



**CURSO DE ESPECIALIZAÇÃO EM ADMINISTRAÇÃO EM
REDES LINUX**

**IPV6
A NOVA GERAÇÃO DE COMUNICAÇÃO**

WILSON MIRANDA JÚNIOR

2006

Wilson Miranda Júnior

IPv6
A Nova Geração de Comunicação

Monografia apresentada como pré-requisito de conclusão do Curso de Pós-Graduação de Administração em Redes Linux do Centro Universitário Federal de Lavras em Lavras-MG.

Prof. Joaquim Quinteiro Uchôa
Orientador

LAVRAS
MINAS GERAIS - BRASIL
2006

Wilson Miranda Júnior

IPv6
A Nova Geração de Comunicação

Monografia apresentada ao departamento de Ciência de Computação da Universidade Federal de Lavras, como parte das exigências do curso de Pós-Graduação *Lato Sensu* em Administração de Redes Linux, para obtenção do título de Especialista em Redes Linux

APROVADA em 29 de Setembro de 2006.

Prof. MSc. Denilson Vedoveto Martins

Prof. MSc. Simone Markenson Pech

Prof. Joaquim Quintero Uchôa
Orientador

LAVRAS
Minas Gerais – Brasil
2006

Agradeço e Dedico minha
conquista de mais um passo na vida
a minha Esposa, aos meus Pais
Ambos (*in memoriam*),
especialmente a minha Mãe que
nunca mediu seus esforços para que
eu pudesse atingir meus objetivos.

Wilson Miranda Júnior

Resumo

Este trabalho serve como referência ao conhecimento do Ipv6, novo modelo de comunicação e de tramitação dos pacotes em rede TCP/IP.

Com o crescimento da internet , a escassez de endereços e falta de segurança, aborda-se nessa monografia como o novo protocolo de internet IPv6 pode atender as exigências de endereçamento. Provendo o novo formato de subdivisões de redes baseadas em prefixos e como seu formato permite atribuir confidencialidade, integridade e privacidade diretamente na camada de IP.

Este trabalho tem como objetivo principal tratar da implementação do IPv6 integrado com o uso de software livre no qual foi configurado endereçamento, resolução de nomes e roteamento.

LISTA DE FIGURAS

Figura 2.1:	Classes de endereço IP	08
Figura 4.1:	Comparação e exposição do Cabeçalho IPv6 x IPv4	20
Figura 4.2:	Datagrama IPv6 com cabeçalhos de extensão	23
Figura 4.3:	Formação do cabeçalho de extensão IPv6	24
Figura 4.4:	Hop-by-Hop	25
Figura 4.5:	Encaminhamento	25
Figura 4.6:	Fragmentação e encapsulamento do datagrama	27
Figura 4.7:	Endereçamento IPv6 embutido com endereçamento IPv4	31
Figura 4.8:	Endereçamento IPv4 mapeado com endereçamento IPv6	31
Figura 4.9:	Endereçamento Unicast	33
Figura 4.10:	Identificação de Agregação do Próximo Nível	34
Figura 4.11:	Divisão NLA	35
Figura 4.12:	Subnets SLA ID	36
Figura 4.13:	Endereçamento de link local	37
Figura 4.14:	Endereçamento do Site Local	38
Figura 4.15:	Funcionamento Multicast	39
Figura 4.16:	Distribuição Anycast	40
Figura 6.1:	Envio de pacotes pelo Mobile IPv4	48
Figura 6.2:	Visão Geral IPv6	51
Figura 6.3:	Modos de Comunicação entre CN e MN	52
Figura 7.1:	Cabeçalho de Autenticação	66
Figura 7.2:	Cabeçalho de Encapsulamento de Dados de Segurança	66
Figura 8.1:	Topologia de Implementação	71

LISTA DE TABELAS

Tabela 3.1.: Tabela dos países integrados ao IPv6	15
Tabela 4.1.: Divisão dos endereços IPv6	30
Tabela 5.1.: Prioridades	44

Acrônimos

CIDR	Classless InterDomain Routing
CLNP	Connection-Less Network Protocol
IAB	Internet Architecture Board.
IANA	Internet Assigned Numbers Authority
ICANN	Internet Corporation for Assigned Names and Numbers - Corporação para Atribuição de Nomes e Números na Internet
IETF	Internet Engineering Task Force.
IP	Internet Protocol
IPng	Internet Protocol Next Generation.
IPv6	Internet Protocol version 6.
NGtrans	Next Generation Transition
IPng Transition	Internet Protocol Next Generation Transition
SIPP	Simple IP Plus
TCP	Transmission Control Protocol
WAN	Wide Area Network

Sumário

1 - INTRODUÇÃO.....	1
2- O PROTOCOLO TCP/IP.....	3
2.1 – Modelo TCP/IP.....	4
2.1.2 - Funcionalidade do IPv4.....	6
2.1.3 - Endereçamento IPv4.....	7
2.2 – Comentários Finais	9
3- IPv6 - O PROTOCOLO DA NOVA GERAÇÃO.....	11
3.1 – Necessidades de uma nova versão do IP.....	11
3.2 – Histórico da Evolução do IP.....	11
3.3 – Instituições Ligadas ao IPv6 - IETF, NGTRANS e o 6BONE....	13
3.3.1 - IETF.....	13
3.3.2 – NGTrans.....	14
3.3.3 - 6Bone.....	14
3.4 - Principais Objetivos de Criação do IPv6.....	17
3.5 - Os Caminhos da Padronização.....	18
4- CARACTERÍSTICAS DO IPv6.....	19
4.1 - Datagrama IPv6.....	21
4.1.1 - Cabeçalhos Simplificados.....	22
4.1.2 - Cabeçalhos Estendidos IPv6.....	22
4.2 - Maior Endereçamento.....	29
4.2.1 – Divisão de Endereços.....	29
4.2.2 - Transição e Codificação de Endereços IPv4.....	31
4.3 - Endereços Unicast.....	32
4.3.1 –Endereços Globais Agregados (Aggregable Global Address)	
.....	33
4.3.2 – Endereçamento de link local (Link-Local Address).....	36
4.3.3 – Endereçamento do Site Local (Site-Local Address).....	37
4.3.4 – Endereçamento não Especificado (Unspecified Address)....	38
4.3.5 – Endereçamento de Retorno (Loopback Address).....	38
4.4 – Multicast.....	39
4.5 - Anycast.....	40
4.6 - Nova Notação.....	40

4.7 - Autoconfiguração.....	41
5- SERVIÇOS IPv6.....	43
5.1 - QOS.....	43
5.1.1 Flow Label	43
5.1.2 - Prioridade	44
5.2 - Mapeamento de Endereços em Nomes.....	45
6- MOBILIDADE.....	46
6.1 - Mobile IPv4.....	46
6.2 - Mobile IPv6.....	49
6.2.1 – Funcionalidade IPv6 Mobile	49
6.3 - Operação do Host Móvel.....	52
6.4 - Considerações de Segurança.....	56
6.5 - Futuro do MIPv6.....	58
6.6 - Mobile IPv6 x Mobile IPv4.....	59
7- SEGURANÇA.....	62
7.1 - IP Security Protocol (IPSec).....	62
7.1.1 - IPSec - Características.....	63
7.1.2 - Cabeçalhos de Autenticação (AH).....	65
7.1.3 - Cabeçalho de Encapsulamento de Dados de Segurança.....	66
7.1.4 - Mecanismos de Gestão de Chaves.....	67
8 – IMPLEMENTAÇÕES IPv6 – IPv4.....	70
8.1 - Implementação de duas máquinas com suporte nativo IPv6.....	70
8.2 - Análise e compatibilidade funcional de sistema operacional.....	71
8.3 - Instalação de servidores de DNS IPv6.....	72
8.4 - Resolução de problemas de endereçamento.....	74
8.5 - Definição, implementação e análise de políticas de roteamento..	76
8.6 - Análise de compatibilidade e interoperacionalidade IPv4/IPv6.....	77
9 - CONCLUSÃO.....	79
BIBLIOGRAFIA.....	80

1 - INTRODUÇÃO

A motivação para o tema desta monografia, IPv6 – O Protocolo da Nova Geração, foi selecionado por ser um assunto de pesquisa inovador, que direciona o desenvolvimento acadêmico, fornecendo os fundamentos necessários à compreensão do funcionamento de redes de computadores baseadas no conjunto de protocolos TCP/IP. Devido a influência que o protocolo de comunicação IP desempenha em relação a Internet em diversos meios de comunicação, é apresentado nesse trabalho estudos referentes à atualização dessa nova tecnologia.

O objetivo deste trabalho de pós-graduação é tratar de alguns aspectos históricos e técnicos da evolução do IP, base da pilha de protocolos TCP/IP, situação atual global para apontar indicativos mediante necessidades de melhorias. Este trabalho deve contribuir como fonte de estudo para aqueles que buscam alternativas mais seguras e estruturadas para conexões diversas, desejando obter garantia de maior desempenho para aplicações que utilizam a Internet como meio de transmissão. Isso é feito focalizando principalmente a metodologia de funcionamento e eficácia deste novo protocolo universal de comunicação, IPv6.

O trabalho também permite a compreensão do comportamento tecnológico perante as novas tendências para o mundo da Internet, em que o protocolo IP tem o papel principal de intermediar toda essa interação entre o homem, a máquina, e a rede Internet. O *software* livre adotado no trabalho tem como fundamental papel de prover subsídios necessários para apoio na implementação do IPv6. Com isso, foi possível atribuir uma estrutura para base de novas pesquisa em relação ao protocolo de comunicação entre redes TCP/IP junto ao Linux, explorando novos desafios entre endereçamento, roteamento,

resolução de nomes, integração entre duas versões diferentes dos protocolos IPv4 e IPv6.

Para apresentação deste trabalho, adotou-se a estrutura apresentada a seguir. Nos capítulos 1 e 2 é exposto a existência do protocolo IPv4 apresentando fundamentação teórica e o protocolo TCP/IP como sua arquitetura, modelo, funcionalidade e endereçamento. O capítulo 3 trata da necessidade de existência do protocolo de nova geração atribuindo um histórico de sua existência e informando o poder de atribuição aos seus serviços agregados. Em continuidade, no capítulo 4, faz-se uma comparação entre os protocolos IPv4 e IPv6, mencionando as características do IPv6 entre arquitetura, modelo e funcionalidade. Os capítulos 5, 6 e 7 estão voltados aos serviços de principais necessidades de melhorias no IPv4, tais como qualidade de serviço, mobilidade e segurança. Finalizando com o capítulo 8 é apresentado o *software* livre (Linux) como base das atribuições e implementações de endereçamento do IPv6, junção ao IPv4, roteamento IPv6 e resolução de nomes em IPv6. Trata-se de um modelo prático no qual representa toda fundamentação de pesquisa desenvolvida para realização dessa monografia.

