UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS DE TELECOMUNICAÇÕES

LIZANDRO RIVATTO
MARCELO ANTONIO MANFRON
TALITA ALINE SOLDAN

TUNNEL IPV6

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2014

LIZANDRO RIVATTO
MARCELO ANTONIO MANFRON
TALITA ALINE SOLDAN

TUNNEL IPV6

Trabalho de Conclusão de Curso de Graduação, apresentado ao Curso Superior de Tecnologia em Sistemas de Telecomunicações, do Departamento Acadêmico de Eletrônica, da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Tecnólogo.

Orientador: Prof. Dr. Augusto Foronda

CURITIBA
2014

TERMO DE APROVAÇÃO

LIZANDRO RIVATTO
MARCELO ANTONIO MANFRON
TALITA ALINE SOLDAN

TUNNEL IPV6

Este trabalho de conclusão de curso foi apresentado no dia 27 de novembro de 2014, como requisito parcial para obtenção do título de Tecnólogo em Sistemas de Telecomunicações, outorgado pela Universidade Tecnológica Federal do Paraná. Os alunos foram arguidos pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

_____
Prof. Dr. Luis Carlos Vieira
Coordenador de Curso
Departamento Acadêmico de Eletrônica

_____
Prof. Esp. Sérgio Moribe
Responsável pela Atividade de Trabalho de Conclusão de Curso
Departamento Acadêmico de Eletrônica

BANCA EXAMINADORA

_____
Prof. Dr. Kleber K. H. Nabas
UTFPR

_____
Prof. MsC. Lincoln H. Teixeira
UTFPR

_____
Prof. Dr. Augusto Foronda
Orientador - UTFPR

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso”

Agradecemos em primeiro lugar a Deus que iluminou o nosso caminho durante esta caminhada. Ao nosso orientador Prof. Dr. Augusto Foronda por todo conhecimento transmitido. Aos nossos pais pelo apoio e educação durante nossas vidas.

32 bits should be enough address space for Internet. (Vint Cerf - Honorary Chairman of IPv6 Forum/2000, 1977)

32 bits deve ser espaço de endereço suficiente para a Internet. (Vint Cerf - Presidente Honorário do Fórum IPv6 / 2000, 1977)

RESUMO

RIVATTO, Lizandro; MANFRON, Marcelo Antonio; SOLDAN, Talita Aline. **Tunnel IPv6**. 2014. 44 f. Trabalho de Conclusão de Curso (Curso Superior de Tecnologia em Sistemas de Telecomunicações), Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2014.

Devido ao esgotamento de endereços na rede, o IPv6 foi criado para substituir o IPv4, os dois protocolos podem funcionar simultaneamente. Em um primeiro momento se pensava que sua utilização seria feita de forma progressiva, de forma que os protocolos IPv4 e IPv6 funcionassem simultaneamente, porém isso não aconteceu, a realidade é que os endereços IPv4 já se esgotaram e há uma necessidade de implementar o IPv6 urgentemente, pois o uso da Internet é sempre crescente. Para usuários que querem começar o processo de implantação em suas redes, onde seus provedores de acesso ainda não oferecem suporte ao protocolo IPv6 é recomendada a utilização do Tunnel Broker.

Palavras chave: IPv4. IPv6. Internet.

ABSTRACT

RIVATTO, Lizandro; MANFRON, Marcelo Antonio; SOLDAN, Talita Aline. **Tunnel IPv6**. 2014. 44 f. Trabalho de Conclusão de Curso (Curso Superior de Tecnologia em Sistemas de Telecomunicações), Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná. Curitiba, 2014.

Due to the exhaustion of network addresses, IPv6 is designed to replace IPv4, the two protocols can operate simultaneously. At first thought that their use would be made gradually, so that the IPv4 and IPv6 protocols work simultaneously, but that didn't happen, the reality is that the IPv4 addresses are already sold out and there is a need to implement IPv6 urgently, because Internet usage is always growing. For users who want to start the deployment process on their networks, where their access providers still do not support the IPv6 protocol is recommended using the Tunnel Broker.

Keywords: IPv4. IPv6. Internet.

LISTA DE ILUSTRAÇÔES

LISTA DE SIGLAS

APIs	Application Programming Interface
ARPANET	Advanced Research Project Agency Network
AYIYA	Anything in Anything
CIDR	Classless Inter-Domain Routing
CLNS	Connectionless Network Service
DARPA	Defense Advanced Research Projects Agency
DNS	Domain Name System
DSTM	Dual-Stack Transition Method
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
IETF	Internet Engineering Task Force
IHL	Internet Header Lengh
IP	Internet Protocol
IPng	Internet Protocol next generation
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IPX	Internetwork Packet Exchange
ISATAP	Intra-Site Automatic Tunnel Addressing Protocol
MTU	Maximum Transmission Unit
NAT	Network Address Translation
NSFNET	National Science Foundation Network
OSI	Open Systems Interconnection
P2P	Peer-to-peer
PARC	Packet Universal protocolo suite
RFC	Request for Comments
RTT	Round-trip Time
TB	Tunnel Broker
TC	Tunnel Client
TCP	Transmission Control Protocol
TOS	Type of service
TS	Tunnel Server
TSP	Tunnel Setup Protocol
TTL	Time to live
UCL	University College London
UDP	User Datagram Protocol

SUMÁRIO

1. INTRODUÇÃO

O *Tunnel Broker* permite que uma rede ipv4, ou dispositivos isolados, obtenham conectividade ipv6 através de um túnel com um provedor, tornando-se uma rede pilha dupla. Esse processo é descrito na (RFC 3053, 2001).

Os *Tunnel Brokers* usam diversas tecnologias para prover os túneis. Podem usar, por exemplo, túneis 6in4, encapsulamento em UDP, o protocolo AYIYA (*Anything in Anything*) ou TSP (*Tunnel Setup Protocol),* definido na (RFC 5572, 2010).

A utilização de *Tunnel Broker* é recomendada para usuários que querem testar o IPv6, ou começar um processo de implantação em suas redes, onde seus provedores de acesso ainda não oferecem suporte ao protocolo.

Para ter acesso a um *Tunnel Broker*, é necessária uma requisição a um provedor de túnel, e realizar *download* de um *software* de configuração ou através de linhas de comando.

1.1. PROBLEMA

O IPv4 é bastante antigo. Sua capacidade de expansão já se esgotou; graves falhas de segurança, que são descobertas periodicamente e não possuem solução; muitos ataques contra computadores hoje só é possível devido a falhas no protocolo IP.

O IPv6 resolveria todos esses problemas, pois define 128 bits para endereçamento e, portanto conta com cerca de $3{,}4\times10^{38}$ endereços disponíveis; dá fim a praticamente todas as brechas de segurança conhecidos no IPv4, tornando as comunicações mais seguras.

1.2. JUSTIFICATIVA

Os protocolos IPv6 e IPv4 podem funcionar simultaneamente. Atualmente há três importantes métodos de migração do protocolo IPv4 para o IPv6 para redes particulares e/ou públicas:

- **Tunelamento:** basicamente faz a transmissões dos pacotes encapsulando o conteúdo do pacote IPv6 em um pacote IPv4;
- **Teredo:** traduz os cabeçalhos IPv4 em cabeçalhos IPv6 e vice-versa, realizando conversões de endereços, de APIs (*Application Programming Interface*) de programação, ou atuando na troca de trafego TCP (*Transmission Control Protocol*) ou UDP (*User Datagram Protocol*);
- **Pilha Dupla:** permite que hosts e roteadores estejam equipados com pilhas para ambos os protocolos, tendo a capacidade de enviar e receber os dois pacotes, IPv4 e IPv6.

1.3. OBJETIVOS

1.3.1. Objetivo Geral

Simular o acesso de uma rede IPv4 a algum site ou host IPv6.

1.3.2. Objetivos Específicos

- Análise dos protocolos (IPv4 e IPv6);
- Expor um processo de implantação em uma rede para testar o IPv6.

1.4. METODOLOGIA

O desenvolvimento deste projeto será orientado pelas simulações realizadas, normas, artigos sobre o assunto.

A primeira etapa do trabalho será baseada no contexto histórico, estudo dos temas envolvidos no título do trabalho.

Na segunda etapa será analisada e explicada a técnica e como é seu funcionamento.

A última etapa será apresentar como baixar o programa e como utilizá-lo.

2. HISTÓRICO

O TCP/IP foi desenvolvido em 1969 pelo *U.S. Departament of Defense Advanced Research Projects Agency*, como um recurso para um projeto experimental chamado de ARPANET (*Advanced Research Project Agency Network*) para preencher a necessidade de comunicação entre um grande numero de sistemas de computadores e várias organizações militares dispersas. O objetivo do projeto era disponibilizar links de comunicação com alta velocidade, utilizando redes de comutação de pacotes.

O protocolo deveria ser capaz de identificar e encontrar a melhor rota possível entre dois sites, além de ser capaz de procurar rotas alternativas para chegar ao destino, caso qualquer uma das rotas tivesse sido destruída. O objetivo principal da elaboração de TCP/IP na época foi encontrar um protocolo que pudesse tentar de todas as formas uma comunicação caso ocorresse uma guerra nuclear. A partir de 1972 o projeto ARPANET começou crescer em uma comunidade internacional e hoje se transformou no que conhecemos como Internet. Em 1983 ficou definido que todos os computadores conectados ao ARPANET passariam a utilizar o TCP/IP. No final dos anos 80 a Fundação Nacional de Ciências em Washington, D.C, começou construir o NSFNET, um *backbone* para um supercomputador que serviria para interconectar diferentes comunidades de pesquisa e também os computadores da ARPANET. Em 1990 o NSFNET se tornou o *backbone* das redes para a Internet, padronizando definitivamente o TCP/IP.

De 1973 a 1974, o grupo CERF de redes de pesquisas de Stanford trabalhou os detalhes da ideia do protocolo TCP/IP, resultando em sua primeira especificação. A influência técnica significativa foi o trabalho da Xerox PARC, que produziu o PARC (*Packet Universal protocolo suite*), muito do que existia naquela época.

DARPA então contratado pela BBN Technologies, da Universidade de Stanford e da *University College London* (UCL) para desenvolver versões operacionais do protocolo sobre diferentes plataformas de hardware. Quatro versões foram desenvolvidas: TCP v1, v2 TCP, TCP v3 e v3 IP e TCP / IP v4. O último protocolo ainda está em uso hoje.

Em 1975, foi realizado um teste de comunicação entre as duas redes TCP/IP entre Stanford e UCL. Em novembro de 1977, foi realizado um teste entre três redes TCP/IP entre os sites nos EUA, Reino Unido e Noruega. Vários outros protótipos TCP/IP foram desenvolvidos em múltiplos centros de pesquisa entre 1978 e 1983. A migração da ARPANET para o TCP/IP foi oficialmente concluído no dia da bandeira, 01 de janeiro de 1983, quando os novos protocolos foram permanentemente ativados.

2.1. ARQUITETURA TCP/IP

O modelo TCP/IP é composto por quatro camadas e embora o seu conjunto de protocolos tenha sido desenvolvido antes da definição do modelo OSI, a funcionalidade dos protocolos da camada de aplicação TCP/IP se ajusta à estrutura das três camadas superiores do modelo OSI: camadas de Aplicação, Apresentação e Sessão (CISCO, 2013). A figura 1 mostra a comparação em termos das camadas.

