André Manoel da Silveira

Rede IPv6 com Integração IPv4

São José – SC
Julho/2012

André Manoel da Silveira

Rede IPv6 com Integração IPv4

Monografia apresentada à Coordenação do Curso Superior de Tecnologia em Sistemas de Telecomunicações do Centro Federal de Educação Tecnológica de Santa Catarina para a obtenção do diploma de Tecnólogo em Sistemas de Telecomunicações.

Orientador:
Prof. Marcelo Maia Sobral

CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS DE TELECOMUNICAÇÕES
CENTRO FEDERAL DE EDUCACAO TECNOLOGICA DE SANTA CATARINA

São José – SC
Julho/2012

Monografia sob o título “Rede IPv6 com Integração IPv4”, defendida por André Manoel da Silveira. e aprovada em 13 de Julho de 2012, em São José, Santa Catarina, pela banca examinadora assim constituída:

_____

Prof. Marcelo Maia Sobral
Orientador

_____

Prof. Eraldo Silveira e Silva
IFSC/SJ

_____

Prof. Ederson Torresini
IFSC/SJ

Sempre que te perguntarem se podes fazer um trabalho, respondas que sim e te ponhas em seguida a aprender como se faz.

F. Roosevelt

Agradecimentos

Dedico meus sinceros agradecimentos àqueles que muito me ajudaram para concluir este trabalho. Com certeza essas pessoas tornaram a realização deste trabalho uma tarefa prazerosa.

Resumo

Este TCC (Trabalho de Conclusão de Curso) tem como finalidade investigar um cenário de transição entre os protocolos IPv4 e IPv6, no qual uma subrede IPv6 deve ser capaz de acessar e ser acessada por uma subrede IPv4. Mais especificamente, este trabalho trata de subredes IPv6 que se apresentam como ilhas dentro de uma Internet majoritariamente IPv4. Foram avaliadas a técnica de transição de pilha dupla, em que todos os equipamentos da subrede IPv6 possuem ambas pilhas de protocolos, a técnica denominada NAT64/DSN64, que se assemelha ao NAT usado em redes IPv4, e por fim a técnica IVI, que mapeia blocos de endereço IPv4 para IPv6. Os experimentos realizados possibilitaram concluir em que cenários cada técnica pode ser utilizada.

Abstract

This work investigates a scenario related to the foreseen transition from IPv4 to IPv6, where a IPv6 network must access and be accessed by a IPv4 network. More specifically, it is considered a IPv6 network as an island within the Internet which is mostly IPv4. It was evaluated the *dual stack* transition mechanism, where both protocol stacks are available in every host of the IPv6 network, the NAT64/DNS/64 mechanism, which ressembles NAT in IPv4 networks, and the IVI mechanism, which maps IPv4 to IPv6 address blocks. A set of experiments were performed to conclude upon in which scenarios each mechanism can be applied.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

A Internet é uma rede que interconecta outras redes de computadores ao redor do mundo. Um componente fundamental dessa rede é o protocolo IP (Internet Protocol), responsável por atribuir aos computadores que compõem essas redes um endereço ”único” que possibilita a comunicação entre eles.

O protocolo IP usado atualmente é o IPv4 (*Internet Protocol version 4*). Cada endereço IPv4 é composto por um número de 32 bits, portanto existem pouco mais de 4 bilhões de endereços. Esses endereços são divididos e organizados pela IANA (*Internet Assigned Numbers Authorithy*). Porém os endereços IPv4 estão se esgotando, e logo não haverá mais endereços disponíveis para criação de novas redes.

Para resolver o problema do esgotamento de endereços IPv4 foi criado o IPv6 (*Internet Protocol version 6*). Nesta nova versão do protocolo IP, os endereços são compostos por 128 bits, gerando assim uma quantidade virtualmente ilimitada de endereços IP (2^{128} endereços, ou aproximadamente 3,4 x 10^{38}). Os endereços IPv6, assim como Ipv4, são regulados pela IANA e suas divisões hierárquicas regionais, as RIRs (*Regional Internet Registries*). As RIRs por sua vez executam a distribuição dos endereços em âmbito regional geográfico. O Brasil, por exemplo, é atendido pela LACNIC (*Latin America and Caribbean Network Information Centre*).

Nos dias atuais, o IPv6 vem sendo adotado gradualmente, porém quase que unicamente por órgãos públicos e outras entidades ligadas a Internet. Em diversos países, o uso do IPv6 já é incentivado pelos governos que constroem suas redes baseadas em IPv6 e oferecem incentivos fiscais para provedores que migrarem para o IPv6.

Com o intuito de avaliar mecanismos de convivência Ipv4/IPv6 no cenário considerado, sendo que “convivência” significa a comunicação transparente entre ilhas IPv6 e demais redes em uma Internet majoritariamente IPv4, deseja-se neste TCC criar uma rede IPv6 capaz de conviver com as redes IPv4 já existentes. Para esta finalidade

serão focadas técnicas que possibilitem essa convivência, são elas: Pilha Dupla (Dual stack) e Tradução (NAT64/DNS64 e IVI). Essas técnicas serão testadas através de maquinas virtuais, onde uma rede IPv6 conviverá com uma Internet majoritariamente IPv4.

1.1 Objetivo

Este TCC tem como finalidade estudar o IPv6 em um cenário de transição num momento onde o IPv4 ainda é dominante, mas onde o uso do IPv6 se torna cada vez mais necessário em vista da escassez de endereços IPv4. Para entender este cenário de transição serão estudadas técnicas que possibilitam a convivência do IPv4 e do IPv6 em um mesma rede.

A partir de uma rede puramente IPv6, métodos para a convivência com outras redes IPv4 foram estudados, possibilitando então que a rede IPv6 possa se comunicar e compartilhar serviços com as redes IPv4. Essas técnicas de convivência entre os protocolos IP serão avaliadas para definir qual técnica terá melhor desempenho no cenário proposto.

1.2 Organização do texto

O texto é dividido em seções que representam a evolução do estudo sobre o caso no capítulo 2 podem ser observadas as referências bibliográficas usadas para entender e aplicar as técnicas de convivência entre os protocolos IPv4 e IPv6 já no capítulo 3 são detalhados os cenários e as necessidades de implantação de cada técnica. Ainda no capítulo 3 são apresentados os testes e resultados obtidos com o uso das técnicas de convivência entre os protocolos. E por fim no capítulo 4 são feitas as considerações finais sobre o trabalho.

2 Fundamentação teórica

2.1 O IPv6

A evolução da arquitetura TCP/IP sempre esteve interligada à evolução da Internet global. Hoje em dia centenas de milhões de usuários dependem da internet em seu ambiente de trabalho diário. No inicio da década de 1990, os pesquisadores argumentaram que o IPv4 seria insuficiente para as novas aplicações, visto que a expansão global da rede seria dada de maneira desenfreada. Isso porque o crescimento da Internet, que dobrava de tamanho em media a cada nove meses, logo esgotaria o conjunto de endereços disponíveis (COMER, 2006).

Foram necessários anos para que o IETF (Internet Engineering Task Force), formulasse uma nova versão do protocolo IP. Vários esforços paralelos foram iniciados para explorar formas de resolver essas limitações de endereços, enquanto ao mesmo tempo fornecer funcionalidades adicionais. Depois de inúmeras revisões do novo protocolo IP, o IETF decidiu atribuir o nome IPv6 no ano de 1995 (RFC1752).

O protocolo IPv6 na arquitetura TCP/IP compartilha características que contribuíram para o sucesso do Ipv4. Dentre elas, o encaminhamento sem conexão de pacotes, chamados de datagramas. Mais especificamente, cada datagrama é encaminhado independentemente, tendo como base o endereço do destinatário inscrito em seu cabeçalho (COMER, 2006).

Apesar das semelhanças conceituais, o IPv6 muda a maior parte dos detalhes. A alteração mais chamativa diz respeito ao uso de endereços maiores, de 32 bits para 128 bits, aumentando significantemente a quantidade de endereços IP possíveis. Além disso, revisa completamente o formato do datagrama, substituindo o campo de opções de tamanho variável por uma série de cabeçalhos fixos (COMER 2006).

2.2 Formato do Datagrama IPv6

O IPv6 muda é o formato do datagrama em relação ao IPv4. Ele apresenta um cabeçalho base simplificado que utiliza somente campos essenciais, sendo que outros campos opcionais também podem ser adicionados ao datagrama através de cabeçalhos de extensão (COMER, 2006). A figura 1 mostra o conteúdo e o formato do datagrama IPv6 básico:

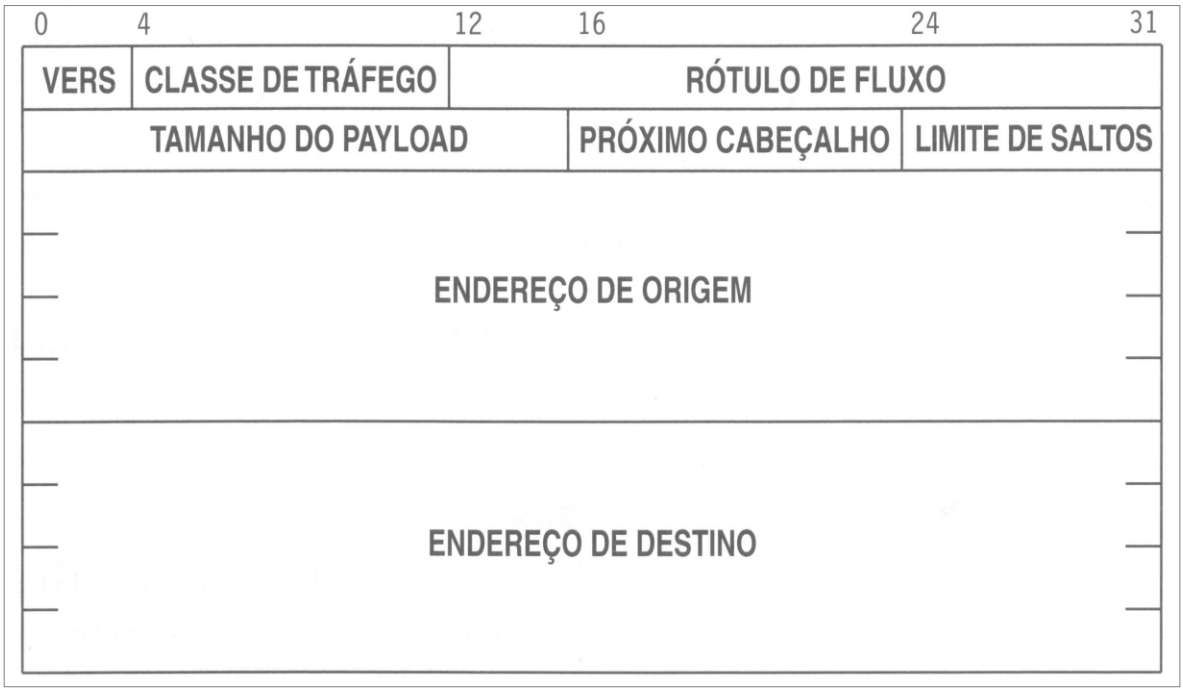

Figura 1: Formato do cabeçalho IPv6. Fonte: COMER, 2006, p.372.

Vários campos do cabeçalho básico do IPv6 correspondem diretamente a campos do cabeçalho IPv4, assim como outros campos foram retirados. Abaixo descrição dos campos segundo Forouzan (2006):

- *Vers:* Este campo define a versão do IP.
- *Classe de Tráfego:* Este campo define o nível de prioridade do pacote para uso em políticas de QoS opcionalmente implementadas em redes. Em particular, seu valor pode ser usado em uma estrutura DiffServ.
- *Rótulo de fluxo:* Identifica e diferencia pacotes do mesmo fluxo na camada de rede, sem a necessidade de verificar sua aplicação.
- *Tamanho do payload:* Este campo define o tamanho total do datagrama IP, excluindo o cabeçalho base.