2- O PROTOCOLO TCP/IP

A plataforma TCP/IP surgiu através dos trabalhos do DARPA (*Defense Advanced Research Projects Agency*) dos Estados Unidos, em meados da década de 1970, constituindo a ARPANet, que mais tarde se desmembrou em ARPANet, para pesquisa, e MILNET, para instituições militares (ROBERTO, 1999). Para encorajar os pesquisadores universitários a adotar o TCP/IP, o DARPA fez uma implementação de baixo custo, integrando-o ao UNIX da Universidade de Berkeley (BSD), já em uso em todas as universidades americanas. Além disso, teve-se o cuidado de definir aplicações de rede similares às já conhecidas em Unix, como *rusers* e *rnp*.

Mais tarde a NSF (*National Science Foundation*) estimulou o seu crescimento criando a NSFnet, que ligava centros de supercomputação espalhados por todo o país, numa rede de longa distância, também com os protocolos TCP/IP. Há aproximadamente 30 anos, nascia o protocolo IP e, com ele, nascia também o embrião da *Internet*. O IP versão 4 é a que está em uso atualmente. O sucesso desse protocolo de comunicação é indiscutível. Basta ver o ritmo de crescimento da *Internet* nos últimos anos.

No início de 1994 uma estimativa dizia que a *Internet* era composta por cerca de 40 mil redes. Em 1996 já eram 100 mil redes e a taxa de crescimento parecia duplicar o número de redes a cada ano (ou ainda em menos tempo). Em 1998, o crescimento da *Internet* não dependia mais das Universidades ou dos Centros de Pesquisas (ROBERTO, 1999).

2.1 – Modelo TCP/IP

A *Internet* é a mais bem sucedida aplicação prática do conceito de *Internet working*, que consiste em conectividade de redes de tecnologias distintas. Essa conectividade foi conseguida pelo uso do conjunto de protocolos conhecido como *TCP/IP Protocol Suite*, ou simplesmente TCP/IP. O TCP/IP nome derivado de seus protocolos principais, *Transmission Control Protocol / Internet Protocol* executa essa conectividade em nível de rede, o que permite a comunicação entre aplicações em computadores de redes distintas sem a necessidade de conhecimento da topologia envolvida nesse processo (COMER, 1998)

Uma outra característica importante do TCP/IP é a flexibilidade de adaptação às tecnologias de redes existentes e futuras, que é possível porque o TCP/IP foi concebido de forma independente das tecnologias de redes. Os equipamentos que executam a conexão entre redes na *Internet* baseiam-se no protocolo IP para o encaminhamento (ou roteamento) de informações através das redes envolvidas, e por isso são denominados como Roteadores IP (COMER, 1998).

O número de serviços que podem estar disponíveis na *Internet* é ilimitado, dada a transparência que o protocolo TCP/IP dá a essa rede, facilitando assim o desenvolvimento contínuo de novas aplicações e serviços.

A arquitetura do protocolo TCP/IP é composta por quatro camadas, cujas funções principais são:

Camada de Aplicação - Os protocolos de mais alto nível incluem os detalhes das camadas de Aplicação, Apresentação e de Sessão do modelo OSI. Esta camada fornece serviços e utilitários que permitem que os aplicativos

acessem os recursos de rede. O TCP/IP combina todas as camadas OSI 5, 6 e 7 na camada de aplicação. Qualquer usuário pode criar suas aplicações, pois TCP/IP é uma arquitetura aberta. HTTP, SMTP, POP3, SNMP, DHCP e DNS são alguns exemplos de aplicações, além de outras.

Camada de Transporte - Fornece serviços de entrega de dados ponto a ponto. Essa camada é responsável por garantir a integridade das mensagens enviadas pela camada de aplicação. Segundo Comer (1998), para que se forneça um transporte confiável, e seja assegurado que os dados cheguem sem erros e em sequência, o protocolo de transporte faz com que sejam enviadas informações do lado receptor e retransmissão de pacotes perdidos, por parte do transmissor. Similar à manipulação de *frames* pela camada de rede, esta camada agrega pequenas mensagens em um único pacote, e quebra mensagens grandes em vários pacotes, visando otimizar a performance na rede. 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 *et al.* 1995).

Camada de Rede - Nessa camada reside o protocolo IP. Essa camada é responsável pelas tarefas de endereçamento de mensagens, conversão de endereços e nomes lógicos em físicos, determinação do caminho entre o computador origem e destino baseados nas condições da rede, prioridade do serviço, administração de problemas de tráfegos tais como roteamento e controle de número de pacotes na rede. Essa camada agrega *frames* pequenos, e reestrutura *frames* grandes em *frames* menores, antes de enviá-los à rede. No lado destino, estes *frames* são restaurados para sua estrutura original.

Camada de Interface de 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 *et al.* 1995). Essa camada 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 OSI 1 e 2.

2.1.2 - Funcionalidade do IPv4

A camada IP tem que reconstruir o *frame* a partir dos fragmentos que recebe, assegurar-se de que não falta nenhum e verificar se eles estão na ordem correta. A camada IP também tem que tratar uma variedade de formatos de endereçamento que são utilizados entre sistemas TCP/IP.

Segundo Soares *et al.*(1995), a função do protocolo IP é a transmissão dos pacotes de dados entre dois *hosts*. Estes dados são recebidos das camadas superiores, como o TCP, e podem trafegar por diversas redes antes de atingir o seu destino final. O protocolo IP provê ainda um mecanismo de controle de fragmentação dos pacotes de dados, quando são transmitidos para hosts onde a janela de recepção é menor que o tamanho do pacote de dados.

No envio de um pacote de dados via IP, ocorre primeiro um processo de multiplexação, onde os dados provenientes da camada de transporte TCP são concatenados através do protocolo IP, que utiliza um cabeçalho próprio e os

envia para camadas de enlace e posteriormente para a camada física. Quando este pacote de dados chega ao seu destino, ocorre o processo de demultiplexação, onde o protocolo IP recebe os pacotes de dados provenientes das camadas física e enlace, e através da leitura do cabeçalho IP, identifica se o pacote de dados deve ser enviado para a camada de transporte TCP.

2.1.3 - Endereçamento IPv4

O endereço IP é composto por um campo de 32 *bits*, numerados de 0 a 31. No campo de endereço IP, estão contidas duas importantes informações: identificação do *host* e identificação da rede à qual o *host* está conectado. Na implantação inicial do protocolo IP o campo de endereço era de 32 bits, sendo 8 bits designado para identificação da rede, e 24 bits para identificação dos *hosts*. Isso foi modificado nas versões mais recentes, como será visto a seguir.

Segundo Almeida (1999), cada máquina de uma rede TCP/IP possui um endereço IP, tal como 200.252.155.9. O endereço IP, às vezes chamado de *dotted quad*, é composto por quatro números separados por ponto, cada qual na faixa de 0 a 255. Quando uma LAN inteira se liga à *Internet*, é comum atribuir endereços IP às máquinas da LAN que são facilmente distinguidas dos endereços IP do restante da *Internet*. Os grupos de endereços relacionados são chamados de endereços Classe A, B ou C conforme mostra a figura 2.1.

Classe	1º Octeto	Nº Max. De Redes	Nº Max. De Hosts	Formato	Exemplo
A	1-126	125	16.777.214	R.H.H.H	120.2.1.0
B	128-191	16.382	65.534	R.R.H.H	132.23.1.0
C	192-223	2.097.150	254	R.R.R.H	220.0.1.1

Figura 2.1 - Classes de endereços IP

Um endereço Classe C é aquele que pode possuir até 254 endereços IP, cada um deles tendo os mesmos valores nos três primeiros componentes. O último componente é diferente em cada máquina da LAN. Uma rede de Classe B pode comportar até cerca de 60.000 endereços IP, cada um com os mesmos valores nos dois primeiros componentes. Os dois últimos componentes podem variar. Geralmente, os proprietários de endereços Classe B têm muito menos que 60.000 máquinas em suas redes internas, mas esse número é maior que 254. Já a Classe A é capaz de suportar cerca de 15 milhões de endereços IP, todos tendo como primeiro componente o mesmo valor e três componentes diferentes no final.

Somente uma quantidade limitada dessas classes está disponível. Por exemplo, os primeiros componentes com valores entre 1 e 126 são reservados para os endereços Classe A. Na prática menos de 50 endereços Classe A foram atribuídos, basicamente para os criadores da Internet, como as forças armadas e empresas de telecomunicação americanas. Os primeiros componentes entre 128

e 191 são para os endereços Classe B, e os que estão entre 192 e 223 são para os endereços Classe C. Os primeiros componentes a partir de 224 são endereços reservados para classes D e E (ALMEIDA, 1999).

Para garantir na Internet que cada *host* tenha um endereço IP que seja único, existe um órgão regulamentador responsável por designar endereços IP para todas as empresas, organizações públicas e educacionais, que irão se conectar à *Internet*. Esse órgão cede um endereço IP contendo a porção de rede, cabendo ao administrador da rede fazer o projeto de endereçamento do número de *hosts* desejados, utilizando-se do endereço fornecido.

Muitas empresas constroem sua *Intranet*, definindo internamente os endereços IP para seus *hosts*. É importante lembrar que no momento em que se deseja conectar *Intranet* à *Internet*, será necessário solicitar ao órgão competente um endereço IP que possa ser utilizado na Internet, o que certamente implicará em alterar todos os endereços IP previamente definidos.

Se for planejado ligar sua *Intranet* à *Internet*, antes de definir inteiramente o projeto de endereçamento IP a ser utilizado, deve ser solicitado um endereço IP aos provedores de acesso (ALMEIDA, 1999).

2.2 – Comentários Finais

De acordo com alerta feito em julho de 2001 pela *Internet Assigned Numbers Authority* (IANA), instituição responsável pelo sistema geral de registro de domínios no mundo, poderia haver um “apagão” de endereços IP no mundo. O IANA realizou um estudo que estima que já foram consumidos cerca de 68% dos endereços disponíveis no mundo. Com a explosão da Internet e com o surgimento constante de mais e mais serviços e aplicações, os atuais endereços

IP (IPv4) estão se tornando um recurso escasso. Já se estimava que, em um ano atrás, eles estariam esgotados (SILVA, 1998).

De acordo com Silva (1998), os protocolos TCP/IP mostram um importante desafio arquitetônico com a contínua e fenomenal expansão da *Internet*. Com o crescimento anual de 100% no número de redes ligadas à Internet coloca-se em xeque o sistema de roteamento. As classes B estão paulatinamente sendo esgotadas e o uso das técnicas de CIDR¹ usa máscaras de comprimento variável para alocar endereços IP em subredes de acordo com as necessidades individuais representam uma solução paliativa a este problema. Uma nova versão do IP faz-se necessária para suportar um endereçamento muito maior e prover suporte ao problema de escalabilidade. Ao mesmo tempo, novas e bastante ambiciosas aplicações já sugerem que a *Internet* precisa suportar pacotes de voz e vídeo em proporções cada vez maiores.

