

IPV4 IR IPV6 PROTOKOLŲ SUDERINAMUMO GALIMYBIŲ ANALIZĖ

Regina Misevičienė, Julius Kriukas

Kauno technologijos universitetas, Studentų g. 50, LT-51368 Kaunas
regina.miseviciene@ktu.lt, julius.kriukas@ktu.lt

Anotacija. Didelis interneto tinklo populiarumas lėmė labai spartų vartotojų augimą. Tai iššaukė plačiai naudojamą interneto protokolo IPv4 (Internet Protocol v4) adresų trūkumą, todėl buvo sukurta nauja interneto protokolo versija – IPv6 (Internet Protocol v6). Šiuo metu veikia dvi IP versijos: IPv4 ir IPv6. Dėl didelių skirtumų adresacijos dalyje IPv4 ir IPv6 tinklai yra nesuderinami, todėl teko ieškoti būdų, kaip, tolygiai nesutrikdant interneto tinklo darbo, pereiti nuo IPv4 prie IPv6. Perėjimui nuo vienos versijos prie kitos palengvinti kuriami įvairūs mechanizmai ir strategijos. Šiame darbe atlikta dvigubo steko, 6to4 tunelio bei NAT64 transliacijos mechanizmų palyginamoji analizė, kuri padėjo atskleisti šių mechanizmų ypatybes.

Pagrindiniai žodžiai: protokolai IPv4/ IPv6, dvigubo steko mechanizmas, 6to4 tunelis, NAT64 transliatorius

Įvadas

Interneto protokolas (IP) – tai taisyklių rinkinys, reglamentuojantis kompiuterių adresavimą, duomenų skaidymą į paketus prieš siunčiant, ir surinkimą juos atsiuntus, jų maršrutų parinkimą. Šiuo metu interneto tinkle veikia dvi IP versijos: IPv4 ir IPv6. Kaip tinklo standartas IPv4 versija buvo patvirtinta 1981 metų rugsėjį, ją aprašo RFC791 (Postel, 1981). Ketvirtoji versija IPv4 naudoja 32 bitų adresus. Tokio ilgio adresas turi 2^{32} (apie 4 milijardus) galimų skirtingų kombinacijų. Sparčiai augantis interneto populiarumas lėmė IP adresų trūkumą, todėl buvo sukurta nauja versija – IPv6. Kaip interneto standartas IPv6 buvo patvirtintas 1998 metų gruodį, jį aprašo RFC2460 (Hinden ir kt., 1998). Šioje versijoje adresas koduojamas 128 bitais ir todėl, turi 2^{128} (apie $3,4 \cdot 10^{38}$) galimų adresų variantų. Kiti ne tokie žymūs pakeitimai – IP antraštės supaprastinimas, lankstumo praplėtimams padidinimas bei srautų žymėjimo patobulinimai. Didžiulę adresų erdvę turintis IPv6 suteikia pagrindą kurti IP grindžiamų paslaugų ir pritaikymų inovacijas. IPv6 nauda akivaizdžiausia tada, kai reikia lengvai sujungti į tinklą daug prietaisų ar produktų, kurie turi būti matomi ir tiesiogiai pasiekiami internetu. Dėl didelių skirtumų adresacijos dalyje IPv4 ir IPv6 tinklai nėra suderinami, todėl IETF (Internet Engineering Task Force) organizacija kuria įvairius mechanizmus, palengvinančius perėjimą nuo vienos IP protokolo versijos prie kitos, plačiau tai aprašo RFC4213 (Nordmark ir kt., 2005). Vienas iš mechanizmų, leidžiančių skirtingas IP versijas naudojamiems mazgams užmegzti ryšį yra paketų transliacija. Du pagrindiniai transliacijos mechanizmai dvigubas stekas ir tuneliavimas (6to4 tunelis) plačiai naudojami Lietuvoje (v6.ipv6.lt). Šiuo metu taip pat sparčiai vystomas transliacijos mechanizmas NAT64, leidžiantis transliuoti IP paketus iš šeštos IP versijos į ketvirtą ir atvirkščiai.

Kuris iš paminėtų transliacijos mechanizmų tinkamiausias? Visi minėti mechanizmai turi ir savo privalumų ir trūkumų. Prieš pasirenkant, kurią iš jų naudoti, svarbu išaiškinti jų administravimo sudėtingumo, resursų panaudojimo efektyvumo bei IPv4 ir IPv6 versijų suderinamumo galimybes. Šio straipsnio autoriai, išnagrinėję mokslinius darbus šia tema, pastebėjo, kad dauguma jų aprašo įvairius IPv4 ir IPv6 versijų transliacijos mechanizmus, tačiau darbuose nėra atlikta lyginamoji analizė tarp egzistuojančių mechanizmų. Siekiant išaiškinti IPv4 ir IPv6 suderinamumo galimybes būtina atlikti transliacijos mechanizmų galimybių analizę.

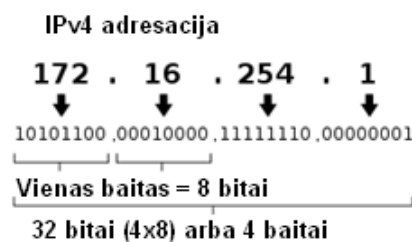
Šio straipsnio tikslas – atlikti IPv4 ir IPv6 protokolų transliacijos mechanizmų palyginamąją analizę. Tuo tikslu reikia išanalizuoti transliacijos mechanizmų privalumus ir trūkumus bei peržiūrėti standartus, susijusius su IPv4 ir IPv6 protokolų transliacijos mechanizmais. Taip pat, siekiant apibendrinti egzistuojančius transliacijos mechanizmus, reikia atlikti eksperimentinius tyrimus realizuojančius 6to4 tunelio, dvigubo steko ir NAT64 veikimą. Šie mechanizmai pasirinkti dėl to, kad naudojami Lietuvos tinkluose (v6.ipv6.lt).