- *Próximo cabeçalho:* Identifica o cabeçalho que vem após o cabeçalho básico do IPv6, estes cabeçalhos podem ser os cabeçalhos de extensão estudados na próxima seção.
- *Salto limite:*Indica o numero máximo de saltos que o datagrama IPv6 deve dar antes de ser descartado.
- *Endereço de Origem:* Identifica o *host* de origem do datagrama.
- *Endereço de Destino:* Este campo usualmente identifica o destino final do datagrama. Entretanto, se o esquema de roteamento da origem for utilizado, este campo irá conter o endereço do próximo salto (roteador).

2.2.1 Cabeçalho de extensão

Os cabeçalhos de extensão do IPv6 funcionam de forma semelhante ao campo opção do datagrama do IPv4. Um transmissor pode incluir ou retirar cabeçalhos de extensão de um determinado datagrama. O IPv6 apresenta um esquema de módulos, onde informação adicional é transmitida através dos cabeçalhos de extensão. Este esquema fornece ao IPv6 facilidade para transportar informação relevante para encaminhamento e aplicações, bem como fornecer mecanismo de segurança, fragmentação, qualidade de serviço e gestão da rede. Os cabeçalhos de extensão são incluídos entre o cabeçalho do IPv6 e o cabeçalho da camada superior, estando ligados pelo campo *Próximo cabeçalho*, formando uma cadeia (CRUZ, 1999). A figura 2 ilustra o uso dos cabeçalhos de extensão:

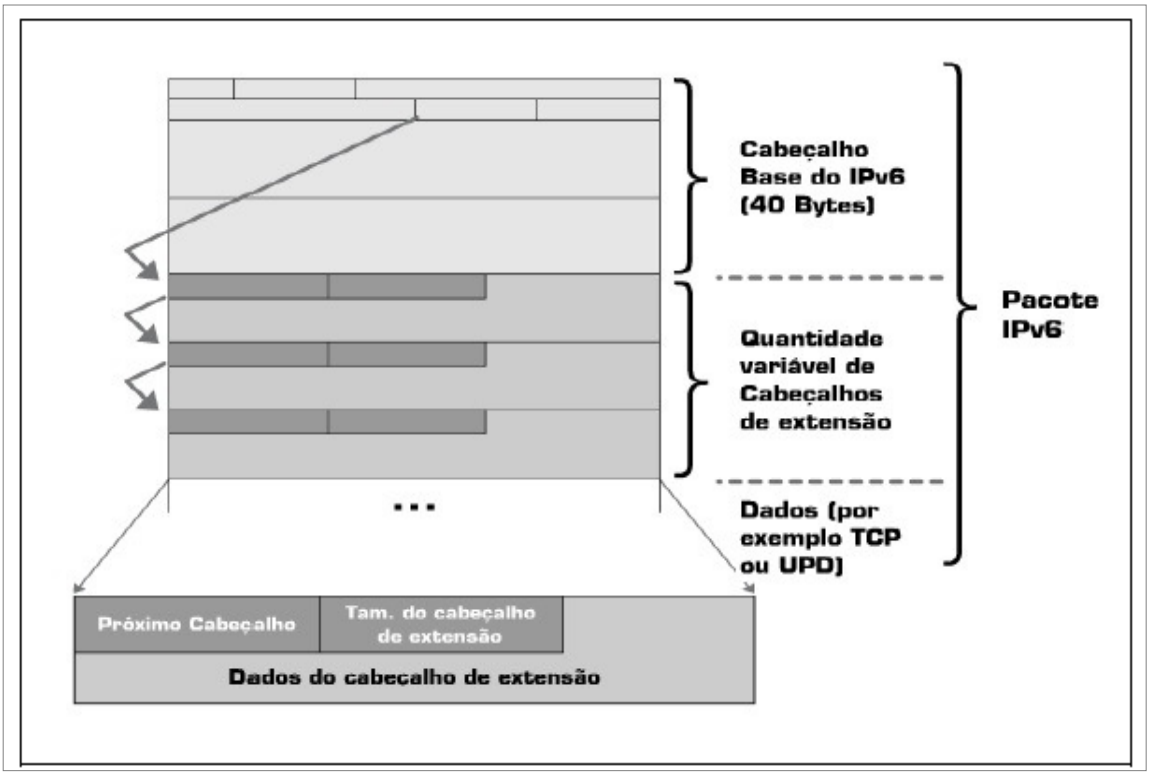

Figura 2: Uso dos cabeçalhos de extensão Fonte: HTTP://ipv6.br.

Atualmente estão definidos os seguintes cabeçalhos de extensão, segundo (RFC2460):

- *Opções de cabeçalho nó-a-nó* (*Hop-by-Hop Options Header*): Usado para transportar informação opcional que tem de ser examinada por cada nó ao longo do caminho do pacote.
- *Opções de Destino IPv6* (*Destination Options Header*): Usado para transportar informação opcional a ser analisada apenas no destino do pacote.
- *Cabeçalho de Roteamento* (*Routing Header*): Usado por uma fonte IPv6 para listar um ou mais nós intermediários que devem ser visitados até o pacote chegar ao destino.
- *Fragmentação do cabeçalho* (*Fragment Header*): Usado para enviar módulos de dados maiores do que a *Maximum Transmit Unit* (MTU) de um caminho.
- *Autenticação* do cabeçalho (*Authentication Header*): Usado para fornecer confidencialidade, autenticação e integridade do conteúdo do datagrama IPv6.

- *Encapsulamento de dados de segurança* (*Encapsulating Security Payload Header*): fornecer confidencialidade, autenticação e integridade do conteúdo do datagrama transmitido.
- *Criptografia do cabeçalho* (*IPv6 Encryption Header*): Usado para providenciar confidencialidade e integridade através da criptografia de dados.
- *Opções de Destino do cabeçalho* (*End-to-End Option Header*): Usado para o transporte de informação opcional que apenas necessita de ser examinada pelo nó destino de um pacote. Este cabeçalho pode surgir duas vezes no mesmo datagrama.

2.3 Endereçamento IPv6

Segundo Hagen (2002) "O endereçamento foi uma das razões de condução para o desenvolvimento do IPv6, juntamente com a otimização de tabelas de roteamento, especialmente na internet".

No IPv6 cada endereço ocupa 16 octetos, quatro vezes o tamanho de um endereço IPv4. A grande quantidade de endereços IPv6 garante que ele pode tolerar qualquer esquema de atribuição de endereços razoável. Para ter idéia da imensa quantidade de endereços disponíveis com o IPv6 pode-se dizer que cada equipamento eletrônico poderia possuir um endereço IPv6 (COMER, 2006).

O tamanho dos endereços impõe problemas como sua dificuldade de leitura, escrita e manipulação, sendo impraticável a utilização da notação binária. Tão pouco, a notação decimal usada para o IPv4 se mostra apropriada. Para tornar os endereços mais simples, o IPv6 especifica a notação hexadecimal com dois pontos. Com esta notação, os 128 *bits* são divididos em oito seções com dois *bytes* de tamanho cada, o que requer quatro dígitos hexadecimais por seção. O endereço assim consiste de 32 dígitos hexadecimais, com cada seção de quatro dígitos separada por dois pontos, exemplificando na figura 3:

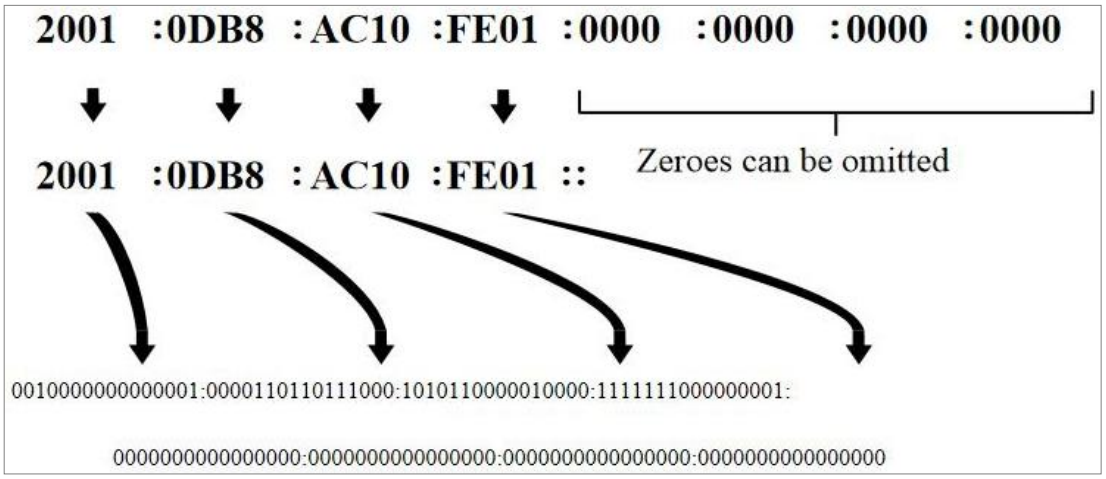

Figura 3: Endereço IPv6. Fonte: DLTEC

Embora os endereços IP na forma hexadecimal ainda sejam extensos, muitos dígitos tem valor zero, tendo-se isso em conta permite-se a compactação de zeros, em que uma sequência de zeros pode ser substituída por um par de sinais de dois pontos. Conforme Comer (2006, p. 377):

“ Para garantir que a compactação de zeros produza uma interpretação não ambígua, a proposta especifica que ela pode ser aplicada apenas uma vez em qualquer endereço. A compactação de zero é especialmente útil quando usada como o esquema de atribuição de endereços proposto, pois muitos endereços terão sequências contiguas de zeros.”

O IPv6 permite também endereçamento sem classe e a notação CIDR, permitindo que um endereço seja seguido por uma barra e um inteiro que especifica um número de bits. Isto serve para especificar a máscara de rede, que tem a mesma finalidade que no IPv4 (FOROUZAN, 2006). A figura 4 especifica um endereço Ipv6 cujo prefixo de rede possui 60 bits, conforme informado pela máscara de rede:

```
12AB::CD30:0:0:0:0/60
```

Figura 4: Endereço IP em hexadecimal com compactação de zero e com a máscara de rede.

Fonte: COMER, 2006, p. 377.

2.3.1 Tipos de Endereçamento

Conforme a (RFC3513), o IPv6 define 3 tipos de endereços: *unicast*, *anycast*, *multicast* e não existem endereços *broadcast* que foram substituídos por endereços *multicast*. Abaixo segue descrição destes endereços através da tradução livre da RFC 3513:

- *Unicast:* Um identificador para interface única rede. Um pacote enviado a um endereço unicast é fornecido para a interface identificada por esse endereço.
- Anycast: Um identificador para um conjunto de interfaces (tipicamente pertencentes a diferentes nodos). Um pacote enviado a um endereço anycast é entregue a uma das interfaces identificadas por esse endereço, no caso a interface mais próxima.
- *Multicast:* Um identificador para um conjunto de interfaces (tipicamente pertencentes a diferentes nodos). Um pacote enviado a um endereço multicast é entregue a todas as interfaces identificadas por esse endereço.

No IPv6 a representação dos prefixos de rede continuam sendo escritos como no IPv4. Os prefixos são representados na forma "*endereço-IPv6/tamanho do prefixo",* onde "*tamanho do prefixo*" é um valor decimal que especifica a quantidade de bits contíguos à esquerda do endereço que compreendem o prefixo. O exemplo de prefixo de sub-rede apresentado a seguir indica que dos 128 bits do endereço, 64 bits são utilizados para identificar a sub-rede.

