`:

```
# soubor /etc/rc.conf
ipv6_prefix_de0="fec0:0000:1111:0001 fec0:ffff:f0f0:0002"
```

  Tento zápis demonstruje vytvoření dvou síťových adres s prefixy uvedenými v uvozovkách. Načtení konfigurace lze zkontrolovat příkazem `$ifconfig de0 inet6`.

```
de0:flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet6 fec0:0:1111:1:203:ffff:fe30:dd63 prefixlen 64
        inet6 fec0:ffff:f0f0:2:203:ffff:fe30:dd63 prefixlen 64
```

- Druhou možností lze specifikovat kompletní IPv6 adresu s délkou prefixu. Tento příklad demonstruje použití volby `ipv6_ifconfig_<interface>`:

```
# soubor /etc/rc.conf
ipv6_ifconfig_de0="fec0:0:0:0::1 prefixlen 64"
```

  Použití opět zkontrolujeme příkazem `$ifconfig de0 inet6`.

```
de0:flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet6 fec0::1 prefixlen 64
```

  Pro konfiguraci směrovačů je vhodnější druhá možnost, díky které se specifikuje celá IP adresa.

- Dalším důležitým nastavením je nastavení implicitního směrovače. Pro nastavení implicitního směrovače je přiřazena volba `ipv6_defaultrouter="<ipv6_adresa>"` konfiguračního souboru `/etc/rc.conf`. Tato volba úzce souvisí s povolením bezstavové automatické konfigurace IP adres, které se nastavují podle směrovače rozesílajícího informace o dané síti. Tímto příkazem lze bezstavovou konfiguraci zapnout či vypnout:

```
$sysctl net.inet6.ip6.accept_rtadv=0 # 0 = vypnuto, 1 = zapnuto
```

Jak jsem již zmínil, systém FreeBSD je robustní síťový systém, ve kterém je nutno rozlišovat role mezi směrovačem a počítačem. U počítače je pro automatickou konfiguraci vhodné povolit bezstavovou konfiguraci, ale u směrovače by to vedlo k nekonzistenci směrovacích informací systému. Proto se důrazně nedoporučuje povolit zpracování těchto informací. Je nutné porozumět novým principům IPv6 protokolu a nenasazovat systém do role, která by vedla k nekonzistentní funkcionalitě.

7.3.3 Konfigurace DNS serveru

Konfigurace je zcela ekvivalentní konfiguraci DNS u systému Linux. Pro nastavení DNS serverů je vyhrazen konfigurační soubor `/etc/resolv.conf`, kterým se řídí systémový překladač doménových jmen.

7.3.4 Konfigurace směrování

Vzhledem k rozsáhlým síťovým možnostem systémů BSD lze ze systému vytvořit plnohodnotný softwarový směrovač, zajišťující směrování jak IPv4, tak IPv6 sítě.

Konfigurace směrování je velice podobná systému Linux. S projektem KAME do systému přibyly nástroje zajišťující dynamické směrování. Díky těmto nástrojům se směrovací tabulka systému mění, na základě topologie sítě, bez zásahu administrátora daného systému.

O dynamické IPv6 směrování se stará démon `route6d`. Spuštění tohoto démona zajišťuje volba `ipv6_router_enable` v souboru `/etc/rc.conf`.

Následující příklad demonstruje možné použití démona:

```
ipv6_router_enable="YES"     # zapnutí směrovacího IPv6 démona
ipv6_router="/usr/sbin/route6d"    # název směrovacího démona
ipv6_router_flags=""               # směrovací příznaky
```

7.3.5 Tunelování

Následující příklad vytvoří tunelované spojení IPv6 komunikace přes IPv4 konektivitu. Výhodou tohoto způsobu je uchování a znovunačtení nastavení po restartu operačního systému. Nastavení se zapisuje do konfiguračního souboru `/etc/rc.conf`.

Vytvoření tunelovaného rozhraní `gif0`:

```
gif_interface=“gif0“
```

Nastavení tunelu založené na IPv4 konektivitě, kde adresa `10.0.0.1` určuje lokální adresu a adresa `10.0.0.2` vzdálenou IPv4 adresu uzlu:

```
gifconfig_gif0=“10.0.0.1 10.0.0.2“
```

K vytvoření IPv6 konektivity přiřadíme lokálnímu rozhraní tunelu IPv6 adresu:

```
ipv6_ifconfig_gif0=“fec0:0:0:0::1/64“
```

Posledním krokem je přidání záznamu o výchozí bráně pro IPv6 komunikaci do směrovací tabulky. Jedná se o IPv6 adresu vzdáleného uzlu:

```
ipv6_defaultrouter=“fec0:0:0:0::2/64“
```

Pokud systém zajišťuje funkci směrovače pro další prvky sítě, je vhodné povolit jeho funkci výchozí brány:

```
ipv6_gateway_enable="YES"
```

7.3.6 Bezpečnost

Kontrolu IPv6 síťového provozu systémů řady BSD zajišťuje nástroj `ip6fw`. Následující příklady demonstrují použití paketového filtru protokolu IPv6.

Zobrazení aktuálně použitých pravidel filtru:

```
$ip6fw –a list
```

Příklad konfigurace pravidla povolující ICMPv6 zprávy typu výzva a ohlášení souseda:

```
$ip6fw add pass ipv6-icmp from any to any icmptype 135,136
```

Zapnutí paketového filtru lze povolit ihned po načtení systému. Zapnutí filtru se provede v konfiguračním souboru `/etc/rc.conf` pod direktivou `ipv6_firewall_enable`.

```
ipv6_firewall_enable="YES"
```

Další direktiva `ipv6_firewall_type` specifikuje typ paketového filtru. Paketový filtr může pracovat v následujících režimech:

`open`	- povolí veškerý síťový provoz
`client`	- pokusí se zabezpečit tento stroj
`simple`	- pokusí se zabezpečit celou síť
`closed`	- zakáže veškerý síťový provoz kromě lo0 rozhraní
`UNKNOWN`	- vypne načítání filtrující pravidla filtru
`filename`	- načte pravidla z uvedeného souboru

Příklad použití filtru v režimu klient:

```
ipv6_firewall_type=“client“
```

Direktiva `ipv6_firewall_logging` slouží k zaznamenávání událostí. Direktiva `ipv6_firewall_quiet` potlačuje zobrazení pravidel na obrazovku.

7.4 HW směrovače Cisco

Společnost Cisco Systems je světovým dodavatelem HW směrovačů. Cisco si již od počátku kladla velké cíle, jak z hlediska dostupnosti tak podpory v oblasti IPv6.

V roce 2000 firma oznámila třífázovou strategii podpory pro IPv6, popsanou v dokumentu Cisco IOS IPv6 Statement of Direction. Tato strategie obsahuje implementaci podpory základních prvků, jako formát IPv6 datagramů a adres, mechanismus objevování sousedů, ICMPv6, tunelování, statické a dynamické směrování, firewall, IPsec, QoS, NAT-PT a další. Oficiální podpora IPv6 byla uvedena od verze operačního systému IOS 12.2 T.

7.4.1 Konfigurace síťových rozhraní

Propracovaný způsob práce s IOS nabízí snadnou a intuitivní práci při jakékoli konfiguraci zařízení. Zapnutí podpory IPv6 provedeme konfigurací IPv6 adresy síťového rozhraní.