1. IPv4 ir IPv6 nesuderinamumas

1.1. IPv4 ir IPv6 adresacija

Interneto protokolas (IP) tai taisyklių rinkinys, reglamentuojantis kompiuterių adresavimą, duomenų skaidymą į paketus prieš siunčiant ir surinkimą juos atsiuntus, jų maršrutų parinkimą. IP protokolas kartu su TCP protokolu sudaro TCP/IP protokolą, kuris yra pagrindinis interneto protokolas. Jis buvo sukurtas, tam kad būtų galima sujungti skirtingų kompanijų pagamintus LAN ir komutuojamų paketų tinklus.

Dabar daugiausia paplitusi ketvirtoji versija IPv4, naudojanti 32 bitų IP adresus (Postel, 1981). Šio IP protokolo standartas tiksliai apibrėžia, kad kiekvienas pagrindinis kompiuteris turi unikalų 32 bitų numerį, kuris vadinamas pagrindinio kompiuterio tarptinklinio protokolo adresu, (dažnai trumpinamas IP adresas). IP adresas – tai unikalus skaičius tam tikrame tinkle, naudojamas identifikuoti visus tiesiogiai prie interneto prijungtus prietaisus.

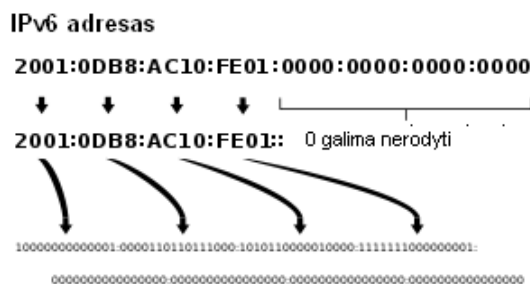


1 pav. IPv4 adresų sudarymas.

IPv4 adresai (1 pav. pagal http://en.wikipedia.org/wiki/IP_address) sudaromi iš keturių, taškais atskirtų skirtingų skaičių, kurių kiekvienas nėra didesnis už 255. Ši sistema adresams sudaryti naudoja 32 bitus (kiekvienas tašku atskirtas skaičius atstoja 8 bitus), todėl teoriškai iš viso galima sudaryti virš 4 mlrd. skirtingų adresų.

Dėl milžiniškos interneto plėtros ir mažėjant laisvų IPv4 adresų kiekiui, 1995 metais buvo sukurta šeštosios kartos IP adresų (IPv6) sistema (Hinden ir kt., 2006). Lyginant su

IPv4, naujoji versija suteikia žymiai didesnę adresų erdvę (adreso ilgis padidėjo nuo 32 bitų iki 128). IPv6 adresai (2 pav. pagal http://en.wikipedia.org/wiki/IP_address) užrašomi nauja forma: tai šešioliktaine skaičiavimo sistema koduojami adresai.



2 pav. IPv6 adresų sudarymas.

Nauja IPv6 versija buvo sukurta tam, kad pakeistų dabar naudojamą IPv4 protokolą ir išspręstų kai kurias su juo susijusias problemas. Pagrindinis naujojo protokolo tikslas yra praplėsti IP adresų erdvę. Be IP adresų erdvės praplėtimo naujasis protokolas turi ir daugiau patobulinimų lyginant su senuoju protokolu (Hinden ir kt., 1998):

- IPv6 praplėtė IP adreso ilgį nuo 32 bitų iki 128 bitų. Dėl to galima turėti daugiau adresavimo hierarchijos lygių, adresai gali būti suteikti žymiai didesniai įrenginių skaičiui.
- Kai kurie IPv4 antraštės laukai tapo nebenaudojami ar neprivalomi, todėl IPv6 antraštės formatas tapo paprastesnis.
- Pridėta galimybė praplėtimams ir parametrų, nes IPv6 paketo antraštėje nėra jiems fiksuotos vietos. Tai leidžia siųsti parametrus tik tuomet kai jų reikia.
- IPv6 leidžia žymėti paketus pagal tai kokiam duomenų srautui jie priklauso. Taip maršrutizatoriai gali lengvai skirstyti srautus į skirtingų prioritetų klases.
- IPv6 pridėti papildomi mechanizmai, leidžiantys įrenginiams automatiškai atrasti maršrutizatorius ir prisiskirti sau adresą. Tam nebereikalingi papildomi įrenginių konfigūravimo servais.

Dėl didelių skirtumų adresacijos dalyje IPv4 ir IPv6 tinklai nėra suderinami, todėl IETF (Internet Engineering Task Force) organizacija kuria įvairius mechanizmus palengvinančius perėjimą nuo vienos IP protokolo versijos prie kitos.

1.2. Transliacijos mechanizmų panaudojimo mokslinių darbų apžvalga

Šiame straipsnyje nagrinėti plačiai naudojami transliacijos mechanizmai: dvigubas stekas, 6to4 tunelis ir šiuo metu taip pat sparčiai vystomas NAT64 transliatorius.