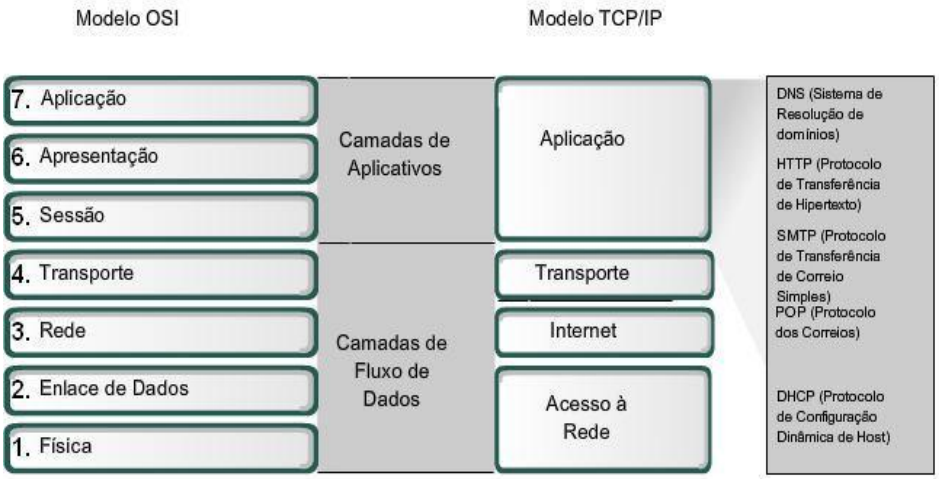

Figura 1 - Descrição do protocolo TCP/IP.
Fonte: CISCO, 2013

As camadas do modelo TCP/IP são:

- Aplicação: fornece a interface entre as aplicações que utilizamos para comunicação e a rede subjacente pelas quais nossas mensagens são transmitidas. Os protocolos da camada de aplicação são utilizados para troca de dados entre programas executados nos hosts de origem e de destino (CISCO, 2013).

- Transporte: prepara os dados de aplicativos para o transporte através da rede e processa os dados da rede para o uso pelos aplicativos. Proporciona a segmentação de dados e o controle necessário para reagrupar esses segmentos em fluxos de comunicação. Realiza esse processo através do rastreamento da comunicação individual entre as aplicações nos hosts de origem e destino, segmentando os dados e gerenciando cada segmento, reagrupando os segmentos em fluxos de dados de aplicação e Identificando as diferentes aplicações. São dois os protocolos dessa camada: o TCP (*Transmission Control Protocol*), que é orientado a conexão e garante a entrega dos dados, na ordem correta; e UDP (*User Datagram Protocol*), que opera no modo sem conexão e fornece um serviço datagrama não confiável (SOARES, 1995).
- Rede: Fornece serviços para realizar trocas de fragmentos individuais de dados na rede entre dispositivos finais identificados. Para realizar o transporte de uma ponta à outra utiliza os processos de endereçamento, encapsulamento, roteamento e desencapsulamento. Os protocolos existentes nessa camada são: *Internet Protocol version* 4 (IPv4), *Internet Protocol version 6* (IPv6), *Novell Internetwork Packet Exchange* (IPX), *AppleTalk e Connectionless Network Service* (CLNS/DECNet) (CISCO, 2013).
- Acesso à Rede: Consiste de rotinas de acesso à rede física. A camada de Interface de Rede interage com o hardware, permitindo que as demais camadas sejam independentes do hardware utilizado (COMER, 1998; SOARES, 1995). Define como o cabo está conectado à placa de rede, como por exemplo, o tipo de conector e quais pinos serão utilizados. Ela também define qual técnica de transmissão será utilizada para enviar os dados para o cabo da rede. Essa camada corresponde às camadas um e dois do modelo OSI.

2.2. INTERNET PROTOCOL – IP

Protocolo de Internet (*Internet Protocol - IP*) é um protocolo de comunicação usado entre duas ou mais máquinas em rede para encaminhamento dos dados. Tanto no Modelo TCP/IP, quanto no Modelo OSI, o importante do IP está na camada intitulada camada de rede.

O IP é o elemento comum encontrado na Internet pública dos dias de hoje. É descrito no (RFC 791, 1981). Este documento descreve o protocolo da camada de rede mais popular e atualmente em uso. Esta versão do protocolo é designada de versão quatro, ou IPv4.

2.3. INTERNET PROTOCOL VERSION 4 – IPV4

Na figura 2 temos o formato do cabeçalho IPv4:

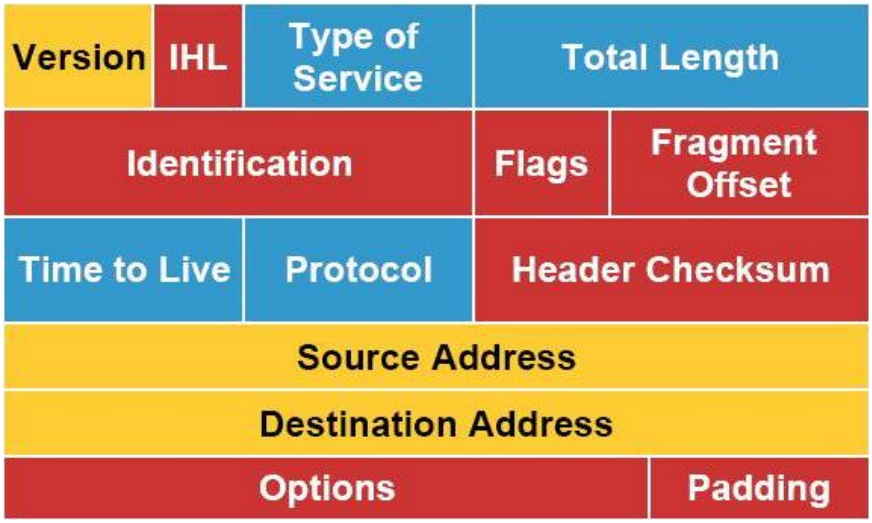

Figura 2 – Cabeçalho IPV4
Fonte: http://www.ccna-wiki.com/

Campos chaves do cabeçalho:

- ***Version:*** a versão atual é a quatro, motivo pelo qual chamamos de IPv4 o protocolo IP. O campo versão tem o tamanho de quatro bits (RFC 0791, 1981).
- **IHL (*Internet Header Lengh*):** tamanho do cabeçalho do pacote. Informa seu tamanho em palavras de 32 bits. O valor mínimo é cinco, quando não há nenhuma opção presente. O valor máximo desse campo de quatro bits é 15, o que limita o cabeçalho a 60 bytes e o campo Options a 40 bytes (RFC 0791,1981);
- **TOS (*Type of service*):** possui oito bits e é utilizado para indicar o QoS (*Quality of Service*) desejado (SOUZA, 2005). Seus bits caracterizam os serviços escolhidos para serem considerados pelos gateways para processar o pacote. Originalmente o campo de seis bits continha (da esquerda para a direita) um campo *Precedence* de três bits e três flags, D, T e R. O campo *Precedence* tinha uma

prioridade que variava de 0 (normal) a 7 (pacote de controle de rede). Os três bits de flags permitiam que o host especificasse o que era mais importante no conjunto {Retardo, *Throughp ut,* Confiabilidade} (RFC 0791,1981).

- ***Total Length***: campo de 16 bits que fornece o tamanho total do pacote em bytes, incluindo o cabeçalho e os dados. O tamanho mínimo do pacote é de 20 bytes e o máximo de 65.535 (RFC 0791, 1981).
- ***Identificacion***: identifica unicamente os fragmentos de um pacote IP original, permitindo que o host de destino determine a qual datagrama pertence um fragmento recém-chegado. Todos os fragmentos de um datagrama contêm o mesmo valor de identificação (RFC 0791, 1981).
- **Flags**: bits que identificam a transmissão de sinais de controle;
- **Fragmento *Offset***: informa a que ponto do datagrama atual o fragmento pertence. Todos os fragmentos de um datagrama, com exceção do último, devem ser múltiplos de 8 bytes, a unidade elementar de fragmento. Como são fornecidos 13 bits, existem no máximo 8192 fragmentos por datagrama, resultando em um tamanho máximo de datagrama igual a 65.536 bytes, um a mais que o campo Total *Length* (CISCO, 2013).
- ***TTL (Time to live):*** contador usado para limitar a vida útil dos pacotes. Esse campo conta o tempo em segundos, permitindo uma vida útil máxima de 255s. Ele é decrementado a cada hop e supõe-se que seja decrementado diversas vezes quando estiverem enfileirados durante um longo tempo em um roteador. Na prática, ele simplesmente conta os hops. Quando o contador chega à zero, o pacote é descartado e enviado uma advertência para o host de origem. Com isso evita-se que os datagramas fiquem vagando indefinidamente, algo que aconteceria se as tabelas de roteamento fossem danificadas (CISCO, 2013).
- ***Protocol***: este campo informa a que processo de transporte o datagrama deve ser entregue quando o mesmo estiver montado por completo. O número do TCP, por exemplo, é seis, UDP é 17 e ICMP igual a um. O campo protocolo tem o tamanho de oito bits (RFC 0791, 1981);
- ***Header Checksum***: utilizado somente para verificação de erros no cabeçalho do pacote. Em cada salto o *checksum* do cabeçalho é comparado com o valor deste campo. Se o valor não corresponder ao *checksum* calculado, o pacote é

descartado. Em cada salto, o campo TTL é reduzido e o *checksum* é recalculado em cada salto (CISCO, 2013).

- ***Source Address***: segundo a CISCO este campo informa o endereço de origem do host que está enviando o pacote (CISCO, 2013).
- ***Destination Address***: este campo de endereço é destinado ao host que receberá o pacote (CISCO, 2013).
- ***Options***: este campo foi projetado para permitir que versões posteriores do protocolo incluam informações inexistentes no projeto original, possibilitando a experimentação de novas ideias e evitando a alocação de bits de cabeçalho para informações raramente necessárias (CISCO, 2013).
- ***Padding***: tamanho variável, entre 0 e 31 bits. Serve apenas para que o cabeçalho IP tenha um tamanho múltiplo de 32 bits e é feito seu preenchimento (obrigatoriamente com 0), somente se o tamanho do campo *Options* não for múltiplo de 32 bits (CISCO, 2013).

Dentro do intervalo de endereço de cada rede IPv4, temos três tipos de endereço de acordo com *Networking Academy* da CISCO:

- Endereço de rede: dentro do intervalo de endereços IPv4 de uma rede, o primeiro endereço é reservado para o endereço de rede. Esse endereço possui o valor 0 para cada bit de host do endereço (CISCO, 2013).
- Endereço de *broadcast*: endereço especial usado para enviar dados a todos os hosts da rede. O envio de dados para todos os hosts em uma rede pode ser feito por um host que envia um único pacote que é endereçado para o endereço de broadcast da rede (CISCO, 2013).
- Endereços de *host*: os endereços designados aos dispositivos finais da rede. (CISCO, 2013).

Existem três tipos de endereços de hosts que não podem ser usados para uma comunicação com outro host individual. São eles:

- Endereços Experimentais: um intervalo principal de endereços reservados para propósitos especiais IPv4 que vão de 240.0.0.0 a 255.255.255.254. Atualmente, esses endereços são registrados como reservados para uso futuro (RFC 3330, 2002). Atualmente, não podem ser usados em redes IPv4, mas podem ser usados para pesquisa ou testes;

- Endereços *Multicast*: os endereços *multicast* IPv4 de 224.0.0.0 a 224.0.0.255 são endereços locais de link reservados. Esses endereços são usados para grupos *multicast* em uma rede local. Os pacotes para esses destinos sempre são transmitidos com um valor TTL igual a 1. Portanto, um roteador conectado à rede local nunca deve encaminhá-los. Uma utilização típica é o de endereços locais de link reservados para protocolos de roteamento usando transmissão *multicast* para trocar informações de roteamento. Os endereços globalmente restritos são de 224.0.1.0 a 238.255.255.255. Eles podem ser usados para dados *multicast* pela Internet;
- Endereços de *Host*: depois de retirado os intervalos reservados para endereços experimentais e de *multicast* foi determinado o intervalo de 0.0.0.0 a 223.255.255.255 para utilização dos hosts IPv4. Contudo, dentro desse intervalo há muitos endereços reservados para fins especiais, denominados de endereços privados que são: de 10.0.0.0 a 10.255.255.255 (10.0.0.0 /8), de 172.16.0.0 a 172.31.255.255 (172.16.0.0 /12) e de 192.168.0.0 a 192.168.255.255 (192.168.0.0 /16) para utilização em redes privadas.

O endereçamento IP é estruturado em classes em que parte do endereço IP representa o endereço da rede e a outra parte, o endereço do computador dentro da rede.

Um endereço IP é um endereço de 32 bits, geralmente notado sob a forma de 4 números inteiros separados por pontos. Distinguem-se, com efeito, duas partes no endereço IP:

- Uma parte dos números à esquerda designa a rede e chama-se *netID*.
- Os números à direita designam os computadores desta rede e chamam-se *host-ID*.