Následující příklad demonstruje konfiguraci síťového rozhraní `f0/0` a jeho síťové adresy `3ffe:ffff::1/64`. Více informací lze nalézt v Cisco IOS IPv6 Configuration Guide [3].

```
Router> enable
Router# configure terminal
Router(config)# interface f0/0
Router(config-if)# ipv6 address 3ffe:ffff::1/64
```

Důležitou součástí nastavení je zapnutí IPv6 směrování, které je ve výchozím nastavení vypnuto.

```
Router(config)# ipv6 unicast-routing
```

Nyní je rozhraní `f0/0` schopno plné IPv6 síťové komunikace.

Zajímavou vlastností je následující rozesílání ICMPv6 zpráv typu ohlášení směrovače, které spadá do bezstavové automatické konfigurace zařízení. Směrovač je tak při konfiguraci IPv6 rozhraní se zapnutým směrováním nastaven jako směrovač, zajišťující síťovým uzlům službu implicitního směrovače. Tento způsob chování je velice přínosný, jelikož se v mnoha případech Cisco zařízení nasazují do role prvku sloužícího jako výchozí brána do internetu.

7.4.2 Konfigurace směrování

IOS zařízení nabízí dvě možnosti směrování. Směrování statické manuálně nastavené, či směrování dynamické nastavené algoritmem daného směrování.

Příklad nastavení implicitní cesty IPv6 směrování na rozhraní `f0/0`:

```
Router(config)# ipv6 route ::/0 f0/0
```

7.4.3 Tunelování

Tunelování pro IPv6 zapouzdřuje IPv6 pakety do IPv4 paketů, které přenáší přes IPv4 síť. Nutnou podmínkou realizace tunelu je podpora IPv4 a IPv6 zásobníku na koncových zařízeních, která realizují tunel. IOS Cisco směrovačů podporuje manuální, GRE, IPv4-kompatibilní, 6to4 a ISATAP tunelové IPv6 mechanismy. Jejich princip či konfiguraci lze nalézt v manuálových stránkách. Další informace lze opět nalézt v Cisco IOS IPv6 Configuration Guide viz [3].

Příklad konfigurace manuálního IPv6 tunelu, který využívá lokální IPv4 adresy `10.0.0.1` a vzdálené IPv4 adresy `10.0.0.2` zajišťující tunelování IPv6 adresy `fec0:0:0:0::1/64` :

```
Router(config)# interface f0/0
Router(config-if)# ip address 10.0.0.1 255.255.255.0
Router(config-if)# end
Router(config)# interface tunnel 0
Router(config-if)# ipv6 address fec0:0:0:0::1/64
Router(config-if)# tunnel source f0/0
Router(config-if)# tunnel destination 10.0.0.2
Router(config-if)# tunnel mode ipv6ip
```

7.4.4 Bezpečnost

Bezpečnost Cisco směrovačů zajišťují:

ACL seznamy, určují, který síťový provoz bude blokován a který bude povolen. Nabízí filtrování na základě zdrojové a cílové adresy příchozích či odchozích paketů na konkrétním rozhraní.

IPv6 Firewall je pokročilý filtr síťového provozu, pod který spadá např. kontrola fragmentovaných paketů, zmírnění DoS útoků, kontrola tunelovaných paketů či kontrola rozšiřujících hlaviček paketů.

Následující příklad ukazuje nastavení ACL seznamu s připojením na konkrétní síťové rozhraní, které zakáže veškerou příchozí ICMPv6 síťovou komunikaci:

```
Router(config)# ipv6 access-list acllist
Router(config-ipv6-acl)# deny icmp any any
Router(config-ipv6-acl)# end
Router(config)# interface f0/0
Router(config-if)# ipv6 traffic-filter acllist in
```

Příklad zobrazení ACL listu:

```
Router# show access-lists acllist
```

8 Rozdělení adres pro lokální síť

Důležitou součástí IPv6 adresování je vytvoření kvalitního návrhu rozdělení adres pro celou síť dané organizace. Obecně lze koncovému zákazníkovi přidělit síťový prefix v rozmezí 32-64 bitů. Ve většině případů se bude jednat o 48-bitový prefix, za kterým se nachází 16 bitů (tzv. subnet ID) pro rozdělení adres do podsítí dané organizace.

Tato kapitola popisuje způsob rozdělení oněch šestnácti bitů a vytvoření vhodné infrastruktury sítě.

Proč je důležité navrhnout dobrou strategii rozdělení IPv6 adres ?

- kontrola a velikost směrovacích tabulek
- Zlepšuje hierarchii, vyhledávání a výkonnost směrování

Způsob dělení Subnet ID je zcela v kontextu administrátora. Doporučení RIPE k dělení adres vybízí k hierarchické distribuci adres, kopírující topologii síťové infrastruktury.

Možnosti dělení adres:

- Menší organizace s jednou podsítí mohou využívat tzv. ploché síťové struktury, nastavením všech bitů na nuly.
- Středně velké organizace mohou vytvořit ekvivalentní strategii podsítí, jako u již vytvořených podsítí s protokolem IPv4.
- Velké organizace mohou využít více úrovňové hierarchie podsítí. Tento způsob vychází z postupného dělení podsítí na další podsítě podsítí.

8.1 Příklad rozdělení adres

Následující příklad naznačuje možný způsob hierarchického dělení přiděleného prefixu podle působnosti jednotlivých oddělení dané organizace. Přidělený prefix 2001:db8:1234::/48 by měl zastupovat celou organizaci sídlící v různých lokalitách. Organizace sídlí v osmi lokalitách, v každé lokalitě má organizace až 16 budov a každá budova se dělí až do 16 oddělení.

Nyní je důležité stanovit počet všech bitů, které vyžaduje každá podsíť a zároveň určit rozsah každé podsítě, aby nedocházelo k alokaci adres do nesprávné podsítě. K tomuto účelu využijeme následující vzorce:

$f = m - 48$	*(kde m je délka přiděleného prefixu)*
$n = 2^s$	*(kde n je počet získaných prefixů podsítí;s je počet použitých bitů)*
$i = 2^{16-(f+s)}$	*(kde i představuje přírůstek, podle kterého dělíme adresu na podsítě)*

1. Prvním krokem je dělení adresy do podsítí lokalit organizace.

   $f = 48 - 48 = 0$; $n = 2^3 = 8$; $i = 2^{16-(f+s)} = 2^{13} = 8192 = 0x2000$

Adresový prefix	*Adresa prefixu podsítě*
1	2001:db8:1234:0000::/51
2	2001:db8:1234:2000::/51
3	2001:db8:1234:4000::/51
4	2001:db8:1234:6000::/51
5	2001:db8:1234:8000::/51
6	2001:db8:1234:A000::/51
7	2001:db8:1234:C000::/51
8	2001:db8:1234:E000::/51

Každá lokalita bude zastupována jedním z těchto prefixů.

2. Nyní vytvoříme rozdělení prefixu 2001:db8:1234:C000::/51 zastupující konkrétní lokalitu.

   $f = 51 - 48 = 3$; $n = 2^4 = 16$; $i = 2^{16-(f+s)} = 2^9 = 512 = 0x200$

Adresový prefix	*Adresa prefixu podsítě*	*Adresový prefix*	*Adresa prefixu podsítě*
1	2001:db8:1234:C000::/55	9	2001:db8:1234:D000::/55
2	2001:db8:1234:C200::/55	10	2001:db8:1234:D200::/55
3	2001:db8:1234:C400::/55	11	2001:db8:1234:D400::/55
4	2001:db8:1234:C600::/55	12	2001:db8:1234:D600::/55
5	2001:db8:1234:C800::/55	13	2001:db8:1234:D800::/55
6	2001:db8:1234:CA00::/55	14	2001:db8:1234:DA00::/55
7	2001:db8:1234:CC00::/55	15	2001:db8:1234:DC00::/55
8	2001:db8:1234:CE00::/55	16	2001:db8:1234:DE00::/55

Tyto prefixy představují dělení budov jedné konkrétní lokality organizace.