Segurança é também um dos fatores mais importantes, especialmente com a expansão de redes aplicadas ao mundo dos negócios. Procurar uma maneira uniforme de suportar a *Internet* e ainda lidar com a variedade de tecnologias pelo mundo, algumas das quais sujeitas a restrições de exportação, é um desafio de enormes proporções.

1 CIDR – *Classe Inter-Domain Routing*

3- IPv6 - O PROTOCOLO DA NOVA GERAÇÃO

3.1 – Necessidades de uma nova versão do IP

Segundo a *Internet Corporation for Assigned Names and Numbers* (ICANN², 2005), com a inovação e o crescimento contínuo da *Internet*, trazem-se novos desafios para manter sua estabilidade. Na concepção do protocolo IP, a aproximadamente duas décadas, não foram previstas necessidades emergentes da sociedade e da sua relação com os sistemas de comunicação atuais, em particular com a *Internet*. Devido a tal, há características que necessitam de alterações.

Essas características necessárias são espaço de endereçamento, autoconfiguração e mobilidade, segurança na camada da rede, qualidade de serviço, suporte de aplicações no tempo real.

3.2 – Histórico da Evolução do IP

O projeto IPng - *IP the Next Generation* - representa o resultado da evolução de diferentes propostas IETF, bem como o esforço conjunto de vários grupos de trabalho. Conforme a RFC 1752 (BRANDER & MANKIN, 1995) o projeto IPng sofreu a seguinte evolução:

a) 1990 - No encontro IETF de Vancouver conforme *Frank Solensky, Phill Gross* e *Sue Hares* afirmaram que à taxa de atribuição do espaço de endereçamento IPv4, as classes do tipo B estariam esgotadas possivelmente por volta de Março de 1994.

² ICANN - Corporação para Atribuição de Nomes e Números na Internet

b) 1991 - A *Internet Engineering Task Force (IETF)* forma o grupo de trabalho *Routing and Addressing (ROAD)* no encontro de Santa Fé com o objetivo de encontrar uma solução para a exaustão do espaço de endereçamento IPv4.

c) 1992 - A *Internet Association Board (IAB)* apresenta o documento *IP version 7* paralelamente aos esforços do grupo de trabalho ROAD, em que recomenda à IETF a preparação de um plano detalhado para o sucessor do protocolo IP. A IETF rejeita esta sugestão e apresenta pedido de propostas recomendadas pelo grupo ROAD. Como resposta a este pedido surgiram algumas propostas:

- *IP Encaps*
- *Nimrod*
- *Simple CLNP*

d) 1992 (finais) - Surgem mais três propostas:

- *The P Internet Protocol (PIP)*
- *The Simple Internet Protocol (SIP)*
- *TP/IX*

e) 1993 – *Phil Gross – Internet Engineering Steering Group (IESG)* apresenta um memorando intitulado "*A Direction for IPng*" onde anuncia a criação de uma área temporária para o IPng. CLNP e IP Encaps evoluem, dando origem respectivamente a *TCP and UDP with Bigger Addresses, TUBA* e *IPAE*.

f) 1993 (finais) - SIP evoluiu, abrangendo características do IPAE (*IP Address Encapsulation*). O IPAE foi adotado como estratégia de transição para o SIP (*Simple IP*), proposto por *Steve Deering* em Novembro de 1992. O SIP aumentava o endereço IP para 64 bits, tornava a fragmentação de pacotes opcional e eliminava vários aspectos obsoletos do IP.

g) Em junho de 1994, a comissão do IPng revisou todas as propostas, SIP e PIP, deram origem à proposta *The Simple Internet Protocol Plus (SIPP)* e publicou sua recomendação sugerindo o SIPP como a base para o novo protocolo IP, mas com mudanças em algumas características chaves da especificação original. Particularmente, o novo protocolo teria 128 *bits* e se chamaria IPv6. Ocorreu então a aprovação do documento *The Recommendation for the IP Next Generation Protocol* como norma oficial de desenvolvimento do IPng (Rosa, 1999).

A primeira conexão foi estabelecida em março de 1998 com a *Cisco System*, nos EUA. Em outubro do mesmo ano foi estabelecida a conexão com a *Nippon Telephone and Telegraph*, no Japão.

3.3 – Instituições Ligadas ao IPv6 - IETF, NGTRANS e o 6BONE

3.3.1 - IETF

A IETF é uma sociedade aberta da qual participam pesquisadores, projetistas, operadores de telecomunicações e de provedores de serviços *Internet*, bem como fabricantes de equipamentos. Todos são voluntários e estão, direta ou indiretamente, relacionados com a arquitetura da *Internet*, com a especificação e o desenvolvimento de protocolos de comunicação e aplicações, ou com a operação, a segurança e o gerenciamento desta rede.

3.3.2 – NGTrans

O NGTrans³ é um grupo de trabalho da IETF que visa estudar e definir os mecanismos e procedimentos para suportar a transição da *Internet* do IPv4 para o IPv6. Sua estratégia se baseia em:

- Produzir um documento detalhando a infra-estrutura, como será e o que será necessário para a transição;
- Definir e especificar os mecanismos obrigatórios e opcionais a serem implementados pelos fabricantes nos *hosts*, roteadores e outros equipamentos de rede, a fim de suportar o período de transição;
- Articular um plano operacional concreto a ser executado pelos ISPs (*Internet Service Providers*) quando da transição entre o IPv4 e o IPv6.

3.3.3 - 6Bone

O 6Bone é um projeto colaborativo informal, empreendido pelo NGtrans (grupo de trabalho do IETF) que consiste no desenvolvimento do *backbone IPv6* experimental para pôr à prova as funcionalidades deste novo protocolo IPng e desenvolver os mecanismos para a transição do uso do IPv4 para o IPv6. O Brasil está participando destas pesquisas através do projeto Br 6Bone, empreendido pelo LCT⁴ - Laboratório de Configuração e Testes da RNP.

O 6Bone é um teste de campo para auxiliar na evolução, no desenvolvimento e no aperfeiçoamento do protocolo IPng. Atualmente integra

3 NGTrans – *Next Generation Transition*

4 LCT – Laboratório de Configuração e Testes

aproximadamente 39 países, apresentados na Tabela 3.1, dentre os quais também o Brasil desde janeiro do 2002.

AT	Áustria	AU	Austrália
BE	Bélgium	BG	Bulgaria
BR	Brazil	CA	Canada
CH	Switzerland	CM	Cameroon
CN	China	CZ	Czech Republic
DE	Germany	DK	Denmark
ES	Spain	FI	Finland
FR	France	GB	Great Britain
GR	Greece	HK	Hong Kong
HU	Hungary	IE	Ireland
IT	Italy	JP	Japan
KR	Korea	KZ	Kazakhstan
LT	Lithuania	MX	México
NL	Netherlands	NO	Norway
PL	Poland	PT	Portugal
RO	România	RU	Russian Federation
SE	Sweden	SG	Singapore
SI	Slovenia		

Tabela 3.1 – Tabela dos países integrados ao IPv6

Operacional desde junho de 1996, este *backbone* é implementado através de uma rede virtual sobre a rede física IPv4 da atual *Internet*. A rede virtual é composta de redes locais IPv6 ligadas entre si por túneis ponto-a-ponto IPv6 sobre IPv4. Os túneis são realizados por roteadores com pilha dupla (IPv6 e IPv4) com suporte para roteamento estático e dinâmico (RIPng e BGP4+), e as redes locais IPv6 são compostas por estações com sistemas operacionais com suporte a IPv6 ou com pilha dupla (IPv4 e v6).

No entanto, a rede 6bone é independente da IETF, sendo um projeto que reúne voluntariamente diversas instituições do mundo inteiro. O projeto 6bone tem sofrido a seguinte evolução:

- a) 1995 - Concepção do projeto 6bone.
- b) 1996 (Março) - Formalização do projeto 6bone em Março, no encontro IETF em Los Angeles.
- c) 1996 (Junho) - Início da rede 6bone, através da criação de dois túneis: um entre a Universidade de Lisboa (UL/PT), o *Navy Research Laboratory* (NRL/US) e a CISCO (CISCO/US) e o outro entre a UNI-C (UNIC/DK), o grupo G6 do instituto de pesquisa IMAG (G6/FR) e o grupo WIDE, do Japão (WIDE/JP).
- d) 1996 (Dezembro) - Formação do grupo de trabalho 6bone (atualmente NGtrans).
- e) 1997 (Agosto) - No encontro do grupo de trabalho ngtrans-6bone que teve lugar em Munique, referiu-se que existiam mais de 150 sites IPv6 em 28 países participantes na 6bone. O protocolo de encaminhamento do *backbone* da 6bone passou a ser o BGP4+. Foi criado o domínio 6bone.net, através do qual se pode aceder às páginas e base de dados 6bone.
- f) 1997 (Dezembro) - No encontro de Washington, a 6bone apresentava já 206 sites em 30 países participantes.
- g) 1998 (Março) - No encontro de Los Angeles, a 6bone apresentava 240 sites IPv6 em 32 países. Neste encontro, foi acordado que os pTLA usariam entre si o protocolo de encaminhamento BGP4+ (HINDEN, 1994).

3.4 - Principais Objetivos de Criação do IPv6

Os principais objetivos da criação do IPv6 foram colmatar as principais falhas do IPv4 e introduzir novas funcionalidades:

a) Aumento do número de endereços na Internet: Passamos assim a ter muitas vezes mais endereços IPs.

b) Melhoramento do *Routing*: carregando menos os *routers* é possível ter uma melhor performance na Internet, especialmente quando são *routers* principais

c) Possibilitar autoconfiguração: de modo a que seja retirado trabalho para o utilizador final, que deseja conectar a sua máquina em qualquer ponto da rede e obter rede e end. IP.

d) Operação em redes de alto débito: com cada vez mais rápidas redes, é necessário adaptar o protocolo para retirar a maior performance delas. O IPv4 nunca foi pensado para redes de alto débito.

e) Prover melhor segurança, criar mecanismos de proteção no próprio protocolo. Confidencialidade e privacidade: algo que falta no IPv4 e que é implementado a nível aplicacional é justamente a confidencialidade. Torna-se necessário codificar os dados a nível de IP.

f) Capacidade de QoS: dado que existe uma variedade de tipos de tráfego sobre IP, é necessário saber distinguí-los e dar prioridades diferentes para cada caso, de modo a garantir uma qualidade de serviço.

g) Suportar bilhões de *hosts*, mesmo que os endereços fossem alocados de forma ineficiente.

h) Reduzir o tamanho da tabela de rotas.

i) Simplificar o protocolo para permitir que os roteadores processem os pacotes mais rapidamente.

j) Dar maior atenção ao tipo de serviço trafegado, particularmente para dados de Tempo Real..

k) Permitir o escopo do alcance de um pacote (*Multicasting*).

l) Fazer o possível para que um *host* tenha mobilidade sem mudar seu endereço.

m) Permitir que o protocolo evolua no futuro.

