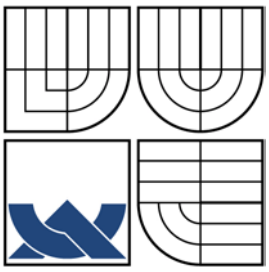
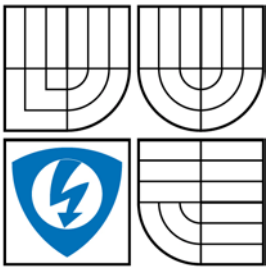


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

PROTOKOL IPV6 A MOŽNOSTI JEHO IMPLEMENTACE

PROTOCOL IPV6 AND ITS IMPLEMENTATION SCENARIOS

BAKALÁŘSKÁ PRÁCE
BACHELOR´S THESIS

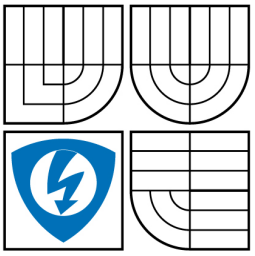
AUTOR PRÁCE
AUTHOR

ZDENĚK RÝZNER

VEDOUCÍ PRÁCE
SUPERVISOR

ING. PETRA LAMBERTOVÁ

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Zdeněk Rýzner ***ID:*** 78366
Ročník: 3 ***Akademický rok:*** 2008/2009

NÁZEV TÉMATU:

Protokol IPv6 a možnosti jeho implementace

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte a popište vlastnosti protokolu IPv6, porovnejte rozdíly mezi IPv6 a IPv4. Zaměřte se především na adresaci, strukturu hlavičky zabezpečení přenosu a možnost autokonfigurace. Popište základní metody přechodu z IPv4 na IPv6, vybrané metody realizujte a proveďte analýzu zasílaných paketů.

DOPORUČENÁ LITERATURA:

[1] SATRAPA, Pavel. IPv6 - Internet Protokol verze 6. Praha : Neocortex, 2002. 238 s. ISBN 80-86330-10-9.
[2] ŠMRHA, P. Internetworking pomocí TCP/IP. České Budějovice : KOPP, 1994. 125 s. ISBN 80-85828-09-X

Termín zadání: 9.2.2009 ***Termín odevzdání:*** 2.6.2009

Vedoucí práce: Ing. Petra Lambertová

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

Anotace

Tato práce se zabývá síťovým protokolem IPv6 a jeho pozicí v současném Internetu. Teoretická část se věnuje hlavním změnám, které nový protokol do oblasti síťové komunikace přináší – především adresování, formátu datagramů a automatické konfiguraci. Je zde též uvedeno porovnání IPv6 s IPv4. Praktická část zahrnuje připojení k IPv6-Internetu pomocí vybraných přechodových metod, analyzuje jednotlivé metody a uvádí možnosti jejich realizace v závislosti na síťovém prostředí.

Klíčová slova: internetový protokol, adresace, formát datagramu, autokonfigurace, IPsec, přechodové metody, 6to4, Teredo, Tunnel Broker

Abstract

This thesis deals with IPv6 – the Internet layer protocol and its situation in today's Internet. In theoretical part are described major changes, which the new protocol brings to networking area – especially addressing, headers format and autoconfiguration. Comparison of IPv6 and IPv4 is also included. Practical part covers connecting to IPv6-Internet with selected transition methods, analyses these methods and features its implementation in dependence on network environment.

Keywords: internet protocol, addressing, header format, autoconfiguration, IPsec, transition methods, 6to4, Teredo, Tunnel Broker

Bibliografická citace práce

RÝZNER, Z. *Protokol IPv6 a možnosti jeho implementace*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 59 s.
Vedoucí bakalářské práce Ing. Petra Lambertová.

Prohlášení

Prohlašuji, že svou bakalářskou práci na téma vlastnosti protokolu IPv6 a možnosti jeho implementace jsem vypracoval samostatně pod vedením vedoucího semestrálního projektu a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedeného semestrálního projektu dále prohlašuji, že v souvislosti s vytvořením tohoto projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne ……. ……

podpis autora

Poděkování

Rád bych poděkoval Ing. Petře Lambertové za odborné vedení, za pomoc a rady při zpracování této práce.

OBSAH

SEZNAM POUŽITÝCH ZKRATEK

6to4	Přechodová metoda založená na automatických tunelech
AES	Šifrovací standard (Advanced Encryption Standard)
AfriNIC	Regionální registrátor pro oblast Afriky (African Network Information Center)
AH	Autentizační hlavička (Authnetication Header)
APNIC	Regionální registrátor pro oblast Asie (Asia Pacific Network Information Centre)
ARIN	Regionální registrátor pro USA a Kanadu (American Registry for Internet Numbers)
B	Bajt
b	Bit
BIS	Jedna z translačních přechodových metod (Bump in the Stack)
CIDR	Mechanismus pro beztřídní adresování v IPv4 (Classsless Inter-Domain Routing)
CLNP	Síťový protokol vytvořený OSI (Connectionless Network Protocol)
DES	Šifrovací standard (Data Encryption Standard)
DHCP	Protokol pro automatickou konfiguraci stanic v síti (Dynamic Host Configuration Protocol)
DHCPv6	Protokol pro automatickou konfiguraci stanic v síti – verze 6 (Dynamic Host Configuration Protocol Version 6)
DNS	Systém doménových jmen (Domain Name System)
DSTM	Přechodová metoda využívající dvojí zásobník (Dual Stack Transition Mechanism)
DUID	Identifikátor DHCP (DHCP Unique Identifier)
ESP	Zabezpečení zapouzdření dat (Encapsulating Security Payload)
EUI-64	Rozšířený identifikátor rozhraní (Extended Unique Identifier)
HDTV	Formát TV vysílání (High-definition Television)
IA	Identifikační asociace (Identity Association)
IANA	Organizace řídící přidělování IP adres a další náležitosti týkající se internetových protokolů (Internet Assigned Numbers Authority)

ICMP	Protokol pro přenos řídících zpráv (Internet Control Message Protocol)
ICMPv6	Protokol pro přenos řídících zpráv verze 6 (Internet Control Message Protocol version 6)
ID	Identifikátor
IETF	Organizace zabezpečující rozvoj Internetu a tvorbu standardů (Internet Engineering Task Force)
IKE	Protokol pro výměnu klíčů v IPsec (Internet Key Exchange)
IKEv2	Protokol pro výměnu klíčů a správu bezpečnostních asociací v IPsec (Internet Key Exchange version 2)
IP	Síťový protokol využívaný pro provoz Internetu (Internet Protocol)
IPsec	Sada bezpečnostních prvků pro zabezpečení IP protokolu (IP security)
IPng	Další generace IP protokolu (IP next generation)
IPv4	IP protokol, verze 4
IPv6	IP protokol, verze 6
ISAKMP	Protokol pro správu bezpečnostních asociací (Internet Security Association and Key Management Protocol)
ISATAP	Jedna z tunelovacích přechodových metod (Intra-Site Automatic Tunnel Addressing Protocol)
ISP	Poskytovatel připojení k Internetu (Internet Service Provider)
LACNIC	Regionální registrátor pro oblast Jižní Ameriky (Latin American and Caribbean Internet Addresses Registry)
LIR	Místní registrátor (Local Internet Registry)
MAC	Identifikátor síťového rozhraní (Media Access Control)
MD5	Hashovací algoritmus, verze 5 (Message Digest Algorithm 5)
MTU	Maximální přenosová velikost paketů (Maximum Transmission Unit)
NAT	Metoda pro překlad síťových adres (Network Address Translation)
NAT64	Jedna z translačních přechodových metod
NAT-PT	Jedna z translačních přechodových metod (Network Address Translation - Protocol Translation)
NSAP	Přístupový bod síťové služby (Network Service Access Point)
OSI	Projekt pro standardizaci počítačových sítí a protokolů (Open Systems Interconnection)
QoS	Sada mechanismů pro řízení datových toků v síti (Quality of Service)

RFC	Označení pro standardy a dokumenty vydávané IETF (Request For Comments)
RIPE NCC	Regionální registrátor pro oblast Evropy (Réseaux IP Européens Network Coordination Centre)
RIR	Regionální registrátor (Regional Internet Registry)
RSA	Šifrovací algoritmus (Rivest, Shamir, Adleman Algorithm)
SA	Bezpečnostní asociace (Security Association)
SAD	Databáze bezpečnostních asociací (Security Association Database)
SASL	Rozhraní pro autentizaci a bezpečnost dat v internetových protokolech (Simple Authentication and Security Layer)
SIIT	Jedna z translačních přechodových metod (Stateless IP/ICMP Translation)
SOCKS64	Jedna z translačních přechodových metod
SPD	Databáze bezpečnostní politiky (Security Policy Database)
SPI	Identifikátor bezpečnostní asociace (Security Parameters Index)
TCP	Jeden z protokolů sady TCP/IP pracující na transportní vrstvě (Transmission Control Protocol)
TSP	Signalizační protokol pro vyjednání tunelu (Tunnel Setup Protocol)
TRT	Jedna z translačních přechodových metod (Transport Relay Translator)
TTL	Životnost paketu (Time To Live)
UDP	Jeden z protokolů sady TCP/IP pracující na transportní vrstvě (User Datagram Protocol)
XML	Značkovací jazyk pro výměnu dat mezi aplikacemi a pro publikování dokumentů (Extensible Markup Language)

ÚVOD

Současná verze síťového protokolu, používaného pro provoz Internetu – IPv4 – byla vyvinuta již v 70. letech minulého století. V té době nikdo nemohl předpokládat, k jak dynamickému rozšíření tohoto média dojde v následujících letech, kdy se z původně armádního projektu stala globální počítačová síť s počtem uživatelů, který dnes vysoce přesahuje miliardu.

Spolu s rychle rostoucím počtem uživatelů se však brzy objevil i problém vyčerpávání volného adresního prostoru. Proto se už v 90. letech začalo pracovat na nové verzi protokolu – IPv6. Protože byl vývoj zahájen s dostatečným předstihem, přinesla nová verze nejen úpravy nezbytné k dalšímu rozvoji Internetu, ale také klíčové změny týkající se některých oblastí v architektuře protokolu.

Cílem první části této práce je popis síťového protokolu IPv6 – především jeho klíčových vlastností a porovnání s předchozí verzí (IPv4) S tím souvisí druhá část práce, která se věnuje metodám používaným k zajištění funkčnosti nového protokolu s jeho předchůdcem – zabývá se principy, které se zde využívají, samotnou realizací připojení pomocí vybraných metod, které se dočkaly širšího nasazení v praxi a jsou zde zmíněny též výhody a nevýhody, které použití těchto metod přináší.

Práce by měla poskytnout pohled na tento protokol, jejíž praktické nasazení ve velkém měřítku je otázkou již velmi blízké budoucnosti a bez níž by pravděpodobně došlo ke kolapsu největší existující počítačové sítě – Internetu.

1 CHARAKTERISTICKÉ VLASTNOSTI IPV6 A POROVNÁNÍ S IPV4

Protokol IPv6 přináší změny v několika klíčových oblastech, jedná se zejména o: adresní prostor, formát záhlaví a volitelné záhlaví, vylepšenou podporu QoS, bezpečnostní mechanismy přímo v IP, podpora mobilních zařízení nebo autokonfiguraci.

1.1 Adresní prostor IPv6

Nejzřetelnějším rysem IPv6 je použití delších adres. Délka adresy IPv6 je 128 bitů, což je čtyřikrát více než má IPv4 adresa. Zatímco 32-bitový adresní prostor IPv4 může nabídnout maximálně 2^{32} neboli 4 294 967 296 možných adres, 128-bitový adresní prostor nabízí 2^{128} neboli 340 282 366 920 938 463 463 374 607 431 768 211 456 dostupných adres, což definitivním způsobem řeší jejich nedostatek v IPv4. S adresním prostorem IPv6 lze například každému člověku na planetě přidělit několik trilionů adres (každý člověk jich tak může mít víc, než existuje v celém IPv4 adresním prostoru), nebo lze na každém čtverečním metru zeměkoule využívat 665 570 793 348 866 943 898 599 adres [13].

1.2 Formát záhlaví IPv6

Záhlaví IPv6 se skládá z osmi polí (dvě z nich jsou adresa odesílatele a adresa příjemce) a má délku 40 bajtů. Jeho zjednodušení přináší výhody při směrování, které je efektivnější, pokud má záhlaví konstantní délku a obsahuje méně polí, která musí směrovač prozkoumat a zpracovat.

Za povinným záhlavím může následovat rozšiřující záhlaví. Rozšiřující záhlaví obsahuje volitelné možnosti týkající se směrování, šifrování obsahu, autentizace nebo fragmentace.

1.3 Podpora QoS

Podporu QoS zajišťují nová pole v záhlaví IPv6 – třída provozu (Traffic Class Field) a označení datového toku (Flow Label Field). Ty umožňují identifikovat tok dat a zpracovávat pakety patřící do toku, což umožní směrovačům použít na konkrétní pakety speciální zacházení, které si vyžádal zdrojový uzel.

1.4 Bezpečnostní mechanismy

Bezpečnost na úrovni IP vrstvy zajišťuje sada prvků označovaných jako IPsec. Implementace IPsec je povinná a dvě základní služby – autentizace a šifrování jsou realizovány v podobě rozšiřujících záhlaví. Jsou to:

Authentication Header (autentizace) – slouží k ověření totožnosti odesílatele a správnosti obsahu paketu.

Encapsulating Security Payload (zabezpečení zapouzdření dat) – nabízí primárně šifrování neseného paketu a dále služby ekvivalentní autentizaci.

1.5 Podpora mobilních zařízení

S rozmachem přenosných počítačů a komunikátorů umožňujících přístup k Internetu vyvstal problém, jak se spojit se zařízením, které během cesty mění svou IP adresu. IPv6 tento problém řeší tak, že pokud je mobilní uzel mimo „domov“, zastupuje jej domácí agent (typicky směrovač v domácí síti mobilního uzlu). Domácí agent na sebe přesměruje data určená pro mobilní uzel, zatímco ten průběžně informuje domácího agenta o své aktuální IP adrese.

1.6 Autokonfigurace

Pro větší flexibilitu a podporu mobility existuje v IPv6 vedle stavové konfigurace (pomocí protokolu DHCP) i konfigurace bezstavová, která umožňuje automaticky nakonfigurovat zařízení bez použití serveru.

1.7 Porovnání IPv6 a IPv4

Z porovnání obou verzí protokolů vyplývá řada rozdílů. Kvůli zachování určité struktury této práce zde uvádím tabulku porovnávající několik základních vlastností spolu s odkazy na příslušnou kapitolu v textu, kde je problematika podrobněji rozebrána. Porovnání celého záhlaví popisuje kapitola 2.3.