Siekiant geriau išsiaiškinti transliacijos mechanizmus bei jų veikimo principus buvo išnagrinėti standartai, susiję su dvigubo steko mechanizmu, 6to4 tuneliu ir NAT64 transliacija. Dokumentas (Postel, 1981) aprašo modelį, funkcijas, ryšį su kitais protokolais, tinklų sąsają. Standartas (Hinden ir kt., 1998) nagrinėja IPv6 protokolo antraštės formatą, išplėtimus, paketo dydį, viršutiniame lygmenyje veikiančio protokolo aspektus. Specifikacija (Hinden ir kt., 2006) apibrėžia adresavimo architektūrą IPv6 versijos (IPv6) protokolui, apibrėžiami *unicast*, *anycast* ir *multicast* adresai. Standarte (Bagnulo ir kt., 2011)

yra aprašytas DNS64 mechanizmas, sintezuojantis AAAA įrašus iš A įrašų, pateikiamas generavimas iš IPv6 į IPv4. Dokumente (Bao ir kt., 2010) aptariama algoritminė IPv6 transliacija į atitinkamą IPv4, ir atvirkščiai. Dokumentas (Baker ir kt., 2010) apibūdina IPv4/IPv6 transliavimo sistemą, pakeičiant NAT-PT, siekiant kad tinklai IPv4 ir IPv6 kartu veiktų racionaliai. Standarte (Nordmark ir kt., 2005) nurodomi IPv4 suderinamumo mechanizmai, kurie gali būti įgyvendinti IPv6 pagrindiniuose kompiuteriuose ir maršrutizatoriuose. Dokumentas (Kent, 2005) aprašo atnaujintą IP Authentication Header (AH) versiją, kuri skirta užtikrinti autentifikavimą IPv4 ir IPv6.

Siekiant atskleisti platesnes galimybes tarp IPv4 ir IPv6 tinklų, taip pat buvo nagrinėti ir mokslinės literatūros šaltiniai. Autoriai (Gnana ir kt., 2010) aprašo IPv4 ir IPv6 architektūrą ir jų skirtumus, o straipsnis (Wing, 2010) apžvelgia IPv4 ir IPv6 transliacijos mechanizmus ir jų standartus. Kitame darbe (Xiang ir kt., 2011) nagrinėjamos IPv4/IPv6 mobilumo problemos, aprašomas MIPv4/v6-TG transliacijos mechanizmas, o autoriai (Hodzic ir kt., 2012) aprašo eksperimentinį tyrimą panaudojant DNS64/NAT64 transliatorių. Straipsnyje (Vu ir kt., 2012) pateikiamas dviejų transliacijos mechanizmų OpenVPN and IVP1 palyginimas, o darbe (Choi ir kt., 2010) aprašomas komunikavimo metodas 6LoWPAN tarp tinklo ir išorės mazgų. Autoriai (Lin ir kt., 2010) apžvelgia SSN (Smart Sensor Networks) ir iliustruoja ryšį tarp SSN ir IPv4 tinklų, tam pateikdami autokonfigūravimo savybę. Straipsnyje (AlJa'afreh ir kt., 2009) aprašoma transliacija „v4-to-v4 ir v6-to-v6“ panaudojant OMNeT++ simuliacijos platformą.

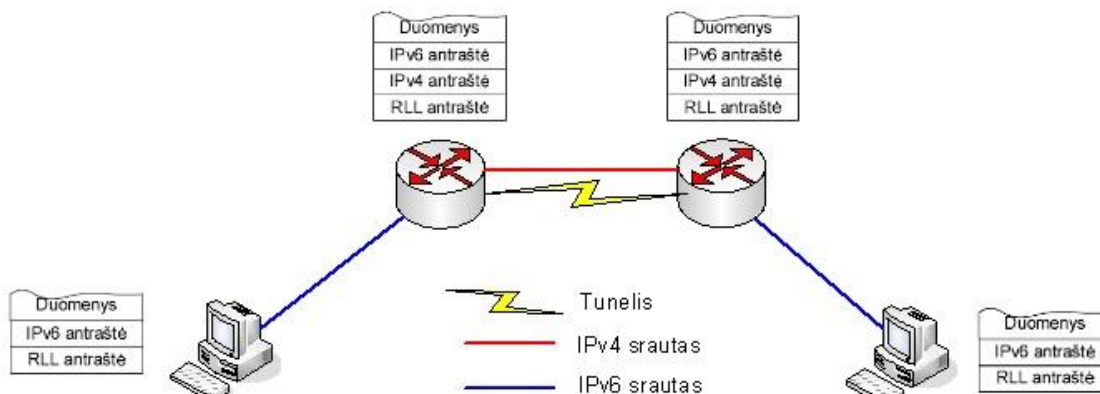
Nors išnagrinėti moksliniai straipsniai aprašo įvairius transliacijos mechanizmus, tačiau juose nėra palyginamosios galimybių analizės tarp naudojamų mechanizmų.

2. Transliacijos mechanizmai ir jų savybių tyrimas

Siekiant apibendrinti egzistuojančius transliacijos mechanizmus, taip pat buvo atlikti eksperimentiniai tyrimai realizuojantys 6to4 tunelį, dvigubo steko mechanizmą ir NAT64 transliatoriaus veikimą. Šie mechanizmai pasirinkti dėl to, kad naudojami Lietuvos tinkluose.