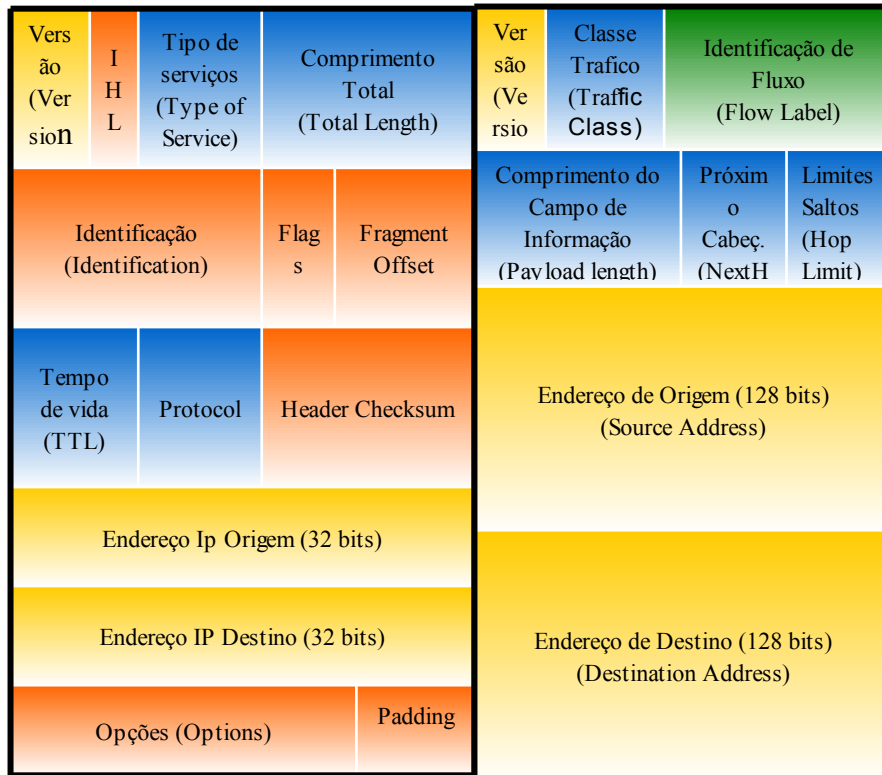
n) Permitir que protocolos antigos e novos coexistam por anos.

3.5 - Os Caminhos da Padronização

O processo de padronização da IETF exige que para se tornar um *Proposed Standard*, uma idéia precisa ser completamente explicada em uma RFC (*Request For Comments*) e ter interesse da comunidade científica. Para avançar para o estágio de *Draft Standard*, é preciso existir uma implementação que tenha sido testada por pelo menos dois *sites* durante 4 meses. Em dezembro de 1998, uma grande parcela dos resultados do grupo de trabalho IPng já tinham sido discutido e implementado por diversos *sites* em todo o mundo, portanto o IPv6 encontrava-se em estágio “avançado” de padronização. Grandes fabricantes de roteadores como a Cisco e a IBM já tem produtos que suportam a nova versão do IP. Mesmo assim, os autores da especificação do IPv6, *Deering* e *Hinden*, solicitam que o documento ainda seja referido como trabalho em progresso. A maior parte do trabalho do IPng encontra-se na categoria Padrão Proposto (*Proposed Standard*) e apenas um relatório técnico relacionado ao *path MTU discovery* ainda é um Padrão Rascunho (*Draft Standard*) (NA-CP, 1999).

4- CARACTERÍSTICAS DO IPv6

Conforme Sofia (1998) o IPv6 aumentou o endereçamento de IP de 32 bits para 128 bits. O IPv6 mantém as principais características que fizeram do IPv4 um sucesso mundial. Assim como o IPv4, é um protocolo sem conexão - cada datagrama contém um endereço de destino e é roteado de forma independente. O IPv6 também possui um número máximo de roteadores por onde pode passar (*Hop Limit*). Com objetivo de simplificar a principal função do IP, rotear pacotes, sete campos no IPv6 foram suprimidos: *IHL*, *identification*, *flags*, *fragment offset*, *header checksum*, *options* e *Padding*. Quatro foram renomeados e em alguns casos, modificados: *Total length*, *protocol type*, *time to live (TTL)*, *type of service*. Três foram mantidos: *Version*, *Source Address*, *Destination Address*. E um criado: *Flow Label*. Na Figura 4.1 apresentada a seguir são mostradas as diferenças entre os cabeçalhos IPv4 e IPv6.



- Campos mantidos IPv4 to IPv6
- Campos não mantidos IPv6
- Nomes e posições trocados no IPv6
- Novos campos no IPv6

Figura 4.1 – Comparação e exposição do Cabeçalho IPv6 x IPv4

As características que definem o protocolo IPv6 são:

- a) expansão da capacidade de endereçamento e encaminhamento;
- b) capacidade de qualidade de serviço;
- c) capacidade de providenciar autenticação e privacidade;
- d) simplificação dos cabeçalhos.

4.1 - Datagrama IPv6

Desenhado para ser o sucessor do IPv4 o datagrama do IPv6, segundo Comer (1995), é composto por um cabeçalho (*header*) com menos campos, alguns cabeçalhos estendidos opcionais e o campo para dados. O datagrama mínimo tem o cabeçalho base seguido dos dados. O formato do campo é:

- a) Versão – 4-*bits* Protocolo de Internet versão 6.
- b) Classe do Tráfico (*Traffic Class*) – 4-*bits* Campo de Classe do Tráfico (Serviços).
- c) Identificação de Fluxo (*Flow Label*) – 24-*bits* Identificação e diferenciação de pacotes do mesmo fluxo na camada de rede.
- d) Comprimento do campo de Informação (*Payload Length*)– 16-*bits* – Tamanho total dos dados no pacote.
- e) Próximo cabeçalho (*Next Header*)– 8-*bit* Identifica o próximo cabeçalho seguido do cabeçalho IPv6. Poderá ser um pacote no nível da camada de transporte (TCP/UDP) ou cabeçalhos denominados de (*extension headers*).
- f) Limites de Saltos (*Hop Limit*) – 8-*bit* número máximo de saltos nos

equipamentos, decremento de 1 a 1 cada nó que segue o pacote.

- g) Endereço de Origem (*Source Address*) – 128-bit endereço de origem do pacote.
- h) Endereço de Destino (*Destination Address*) – 128-bit endereço da intenção do destino do pacote (possibilidade de não ser o destino final, depende da existência do cabeçalho de roteamento).

4.1.1 - Cabeçalhos Simplificados

O cabeçalho do IPv6 conta com menos campos e não tem mais o equivalente ao “checksum” do IPv4. Como cada roteador precisa decrementar o TTL (*Time To Live*), o *checksum* do cabeçalho IPv4 precisa ser recalculado. Nesse ponto encontra-se um dos motivos da alta taxa de utilização da CPU dos roteadores. O objetivo de não utilizar o *checksum* com o IPv6 é processar mais rapidamente os datagramas no roteador. O cabeçalho é formado pelo trecho inicial de 64 *bits* mais dois campos com endereço origem e destino, com 128 *bits* cada um. Assim o tamanho total (constante) é 320 *bits* ou 40 *bytes* (SILVA 1998). O cabeçalho IPv6 apresenta assim uma estrutura mais simplificada, conforme Figura 4.1 mostrada anteriormente.

4.1.2 - Cabeçalhos Estendidos IPv6

Segundo Silva (1998), o modo de tratar as opções do IPv6 é a apresentação um esquema de módulos: a informação adicional é transmitida através dos cabeçalhos de extensão, Figura 4.2.

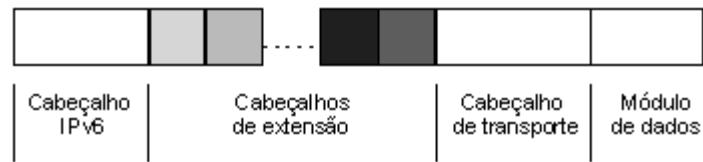


Figura 4.2 - Datagrama IPv6 com cabeçalhos de extensão.

Este esquema fornece ao IPv6 flexibilidade para transportar informação relevante para encaminhamento e aplicações, bem como fornecer mecanismos de segurança, fragmentação, qualidade de serviço e gestão de rede, com escalabilidade ilimitada. Na medida em que estes módulos são opcionais, este esquema ajuda ainda a reduzir o custo de processamento de pacotes IPv6.

Os cabeçalhos de extensão são colocados entre o cabeçalho IPv6 e o cabeçalho do protocolo de transporte, estando ligados entre si pelo campo Próximo Cabeçalho (*Next Header*), formando uma cadeia, conforme mostrado na Figura 4.3.

Versão (Version)	Classe Tráfego (Traffic Class)	Identificação de Fluxo (Flow Label)	
Comprimento do Campo de Informação (Payload length)		Próximo Cabeç. (NextHead)	Limites Saltos (Hop Limit)
Endereço de Origem (128 bits) (Source Address)			
Endereço de Destino (128 bits) (Destination Address)			
Próximo Cabeçalho (Next Header)		Informação do Cabeçalho de Extensão (Extension Header Information)	
...Exemplo (TCP ou UDP)			

Figura 4.3 – Formação do cabeçalho de extensão IPv6.

Exceto quando os cabeçalhos de extensão não são processados ou examinados por nenhum nodo no caminho da entrega do pacote até que o pacote atinja o nodo do endereço de destino do cabeçalho IPv6. Atualmente, encontram-se já definidos os seguintes cabeçalhos de extensão:

- a) *Hop-by-Hop*. Usado para transportar informação opcional que tem de ser examinada por cada nodo ao longo do caminho do pacote. Incluindo a origem e o destino do nodo. Deve ser seguido pelo cabeçalho IPv6 e sua presença é indicada pelo valor 0 no Próximo Cabeçalho (*Next Header*), conforme Figura 4.4.

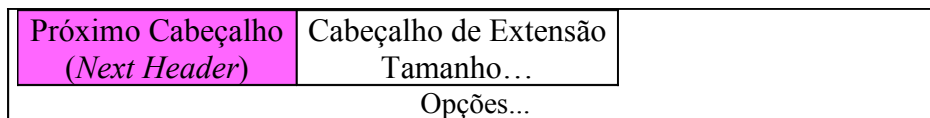


Figura 4.4 – *Hop-by-Hop*

- b) Opções de Destino IPv6 (*Destination Options Header*). Usado para transportar informação opcional a ser analisada apenas no destino do pacote.
- c) Encaminhamento (*Routing Header*). Usado por uma fonte IPv6 para listar um ou mais nodos intermediários que devem ser visitados até o pacote chegar ao destino. O IPv6 mantém a habilidade da origem especificar a rota, porém para isso há um cabeçalho extendido opcional, semelhante às opções *Loose Source* e *Record Route* do IPv4. Este cabeçalho contém entre outras informações uma lista de endereços de roteadores intermediários por onde o datagrama precisa passar, conforme Figura 4.5.