Ex: Prefixo 2001:FACA:1234:1::/64
Prefixo Global: 2001:FACA::/32
ID da sub-rede: 1234:1

A maior parte dos endereços IPv6 são endereços *unicast* globais (equivalentes aos endereços públicos IPv4), que podem ser usados na maioria das situações. Porém alguns intervalos de endereços foram reservados para usos específicos.A tabela abaixo mostra alguns dos intervalos de endereços atualmente em uso:

::	Endereços não especificados
::1	Endereço de loopback
2001(...)::/16	Endereços Globais
::FFFF:a.b.c.d	Endereço IPv4 mapeado em IPv6
FE80::/10	Endereços de Link-local
FEC0::/10	Endereços de site-local
FF00::/8	Endereços multicast

TABELA 1 – Intervalos de Endereços.

Os endereços ilustrados na tabela 1 podem ser descritos como:

- *Endereços não especificados* (::): Indica ausência de um endereço, não deve ser atribuído a nenhum nó.
- *Endereço de loopback* (::1): Este endereço serve para referenciar a própria maquina.
- *Endereços Globais* (2001(...)/16): Estes endereços são equivalentes as endereços IPv4 públicos.
- *Endereço IPv4 mapeado em IPv6* (::FFFF:a.b.c.d): Estes endereços representão endereços IPv4 mapeados em Endereços IPv6.
- *Endereços de link-local* (FE80::/10): Estes endereços são criados a partir da auto-configuração do IPv6.
- *Endereços de site-local* (FEC0::/10): Estes endereços são equivalente aos endereços IPv4 privados.
- *Endereços multicast* (FF00::/8): Este endereço tem a função de enviar mensagens a grupos de computadores em uma rede IPv6.

No IPv6 é possível atribuir a uma única interface múltiplos endereços de rede, independentemente do seu tipo (Ex: Endereço loopback + Endereço Link-local + Endereço Global) isto torna a rede mais dinâmica, pois pensando no exemplo citado, antes mesmo de ter um endereço global associado a sua interface para navegar na internet, o *host* já poderá se comunicar com os demais *hosts* da rede através do endereço de link local associado automaticamente a sua interface com o mecanismo de auto-configuração do IPv6.

Endereços IPv4 podem ser mapeados em IPv6, este tipo de mapeamento normalmente é usado em técnicas que traduzem os cabeçalhos dos datagramas IPv4 para o IPv6 (NAT64 e o algoritmo de tradução SIIT), nesse caso os últimos 32bits do endereço IPv6 são representados pelo endereço IPv4. Ex: "::FFFF:0:129.144.52.38".

2.4 Transição

Num mundo que convive com o IPv4 a transição do IPv6, apresenta um problema. Segundo (KUROSE, 2003) "*O problema é que os sistemas habilitados para IPv6 podem ser 'inversamente compatíveis', isto é,podem enviar, rotear e receber datagramas IPv4, enquanto os sistemas habilitados para IPv4 não podem manusear datagramas IPv6".*

Na atual fase de implantação do IPv6, órgãos governamentais, instituições de ensino, entidades ligadas a internet e empresas buscando inovação começaram a implantar massiçamente o IPv6, formando assim ilhas IPv6 flutuantes em uma internet majoritariamente IPv4 este cenário implicará na convivência entre os protocolos IPv6 e IPv4, pois os novos *hosts* IPv6 terão a necessidade de acessar servidores que ainda usam o IPv4. Para que *hosts* IPv6 possam acessar servidores IPv4 serão necessárias o uso de técnicas que possibilitam a convivência entre os dois protocolos, como pode ser visto abaixo:

- ***Pilha dupla****: Segundo (*MOREIRAS, 2012*),* consiste na convivência do IPv4 e do IPv6 nos mesmos equipamentos ou seja cada equipamento possuirá endereços IPv4 e IPv6 de forma nativa e simultaneamente. Essa é a técnica padrão escolhida para a transição para IPv6 na Internet e deve ser usada sempre que possível.
- ***Túneis:*** De acordo com (MOREIRAS, 2012), Esta técnica consiste em na criação de túneis, que permitem transmitir pacotes IPv6 através de redes IPv4 ou vice-versa, encapsulando o conteúdo dos datagramas IPv6 em datagramas IPv4 ou vice-versa, mas não a comunicação entre rede IPv4 e IPv6.
- ***Tradução:*** Segundo (MOREIRAS, 2012), permitem que equipamentos usando IPv6 comuniquem-se com outros que usam IPv4, por meio da conversão de cabeçalhos dos datagramas. O uso deste tipo de técnica implica em não utilizar

as novas funcionalidades do IPv6, ao se efetuar a comunicação com equipamentos que usem IPv4.

Deve-se notar que tanto os túneis quanto as técnicas de tradução podem ser com estado (*stateful*) ou sem estado (stateless). Técnicas com estado são aquelas em que é necessário manter tabelas de estado com informações sobre os endereços ou pacotes para processá-los. Nas técnicas sem estado não é necessário guardar informações, pois cada datagrama é tratado de forma independente. De forma geral técnicas com estado são mais caras, pois demandam mais processamento e memória, mais possibilitam guardar informações em tabelas, isso é necessário em algumas técnicas de tradução como NAT64. (MOREIRAS, 2012)

2.4.1 Pilha Dupla (Dual Stack)

A utilização da técnica de Pilha Dupla possibilita que nós de uma rede estejam aptos a tratar datagramas de ambos os protocolos. Um nó que utiliza a Pilha Dupla, também chamado de nó IPv4/IPv6, ao se comunicar com outros nós IPv6 se comportará como um nó IPv6, e ao se comunicar com nós IPv4 se comportará como um nó IPv4. Isso é possível porque os nós que usam a Pilha Dupla possuem endereços IPv4 e endereços IPv6 atribuídos a suas interfaces. A figura 5 ilustra o funcionamento da pilha Dupla:

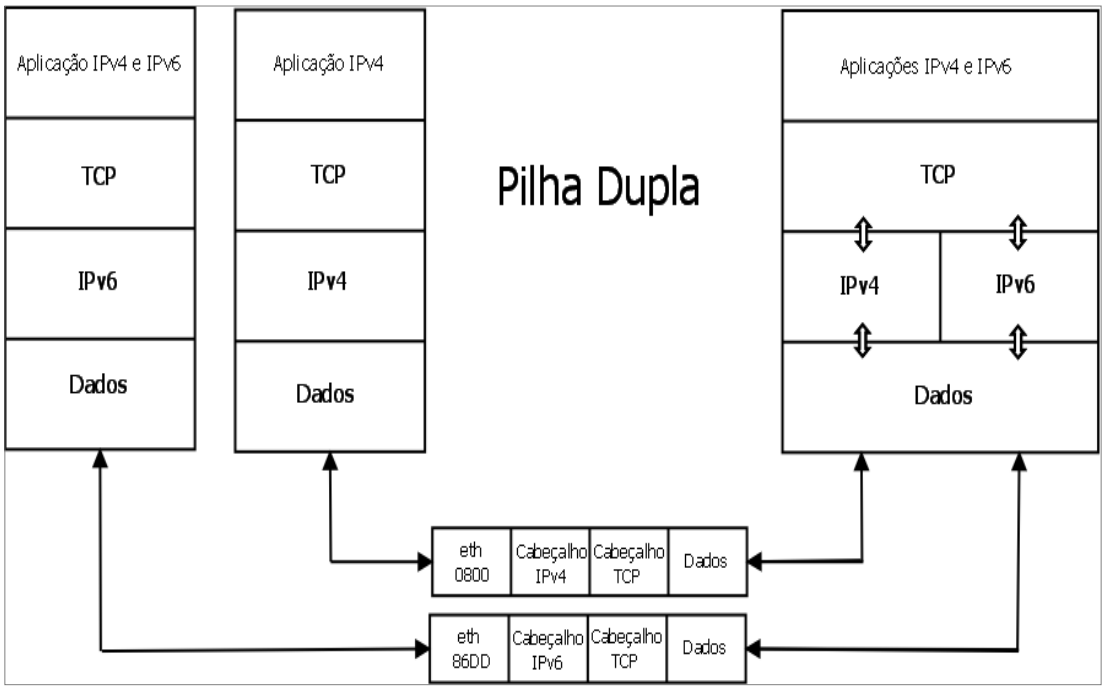

Figura 5 : Funcionamento da Pilha Dupla.

Esta técnica não tem como finalidade resolver o problema da falta de endereços IPv4 que virá a ocorrer, mas sim construir uma rede cujos *hosts* sejam capazes de se comunicarem tanto com IPv4 quando IPv6. Entre todas as técnicas de transição, a Pilha Dupla possui implementação mais simples, possibilitando uma implantação gradual nos nós da rede. No futuro, isso pode simplificar a descontinuação do uso do IPv4, pois seria necessário somente desabilitar a pilha IPv4 nos nós IPv4/IPv6.

A Pilha Dupla é suportada de forma nativa pelos sistemas operacionais atuais. Dessa forma, sua implantação consiste em atribuir os endereços de ambos os protocolos às interfaces de rede, e configurar as necessidades que cada protocolo precisa para operar da maneira desejada, afinal todas as configurações de rede desde rotas até *firewalls* serão distintas para cada protocolo.

2.4.2 Túneis

Neste TCC as técnicas de tunelamento, não foram abordadas, pois não contemplam a possibilidade de se criar uma cenário de convivência entre redes IPv4 e IPv6. A técnica de tunelamento torna possível a comunicação entre ilhas IPv6 através de redes IPv4, encapsulando datagramas IPv6 em datagramas IPv4. Entretanto com o tunelamento não é possível que ilhas IPv6 comuniquem-se com redes IPv4.

2.4.3 NAT64/DNS64

NAT64 é um mecanismo para transição e coexistência Ipv4-IPv6 baseado em tradução dos cabeçalhos dos datagramas (RFC 6146). Juntamente com DNS64, possibilita que um cliente IPv6 possa iniciar comunicações com um servidor IPv4. O contrário também é possível, , porém para isso configurações estáticas de aplicações que permitam o NAT-transversal serão necessárias, como por exemplo *proxie* HTTP que possibilitem o acesso de um *host* Ipv4 a um servidor web IPv6.

O NAT64 traduz datagramas IPv6 para IPv4 e vice-versa. A tradução é efetuada sobre os cabeçalhos dos datagramas, seguindo o algoritmo de tradução IP / ICMP (*SIIT*), usando um prefixo IPv6 atribuído ao NAT64 para esse fim específico. A especificação atual apenas define como o NAT64 traduz datagramas *unicast* que contenham PDUs TCP, UDP e ICMP. Datagramas enviados para *multicast* estão fora desta especificação, assim como outros protocolos e extensões do IPv6, tais como IPsec.

Resumindo o NAT64 é uma técnica de tradução de datagramas IPv4 para IPv6, Para que esta tradução seja possível é necessário que ele traduza os endereços IPv4 para endereços IPv6, a figura 6 ilustra esta tradução:

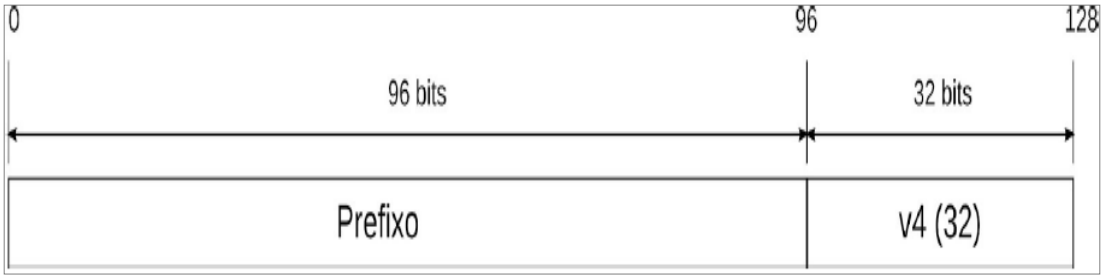

Figura 6: Tradução de um endereço IPv4 em IPv6.
Fonte: MOREIRAS,2012,p.36.