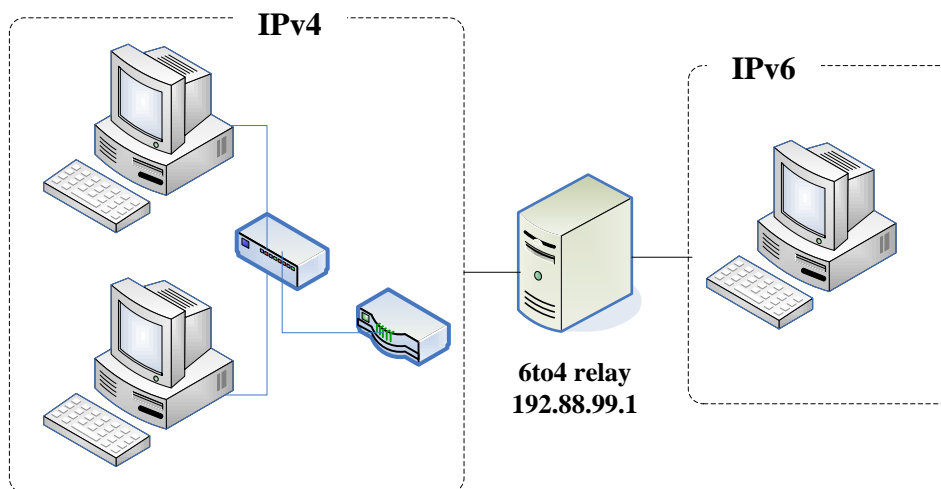
2.1. 6to4 tunelio konfigūravimas

Tuneliavimas – tai procesas, kada vieno kompiuterinio tinklo protokolo paketo informacija yra pernešama kitame pakete. Šis procesas dar yra vadinamas enkapsuliacija. Pagrindinė 6to4 tunelio idėja yra ta, kad prie IPv6 tinklo galima prisijungti persiunčiant paketus per IPv4 tinklą iki artimiausio mazgo galinčio perkelti paketus į IPv6 tinklą. Mazgas atliekantis šį perkėlimą vadinamas 6to4 relay. Tuneliuojant IPv6 paketas yra įpakuojamas į IPv4 paketą viename tunelio gale ir išpakuojamas kitame tunelio gale. Įpakuojant IPv6 paketas įdedamas į IPv4 paketą kaip naudingi duomenys (žr. 3 pav., šaltinis: <http://www.tinklusaugumas.lt/>).



3 pav. Duomenų perdavimas per tunelį.

Ekspimentinio tyrimo schema realizuojanti 6to4 tunelį pavaizduota paveiksle (žr. 4 pav.). Šiame paveiksle pavaizduotas 6to4 tunelis, kai konfigūruojamas vidinis tinklas, o galinis IPv4 tinklo mazgas (maršrutizatorius) turi viešą adresą. Įrenginiui naudojančiam 6to4 tunelį norint išsiųsti paketus į IPv6 tinklą, paketai pirmiausia įpakuojami į IPv4 paketus ir per IPv4 tinklą persiunčiami iki artimiausio 6to4 *relay*. Įrenginiai atliekantys 6to4 *relay* darbą nukabina IPv4 antraštę ir išleidžia IPv6 paketus į IPv6 tinklą. Taip pat iš IPv6 tinklo išsiųstas paketas, kurio gavėjas naudoja 6to4 tunelį pirmiausia pasiekia 6to4 *relay*, kuris prikabina IPv4 antraštę ir perduoda paketą gavėjui IPv4 tinklu.



4 pav. 6to4 tunelio eksperimento schema.

Ekspimente buvo panaudoti du serveriai, kuriuose įdiegtos skirtingos OS (Linux ir Windows XP) operacinės sistemos. Panaudotos tik tokios operacinės sistemos, kurias galėjome instaliuoti nemokamai. Sąveika iš kompiuterių tinklo iki IPv6 tinklo vykdoma per 6to4 *relay* specialų *anycast* tipo adresą 192.88.99.1. Tai realus viešas IPv4 adresas, kurį užsideda visi 6to4 *relay* įrenginiai.

Ekspimente tiriamos 6to4 tunelio konfigūravimo darbų apimtys.

Konfigūravimas Linux operacinėje sistemoje. Tyrimams reikėjo išjungti Native IPv6, (kadangi Linux sistemoje IPv6 adresai automatiškai sudaromi pagal IPv4 adresus) ir tik po to, atlikti eksperimentus tik su Native IPv4 adresais.

Pirmiausia įjungiamas, sukonfigūruojamas tunelis ir atliekami eksperimentai. Po to tunelis išjungiamas. Tai buvo atlikta su tokiomis komandomis:

Native IPv6 išjungimas:

```
root@linux:~# echo 0 > /proc/sys/net/ipv6/conf/eth0/autoconf  
root@linux:~# ifdown eth0; ifup eth0
```

Native IPv6 įjungimas:

```
root@linux:~# echo 1 > /proc/sys/net/ipv6/conf/eth0/autoconf  
root@linux:~# ifdown eth0; ifup eth0
```

6to4 tunelio įjungimas:

```
root@linux:~# ip tunnel add tun6to4 mode sit remote any local 158.129.0.86  
root@linux:~# ip link set dev tun6to4 up  
root@linux:~# ip -6 addr add 2002:9e81:0056::1/6 dev tun6to4  
root@linux:~# ip -6 route add 2000::/3 via ::192.88.99.1 dev tun6to4
```

6to4 tunelio sunaikinimas:

```
root@linux:~# ip tunnel del tun6to4
```

Sukonfigūravus tunelį, tikrinama ar pasiekiamas svetainė v6.ipv6.lt, kuri naudoja IPv6 adresus. Pradžioje IPv6 svetainė v6.ipv6.lt yra pasiekiamas, nes tiriama kompiuteris su Linux operacine sistema turi automatiškai sukonfigūruotus IPv4 ir IPv6 adresus. Išjungus Native IPv6 svetainė nepasiekiamas (nes, eksperimentinis kompiuteris turi tik IPv4 adresą). Po to, tik sukonfigūravus tunelį, svetainė tapo v6.ipv6.lt pasiekiamas.

Konfigūravimas Windows XP operacinėje sistemoje (pagal ipv6int.net). Konfigūravimas atliekamas kitame kompiuteryje, kuriame įdiegta Windows XP operacinė sistema. Kadangi Windows XP operacinėje sistemoje IPv6 protokolas automatiškai neįjungiamas, tai reikėjo padaryti rankiniu būdu.