Tab. 1: Porovnání některých vlastností IPv4 a IPv6

	IPv4	**IPv6**
Délka adres (viz podkapitola 1.1)	32 bitů	128 bitů
Podpora IPsec (viz kapitola 5)	Volitelná	Povinná
Identifikace toku paketů za účelem využití QoS (viz podkapitola 2.2)	Neexistuje	Speciální pole v záhlaví pro označení datového toku (Flow Label Field)
Fragmentace (viz podkapitoly 2.3, 2.4)	Kterýkoliv uzel může fragmentovat pakety	Fragmentaci může provádět pouze odesílatel
Kontrolní součet v záhlaví (viz podkapitola 2.3)	Pole Header Checksum	Neexistuje
Volitelné možnosti (viz podkapitoly 2.3, 2.4)	Součást záhlaví	Existují ve formě rozšiřujících záhlaví
Všesměrové adresy (viz podkapitola 3.1)	Existují	Neexistují
Bezstavová autokonfigurace (viz podkapitola 4.3)	Neexistuje	Existuje
Dopředné DNS záznamy (viz podkapitola 3.7.1)	Používá A záznamy	Používá AAAA záznamy
Reverzní DNS záznamy (viz podkapitola 3.7.2)	PTR záznamy v *in-addr.arpa* doméně	PTR záznamy v *ip6.arpa* doméně

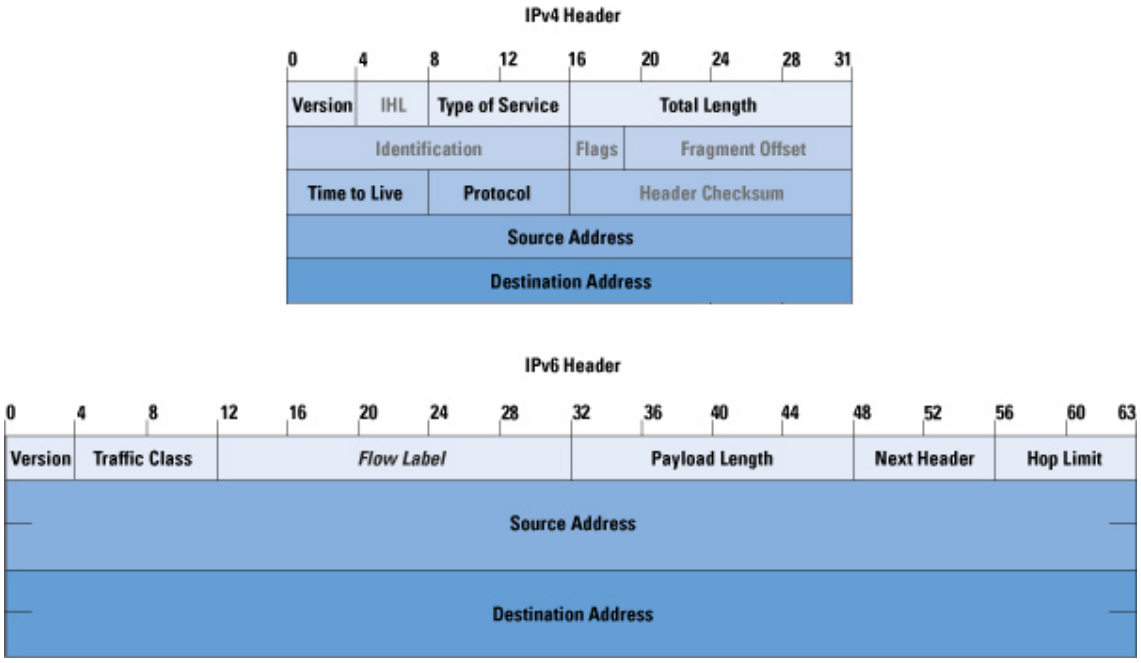
2 FORMÁT DATAGRAMU

2.1 Datagram

Formát datagramu pro IPv6 popisuje RFC 2460: Internet Protocol, Version 6 (IPv6) Specification. Toto RFC je tedy jakýmsi základním kamenem celého IPv6.

Tvar samotného datagramu je tvořen záhlavím (nebo záhlavími), za kterým(i) následují přenášená data. Už první pohled na schéma záhlaví IPv4 a IPv6 zobrazené na obr. 1, naznačuje, že IPv6 by měl být značně jednodušší protokol, protože jeho záhlaví obsahuje méně informací, než záhlaví jeho předchůdce. Důvodem zjednodušení bylo to, že některé informace v záhlaví IPv4 už nejsou nutné nebo žádoucí, ovšem zůstává zachována minimálně taková funkcionalita, jako tomu bylo u IPv4.

Záhlaví IPv6 tedy bylo omezeno pouze na nejnutnější prvky a byla mu dána konstantní velikost. Veškeré další údaje (například nepovinné a doplňující údaje) se přesunuly do rozšiřujících záhlaví, které se do datagramu vkládají podle potřeby. Tím došlo také k tomu, že ačkoliv délka adres se v IPv6 zvýšila čtyřikrát, délka základního záhlaví pouze dvakrát: IPv4 – 20B, IPv6 – 40B. Přitom celých 32B z těchto 40B zabírají v IPv6 adresy [16].



Obr. 1: Porovnání záhlaví IPv4 a IPv6
Zdroj: CISCO SYSTEMS. *The Internet Protocol Journal - Volume 9, Number 3* [online]. <http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_9-3/ipv6_internals.html>.

2.2 Popis polí záhlaví

Verze (Version) – čtyřbitová hodnota, která udává verzi protokolu, v případě IPv6 je tedy 6 [5].

Třída provozu/Priorita paketu (Traffic Class) – toto osmibitové pole, které je využíváno odesílajícím uzlem nebo mezilehlými směrovači, může nabývat různých hodnot, což umožňuje jednotlivé pakety rozlišit a přiřadit jim rozdílnou prioritu. Mezilehlé směrovače tak mohou identifikovat pakety patřící do stejné třídy provozu a rozlišovat mezi pakety s různými prioritami. Hodnoty tohoto pole nejsou v RFC 2460 specifikovány, prozatím je implicitně nastavena 0 a probíhají experimenty, které mají stanovit rozdělení datového provozu do tříd nejvýhodnějších pro IP pakety [5], [16].

Značka toku/Správa QoS (Flow Label) – další nové (a zatím experimentální) pole v záhlaví je dvacetibitová značka toku. Pakety s určitými společnými vlastnostmi označí zdrojový uzel jako tok, směrovač pak podle značky toku určí, které pakety patří ke stejnému toku a bude se všemi pakety jednoho toku nakládat stejným způsobem. Pakety, které nepatří k žádnému toku, mají toto pole nastavené na 0 [5], [16].

Délka dat (Payload Length) – šestnáctibitové číslo představující délku dat - základní záhlaví se mezi tyto data nepočítá, případná rozšiřující záhlaví jsou zde započtena (považují se za součást dat) [5].

Další záhlaví (Next Header) – osmibitový identifikátor toho, co následuje za záhlavím. Specifikuje se zde buď typ následujícího rozšiřujícího záhlaví, nebo druh dat [5].

Maximální počet skoků (Hop Limit) – osmibitová hodnota, která se snižuje o 1 pokaždé, když uzel pošle paket dál sítí. Pokud hodnota dosáhne 0, je paket zahozen. Smyslem položky je ochrana proti zacyklení při směrování, jedná se o náhradu životnosti paketu (Time-To-Live – TTL), známé z IPv4 [5].

Zdrojová adresa (Source Address) – 128-bitová adresa odesílatele paketu.

Cílová adresa (Destination Address) – 128-bitová adresa příjemce paketu. Pokud je použito rozšiřující směrovací záhlaví (Routing Header), nejedná se o adresu konečného příjemce.

2.3 IPv4 versus IPv6

Z obr. 1 je patrné, že záhlaví obou verzí protokolů jsou si podobné, ale pouze jedno pole – verze, zůstalo v IPv6 beze změny (toto pole muselo zůstat beze změny, aby mohlo na jedné lokální lince fungovat jak IPv4 tak IPv6). Ostatní pole z IPv4 se liší následovně:

Délka záhlaví (Header Length, IHL) – pro IPv6 nepodstatná informace, protože používá záhlaví se stejnou délkou. Verze 4 potřebovala toto pole, protože záhlaví mohlo nabývat délky 20B – 60B v závislosti na použitých volbách [14].

Typ služby (Type of Service) – v IPv4 sloužilo toto pole pro QoS, tedy pro určení, jak má směrovač zacházet s pakety, u kterých bylo QoS vyžadované. Ekvivalentem tohoto pole je v IPv6 pole třída provozu [14].

Délka datagramu (Datagram Length) – z tohoto pole se v IPv6 stala délka dat. Rozdíl spočívá v tom, že délka datagramu v IPv4 uváděla délku včetně záhlaví [14].

Identifikace (Datagram Identification) – se v IPv4 používala k určení datagramu, který byl součástí původního fragmentovaného paketu. Protože IPv6 nepovoluje fragmentaci mezilehlým uzlům („po cestě“), stalo se toto pole v IPv6 zbytečným [14].

Pole Flags a posun fragmentu (Fragment Offset) – též souvisely s fragmentací v IPv4 a ve verzi 6 jsou zbytečné [14].

Životnost (TTL) – pole životnosti se v IPv6 přeměnilo na maximální počet přeskoků. TTL slouží jako ochrana proti zacyklení paketů. Podle původní specifikace TTL měl směrovač snižovat hodnotu TTL o dobu, kterou paket ve směrovači strávil, v praxi však většina směrovačů jednoduše snižuje tuto hodnotu o 1 [14].

Protokol (Protocol) – udává druh dat (typ protokolu) následující vyšší vrstvy, který je zapouzdřen v daném IPv4 paketu. Toto pole se v IPv6 přeměnilo v pole další záhlaví [14].

Kontrolní součet (Header Checksum) – kontrolní součet slouží ke zjištění, zda se během cesty nezměnila (nepoškodila) záhlaví. Vzhledem k tomu, že kontrolní součty provádí i protokoly vyšších vrstev (např. TCP, UDP), bylo toto pole z IPv6 odstraněno. Tím nedochází ke zpomalování, které vznikalo kontrolou a přepočítáváním kontrolního součtu v uzlech [14].

Zdrojová/cílová adresa (Source/Destination Address) – došlo k rozšíření z 32 na 128 bitů.

Volby (IP Options) – místo pole volby IPv6 používá samostatná rozšiřující záhlaví, které se umísťují mezi základní záhlaví a nesená data [14].

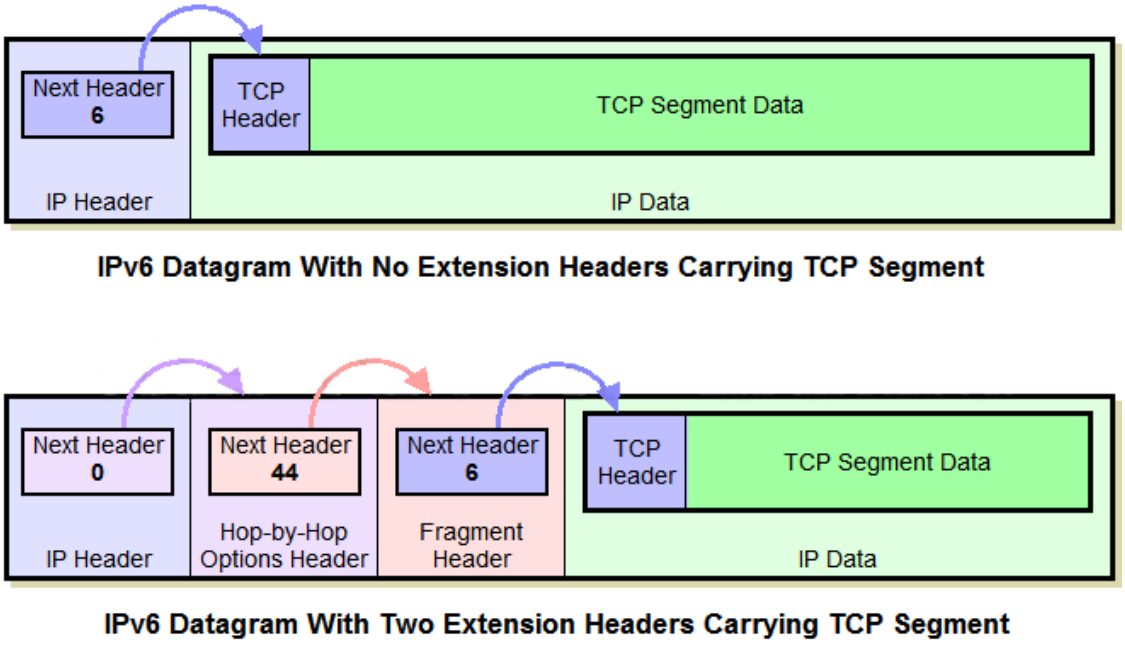
2.4 Rozšiřující záhlaví

Volitelné informace internetové vrstvy jsou zakódované v samostatných záhlavích, které se umisťují mezi základní záhlaví a nesená data. Každé záhlaví je samostatným blokem. Tyto bloky jsou vzájemně propojeny pomocí pole další záhlaví (Next Header) v předcházejícím záhlaví. Jednotlivá záhlaví jsou identifikována podle hodnoty v tomto poli. Některé z těchto hodnot jsou uvedeny v tab. 2, řetězení záhlaví je uvedeno na obr. 2. Paket IPv6 může mít jedno nebo více rozšiřujících záhlaví, ale nemusí mít také žádné. Pokud nemá žádné, je v poli další záhlaví uveden identifikátor nesených dat.

Tab. 2: Vybrané hodnoty pole další záhlaví

Typ záhlaví	Hodnota
Volby pro všechny (Hop-by-hop Options)	00
Směrování (Routing)	43
Fragmentace (Fragment)	44
Autentizace (Authentication)	51
Volby pro cíl (Destination Options)	60
Šifrování obsahu (Encapsulating Security Payload)	50
Poslední hlavička (No Next Header)	59
Hodnoty pro protokoly vyšších vrstev	06 (TCP), 17 (UDP)

Zdroj: IANA. *Assigned Internet Protocol Numbers* [online]. <http://www.iana.org/assignments/protocol-numbers>.



Obr. 2: Příklad řazení rozšiřujících záhlaví
Zdroj: THE TCP/IP GUIDE. *IPv6 Datagram Extension Headers* [online]. <http://www.tcpipguide.com/free/t_IPv6DatagramExtensionHeaders-2.htm>.

Největší výhodou rozšiřujících záhlaví je pružnost a úspornost. Součástí datagramu jsou pouze ty informace, které jsou potřeba pro správné doručení. Rozšiřující záhlaví musí být zpracovávána přesně v tom pořadí, v jakém jsou v datagramu přítomná, aby se předešlo tomu, že uzel bude zpracovávat záhlaví, které mu není určeno a tím brzdit rychlost zpracovávání celého paketu [16].

Doporučené pořadí rozšiřujících záhlaví a jejich krátký popis:

- *Volby pro všechny (Hop-by-hop Options Header)* – Parametry pro všechny mezilehlé uzly
- *Volby pro cíl (Destination Options Header)* – Parametry pouze pro cílový uzel
- *Směrování (Routing Header)* – Doplňkové informace pro směrování paketu, zejména druh směrování, seznam adres uzlů, kterými by měl paket po cestě projít a ukazatel na ještě nenavštívené uzly.
- *Fragmentace (Fragment Header)* – Fragmentaci používá (na rozdíl od IPv4) výlučně zdrojový uzel, pokud je potřeba přenést větší data, než dovoluje maximální přenosová velikost paketu (MTU). Obsahuje identifikaci původního neděleného paketu, pořadí fragmentu a označení, zda se jedná o poslední paket.
- *Autentizace (Authentication Header)* – Obsahuje kontrolní informace o tom, zda nebyla porušena integrita.

- *Šifrování obsahu (Encapsulating Security Payload Header)* – obsahuje parametry týkající se šifrování a pozdějšího dešifrování dat.
- *Volby pro cíl (Destination Options Header)* – viz výše.

Každé záhlaví by se mělo v paketu objevit maximálně jednou. Výjimku tvoří volby pro cíl, které se vyskytují dvakrát – poprvé před směrováním a podruhé před nesenými daty.

Mezilehlý směrovač (směrovač „po cestě“ paketu) zajímají jen volby pro všechny, pokud narazí za základním záhlavím na kód jiného rozšiřujícího záhlaví, tak s analýzou paketu skončí.

Ostatní rozšiřující záhlaví mají význam pouze pro cílový uzel. Je ale potřeba rozlišovat průběžný cílový uzel a koncový cílový uzel. Průběžný cílový uzel je uveden v rozšiřujícím směrovacím záhlaví a jedná se o uzel, kterým by měl paket projít. Průběžný cílový uzel proto budou zajímat první tři rozšiřující záhlaví – volby pro všechny, volby pro cíl a směrování. Koncový cílový uzel, kterému jsou data určena, budou zajímat všechny rozšiřující záhlaví.

3 ADRESY V IPV6

Ačkoliv se na jednu stranu může zvětšení adresního prostoru a délky adres jevit jako nejmarkantnější změna u IPv6, může to být na druhou stranu změna nejméně zřejmá, protože délka adres bude pro většinu uživatelů neviditelná. Kvůli délce (a složitosti) adres je totiž nezbytně nutné spolehnout se na systém doménových jmen (Domain Name System), který přiřazuje adresám jména.