O NAT64 realiza a tradução usando o algoritmo SIIT (*Stateless IP/ICMP Translation Algorithm*). O SIIT é um mecanismo de tradução sem estado (*stateless*) de cabeçalhos IP/ICMP, que permite a comunicação entre *hosts* puramente IPV6 com *hosts* puramente IPv4. Ele utiliza um tradutor localizado na camada de rede, que converte campos específicos do cabeçalho do datagrama IPV6 em cabeçalho do datagrama IPv4 e vice-versa. Para realizar este processo, o tradutor necessita de um endereço IPv4 mapeado em IPv6, o mapeamento deve estar no formato 0::FFFF:a.b.c.d. (MOREIRAS,

2010) Quando o datagrama chega ao SIIT, o cabeçalho é traduzido, convertendo assim o endereço IPv6 para um endereço para IPV4 que é encaminhado ao nó de destino. As figuras 7 e 8 mostram os campos dos cabeçalhos antes e depois da tradução do datagrama:

Campo IPv4	Tradução para IPv6
Versão (0x4)	Versão (0x6)
IHL	(descartado)
Tipo de Serviço	Classe de Tráfico
Tamanho Total	Tamanho do Payload = Tamanho Total - IHL * 4
Identificação	(descartado)
Flags	(descartado)
Offset	(descartado)
Tempo de vida	Limite de Nós
Protocolo	Próximo Cabeçalho
CRC do Cabeçalho	(descartado)
Endereço de Origem	Endereço de Origem
Endereço de Destino	Endereço de Destino
Opções	(descartado)

Figura 7: Endereço IPv4 traduzido para IPv6. Fonte: MOREIRAS, 2012.

Campo IPv6	Cabeçalho IPv4 Traduzido
Versão (0x6)	Versão (0x4)
Classe de Tráfico	Tipo de Serviço
Etiqueta de Fluxo	(descartado)
Tamanho do Payload	Tamanho Total = Tamanho do Payload + 20
Próximo Cabeçalho	Protocolo
Limite de Nós	Tempo de Vida
Endereço de Origem	Endereço de Origem
Endereço de Destino	Endereço de Destino
----	IHL = 5
----	CRC do Cabeçalho Recalculado

Figura 8: Endereço IPv6 traduzido para IPv4. Fonte: MOREIRAS, 2012.

Para um funcionamento completo o NAT64 precisa trabalhar em conjunto com o DNS64 Esse serviço converte as solicitações DNS IPv4 em solicitações DNS IPv6 e vice-versa, respondendo assim os *hosts* IPV6 com respostas do tipo AAAA (*quad A*), e *hosts* IPv4 com respostas do tipo A. O endereço IPv6 contido no registro AAAA é gerado com o endereço IPv4 mapeado com o prefixo definido pelo NAT64. Com o DNS64 os *hosts* puramente IPv6 poderão acessar transparentemente a Internet IPv4.

Quando um *host* IPv6 fizer uma consulta DNS a registros AAAA usando um endereço IPv4 mapeado em IPV6, o DNS64 consultará o servidor DNS autoritário da rede com registros A e depois responderá a solicitação AAAA com a conversão da resposta A do servidor autoritário. Caso o *host* use seu endereço IPv6 puro para fazer a solicitação AAAA será o servidor DNS autoritário quem fará as consultas sem que haja necessidade de atuação do DNS64, para isso os servidores DNS consultados pelo servidor DNS autoritário terão de ser capazes de resolver registros do tipo AAAA. Já no sentido reverso de acesso IPv4 a servidores IPv6 da rede, a resolução DNS ficaria por conta do *ALG* usado para a determinada aplicação. A figura 9 demonstra a seqüência de funcionamento do NAT64 em conjunto com DNS64:

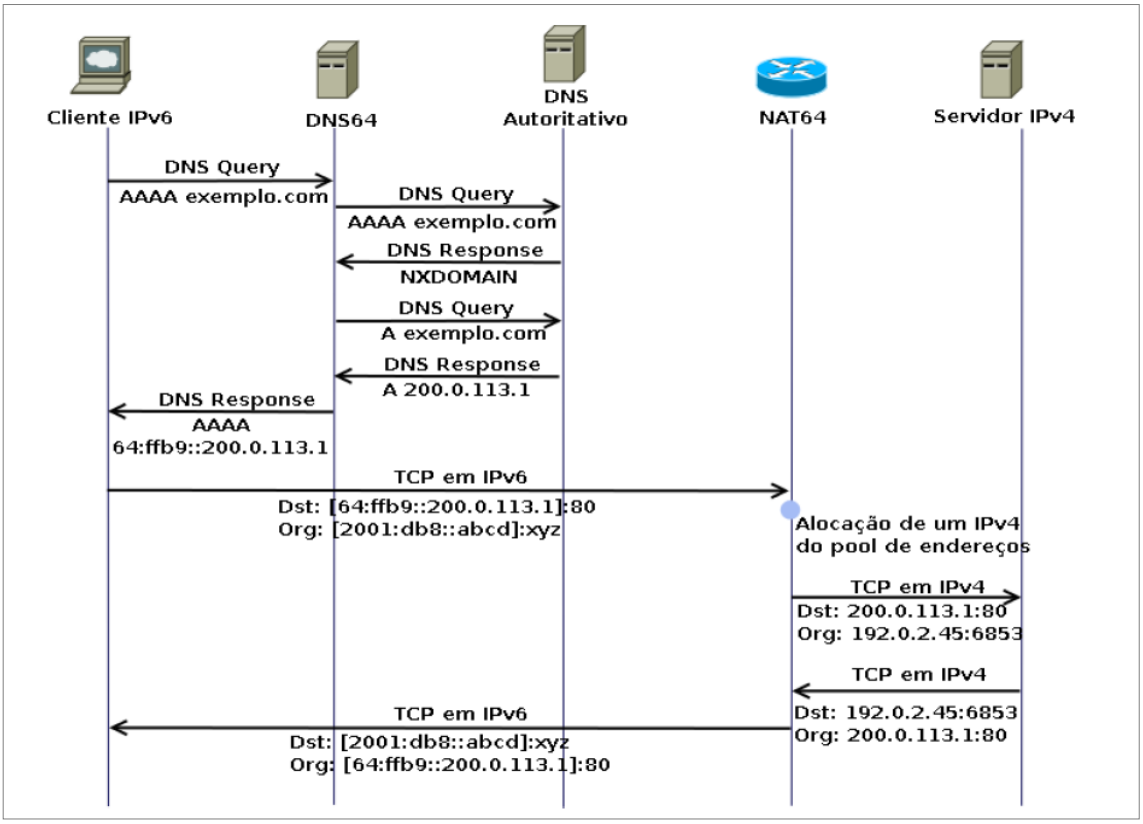

Figura 9: Seqüência de funcionamento do NAT64/DNS64. Fonte: MOREIRAS, 2012, p.37.

Segundo tradução livre da (RFC6147) “DNS64 é um mecanismo para a síntese de Registros AAAA para registros A. Um registro sintético AAAA criado pelo DNS64 a partir de um registo A original, contém o nome do proprietário do registro A original, mas ele contém um endereço IPv6 em vez de um endereço IPv4. O endereço IPv6 é uma representação do endereço IPv4 contido no registro A original. A representação IPv6 do endereço IPv4 é algoritmicamente gerada a partir do endereço IPv4 retornado no registro A e um conjunto de parâmetros configurados no DNS64 (tipicamente, um prefixo IPv6 usado por representações IPv6 de endereços IPv4 e, opcionalmente, outros parâmetros).”

O NAT64 é unidirecional, portanto para que *hosts* IPv4 possam acessar servidores na rede IPv6 são necessários ALGs (*Aplication level Gateways*). Essas aplicações são responsáveis por fazerem uma “ponte” entre os dois protocolos, a fim de permitir conectividade a aplicações especificas. Um exemplo de ALG é o próprio DNS64, que pode ser considerado um DNS-ALG, já que o mesmo utiliza registros AAAA puramente IPv6 para uma conversão em registros A, com a finalidade de hosts puramente IPv6 poderem receber respostas a consultas DNS feitas para servidores puramente IPv4. Outro exemplo de ALGs que facilmente podem ser observados são os *proxie* HTTP que possibilitam o acesso a servidores IPv6 através de *hosts* puramente IPv4.

O NAT64 foi desenvolvido a fim de ser uma opção para uma rede IPv6 se comunicar com as redes IPv4, que ainda predominam na Internet. O NAT64 não é a solução para os problemas do período de transição, já que sua implantação mantém o problema com NAT já enfrentado pelo IPv4, para aplicações como SIP e P2P um cenário fim a fim seria o ideal, já que nele todos os computadores da rede são visíveis na Internet, favorecendo assim o desenvolvimento e implantação dessas e outras aplicações. Outros problemas também são encontrados no NAT64. Além da necessidade de usar ALGs no sentido reverso, o NAT64 apresenta problemas com aplicações que enviam endereços IP no protocolo da camada de aplicação (FTP ativo, SIP, serviços do tipo *torrent* e alguns *streams* de vídeos).

A (RFC 6586) apresenta exemplos de problemas enfrentados pelo NAT64 com aplicações que enviam endereços IP no protocolo da camada de aplicação, também chamados de IPv4 *literals.* Enquanto se navega pela web, encontrar endereços IPv4 embutidos no código HTML pode quebrar algumas partes das páginas da web ao usar

o IPv6 para acesso. Isso acontece porque o DNS64 não pode sintetizar registros AAAA para estes endereços embutidos no código, porque estes endereços não são consultados a partir do DNS. Felizmente, o IPv4 *literals* parecem ser raramente encontrados durante a navegação web, porém a própria (RFC 6586) descreve dois casos que apresentarão problemas na navegaçaõ, um caso já foi encontrado no YouTube, onde algumas das aplicações de terceiros para download de conteúdo não funcionaram, e outro em uma página web de um hotel, a qual tinha uma ligação literal de seu sistema de reservas. Outras aplicações especificas (FTP ativo, SIP, serviços do tipo *torrent* e alguns *streams* de vídeos) também carregam o IPv4 na camada de aplicação, o que torna a aplicação do NAT64 difícil de ser usada com estas aplicações.

2.4.4 IVI

O IVI é um mecanismo de mapeamento de endereços sem estado, usado para prover conectividade IPv4 a redes onde existam apenas nós IPv6.