6to4 tunelio įjungimas:

Įrenginių konfigūravimui reikia: IPv4 adresas: 158.129.0.87; IPv6 6to4 adresas: 2002:9e81:0057:: ir 6to4 gateway: ::192.88.99.1

Tunelis įjungiamas taip:

```
ipv6 rtu 2002::1/6 2 (įjungiamas 6to4 palaikymas)
```

```
ipv6 adu 2/2002:9e81:0057::
```

```
ipv6 rtu ::/0 2::192.88.99.1 pub
```

Tunelis panaikinamas:

```
ipv6 rtu ::/0 7/2001:470:19:13c::1 life 0
```

Svetainė v6.ipv6.lt nepasiekiamas ir po tunelio įjungimo. Sukonfigūruoti nepavyko.

Atlikus 6to4 tunelio konfigūravimo darbus pastebėti tokie konfigūravimo ypatumai:

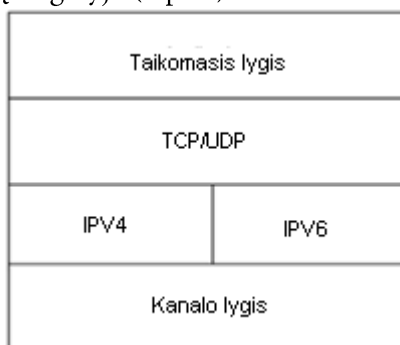
- Konfigūracija gana sudėtinga, nes reikia konfigūruoti kiekvieną įrenginį. Šis 6to4 konfigūravimo darbas atliekamas kiekviename įrenginyje, todėl konfigūravimo darbas tampa sudėtingas bei priklauso nuo naudojamų įrenginių skaičiaus. Konfigūracija yra paprastesnė, kai 6to4 mechanizmas naudojamas įrenginyje, kuris teikia IPv6 ryšį vidiniam tinklui. Tokiu atveju 6to4 tunelis konfigūruojamas tik viename (galiniame) įrenginyje per vieną tinklą.
- Maršrutas ne visada optimalus, nes reikalingas specialus mazgas, šiuo atveju 6to4 relay įrenginys, per kurį pasiekiamas IPv6 tinklas per IPv4 tunelį. Dėl to, 6to4 kitas tunelio galas nebūtinai yra trumpiausiam kelyje tarp siuntėjo ir

gavėjo. Dėl šios priežasties paketai ne visada keliauja trumpiausiu keliu ir taip labiau apkrauna bendrą tinklą.

- Resursai naudojami neefektyviai, nes didėja paketai (enkapsuliuojami) ir pridedama IPv4 antraštė. Per tunelį siunčiami IPv6 paketai yra įvelkami į IPv4 paketus, todėl prie bendro išsiunčiamų ir gaunamų duomenų srauto papildomai prisideda IPv4 antraštės.
- Kadangi 6to4 tunelis naudoja IPv4 kaip ryšio linijos sluoksnį, todėl jis priklauso nuo IPv4 tinklo funkcionalumo. Naudojant 6to4 tunelį įrenginys tinkle gauna IPv6 adresą priklausomą nuo tuneliui sudaryti naudojamo IPv4 adreso, todėl pasikeitimas IPv4 tinkle įtakoja ir gaunamo IPv6 adreso pasikeitimą.

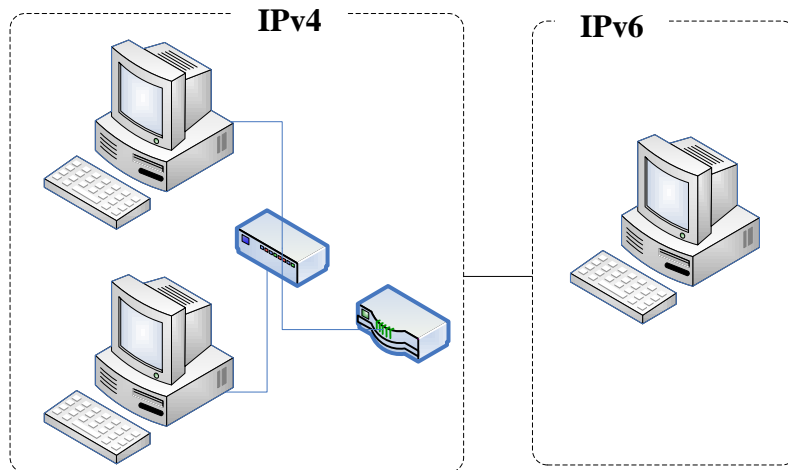
2.2. Dvigubo steko konfigūravimas

Pagrindinė šio metodo idėja sukonfigūruoti kiekvieno įrenginio tinklo nustatymus abiem tinklams (IPv4 ir IPv6). Dvigubo steko mechanizmas yra taisyklių rinkinys įrenginyje nurodantis kaip įrenginys turėtų veikti, kad abi IP versijos galėtų netrukdydamos viena kitai dirbti tame pačiame įrenginyje (5 pav.).



5 pav. Dvigubo steko mechanizmo adresacija.