3. Posledním krokem je rozdělení adresy každé budovy na oddělení. Pro náš příklad použijeme prefix 2001:db8:1234:C200::/55.

   $f = 55 - 48 = 7$; $n = 2^4 = 16$; $i = 2^{16-(f+s)} = 2^5 = 32 = 0x20$

Adresový prefix	*Adresa prefixu podsítě*	*Adresový prefix*	*Adresa prefixu podsítě*
1	2001:db8:1234:C200::/59	9	2001:db8:1234:C300::/59
2	2001:db8:1234:C220::/59	10	2001:db8:1234:C320::/59
3	2001:db8:1234:C240::/59	11	2001:db8:1234:C340::/59
4	2001:db8:1234:C260::/59	12	2001:db8:1234:C360::/59
5	2001:db8:1234:C280::/59	13	2001:db8:1234:C380::/59
6	2001:db8:1234:C2A0::/59	14	2001:db8:1234:C3A0::/59
7	2001:db8:1234:C2C0::/59	15	2001:db8:1234:C3C0::/59
8	2001:db8:1234:C2E0::/59	16	2001:db8:1234:C3E0::/59

Takto lze vytvořit strukturu pro oddělení jedné budovy v jedné lokalitě dané organizace. Problém, který by mohl nastat, je rozšíření působení organizace o další lokalitu. Tomu lze předejít kvalitním návrhem s předpokládaným možným rozšiřováním organizace o nové lokality, budovy či oddělení.

Tento příklad ukázal jednoduchost a zároveň robustnost možného hierarchického dělení na jednotlivé části organizace.

9 Přidělení adres pro lokální síť

O přidělení či automatickou konfiguraci IP adres se u IPv6 starají dva mechanismy. První způsob zastupuje stavová automatická konfigurace, známá pod názvem DHCPv6.

Druhou možností je bezstavová konfigurace, která představuje novinku v oblasti konfigurace IPv6 adres. Principem konfigurace je prvek rozesílající informace o síti. Z těchto informací si hostitelské stanice vytvoří novou IPv6, aniž by se vedla nějaká centrální evidence vypůjčených IP adres, jak je tomu u DHCP. Tento princip vyžaduje další mechanismy, jako je detekce duplicitních adres.

9.1 Nastavení počítačů a aktivních prvků sítě

Tento příklad demonstruje vytvoření softwarového směrovače, který zajišťuje bezstavovou automatickou konfiguraci na základě periodického ohlašování informací o dané síti.

Užitečným nástrojem systému FreeBSD je démon `rtadvd`, zajišťující ohlašování informací. Konfigurace systému probíhá v následujících krocích:

1. Prvním krokem je vypnutí automatické konfigurace směrovače, jelikož by měl mít své adresy nastaveny staticky.

```
$sysctl net.inet6.ip6.accept_rtadv=0
```

2. Nastavení systému do role směrovače zajistí následující volba nastavení `/etc/rc.conf`.

```
ipv6_gateway_enable="YES"
```

3. Pro zapnutí démona jsou vyhrazeny následující volby nastavení souboru `/etc/rc.conf`.

```
rtadvd_enable=“YES“     # zapnutí démona
rtadvd_interfaces=“de0“ # definuje rozhraní, na které jsou
                        # odesílány ohlášení směrovače
```

Důležitá volba `rtadvd_interfaces` specifikuje síťové rozhraní, na které budou odesílány ohlašovací informace.

Vzhledem k bezpečnostním důvodům se důrazně nedoporučuje rozesílat informace do všech sítí, tedy i do veřejného internetového rozhraní, jak je tomu u výchozího nastavení této volby.
V nastavení by měla být uvedena pouze síťová rozhraní do privátních sítí.

4. Nastavení ohlašovaných informací démona je vyhrazen konfigurační soubor `/etc/rtadvd.conf`. Následující příklad demonstruje možné nastavení démona:

```
de0:\
   :addr="2001:0DB8:0:2::":prefixlen#64:tc=ether:
```

Démon v tomto případě ohlašuje na rozhraní `de0` prefix sítě `2001:0DB8:0:2::` délky 64 bitů.
Tento postup ukázal jak vytvořit softwarový směrovač, zajišťující funkci implicitního směrovače pro lokální IPv6 síť.

10 Dostupnost IPv6 ve službě DNS

Rozšiřující standard DNS o IPv6 adresy vychází z dokumentu RFC 3596 (viz [12]). Tento standard obohacuje DNS specifikaci o nový záznam AAAA, zpětnou IPv6 doménu a aktualizuje formát dotazování DNS služby.

Nový záznam AAAA mapuje doménová jména na IPv6 adresy. Jedná se o stejný typ záznamu, jak je tomu u protokolu IPv4, kde se označuje pouze jedním A. Čtyři písmena A zde symbolizují čtyřikrát větší adresu IPv6 oproti klasické IPv4 adrese.

Tento příklad naznačuje zápis záznamu typu A, tak jeho nástupce AAAA:

```
nameserver      IN      A       12.34.56.78
ipv6nameserver  IN      AAAA    3ffe::ffe0:100:11
```

Zpětné dotazy slouží k získávání doménových jmen na základě IP adresy. Stejně jako u IPv4 adres, se IPv6 adresy zapisují odzadu. S tím rozdílem, že u IPv4 se zapisují po celých bajtech a u IPv6 v tzv. nibble-formátu, tedy po půl bajtech.

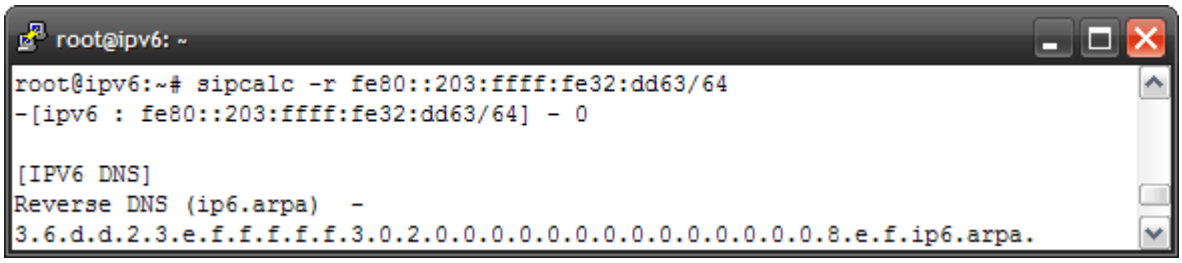
Další změnu zaznamenala koncová doména ukončující reverzní adresu, která se na základě dokumentu RFC 3152 z původního formátu `.ip6.int` změnila na adresu `.ip6.arpa`. Následující příklad demonstruje zápis reverzní IPv6 adresy k adrese `3ffe::100:201:2ff:feb5:1`.

```
1.0.0.0.5.b.e.f.f.f.2.0.1.0.2.0.0.0.0.1.0.e.f.f.3.ip6.arpa.
```

Jak je z příkladu patrné, nuly se nevynechávají.