Adresní schéma IPv6 bylo poprvé specifikováno v RFC 1884 IP Version 6 Addressing Architecture, kde byly specifikovány tři typy adres – unicast, multicast a anycast. Tato specifikace byla poté dvakrát upravena, poprvé v RFC 2373 a podruhé v RFC 3513 [14].

3.1 Druhy adres IPv6

Jak již bylo uvedeno, existují tři typy adres:

- *Individuální (Unicast)* – identifikátor jednoho rozhraní. Paket odeslaný na individuální adresu je doručen na rozhraní, které se identifikuje touto adresou.
- *Skupinové (Multicast)* – identifikátor skupiny rozhraní (zpravidla patřící různým uzlům). Paket odeslaný na skupinovou adresu je doručen na všechna rozhraní, která se touto adresou identifikují.
- *Výběrové (Anycast)* – identifikátor skupiny rozhraní (zpravidla patřící různým uzlům). Paket odeslaný na výběrovou adresu je doručen jednomu rozhraní – tomu, které je nejblíže (vzdálenost určuje metrika použitého směrovacího protokolu).

Všesměrové (Broadcast) adresy byly zrušeny, jejich funkce převzaly skupinové a výběrové adresy. U IPv6 se všesměrového zasílání paketů dosahuje pomocí speciálních multicast skupin, jako například všechny uzly na dané lince, nebo všechny směrovače na dané lokální lince, což je daleko efektivnější než broadcast z IPv4 [4].

3.2 Typy adres IPv6

Adresní prostor IPv6 byl rozdělen na několik skupin – typů adres. Každý typ sdružuje adresy se společnou charakteristikou. Z obrovského adresního prostoru bylo takto definováno zatím jen 15%.

Tab. 3: Základní rozvržení adres

IPv6 prefix	Význam
::/128	Nespecifikovaná adresa
::1/128	Lokální smyčka (Loopback)
FF00::/8	Skupinové adresy
FE80::/10	Individuální lokální linkové adresy
FC00::/7	Unikátní individuální lokální adresy
Ostatní	Individuální globální adresy

Zdroj: SATRAPA, P. *IPv6 – Internetový protocol verze 6*. Praha : CZ.NIC, 2008. s. 54

Nejrozšířenější adresy – individuální, se dělí na několik typů:

- *Individuální globální (Global Unicast)* – jsou globálně jedinečné, takže mohou být směrovány celým IPv6 Internetem.
- *Individuální lokální linkové (Link Local Unicast)* – tyto adresy se používají pro adresaci jedné linky pomocí bezstavové autokonfigurace, objevování sousedů (Neighbor Discovery) nebo v situacích, kdy není přítomen směrovač. Paket odeslaný na takovou adresu by nikdy neměl být směrován mimo lokální linku.
- *Unikátní individuální lokální (Unique Local Unicast)* – jedná se o globálně unikátní adresy, které ale slouží pro lokální komunikaci, většinou uvnitř nějakého místa (site), nelze je použít v globálním Internetu. Jde o náhradu individuálních lokálních místních adres (Site Local Unicast), které byly formálně zamítnuty v RFC 3879 v září 2004 [7].
- *Individuální adresy obsahující IPv4 adresy nebo zakódované NSAP adresy* – adresy zajišťující funkčnost mezi IPv6 a jinými protokoly síťové vrstvy – např. IPv4 a Connectionless Network Protokol (CLNP) vytvořený OSI.

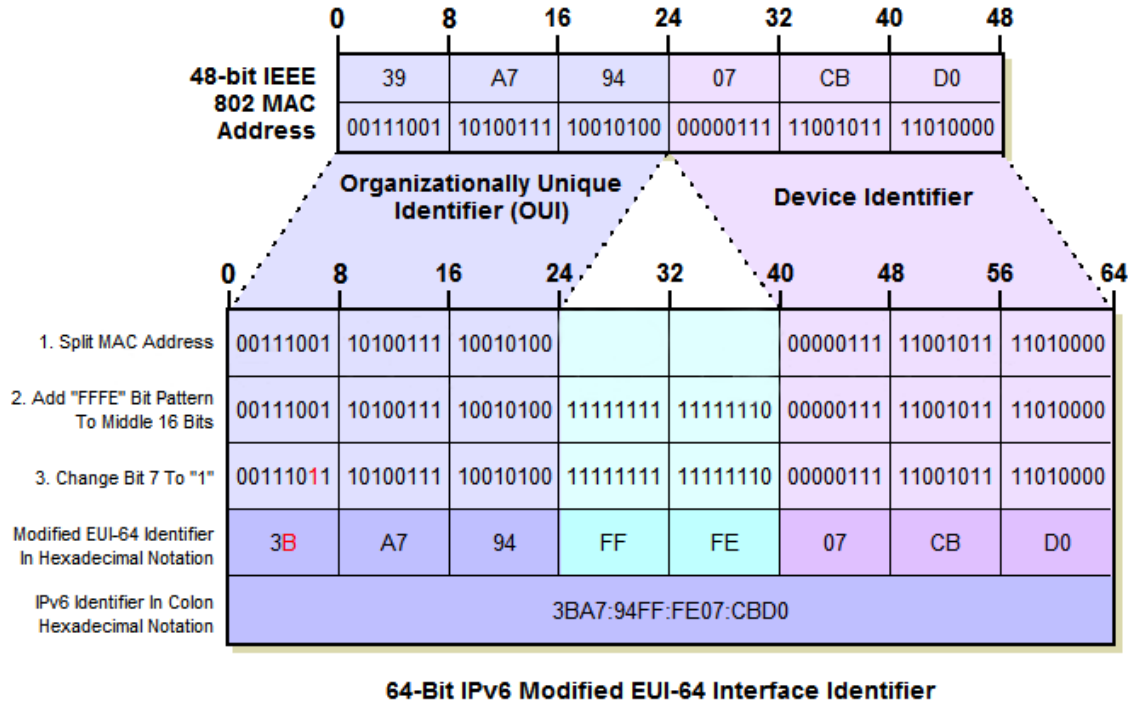
3.3 Adresa sítě a adresa uzlu

IPv6 adresy jsou zpravidla rozděleny na dvě části, kde horních 64 bitů identifikuje síť a dolních 64 bitů představuje identifikátor rozhraní – modifikované EUI-64.

Ve standardním EUI-64 je předposlední bit v prvním bajtu příznakem globality, hodnota 0 značí celosvětově jedinečnou adresu, hodnota 1 značí adresu lokální. Modifikované EUI-64 vznikne inverzí hodnoty tohoto bitu, což usnadňuje tvorbu identifikátorů.

Nejběžněji používané ethernetové rozhraní má výrobcem přidělené celosvětově jednoznačné MAC adresy o délce jen 48 bitů. Transformace na modifikované EUI-64 se

proto provádí tak, že mezi třetí a čtvrtý bajt MAC adresy se vloží 16 bitů s hodnotou FFFE a obrátí se příznak globality [12]. Viz obr. 3.



Obr. 3: Vytvoření modifikovaného EUI-64 z MAC adresy
Zdroj: THE TCP/IP GUIDE. *IPv6 Interface Identifiers and Physical Address Mapping* [online]. <http://www.tcpipguide.com/free/t_IPv6InterfaceIdentifiersandPhysicalAddressMapping-2.htm>.

Zde může vyvstat problém, týkající se soukromí uživatele. MAC adresa je jedinečná a nemění se ani v případě, že se počítač pohybuje, což umožňuje jej v síti jednoznačně identifikovat – zjistit, kde se pohybuje, nebo s kým komunikuje [16]. V této situaci nepomůže ani šifrování, protože se šifruje pouze datová část paketu a ne jeho záhlaví.

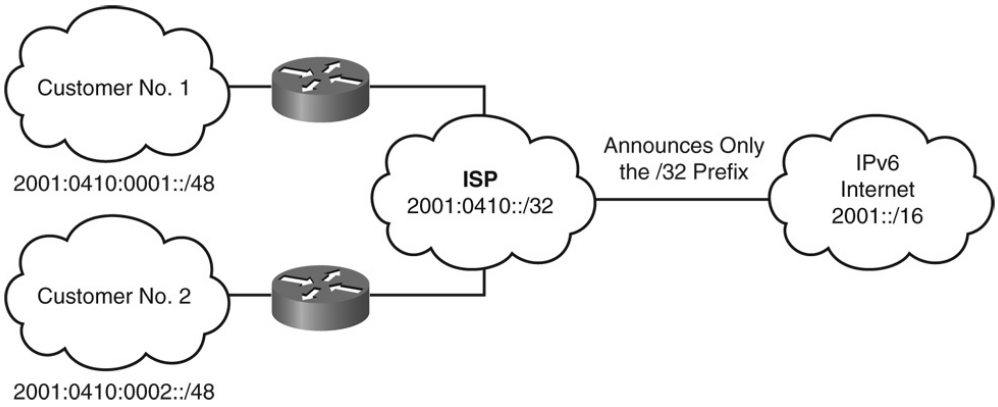
3.4 Agregace adres

Agregace adres umožňuje slučovat adresy do skupin, reprezentovaných vždy jedinou adresou sítě, jejichž velikost závisí na tom, z jakého místa v síti se na ně nahlíží. To značným způsobem snižuje počet záznamů, které musí ve svých tabulkách udržovat páteřní směrovače.

Situace je znázorněna na obr. 4: Směrování je hierarchické, proto směrovačům v síti IPv6 Internet stačí znát pouze kratší /32 prefix do sítě ISP a nemusí mít ve svých tabulkách speciální záznam jak pro síť zákazníka č. 1, tak pro zákazníka č. 2. Teprve

směrovače v síti ISP budou zkoumat další bity adresy, aby určily, kterému zákazníkovi data poslat.

Směrovače ve větší vzdálenosti od cíle tedy na adresy nahlížejí obecněji (zkoumají méně bitů adresy, nepotřebují znát přesnou cestu k cíli, ale pouze cestu na další mezilehlý směrovač) a jak se paket blíží k cíli, počet bitů adresy, které směrovač musí prozkoumat, se zvyšuje.



Obr. 4: Princip agregace adres
Zdroj: TEARE, D. - PAQUET, C. *CCNP Self-Study: Advanced IP Addresing*. Cisco Press, 2004. s. 49

3.5 Zápis IPv6 adres

Ačkoliv v RFC 3513 existují tři způsoby, jak zapisovat IPv6 adresy, preferovaným způsobem je zápis IPv6 adresy jako osmi skupin po čtyřech číslicích šestnáctkové soustavy, které se oddělují dvojtečkami. Každá čtveřice číslic tak představuje 16 bitů.

Vzhledem ke způsobu alokace IPv6 bude častým jevem, že adresy budou obsahovat dlouhé řetězce nul. Proto byl zaveden speciální zástupný znak „::“, který nahrazuje několik nulových šestnáctibitových skupin za sebou. Tento zástupný znak se smí objevit v adrese pouze jedenkrát, aby byla zachována její jednoznačnost. Příklad tohoto zápisu uvádí první řádek tab. 4.

Další možností zkrácení IPv6 adresy je vynechání počátečních nul v každé čtveřici. V zápise adresy se proto může objevit místo „0000“ pouze „0“. Viz druhý řádek tab. 4.

Pokud jsou v některých situacích IPv4 adresy zapouzdřeny v IPv6 adresách (jedná se o takzvané IPv4-kompatibilní IPv6 adresy a IPv4-překládané IPv6 adresy), můžou být poslední čtyři bajty adresy nahrazeny klasickým zápisem IPv4 adresy, zatímco

počátek adresy je zapsán jako normální IPv6 adresa. Jak výsledný zápis vypadá, uvádí čtvrtý řádek tab. 4.

Tab. 4: Způsob zápisu IPv6 adres

Adresa	Standardní zápis	Zkrácený zápis
A	1080:0000:0000:0000:0008:0800:200C:417A	1080::8:800:200C:417A
B	FF01:0000:0000:0000:0000:0000:0000:0101	FF01:0:0:0:0:0:0:101
C	0000:0000:0000:0000:0000:0000:0000:0001	::1
D	0000:0000:0000:0000:0000:0000:0000:0000	::
E	0000:0000:0000:0000:0000:0000:147.230.49.73	::147.230.49.73

Poznámka:

adresa A = individuální

B = skupinová

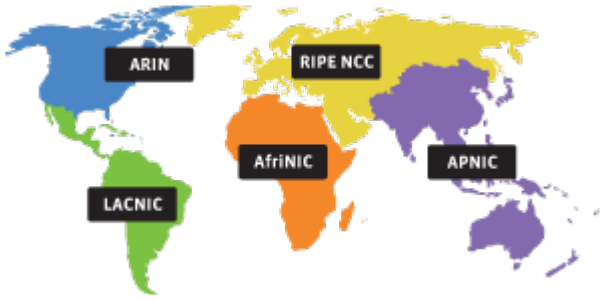
C = lokální smyčka

D = nespecifikovaná

E = IPv4-kompatibilní

3.6 Přidělování adres

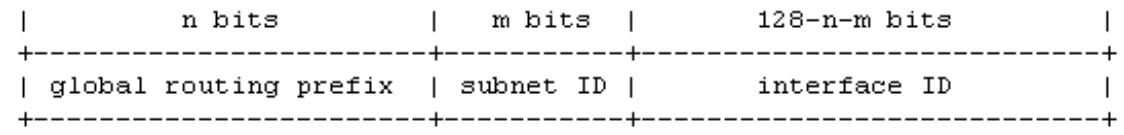
Mechanismus přidělování adres připomíná způsob z IPv4 – svět je rozdělen na regiony, které obhospodařují regionální registrátoři (Regional Internet Registry, RIR). Konkrétně ARIN (oblast Severní Ameriky), LACNIC (oblast Jižní Ameriky), AfriNIC (oblast Afriky), RIPE NCC (oblast Evropy a Ruska) a APNIC (pro oblast Asie a Austrálie) – viz obr. 5. Ti přidělují adresní rozsahy lokálním registrátorům (Local Internet Registry, LIR), což často bývají poskytovatelé Internetu. Od nich získávají adresy koncové instituce – zákazníci. Vzhledem k hierarchickému uspořádání přidělovaných rozsahů je zajištěna agregovatelnost adres [16].



Obr. 5: Pole působnosti jednotlivých regionálních registrátorů (RIR)
Zdroj: IANA. *Number Resources* [online]. <http://www.iana.org/numbers>.

Původní návrh struktury globálních individuálních adres, popsaný v RFC 2374, rozděloval horních 64 bitů na několik identifikátorů, představující úroveň agregace.

Praxe ovšem ukázala, že to nebylo ideální řešení, proto bylo RFC 2374 nahrazeno RFC 4291, které definuje strukturu globálních individuálních adres tak, jak je uvedeno na obr. 6.



Obr. 6: Struktura globálních individuálních adres
Zdroj: HINDEN, R. - S. DEERING. *IP Version 6 Addressing Architecture.* RFC 4291, 2006. s. 8

Global routing prefix představuje hierarchicky strukturovanou hodnotu přiřazenou určitému místu, *subnet ID* je identifikátorem jedné linky v tomto místě, a *interface ID* je identifikátor konkrétního rozhraní [6].

3.7 DNS

3.7.1 Dopředné dotazy

Pro převod jména na IPv6 adresu slouží záznamy typu AAAA. Název byl odvozen od A záznamů, které obstarávají tentýž úkol pro IPv4. Jelikož je délka IPv6 adresy čtyřnásobná, bylo A v názvu záznamu také čtyřikrát zopakováno. Použití v zónovém souboru je obvyklé:

jméno AAAA *IPv6_adresa*

například

```
pc AAAA 2001:db8:1c01:1:204:76ff:fe47:8e81
```

3.7.2 Reverzní dotazy

Zatímco v adrese je nejobecnější část na začátku (adresa sítě, pak podsíť a až na závěr adresa počítače v podsíti), u domén je tomu přesně naopak. Obecné domény (např. cz) jsou na konci a směrem dopředu se konkretizují.