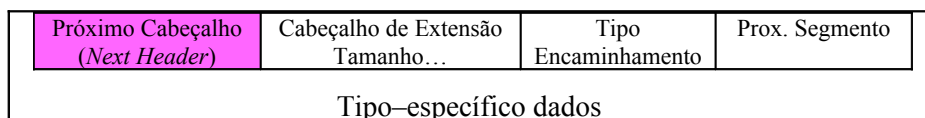
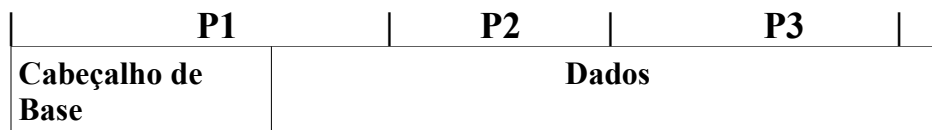


Figura 4.5 – Encaminhamento

- d) Fragmentação (*Fragmentation Header*). Usado para enviar módulos de dados maiores do que a *Maximum Transmit Unit* (MTU) de um caminho. O IPv6, assim como o IPv4, deixa a cargo do destino final a ação de reconstruir o datagrama caso tenha sido fragmentado. Porém, no IPv6 a fragmentação é fim-a-fim, isto é, antes de enviar um datagrama a origem utiliza-se da técnica *MTU Discovery* para descobrir qual é o menor MTU

ao longo do caminho a ser percorrido. Antes de enviar o datagrama, a própria origem sabe a quantidade de fragmentos necessária para esta transmissão. A fragmentação fim-a-fim permite que cada roteador manipule mais datagramas por unidade de tempo. Um roteador tradicional fragmenta boa parte dos datagramas que recebe, portando a carga de sua CPU frequentemente alcança valores próximos do 100%. Porém a fragmentação fim-a-fim tem uma importante consequência: muda fundamentalmente uma das prerrogativas da Internet. Hoje em dia, a flexibilidade do IPv4 permite que rotas sejam mudadas a qualquer momento. Se roteadores intermediários falharem, o tráfego pode seguir novo caminho sem que os extremos, origem e destino, tomem conhecimento dessa mudança e, teoricamente, sem interromper o serviço oferecido. No IPv6 a rota não pode ser mudada com tanta facilidade porque a mudança pode também resultar na mudança do caminho do *Path MTU*. Se o *Path MTU* ao longo da nova rota é menor que o *Path MTU* ao longo da rota original, ou um roteador intermediário precisa fragmentar ou a fonte de datagramas precisa ser informada. Para resolver o problema de mudança que possa afetar o *Path MTU*, o IPv6 permite que roteadores intermediários utilizem-se de um recurso chamado tunelamento de IPv6 através de IPv6. Quando precisam fragmentar os roteadores IPv6 intermediários criam um datagrama inteiramente novo que encapsula o datagrama original como dado deste, da maneira que mostra a Figura 4.6 (SILVA, 1998).



(a)



(b)



(c)



(d)

Figura 4.6 - Fragmentação e encapsulamento do datagrama

- e) Autenticação (*Authentication Header*). Usado para providenciar autenticação e garantia de integridade aos pacotes IPv6.
- f) Cifra (*IPv6 Encryption Header*). Usado para providenciar confidencialidade e integridade através da cifra de dados.
- g) Opções de Destino IPv6 (*End-to-End Option Header*). Usado para o transporte de informação opcional que apenas necessita de ser examinada pelo nodo destino de um pacote. Este cabeçalho pode surgir duas vezes no mesmo datagrama.
- h) *Encapsulating Security Payload*. Um dos principais objetivos

da próxima versão do IP é oferecer um mecanismo que garanta a privacidade na comunicação sem ter que depender de implementações da camada de aplicação. Ambos cabeçalhos visam prover um tipo de segurança a quem enviou a mensagem. O primeiro, *Authentication*, tem por objetivo garantir a autenticação e integridade (sem confidencialidade). Foi proposto o uso de chaves MD5 (*Message Digest 5*, um tipo de *checksum*) para garantir a interoperabilidade. A inclusão deste mecanismo permite eliminar ataques do tipo *IP spoof* que hoje em dia precisam ser configurados nos *firewalls*. O *IP spoof* consiste em forjar os pacotes de origem: caso a máquina B confie na máquina A, qualquer pacote que tiver o endereço IP da máquina A será “confiável”. Um dos motivos da escolha do MD5 é que o mesmo pode ser exportado pelos EUA e por outros países que possuem as mesmas restrições de exportação de algoritmos de criptografia. O segundo cabeçalho estendido opcional relacionado a segurança é o *Encapsulating Security Payload*. Este mecanismo provê integridade e confidencialidade para os datagramas. Ele é mais simples que alguns protocolos de segurança similares e possui flexibilidade e independência. O algoritmo padrão é o DES-CBC (*Data Encryption Standard*). Este cabeçalho estendido opcional contém informações sobre a associação que vai ser estabelecida. São definidos dois modos:

- Na Criptografia fim-a-fim a carga e os cabeçalhos estendidos são criptografados.
- No modo de Tunelamento, requerido para criptografia

entre sistemas e *gateways* seguros ou entre dois *gateways* seguros, o datagrama IP completo é criptografado e encapsulado por um novo datagrama não-criptografado. Esta é a alternativa do IPv6 quando as VPN (*Virtual Private Networks*). Um exemplo da comunicação de um *gateway* seguro e o sistema final é o caso do acesso discado onde utiliza-se uma linha telefônica comum ou comunicação sem fio. São baseadas em um backbone público não seguro (Silva,1998).

4.2 - Maior Endereçamento

O IPv6 utiliza 128 *bits* para endereçamento de nós, enquanto o atual IP (IPv4) utiliza 32 *bits*. Com isso é possível prover uma grande quantidade de endereços (2^{128}). Na prática, a criação de hierarquias no endereço diminuem a eficiência da utilização do espaço de endereçamento disponível. Assim mesmo, estudos estimam que os 128 *bits* são capazes de acomodar na mais pessimista estimativa 1564 endereços/m² da superfície da Terra (SILVA,1998).

4.2.1 - Divisão dos Endereços

Segundo a RNP⁵ (1998) são três tipos endereçamentos que envolvem o IPv6: *unicast*, *anycast* e *multicast* conforme mostra a Tabela 4.1, cerca de 85% dos endereços estão reservados para uso futuro.

5 RNP - Rede Nacional de Ensino e Pesquisa

Tabela 4.1- Divisão dos endereços Ipv6
(Construída a partir de dados disponíveis na RFC 2373)

Alocação	Prefixo (Binário)	Fração do Espaço de endereçamento
Reservado Compatibilidade IPv4	0000 0000	1/256
Não atribuido	0000 0001	1/256
Reservado para Alocação NSAP	0000 001	1/128
Reservado para Alocação IPX	0000 010	1/128
Não atribuido	0000 011	1/128
Não atribuido	0000 1	1/32
Não atribuido	0001	1/16
Não atribuido	001	1/8
Endereços <i>Unicast</i> Baseado em Provedor	010	1/8
Não atribuido	011	1/8
Endereços <i>Unicast</i> Reservados para Bases Geográficas	100	1/8
Não atribuido	101	1/8
Não atribuido	110	1/8
Não atribuido	1110	1/16
Não atribuido	1111 0	1/32
Não atribuido	1111 10	1/64
Não atribuido	1111 110	1/128
Não atribuido	1111 1110 0	1/512
Endereço Usados <i>Link</i> Local	1111 1110 10	1/1024
Endereço Usados <i>Site</i> Local	1111 1110 11	1/1024
Endereços <i>Multicast</i>	1111 1111	1/256

Para melhor exemplificar o projeto de endereçamento do IPv6, pode-se considerar a Tabela 4.7 a linha de endereços *unicast baseada em provedor* cujo prefixo é 010 e que contém 12,5% (ou 1/8) dos endereços. Um endereço deste tipo deve ser ainda dividido nos campos provedor *ID*, subscritor *ID*, sub-rede *ID* e nó *ID*. Recomenda-se que este último tenha pelo menos 48 *bits* para que possa armazenar o endereço MAC IEEE802.3 (*Ethernet*). O prefixo de identificação do provedor é portanto os bits 010 seguidos do provedor *ID*. O prefixo de identificação do subscritor é formado pelo prefixo de identificação do provedor seguido do subscritor *ID*. Finalmente, o prefixo de identificação da sub-rede é formado pelo prefixo de identificação do subscritor mais a informação referente à sub-rede.

4.2.2 - Transição e Codificação de Endereços IPv4

Da Figura 4.7 observa-se que o prefixo 00000000 está reservado para codificar endereços IPv4 de 32 bits. De acordo a *RFC 2373* (HINDEN & DEERING, 1998), teoricamente, qualquer endereço cujo prefixo seja 80 zeros seguido de 16 *bits* 1 ou 16 *bits* 0 contém um endereço IPv4 nos últimos 32 *bits* conforme mostram as Figuras 4.7 e 4.8.

80 Bits	16 Bits	32 Bits
0000.....0000	0000	Endereçamento IPv4

Figura 4.7 - Endereçamento IPv6 embutido com endereçamento IPv4

80 Bits	16 Bits	32 Bits
0000.....0000	FFFF	Endereçamento IPv4

Figura 4.8 - Endereçamento IPv4 mapeado com endereçamento Ipv6

A codificação será necessária por dois motivos principais:

- a) Um computador pode escolher fazer o *upgrade* do IPv4 para IPv6 antes de ter um endereço IPv6 válido atribuído a ele.
- b) Um computador IPv6 pode precisar se comunicar com um computador que roda apenas IPv4.

Porém, permitir esta codificação ainda não resolve o problema de comunicação entre as duas versões, é preciso também um tradutor. Para usar o tradutor o computador com IPv6 gera um datagrama IPv6 que contém um endereço destino IPv4. O computador IPv6 envia o datagrama para o tradutor que usa IPv4 para se comunicar com o destino. Quando o tradutor recebe a resposta do destino, ele transforma o datagrama IPv4 para IPv6 e envia-o de volta para a fonte IPv6.

Uma das maiores preocupações do IPng da IETF foi procurar uma maneira de fazer a transição entre o atual protocolo *Internet* (IPv4) para o novo protocolo (IPv6).