Nástroj `sipcalc` slouží pro generování reverzních adres. Jeho funkcí je síťový kalkulátor podporující jak IPv4 tak IPv6 protokol.

Parametr –r zajistí vytvoření reverzní IPv6 adresy k adrese v příkladu:



Více informací o nástroji `sipcalc` viz [22].

Změn se dočkaly i samotné dotazy na DNS záznamy typu NS (jmenný server) a MX (emailový server), které před uvedením standardu RFC 1886 nebyly schopny nést informaci o IPv6 adrese.

10.1 BIND 9

Dnes již existuje velké množství DNS implementací. Nejznámější na poli unixových systémů je BIND 9.x.

BIND je dnes nejpoužívanějším DNS serverem internetu. Jeho první vydání se objevilo již u systému BSD verze 4.3, jako projekt čtyř absolventů na univerzitě v Berkeley v roce 1986.

Podpora IPv6 byla postupně rozšiřována o nové typy záznamů a pokračovala do dnešní podoby (verze 9.4.2), kdy je server schopen přijímat a odesílat DNS zprávy na základě IPv6 komunikace.

Následující kapitola popisuje způsob konfigurace DNS serveru BIND 9.x podporující oba protokoly IPv4 a IPv6.

10.2 Konfigurace DNS serveru BIND 9

Hlavním konfiguračním souborem serveru je soubor `/etc/named.conf`, který řídí chování a funkcionalitu serveru.

Ve výchozím nastavení BIND nenaslouchá na IPv6, ale pouze na IPv4 síťovém socketu. IPv6 povolíme přidáním direktivy `listen-on-v6` do sekce `options` souboru `named.conf`. V závorkách se určí adresy, na kterých bude server naslouchat.

```
options {
       listen-on-v6 { 3ffe:1234:5678::1; };
}
```

Další změny se budou týkat samotných zónových souborů, nesoucí informace o spravovaných doménách. Následující příklad demonstruje vytvoření zónových souborů pro doménu domena.cz.

Předtím, než se vytvoří zónové soubor, musí BIND znát o jaké domény se má starat. Informace o zónových souborech se zapisují v následujícím formátu do souboru `named.conf`:

```
zone "název domény" {
        type master;
        file "cesta k zónovému souboru";
};
```

Pro příklad domény domena.cz, by soubor vypadal následovně:

```
zone "domena.cz" {
        type master;
        file "/etc/zones/domena.cz";
};
```

Obdobným způsobem se zapisují zóny pro reverzní záznamy. Jestliže DNS server spravuje IPv6 prefix `3ffe:1234:5678::/48`. Tento prefix, by měl v názvu zóny následující zápis:

```
zone "8.7.6.5.4.3.2.1.e.f.f.3.ip6.arpa" {
        type master;
        file "/etc/zones/8.7.6.5.4.3.2.1.e.f.f.3.ip6.arpa";
};
```

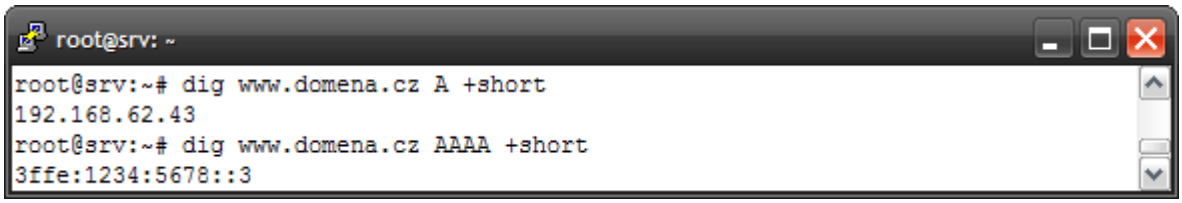
Samozřejmě se musí změnit i název souboru pro danou zónu spravující odpovídající záznamy na reverzní dotazy.

Posledním krokem je vytvoření konkrétních zónových souborů, nesoucí záznamy o překladech doménových jmen na IP adresy. Ty mají následující zápis:

```
$TTL 3D
@           IN          SOA   ns1.domena.cz. dns.domena.cz. (
                        1999080801  ; sériové číslo
                        10800       ; obnovovací frekvence, sekundy
                        1800        ; opakované pokusy, sekundy
                        120960      ; exspirace, sekundy
                        604800 )    ; minimum, sekundy
;
            IN          NS                ns1
            IN          NS                ns2
;
ns1         IN          A                 192.168.62.169
            IN          AAAA              3ffe:1234:5678::1
ns2         IN          A                 192.168.62.230
            IN          AAAA              3ffe:1234:5678::2
www         IN          A                 192.168.62.43
            IN          AAAA              3ffe:1234:5678::3
```

Dotaz na IPv6 adresu www.domena.cz vrátí IPv6 adresu `3ffe:1234:5678::3`, nikoli adresu `192.168.62.43`. To je zapříčiněno samotným DNS dotazem, požadujícím typ záznamu AAAA a nikoli typ A. Správnou funkci serveru lze prověřit nástrojem `dig`, který vyhledává DNS záznamy na DNS serverech a zobrazuje přehledný výpis výsledku dotazu.

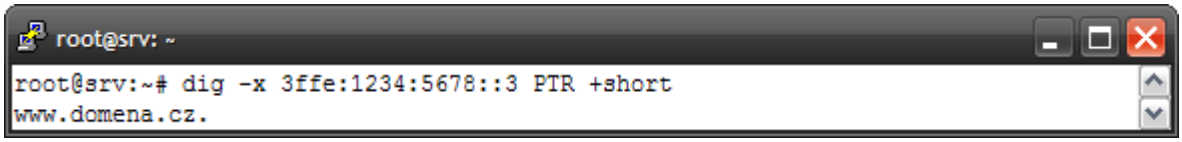
Příklad vyhledávání IPv4 a IPv6 adres doménového jména www.domena.cz:



Vytvořený zónový soubor pro reverzní záznamy obsahuje stejnou strukturu pro zápis záznamů. Příklad souboru pro reverzní zónu:

```
$TTL 3D
@           IN    SOA   ns1.domena.cz. dns.domena.cz. (
                  1999080801  ; sériové číslo
                  10800       ; obnovovací frekvence, sekundy
                  1800        ; opakované pokusy, sekundy
                  120960      ; exspirace, sekundy
                  604800 )    ; minimum, sekundy
;
            IN    NS                ns1
            IN    NS                ns2
;
1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0   IN    PTR ns1.domena.cz.
2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0   IN    PTR ns2.domena.cz.
3.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0   IN    PTR www.domena.cz.
```

Příklad vyhledávání doménového jména z IPv6 adresy:



10.3 Ukládání záznamů

Ukládání nových typů záznamů na první pohled vypadá jednoduše. Skutečnost je poněkud složitější. Mnoho nových aplikací preferuje dotazování na nový typ AAAA záznamu před typem A. Dotaz takovéto aplikace bez IPv6 konektivity způsobí nedostupnost dané služby i přes to, že je DNS server naprosto v pořádku. Vše je zapříčiněno ukládáním všech záznamů do stejné domény. Aplikace obdrží IPv6 adresu dané služby, ale samotné připojení nelze navázat z důvodu chybějící konektivity aplikace k dané IPv6 síti.