Classes dos endereços IPv4:

- **Classe A**: nessa classe, o primeiro byte representa o número da rede e os outros três bytes, o número do host. Está classe permite representar 126 redes e 16.777.214 hosts.
- **Classe B**: nessa classe, os dois primeiros bytes representam o número da rede e os outros dois bytes, o número do *host*. Permite representar 16.000 redes e 64.000 hosts para cada uma das redes.

- **Classe C**: nessa classe, os três primeiros bytes representam o número da rede e o último byte o número do *host*. Permite representar mais de 2 milhões de redes e 254 hosts para cada uma das redes.
- **Classe D/E**: nessa classe, todos os bytes representam um endereço *broadcasting* para envio de mensagens a toda rede.

Além disso, os 32 blocos /8 restantes foram reservados para *Multicast* e para a IANA (*Internet Assigned Numbers Autorithy*).

2.4. INTERNET PROTOCOL VERSION 6 – IPV6

IPv6 é a versão mais atual do Protocolo de Internet. Originalmente oficializada em 6 de junho de 2012, é fruto do esforço do IETF para criar a "nova geração do IP" (*IPng: Internet Protocol next generation*), cujas linhas mestras foram descritas por Scott Bradner e Allison Marken, em 1994, na (RFC 1752, 1995). Sua principal especificação encontra-se na (RFC 2460, 1998).

O protocolo está sendo implantado gradativamente na Internet e deve funcionar lado a lado com o IPv4, numa situação tecnicamente chamada de "pilha dupla" ou "*dual stack*", por algum tempo. Em longo prazo, o IPv6 tem como objetivo substituir o IPv4, que só suporta cerca de 4 bilhões (4x10^9) de endereços IP, contra cerca de 3,4x10^38 endereços do novo protocolo.

O IP versão 6 começou a ser desenvolvido no início da década de 1990, com o objetivo de ser a solução definitiva para o esgotamento de endereços IPs na Internet. Tendo esse como o principal objetivo.

Outra diferença em relação à versão anterior do protocolo é em relação ao espaço de endereçamento, aumentado de 32 bits para 128 bits.

Os endereços passam a ser representados por números hexadecimais de 16 bits, separados por (dois pontos “:”). É indiferente representar as letras com maiúsculas ou minúsculas, e algumas abreviações são possíveis, como a omissão de zeros à esquerda e a representação de um conjunto contínuo de zeros por “::”. As redes são representadas como no CIDR (*Classless Inter-Domain Routing*), utilizado no IPv4, utilizando a “/”, seguida do número de bits representativos da sub-rede.

Na figura 3 temos o formato do cabeçalho IPv6:

Figura 3 – Cabeçalho IPV6
Fonte: http://www.ccna-wiki.com/

O cabeçalho do IPv6 tem menos informação que o cabeçalho do IPv4. Por exemplo, o *checksum* será removido do cabeçalho, uma vez que nesta versão considera-se que o controle de erros das camadas inferiores é confiável.

- **Version (versão – 4 bits):** Usado para os roteadores identificarem qual o protocolo do pacote, no caso, 6.
- ***Traffic Class* (classe de tráfego - 8 bits):** usado para assinalar a classe de serviço a que o pacote pertence, permitindo assim dar diferentes tratamentos a pacotes provenientes de aplicações com exigências distintas. Este campo serve de base para o funcionamento do mecanismo de qualidade de serviço (QoS) na rede.
- ***Flow Label* (identificação de fluxo - 20 bits):** usado com novas aplicações que necessitem de bom desempenho. Permite associar datagramas que fazem parte da comunicação entre duas aplicações. Usados para enviar datagramas ao longo de um caminho pré-definido. É classificado como fluxo orientado, aquele que demanda muitos pacotes, e fluxo não orientado, aquele que não demanda muitos pacotes, muito tráfego. Dentro de cada categoria, há um identificador de fluxo que sugere o tratamento daquele caso.
- ***Payload Length* (comprimento dos dados - 16 bits):** representa o volume de dados em bytes que pacote transporta.
- ***Next Header* (próximo cabeçalho - 8 bits):** aponta para o primeiro *header* de extensão. Usado para especificar o tipo de informação que está a seguir ao cabeçalho corrente.

- ***Hop Limit* (limite de saltos - 8 bits):** tem o número de *hops* transmitidos antes de descartar o datagrama, ou seja, este campo indica o número máximo de saltos (passagem por encaminhadores) que o datagrama pode dar, antes de ser descartado, semelhante ao TTL (time to live) do IPv4.
- ***Source Address* (Endereço da fonte - 128 bits):** identifica o endereço de origem do pacote.
- ***Destination Address* (Endereço de destino - 128 bits):** identifica o endereço de destino do pacote.

No IPv6, o responsável pela fragmentação é o host que envia o datagrama, e não os roteadores intermediários como no caso do IPv4. Os roteadores intermediários descartam os datagramas maiores que o MTU (*Maximum Transmission Unit*) da rede - MTU máximo suportado pelas diferentes redes entre a origem e o destino. Para isso o host envia pacotes ICMP (*Internet Control Message Protocol*) de vários tamanhos. Quando um pacote chega ao *host* destino, todos os dados a serem transmitidos são fragmentados no tamanho deste pacote que alcançou o destino.

O processo de descoberta do MTU tem que ser dinâmico, porque o percurso pode ser alterado durante a transmissão dos datagramas.

No IPv6, um prefixo não fragmentável do datagrama original é copiado para cada fragmento. A informação de fragmentação é guardada num cabeçalho de extensão separado. Cada fragmento é iniciado por uma componente não fragmentável seguida de um cabeçalho do fragmento.

2.5. TRANSIÇÃO IPV4 / IPV6

A palavra chave na transição entre as duas versões do protocolo IP é interoperação. As duas versões podem permanecer na rede simultaneamente, se comunicando e endereçando. A segunda palavra chave é facilidade, poder dar um upgrade nos softwares da versão 4 para a 6, tanto para administradores de rede, técnicos, como para o usuário final.

Os objetivos da transição são:

- Roteadores e máquinas devem ter seus programas de rede trocados sem que todos os outros no mundo tenham que trocar ao mesmo tempo;

- Pré-requisitos mínimos. O único pré-requisito é que os servidores de DNS (*Domain Name System*) devem ter a sua versão trocada antes. Para os roteadores não existem pré-requisitos.

2.6. TÉCNICA DE TRANSIÇÃO ENTRE REDES IPV4 / IPV6

Quando as máquinas sofrerem o upgrade devem poder manter seus endereços IPv4, sem a necessidade de muitos planos de um re-endereçamento.

Nós IPv6 devem poder se comunicar com outros nós IPv6, mesmo que a infraestrutura entre eles seja IPv4.

Para o último objetivo, dois mecanismos foram trabalhados:

- ***Dual-stack***: com esse mecanismo, nodos IPv6 devem ter as duas pilhas TCP/IP internamente, a pilha da versão 6 e a da versão 4. Através da versão do protocolo, se decide qual pilha processará o datagrama. Esse mecanismo permite que nodos já atualizados com IPv6 se comuniquem com nodos IPv4, e realizem roteamento de pacotes de nodos que usem somente IPv4. Os nodos com *dual-stack* usam o "mesmo" endereço para ambos os pacotes - sejam IPv4 ou IPv6. Nodos que trabalham apenas com IPv4 podem enviar pacotes para nodos *dual-stack* usando endereçamento IPv4, enquanto nodos que trabalham com IPv6 podem enviar pacotes utilizando para isto endereçamento IPv6.
- ***Tunneling***: esse mecanismo consiste em transmitir um datagrama IPv6 como parte de dados de um datagrama IPv4, a fim de que dois nodos IPv6 possam comunicar-se através de uma rede que só suporte IPv4. A rede IPv4 é vista como um túnel, e o endereço IPv4 do nodo final deste túnel consta como destino do datagrama. Neste nodo o pacote IPv6 volta a trafegar normalmente a seu destino. Esse nodo final, portanto, deve ter a pilha que suporte IPv6.

Com o intuito de facilitar o processo de transição entre as duas versões do Protocolo Internet, algumas técnicas foram desenvolvidas para que toda a base das redes instaladas sobre IPv4 mantenha-se compatível com o protocolo IPv6. Cada uma dessas técnicas apresenta uma característica específica, podendo ser utilizada individualmente ou em conjunto com outras técnicas, de modo a atender as necessidades de cada situação, seja a migração para o IPv6 feita passo a passo,

iniciando por um único host ou sub-rede, ou alcançando toda uma rede corporativa de uma vez.

2.7. TUNELAMENTO

A técnica de criação de túneis permite transmitir pacotes IPv6 através da infraestrutura IPv4 já existente, sem a necessidade de realizar qualquer mudança nos mecanismos de roteamento, encapsulando o conteúdo do pacote IPv6 em um pacote IPv4.

Essas técnicas, têm sido as mais utilizadas na fase inicial de implantação do IPv6, por serem facilmente aplicadas em testes, onde há redes não estruturadas para oferecer trafego IPv6 nativo (RFC 4213, 2005). Os túneis podem ser configurados nos seguintes modos:

- ***Host-a-Host***: permite a *hosts dual-stack* conversarem entre si por uma rede IPv4, utilizando pacotes IPv6 encapsulados em pacote IPv4, conforme figura 4. Consiste em uma comunicação direta tipo P2P, é utilizada na maioria dos tipos de tunelamento utilizados.

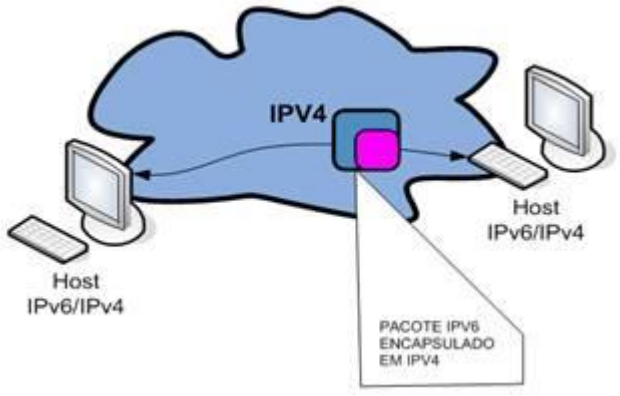

Figura 4 – Configuração Host a Host
Fonte: Gomes (2009)

- ***Host*-a-Roteador**: *Hosts* IPv6/IPv4 enviam pacotes IPv6 a um roteador IPv6/IPv4 intermediário via rede IPv4, ligando o primeiro segmento no caminho entre dois *hosts*, permitindo a comunicação entre esses dois *hosts* por IPv6, conforme figura 5:

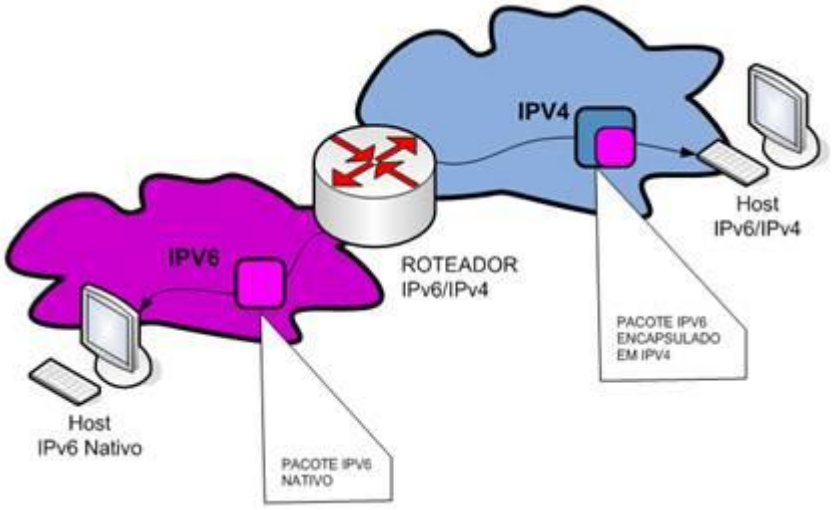

Figura 5 – Configuração Host a Roteador
Fonte: Gomes (2009)