O IVI é um mecanismo de mapeamento de endereços sem estado para cenários como "uma rede IPv6 se conectando a internet IPv4", e "a internet IPv4 se conectando a uma rede IPv6". No projeto IVI, blocos de endereços IPv4 são incorporados a blocos de endereços IPv6, desta forma os *hosts* IPv6 podem, portanto comunicar-se com a internet IPv6 diretamente e com a internet IPv4 através do mapeamento sem estado. No IVI as comunicações podem ser iniciadas tanto por *hosts* IPv4 quanto por *hosts* IPv6, o mecanismo IVI também suporta a transparência de endereços fim-a-fim e implantação incremetal. O IVI inicialmente foi implantado pela CERNET (*The China Education and Research Network*) como uma referencia a documentos padrão IETF sobre tradução IPv4/IPv6 sem estado (RFC6219). A figura 10 mostra o mapeamento de endereços IPv4 em IPv6 e vice-versa:

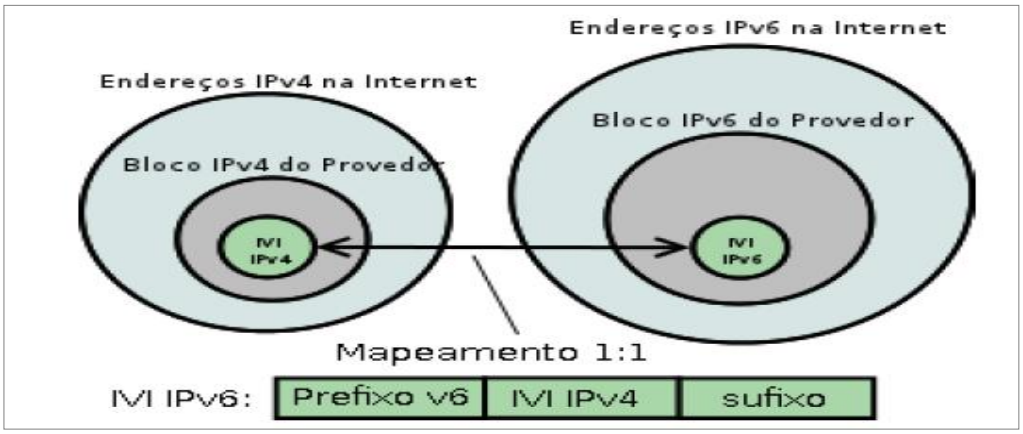

Figura 10: Mapeamento de endereços IVI. Fonte: MOREIRAS, 2012, p.30.

O nome IVI vem dos numerais romanos IV (4) e VI (6), esta técnica de tradução stateless foi desenvolvida por pesquisadores da CERNET2, a rede acadêmica chinesa. A CERNET2 implantou uma rede puramente IPv6 visando o desenvolvimento de novas tecnologias. Com base nisso e nos incentivos fiscais para o uso da nova rede o trafego IPv6 nas universidades chinesas já é maior que o trafego IPv4.(MOREIRAS,2012)

O IVI foi criado a fim de prover conectividade IPv4 a servidores puramente IPv6, para que isso seja possível o IVI cria uma interface virtual, a qual ele atribui um endereço IPv4, com isso o servidor IPv6 se torna visível na internet IPv4, da mesma forma o *host* IPv4 é espelhado na rede IPv6. Para que os *hosts* espelhados possam se comunicar com as redes onde o acesso é desejado, estes *hosts* necessitarão traduzir seus endereços para que possam se comunicar com os *hosts* reais da rede em questão. O IVI trata essa tradução associando um endereço IPv4 relativo ao endereço IPv4 traduzível que está associado ao *host* IPv6 e vice-versa.(MOREIRAS, 2012). A figura 11 ilustra este conceito:

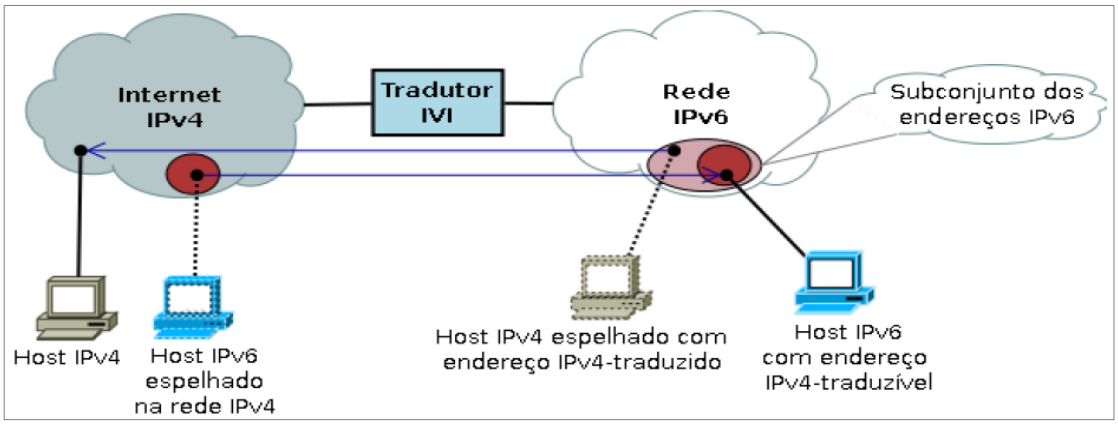

Figura 11: Exemplo conceitual do IVI. Fonte: MOREIRAS, 2012, p.30.

2.4.4.1 Mapeamento IVI.

O mapeamento IVI pode ser 1:1 (1endereço IPv4 para 1endereço IPv6) ou 1:N (1 endereço IPv4 para vários endereços IPv6). Usa-se o algoritmo de tradução SIIT de forma modificada para fazer a tradução de datagramas, além de que (NAT64 pode ser usado em alguns casos para esta finalidade). Para implementações 1:N uma quantidade definida de portas no tradutor IVI com conectividade IPv4 é usada por *hosts* puramente IPv6 de forma que cada endereço IPv6 terá um número de portas no host IVI para acessarem a internet IPv4.

Segundo tradução livre da (RFC6219): " Para representar os endereços IPv4 em IPv6, um único prefixo IVI é definido entre endereços IPv4 e o bloco de endereços IPv6 , de modo a que cada provedor de blocos terá uma pequena porção de endereços que serão mapeados em endereços IPv4 globais de forma espelhada, o sufixo neste mapeamento é representado por zeros." A figura 10 ilustra o endereço IPv4 mapeado para um endereço IPv6:

PREFIXO (IPv6)	**IVI**	**ENDEREÇO IPv4 GLOBAL**	**SUFIXO (ZEROS)**
|0 32|		**|40 72|**	**127|**

Figura 12: Mapeamento de um endereço IPv4 em um endereço IPv6.

No espaço entre os bits 0 e 31 é representado o prefixo IPv6 / 32 (por exemplo, utilizando um endereço IPv6 = 2001: db8 :: / 32),entre os bits 32 e 39 é representado o identificador de endereços IVI, entre os bits 40 e 71 bits é incorporado o endereço global IPv4 espaço representado em formato hexadecimal e dos bits 72 a 127 o sufixo do endereço é preenchido por zeros, desta forma quando o *host* espelhado for criado o IVI associará a ele um endereço IPv4 relativo ao endereço IPv4 mapeado. Note que com base no mecanismo de mapeamento IVI, um endereço IPv4 / 24 é mapeado para um IPv6 / 64, e um IPv4 / 32 é mapeado para um IPv6 / 72.(RFC6219)

As regras usadas na tradução dos datagramas são executadas de acordo com SIIT exceto pelos campos "endereço de origem" e "endereços de destino". As figuras 13 e 14 ilustram a tradução dos cabeçalhos:

Campo IPv4	Tradução para IPv6
Versão (0x4)	Versão (0x6)
IHL	(descartado)
Tipo de Serviço	Classe de Tráfico
Tamanho Total	Tamanho do Payload = Tamanho Total - IHL * 4
Identificação	(descartado)
Flags	(descartado)
Offset	(descartado)
Tempo de vida	Limite de Nós
Protocolo	Próximo Cabeçalho
CRC do Cabeçalho	(descartado)
Endereço de Origem	Aplicar mapeamento stateless IVI
Endereço de Destino	Aplicar mapeamento stateless IVI
Opções	(descartado)

Figura 13: Endereço IPv4 mapeado em um endereço IPv6.Fonte: MOREIRAS, 2012, p.32.

.

Campo IPv6	Cabeçalho IPv4 Traduzido
Versão (0x6)	Versão (0x4)
Classe de Tráfico	Tipo de Serviço
Etiqueta de Fluxo	(descartado)
Tamanho do Payload	Tamanho Total = Tamanho do Payload + 20
Próximo Cabeçalho	Protocolo
Limite de Nós	Tempo de Vida
Endereço de Origem	Aplicar mapeamento stateless IVI
Endereço de Destino	Aplicar mapeamento stateless IVI
----	IHL = 5
----	CRC do Cabeçalho Recalculado

Figura 14: Endereço IPv6 mapeado em um endereço IPv4. Fonte: MOREIRAS, 2012, p.32.

3 Avaliação das Técnicas de Transição

Esta seção descreve o desenvolvimento do trabalho, em que se implantaram as diferentes técnicas em um cenário de testes. A figura 1 ilustra os cenários de testes usados neste tcc:

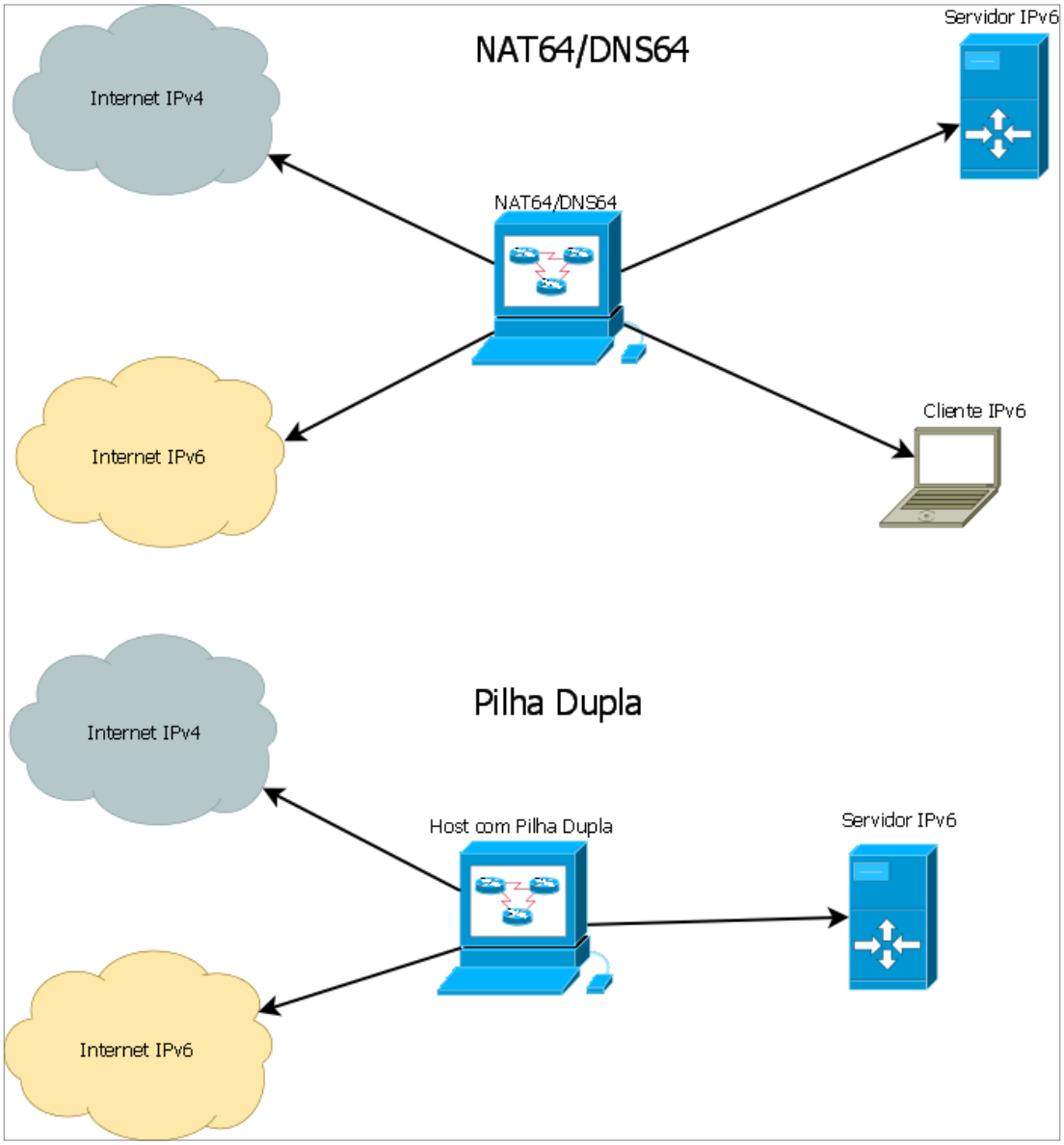

Figura 1: Topologia das redes de testes.