Ekspimentinio tyrimo schema, realizuojanti dvigubo steko transliaciją, pavaizduota paveiksle (žr. 6 pav.). Eksperimente buvo panaudoti tie patys serveriai su skirtingomis OS: Linux ir Windows XP. IPv6 adresai kiekvienam kompiuteriui suteikiami tokia tvarka: IPv6 adresai automatiškai suteikiami pagal jų IPv4 adresą. Įrenginiuose su Linux operacine sistema ir turint viešą IPv4 adresą sistema automatiškai sukonfigūruoja IPv6 adresus. Windows XP reikia įjungti 6to4 tunelio palaikymą. Tinklo atveju kadangi vienas įrenginys turintis viešą unikalų IPv4 adresą gauna /48 ilgio IPv6 prefiksą, jis visus šios erdvės adresus gali toliau maršrutizuoti savo tinkle. /48 ilgio prefiksas apima $2^{28} = 280$ IPv6 adresus. Dažniausiai vietiniam IPv6 tinklui išskiriami /64 ilgio prefiksai. Todėl turint vieną įrenginį, kuris prie IPv6 prijungtas per 6to4 tunelį, galima papildomai turėti dar $2^{28} - 2^{16} = 65536$ IPv6 tinklus, kurių prefikso ilgis yra /64 (pagal šaltinį v6.ipv6.lt).



6 pav. Dvigubo steko testavimo aplinkos schema.

Ekspimente tiriamos dvigubo steko konfigūravimo darbų apimtys.

Konfigūravimas Linux operacinėje sistemoje. Linux operacinėje sistemoje adresus IPv6 kiekvienam kompiuteriui sukongūruojamas automatiškai pagal IPv4 adresą. Po to tikrinami adresai su komanda *ifconfig*.

```
root@linux:~# ifconfig
eth0  Link encap:Ethernet HWaddr 00:50:56:b6:17:9c
      inet addr:158.129.0.86 Bcast:158.129.1.255 Mask:255.255.254.0
      inet6 addr: 2001:778:200:4280:250:56ff:feb6:179c/64 Scope:Global
      inet6 addr: fe80::250:56ff:feb6:179c/64 Scope:Link
```

Testuojama ar pasiekiamas svetainė v6.ipv6.lt. Tai atliekama *ping* komanda.

Konfigūravimas Windows XP operacinėje sistemoje pagal šaltinį *ipv6int.net*). Pagal nutylėjimą IPv6 nėra instaliuotas. Tai reikia atlikti rankiniu būdu komandinėje eilutėje: *netsh interface ipv6 install*. Sukongūravus dvigubo steko mechanizmą, svetainė v6.ipv6.lt tapo pasiekiamas.

Atliekant dvigubo steko mechanizmo konfigūravimą, buvo pastebėta, kad mechanizmas labiausiai paplitęs ir efektyviai naudoja tinklo resursus, priklausomybės nuo IPv4 nėra, o suderinamumas puikus, tačiau konfigūracija yra gana sudėtinga ir reikalaujanti daug administratoriaus pastangų, nes reikia konfigūruoti kiekvieną įrenginį.

2.3. NAT64 transliacija

Didžioji dalis migraciją palengvinančių mechanizmų tarp IPv4 ir IPv6 tinklų buvo orientuoti į pirmąją migracijos stadiją, kada buvo siekiama kuo greičiau ir kuo lengviau praplėsti IPv6 tinklą - pavyzdžiui IPv6 autokongūracija, 6to4 tunelis ar dvigubo steko konfigūracija. Šių mechanizmų pagalba turėjo būti sukurtas naujas IPv6 tinklas lygiavertis IPv4 tinklui. NAT64 mechanizmas yra skirtas vėlesniam migracijos etapui, kai pradės daugėti tik prie IPv6 tinklo prijungtų kompiuterių. Šio mechanizmo tikslas leisti šiems tik IPv6 tinklą pasiekiantiems įrenginiams komunikuoti su tik IPv4 tinklą pasiekiančiais įrenginiais.

NAT64 veikimo principas labai panašus į įprastą IPv4 NAT44 mechanizmą. Kaip

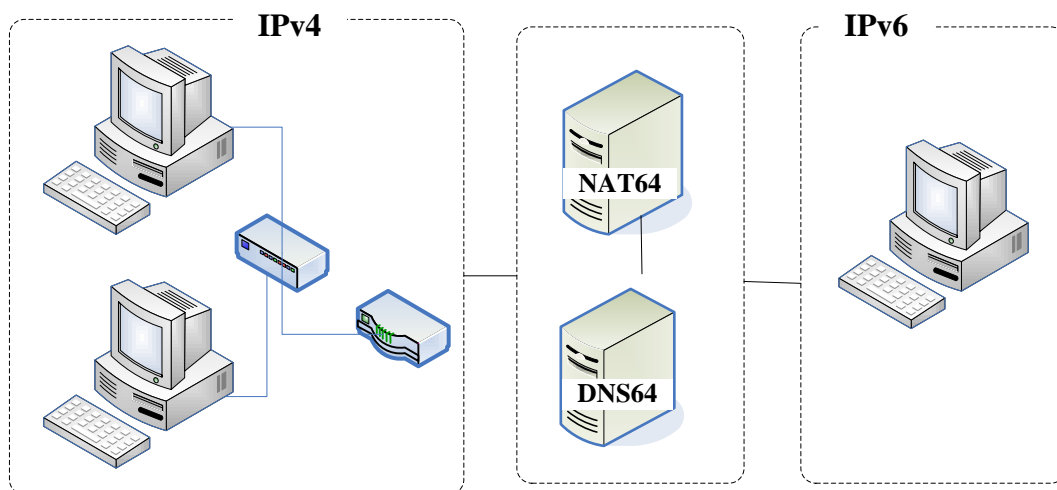
įprastas NAT44 transliuoja IPv4 paketus iš privataus tinklo adresų erdvės į viešų IP adresų erdvę, taip įrenginys atliekantis NAT64 transliuoja IPv6 paketus į IPv4 paketus ir atvirkščiai. Priešingai nei tuneliuose paketai keliaujantys tarp šių tinklų nėra įpakuojami į papildomus protokolus, o transformuojami/transliuojami. Šiuo metu Lietuvoje yra žinoma viešai prieinama NAT64 realizacija: Linux NAT64 - tai KTU ITPI sukurta eksperimentinė NAT64 realizacija, skirta Linux operacinei sistemai (<http://ipv6.lt/nat64.php>).