- **Roteador-a-Roteador:** *gateways dual-stack* IPv6/IPv4 e com uma conexão IPv4 entre si são configurados para trocarem pacotes IPv6 de redes IPv6 passando por uma rede IPv4 permitindo a comunicação de dois segmentos de rede IPv6, exemplo na figura 6:

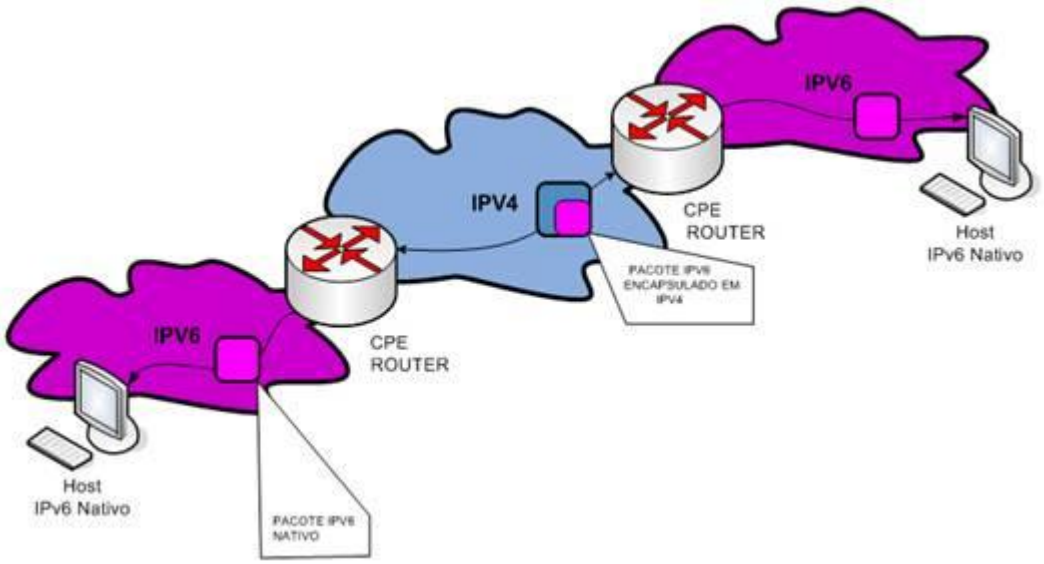

Figura 6 – Configuração Roteador a Roteador
Fonte: Gomes (2009)

O encapsulamento, de uma forma geral, é algo simples e dinâmico. O primeiro *host* (A) pega o pacote IPv6 e o insere em um pacote com cabeçalho IPv4 e então o transmite. O *host* de destino (B) recebe esse pacote IPv4, desencapsula retirando o cabeçalho IPv4 e processa o pacote IPv6 recebido, conforme figura 7:

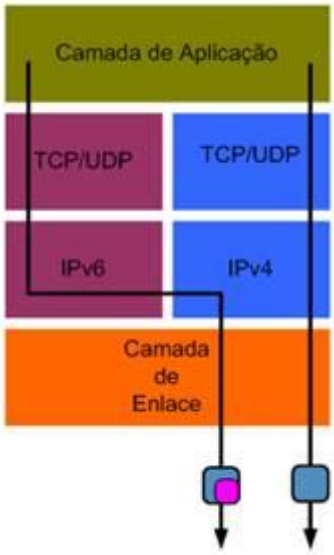

Figura 7 – Pilha Dupla IPV6 encapsulado IPV4
Fonte: Gomes (2009)

Os túneis podem ser criados com configuração manual, que podem utilizar mecanismos genéricos de encapsulamento. Há, também, mecanismos de criação semiautomáticas de túneis, como, por exemplo, os serviços de túnel Broker e existem, também, mecanismos totalmente automáticos para a criação de túneis, por exemplo: o 6to4, o ISATAP ou o Teredo.

Túneis manuais são usados entre dois pontos e necessitam da configuração dos endereços de origem e destino do túnel. Os túneis automáticos necessitam apenas serem ativados, e o respectivo protocolo é responsável pela criação e manutenção dos túneis.

As variedades de cenários existentes colaboram com a existência de diversos tipos de túneis, com variações em desempenho, implementação e segurança. Este tipo de solução é útil quando se deseja conectar ilhas IPv6 isoladas, no meio de “oceanos” IPv4.O tunelamento requer que os nodos IPv6 em ambas as partes do túnel sejam capazes de transmitir pacotes IPv4.

O processo de encapsular IPv6 dentro de IPv4 é similar ao método de encapsulação de outros protocolos: o nodo de um dos lados do túnel pega o datagrama IPv6 e envia como sendo dados do *payload* para o nodo que está do outro lado do túnel. O resultado é um *stream* de datagramas IPv4 que contém datagramas IPv6.

Existem diversas técnicas de tunelamento. E a técnica que será objeto de pesquisa nesse trabalho é a *Tunnel Broker*. Que permite que dispositivos isolados, ou toda uma rede IPv4, obtenha conectividade IPv6 por meio do estabelecimento de um túnel com um provedor, tornando-se, na prática, dispositivos, ou uma rede, pilha dupla.

3. TUNNEL BROKER

Seu funcionamento é bastante simples: primeiramente é necessário realizar um cadastro, normalmente via Web, em um provedor que ofereça esse serviço, chamado, neste contexto, de *Tunnel Broker*. O provedor realizará de forma automática, ou semi automática, a configuração do seu lado do túnel e permitirá o download de instruções, ou de um *software* ou *script* de configuração, para configurar o lado do usuário. Os *Tunnel Brokers* normalmente oferecem blocos fixos IPv6 que variam de /64 a /48.

Os *Tunnel Brokers* podem usar tecnologias diversas para prover os túneis. Podem usar, por exemplo, túneis 6in4, encapsulamento em UDP, o protocolo AYIYA, que significa *Anything in Anything*, ou TSP (*Tunnel Setup Protocol*), definido na RFC 5572.

A utilização de *Tunnel Brokers* é recomendada para usuários domésticos e corporativos que querem testar o IPv6, ou começar um processo de implantação em suas redes, mas cujos provedores de acesso ainda não oferecem suporte ao protocolo. Muitos Sistemas Autônomos brasileiros têm utilizado com sucesso túneis com a *Hurricane Electric* para anunciar seus blocos em caráter de teste e muitas empresas e usuários domésticos têm utilizado túneis SixXS para familiarizar-se com o IPv6.

A implantação de um serviço de *Tunnel Broker* em um provedor de Internet não é trivial, pois não há softwares abertos disponíveis para a funcionalidade de Servidor Broker.

As figuras 8 e 9 mostram a topologia lógica do *Tunnel Broker*:

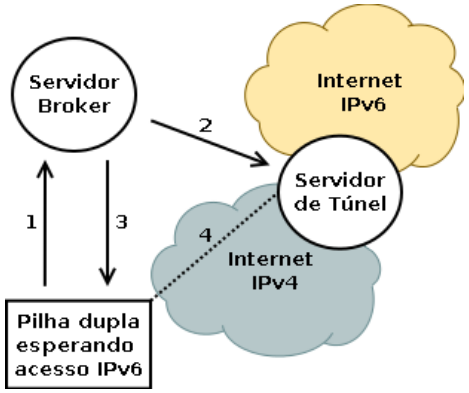

Figura 8 – Topologia lógica do *Tunnel Broker*
Fonte: http://ipv6.br/

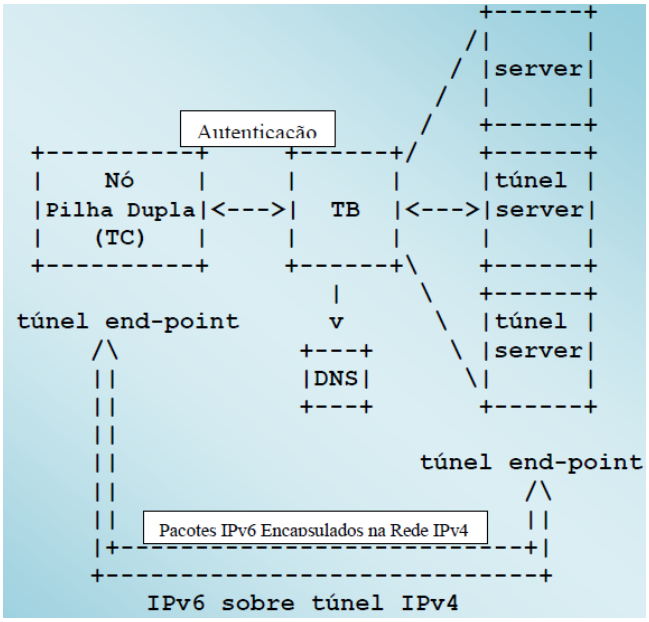

Figura 9 – Topologia lógica do *Tunnel Broker*
Fonte: http://ipv6.br/

A figura 10 mostra a topologia física do *Tunnel Broker*:

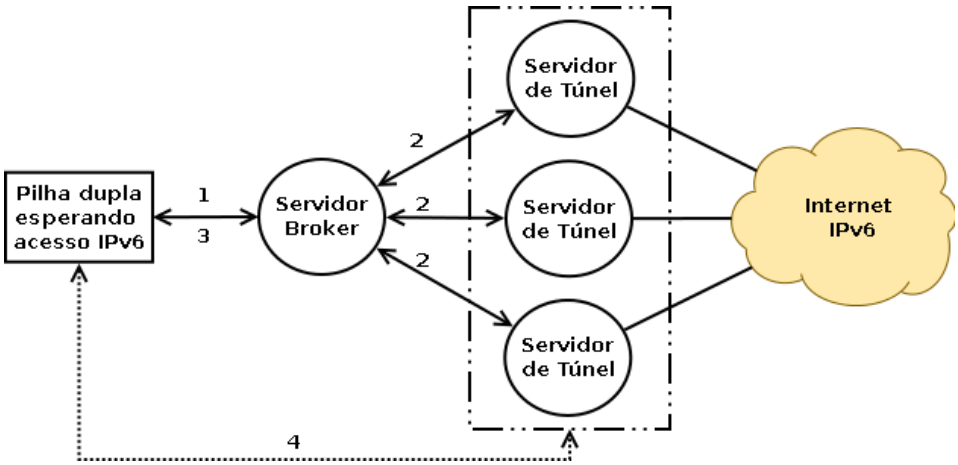

Figura 10 – Topologia física do Tunnel Broker
Fonte: http://ipv6.br/

1 – Cliente pilha dupla solicita túnel (pode ser solicitada autenticação) via IPv4
2 – Broker cadastra usuário no Servidor de túnel
3 – Broker informa cliente parâmetros para criação do túnel
4 – Túnel estabelecido

A ideia do *Tunnel Broker* é uma abordagem alternativa baseada na oferta de servidores dedicados, a gerenciar automaticamente os pedidos vindos dos túneis dos usuários.

O túnel pode ser visto como um provedor virtual, que proporciona conectividade para os usuários já conectados na rede IPv4.

O TC (*Tunnel Client*) envia um pacote pela Internet IPv4 para autenticar-se e requisitar o serviço do TB (*Tunnel Broker*). O *Tunnel Broker* é onde o usuário se conecta para registrar e ativar o túnel. Na continuidade, o *Tunnel Broker* gerencia a criação, alteração e exclusão do túnel pelo usuário e cria registros para utilizar de nomes de IPv6 no DNS.

O TS (*Tunnel Server*) é um roteador de pilha dupla (IPv4 e IPv6) conectado a Internet global. Após a recepção de uma ordem do *Tunnel Broker*, o *Tunnel Server* cria, modifica ou excluí o servidor de cada túnel, podendo também manter as estatísticas dos mesmos. O *Tunnel Server* é o servidor que fecha o túnel com o cliente do túnel, trabalhando para fazer a convergência entre o IPv6 e o IPv4.

O usuário do *Tunel Broker* é um roteador IPv6 de Pilha-Dupla (*dual-stack*) conectado a internet IPv4. Antes do usuário se conectar com o *Tunnel Broker*, o cliente deve se identificar e inserir as credenciais de autenticação do usuário, de modo que o túnel seja adequado conforme a configuração. O *Tunnel Broker* é o responsável por receber as requisições de túnel e autenticação dos seus clientes e também é o responsável por fazer as trocas de pacotes IPv6 e IPv4 entre o *Tunnel Server* e o *Tunnel Client* para o fechamento do túnel.