O desenvolvimento do trabalho adotou um cenário de teste descrito na figura 1. Nesse cenário maquinas virtuais simulam *hosts* (cliente e servidor) puramente IPv6 estão interligados a Internet IPv4 e a Internet IPv6 através de um *host* que possui a Pilha Dupla configurada e é usado como *gateway* na implantação do NAT64/DNS64. Por fim, nos procedimentos descritos nesta seção, assumiu-se que os computadores usam sistema operacional Linux.

3.1 Implantação da Pilha Dupla (Dual Stack)

De forma geral, com esta técnica a configuração da rede deve ser duplicada, pois ela deve ser feita tanto para IPv4 quanto IPv6, sendo um independente do outro. Além disso, as aplicações de rede precisam ter sido escritas para fazer uso de ambos os protocolos. Assim, a implantação da Pilha Dupla em uma rede implica em:

- Configuração de rotas e regras de roteamento independentes para cada protocolo.
- Configuração de firewalls independentes para cada protocolo.
- Configuração de servidores de DNS para que possam resolver nomes e endereços de ambos os protocolos.
- Maior complexidade no gerenciamento de rede, já que a rede deverá ser duplicada.
- Maior custo, porque se os nós da rede não tiverem suporte ao IPv6 deverão ser substituídos por outros com suporte ao protocolo IPv6. (MOREIRAS, 2010)
- Implantação de serviços de rede com softwares capazes de responder a requisições feitas com ambos protocolos.

Numa rede onde existem nós IPv4/IPv6 será necessário atribuir endereços de ambos protocolos a um nó isto pode ser feito de forma manual ou automática (DHCP, DHCPv6, mecanismo de auto-descoberta do IPv6). As configurações de roteamento também serão feitas separadamente para cada protocolo.

A forma de como a filtragem de pacotes na rede terá de ser feita separadamente, nos firewalls as configurações também deverão ser feitas de formas distintas para cada protocolo, pois os softwares que iram tratar os fluxos IPv4 e IPv6 iram executar suas tarefas visando apenas um fluxo especifico.

Segundo (MOREIRAS, 2012) “Em relação ao DNS, é preciso que este esteja habilitado para resolver nomes e endereços de ambos os protocolos. No caso do IPv6, é preciso responder a consultas de registros do tipo AAAA (quad-A), que armazenam endereços no formato do IPv6, e para o domínio criado para a resolução de reverso, o ip6.arpa.”

O apêndice A demonstra a configuração de Pilha dupla em um ambiente Linux. Estes passos já são suficientes para que o nó possa se comunicar com um *gateway* para acessar a internet ou para se comunicar com outro nó da rede. A figura 1 ilustra uma interface ethernet configurada com a Pilha Dupla:

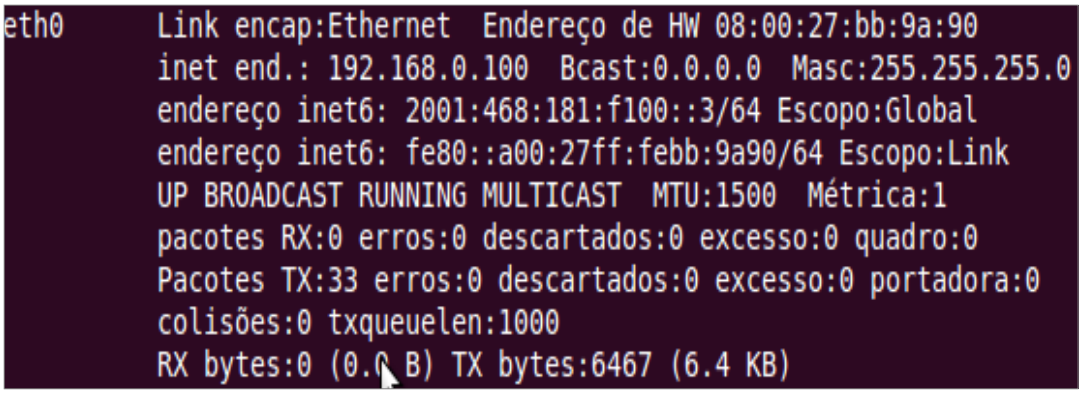

Figura 2: Interface Ethernet configurada com Pilha Dupla em ambiente Linux.

Como nota vale a pena lembrar que as configurações serão independentes para cada protocolo. Para execução de firewalls, servidores web, servidores de email será necessário que os aplicativos usados na implementação tenham suporte ao IPv6, e sejam configurados para atender ambos os protocolos, para servidores DNS, este terá que prover suporte ao IPv6 e ser capaz de resolver nomes e endereços de ambos os protocolos, um exemplo de software que possui suporte IPv6 para implantação de servidores DNS numa rede com Pilha Dupla é o BIND (*Berkeley Internet Name Domain*).

3.2 Implantação do NAT64/DNS64

Para a implantação do NAT64/DNS64 em uma rede, é necessário que pelo menos um *host* da rede possua duas interfaces, sendo uma configurada com um

endereço IPv4 e outra configurada com um endereço IPv6 (este *host* não necessariamente precisa ser o *gateway* da rede). Nele será necessário compilar o módulo do *kernel* responsável por executar o NAT64. Nesta rede também será necessário que o *host* tradutor ou o próprio servidor DNS autoritário da rede esteja configurado para executar o DNS64 e assim atender as solicitações DNS no sentido IPv6 para IPv4.

Neste TCC, para implantação do NAT64, foi usado um módulo para o *kernel* do Linux ubuntu desenvolvido pelo projeto *ecdsys.* Este módulo foi desenvolvido para implementação em *kernels* entre as versões 2.6.31 e 2.6.35, porém o próprio projeto ecdsys lançou um *patch* para que o modulo NAT64 pudesse ser usado em *kernels* superiores ao 2.6.38. O sistema operacional escolhido para a implantação foi o Ubuntu 10.04 com versão de *kernel* 2.6.32, e para implantação do DNS64 foi escolhido o software BIND9 que possui suporte ao IPv6 e ao DNS64.

A implantação do NAT64/DNS64 pode ser feita seguindo os passos descritos no apêndice B. Com isso NAT64 estará apto a prover navegabilidade IPv4 aos clientes IPv6 com transparência. A figura 3 mostra a navegação web transparente através do acesso a url www.terra.com ou ao endereço IPv4 mapeado (64::ff9b::208.70.188.151) usando o NAT64:

Figura 3: Navegação com NAT64.

A implementação do NAT64 para Linux cria uma interface virtual responsável por tratar os datagramas traduzidos do IPv4 para o IPv6. Para que os datagramas IPv4 possam ser mapeados para tradução NAT64 cada *host* na rede terá uma rota apontando os datagramas com prefixo 64:ff9b::/96 para o *host* que possui a interface NAT64. A figura 4 ilustra uma interface virtual nat64 que é criada virtualmente no *gateway* NAT64 da rede:

```
nat64     Link encap:Não Especificado  Endereço de HW 00-00-00-00-00-00-00-00-00-00-00-00-00-00-0
0-00
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Métrica:1
          pacotes RX:4 erros:0 descartados:0 excesso:0 quadro:0
          Pacotes TX:4 erros:0 descartados:0 excesso:0 portadora:0
          colisões:0 txqueuelen:0
          RX bytes:416 (416.0 B) TX bytes:416 (416.0 B)
```

Figura 4: Interface NAT64 ativa.

Serviços como navegação web, e-mail, SSH, NTP, FTP, entre outros podem ser acessados normalmente. Porém problemas podem ser observados em aplicações que carregam o endereço IP na camada de aplicação (FTP ativo, SIP, serviços do tipo *torrent* e alguns *streams* de vídeos), como explicado na seção 2.4.3.

3.3 Implantação do IVI.

Por limitações nos softwares disponíveis, a técnica IVI não foi testada neste TCC. Porém, nesta seção são apresentados os resultados dos estudos sobre o funcionamento e capacidades do IVI.

O tradutor IVI pode ser implantado nos formatos de tradução 1:1 ou 1:N. A figura 5 ilustra esses possíveis cenários:

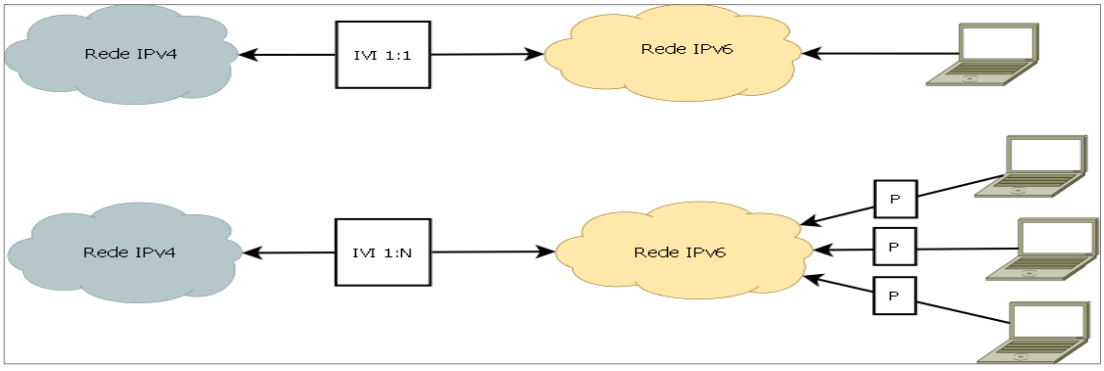

Figura 5: Possíveis cenários de implantação do IVI.

O primeiro cenário mostrado pela figura 4 ilustra o uso da tradução IVI 1:1, onde os endereços IPv6 globais ou locais são mapeados em endereços IPv4 globais que serão usados para navegar na internet IPv4 e vice-versa no caso de endereços IPv6 globais serem usados.Neste cenário é usado um tradutor IVI com duas interfaces uma conectada a rede IPv4 e outra conectada a rede IPv6, esse tradutor será responsável por criar os nós virtuais nas redes e traduzir os pacotes na comunicação. Este cenário é stateless, ou seja, necessita de menos gasto de CPU e memória, não necessita de uso de técnicas de tradução auxiliares como NAT64 e pode ser iniciado por ambos os lados da comunicação. O ponto negativo deve ser observado que por ser um mapeamento 1:1 ele não é escalavel, já que será necessário um endereço IPv4 para cada endereço IPv6 mapeado.(XING LI, 2009)

O segundo cenário mostrado pela figura 4 ilustra o uso da tradução 1:N, onde os endereços IPv6 juntamente com suas portas previamente definidas para este uso são mapeados em um único endereço IPv4 que terá um intervalos de portas definido para cada endereço IPv6 mapeado (o intervalo de portas e as portas usadas pelos *hosts* IPv6 deverão ter o mesmo identificador numeral). Neste cenário também é usado um tradutor IVI com duas interfaces, uma ligada ao lado IPv4 da rede e outra ligada ao lado IPv6 da rede, esse tradutor usará a mesma técnica de tradução IVI só que na versão A + P (*address + port*) sendo assim os endereços IPv6 serão mapeados juntamente com usas portas para o IPv4, com isso um único endereço IPv4 será necessário para atender vários endereços IPv6, esta técnica continua sendo stateless, não necessita do uso de tradutores auxiliares como NAT64 e pode ser iniciada por ambos os lados. Esta técnica também chamada de dIVI (Double IVI) ainda não possui implementação publica disponível. (XING LI, 2009). A figura 6 mostrada abaixo ilustra a tradução 1:N descrita no segundo cenário:

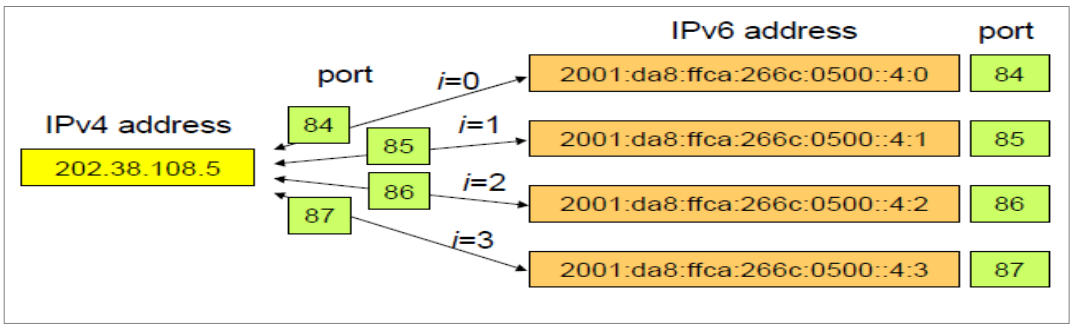

Figura 6: Tradução IVI 1:N Fonte: XING LI, 2009, p22.