Aby se dala databáze reverzního DNS distribuovat obvyklým způsobem, je třeba adresu v dotazu obrátit. U IPv6 se reverzní dotaz z IPv6 adresy vytvoří tak, že se obrátí pořadí šestnáctkových číslic v celé adrese (nesmí se vynechávat žádné nuly) a každá z nich je brána jako poddoména. Na konec se pak připojí ip6.arpa.

Dotaz zjišťující doménové jméno k IPv6 adrese *2001:db8:1c01:1:204:76ff:fe47:8e81* by vypadal takto:

```
1.8.e.8.7.4.e.f.f.f.6.7.4.0.2.0.1.0.0.0.1.0.c.1.8.b.d.0.1.0.0.2.ip6.arpa
```

Do zónového souboru se zapisují prostřednictvím obvyklých PTR záznamů. Pro výše uvedenou adresu by instituce dostala přidělen prefix 2001:db8:1c01::/48 a jemu odpovídající reverzní doménu *1.0.c.1.8.b.d.0.1.0.0.2.ip6.arpa*. V jejím zónovém souboru by pak bylo uvedeno:

```
1.8.e.8.7.4.e.f.f.f.6.7.4.0.2.0.1.0.0.0 PTR pc.kdesi.cz
```
[10].

4 AUTOMATICKÁ KONFIGURACE

4.1 Možnosti automatické konfigurace

Jednou z dalších novinek IPv6 je automatická bezstavová konfigurace – jakési síťové plug and play. IPv6 ovšem nabízí dva způsoby automatické konfigurace.

Stavová konfigurace je známá už řadu let. V tomto případě využívá nového protokolu DHCPv6. Základní mechanismus spočívá v tom, že počítač rozešle paket s dotazem ohledně svých komunikačních parametrů. Jakmile tento paket dorazí k DHCP serveru, ten mu odpoví. Z odpovědi se žadatel dozví potřebné informace jako IP adresu, prefix sítě, adresu DNS serveru a podobně.

Bezstavová konfigurace představuje zcela nový mechanismus, jde o jeden z prvků pro podporu mobility. Základní myšlenkou je, že si stanice vygeneruje dočasnou adresu, kterou používá do doby, než zjistí informace o síti, ve které se nachází. Na základě těchto informací si poté vytvoří stálou adresu. Metoda je založena na přijímání speciálních paketů – tzv. oznámení směrovače (Router Advertisement), což je paket protokolu ICMP. Stanici, která vstupuje do sítě, tak stačí pouze určitou dobu naslouchat nebo si oznámení vyžádat a dozví se příslušné informace o síti [4].

4.2 Stavová autokonfigurace

Na DHCP se podílejí tři kategorie zařízení:

- *DHCP klient* – stanice vyžadující konfiguraci,
- *DHCP server* – poskytovatel konfiguračních informací,
- *DHCP agent (DHCP Relay, DHCP Agent)* – zařízení zprostředkovávající komunikaci mezi klientem a serverem, pokud se nacházejí na různých linkách.

Pro identifikaci zařízení a správu adres se využívá dvou identifikátorů – DUID a IA. DUID (DHCP Unique Identifier) jednoznačně identifikuje zařízení v DHCP procesu. IA (Identity Association) identifikuje jednotlivá rozhraní u klienta a obsahuje konfigurační informace přidělené jednotlivým rozhraním.

Počáteční zprávy už nejsou zasílány na všesměrovou adresu jako u IPv4, ale jsou definovány dvě lokální linkové adresy:

- *FF02::1:2* všichni DHCP agenti a servery (na tuto adresu zasílá zprávy klient)

- *FF05::1:3* všechny DHCP servery (tuto adresu může využít agent při posílání zpráv serverům)

4.2.1 Komunikace klient-server

Pokud se server a klient nachází na stejné lince, probíhá komunikace přímo mezi nimi. Komunikaci zahajuje klient posláním zprávy výzva (Solicit), která obsahuje DUID, všechny IA a lokální linkovou adresu, kterou si klient přidělil. Server, nebo více serverů na zprávu odpoví zprávou ohlášení serveru (Advertise), kde uvádí hodnotu preference a parametry, které by přidělil jednotlivým IA. Klient si ze všech přijatých ohlášení vytvoří seznam DHCP serverů, z nichž si podle hodnoty preference vybere ten nejvýhodnější. Klient odešle zprávu žádost (Request), ve které uvádí DUID serveru, pro který je zpráva určena (zprávu totiž odesílá opět na obecnou adresu všech DHCP agentů). Server, pro který je žádost určena, zašle zpět zprávu odpověď (Reply), obsahující IPv6 adresu a konfigurační parametry požadované klientem. Klient si ověří přidělené adresy pomocí mechanismu pro detekci duplicitních adres a pokud zjistí, že dané adresy někdo používá, odmítne je pomocí zprávy odmítnutí (Decline) [14].

4.2.2 Komunikace klient-agent-server

Pokud se klient a server nenachází na stejné lince, probíhá komunikace obdobným způsobem s tím rozdílem, že zprávy přicházející od klienta zabalí DHCP agent do zpráv předání (Relay Forward), které odesílá serveru. V opačném směru server posílá zprávy zprostředkovaná odpověď (Relay Reply), které jsou agentem zabaleny do „běžných zpráv“ (Advertise nebo Reply) a pak předány klientovi.

Přidělené adresy mají omezenou životnost. Po jejím uplynutí klient žádá o prodloužení – posílá zprávu obnovení (Renew) serveru, který mu adresu přidělil. Pokud tento server neodpovídá, obrátí se klient na všechny servery se zprávou převázání (Rebind). Pokud klient opouští síť, měl by o tom informovat server pomocí zprávy uvolnění (Release), čímž umožní, aby jeho adresa byla poskytnuta dalšímu zájemci. Pokud se počítač vrací do sítě například po restartu, ověřuje správnost svých parametrů pomocí zprávy potvrzení (Confirm), ve které uvádí parametry svých IA. Server klientovi odpoví, zda jsou parametry platné či nikoliv. Pokud by došlo ke změně síťových parametrů a server si potřebuje vynutit na klientech, aby se přizpůsobili nové situaci, rozesílá zprávu rekonfigurace (Reconfigure), v ostatních případech typicky zahajuje komunikaci klient [14].

Tab. 5: Přehled typů zpráv DHCPv6

Typ zprávy	Numerické označení
Výzva (Solicit)	1
Ohlášení serveru (Advertise)	2
Žádost (Request)	3
Potvrzení (Confirm)	4
Obnovení (Renew)	5
Převázání (Rebind)	6
Odpověď (Reply)	7
Uvolnění (Release)	8
Odmítnutí (Decline)	9
Rekonfigurace (Reconfigure)	10
Žádost o informace (Information Request)	11
Předání (Relay Forward)	12
Zprostředkovaná odpověď (Relay Reply)	13

Zdroj: SATRAPA, P. *IPv6 – Internetový protokol verze 6*. Praha : CZ.NIC, 2008. s. 122

Přestože se DHCPv6 již od svého vzniku potýká s problémy se zabezpečením, s postupným odstraňováním těchto problémů se pravděpodobně stane dominantním přístupem k automatické konfiguraci (v těch případech, kdy je k dispozici DHCP server), především proto, že bezstavová autokonfigurace nabízí pouze omezené množství informací o síti, ve které se klient nachází.

4.3 Bezstavová autokonfigurace

Proces bezstavové autokonfigurace tvoří tři základní kroky:

1. Vytvoření lokální linkové adresy pro uzel, který se bude konfigurovat
2. Ověření jedinečnosti adresy vytvořené v předchozím kroku
3. Určení, které informace je potřeba automaticky konfigurovat a jak tyto informace získat

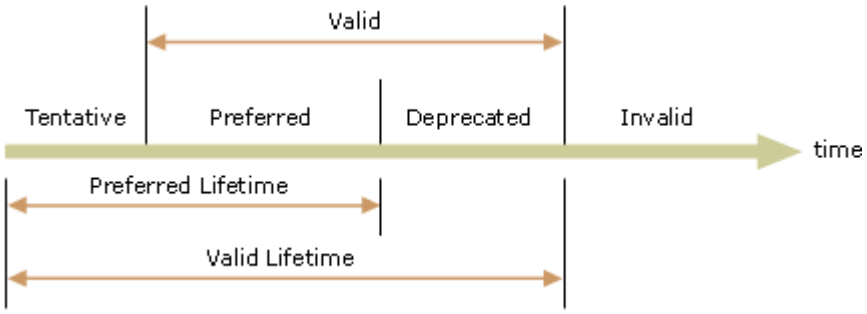
Vytvoření lokální linkové adresy se provede tak, že se ke standardnímu prefixu lokálních linkových adres FE80:: připojí identifikátor rozhraní. Identifikátor rozhraní se odvodí jednoznačně z fyzické adresy (MAC) rozhraní.

Pro ověření jedinečnosti lokální linkové adresy uzel rozešle sousedům výzvu, ve které hledá vlastníka adresy, kterou sám sobě vygeneroval. Pokud by od některého sousedního uzlu dostal ohlášení, znamená to, že daná adresa už existuje a autokonfigurace by nemohla pokračovat dál.

To však ve většině případů nenastane. Pro pokračování automatické konfigurace uzel potřebuje informace o svém okolí, které se dozví z oznámení směrovače (Router

Advertisment). Oznámení směrovače posílá každý směrovač do všech sítí, ke kterým je připojen. Po obdržení oznámení již uzel ví, v jaké síti se nachází, jak se zde komunikuje a kdo je implicitní směrovač a také informaci, zda má použít bezstavovou či stavovou konfiguraci.

V paketu existují mimo jiné také dva časové údaje (doba platnosti a doba preferování), které stanovují dobu trvání jednotlivých fází života adres vytvořených bezstavovou autokonfigurací. Stavy adresy v průběhu času zachycuje obr. 7. Po vzniku má adresa stav „preferována“ (Preferred) – adresa je platná a lze ji používat podle libosti. Po vypršení nastavené doby preferování se adresa dostane do stavu „odmítání“ (Deprecated) – adresa zůstává platná, ale uzel by ji pokud možno již neměl používat. Po uplynutí doby platnosti se stav adresy změní na „neplatná“ (Invalid) a uzel by ji již neměl používat (a odstranit z odpovídajícího rozhraní) [4], [16].

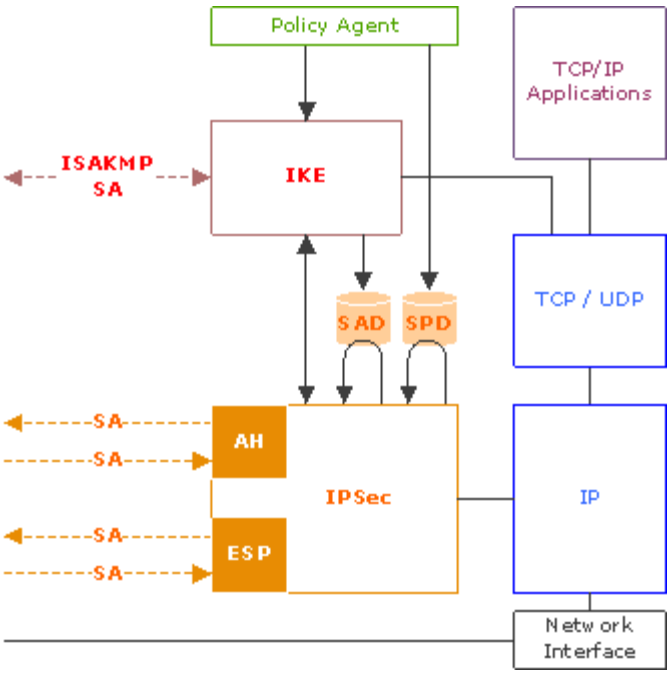


Obr. 7: Autokonfigurace – stavy adresy
Zdroj: DAVIES, J. *Understanding IPv6*. Microsoft Press, 2003. s. 217

Největším kladem autokonfigurace je její jednoduchost – po zapojení do sítě zařízení prostě funguje (autokonfigurace je implicitně zapnutá v operačních systémech). Problém představuje jednak získání informací z DNS a pak také možnost připojovat do sítě množství zařízení bez možnosti efektivní kontroly nebo evidence správcem sítě.

5 IPSEC

Bezpečnost na úrovni IP (na síťové vrstvě) zajišťuje sada mechanismů, souhrnně nazývaných IPsec. IPsec existuje jak pro IPv4, tak pro IPv6, jeho implementace je však povinná až v IPv6. Právě díky povinné implementaci v IPv6 existuje standardizované řešení zabezpečení a je také zajištěna součinnost mezi různými typy (způsoby) implementace IPv6. IPsec využívá dvou typů rozšiřujících záhlaví a protokolu, který slouží k vyjednávání bezpečnostních nastavení. Prvním rozšiřujícím záhlavím je Authentication Header (AH), které zajišťuje integritu dat, autentizaci a ochranu proti opakování pro celý IPv6 paket (mimo ty pole v záhlaví, které se během provozu musí měnit). Druhé rozšiřující záhlaví - Encapsulating Security Payload (ESP) zajišťuje integritu dat, autentizaci, utajení a ochranu proti opakování pro data, která jsou „zabalena“ v ESP. Protokol, který se běžně používá pro vyjednání bezpečnostních nastavení IPSec pro komunikaci je Internet Key Exchange (IKE) protokol.



Obr. 8: Schéma architektury IPsec
Zdroj: TECH-INVITE. *IPsec Guides* [online]. <http://www.tech-invite.com/Ti-IPSec.html>.

5.1 Základní principy realizace zabezpečení

Samotná realizace zabezpečení stojí právě na použití rozšiřujících záhlaví Authentication Header (AH) a Encapsulating Security Payload (ESP). V datagramu lze použít jedno z nich, nebo obě zároveň (pro vyšší úroveň zabezpečení).

Tato záhlaví lze aplikovat v jednom ze dvou režimů: transportním nebo tunelujícím. V transportním režimu jsou záhlaví umístěna mezi ostatní rozšiřující záhlaví. Tunelující režim zabalí celý původní datagram do nového datagramu, který kromě základního záhlaví bude obsahovat také rozšiřující bezpečnostní záhlaví.

Datagramy nemusí být chráněny po celé své cestě. Například v lokální síti většinou není potřeba data šifrovat. Pokud ale datagram prochází přes médium, které lze odposlouchávat (typicky veřejný Internet), je výhodné datagram mezi hraničními směrovači odeslat tunelem – hraniční směrovač odesílající datagram ho nejprve zabalí do nového datagramu a přidá záhlaví ESP zajišťující šifrování. Hraniční směrovač přijímající datagram jej nejprve dešifruje a předá na lokální síť v původní podobě. Hraniční směrovače, nazývané bezpečnostní brány (Security gateway), jsou směrovače, přes které prochází data z lokální sítě do Internetu, nebo obráceně.

Základním elementem IPsec je bezpečnostní asociace (Security Association, SA). Jde o virtuální spojení, které zajišťuje zabezpečený přenos dat. SA se skládá ze tří částí:

- *Indexu bezpečnostních parametrů (Security Parameter Index, SPI)*
- *Cílové IP adresy*
- *Identifikátoru bezpečnostního protokolu (AH nebo ESP)*

Protože se jedná o jednosměrné spojení, je potřeba pro každý směr jedna SA. Pokud chceme využít zabezpečení jak pomocí AH tak pomocí ESP, budeme potřebovat další dvojici SA.

Index bezpečnostních parametrů (SPI) je identifikátor bezpečnostní asociace, určuje, zda je použito AH nebo ESP a slouží pro rozlišení různých SA spojených s jednou cílovou IP adresou.

Každé SA tedy obsahuje informace nezbytné ke správnému zpracování paketů – mimo použitého bezpečnostního protokolu také šifrovací algoritmus, jeho klíče, čítače, dobu životnosti a podobně.

S SA souvisí databáze bezpečnostní politiky (Security Policy Database, SPD), což je sada pravidel, která se používají k řízení provozu. Bezpečnostní politiku lze spravovat buďto manuálně (což je dost nepraktické), nebo automaticky (k tomuto účelu byl vytvořen protokol IKEv2) [14].