Após a autorização do cliente a acessar o serviço, se a maquina cliente conter um serviço de roteamento IPv6, ela estará disposta a distribuir endereços de IPv6 para vários host.

O *Tunnel Server* gerencia o cliente da seguinte maneira:

- Escolhe o prefixo IPv6 a ser alocado para o cliente;
- Determina uma vida útil para o túnel;
- Registra automaticamente no DNS os endereços de IPv6 globais;
- Configura o TB;

Notifica informações relevantes para a configuração do cliente, incluindo parâmetros do túnel e registros de DNS.

O cliente deve especificar a quantidade de endereços IPv6 que será utilizado, possibilitando assim que o roteador possa resolver a conectividade para vários host na rede.

Os tipos de endereçamento IPv6 recebido pelo *tunnel broker* são *unicast* global, o mesmo que receberia diretamente do provedor de Internet.

Após as etapas de configuração serem concluídas, o túnel IPv6 sobre IPv4 estará ativado e operando, permitindo que o usuário possa ter acesso ao 6bone ou qualquer outra rede IPv6.

3.1. TUNNEL VIA FREENET 6 – GOGO6

Para realizar os testes para o estudo do *Tunnel Broker* foi utilizado o programa da gogo6 - Freenet6.

Para download do programa, é necessário estar logado no site: http://www.gogo6.com/profile/gogoCLIENT. Na figura 11 mostra a página inicial do gogo6:

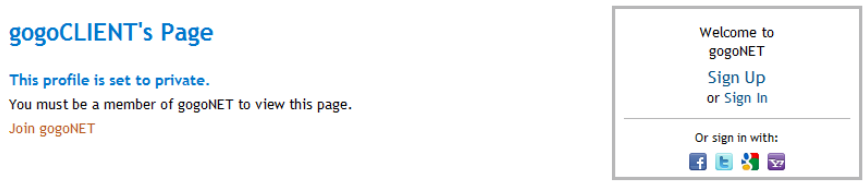

Figura 11 – Página Inicial
Fonte: Túnel via Freenet 6

Caso seja cadastrado, no menu direito, clique em "*Sign In*" ou conecte-se utilizando uma das redes sociais listadas.

Caso contrário, clique na opção "*Join gogoNET*" para realizar seu cadastro. Preencha o formulário de cadastro e espere pelo e-mail de confirmação, ou conecte-se utilizando alguma rede social.

Em ambos os casos, será necessário o preenchimento de informações adicionais para completar seu cadastro.

Na figura 12 mostra a página para criar a conta para acesso do gogo6:

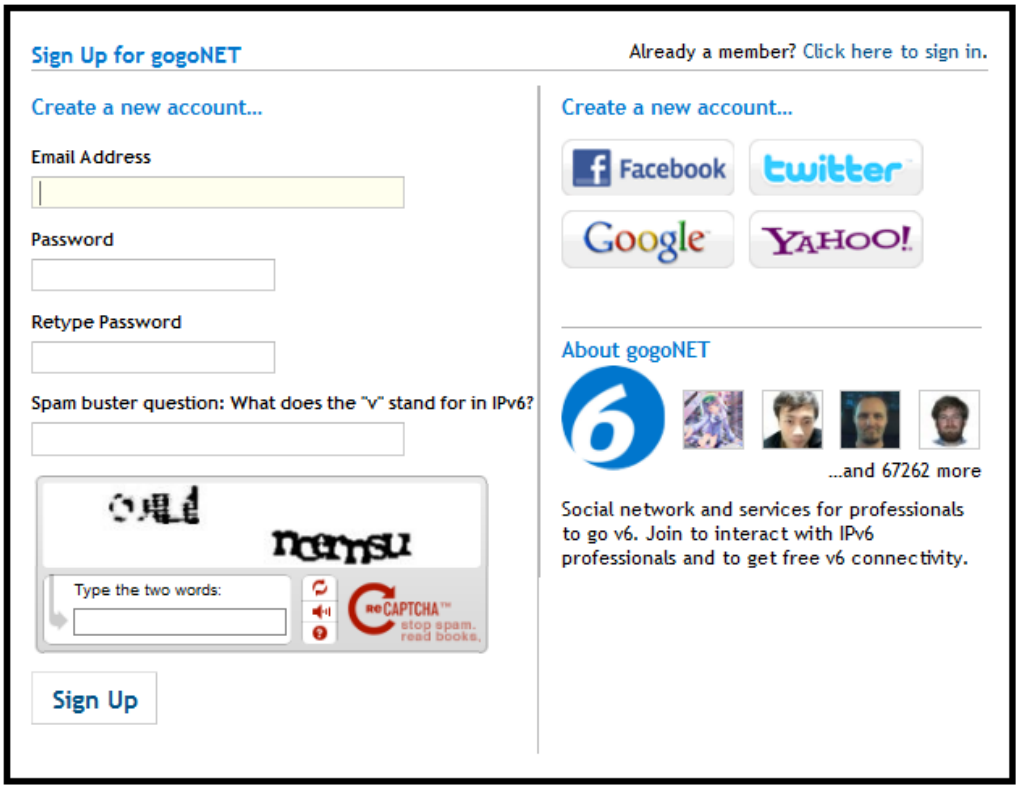

Figura 12 – Criar conta
Fonte: Túnel via Freenet 6

3.2. INSTALAÇÃO NO WINDOWS (XP, VISTA E 7)

A instalação para Windows é bastante intuitiva possuindo interface gráfica.O tutorial abaixo foi realizado utilizando Windows 7, mas, usuários das versões XP e Vista podem se basear nele para realizar a instalação e configuração do *tunnel broker*.

Já logado, acesse novamente a página (http://www.gogo6.com) para realizar o *download* do gogoCLIENT.

Escolha a versão desejada para Windows (32 ou 64 bits) e faça *download* do arquivo, conforme figura 13:

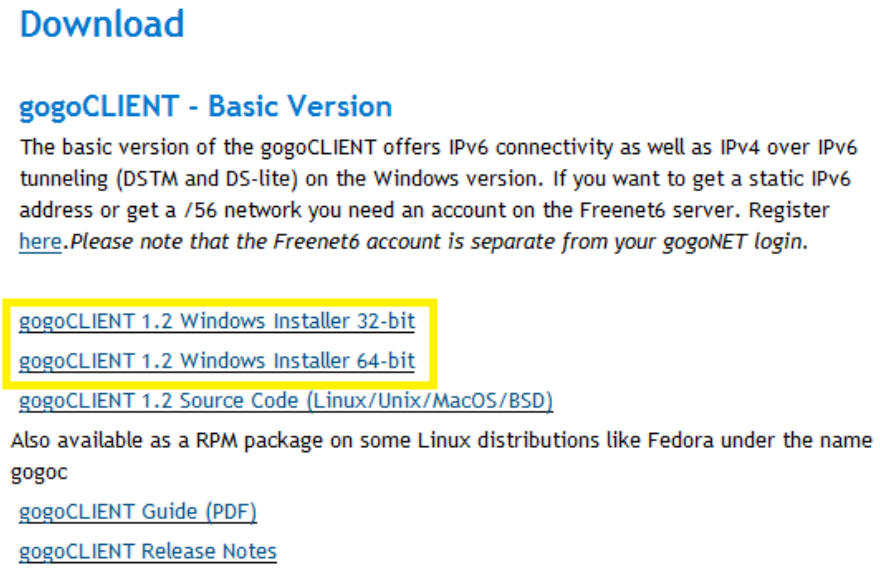

Figura 13 – Download da Versão
Fonte: Túnel via Freenet 6

Execute o arquivo, conforme figura 14:

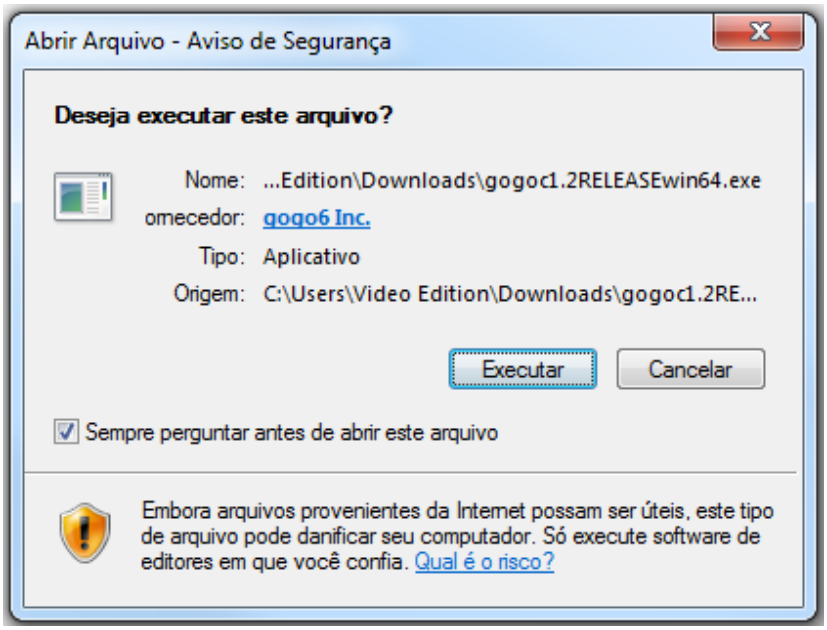

Figura 14 – Executar
Fonte: Túnel via Freenet 6

Aceite os termos de uso, conforme figura 15:

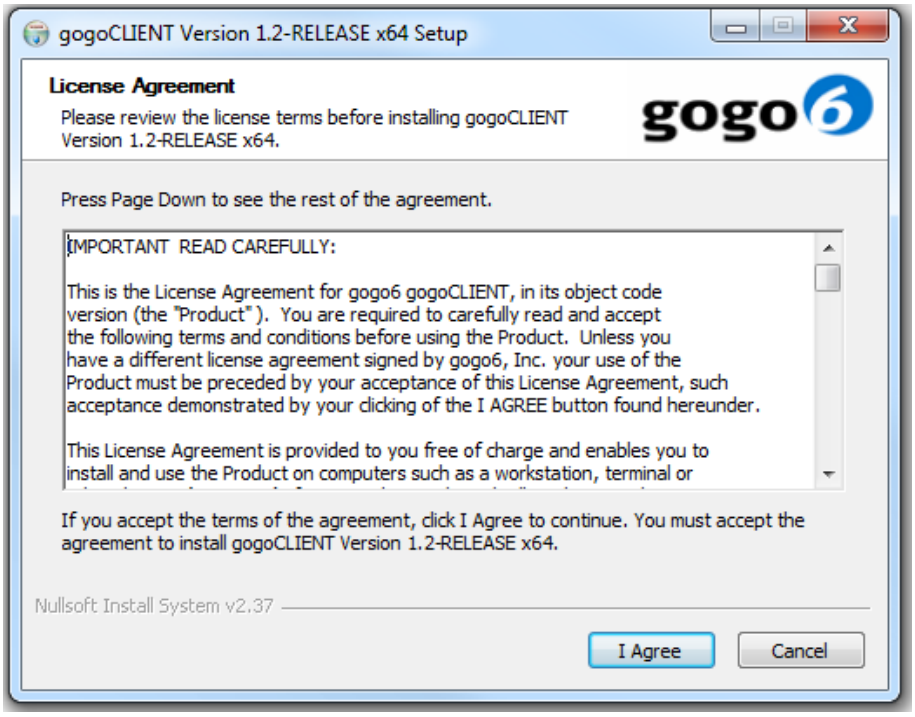

Figura 15 – Termos de Uso
Fonte: Túnel via Freenet 6

Instale todos os componentes, conforme figura 16:

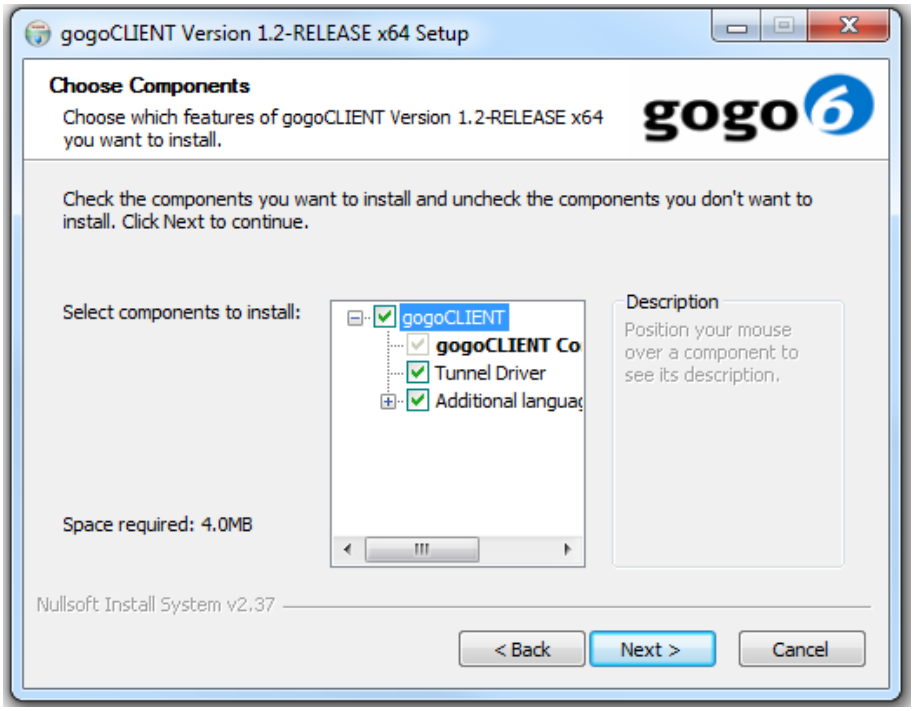

Figura 16 – Componentes
Fonte: Túnel via Freenet 6

- *gogoCLIENT*: Obrigatório e essencial para o funcionamento do cliente.