4.3 - Endereços *Unicast*

Conforme a RFC 2073, de Rekhter *et. al.* (1997), apenas uma interface é identificada onde um pacote destinado a um endereço *unicast* é enviado diretamente para o interface associado a esse endereço. No IPv6 foram definidos vários tipos de endereços unicast, que serão apresentados nas seções a seguir.

4.3.1 –Endereços Globais Agregados (*Aggregable Global Address*)

Segundo a RFC 2373, de Hinden & Deering (1998), *Aggregable Global Address* representa um endereço que será globalmente usado. Baseia-se no mesmo princípio do CDIR (*Classless InterDomain Routing*) possibilitando uma estreita agregação de prefixos de roteamento e contribuindo para a diminuição do número de entradas nas tabelas globais de roteamento.

Este tipo de endereços quando utilizados em *links*, são agregados hierarquicamente, começando pelos clientes, em seguida por ISP's intermédios e, eventualmente por um ISP de topo. Conforme Figura 4.9 o prefixo 2000::/3 (001) indica um endereço do tipo "*Aggregable Global*".

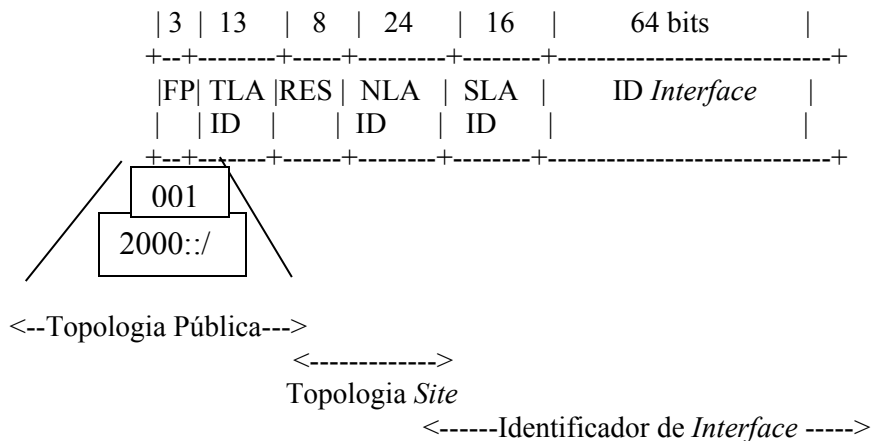


Figura 4.9 - Endereçamento *Unicast*
(Conforme RFC 2373)

Segue-se a identificação dos campos do endereçamento *Unicast*:

- a) FP - Formato do Prefixo (001)
- b) TLA ID - Identificação do Alto Nivel Agregado (*Top-Level Aggregation*)
- c) RES - Reservado para o Uso Futuro
- d) NLA ID - Identificação do Próximo Nivel Agregado (*Next-Level Aggregation*)

- e) SLA ID - Identificação do Nível Local Agregado
(Site-Level Aggregation)
- f) ID *Interface* - Identificação da *Interface*

A Identificação do alto nível agregado (TLA ID) é o mais alto nível na hierarquia de roteamento. Todas as TLA's são ligados numa zona livre e todos os roteadores existentes nessa zona devem possuir uma tabela de roteamento livre contemplando todas as identificações dessas mesmas TLA's.

Os endereços reservados para o futuro (RES) devem ser atribuídos com o valor zero. Os campos reservados para o futuro permitira o crescimento dos campos TLA`s e NLA`s apropriadamente.

Identificação do próximo nível agregado (NLA) são usados pelas organizações para assinalar a TLA ID, criando hierarquia de endereçamento e identificação local. A organização pode assinalar o mais alto nível da NLA ID de modo a criar o endereçamento hierárquico apropriado para a sua rede. Isso pode apontar os *bits* dos campos de identificação local onde se pretende servir. Como mostra a Figura 4.10.

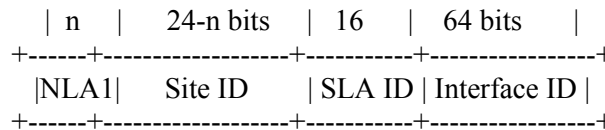


Figura 4.10 - Identificação do Próximo Nível Agregado

Cada organização com a sua TLA ID recebe 24 *bits* da NLA ID. O espaço da NLA ID permitir que cada organização de prover serviços aproximadamente ao total de organizações que o IPv4 pode suportar.

Organizações assinaladas com a TLA ID`s também podem suportar NLA ID`s no seu próprio espaço local ID. Isso permite que as organizações TLA ID prover serviços para as organizações de serviços de transito público e para as

que não provem serviços de transito público. Nesse caso as organizações recebem NLA ID e podem escolher o espaço do seu local ID para suportar outras NLA ID's . Como apresentado na Figura 4.11.

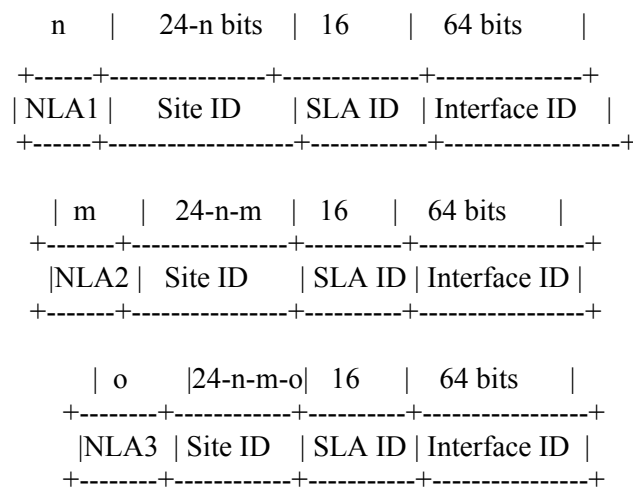


Figura 4.11 – Divisão NLA

O modelo dos *bits* apresentado no espaçamento NLA ID para uma específica TLA ID, é derivado em responsabilidade das organizações para essa TLA ID. De outro modo o modelo dos *bits* do próximo nível da NLA ID é de responsabilidade do nível antecessor da NLA ID.

O campo SLA ID (Identificação do Nível Local Agregado) é usado pela organização para criar sua própria hierarquia de endereçamento local para identificar *subnets*, isso é, inferente as *subnets* no IPv4 na expectativa de que cada organização tenha um número maior de *subnets*. O campo de 16 *bits* da SLA ID suporta 65.535 *subnets* individuais o equivalente a 256 classes de 256 endereços no IPv4..

As organizações devem escolher roteamento direto na SLA ID ou criar duas ou mais níveis hierárquicos (resulta um número de tabela de roteamento) no campo SLA ID como mostrado na figura 4.12.

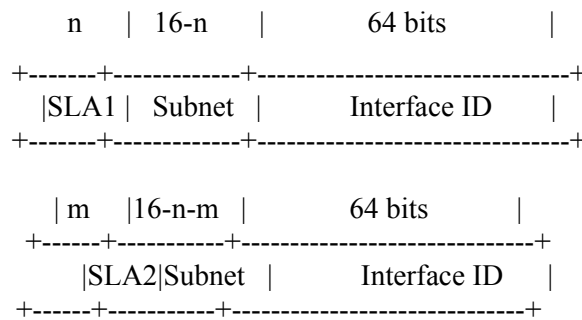


Figura 4.12 – Subnets SLA ID

É de responsabilidade de cada organização de formar a estrutura na sua SLA ID.

O número de *subnets* que pode ser composta em seu formato de endereçamento devem ser o suficiente para cobrir até mesmo as maiores organizações existentes.

As Interfaces ID são usadas para identificar interfaces nos *links*. São requeridas para serem únicas no segmento (*link*). Elas podem ser únicas até mesmo em todo formato. Em alguns casos as Interfaces ID, serão os mesmos ou serão baseadas no endereçamento de interface da camada de link. Identificadores de Interfaces são usados nos endereços globais agregados *unicast* em que são requeridos a serem 64 bits e serem construídos dentro do IEEE EUI-64

4.3.2 – Endereçamento de *link* local (*Link-Local Address*)

Segundo Hinden & Deering⁶ (1998), o endereçamento de *link* local pode ser automaticamente configurado em qualquer interface pela conjugação do seu prefixo FE80::/10 (111111010) conforme Figura 4.13, e a identificação da interface no formato EUI-64. Estes endereços são utilizados nos processos de configuração dinâmica automática e no processo de descoberta de elementos na hierarquia de roteamento (*Neighbour Discovery*). Este endereçamento permite também a comunicação entre nós pertencentes ao mesmo *link* local. Equipamentos de roteamentos não devem enviar pacotes que contenham este tipo de endereçamento como origem ou destino.

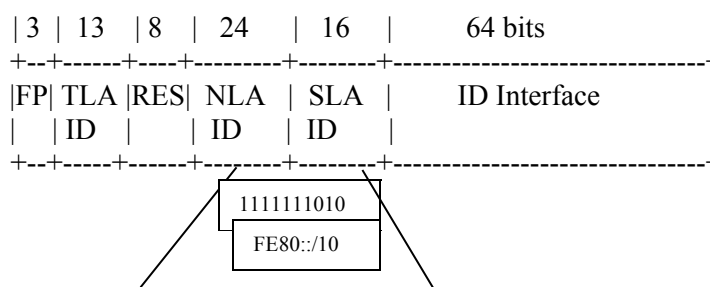


Figura 4.13 – Endereçamento de *link* local

4.3.3 – Endereçamento do *Site Local* (*Site-Local Address*)

Conforme Hinden & Deering (1998) o endereço de site local é identificado pelo prefixo FEC0::/10 (111111011) conforme Figura 4.14 e pode ser definido para uso interno numa organização através da concatenação do campo de SLA (16 *bits*) com a identificação da interface (64 *bits*). Este tipo de endereçamento pode ser considerado como privado; visto estar restrito a um

⁶ RFC 2373.

domínio sem ligação à *Internet*. Este tipo de endereçamento não pode ser anunciado externamente por roteadores.

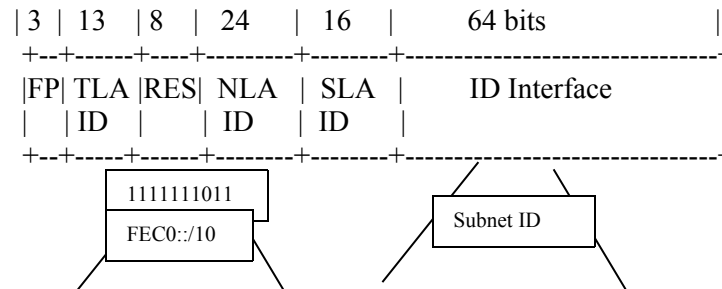


Figura 4.14 – Endereçamento do Site Local

4.3.4 – Endereçamento não Especificado (*Unspecified Address*)

O endereçamento não especificado representado por 0:0:0:0:0:0:0 ou "::", indica a ausência de um endereço e nunca deverá ser utilizado em nenhum nó. Este endereço apenas poderá ser utilizado como "*source address*" de máquinas/nós que não tenham obtido os seu próprio endereçamento.