Tento nepříjemný problém se řeší ukládáním IPv6 záznamů do jiné vyhrazené domény, určené pouze pro IPv6 záznamy. Nejlepším způsobem je ukládání IPv6 záznamů do domény s názvem ipv6.domena.cz.

11 Dostupnost IPv6 ve službě DHCP

Server DHCPv6 je součástí většiny unixových systémů, lze jej zdarma získat ve formě zdrojových kódů na internetových stránkách projektu.

DHCPv6 server realizuje démon `dhcp6s`. Tímto démonem lze nabízet informace např. IPv6 prefixy pro danou síť, seznam adres dostupných DNS serverů, seznam adres SIP serverů a další.

11.1 Konfigurace DHCP serveru DHCPv6

Konfigurace `dhcp6s` démona je zapsána v souboru `/etc/dhcp6s.conf`. Následující text popisuje konfiguraci zdrojových informací souboru `dhcp6s.conf`, nabízející DHCPv6 server ve formě zpráv DHCPv6 klientům.

Konfigurační soubor DHCPv6 serveru vypadá následovně:

```
interface eth0 {
    server-preference 255;
    renew-time 60;
    rebind-time 90;
    prefer-life-time 130;
    valid-life-time 200;
    allow rapid-commit;
    option dns_servers 2001:db8:0:f101::1 domena.cz;
    link BBB {
         pool{
         range 2001:db8:0:f101::4 to 2001:db8:0:f101::ffff/64;
         prefix 2001:db8:0::/48;
         };
    };
};
```

Následující tabulka popisuje některá nastavení:

`interface`	Tato volba určuje rozhraní, na kterém DHCPv6 server naslouchá.
`server-preference`	Hodnota tohoto parametru určuje váhu daného serveru. Klient se na základě této hodnoty rozhodne, který DHCP server bude využívat.
`renew-time`	Určuje dobu, po které dojde k obnovení zapůjčení adresy serveru DHCPv6 (v sekundách).
`options dns_servers`	Udává seznam IPv6 adres nebo doménových jmen DNS serverů.
`Range`	Určuje rozsah adres, které je možné zapůjčit.
`Prefix`	Umožňuje určit prefix sítě.

Z popsané konfigurace je zřejmá možnost vytvoření většího počtu rozsahů, patřící různým síťovým rozhraním. Jeden DHCPv6 server lze použít pro dva oddělené segmenty sítě.

DHCPv6 server lze spustit příkazem:

```
$dhcp6s -df eth0 -c /etc/dhcp6s.conf
```

Větší množství nastavení lze nalézt v manuálových stránkách konfiguračního souboru `dhcp6s.conf`.

11.2 Konfigurace DHCPv6 klienta

O funkci klienta se stará démon s názvem `dhcp6c`. Ten získává informace od DHCPv6 serveru a nastavuje určené síťové rozhraní. Konfigurace DHCPv6 klienta probíhá podobně jako u serveru, s tím rozdílem, že se konfigurační soubor nazývá `/etc/dhcp6c.conf`. Konfigurační soubor obsahuje informace o síťovém rozhraní určené k vysílání DHCPv6 požadavků. Součástí každého nastavení je typ informací, které klient od serveru žádá.

Příklad nastavení souboru `dhcp6c.conf`:

```
interface eth0 {
        request domain-name-servers;
};
```

Volba `interface` definuje rozhraní, na které jsou odesílány požadavky DHCP klienta. Nastavení `request domain-name-servers`, přidá do DHCPv6 zprávy požadavek na seznam DNS serverů.

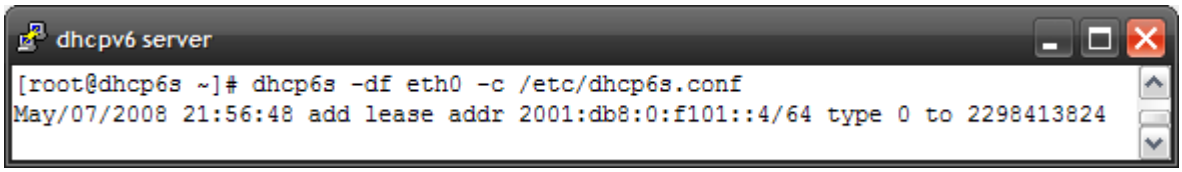
DHCPv6 klient může získat pouze ty informace, které nabízí DHCPv6 server. Pokud klient žádá typ informací, které server nenabízí, musí si vystačit pouze s nabídnutými informacemi.

Spuštění klienta zajistí tento příkaz:

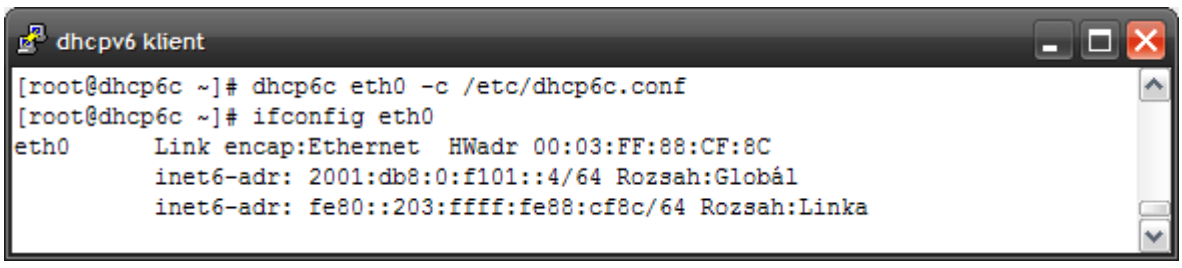
```
$dhcp6c -df eth0 –c /etc/dhcp6c.conf
```

Větší množství nastavení lze nalézt v manuálních stránkách souboru `dhcp6c.conf`.

Tento příklad ukazuje spuštění DHCPv6 serveru a přidělení jedné IPv6 adresy klientskému počítači:



Ukázka spuštění DHCPv6 klienta a výpis nastavení síťového rozhraní:



12 Připojení firemní sítě do IPv6 internetu

Hlavním problémem dnešního IPv6 internetu je nízký zájem poskytovatelů nabízet IPv6 adresy. Drtivá většina zákazníků je tak připojena prostřednictvím klasického IPv4 protokolu. I přes tuto skutečnost, lze využít dostupné techniky tunelování zajišťující alternativní způsob připojení do IPv6 internetu. Tento způsob připojení bude po několik let spojovat odříznuté IPv4 sítě k IPv6 internetu.

V uvedeném případě potřebujeme realizovat připojení IPv6 sítě přes dostupnou IPv4 konektivitu. Toto lze realizovat vytvořením prvku podporující jak IPv4 tak IPv6 protokol, který bude zajišťovat tunelování IPv6 síťového provozu přes IPv4 síťovou infrastrukturu internetu.

Vhodným mechanismem je tunelování typu 6to4 (viz [16]), který vytváří IPv6 adresu z přidělené veřejné IPv4 adresy. Tato IPv6 adresa se skládá z přiděleného prefixu 2002::/16 (tento prefix je vyhrazen pro adresy mechanismu 6to4), za kterým následuje 32-bitová IPv4 adresa v hexadecimálním zápisu. Takto se vytvoří vlastní 48-bitový prefix sítě, za který se připojí 16 bitů identifikující podsíť daného rozsahu. Poslední 64 bitů identifikuje koncový počítač či směrovač v dané podsíti.