Eksperimentams atlikti ir buvo naudojama KTU ITPI LITNET tinkle sukurta eksperimentinė NAT64 realizacija. LITNET tinkle yra paleisti įrenginiai teikiantys NAT64/DNS64 paslaugas. Tam LITNET tinkle yra įrengti šie serveriai ir išskirti IP adresų ruožai (<http://ipv6.lt/nat64.php>):

- nat64.litnet.lt (2001:778::75) - serveris atliekantis NAT64 transliavimą;
- krioklys.litnet.lt (2001:778::37) - DNS serveris atliekantis DNS64 funkciją;
- 2001:778:0:ffff:64::/96 - IPv6 NAT64 prefiksas;
- 193.219.61.0/24 - IPv4 NAT64 prefiksas.

Priešingai nei įprastas IPv4 NAT, NAT64 gali būti naudojamas ne tik iš vietinio tinklo, bet iš bet kurio kompiuterio visame IPv6 tinkle. Išbandant LITNET NAT64 transliatorių, tereikia savo kompiuteryje vietoje pagrindinio DNS serverio nurodyti LITNET DNS64 serverį (2001:778::37). Šis DNS serveris automatiškai vietoje visų IPv4 adresų gražina sugeneruotą IPv6 adresą iš LITNET NAT64 IPv6 prefikso ir siunčiami duomenys bus automatiškai nukreipiami į LITNET NAT64 transliatorių.

Eksperimente tiriamos NAT64 mechanizmo konfigūravimo darbų apimtys. Tyrimas vyko tarp kompiuterio veikiančio IPv6 tinkle ir kompiuterių veikiančių IPv4 tinkle. Testavimo aplinkos schema pateikta 7 pav.



7 pav. NAT64 testavimo aplinkos schema.

Konfigūravimas Linux operacinėje sistemoje. Linux operacinėje sistemoje faile *resolv.conf* nurodomas DNS serveris atliekantis DNS64 funkciją *nameserver 2001:778::37*

Po to, tikrinami adresai su komanda *ifconfig*:

```
root@linux:~# ifconfig
eth0    Link encap:Ethernet HWaddr 00:50:56:b6:17:9c
        inet addr:158.129.0.86 Bcast:158.129.1.255 Mask:255.255.254.0
```

```

inet6 addr: 2001:778:200:4280:250:56ff:feb6:179c/64 Scope:Global
inet6 addr: fe80::250:56ff:feb6:179c/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:3408083 errors:0 dropped:0 overruns:0 frame:0
TX packets:9263320 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:348960830 (332.7 MiB) TX bytes:13678726154 (12.7 GiB)
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:35 errors:0 dropped:0 overruns:0 frame:0
TX packets:35 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:4974 (4.8 KiB) TX bytes:4974 (4.8 KiB)
    
```

Buvo testuojama ar pasiekiamą svetainė v6.ipv6.lt (iš IPv6 tinklo). Po NAT64 transliacijos svetainė tapo pasiekiamą.

Windows XP sistemai NAT64 transliacijos mechanizmo sukonfigūruoti nėra galimybės.

Tyrimas parodė, kad NAT64 mechanizmo konfigūravimas reikalauja mažai darbo, nes vienai autonominei sistemai ar tinklui pakanka sukonfigūruoti vieną transliacijos įrenginį. Resursai naudojami beveik taip pat efektyviai kaip ir dvigubo steko konfigūracijoje. Priešingai nei 6to4 tunelis, NAT64 mechanizmas ne prideda papildomą antraštę, o ją pakeičia kitos versijos antrašte. Trūkumas yra tas, kad visi į IPv4 tinklą išeinantys paketai turi keliauti per NAT64 transliatorių. Be to, NAT64 transliatoris priklauso nuo DNS64 mechanizmo, nes IPv4 adresų pakeitimą į IPv6 adresus atlieka DNS64 serveris. Todėl šis mechanizmas gali veikti tik su programomis, kurios kitų mazgo adreso parinkimui naudoja DNS.

3. Lyginamoji transliacijos mechanizmų analizė

Pagal tyrimų rezultatus apibendrintos transliacijos mechanizmų (dvigubo steko, 6to4 tunelio ir NAT64) ypatybės: konfigūravimo paprastumas, resursų panaudojimo efektyvumas, priklausomybė nuo IPv4 ir suderinamumas (žr. 1 lentelė).

Konfigūravimo paprastumas. Tiriamos konfigūravimo darbų apimtys.

Tinklo resursų panaudojimo efektyvumas. Analizuojama ar efektyviai naudojami tinklo resursai.

IPv4 ir IPv6 priklausomybė. Analizuojama kaip priklauso vieno protokolo darbas nuo kito.

Suderinamumas. Analizuojama kaip yra suderinami abu protokolai.

1 lentelė. Transliacijos mechanizmų palyginimas

Kriterijai	Dvigubas stekas	6to4 tunelis	NAT64
Konfigūravimo paprastumas	Sudėtingas, reikalingas kiekvienam įrenginiui	Paprastesnis, jei konfigūruojamas visas tinklas, o ne kiekvienas įrenginys atskirai	Konfigūruojamas vienas įrenginys visam tinklui

Kriterijai	Dvigubas stekas	6to4 tunelis	NAT64
Resursų panaudojimo efektyvumas	Geras	Mažiau efektyvus, didesni paketai ir ne parenkamas trumpiausias maršrutas	Paketai į IPv4 tinklą privalo keliauti per NAT64 transliatorių.
IPv4 ir IPv6 priklausomybė	Nėra	Tiesiogiai priklauso nuo IPv4 funkcionavimo, bei konfigūracijos pasikeitimo.	Nėra
Suderinamumas	Apribojimų nėra	Apribojimų nėra	Tinka tik taikymams, naudojantiems DNS, o ne IP adresus.