5.2 Authentication Header

Úkolem Authetication Header je ověřit, že datagram skutečně odeslal ten počítač, jehož adresa je v záhlavích paketů a že data jsou v té podobě, v jaké byly odeslány. Mimo to může zajišťovat i ochranu proti opakování. AH funguje tímto způsobem: odesílatel vloží do datagramu záhlaví AH, vyplní jeho položky a vypočte autentizační data (pro výpočet autentizačních dat se využívá hash, například MD5). Příjemce taktéž provede výpočet autentizačních dat a porovná, zda je výsledek shodný s tím, co odesílatel vložil do položky Autentizační data v záhlaví AH. Pokud jsou shodné, je odesílatel autentizován, v opačném případě je paket zahozen bez informování odesílatele, čímž případný útočník přijde o zpětnou vazbu. Pokud je využívána i ochrana proti opakování, je navíc kontrolováno i pořadové číslo datagramu. Pokud již bylo číslo použité, je datagram zahozen [16]. Vzhledem k tomu, že ESP nabízí stejné služby jako AH a k tomu navíc šifrování, je implementace AH již pouze volitelnou součástí IPsec a v budoucnu lze počítat s jejím úplným zánikem.

5.3 Encapsulating Security Payload

ESP nabízí především šifrování, mimo to ovšem i služby odpovídající AH. ESP netvoří jen záhlaví, ale i zápatí, takže původní datagram je vložen do ESP jako data. Výjimku představuje použití ESP v transportním režimu, kde se ESP umisťuje až za rozšiřující záhlaví pro každého po cestě, směrování a fragmentaci. Odeslání datagramu s ESP vypadá takto: odesílatel určí pozici, do které vloží záhlaví ESP, pokud je to potřeba, doplní datagram vycpávkou a zašifruje. Poté vygeneruje pořadové číslo (ochrana proti opakování), v případě požadavku na autentizaci a kontrolu integrity vypočítá kontrolní hodnotu a uloží ji do položky autentizační data v záhlaví ESP. Příjemce vyhledá odpovídající bezpečnostní asociaci, zkontroluje pořadové číslo a vypočte autentizační data. Pokud kterákoliv z těchto činností selže (neexistuje bezpečnostní asociace, opakované pořadové číslo, neodpovídající autentizační data), je paket zahozen. V opačném případě příjemce dešifruje paket, odstraní z něj ESP záhlaví a paket dále zpracuje [16].

5.4 Bezpečnostní cíle, šifrovací a autentizační algoritmy

Počítačovou bezpečnost představují tři základní cíle.

- *Autentizace* – neboli schopnost spolehlivě určit, zda byla data přijata tak, jak byla odeslána a ověřit, že entita, která data poslala je tím, za koho se vydává.
- *Integrita* – schopnost spolehlivě určit, že data nebyla pozměněna během přenosu od zdroje k cíli.
- *Utajení* – schopnost přenášet data takovým způsobem, že mohou být použita nebo čtena jen zamýšleným příjemcem a ne kýmkoliv jiným.

Díky moderním kryptografickým metodám může být těchto cílů dosaženo s použitím jedné sady funkcí, která zahrnuje:

- *Digitální podpisy* – Digitální podpis se používá pro autentizaci - reprezentuje ručení odesílatele za pravost a původ dat, příjemce ověřuje platnost podpisu pomocí klíčů. Digitální podpis, kterým jsou opatřena data odesílaná odesílatelem, je vytvořen pomocí tajného klíče odesílatele. K takovému klíči má příjemce veřejný klíč, který použije na dešifrování, spočítá a porovná hash a pokud souhlasí, příjemce ví, že data odeslal správný (pravý) uživatel. Pro digitální podpisy se nejčastěji využívá RSA algoritmu.
- *Bezpečnostní kontrolní součty (Hash)* – Hash je digitální součet dat libovolné velikosti, vytvořený pomocí opakovatelného procesu, který nad daty provádí jak odesílatel, tak příjemce. Odesílatel vypočítá hash a přidá jeho hodnotu k odesílaným datům. Příjemce vypočítá hash z přijatých dat, porovná hodnotu s hodnotou přijatou od odesílatele a pokud se rovnají, může data považovat za neporušená. Běžně používanými kontrolními součty jsou MD2, MD4 a MD5. Hash se používá buď samostatně, nebo jako součást digitálních podpisů.
- *Šifrování* – vratný proces transformace dat za účelem učinit je nečitelné pro kohokoliv, kdo nemá příslušný šifrovací klíč. Lze využít symetrické šifrování (neboli tajné klíčové šifrování - Secret Key Encyption), které je efektivním a nejčastěji používaným typem šifrování dat při přenosu sítí. Nejznámější algoritmy symetrického šifrování jsou Advanced Encryption Standard (AES), který v roce 2000 nahradil již nedostačující Data Encryption Standard (DES). Je možné použít též nesymetrické šifrování (neboli šifrování s veřejným klíčem - Public Key Encryption), které využívá metodu tajného a veřejného klíče. Jde o výpočetně náročnou metodu a používá se například

u digitálních podpisů. Nejpoužívanějším typem je RSA algoritmus vyvinutý Ronem Rivestem, Adi Shamirem a Lenem Adlemanem.

IPsec poskytuje výše uvedené bezpečnostní služby na IP vrstvě a zajišťuje součinnost bezpečnostních mechanismů mezi různými způsoby implementace IPv6. Povinně je stanovena pouze základní sada bezpečnostních algoritmů a jednotlivým uzlům je dovoleno vyjednávat mezi sebou volitelné algoritmy vhodné právě pro ně. IPsec poskytuje strukturu, na jejímž základě (a s jejíž pomocí) mohou uzly vyjednat vhodné algoritmy, protokoly, délky klíčů a další aspekty bezpečné komunikace.

Jak již bylo uvedeno, IPsec zabezpečuje pouze IP vrstvu, nikoliv Internet nebo systémy k němu připojené a procesy, které na těchto systémech běží. IPsec tedy tvoří pouze součást celkové bezpečnostní strategie. Data chráněná pomocí IPsec jsou sice zabezpečena při průchodu globálním Internetem, ale než opustí zdroj, nebo poté co dorazí do cíle, jsou tato data zranitelná na místních linkách, místních systémech, procesech a protokolech, které se zde používají [14].

5.5 Správa klíčů

Je jedním z nejkomplexnějších problémů, kterému lidé zabývající se síťovou bezpečností čelí. Správa klíčů nezajišťuje pouze distribuci klíčů pomocí protokolu pro výměnu klíčů, ale také vyjednávání délky klíče, doby života a kryptografických algoritmů mezi komunikujícími systémy.

Výměna klíčů a nastavení bezpečnostních parametrů je základem pro každou bezpečnostní architekturu. V původním IPsec se k tomuto účelu využíval Internet Key Exchange (IKE) protokol a Internet Security Association and Key Management Protocol (ISAKMP). První jmenovaný předepisoval mechanismus výměny klíčů (například využitím Diffie-Hellman algoritmu). ISAKMP sloužil pro správu bezpečnostních asociací, čili pomocí tohoto protokolu vyjednávaly komunikující entity bezpečnostní parametry a algoritmy, které se mají použít. V současné generaci IPsec byly tyto dva protokoly nahrazeny jedním – IKEv2, který v sobě slučuje funkce obou původně použitých protokolů.

6 METODY PŘECHODU Z IPV4 NA IPV6

Vzhledem k podstatným změnám a tedy vzájemné nekompatibilitě mezi IPv4 a IPv6 byly při tvorbě IPv6 vytvořeny i mechanismy, které zajistí dočasnou funkčnost obou verzí protokolu.

Internet svým současným rozsahem a významem nedovoluje „skokem“ přejít z IPv4 na IPv6, proto je nutné použití těchto mechanismů, které umožní současný provoz obou protokolů, spolupráci mezi nimi a postupný přechod od jednoho k druhému. Podíl IPv6 na Internetu by se měl tedy postupně zvyšovat a vytlačovat starší verzi, která ale nemusí úplně zmizet – mohou zůstat „ostrůvky“ tvořené sítěmi s IPv4. Mechanismy určené pro koexistenci a vzájemnou spolupráci obou protokolů můžeme rozdělit na tři obecné skupiny: dvojí zásobník (Dual Stack), tunelování (Tunneling) a překladače (Translators).

6.1 Obecné rozdělení metod a principy jejich funkce

6.1.1 Dvojí zásobník

Dvojí zásobník je integrační metoda, ve které má dané zařízení podporu obou protokolů, což mu umožňuje komunikovat jak s IPv4 zařízeními, tak s IPv6 zařízeními. Který zásobník použít, se uzel rozhodne na základě cílové adresy paketu, přičemž bude preferovat IPv6 kdekoliv to bude možné. Právě tuto metodu využívá ve svých zařízeních například společnost Cisco, především proto, že všechny ostatní metody pro svou funkčnost vyžadují, aby alespoň některá zařízení (ideálně směrovače) podporovala oba protokoly [13].

6.1.2 Tunelování

Tunelování je metoda umožňující přenášet data přes infrastrukturu, která nemá potřebné vlastnosti. Principem funkce je, že se původní datagram zabalí do jiného datagramu, který může být přenesen danou sítí. Nejběžnějším případem použití tunelu je propojení dvou počítačů (nebo sítí), které sice využívají protokol IPv6, ale síť, kterou jsou propojeny, využívá IPv4.

Existují dva typy tunelů: manuální (explicitně konfigurovány správcem) a automatické (dynamické – navazují se samočinně). Manuální tunely se využívají například u Tunnel Brokeru či TSP. Mezi metody, vycházející z automatických tunelů, patří 6to4, 6over4, ISATAP nebo Teredo.

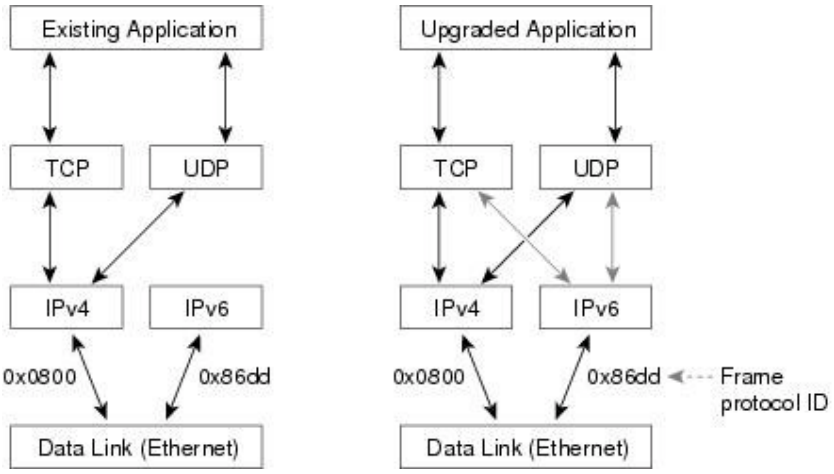
6.1.3 Překladače

Předchozí dva přístupy jsou založeny na tom, že v síti je integrována i nativní IPv6. Může ale nastat případ (např. u mobilních sítí třetí a čtvrté generace), kdy zařízení podporující pouze IPv6 potřebuje komunikovat s uzlem nebo zdrojem podporujícím pouze IPv4. Právě v takovýchto případech se použijí translátory. Translace umožňuje přímou komunikaci mezi uzly, které používají rozdílné verze protokolu. Mezi translační metody přechodu patří SIIT, NAT-PT, NAT64, TRT, BIS a SOCKS64.

6.1.4 Princip funkce přechodových mechanismů

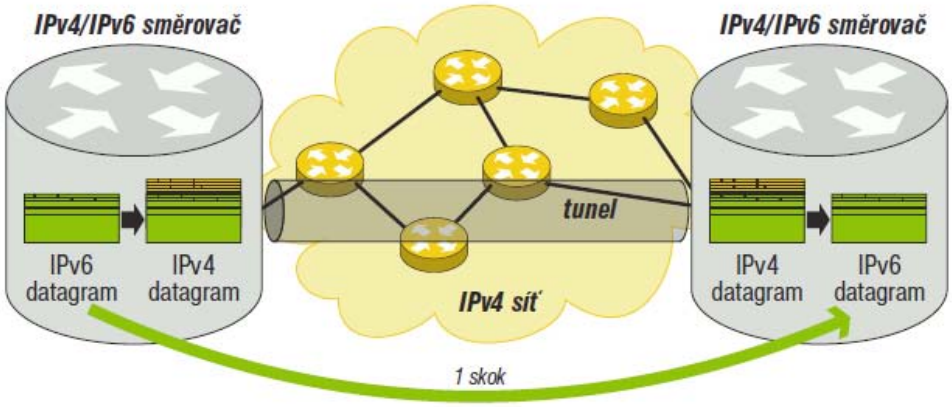
Základními nástroji, které využívá většina přechodových metod, jsou právě dvojí zásobník a princip tunelování. Pro správnou funkčnost obou verzí protokolů je též zapotřebí fungující DNS infrastruktura.

Zařízení mající dvojí zásobník (IPv4/IPv6 uzel) musí podporovat oba protokoly – tedy mít jak IPv4 tak IPv6 adresu a také umět pracovat s DNS záznamy pro obě verze protokolu.



Obr. 9: Dvojí zásobník – vrstvový model
Zdroj: H3C TECHNOLOGIES. *Dual Stack Introduction* [online]. <http://www.h3c.com/portal/>

Princip tunelování se v počítačových sítích vyskytuje běžně. V případě přechodových metod se jedná o „balení“ IPv6 do IPv4 nebo obráceně. Tunel má dva konce, které jsou tvořeny IPv4/IPv6 uzly. Pokud se uzel na jednom konci tunelu rozhodne, že musí IPv6 datagram odeslat data tunelem, vloží jej do IPv4 datagramu, který bude mít cílovou adresu (IPv4) druhého konce tunelu.



Obr. 10: Princip tunelování
Zdroj: SATRAPA, P. *IPv6 – Internetový protokol verze 6*. Praha : CZ.NIC, 2008. s. 237

Aby druhá strana poznala, že se jedná o tunelovaný IPv6 datagram, vloží odesílající konec tunelu do položky *Protocol* obalujícího IPv4 paketu hodnotu 41. Když paket dorazí na druhý konec tunelu, příjemce právě podle hodnoty 41 pozná, že se jedná tunelovaný IPv6 paket, který vybalí a dále zpracuje. Z hlediska IPv6 představuje průchod jakkoliv dlouhým tunelem jeden skok, takže vybalující směrovač zmenší položku *Max Hop Limit* (TTL) o jedna [16].

Pro správnou koexistenci obou verzí IP protokolu je též potřebná DNS infrastruktura, protože používání jmen (k označení síťových zdrojů) převládá nad používáním adres. Správných přiřazení IPv6 jméno – adresa a adresa – jméno se dosáhne přidáním AAAA a PTR záznamů do DNS serveru.

DNS musí tedy obsahovat A záznamy pro IPv4-only a IPv4/IPv6 uzly a AAAA záznamy pro IPv6-only a IPv4/IPv6 uzly, PTR záznamy v IN-ADDR.ARPA doméně pro IPv4-only a IPv4/IPv6 uzly a PTR záznamy v IP6.INT doméně pro IPv6-only a IPv4/IPv6 uzly.

Pokud uzel, který má nakonfigurovánu jak IPv4 tak IPv6 adresu, dostane odpověď na DNS dotaz a tato odpověď obsahuje více IPv4 a IPv6 adres, uzel použije algoritmus pro rozhodnutí, jakou zdrojovou a cílovou adresu a jaký rozsah použít. Existují algoritmy pro výběr zdrojové i cílové adresy, každý obsahuje kolem deseti kritérií, která určují, jakou adresu preferovat. Na pravidlech pro výběr adres se stále pracuje [4], [14]. V současné chvíli jsou IPv6 adresy v odpovědích na DNS dotazy preferovány před IPv4 adresami.