4.3.5 – Endereçamento de Retorno (*Loopback Address*)

O endereçamento de retorno representado por 0:0:0:0:0:0:0:1 ou "::1". Apenas pode ser utilizado quando um nó envia um *datagrama* a si próprio e não pode ser associado a nenhuma interface.

4.4 – Multicast

Identicamente ao endereço *anycast* este endereço corresponde a um conjunto de computadores, por 128 bits em muitos locais em diferentes nós. A entrada e saída deste grupo pode ser alterada a qualquer momento. Quando um *datagrama multicast* é enviado, o IPv6 entrega uma cópia do *datagrama* para cada interface do grupo. O *broadcast* é “emulado” através do *multicast*. O segundo octeto que se segue ao prefixo define o tempo de vida (*lifetime*) e o contexto do endereço *multicast*. Um endereço *multicast* permanente tem um parâmetro de tempo de vida igual a "0" enquanto um endereço temporário tem o mesmo parâmetro igual a "1". O contexto para este tipo de endereço apresenta os valores de 1,2,5,8 ou "E" para identificar um nó, *link*, site, organização ou um contexto global, respectivamente conforme apresentado na Figura 4.15 a seguir.

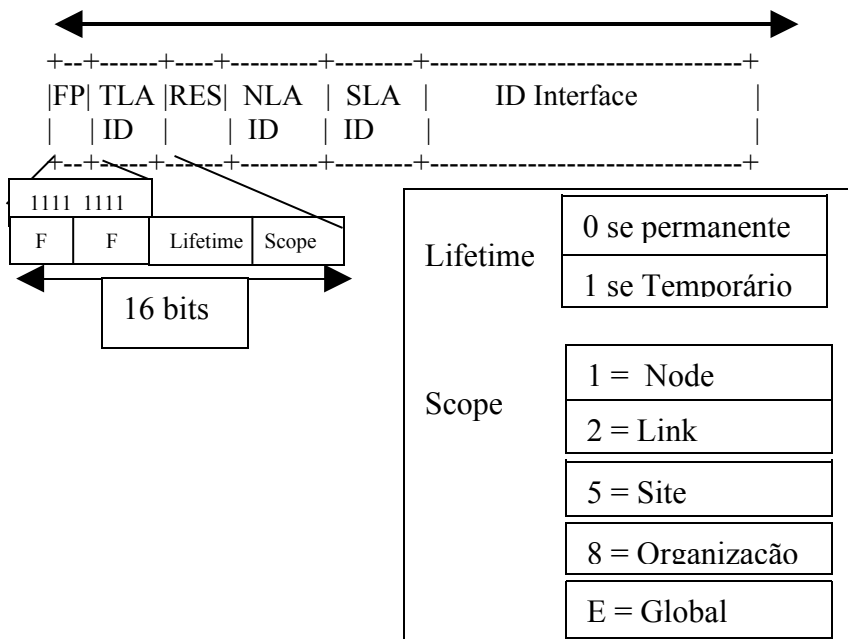


Figura 4.15 – Funcionamento do *Multicast*

4.5 - Anycast

Conforme a RFC 2373 Hinden & Deering (1998) assim como o *multicast*, o destino é um grupo de interfaces, mas em vez de ficar tentando entregar o pacote a todas, o IPv6 tenta entregar a apenas uma interface, especificamente enviado para interface mais próxima, de acordo com o protocolo de roteamento. Por exemplo, ao constatar um grupo de servidores de arquivos cooperativos, um cliente pode usar o anycast para alcançar o mais próximo, sem ter que saber qual é. O *anycast* utiliza-se de endereços regulares *unicast*, como mostra a Figura 4.16 não é possível distinguir sintaticamente qual é um ou qual é o outro, ficando a cargo do sistema de roteamento a escolha do nó que receberá o pacote. Para cada endereço anycast atribuído, existe um prefixo mais longo desse mesmo endereço que identifica a região ao qual todas as interfaces pertencem.

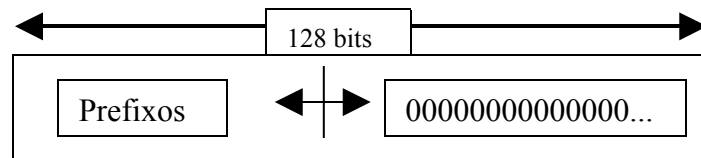


Figura 4.16 – Distribuição Anycast

4.6 - Nova Notação

Os 128 *bits* ou 16 *bytes* de um endereço IPv6 pode ser descrito por oito grupos de 4 dígitos hexadecimais (base 16: de 0 a F), com o símbolo de dois pontos “:” separando cada grupo conforme apresentado no exemplo a seguir:

9000 : 0000 : 0000 : 0000 : 0123 : 4567 : 89AB : CDEF

Para facilitar a escrita, um ou mais grupo de 0000 pode ser substituído por “:” somente uma vez, assim como um zero à esquerda pode ser suprimido, conforme exemplo: 9000 :: 0123 : 4567 : 89AB : CDEF

4.7 - Autoconfiguração

Esta característica de autoconfiguração é denominada *stateless configuration* está presente no IPv6. Dessa maneira não é necessário configurar cada estação da rede manualmente como ocorre hoje.

Existe no IPv6 um conjunto de diversos mecanismos de controle conhecidos como protocolo ND (*Neighbor Discovery protocol*) que é transportado por ICMPv6 (*Internet Control Message Protocolo version 6*). O ND faz com que os nós conectados a um link descubrirem os roteadores ativos através de mensagens enviadas para endereços *multicast*. O DHCPv6 (*Dynamic Host Configuration Protocolo version 6*), também conhecido como *stateful configuration*, fornece mecanismos para configuração automática de endereços IPv6 e registros automáticos e dinâmicos dos nomes dos nós no DNS (*Domain Name System*). O DHCP utiliza-se do tradicional modelo cliente servidor.

O IPv6 apresenta mecanismos de autoconfiguração que visam liberar o usuário da tarefa de configuração, tornando-a automática e transparente. Espera-se, por exemplo, que ao comprar um computador o usuário possa simplesmente conectá-lo a uma rede e acessá-la, sem necessidade de lidar com a configuração de interfaces, protocolos, endereços, etc.

Outro objetivo da autoconfiguração é permitir a mobilidade, ou seja, a utilização de um mesmo computador em vários locais e em redes distintas.

Segue-se um exemplo ilustrativo na computação móvel: um executivo poderia estar utilizando seu computador portátil conectado por cabos à rede local da empresa. Ao desconectar os cabos, o computador deveria utilizar a rede de infravermelho disponível, ainda dentro da empresa, e ao sair do alcance desta, utilizar a rede de radio frequência disponível na cidade. A autoconfiguração deverá permitir o ajuste automático e transparente para o usuário a todas estas situações.

O princípio básico para que os exemplos acima possam se tornar realidade consiste na utilização do endereço *Ethernet* (48 bits) das placas de rede, que a princípio são únicos para cada placa, na constituição do endereço IPv6. Este endereço terá então a forma: FE80:0:0:0:XXXX:XXXX:XXXX, onde XXXX:XXXX:XXXX simboliza o endereço *Ethernet* de 48 bits. Deste modo, a alocação de endereço é altamente facilitada (SILVA 1998).

5- SERVIÇOS IPv6

Os diversos e diferenciados tipos de serviços tais como tráfego de vídeo, voz, imagens, multimídia entre outros hoje em uso na *Internet* e com tendências de crescimento a cada dia, faz com que o IPv4 com seus 8 *bits* para atender os tipos de serviços, sejam tratados de outra forma. No IPv6 o campo *Priority* em conjunto com o *Flow label* que identifica um fluxo contínuo de dados tem como funcionalidade de negociar a Qualidade de Serviço (QoS). O tipo de manipulação necessária pode ser indicada diretamente para o roteador através do RSVP (*ReSource reservation Protocol*) ou por informações no cabeçalho estendido opcional *Hop-by-Hop*. (SILVA 1998).

5.1 - QOS

Os campos de *Flow Label* e *Priority* no cabeçalho são usados para identificar aqueles pacotes que necessitam de "cuidados especiais". São pacotes originados de aplicações multimídia ou de tempo real (SILVA 1998).

5.1.1 *Flow Label*

São 24 *bits* que podem ser usados para identificar um tipo de fluxo de dados (algo como uma conexão ou circuito virtual). Classifica-se em fluxo

orientado aquele que demanda muitos pacotes, e fluxo não-orientado aquele que não demanda muitos pacotes, muito tráfego. A Tabela 5.1 apresenta alguns exemplos de aplicações para esses tipos de fluxo.

Tabela 5.1 – Prioridades

TRÁFEGO ORIENTADO	TRÁFEGO NÃO-ORIENTADO
FTP	DNS
Telnet	SMTP
HTTP	NTP
Multimídia	POP
	SNMP

O uso deste campo não é explicitamente definido, mas imagina-se que um fluxo orientado necessita uma atenção maior que um fluxo não orientado. Caberia aos roteadores negociarem quais são as medidas a serem tomadas. Dentro de cada categoria (orientada ou não) haveria um identificador de fluxo que sugeriria o tratamento daquele caso. Quando os roteadores recebessem um pacote com determinado identificador de fluxo, consultariam uma tabela onde recuperariam o tipo de tratamento (TANEMBAUM, 1996).

5.1.2 - Prioridade

Este campo determina a prioridade do datagrama em relação a outros datagramas na mensagem de origem. Todos os pacotes de determinado fluxo devem ter a mesma prioridade, portanto estes são dois campos usados em

conjunto. Espera-se que esse campo identifique e priorize aplicações iterativas, como sessão remota.

O uso efetivo se dá quando o pacote enfrenta um tráfego congestionado. Valores de 0 a 7 nesse campo lidam com transmissões (geralmente TCP) que podem ser retardadas no caso de um congestionamento. Valores de 8 a 15 se referem a aplicações cujo tráfego é constante e um atraso implicaria em perda de informação, como vídeo e áudio.

5.2 - Mapeamento de Endereços em Nomes

O serviço *Domain Name System* (DNS) é de extrema utilidade para o protocolo IPv6. O documento *DNS Extensions to Support IP version 6*, especifica um novo tipo de campo DNS de 128 *bits* denominado AAAA ou quad A, que permite mapear nomes de domínios em endereços IPv6. Foi também definido o mapeamento de endereços IPv6 em nomes.