Išvados

Mokslinės literatūros šaltinių apžvalga ir standartų analizė leido išsiaiškinti IPv4 ir IPv6 protokolų struktūrą, esminius skirtumus, konfigūravimo aspektus ir protokolų taikymo galimybes. Nors išnagrinėti moksliniai straipsniai aprašo įvairius transliacijos mechanizmus, tačiau juose nėra palyginamosios galimybių analizės tarp naudojamų mechanizmų.

Atlikti eksperimentiniai tyrimai atskleidė atskirų transliacijos mechanizmų ypatybes. Dvigubo steko mechanizmas labiausiai paplitęs ir efektyviai naudoja tinklo resursus, priklausomybės nuo IPv4 nėra ir suderinamumas puikus, tačiau konfigūracija yra gana sudėtinga ir reikalaujanti daug administratoriaus pastangų. 6to4 mechanizmas konfigūravimo atžvilgiu paprastesnis, tačiau yra mažiau efektyvus ir labiau apkrauna tinklą. Nat64 lengvai konfigūruojamas, tačiau priklauso nuo DNS64 mechanizmo.

Šių galimybių įvertinimas gali padėti parenkant konkrečius realizacijos sprendimus.

Literatūra

- Alja'afreh, R., Mellor, J., Awan, I. (2009). A Comparison between the tunneling process and mapping schemes for IPv4/IPv6 transition. *Advanced Information Networking and Applications*, vols. 1-2: 601-606.
- Bagnulo, M., Sullivan, A., Matthews, P., Beijnum I. (2011). DNS64: DNS extensions for Network Address Translation from IPv6 Clients to IPv4 Servers. <http://tools.ietf.org/pdf/draft-ietf-behave-dns64-10.pdf>.
- Baker, F., Li, X., Bao, C., Yin K. (2010). Framework for IPv4/IPv6 Translation. <http://tools.ietf.org/pdf/draft-ietf-behave-v6v4-framework-09.pdf>.
- Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., Li X. (2010). IPv6 Addressing of IPv4/IPv6 Translators. <http://tools.ietf.org/pdf/draft-ietf-behave-address-format-09.pdf>.
- Choi, D., Kim, M., Park, Ch. (2012). An Efficient Global Communication Method for 6LoWPAN. *Information-an international interdisciplinary journal*, vol. 15: 1949-1960.
- Gnana J. J., Rabara, S. A. (2010). IPv4 Addressing Architecture in IPv6 Network. 2nd IEEE international conference on advanced computer control (ICACC), Vol. 3: 282-287.
- Hinden, R., Deering S. (1998). Internet Protocol, Version 6 (IPv6) Specification. <http://tools.ietf.org/html/rfc2460>
- Hinden R., Deering S. (2006). IP Version 6 Addressing Architecture. <http://tools.ietf.org/pdf/rfc4291.pdf>.
- Hodžic, E., Mrdović, S. (2012). Transition Using DNS64/NAT64: Deployment Issues. *Telecommunications (BIHTEL)*, 9th International Symposium on IEEE: 1-6.
- Kent S. IP Authentication Header (2005). <http://tools.ietf.org/pdf/rfc4302.pdf>.

- Lin, H., Wu, Q. (2010). Auto-configuration Support for IPv4/IPv6 Translation in Smart Sensor Networks. Computational Collective Intelligence: Technologies and Applications, Lecture Notes in Artificial Intelligence, vol. 6423: 302-310.
- Nordmark E., Gilligan R. (2005). Basic Transition Mechanisms for IPv6 Hosts and Routers. <http://tools.ietf.org/pdf/rfc4213.pdf>.
- Postel, J. Internet protocol (1981). <http://tools.ietf.org/pdf/rfc791.pdf>.
- Vu, C., Tran, Q., Jiang, F. (2012). A Comparison of Two IPv4/IPv6 Transition Mechanisms - OpenVPN and IVI. AIP Conference Proceedings, vol. 1479: 1532-1538.
- Wing, D. (2010). Network Address Translation: Extending the Internet Address Space. IEEE Internet Computing, vol., 14, issue 4: 66-70.
- Xiang, Z., Ma, Z. (2012). Mobility Support in IPv4/v6 Network. Proceedings of the 2011 2nd international congress on computer applications and computational science. Advances in Intelligent and Soft Computing, Vol. 145: 125-131.

IPV4 AND IPV6 PROTOCOL COMPATIBILITY OPTIONS ANALYSIS

Regina Misevičienė, Julius Kriukas

Summary

The popularity of the internet has led to a very rapid growth of IPv4 (Internet Protocol v4) users. This caused a shortage of IP addresses, so it was created a new version – IPv6 (Internet Protocol v6). Currently, there are two versions of IP for IPv4 and IPv6. Due to the large differences in addressing the protocols IPv4 and IPv6 are incompatible. It is therefore necessary to find ways to move from IPv4 to IPv6. To facilitate the transition from one version to another are developed various mechanisms and strategies. Comparative analysis is done for dual stack, 6to4 tunnel and NAT64 mechanisms in this work. It has helped to reveal the shortcomings of these mechanisms and their application in selection of realization decisions.

Key words: IPv4 / IPv6 dual stack mechanism, 6to4 tunnel, NAT64 translator