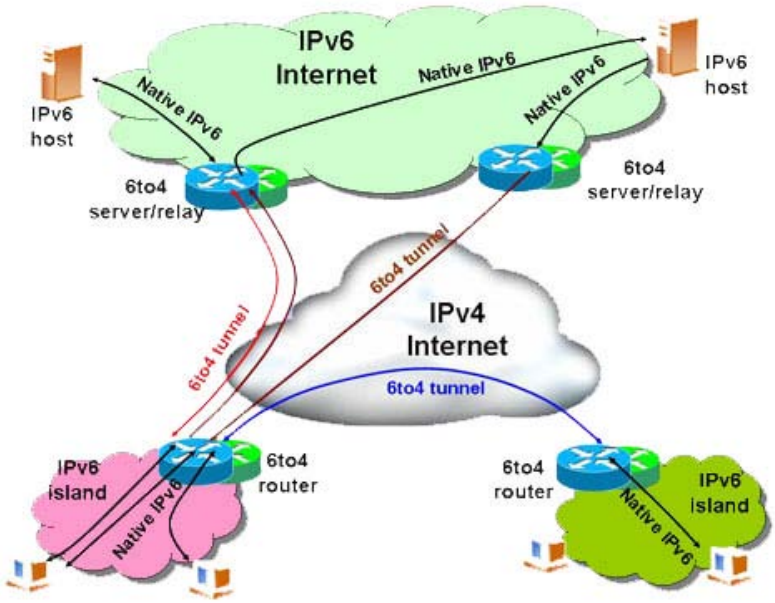
6.2 Konkrétní metody přechodu

Tato podkapitola se věnuje vysvětlení principu funkce vybraných přechodových metod, které byly v rámci této práce použity pro připojení k IPv6-Internetu a také analýze jejich využití. Zastoupeny jsou metody 6to4, Tunnel Broker (TSP) a Teredo.

6.2.1 6to4

6.2.1.1 Popis

6to4 je nejrozšířenějším zástupcem metod využívajících automatického tunelování. Hlavním cílem 6to4 je, za pomoci minimální konfigurace, umožnit koncovým IPv6 sítím vzájemnou komunikaci, přestože jsou připojeny k IPv4 Internetu (tedy do doby, než získají nativní IPv6 konektivitu). Na rozdíl od ostatních zmíněných metod nevyžaduje 6to4 explicitně nakonfigurovaný tunel. Zde stačí nakonfigurovat 6to4 směrovač (6to4 router), který bude přijímat pakety zapouzdřené pomocí 6to4 od kteréhokoliv uzlu.



Obr. 11: Schéma metody 6to4
Zdroj: THE IPV6 PORTAL. Using IPv6 [online]. <http://www.ipv6tf.org/index.php?page=using/connectivity/6to4>

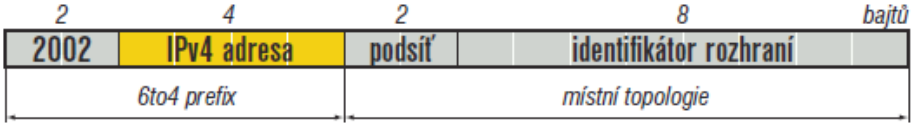
Metoda má využití hlavně v počáteční fázi přechodu od IPv4 k IPv6, v pozdější fázi, kdy již bude běžné nativní IPv6 připojení k Internetu od ISP, se postupně stane zbytečnou a zmizí [2].

6.2.1.2 Princip činnosti a samotná realizace

Základním předpokladem nutným pro použití této metody je alespoň jedna veřejná IPv4 adresa, nakonfigurovaná na zařízení, které bude v roli tzv. 6to4 směrovače. To může být libovolný uzel v dané síti, ale z hlediska efektivity komunikace se zpravidla jedná o směrovač s dvojím zásobníkem, umístěný na rozhraní mezi danou IPv6 sítí a IPv4 Internetem, viz obr. 11.

Uzel, který má nakonfigurovanou pouze 6to4 adresu ale nemůže přímo komunikovat s uzlem s nativním IPv6. Pro komunikaci s nativním IPv6 je nutné využít zprostředkovatele (6to4 relay), což je směrovač v Internetu, který zajistí, že data z 6to4 paketů přicházející na jeho IPv4 rozhraní budou předána do IPv6 sítě a naopak IPv6 pakety na jeho IPv6 rozhraní budou opatřeny 6to4 záhlavím a odeslány přes IPv4 síť [16], opět viz obr. 11.

6to4 adresa, jejíž struktura je zobrazena na obr. 12, je lehce rozpoznatelná – začíná hodnotou 2002:/16. Dalších 32 bitů pak tvoří IPv4 adresa 6to4 směrovače (přístupového směrovače), což dohromady vytváří standardní 48b prefix.



Obr. 12: Struktura 6to4 adresy
Zdroj: SATRAPA, P. *IPv6 – Internetový protokol verze 6*. Praha : CZ.NIC, 2008. s. 241

Jak již bylo řečeno, při komunikaci s nativní IPv6 síti je potřeba využít zprostředkovatele. V původní verzi návrhu 6to4 bylo možné najít nejvhodnějšího zprostředkovatele pouze pomocí externího směrovacího protokolu, jehož použití ovšem značně zvyšuje režii. Proto byla v RFC 3068 definována fixní adresa pro zprostředkovatele (192.88.99.1 resp. 2002:c058:6301::). Protože se jedná o výběrovou adresu, je zaručeno (viz kapitola 3.1), že se datagram dostane k nejbližšímu zprostředkovateli.

```
Frame 1 (899 bytes on wire, 899 bytes captured)
Ethernet II, Src: HonHaiPr_41:9c:20 (00:16:cf:41:9c:20), Dst:
Unispher_41:65:41 (00:90:1a:41:65:41)
PPP-over-Ethernet Session
Point-to-Point Protocol
Internet Protocol, Src: 70.55.213.211 (70.55.213.211), Dst:
192.88.99.1 (192.88.99.1)
    Version: 4
    Header length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN:
0x00)
    Total Length: 877
    Identification: 0x9359 (37721)
    Flags: 0x00
    Fragment offset: 0
    Time to live: 128
    Protocol: IPv6 (0x29)
    Header checksum: 0x64aa [correct]
    Source: 70.55.213.211 (70.55.213.211)
    Destination: 192.88.99.1 (192.88.99.1)
Internet Protocol Version 6
    0110 .... = Version: 6
    .... 0000 0000 .... .... .... .... .... = Traffic class:
0x00000000
    .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
    Payload length: 817
    Next header: TCP (0x06)
    Hop limit: 128
    Source: 2002:4637:d5d3::4637:d5d3 (2002:4637:d5d3::4637:d5d3)
    Destination: 2001:4860:0:2001::68 (2001:4860:0:2001::68)
Transmission Control Protocol, Src Port: routematch (1287), Dst Port:
http (80), Seq: 1, Ack: 1, Len: 797
Hypertext Transfer Protocol
```

Obr. 13: Paket z 6to4 komunikace

Paket na obr. 13 zachycuje HTTP GET dotaz klienta na server ipv6.google.com. To, že klient využívá metodu 6to4 lze identifikovat podle adresy začínající prefixem 2002:/16, následujících 32 bitů (v HEX: 4637:d5d3) po převodu do desítkové soustavy dává posloupnost 70.55.213.211 (IPv4 adresa místního 6to4 směrovače), což potvrzuje platnost pravidel pro tvorbu 6to4 adres uvedených výše. Cílová adresa není 6to4, což ukazuje na komunikaci s nativní IPv6, takže mezi zdrojovým počítačem a cílem (ipv6.google.com) leží 6to4 zprostředkovatel s IPv4 adresou 192.88.99.1 (výběrová adresa vyhrazená pro 6to4 zprostředkovatele). Poslední položkou stojící za povšimnutí je pole `Protocol` v záhlaví IPv4, kde kód `0x29` (41 v desítkové soustavě) značí, že nesená data jsou IPv6 paket.

Předností 6to4 je jeho minimální náročnost. Jednak má nejmenší režii v porovnání se zde popsanými metodami a při jeho použití je potřeba pouze nastavit 6to4 směrovač, aby ohlašoval síť 2002:/16 do lokální sítě. Datagramy adresované do 6to4 sítě pak

budou předány právě jemu, směrovač z cílové 6to4 adresy snadno zjistí IPv4 adresu směrovače ve vzdálené síti (viz obr. 12) a jako zdrojovou adresu použije své IPv4 rozhraní. Datagram je automaticky tunelován, ale tunel není založen trvale.

6.2.2 Tunnel Broker a TSP

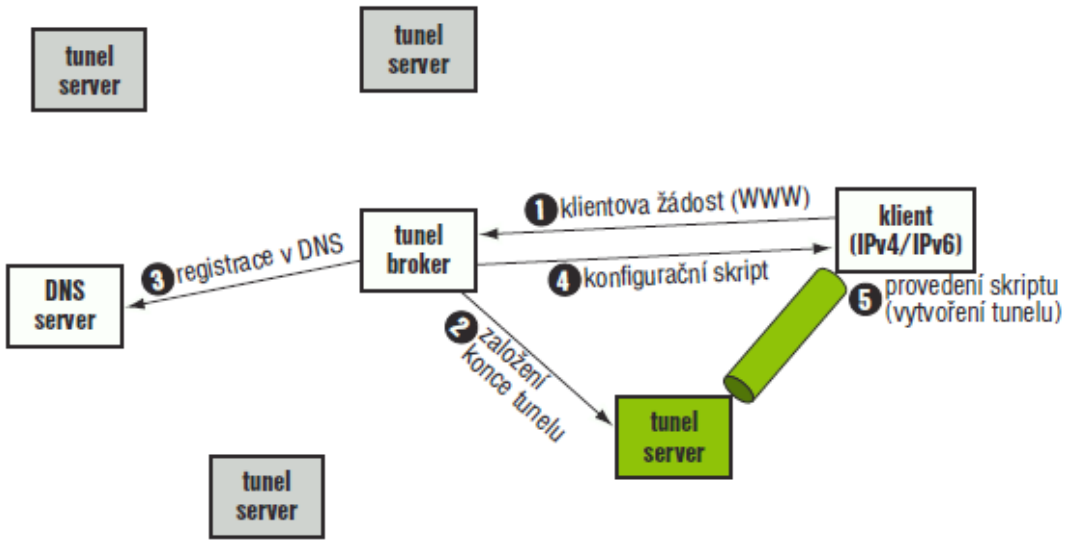
6.2.2.1 Tunnel Broker

Popis

Tunnel Broker je metoda využívající manuálně konfigurované tunely. Její základ tvoří síť vyhrazených serverů, nazvaných Tunnel Brokers, které automaticky spravují žádosti na zřízení tunelu od klientů (tj. uzlů, kterým je poskytována IPv6 konektivita).

Hlavním rozdíl mezi Tunnel Broker mechanismem a 6to4 metodou je (mimo to, že Tunnel Broker vychází z principu manuálních tunelů a 6to4 z automatických), že slouží opačným segmentům sítě: zatímco Tunnel Broker je vhodný hlavně pro izolované IPv6 uzly na IPv4 síti, které chtějí získat snadné připojení k existující IPv6 síti, 6to4 mechanismus byl navržen proto, aby bylo možné snadno propojit existující izolované IPv6 sítě bez nutnosti čekat, než jejich ISP začne poskytovat nativní IPv6 připojení.

Tunnel Broker je možno chápat jako virtuálního IPv6 ISP, který poskytuje IPv6 konektivitu uživatelům, kteří již mají připojení k IPv4 Internetu.



Obr. 14: Tunnel Broker – prvky
Zdroj: SATRAPA, P. *IPv6 – Internetový protokol verze 6*. Praha : CZ.NIC, 2008. s. 239

Princip činnosti a samotná realizace

Model Tunnel Brokeru je založen na několika funkčních elementech zobrazených na obr. 14. Jak je z obrázku patrné, v této metodě se objevují tři základní elementy:

- *Tunnel Broker* – místo, kam se uživatelé připojují, aby zaregistrovali a aktivovali tunel. Tunnel Broker spravuje vytváření, modifikaci a mazání tunelů, dle žádostí klienta. Musí mít IPv4 adresu, IPv6 adresa je nepovinná, komunikace mezi Brokerem a Serverem může probíhat jako po IPv4 tak po IPv6. Jeden Tunnel Broker může spravovat klientské žádosti pro více Tunnel Serverů.
- *Tunnel Server* – je zpravidla tvořen směrovačem s dvojím zásobníkem, který je připojen do globálního Internetu. Na základě informací přijatých od Tunnel Brokeru vytváří, modifikuje nebo ruší svou stranu tunelu. Také se stará o statistiky využití každého aktivního tunelu.
- *DNS Systém* – DNS zodpovídá za mapování IPv6 adres na doménová jména. Každý uživatel zpravidla dostane přiděleno doménové jméno, na které bude namapována jeho IPv6 adresa [11].

Procedury při použití Tunnel Brokeru:
Registrace/zrušení registrace – jednoduchá procedura, při které klient odešle žádost o registraci k Tunnel Brokerovi. Žádost obsahuje informace jako jméno, email a heslo (tyto údaje budou sloužit k autentizaci klienta), případně velikost bloku IPv6 adres. Broker přiřadí klientovi blok adres, DNS záznam a vrátí tyto informace uživateli. Při rušení registrace klient pošle žádost o zrušení registrace, následuje autentizace klienta, po které mu Tunnel Broker odebere přidělený blok adres a smaže veškeré záznamy v systému, které se k tomuto klientovi vztahují. Dokončení procedury oznámí klientovi tzv. good-bye zprávou.

Aktivace tunelu – proces zahajuje klient, který pošle žádost o aktivaci tunelu svému Tunnel Brokerovi. Klientova IPv4 adresa může být přímo poskytnuta v žádosti, nebo je získána z TCP komunikace. Tunnel Broker ověří stav klienta – pokud již byl jeho tunel aktivován a parametry se shodují s těmi, které klient poslal v aktuální žádosti o aktivaci, je vzdálený konec tunelu připraven. V opačném případě Tunnel Broker určí Tunnel Server, který bude pro daného klienta sloužit jako vzdálený konec tunelu. Tunnel Broker informuje vybraný Tunnel Server o příchodu nového klienta. Tunnel Server sestaví svůj konec tunelu a do IPv6 směrovací tabulky si přidá záznam, že adresa (blok

adres) nově příchozího klienta je dostupná právě přes tento nově vytvořený tunel. Tunnel Broker si zaznamená aktuální stav klienta a odešle mu informace o vytvořeném tunelu. Také může klientovi odeslat instrukce, jak nakonfigurovat jeho konec tunelu. Klient nakonfiguruje svou část tunelu – IPv4 adresa odpovídá té, která byla poskytnuta Tunnel Brokerovi a IPv4 adresa druhého konce tunelu je IPv4 adresa Tunnel Serveru. IPv6 adresu klientova konce tunelu poskytne Tunnel Broker.

Deaktivace tunelu – zpravidla taktéž vyvolána klientem. Tunnel Broker informuje příslušný Tunnel Server o zrušení tunelu. Server zruší svou stranu tohoto tunelu a odstraní příslušný záznam ve směrovací tabulce. Tato změna je posléze standardními metodami směrovacích protokolů ohlášena do směrovací infrastruktury. Tunnel Broker si uloží informaci o stavu klienta. Pokud byla deaktivace tunelu vyvolaná přímo klientem, pošle mu Tunnel Broker zprávu o deaktivaci tunelu a klient zruší konfiguraci tunelu na své straně [11].

První použití (registrace tunelu) Tunnel Brokeru tedy vyžaduje osobu s určitou znalostí síťových nastavení – je nutné nakonfigurovat parametry tunelu. Další používání tunelu pak již spočívá pouze v jeho aktivaci. Nutnost konfigurovat parametry tunelu odstraňuje použití pokročilejšího TSP, popsaného dále.