- *Tunnel Driver*: Desselecione esse componente se o *gogoSERVER Client* nunca estiver atrás de NAT ou se não DSTM não é necessário (para a conectividade IPv4 em IPv6).
- *Aditional Languages:* Idiomas adicionais recomenda-se deixar selecionado.

Se necessário, escolha a pasta desejada para instalação. Clique "Install" para continuar, conforme figura 17:

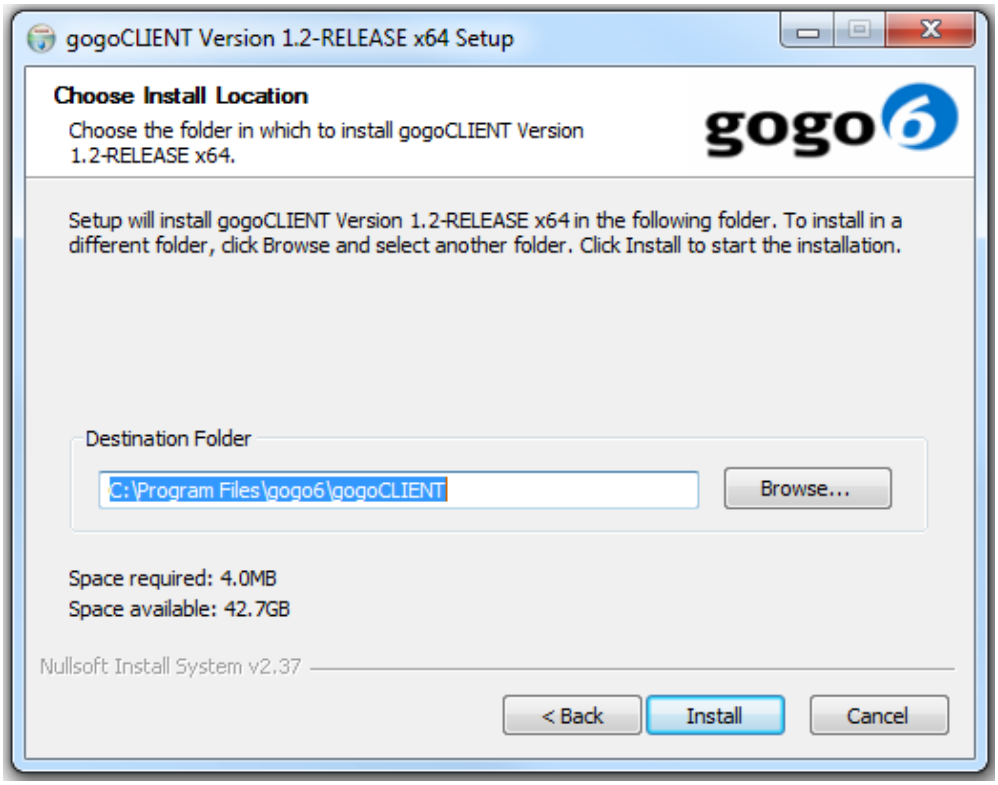

Figura 17 – Instalação
Fonte: Túnel via Freenet 6

Na próxima tela, será exibida uma mensagem de aviso do *Windows*, clique em "Instalar" o gogo6 Adaptador de Rede, conforme figura 18:

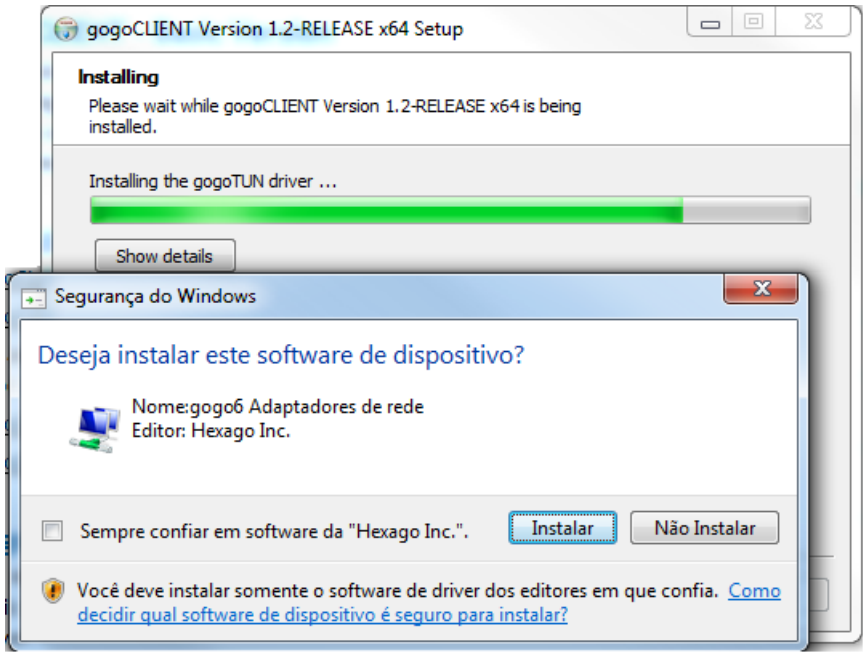

Figura 18 – Adaptador de Rede
Fonte: Túnel via Freenet 6

Selecione se deseja ler o arquivo *README* e abrir o aplicativo. Clique em "*Finish*" para terminar a instalação, conforme figura 19:

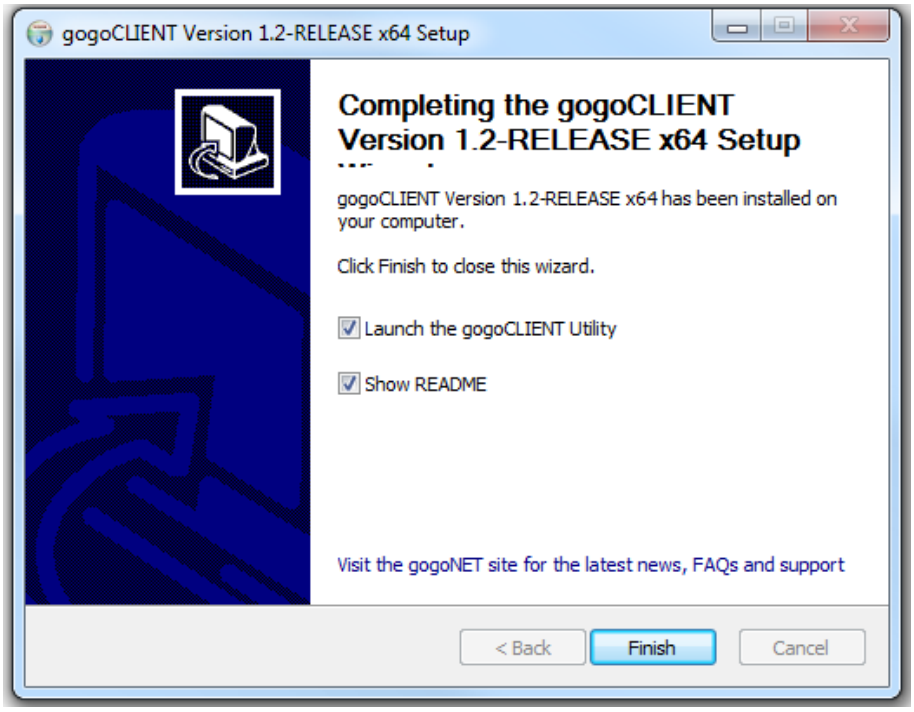

Figura 19 – Finalização da Instalação
Fonte: Túnel via Freenet 6

Se a opção "*Lauch the gogoCLIENT Utility*" foi selecionada, o programa abrirá após o final da instalação. Caso contrário, para executar o programa clique em Iniciar-> Programas-> gogo6-> *gogo6CLIENT -> gogo6CLIENT Utility,* conforme figura 20:

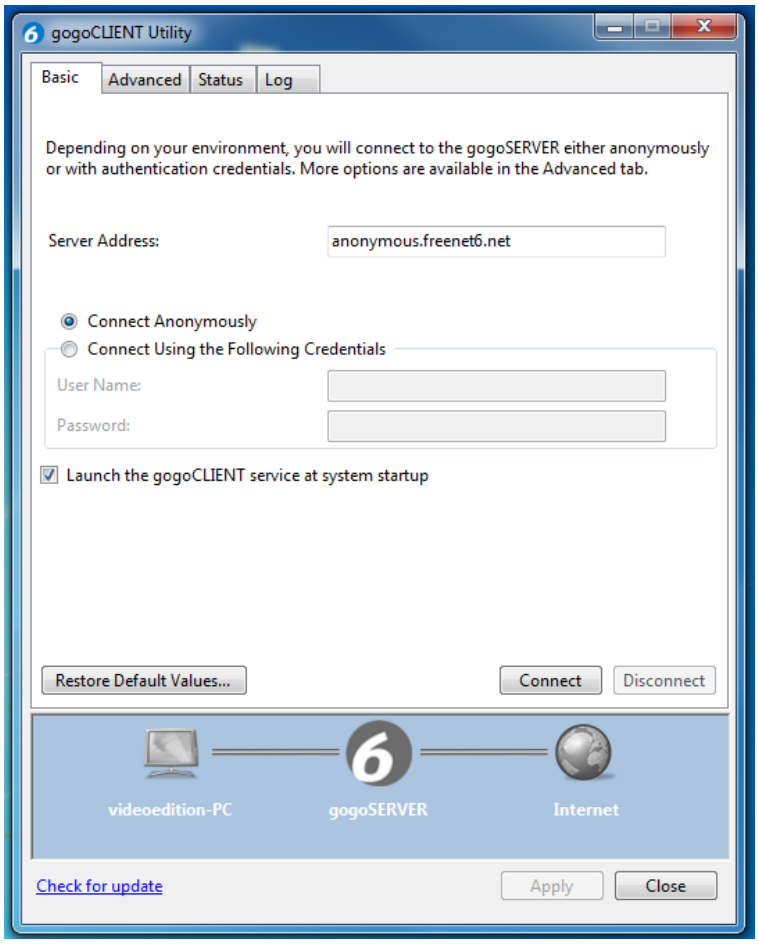

Figura 20 – gogoCLIENT Utility
Fonte: Túnel via Freenet 6

Abra o programa, clique em "*Connect*", se quiser, salve a configuração atual. Quando a imagem inferior ficar colorida, o túnel já estará funcionando. No *Windows* 7, o programa também exibe uma mensagem indicando o endereço e tipo de túnel, conforme figura 21:

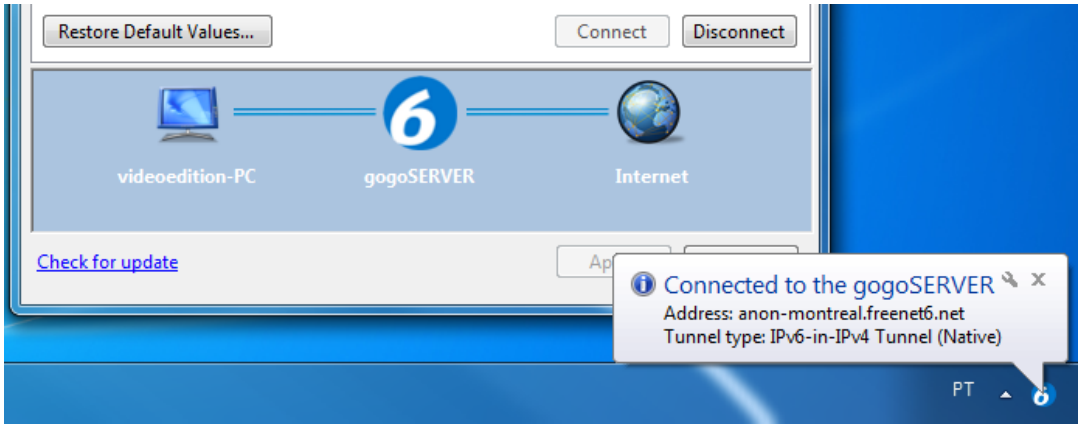

Figura 21 – Funcionamento Windows
Fonte: http://ipv6.br/tunel-freenet

Para testar a conexão, tente conectar nos seguintes sites :

- http://www.google.com.sixxs.org
- http://www.cnn.com.sixxs.org
- http://www.wikipedia.org.sixxs.org

Uma vez estabelecida, se desejar interromper a conexão, clique em "*Disconnect*".

O programa é composto de quatro abas principais e, a qualquer momento, pode se pressionar a tecla F1 para obter ajuda (em inglês) sobre as abas.O guia completo (gogoCLIENT *Guide*) pode ser encontrado na página de download do gogoCLIENT (http://www.gogo6.com/profile/gogoCLIENT).Além da descrição da funcionalidade completa do programa, fornece a lista de parâmetros de configuração e exemplos de configurações (IPv6.br).

Aba Basic: Permite obter conectividade IPv6 com o mínimo de configuração possível. Exemplo na figura 22:

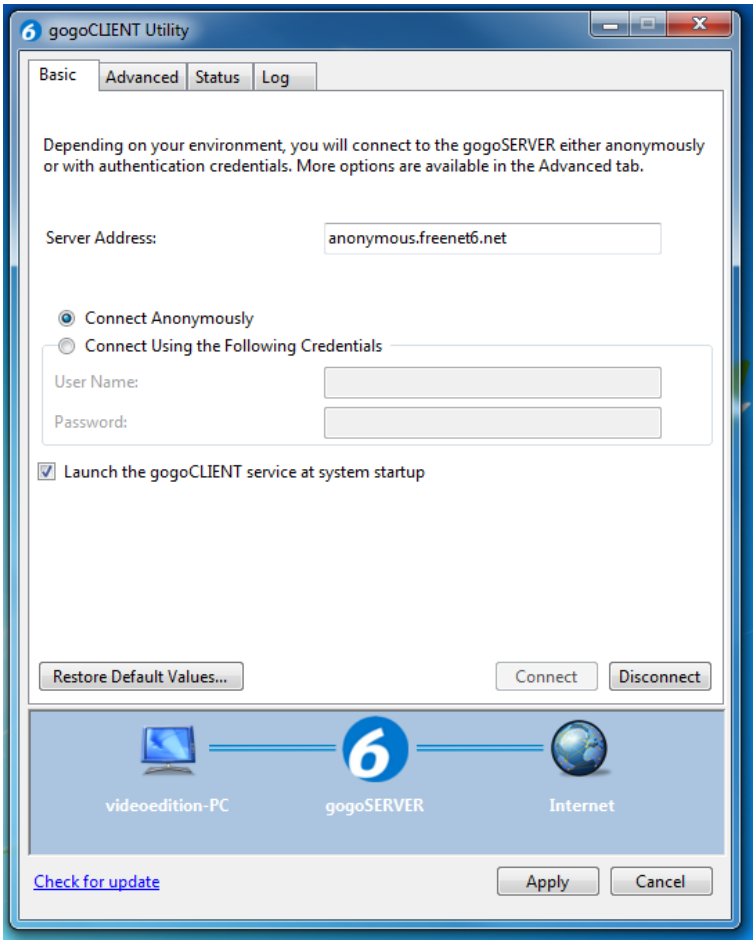

Figura 22 – Aba Basic
Fonte: Túnel via Freenet 6

Server Address: é o endereço da Internet atribuído ao *gogoSERVER*, cujo *gogoCLIENT Utility* se comunicará para estabelecer o túnel.

Anomymous Connection: O endereço IP obtido é renovado dinamicamente, não é necessário realizar autenticação.

Authenticated Connection: O endereço obtido é estático e é necessário realizar autenticação com seu nome de usuário e senha.

Launch the gogoCLIENT service at system startup: se selecionado, o serviço será iniciado automaticamente durante a inicialização do sistema operacional.

Aba Advanced: Utilizada para configurações avançadas, como o tipo de túnel e autenticação a ser utilizado. Exemplo na figura 23:

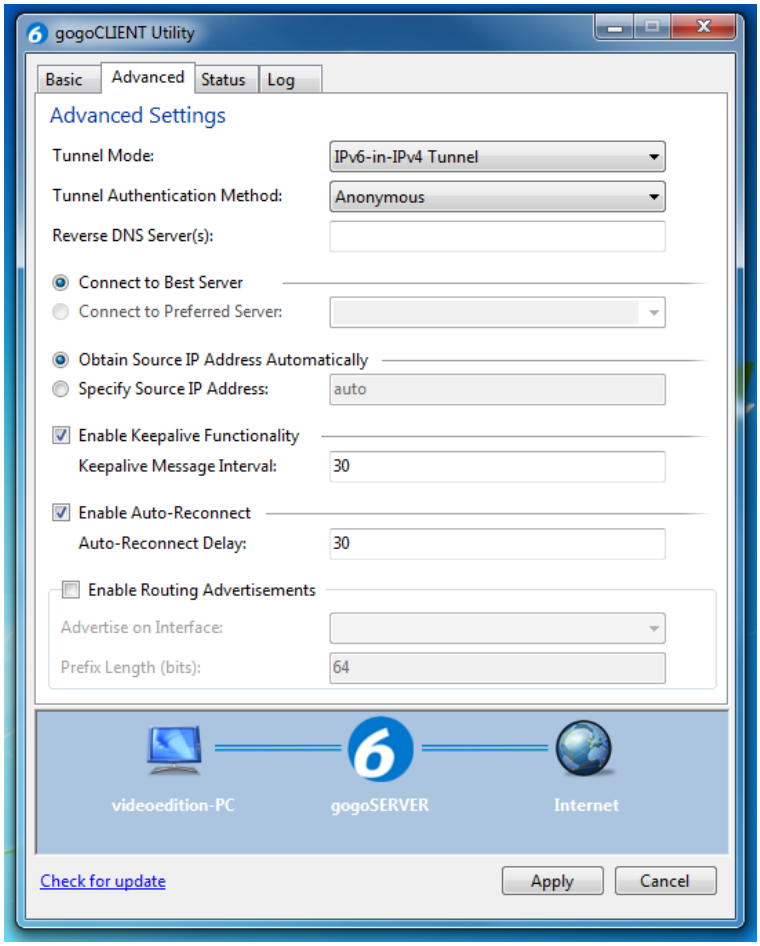

Figura 23 – Aba Advanced
Fonte: Túnel via Freenet 6

Tunnel Mode: são disponibilizados cinco tipos de túneis:

- *IPv6-in-IPv4 Tunnel*: valor padrão, escolhe automaticamente, se é uma rede nativa ou se esta utilizando NAT;
- *IPv6-in-IPv4 Tunnel* (nativo);
- *IPv6-in-IPv4 Tunnel* (NAT Traversal);
- IPv4-in-IPv6 (DSTM);
- IPv4-in-IPv6 (DS-Lite);

Tunnel Authentication Method: Valor padrão é *Anonymous*, mas pode ser alterado, está relacionado à escolha de *Anomymous Connection* ou *Authenticated Connection* na aba Basic, no caso do *Authenticated Connection*, pode definir se os dados serão criptografados, por exemplo.

DNS Server(s): utilizado para identificar os DNS servers que resolverão os domínios obtendo os valores de IPv4 e IPv6; devem ser separados por dois pontos, (sem necessidade de espaços).

Connect Using Best Broker e Connect Using Preferred Broker: utilizados apenas quando se estabelece túneis via *broker redirection* (o *Server Address* na aba *Basic* retorna uma lista de servidores capazes de criar túneis, sendo escolhido o menor RTT).

Obtain Source IP Address Automatically e Specify Source IP Address: O *Specify Source IP Address* é recomendado se o nó local possui muitos adaptadores de rede. Normalmente, a primeira opção (obter o endereço automaticamente) é adequada.

Enable Keepalive Functionality: Persiste a conexão enviando periodicamente pacotes ICMP para o servidor.

Enable Routing Advertisements: se desejar utilizar o nó local como fornecedor de endereços IPv6 para os nós da mesma rede física IPv4, habilite esta funcionalidade.

Aba Status: Fornece informações sobre o estado atual da conexão e estatísticas de uso. Exemplo na figura 24:

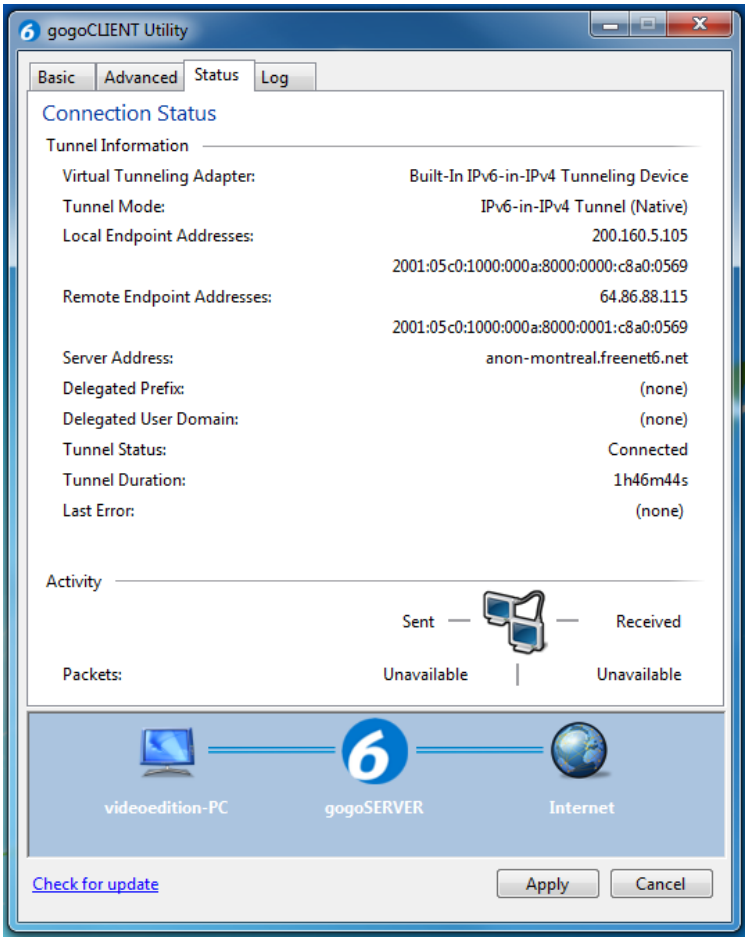

Figura 24 – Aba Status
Fonte: Túnel via Freenet 6

Aba Log

Útil para encontrar erros. Uma vez criado, o arquivo é estático, sendo atualizados somente na próxima tentativa de conexão (IPv6.br). Exemplo na figura 25:

Enable Logging to File: Habilita a manutenção de log.