O IVI necessita de um DNS-ALG assim como o DNS64, o próprio IVI possui um *patch* DNS com funcionamento similar ao DNS64 que é responsável pelo tratamento das respostas AAAA e A para *hosts* IPv6. A figura 7 ilustra o funcionamento desta técnica DNS:

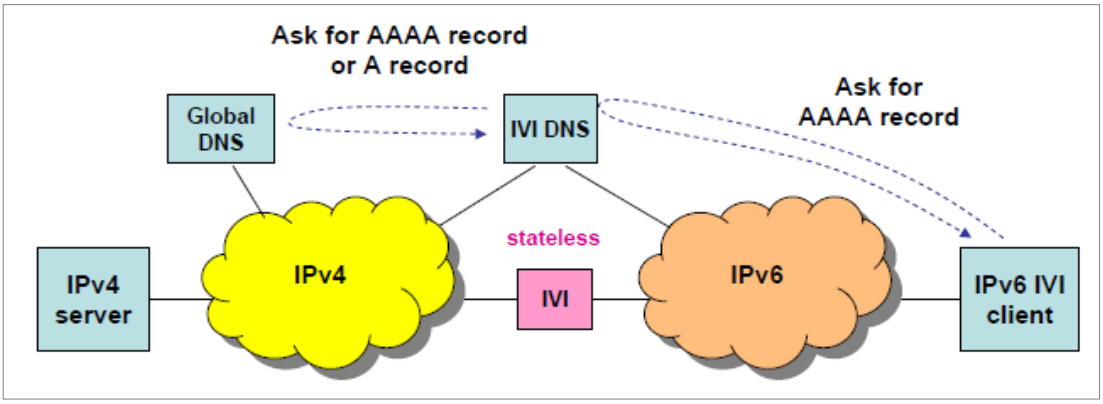

Figura 7: IVI DNS. Fonte: XING LI, 2009, p24.

O código fonte do tradutor IVI assim como o patch DNS e alguns testes podem ser encontrados no site: HTTP://www.ivi2.org/IVI.

3.4 Testes

Nesta seção são apresentados os resultados obtidos com os testes da técnica Pilha Dupla e NAT64/DNS64. A técnica IVI não pôde ser implantada e testada por limitações nos *softwares* disponíveis.

Os testes foram realizados utilizando a rede ilustrada na figura 1 do capítulo 3.

3.4.1 Teste da Pilha Dupla.

A técnica de Pilha Dupla possui um funcionamento simples. Ao tentar acessar um servidor IPv4, a aplicação que gera a requisição usa a pilha IPv4. Para acessar um servidor IPv6, essa aplicação usa a pilha IPv6. Assim, cabe à aplicação escolher apropriadamente o protocolo IP a ser usado.

Devido a falta de endereços IPv6 com rota para a internet a serem usados pelos *hosts* IPv6 da rede, os testes deste cenário em relação a Internet limitaram-se a Pilha IPv4, de certa forma se torna irônico falar sobre a falta de endereços IPv6 quando os estudos o apontam como a solução, porém esse pequeno problema ocorre porque na atual fase do IPv6 seu usa ainda não está difundido entre usuários finais. A figura 8 ilustra a rede usada para testar a técnica de Pilha Dupla:

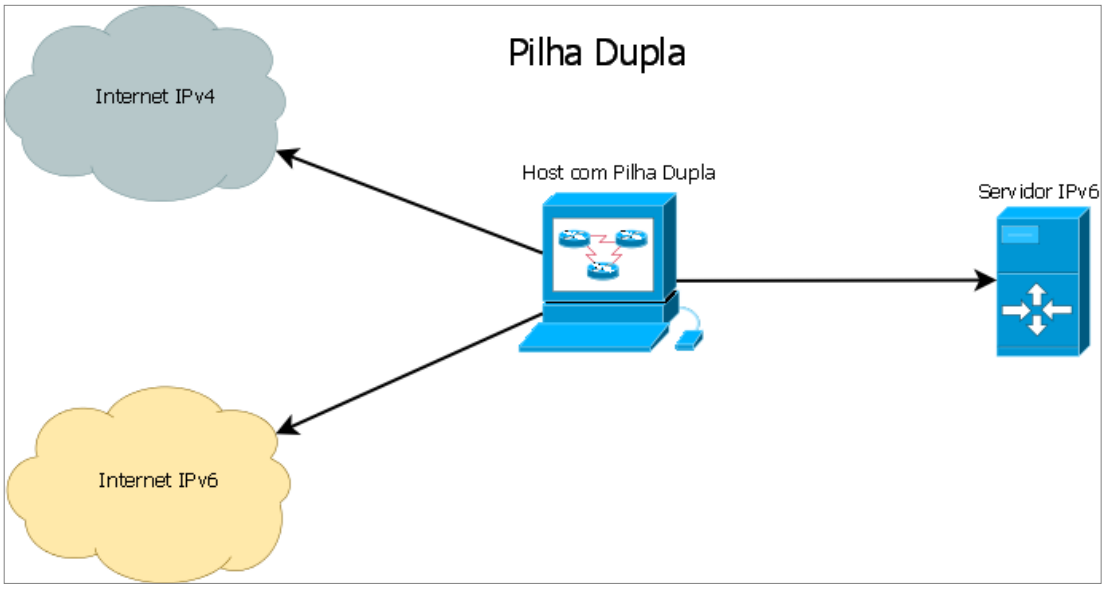

Figura 8: Topologia das redes de teste da Pilha Dupla.

Testes com a comunicação interna entre as redes IPv4 e IPv6 foram feitos, e foi observado que um *host* IPv4/IPv6 se adapta perfeitamente às duas redes, conseguindo acessar servidores tanto IPv4 quanto IPv6. Para estes testes foi usada a maquina virtual que simula um *host* IPv4/IPv6. Com este *host* foram acessados servidores IPv4 da Internet IPv4 e o servidor IPv6 da rede de testes.

A técnica de Pilha Dupla pode ser considerada como a técnica de transição mais versátil e simples, porém implica a necessidade de uma duplicação na rede (configurações de rotas, *firewalls* e servidores DNS). Porém A Pilha Dupla explora os protocolos de maneira distinta, o que implica em um melhor desempenho (Com a Pilha Dupla cada aplicação pode escolher qual protocolo usar, com isso não se aplicam limitações a esta técnica, com exceção da falta de suporte ao IPv6 encontrada em algumas aplicações) e também evita problemas com eventuais traduções e mapeamentos entre os protocolos.

3.4.2 Teste do NAT64/DNS64.

A técnica NAT64, em conjunto com DNS64, possui uma implantação um pouco mais complexa do que a Pilha Dupla em termos de configuração de aplicativos. Porém com o NAT64/DNS64 é necessário que haja duplicação da rede visto que apenas um *host* da rede precisa ter endereços IPv4 e IPv6. A rede usada para os testes do NAT64 está ilustrada na figura 9, nela foram testadas diversas aplicações:

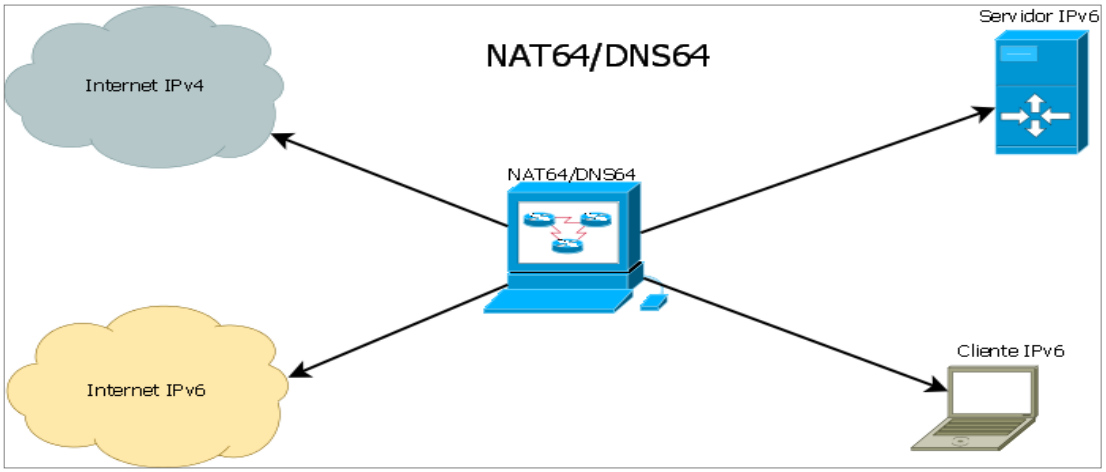

Figura 9: Topologia das redes de teste do NAT64/DNS64.

Os resultados obtidos em cada um destes testes estão listados abaixo:

- *SSH:* O uso do *SSH* foi possível quando um *host* puramente IPv6 fez o acesso remoto a outro *host* IPv6 e também quando fez acesso a um *host* IPv4 através da tradução NAT64. No entanto, *hosts* IPv4 de outras redes não conseguiram acessar o host IPv6 interno, pois o NAT64 é unidirecional. Sendo assim, para que aplicações como o *SSH* consigam fazer o acesso reverso será necessário o uso de ALGs.
- *FTP*: O cliente *FTP* não apresentou problemas para transferência de arquivos, porém no sentido de acesso reverso (do IPv4 para o IPv6) novamente é necessário uso de ALGs.
- *Gerenciador de Emails*: O gerenciador de e-mail testado foi o *Mozilla ThunderBird 12.0*, o servidor de email foi testado através de uma conta *@hotmail* usando *POP3* e *SMTP*. Durante o uso a aplicação não apresentou problemas.
- *MSN*: O serviço mensageiro foi testado através do aplicativo *AMSN*, utilizando uma conta de e-mail @hotmail.com. Seu uso não foi possível, pois aparentemente em algum momento da negociação com o servidor ele solicita um endereço IPv4 através da camada de aplicação, sem que haja uma consulta DNS que possa ser resolvida com o DNS64.
- *SKYPE*: Durante os testes não foi possível conectar-se com o aplicativo *SKYPE* 2.2. Possivelmente o motivo do não funcionamento do SKYPE foi o mesmo do AMSN.
- *NTP*: A sincronização de relógios foi possível através do NAT64.
- *TORRENT*: Para testes com o serviços *Torrent* foi usado o aplicativo BitTorrent, durante os testes pode-se perceber existiam *peers* com endereços IPv6, entretanto estes *peers* não eram acessíveis pela falta de um endereço IPv6 global para ser associado aos *host* IPv6 da rede. Através do mapeamento NAT64, não foi possível acessar *peers* IPv4, visto que em algum momento da comunicação o *bitTorrent* deve solicitar um endereço IPv4 na sua forma literal através da camada de aplicação.
- *Compartilhamento de Arquivos*: Para testar o compartilhamento de arquivos usou-se o serviço D*ropbox 1.4.0.* A sincronização entre o software cliente e o servidor não foi possível. Possivelmente a causa do problema foi a mesma das aplicações anteriores que não funcionaram.