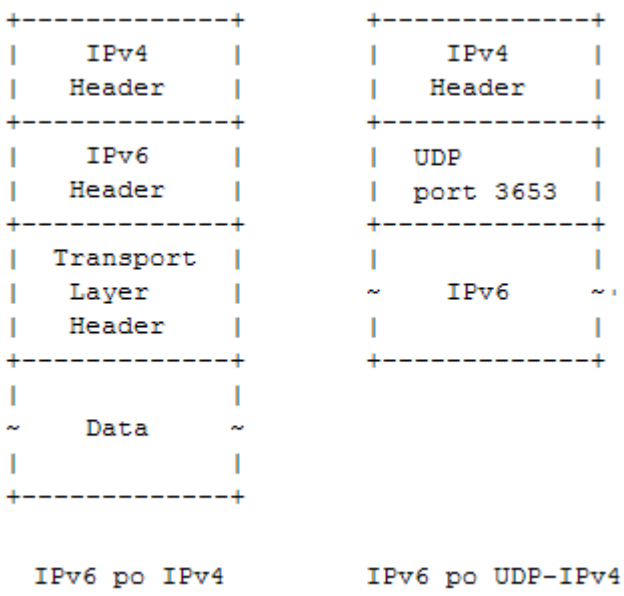
6.2.2.2 Tunnel Setup Protocol (TSP)

Popis

Tunnel Setup Protocol je nástroj, který může využít klient pro vyjednání tunelu s Tunnel Brokerem. Výhodou tohoto řešení je, že značně usnadňuje celý proces. TSP je v podstatě signalizační protokol, který slouží k vytvoření tunelu mezi dvěma koncovými body. Informace jsou přenášeny v XML přes TCP nebo UDP protokol.

TSP lze také použít pro zjištění, zda se některý z koncových bodů tunelu nenachází za NATem. Proces zjišťování NATu je obdobný jako u metody Teredo popsané dále. Pokud TSP detekuje NAT, budou IPv6 datagramy posílány přes UDP-IPv4. Pokud se NAT v cestě nenachází, použije se tunel IPv6 po IPv4 (Rozdíl viz obr. 15). Tento mechanismus zjišťování NATu zajistí, že za každých podmínek je pro každého uživatele sestaven ten nejvhodnější tunel, včetně dynamických situací, kdy se klient pohybuje.

Mobilita – pokud se uzel přesune do jiné sítě (změní se jeho IPv4 adresa při tunelování IPv6 po IPv4), TSP klient se znovu automaticky připojí a znovu sestaví tunel. Jakmile se klient autentizuje, obdrží zpět stejnou IPv6 adresu a prefix, i přesto, že se změnila jeho IPv4 adresa nebo typ zapouzdření.



Obr. 15: Rozdíl v zapouzdření

Princip činnosti a samotná realizace

Parametry, které je před sestavením tunelu nutno vyjednat jsou následující:

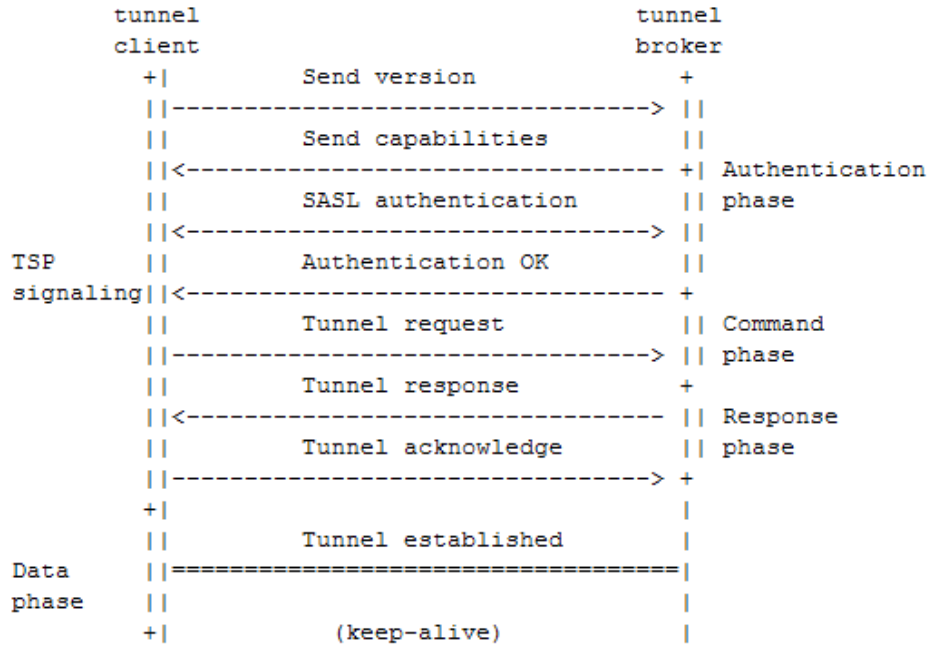
- *Autentizace uživatele* (pomocí SASL) je možné použít i anonymous
- *Zapouzdření tunelu* (IPv6 po IPv4, IPv4 po IPv6 nebo IPv6 po UDP-IPv4 pro překonání NATu
- *Přiřazení IP adres oběma koncům tunelu*
- *DNS registrace obou konců tunelu*

TSP je iniciován z klientského uzlu směrem k Tunnel Brokerovi a má tři fáze (jejich posloupnost znázorňuje obr. 16):
Autentizační fáze (Authentication phase) – nastává, když Tunnel Broker/Server, oznámí klientovi dostupnost a škálu nabízených služeb a klient se u Brokera/Serveru autentizuje.

Příkazová fáze (Command phase) – ta část, kdy klient žádá o tunel nebo aktualizuje tunel stávající.

Fáze odpovědi (Response phase) – ta část komunikace, kdy klient dostane od Brokera/Serveru odpověď, na jejímž základě tunel buď přijme, nebo odmítne.

Na každý příkaz poslaný klientem existuje předpokládaná odpověď serveru. Jakmile skončí pozitivně fáze odpovědi, je tunel sestaven podle požadavků klienta. Na vyžádání mohou být posílány keep-alive pakety z klienta na server [1].



Obr. 16: TSP – výměna zpráv
Zdroj: BLANCHET, M. – PARENT, F. IPv6 *Tunnel Broker with the Tunnel Setup Protocol (TSP) [online]*. <http://www.ietf.org/internet-drafts/draft-blanchet-v6ops-tunnelbroker-tsp-04.txt>

Následující obrázek uvádí žádost a vytvoření tunelu v reálné síti (C značí komunikaci od klienta k Serveru, S od Serveru ke klientovi):

```
<!-- Autentizační fáze -->

        -- Successful TCP Connection --
        C:VERSION=2.0.0 CR LF
        S:CAPABILITY TUNNEL=V6V4 TUNNEL=V6UDPV4 AUTH=DIGEST-MD5 CR LF
        C:AUTHENTICATE ... CR LF
        S:200 Authentication successful CR LF

<!-- Příkazová fáze -->

        C:Content-length: ... CR LF
          <tunnel action="create" type="v6v4">
             <client>
                 <address type="ipv4">1.1.1.1</address>
             </client>
          </tunnel> CR LF
```

```
<!-- Fáze odpovědi -->

        S: Content-length: 234 CR LF
           200 OK CR LF
           <tunnel action="info" type="v6v4" lifetime="1440">
             <server>
                <address type="ipv4">192.0.2.114</address>
                <address type="ipv6">
                2001:db8:c18:ffff:0000:0000:0000:0000
                </address>
             </server>
             <client>
                <address type="ipv4">1.1.1.1</address>
                <address type="ipv6">
                2001:db8:c18:ffff::0000:0000:0000:0001
                </address>
                <address type="dn">userid.domain</address>
             </client>
           </tunnel> CR LF
        C: Content-length: 35 CR LF
           <tunnel action="accept"></tunnel> CR LF
```

Obr. 17: TSP – příklad sestavení tunelu pomocí XML zpráv

Komunikaci zahajuje klient, zasláním informace o podporované verzi protokolu. Následně je informován Serverem o možných typech tunelu a způsobu autentizace (`CAPABILITY TUNNEL=V6V4 TUNNEL=V6UDPV4 AUTH=DIGEST-MD5 CR LF`). Klient pošle autentizací údaje, na které mu Server odpoví hlášením o úspěšné autentizaci. Následuje žádost klienta o vytvoření tunelu typu „IPv6 po IPv4“ (`<tunnel action="create" type="v6v4">`). Element `tunnel` uvozující žádost dále obsahuje také verzi protokolu a IP adresu klienta. Server odpovídá také pomocí elementu `tunnel` s atributem `action` nastaveným na `info`, který se používá při posílání parametrů tunelu od Serveru ke klientovi nebo ke zjištění aktuálních vlastností tunelu. Dále následují v příslušných elementech IPv4 a IPv6 adresy Brokera/Serveru a klienta. Je zde uvedeno i doménové jméno přiřazené klientovi. Klient oznámí Serveru přijetí tunelu opět pomocí elementu `tunnel` s atributem `action` tentokrát nastaveným na `accept`. S pomocí získaných informací uvede klient do provozu svůj konec tunelu a tunel se stává funkčním/aktivním.

TSP nabízí variabilitu, díky které ho lze využít v rozmanitých síťových prostředích (ISP, podniková síť, bezdrátové sítě, domácí síť). Navíc také umožňuje oběma koncům vyjednávat (během autentizované „session“) další parametry jako třeba prefix, DNS nebo směrování. Výhodou je též možnost využít TSP pro mobilní uzly.

6.2.3 Teredo

6.2.3.1 Popis

Běžné tunelovací metody využívají postupu, kde je IPv6 paket odeslán jako data v IPv4 paketu. Problém těchto metod je, že nemusí fungovat, pokud se potenciální IPv6 uzel bude nacházet za zařízením provádějícím překlad adres (NAT).

Řešení tohoto problému představuje metoda Teredo, kde jsou IPv6 pakety přenášeny jako data v UDP paketech. U protokolu UDP je totiž zaručeno, že projde většinou NAT zařízení. Tunelování přes TCP by bylo teoreticky též možné, ale ve výsledku by vedlo k horší kvalitě služby, proto je použití protokolu UDP lepším řešením.

Myšlenka Tereda vychází z faktu, že komunikaci je potřeba zahajovat zevnitř NATované sítě, aby se vytvořila odpovídající vazba. Je třeba zohlednit charakter NATu, jímž datagramy ke klientovi procházejí. Nejjednodušší je trychtýřový NAT (Cone NAT), který klientovi přiřadí určitou adresu a port a pak k němu (na určenou adresu a port) propustí jakýkoliv paket. Složitější je omezený NAT (Restricted NAT), který klientovi také přiřadí jednu adresu a port, ale předává mu datagramy jen z takové adresy a portu, na které už klient dříve něco poslal. Mají-li proto ke klientovi dorazit data zvenčí, musí nejprve on sám kontaktovat jejich odesilatele. Překážkou pro Teredo je symetrický NAT (Symmetric NAT), který se chová jako omezený ale navíc pro data odesílaná k různým cílům přiděluje stejnému klientovi odlišné adresy a porty. Symetrický NAT dnes ovšem není příliš častý a nové modely zařízení, či nové verze firmwarů se již symetrickému designu vyhýbají [16].

Teredo bylo vytvořeno, jako „poslední možnost“ pro přístup k IPv6, a používá se v případech, kdy nelze realizovat jinou přechodovou metodu (například 6to4).

6.2.3.2 Princip činnosti a samotná realizace

Metoda definuje několik základních pojmů:

- *Teredo klient* – uzel s přístupem k IPv4 Internetu, který chce získat přístup k IPv6 Internetu
- *Teredo server* – uzel mající přístup jak k IPv4 tak k IPv6 (má globální adresy), slouží jako pomocník pro poskytování IPv6 konektivity Teredo klientům

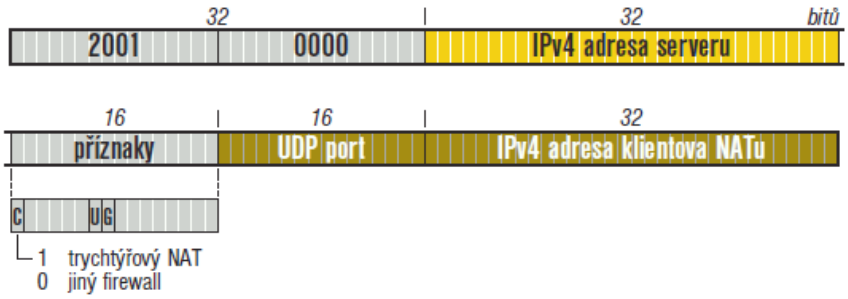
- *Teredo relay* – IPv4/IPv6 směrovač, který může přijímat data určená Teredo klientům a přeposílat je
- *Teredo Bubble* – IPv6 paket minimální velikosti, tvořený z IPv6 hlavičky a žádných dat. Používá se k vytvoření mapování v NATu [8]

Teredo klient zahajuje činnost komunikací s Teredo serverem, probíhá tzv. kvalifikační procedura (Qualification Procedure). Během této procedury klient zjistí, za jakým typem NATu se nachází. V případě trychtýřového nebo omezeného NATu bude procedura úspěšná a klient si nakonfiguruje Teredo adresu.

Pokud bude identifikován omezený NAT, bude potřeba otevírat cestu speciálními pakety (Teredo Bubble, viz dále). Pokud je identifikován symetrický NAT, celá procedura zde končí.

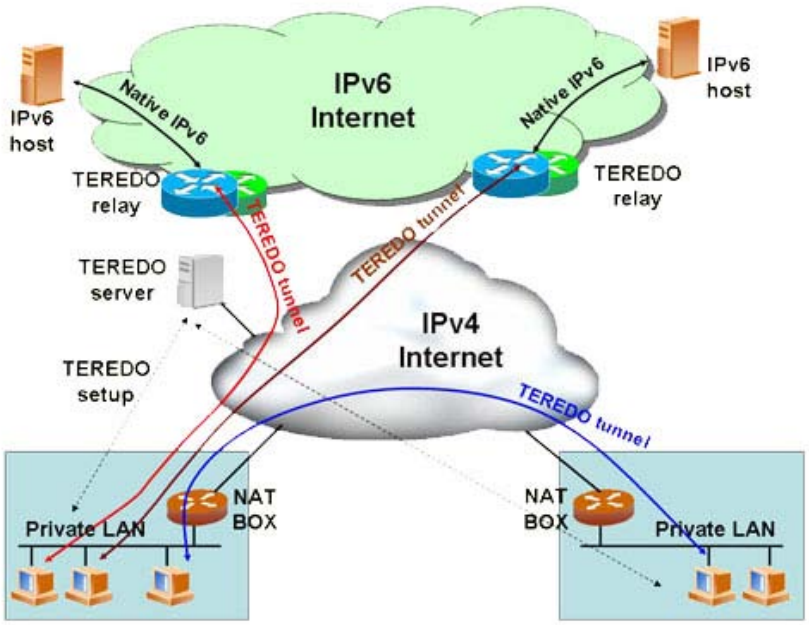
Samotná Teredo adresa je pak složena z pěti částí, viz obr. 18:

- *Prefix:* 32-bitový Teredo prefix
- *Server IPv4:* IPv4 adresa Teredo serveru
- *Příznaky (Flags):* 16 bitů, ve kterých je obsažen typ NATu a adresy
- *Port:* invertované číslo namapovaného UDP portu klienta
- *Client IPv4:* invertovaná namapovaná IPv4 adresa klientova NATu



Obr. 18: Struktura Teredo adresy
Zdroj: SATRAPA, P. *IPv6 – Internetový protokol verze 6*. Praha : CZ.NIC, 2008. s. 247

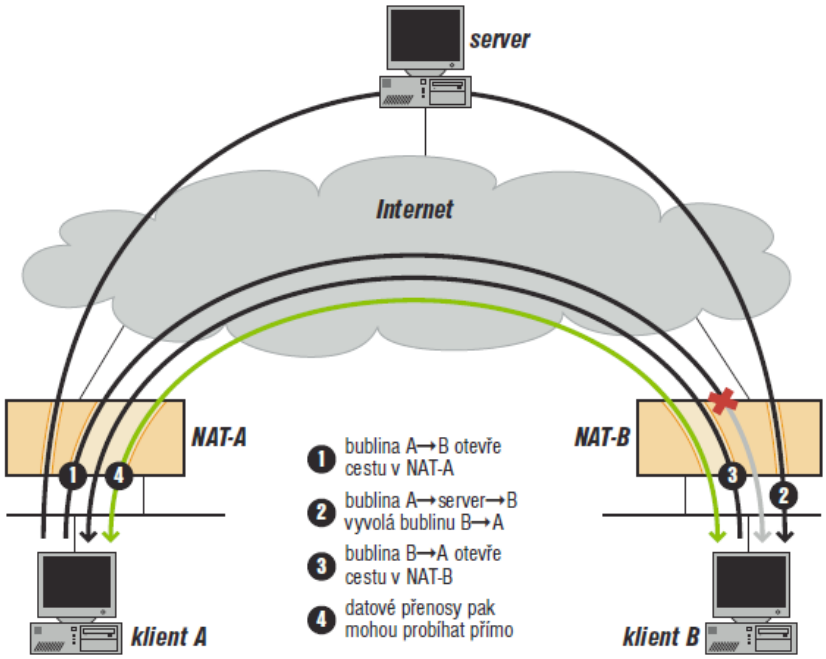
Příznak C slouží k rozlišení trychtýřového NATu od ostatních typů, příznaky U a G jsou obecné pro všechny identifikátory rozhraní a v případě Tereda by měly být nastaveny na 0 a příjemce by je měl ignorovat. Hodnoty v posledních dvou částech (UDP port a IPv4 adresa klienta) jsou invertovány, aby NAT nemohl tyto hodnoty v datagramu vyhledat a přepsat [8].