12.1 Praktické nasazení mechanismu 6to4

Návrh připojení firemní sítě do IPv6 internetu je rozdělen do několika částí. Prvním krokem je vytvoření softwarového směrovače, který zajišťuje spojení mezi vytvořenou IPv6 sítí a internetem. Ve druhém kroku je návrh způsobu rozdělení a přidělení IPv6 adres (jedná se o globálně směrovatelné IPv6 adresy) pro lokální IPv6 síť. Počítače této sítě budou adresy využívat k přístupu do internetu.

K nasazení mechanismu 6to4 lze využít jak HW směrovač Cisco podporující IPv6, tak softwarové řešení postavené na systému Linux či FreeBSD. V tomto případě lze použít systém FreeBSD, který plně dostačuje.

Tento příklad předpokládá přidělenou veřejnou IPv4 adresu na směrovači, zajišťující úlohu implicitního směrovače mezi lokální IPv6 sítí a internetem.

1. První krokem je nastavení síťových rozhraní a směrování. Pro perzistentní nastavení se provede zápis do konfigurace do souboru `/etc/rc.conf`.

```
ipv6_enable=“YES“               # povolení IPv6 protokolu
ipv6_network_interfaces=“auto“  # IPv6 na všech rozhraních
```

Nastavení systému do role směrovače:

```
ipv6_gateway_enable=“YES“
```

Nastavení IPv6 adres směrovače:

```
ipv6_ifconfig_de0="2002:c1a5:505::1 prefixlen 64"
ipv6_ifconfig_de1="2002:c1a5:505::2 prefixlen 64"
```

Konfigurace veřejné IPv4 adresy pro mechanismus 6to4:

```
stf_interface_ipv4addr="193.165.5.5"
```

Dalším krokem je nastavení implicitního směrování na bránu 6to4 v internetu, která bude sloužit jako druhý konec IPv4 spojení. Jeho úkolem je rozbalení zapouzdřených IPv6 paketů a jejich přeposílání do IPv6 internetu. Výchozí brána pro směrovač se nastaví nástrojem `route`:

```
$route add -inet6 default 2002:c058:6301::
```

2. Tato konfigurace popisuje přidělení adres pro vytvořenou IPv6 síť. V tomto příkladě lze využít bezstavový mechanismus konfigurace počítačů, který zajišťuje démon `rtadvd`.

   Nastavení souboru `/etc/rc.conf`, pro automatickou konfiguraci:

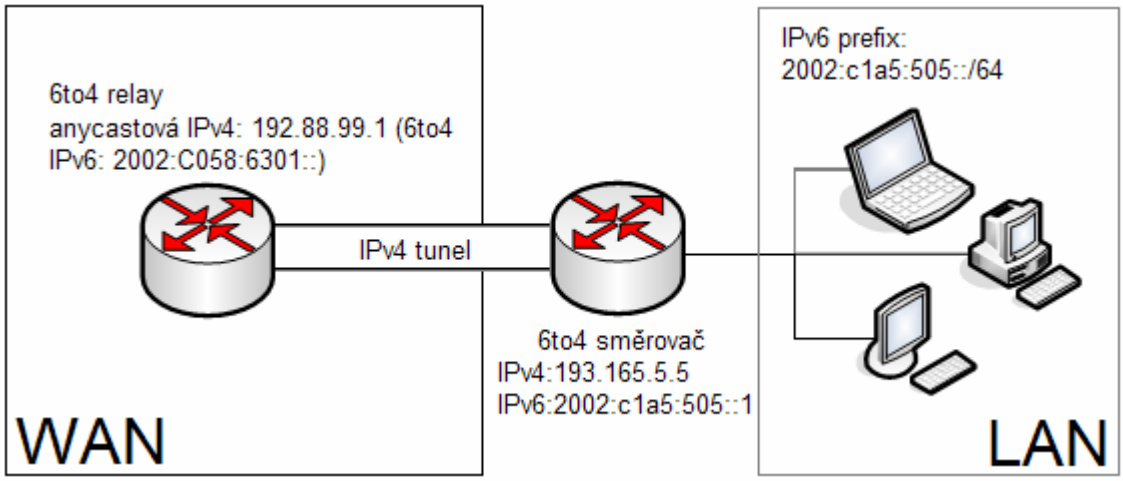
```
rtadvd_enable="YES"          # povolení démon rtadvd
rtadvd_interfaces="de1"      # privátní rozhraní pro lokální síť
```

Konfigurace ohlašovaných informací se zapisují do souboru `/etc/rtadvd.conf`:

```
de1:\
   :addr="2002:c1a5:505::":prefixlen#64:tc=ether:
```

Detailnější popis nastavení je uveden v manuálových stránkách souboru `rtadvd.conf`. Nevýhodou bezstavové konfigurace je absence síření informací o dostupných DNS serverech. Proto je nutné na každém počítači DNS nastavit nebo zapojit DHPCv6 server, který tuto službu poskytuje.

Schéma ukazuje zapojení sítě:



Ukázka nastavení IPv6 adres síťových rozhraní směrovače 6to4:

ifconfig

```
dhcppc5# ifconfig -a inet6
de0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet6 fe80::203:ffff:fe62:5778%de0 prefixlen 64 scopeid 0x1
        inet6 2002:c1a5:505::1 prefixlen 64
de1: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet6 fe80::203:ffff:fe63:5778%de1 prefixlen 64 scopeid 0x2
        inet6 2002:c1a5:505::2 prefixlen 64
plip0: flags=108851<UP,POINTOPOINT,RUNNING,SIMPLEX,MULTICAST,NEEDSGIANT> mtu 150
0
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x4
stf0: flags=1<UP> mtu 1280
        inet6 2002:c1a5:505::1 prefixlen 16
```

Ukázka IPv6 konektivity směrovače:

ping6 www.ipv6.cz

```
dhcppc5# ping6 www.ipv6.cz
PING www.ipv6.cz(netopeer.cz) 56 data bytes
64 bytes from flowmon.cz: icmp_seq=1 ttl=54 time=92.6 ms
64 bytes from mika2.cesnet.cz: icmp_seq=2 ttl=54 time=92.7 ms
64 bytes from rcna.cesnet.cz: icmp_seq=3 ttl=54 time=92.3 ms

--- www.ipv6.cz ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2023ms
rtt min/avg/max/mdev = 92.376/92.589/92.759/0.295 ms
```

Výpis ICMPv6 paketů:

tcpdump

```
dhcppc5# tcpdump -ni stf0 icmp6
listening on stf0, link-type RAW (Raw IP), capture size 96 bytes
11:29:13.690564 IP6 2002:c1a5:505::1 > 2001:718:1:4:214:22ff:fe17:51d0: ICMP6, e
cho request, seq 1, length 64
11:29:13.784753 IP6 2001:718:1:4:214:22ff:fe17:51d0 > 2002:c1a5:505::1: ICMP6, e
cho reply, seq 1, length 64
11:29:14.701740 IP6 2002:c1a5:505::1 > 2001:718:1:4:214:22ff:fe17:51d0: ICMP6, e
cho request, seq 2, length 64
11:29:14.793946 IP6 2001:718:1:4:214:22ff:fe17:51d0 > 2002:c1a5:505::1: ICMP6, e
cho reply, seq 2, length 64
11:29:15.711714 IP6 2002:c1a5:505::1 > 2001:718:1:4:214:22ff:fe17:51d0: ICMP6, e
cho request, seq 3, length 64
11:29:15.804838 IP6 2001:718:1:4:214:22ff:fe17:51d0 > 2002:c1a5:505::1: ICMP6, e
cho reply, seq 3, length 64

6 packets captured
12 packets received by filter
0 packets dropped by kernel
```