Usando a versão *web* do aplicativo foi possível fazer upload e download de arquivos no servidor.

- *Navegação Web*: Para os testes de navegação Web foram usados dois navegadores, o Mozilla FireFox e o Google Chrome, com ambos a navegação se mostrou transparente tanto com HTTP quanto com o HTTPS, porém em alguns *sites* que utilizam o IPv4 em sua forma literal a navegação não foi possível.

Para o sentido reverso de acesso a rede, não foi possível acessar o servidor IPv6, já que não havia endereços IPv6 Globais disponíveis na Implementação. Porém foi possível acessar outros servidores puramente IPv6 através do Proxy HTTP disponível na página http://www.sixxs.net/tools/gateway. Este ALG possibilita o acesso de um *host* IPv4 a um servidor IPv6 vice-versa. Outras aplicações para acesso reverso não foram testadas por falta de softwares para esta finalidade.

4 Considerações Finais

Com o esgotamento dos endereços IPv4 será necessário migrar as redes já existentes para o IPv6, mas esta migração pode ser difícil caso toda infra-estrutura tenha de ser migrada ao mesmo tempo. Para simplificar esta migração este trabalho buscou mostrar alternativas para um cenário de transição entre os protocolos IPv4 e IPv6.

Durante o desenvolvimento deste trabalho primeiramente foram estudados o protocolo IPv6 e suas peculiaridades, num segundo momento foram estudadas técnicas de transição entre o IPv6 e o IPv4. Durante os estudos e testes pode ser observado que a Pilha Dupla é a técnica que melhor se encaixa neste cenário de transição, pois ela facilmente se adapta aos dois protocolos. Porém, para que seu uso seja possível, é necessário ao menos um endereço IPv4 válido e a configuração da rede necessita ser duplicada para poder operar os dois protocolos.

O NAT64/DNS64 se mostrou uma técnica eficaz para navegação web e outros serviços básicos. No entanto, existem problemas com aplicações que fazem uso de endereços IPv4 em sua forma literal. Por meio de testes feitos com NAT64, foi possível perceber que diversas aplicações (*AMSN, SKYPE, DROPBOX, TORRENT e algumas paginas da* WEB) sofrem desse problema, já que geram requisições que esperam um endereço IPv4 como resposta.

A técnica IVI se mostrou uma boa alternativa para redes que possuem somente *hosts* IPv6, pois podem acessar nativamente o IPv6. Por meio da técnica de mapeamento, essas redes podem acessar também a rede IPv4. O IVI não resolve o problema da falta de endereços IPv4, uma vez que necessita de endereços IPv4 roteáveis para sua implementação. Neste trabalho, o mapeamento IVI não pôde ser avaliado por limitações nos softwares disponíveis. Entretanto, o IVI possui vários formatos para implantação, tanto em mapeamentos 1:1 quanto 1:N, e isso pode ser benéfico para a adaptação a diferentes topologias de rede.

Por fim, vale ressaltar que a transição entre os protocolos é inevitável. Em algum momento futuro será necessário realizar essa tarefa. Desta forma, este trabalho teve como finalidade auxiliar futuras empreitadas sobre este tema.

5 Apêndices

Apêndice A: Configuração da Pilha Dupla em um ambiente Linux

Para exemplificar os passos necessários para utilizar a Pilha Dupla em um nó da rede, será usado o sistema operacional Linux Ubuntu 10.4. será necessário atribuir endereços de IP ao nó, para isso podem ser usados os seguintes comandos:

$ip a a 2001:461:181:f100::03/64 dev eth0 – para adicionar um endereço IPv6 a interface

$ip a a 192.168.0.100/24 dev eth0 – para adicionar um endereço IPv4 a interface.

Se necessário rotas relativas a cada pilha podem ser adicionadas ao nó:

$ip -6 route add 64:ff9b::/96 via 2001:468:181:f100::1 dev eth0 – para definir uma rota para outra rede IPv6.

$ip route add 192.168.1.1/24 via 192.168.0.1 dev eth0 – para definir uma rota para outra rede IPv4.

Apêndice B: Configuração do NAT64/DNS64 em um ambiente Linux

Para implantação do NAT64/DNS64 será usado o sistema operacional Ubuntu 10.4 e através dos passos descritos abaixo, poderá ser feita a implantação do NAT64 em uma rede:

1- Fazer o download do software desenvolvido pelo projeto ecdsys, isso pode ser feito com a url : <http://ecdysis.viagenie.ca/>.

2- Compilar o modulo *kernel,usando* os seguintes comandos:
$ sudo make & make install

3- Fazer o download do software BIND9, isso pode ser feito com a url: <http://www.isc.org/software/bind>.

4- Instalar o software BIND9 através dos seguintes comandos:
$ make

$ make install –without-openssl

5- Dentro do diretório onde foi instalado o NAT64, no arquivo nat64-config.sh comentar as seguintes linhas:

 # load the nf_nat64 module

#modprobe –r nf_nat64

#modprobe nf_nat64 nat64_ipv4_addr=$IPV4_addr=$IPV4_ADDR nat64_prefix_addr=$PREFIX_ADDR nat64_prefix_len=$PREFIX_LEN

6- O roteamento IPv6 deve ser habilitado com o seguinte comando:

$ sudo sysctl –w net.ipv6.conf.all.forwarding=1

7- Será necessário habilitar o modulo do *kernel* para isso deverá ser usado o seguinte comando:

$ insmod nf_nat64.ko nat64_ipv4_addr=$IPV4_ADDR nat64_prefix_addr=$PREFIX_ADDR nat64_prefix_len=$PREFIX_LEN

O endereço IPv4 (*$IPV4_ADDR*) escolhido na configuração para este tcc foi atribuído a interface via DHCP, já que o projeto foi configurado no campus do IFSC-sj, o prefixo IPv6 (*$PREFIX_ADDR*)escolhido foi o 64:ff9b::. e a mascara de rede IPv6 (*PREFIX_LEN*) escolhida foi uma /96.

8- O arquivo de configuração deve ser lido com o comando:

$ ./nat64-config.sh $IPV4_ADDR

9- Deve-se verificar que se o modulo foi lido corretamente com o comando:

$ lsmod

10- Deve-se verificar se a interface NAT64 está pronta para atuar, isto pode ser verificado com o comando:

$ ifconfig

11- No arquivo named.conf do BIND9 as seguintes linhas de configuração devem ser adicionadas:

```
options {
        listen-on-v6 { any; }
        dns64 64:ff9b::/96 {
                clients { any; };
                mapped { any; ;}
                sufix ::;
                recursive-only yes;
                break-dnssec Yes;
        };
};
```

12- Os nós puramente IPv6 necessitarão ter uma rota apontando para o prefixo de rede usado pelo NAT64 para converter os endereços IPv4 em IPv6, isso pode ser feito com o comando:

$ip -6 route add 64:ff9b::/96 via $NAT64_DEV_ADDR dev eth0

13- Nos nós puramente IPv6, dentro do arquivo /etc/resolv.conf, definir o *host* DNS64 como servidor DNS primário:

Nameserver $IP_DNS64

6 Referências Bibliográficas

COMER, Douglas E.. Interligação de redes com TCP/IP. 5. Ed. Rio de Janeiro: Elsevier, 2006.

CRUZ, Ademar. IPv6 – Características, 1999 Disponível em: <http://civil.fe.up.pt/acruz/Mi99/asr/caracteristicas.htm>. Acesso em: 05 maio 2012

DUNMORE, Martin. An IPv6 Deployment Guide. 1. Ed. Europa, 2005.

DLTEC. Tradução-ipv6-para-ipv4. Disponível em : http://www.dltec.com.br/blog/cisco/traducao-ipv6-para-ipv4. acesso em : 26 de junho de 2012

FOROUZAN, BEHROUZ A.. Comunicação de dados e redes de computadores. 3. Ed. Porto Alegre: Bookman, 2006.

HAGEN, Silvia. IPv6 Essentials. Editora O'Reilly, 2002.

KUROSE, James F.; ROSS, Keith W. Redes de computadores e a Internet. 1 Ed. São Paulo, 2003.

KUROSE, James F.; ROSS, Keith W. Redes de computadores e a Internet. 5 Ed. São Paulo, 2010.

MOREIRAS, Antonio M., et al. Curso IPv6 básico, 2010 Disponível em: <http://www.ipv6.br/download. Acesso em: 25 maio. 2012.

MOREIRAS, Antonio M., et al. Técnicas de Transição do IPv4 para o IPv6, 2012 Disponível em: <http://www.ipv6.br/download. Acesso em: 25 maio. 2012.

(RFC2460)S. DEERING. Internet Protocol, Version 6 (IPv6), 1998 Disponível em: <http://www.ietf.org/rfc/rfc2460.txt>. Acesso em: 03 Maio 2012.

(RFC1752) S. BRADNER. The Recommendation for the IP next Generation Protocol, 1995 Disponível em: <http://www.ietf.org/rfc/rfc1752.txt>. Acesso em: 26 mar. 2012.

(RFC2765) E. NORDMARK. Stateless IP/ICMP Translation Algorithm (SIIT),2000 Disponível em:< http://www.ietf.org/rfc/rfc2765.txt>. Acesso em: 03 maio 2012

(RFC3513)R. HINDEN. Internet Protocol Version 6 (IPv6) Addressing Architecture, 2003 Disponível em: <http://www.ietf.org/rfc/rfc3513.txt>. Acesso em: 05 Maio 2012.

(RFC6146) M. BAGNULO. Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers,2011 Disponivel em:< http://tools.ietf.org/html/rfc6146>.Acesso em: 02 maio 2012.

(RFC6147) M. BAGNULO. DNS64: DNS Extensions for Network Address Translation from IPv6 Clients to IPv4 Servers.2011 Disponivel em :< http://tools.ietf.org/html/rfc6147> .Acesso em: 02 maio 2012.

(RFC6219) CERNET CENTER. The China Education and Research Network (CERNET) IVI Translation Design and Deployment for the IPv4/IPv6 Coexistence and Transition. 2011 Disponivel em: http://tools.ietf.org/rfc/rfc6219.txt. Acesso em: 02 junho 2012

(RFC6586) J. ARKKO. Experiences from an IPv6-only network Disponivel em:< http://tools.ietf.org/html/rfc6586>. Acesso em: 02 maio 2012.

TECHNET. Roteamento IPV6. Disponível em: <http://technet.microsoft.com/pt-br/library/cc758763(v=ws.10).aspx>. Acesso em: 28 maio. 2012.

XING LI. IVI translation concept and experiences, 2009 Disponivel em: < http://www.canscouncil.net/presentations/>. Acesso em: 28 maio 2012.

XING LI. IVI IPv4/IPv6 Coexistence and Trasition, 2009 Disponivel em: < http://www.canscouncil.net/presentations/>. Acesso em: 28 maio 2012.