Obr. 19: Schéma metody Teredo
Zdroj: THE IPV6 PORTAL. *Using IPv6* [online]. <http://www.ipv6tf.org/index.php?page=using/connectivity/teredo>

Při komunikaci dvou Teredo klientů si klienti vyměňují data přímo. Pokud se adresát nachází za trychtýřovým NATem, stačí odesílateli zabalit IPv6 datagram do UDP zprávy, kterou pošle na adresu a port uvedené na konci adresy.

Pokud jsou klienti za omezeným NATem, je potřeba využít speciálních paketů – bublin (Teredo Bubble) – ty projdou NATem a vytvoří v něm potřebné mapování. První bublinu pošle odesílatel adresátovi, čímž se vytvoří záznam v NATu odesílatele (a tím pádem může přijít odpověď od adresáta). První bublina je zahozena u NATu adresáta (zde zatím neexistuje mapování). Druhou bublinu pošle odesílatel přes Teredo server, pro který mapování u adresáta existuje, takže bublina dorazí k cíli, který se dozví, že s ním chce někdo komunikovat a odpoví bublinou adresovanou odesílateli, kterou se otevře cesta v jeho NATu. Pak již nic nebrání přímé komunikaci mezi těmito dvěma uzly. Popsaná situace je zobrazena na obr. 20.



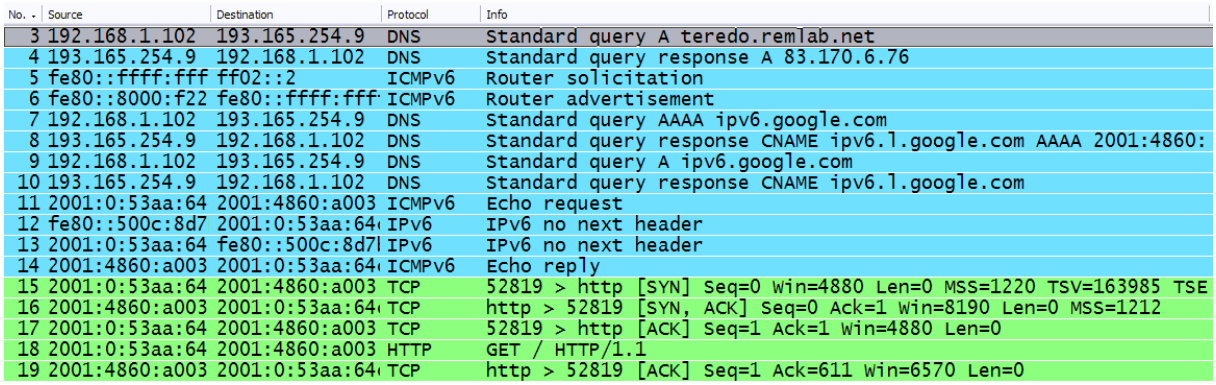
Obr. 20: Teredo – funkce bublin
Zdroj: SATRAPA, P. *IPv6 – Internetový protokol verze 6*. Praha : CZ.NIC, 2008. s. 250

Pokud chce Teredo klient komunikovat s uzly využívajícími jinou technologii, je potřeba využít zprostředkovatele (Teredo Relay) – směrovače, který bude předávat pakety z nativní IPv6 sítě do Teredo sítě a naopak (šíří směrovací informaci, že jeho prostřednictvím je přístupná síť 2001::/32, což je Teredo prefix).

Prvním krokem pro Teredo klienta je najít vhodný směrovač – zprostředkovatele. To provede odesláním „výzvy směrovači“ (ICMPv6 echo zpráva zabalená do UDP paketu) svému Teredo serveru. Ten ji vybalí a IPv6 sítí odešle adresátovi. Odpověď je pomocí směrovacích protokolů doručena na směrovač, který bude sloužit jako zprostředkovatel. Zde další postup opět závisí na tom, jaký NAT odesílatel používá. Pokud používá trychtýřový, zprostředkovatel mu může odeslat paket přímo. Pokud používá omezený NAT, musí zprostředkovatel poslat paket nepřímo přes Teredo server. Klient po jeho přijetí pošle paket na IPv4 adresu zprostředkovatele, a tím vytvoří namapování ve svém NATu. V obou případech získá klient adresu zprostředkovatele a již ví, na kterou IPv4 adresu a UDP port posílat datagramy pro adresáta z IPv6 sítě.

Aby se předešlo nepraktickému procesu s otevíráním NATů pro každý datagram (což navíc zvyšuje režii), udržuje si Teredo klient seznam komunikačních partnerů. V něm si uchovává informace, jak s danými partnery komunikovat. Pokud pro určitý cíl

najde v tomto seznamu platnou položku, bude s datagramem zacházeno podle těchto informací [16].



No.	Source	Destination	Protocol	Info
3	192.168.1.102	193.165.254.9	DNS	Standard query A teredo.remlab.net
4	193.165.254.9	192.168.1.102	DNS	Standard query response A 83.170.6.76
5	fe80::ffff:fff	ff02::2	ICMPv6	Router solicitation
6	fe80::8000:f22	fe80::ffff:fff	ICMPv6	Router advertisement
7	192.168.1.102	193.165.254.9	DNS	Standard query AAAA ipv6.google.com
8	193.165.254.9	192.168.1.102	DNS	Standard query response CNAME ipv6.l.google.com AAAA 2001:4860:
9	192.168.1.102	193.165.254.9	DNS	Standard query A ipv6.google.com
10	193.165.254.9	192.168.1.102	DNS	Standard query response CNAME ipv6.l.google.com
11	2001:0:53aa:64	2001:4860:a003	ICMPv6	Echo request
12	fe80::500c:8d7	2001:0:53aa:64	IPv6	IPv6 no next header
13	2001:0:53aa:64	fe80::500c:8d7	IPv6	IPv6 no next header
14	2001:4860:a003	2001:0:53aa:64	ICMPv6	Echo reply
15	2001:0:53aa:64	2001:4860:a003	TCP	52819 > http [SYN] Seq=0 Win=4880 Len=0 MSS=1220 TSV=163985 TSE
16	2001:4860:a003	2001:0:53aa:64	TCP	http > 52819 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1212
17	2001:0:53aa:64	2001:4860:a003	TCP	52819 > http [ACK] Seq=1 Ack=1 Win=4880 Len=0
18	2001:0:53aa:64	2001:4860:a003	HTTP	GET / HTTP/1.1
19	2001:4860:a003	2001:0:53aa:64	TCP	http > 52819 [ACK] Seq=1 Ack=611 Win=6570 Len=0

Obr. 21: Posloupnost paketů při komunikaci pomocí metody Teredo

Obr. 21 zachycuje posloupnost komunikace při použití této metody. Teredo klient nejprve kontaktuje Teredo server po IPv4 (pakety č. 3 a 4 – DNS dotaz/odpověď na adresu Teredo serveru), následuje kvalifikační procedura (pakety 5 a 6) – klient odesílá výzvu směrovači (Router Solicitation) ze své lokální linkové adresy na skupinovou adresu všech směrovačů (FF02::2), během které se zjistí, za jakým typem NATu se klient nachází a server mu přidělí Teredo prefix, viz obr. 22.

```
ICMPv6 Option (Prefix information)
    Type: Prefix information (3)
    Length: 32
    Prefix length: 64
    Flags: 0x40
        0... .... = Not onlink
        .1.. .... = Auto
        ..0. .... = Not router address
        ...0 .... = Not site prefix
    Valid lifetime: infinity
    Preferred lifetime: infinity
    Prefix: 2001:0:53aa:64c::
```

Obr. 22: Teredo – obsah ICMP zprávy z ohlášení směrovače

K prefixu si klient připojí inverzní hodnoty použitého portu a namapované IPv4 adresy a získá globální IPv6 adresu, viz obr. 23. S globální adresou pak již klient může komunikovat s IPv6 uzly. Na obr. 23 je zachycen ICMPv6 paket směřující od klienta k serveru ipv6.google.com. Identifikátorem použité metody je kromě adresy odesílatele i cílový port 3544 (port pro Teredo).

```
Frame 11 (108 bytes on wire, 108 bytes captured)
Ethernet II, Src: ZyxelCom_a5:2d:a6 (00:13:49:a5:2d:a6), Dst: Cisco-
Li_94:cf:30 (00:1a:70:94:cf:30)
Internet Protocol, Src: 192.168.1.102 (192.168.1.102), Dst:
83.170.6.76 (83.170.6.76)
User Datagram Protocol, Src Port: 46177 (46177), Dst Port: teredo
(3544)
    Source port: 46177 (46177)
    Destination port: teredo (3544)
    Length: 74
    Checksum: 0x86d5 [correct]
Teredo IPv6 over UDP tunneling
Internet Protocol Version 6
    0110 .... = Version: 6
    .... 0000 0000 .... .... .... .... .... = Traffic class:
0x00000000
    .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
    Payload length: 26
    Next header: ICMPv6 (0x3a)
    Hop limit: 128
    Source: 2001:0:53aa:64c:42b:4b9e:3e5a:ef9d
(2001:0:53aa:64c:42b:4b9e:3e5a:ef9d)
    Destination: 2001:4860:a003::68 (2001:4860:a003::68)
Internet Control Message Protocol v6
```

Obr. 23: Paket z Teredo komunikace

Samotné používání této metody běžným uživatelem patří mezi nejjednodušší – operační systémy Windows Vista a Windows 7 přímo obsahují Teredo klienta, takže uživatel může komunikovat s IPv6 uzly bez nutnosti cokoliv nastavovat či instalovat. V případě nejrozšířenějších Windows XP je ovšem potřeba nejprve aktivovat samotné IPv6 (ve výchozím nastavení je protokol vypnutý), v Linuxu nebo v Mac OS X je použití Tereda možné po instalaci příslušného klienta.

Nevýhodou použití této metody je značně delší doba odezvy v porovnání s IPv4 (závisí na umístění vzdáleného konce tunelu), velká režie a nemožnost nastavovat další parametry jako například u TSP.

ZÁVĚR

Dřívější pracovní název pro protokol, který nahradí IPv4 byl IPng, kde písmena „ng“ označující „next generation“ tedy další generaci, což IPv6 stručně a jasně charakterizuje. Dostatek času na vývoj a práce několika vývojových skupin dala vzniknout protokolu, který není jen řešením problému s nedostatkem IP adres, ale představuje značně přepracovaný protokol, který přináší řadu vylepšení s patrným důrazem na jednoduchost, rychlost, bezpečnost a mobilitu – tedy prvky charakterizující komunikaci v 21. století.

K posouzení těchto pozitivních vlastností v praxi by ale byla nejdříve potřeba jeho masivní implementace. Realita je ovšem taková, že i přes neoddiskutovatelná pozitiva nového protokolu je jeho nasazení, v porovnání s předchozí verzí, velmi malé. To je způsobeno jednak absencí služeb, určených výhradně pro IPv6 a patrně ještě větší roli sehrálo i ekonomické hledisko – společnosti raději věnovali své prostředky na vylepšování „starého dobrého“ IPv4 než na vysoké a nevyzpytatelné investice do IPv6. Životnost protokolu IPv4 prodloužilo a IPv6 na okraj zájmu odsunulo především zavedení beztřídního adresování (CIDR) a překladu adres (NAT).

Ani tyto metody ovšem nemohou zabránit úplnému vyčerpání IPv4 adres. S ubývajícími adresami se tak IPv6, jako konečné řešení tohoto problému, opět dostává do popředí zájmu. Dnes si IPv6 může vyzkoušet prakticky každý kdo má připojení k Internetu – k dispozici je celá řada různorodých přechodových metod včetně těch, které umožňují připojení k IPv6 uzlům bez nutnosti jakéhokoliv zásahu či konfigurace ze strany uživatele. Přesto se ale jedná o spojování dvou nekompatibilních „světů“ a z toho, že přechodové metody potřebují pro svou činnost IPv4 vyplývá jedno jasné negativum tohoto přístupu – využitím infrastruktury, která používá starší verzi protokolu, se ztrácí podstatná část výhod, které z IPv6 dělají jednodušší a rychlejší protokol, než byl jeho předchůdce.

Nedá se proto jasně říct, zda je IPv6 dobrý či špatný protokol. Je ovšem jisté, že jeho použití je nevyhnutelné – doba do vyčerpání posledních volných IPv4 adres se totiž dnes udává už jen v řádu několika stovek dní a další prodlužování života IPv4 pomocí masivního používání NATu či obchodování s IP adresami nepředstavuje dlouhodobé řešení do budoucnosti.

LITERATURA

[1] BLANCHET, M. – PARENT, F. IPv6 *Tunnel Broker with the Tunnel Setup Protocol (TSP)*. Draft, 2008.

[2] CARPENTER, B. – MOORE, K. *Connection of IPv6 Domains via IPv4 Clouds*. RFC 3056, 2001.

[3] CISCO SYSTEMS. *The Internet Protocol Journal – Volume 9, Number 3* [online]. 2006 [cit 2008-11-5]. Dostupné z: <http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_9-3/ipv6_internals.html>.

[4] DAVIES, J. *Understanding IPv6*. Redmond : Microsoft Press, 2003.

[5] DEERING, S. – HINDEN, R. *Internet Protocol, Version 6 (IPv6) Specification.* RFC 2460, 1998.

[6] HINDEN, R. – DEERING, S. *IP Version 6 Addressing Architecture.* RFC 4291, 2006.

[7] HINDEN, R. – HABERMAN, B. *Unique Local IPv6 Unicast Addresses.* RFC 4193, 2005.

[8] HUITEMA, C. *Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs).* RFC 4380, 2006.

[9] IPV6.COM. *IPv6 Deployment Around The World* [online]. 2008 [cit 2008-12-1]. Dostupné z: <http://www.ipv6.com/articles/deployment/IPv6-Deployment-Status.htm>.

[10] IPV6.CZ. Přechod od IPv4 k IPv6 [online]. 2009 [cit 2009-4-19]. Dostupné z: <https://www.ipv6.cz/Přechod_od_IPv4_k_IPv6>

[11] KOLEKTIV AUTORŮ. *IPv6 Tunnel Broker*. RFC 3053, 2001.

[12] KOZIEROK, CH. *The TCP/IP Guide: A Comprehensive, Illustrated Internet Protocol Reference*. San Francisco : No Starch Press, 2005.

[13] LAMMLE, T. *CCNA: Cisco Certified Network Associate Study Guide*. Indianapolis : Wiley Publishing, 2007.

[14] LOSHIN, P. *IPv6 Theory, Protocol, and Practice Second Edition*. San Fracisco : Elsevier, 2004.

[15] LUPA.CZ. *Články věnované problematice IPv6* [online]. 1999-2009 [cit 2008-11-24]. Dostupné z: < http://www.lupa.cz/n/ipv6/>

[16] SATRAPA, P. *IPv6 - Internetový protokol verze 6*. Praha : CZ.NIC, 2008.

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

SEZNAM PŘÍLOH

Disk CD-ROM