Ukázka směrovací tabulky směrovače:

směrovací tabulka

```
Internet6:
Destination                       Gateway                       Flags      Netif
Expire
::/96                             ::1                           UGRS        lo0 =
>
default                           2002:c058:6301::              UGS        stf0
::1                               ::1                           UHL         lo0
::ffff:0.0.0.0/96                 ::1                           UGRS        lo0
2002::/24                         ::1                           UGRS        lo0 =
>
2002::/16                         2002:c1a5:505::1              U           de0
2002:7f00::/24                    ::1                           UGRS        lo0
2002:c1a5:505::/64                link#1                        UC          de0
2002:c1a5:505::1                  00:03:ff:62:57:78             UHL         lo0
2002:c1a5:505::2                  00:03:ff:63:57:78             UHL         lo0
2002:e000::/20                    ::1                           UGRS        lo0
2002:ff00::/24                    ::1                           UGRS        lo0
```

13 Možnosti připojení IPv6 v ČR

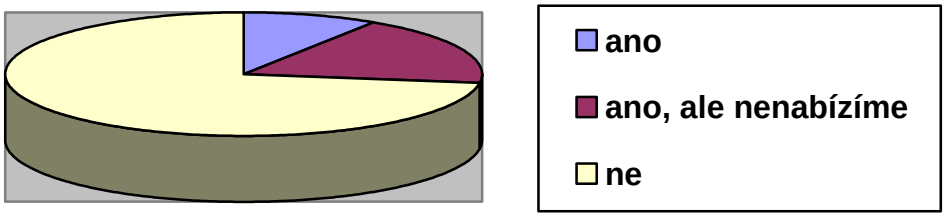
Současný stav dostupnosti IPv6 internetu jsem zmapoval na základě dotazníku, který jsem rozeslal největším ISP poskytovatelům internetu působícím na trhu České republiky. Z celkového počtu třiceti osmi dotázaných poskytovatelů jich odpovědělo třináct. Je patrné, že z odpovědí jedné třetiny dotázaných si nelze vytvářet úplný obraz o stavu současného IPv6 internetu v ČR.

Dotazník:

1. Jste schopni zajistit IPv6 konektivitu svým koncovým zákazníkům (zajistit připojení zákazníka do IPv6 internetu přímo z jeho domu) ?
2. Jaké technologie připojení do IPv6 podporujete ?
3. Pokud se Vaše síť nachází ve fázi testování, ve kterém roce plánujete plnou podporu IPv6 ?
4. Mají zákazníci o IPv6 zájem ?

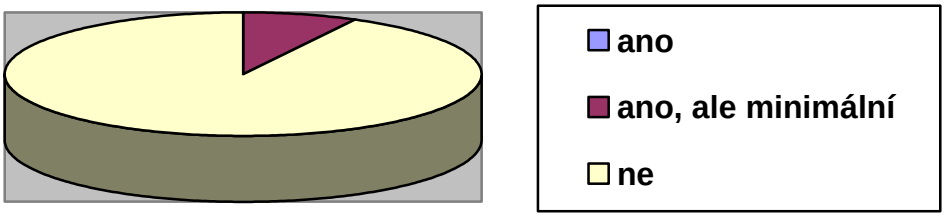
Statistika výsledků:

1. Jste schopni zajistit IPv6 konektivitu svým koncovým zákazníkům ?



Na tuto otázku bylo odpovězeno jednou kladně a to v případě sdružení CESNET, které však nabízí konektivitu pouze institucím pro vědu a výzkum. Dva poskytovatelé jsou schopni IPv6 konektivitu nabídnout, nicméně ji v současné době na základě nulové poptávky nenabízejí. Dalších osm nemá o IPv6 absolutní zájem a neplánují jeho nasazení ani v nejbližší době. Poslední dva se věnují pouze službám Server Hosting (umístění serverů do datacenter s vysokokapacitním připojení k internetu), a proto jsem je do grafu neuvedl.

2. Mají zákazníci o IPv6 zájem ?



K tomuto grafu ani není co dodat. Zákazníci nemají o IPv6 téměř žádný zájem. Poskytovatelé mají zkušenost pouze s obecnými dotazy na IPv6, ale samotný zájem o konektivitu je nulový.

Z těchto grafů si lze udělat menší obrázek o stavu dnešního IPv6 internetu v ČR. Pokud se nalezne koncový uživatel, který by měl zájem o nativní IPv6 konektivitu, má velmi malé šance na její získání. Zákazník je tak odkázán na některé tunelované služby zajišťující IPv6 spojení, jako je tomu u organizace Freenet6 [17] nebo SixXS [24].

Následující tabulka shrnuje odpovědi zástupců dotázaných společností [data 20.2.2008]:

ISP	Otázka 1.	Otázka 2.	Otázka 3.	Otázka 4.
SELF servis, s r.o. http://www.selfservis.cz/	ne	žádné	neplánují	ne
Master Internet, s.r.o. http://www.master.cz/ (Server Hosting)	ne	žádné	neplánují	pouze odborníci
SELF servis, s r.o. http://www.czechbone.cz/	ne	žádné	neplánují	ne
SMART Comp. a.s. http://www.netbox.cz/	ne	žádné	2009 až 2010	pouze odborníci
PODA s.r.o. http://www.poda.cz/	ne	žádné	neplánují	ne
T-Mobile Czech Republic a.s. http://t-mobile.cz/	ano, ale zatím neposkytují	nezjištěno	2011 až 2012	pouze odborníci
CESNET z. s. p. o. http://www.cesnet.cz/	ano, ale pouze institucím	optické okruhy	od roku 2004	University a Akademie věd
FORTECH, s. r. o. http://www.fortech.cz/	ne	žádné	neplánují	ne
České Radiokomunikace a.s. http://www.radiokomunikace.cz/	ne	žádné	2008 až 2009	pouze odborníci
Coma s.r.o. http://www.comacomp.cz/	ne	žádné	neplánují	žádný
ha-vel internet s.r.o. http://www.ha-vel.cz/	ne	žádné	neplánují	ne
Casablanca INT s.r.o. http://www.casablanca.cz/ (Server Hosting)	ne	žádné	neplánují	ne
Telefónica O2 Czech Republic, a.s. http://www.cz.o2.com/	ano, ale zatím neposkytují	CSD	neplánují	ne

14 Závěr

Cílem této práce bylo seznámení se s novým standardem IPv6 protokolu, prověření dostupných implementací jak na poli operačních systémů, tak síťových služeb a návrh vytvoření počítačové sítě pracující výhradně nad IPv6 s přístupem do internetu.

Čtenář má tak možnost získat potřebné informace k pochopení nových principů používaných u nového protokolu. Zároveň získá přehled o stavu implementace u nejpoužívanějších operačních systémů a služeb. Získá cenné informace, které mu usnadní, jak teoretické tak praktické nasazení IPv6 v podnikové síti. Závěr práce se věnuje průzkumu trhu českých ISP a jejich schopnosti a připravenosti poskytovat IPv6 konektivitu koncovým uživatelům. Z kontaktů se zástupci poskytovatelů připojení k internetu jsem zmapoval situaci v České republice.