Logging Level: Possui três níveis de log.

Verbose: útil para encontrar problemas.

Debug: informações de debug.

Log File Name: Nome do arquivo de log.

Log File Rotation Size: Tamanho de cada arquivo de log. Ao atingir o valor estabelecido, o arquivo será renomeado com a data atual, e outro será criado, continuando a salvar os dados.

Open Log Window: Abre o arquivo de log.

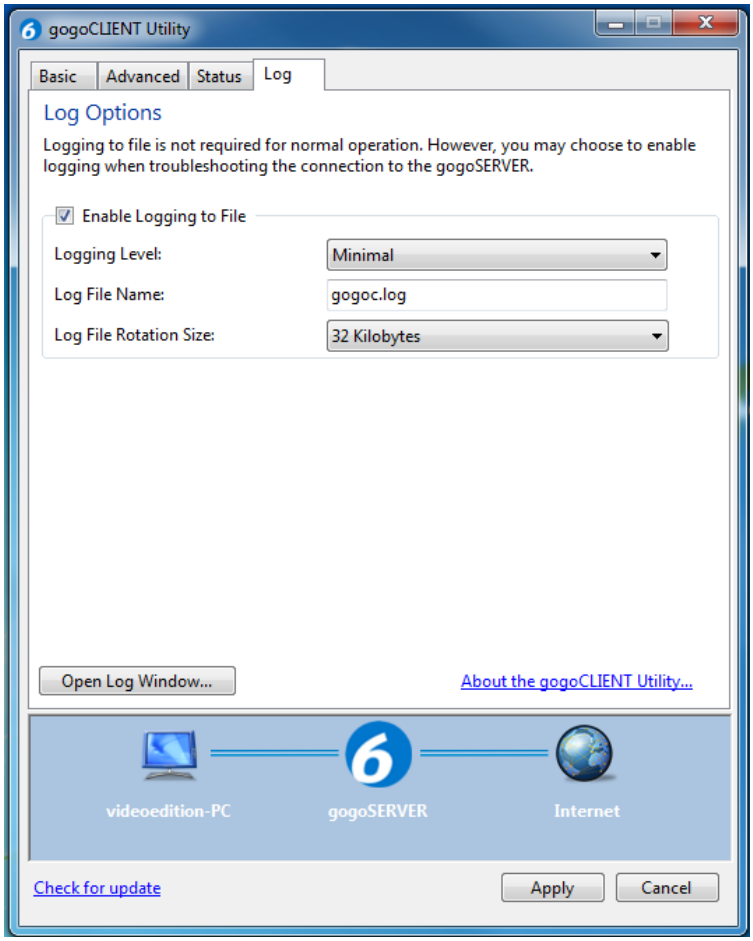

Figura 25 – Aba Log
Fonte: Túnel via Freenet 6

3.3. RESULTADO

Na figura 26 vemos a descrição do teste do google para verificação se não é um robô rodando pois foi detectada uma rede IPV6:

Figura 26 – Verificação do Google
Fonte: Autoria Própria

No site http://test-ipv6.com/ conseguimos realizar um teste no site http://test-ipv6.com, conforme figura 27:

Figura 27 – Teste de conectividade IPv6
Fonte: Autoria Própria

Na figura 28 conseguimos ver o resultado do teste no site http://test-ipv6.com:

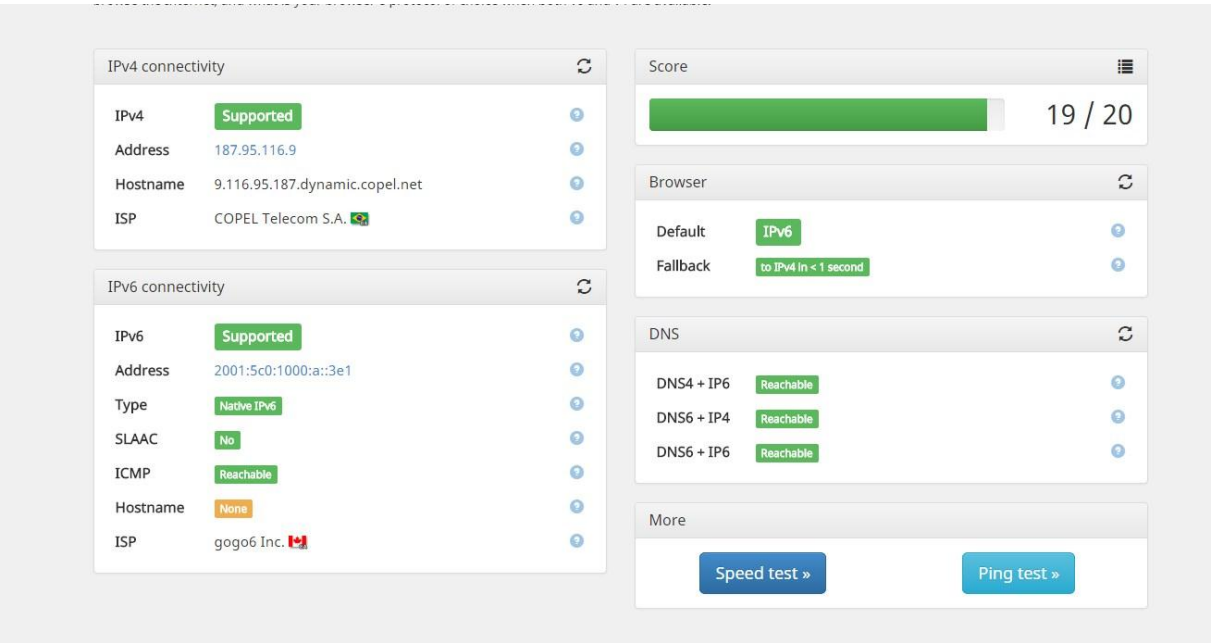

Figura 28 – Resultado do teste no site: test-ipv6.com
Fonte: Autoria Própria

Na figura 29 conseguimos ver o resultado do teste no site http://validador.ipv6.br/:

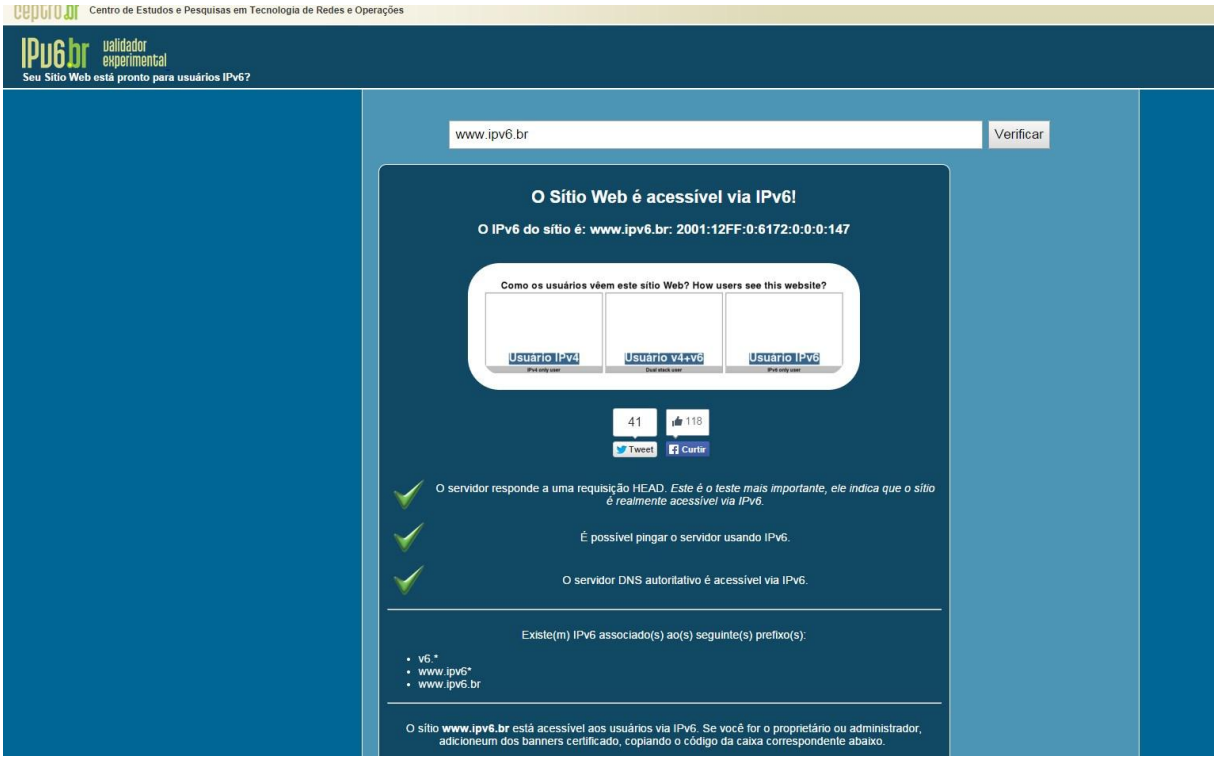

Figura 29 – Resultado do teste no site: validador.ipv6.br
Fonte: Autoria Própria

4. CONCLUSÃO

A proposta do trabalho é permitir a qualquer pessoa acessar a internet introduzindo IPv6 em sua rede. Foi explicado em passo a passo, para melhor entendimento.

Como cada dispositivo conectado a rede tem um endereço único, exclusivo e com o crescimento desacelerado da internet, o esgotamento dos endereços IPv4 já é uma realidade, algumas novas aplicações que poderiam ser geradas, não serão.

A ideia central é incentivar a implantação planejada, a maioria dos dispositivos de rede atualmente estão aptos a lidar tanto com o IPv4, quanto com o IPv6. O IPv4 não irá desaparecer de uma hora para outra, haverá uma fase de transição, porém aqueles que queiram já utilizar o IPv6 em sua rede, poderão fazer seguindo a explicação do nosso trabalho.

REFERÊNCIAS

CISCO. **Cisco Networking Academy.** Disponível em: <http://www.cisco.com/web/learning/netacad/index.html>

COMER, DOUGLAS E. **"Interligação em Rede com TCP/IP"**, 3 ed. Elserver, 1998.

Descrição do IPv4 e IPv6. Disponível em: < http://www.ccna-wiki.com/>

GOMES, Alexandre José Camilo; TRINDADE, Carlos Botelho. **Melhores práticas de migração de uma rede IPv4/IPv6.** 2009. 168 f. Trabalho de Graduação de Curso – Engenharia Elétrica com ênfase em telecomunicações. Instituto de Educação Superior de Brasília, Brasília, 2009

RFC 0791 - **Internet Protocol. Darpa Internet Program. Protocol Specification.** Disponível em: <https://tools.ietf.org/html/rfc791>

RFC 1752 - **The Recommendation for the IP Next Generation Protocol.** Disponível em: <https://tools.ietf.org/html/rfc2460>

RFC 2460 - **Internet Protocol, Version 6 (IPv6).** Disponível em: <http://tools.ietf.org/html/rfc1752>

RFC 3053 - **IPv6 Tunnel Broker.** Disponível em: <http://tools.ietf.org/html/rfc3053>

RFC 3056 - **Connection of IPv6 Domains via IPv4 Clouds.** Disponível em: <http://tools.ietf.org/html/rfc3056>

RFC 4213 - **Basic Transition Mechanisms for IPv6 Hosts and Routers.** Disponível em: <https://tools.ietf.org/html/rfc4213>

RFC 5572 - **IPv6 Tunnel Broker with the Tunnel Setup Protocol.** Disponível em: <http://tools.ietf.org/html/rfc5572>

SOARES, Luiz Fernando Gomes, LEMOS, Guido, COLCHER, Sérgio. **Redes de Computadores.** 2.ed. Rio de Janeiro: Campus, 1995.

SOUZA, Jorge Moreira. **Qualidade de Serviço (QoS).** Dependabilidade: Teleco – Informação em Telecomunicações. 2005.

Túnel via Freenet 6. Disponível em: <http://ipv6.br/tunel-freenet/>

Tunnel via Freenet 6 – gogo6. Disponível em: <http://www.gogo6.com/profile/gogoCLIENT>

Transição IPv4 / IPv6. Disponível em: http://penta2.ufrgs.br/redes296/ipv6/transi.htm