Práce by mohla pokračovat podrobnějším průzkumem trhu a dalším mapování jeho rozrůstání, v používání IPv6 protokolu. Zajímavým zjištěním, by bylo vytvoření seznamu alternativních dodavatelů HW směrovačů a implementace IPv6 u nových operačních systémů MS Windows Vista nebo MS Server 2008. Zajímavé by bylo srovnání výkonu sítě pracující v IPv6 proti IPv4 síti.

Práce mi přinesla velké množství informací, nejen o samotném fungování protokolu, ale také o konfiguraci HW směrovačů Cisco, či síťového operačního systému FreeBSD. Překvapila mě nízká rozšířenost protokolu IPv6 a nezájem ze strany uživatelů i poskytovatelů internetu o jeho možnosti. Je jisté, že se v několika následujících letech situace rapidně změní. Nedostatek IPv4 protokolu představuje příliš velký problém, který vytvoří tlak na nutnost přistoupit k rychlejšímu zavádění IPv6 protokolu.

Literatura

[1] Pavel Satrapa, *IPv6 Internet Protokol verze 6*. Praha, Neokortex 2002.

[2] Pete Loshin, *IPv6: Theory, Protocol, and Practise SECOND EDITION*. 2004 by Elsevier

[3] *Cisco IOS IPv6 Configuration Guide, Release 12.4* [online, navštíveno 15.3.2008]
Dostupné na URL: http://www.cisco.com/en/US/docs/ios/12_2t/ipv6/hipv6_c.html

[4] IPv6 - Wikipedie, *otevřená encyklopedie* [online, navštíveno 21.4.2008]
Dostupné na URL: http://en.wikipedia.org/wiki/IPv6

[5] IPv6, *information page* [online, navšíveno 7.3.2008]
Dostupné na URL: http://www.ipv6.org/

[6] E. Nordmark, R. Gilligan: *Basic Transition Mechanisms for IPv6 Hosts and Routers*.
RFC 4213 (PROPOSED STANDARD), Říjen 2005, Obsoletes RFC 2893.
Dostupné na URL: http://www.ietf.org/rfc/rfc4213

[7] T. Narten, E. Nordmark, W. Simpson, H. Soliman: *Neighbor Discovery for IP version 6*.
RFC 4861 (DRAFT STANDARD), Září 2007, Obsoletes RFC 2461.
Dostupné na URL: http://www.ietf.org/rfc/rfc4861.txt

[8] S. Thomson, T. Narten, T. Jinmei: *IPv6 Stateless Address Autoconfiguration*.
RFC 4862 (DRAFT STANDARD), Září 2007, Obsoletes RFC 2462.
Dostupné na URL: http://www.ietf.org/rfc/rfc4862.txt

[9] A. Conta, S. Deering, M. Gupta, Ed.: *Internet Control Message Protocol (ICMPv6)*.
RFC 4443 (DRAFT STANDARD), Březen 2006, Obsoletes RFC 2463, Updates RFC 2780, Updated by RFC 4884.
Dostupné na URL: http://www.ietf.org/rfc/rfc4443.txt

[10] R. Hinden, S. Deering: *IP Version 6 Addressing Architecture*.
RFC 4291 (DRAFT STANDARD), Únor 2006, Obsoletes RFC 3513.
Dostupné na URL: http://www.ietf.org/rfc/rfc4291.txt

[11] B. Carpenter, K. Moore: *Connection of IPv6 Domains via IPv4 Clouds*.
RFC 3056 (PROPOSED STANDARD), Únor 2001.
Dostupné na URL: http://www.ietf.org/rfc/rfc3056

[12] S. Thomson, C. Huitema, V. Ksinant, M. Souissi: *DNS Extensions to Support IP version 6*.
RFC 3596 (DRAFT STANDARD), Říjen 2003, Obsoletes RFC 3152, RFC 1886.
Dostupné na URL: http://www.ietf.org/rfc/rfc3596

[13] J. Jeong, Ed.: *IPv6 Host Configuration of DNS Server Information Approaches*.
RFC 4339 (INFORMATIONAL), Únor 2006.
Dostupné na URL: http://www.ietf.org/rfc/rfc4339.txt

[14] KAME project: *Implementation of Dynamic Host Configuration Protocol for IPv6* (DHCPv6).

Dostupné na URL: http://sourceforge.net/projects/wide-dhcpv6/

[15] *IPv6 configuration guide for FreeBSD users* [online, navštíveno 22.3.2008]
Dostupné na URL: http://www.kame.net/~suz/freebsd-ipv6-config-guide.txt

[16] 6to4 - Wikipedie, *otevřená encyklopedie* [online, navštíveno 25.4.2008]
Dostupné na URL: http://en.wikipedia.org/wiki/6to4

[17] go6, *The IPv6 portal: Free IPv6 connectivity* [online, navštíveno 5.3.2008]
Dostupné na URL: http://go6.net/4105/home.asp

[18] IPv6 for Microsoft Windows, *Frequently Asked Questions* [online, navštíveno 17.4.2008]
Dostupné na URL: http://www.microsoft.com/technet/network/ipv6/ipv6faq.mspx

[19] IPv6 - Microsoft TechNet, *Networking and Access Technologies* [online, navštíveno 12.4.2008]
Dostupné na URL: http://microsoft.com/ipv6

[20] IPsec - Wikipedie, *otevřená encyklopedie* [online, navštíveno 7.5.2008]
Dostupné na URL: http://en.wikipedia.org/wiki/IPsec

[21] DHCPv6: *Dibbler - a portable DHCPv6* [online, navštíveno 1.5.2008]
Dostupné na URL: http://klub.com.pl/dhcpv6/

[22] Sipcalc: *Console based IP subnet calculator* [online, navštíveno 28.4.2008]
Dostupné na URL: http://www.routemeister.net/projects/sipcalc/index.html

[23] IPv4 address exhaustion - *Wikipedie, otevřená encyklopedie* [online, navštíveno 25.4.2008]
Dostupné na URL: http://en.wikipedia.org/wiki/IPv4_address_exhaustion

[24] SixXS - *IPv6 Deployment & Tunnel Broker* [online, navštíveno 4.5.2008]
Dostupné na URL: http://www.sixxs.net/main/

[25] CentOS - *Community ENTerprise Operating System* [online, navštíveno 8.5.2008]
Dostupné na URL: http://www.centos.org/

[26] R. Hinden, S. Deering: *IPv6 Multicast Address Assignments*.
RFC 2375 (INFORMATIONAL), Červenec 1998.
Dostupné na URL: http://www.ietf.org/rfc/rfc2375

[27] C. Huitema: *An Anycast Prefix for 6to4 Relay Routers*.
RFC 3068 (PROPOSED STANDARD), Červen 2001.
Dostupné na URL: http://www.ietf.org/rfc/rfc3068

Seznam příloh

Příloha 1. CD s příklady konfiguračních souborů.

Obsah CD

- `./dhcpv6/dhcp6c.conf` – konfigurační soubor DHCPv6 klienta
- `./dhcpv6/dhcp6s.conf` – konfigurační soubor DHCPv6 serveru
- `./freebsd 6to4/rc.conf` – konfigurační soubor mechanismu 6to4 systému FreeBSD
- `./rtadvd/rtadvd.conf` – konfigurační soubor démona `rtadvd` pro bezstavovou automatickou konfiguraci
- `./teredo windows xp/teredo windows xp.conf` – příklad nastavení mechanismu Teredo u systému Windows XP
- `./doc/technická zpráva.pdf` – technická zpráva ve formátu `pdf`