Este campo (suportado já nas versões 8.1.x do BIND) permite a utilização de DNS de modo transparente: se um *host* com pilha dupla efetuar uma *query* DNS e receber um endereço de 32 *bits*, utiliza IPv4; se o endereço for de 128 *bits*, utiliza IPv6. O *software* BIND versão 8.1.x apresenta atualmente suporte para IPv6 (*queries* e campos *quad A*) (SILVA 1998).

6- MOBILIDADE

A utilização de dispositivos móveis ligados em rede tem vindo a aumentar, resultando principalmente da explosão de adesões à Internet. Novas tecnologias (redes sem fios, ligação através de micro-ondas) estão a se tornar comuns rapidamente. O IPv4 não previu a utilização crescente deste tipo de dispositivos.

Na medida em que um *host* se movimenta de uma rede para outra, seu endereço IP irá sendo alterado. Isto, a princípio, geraria o problema de nunca se saber o endereço atual deste *host*, inviabilizando a comunicação com o mesmo. No entanto, o IPv6 implementa um método que permite esta mobilidade.

Este método consiste basicamente em todo *host* móvel possuir um endereço fixo em sua rede local original, conhecido como *home address*. Ao se autoconfigurar em uma rede qualquer, o *host* móvel envia uma mensagem a sua rede local “avisando” seu novo endereço na rede na qual é visitante. Deste modo, todos os pacotes destinados ao seu endereço original serão roteados para o seu endereço visitante, permitindo assim a recepção de pacotes de forma transparente.

6.1 - *Mobile IPv4*

A versão original do IP não previa suporte a mobilidade de *hosts*, e o assim como diversos entre outros protocolos, o *Mobile IP* foi desenvolvido *ad hoc*. Muitos usuários da *Internet* possuem computadores portáteis, e necessitam acesso à rede fora de sua base, ou mesmo em movimentação. Infelizmente, o sistema de endereçamento do IP não facilita a tarefa de manter um *host* móvel

conectado. O modo como pacotes são entregues ao *host* de destino não permite que o mesmo saia de sua rede sem que seja necessário alterar o endereço do mesmo, ou, alternativamente, avisar a todos roteadores a nova localização do *host* móvel. Esta última alternativa é impraticável, portanto a solução oferecida não deve contar com esta possibilidade.

Em 1996 foi publicada a RFC 2002 (*IP Mobility Support*), o primeiro documento tratando da implementação do suporte a mobilidade no IP. Vários documentos se seguiram a este, aprimorando o protocolo que será descrito brevemente neste tópico.

Conforme RFC 2002 (PERKINS, 1996), os usuários com MH (*Host Móvel*) possuem uma rede local base, e podem se locomover entre diferentes áreas, onde uma área é tipicamente uma rede remota em que o usuário ou uma célula *wireless* se conecta. Cada uma dessas áreas possui um ou mais FA⁷, que são responsáveis por gerenciar a permanência de MHs na sua respectiva área. Além disso, cada área tem um HA⁸, responsável pelos MHs de suas bases originais, mas que estão visitando outras áreas.

Sempre que um MH adentra uma área, ele deve se registrar com o FA responsável por esta área. O FA periodicamente faz um broadcast de seu endereço, anunciando sua presença para os novos MHs, que, caso não queiram esperar pelo *broadcast* do FA, também podem fazer um *broadcast* perguntando quem é o FA responsável.

O *host* móvel se registra com o FA, dando o endereço do seu *home agent* e informações de segurança. O FA, por sua vez, entra em contato com o HA, fornecendo as informações de segurança recebidas do MH, e anunciando a presença do mesmo na área. Satisfeito com as informações, o HA diz ao FA

7 FA (*Foreign Agent*) ou agente remoto

8 HA (*Home Agent*), ou agente local

para prosseguir com o registro, e este último avisa então ao MH que ele está registrado.

A partir deste ponto, sempre que um *host* tentar contatar o MH, o pacote direcionado ao mesmo será interceptado pelo HA, na rede local base do MH, pois a princípio o remetente não sabe se o MH está na sua área permanente ou em uma outra área. Após interceptar o pacote, o HA faz o seguinte:

1. Encapsula este pacote no campo de dados de um novo pacote, e direciona este novo pacote para o FA responsável pela área onde o MH se encontra. Este procedimento é chamado de *tunneling*, ou, tunelamento.
2. Avisa ao remetente do pacote original para enviar os próximos pacotes destinados ao MH diretamente para o FA, usando a mesma técnica de tunneling. Desta forma, os pacotes subsequentes não precisam passar pelo HA.

A figura 6.1 ilustra o processo:

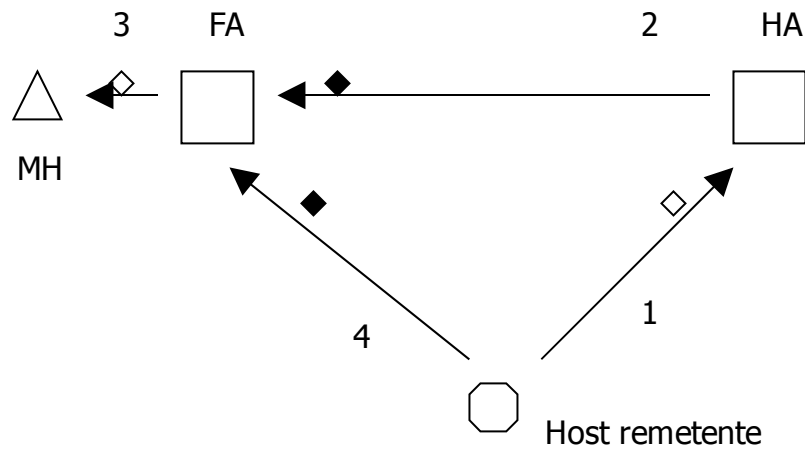


Figura 6.1 – Envio de pacotes pelo *Mobile IPv4*

Na Figura 6.1, tem-se:

- a) Um host deseja contactar o MH, e envia um pacote para seu endereço.
- b) O HA sabe que o MH não está na área, e então intercepta o pacote, encapsula o mesmo em um novo pacote e envia para o FA.
- c) O FA recebe o pacote, desencapsula o pacote original e envia este para o MH.
- d) Pacotes subseqüentes são enviados diretamente para o FA.

6.2 - Mobile IPv6

O MIPv6 (*Mobile IPv6*) é um protocolo que foi desenvolvido como um subconjunto do *Internet Protocol version 6* (IPv6) para dar suporte a conexões móveis. O MIPv6 facilita o movimento do nó de uma rede do tipo *Ethernet* para outra do mesmo tipo assim como o movimento de uma rede *Ethernet* para uma célula de Wireless LAN.

MIPv6 é uma atualização do padrão *Mobile IP* (RFC 2002), criado pela IETF, e foi desenvolvido para autenticar dispositivos móveis (conhecidos como nós móveis) usando endereços IPv6.

6.2.1 – Funcionalidade IPv6 Mobile

O Mobile IPv6, assim como o MIP, permite que um nó móvel se mova de uma rede a outra sem quebrar uma conexão. Isto significa que o endereço original (*home address*) nunca se modifica. O nó móvel (MN) pode estar em

qualquer lugar, que os pacotes serão roteados corretamente para ele através de mecanismos apropriados. O movimento é, desta forma, transparente para a camada de transporte e para aplicações que usam o protocolo TCP/IP e Mobile IPv6.

O *home address* é constituído de um prefixo válido no *link* de sua rede original (*home network*). É através deste endereço que um nó correspondente irá se comunicar com o nó móvel, independente de onde este estiver. Quando o nó móvel muda de rede, ele mantém o *home address* e recebe outro endereço, o *care-of address* (COA), constituído de um prefixo válido em uma rede estrangeira. Este endereço é conseguido de forma *stateless* ou *stateful*, (sem ou com servidor de endereços, respectivamente). Desta forma, o MN terá um *home address* e um ou mais *care-of address* quando está se movendo entre redes.

Para que seja possível saber onde o nó móvel se encontra, uma associação entre *home address* e *care-of address* deve ser realizada (*binding*). Esta associação do *care-of address* é feita pelo nó móvel, no *home agent* (HA). Esta associação é realizada através de um *binding registration*, onde o MN envia mensagens chamadas *Binding Updates* (BU) para o HA, que responde com uma mensagem *Binding Acknowledgement* (BA).

Os nós correspondentes (CNs) no MIPv6 possuem "inteligência" para a otimização de rota, ou seja, eles podem armazenar *bindings* entre *home address* e *care-of address* de nós móveis. Sendo assim, um nó móvel pode fornecer informações sobre sua localização para CNs, através do *correspondent binding procedure*. Neste procedimento, um mecanismo de autorização de estabelecimento de *binding* é realizado, chamado de *return routability procedure*. A figura 6.2 mostra um cenário de mobilidade IPv6 com elementos básicos:

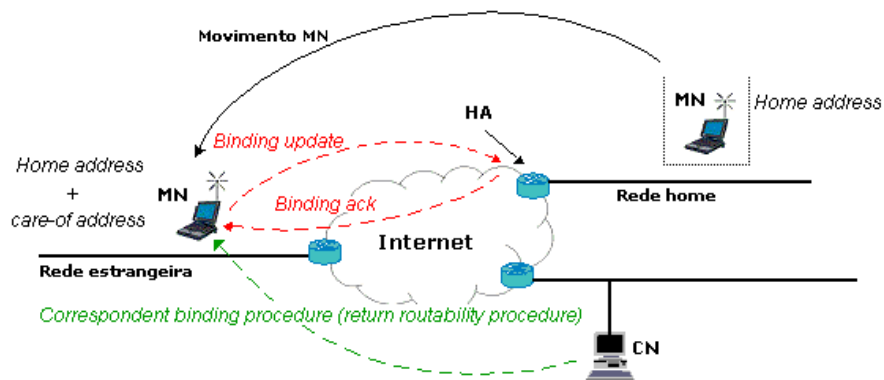


Figura 6.2 – Visão Geral IPv6
(Fonte: Rede Nacional de Ensino e Pesquisa)

Nota-se que o *Foreign agent* (FA), presente no MIPv4, não existe mais.

A comunicação entre MN e CN pode acontecer de dois modos:

- Tunelamento bidirecional: não requer que o CN tenha suporte ao MIPv6 e que o MN tenha se registrado com o CN. Os pacotes são roteados do CN para o HA e do HA é tunelado para o MN. Depois, o MN responde para o HA por túnel que, por sua vez, responde para o CN. Cada pacote interceptado é tunelado para o *care-of address* do MN;
- Otimização de rota: o CN deve ter suporte ao MIPv6 ("inteligência" para *binding*) e o MN deve se registrar com o CN. Neste caso, o CN, antes de enviar o pacote, busca em uma cache uma associação entre *home address* e *care-of address* do MN. Se existir associação, o pacote será roteado para o *care-of address* do nó móvel diretamente. Isto elimina congestionamento no *home link* e no HA. A figura 6.3 ilustra os dois modos:

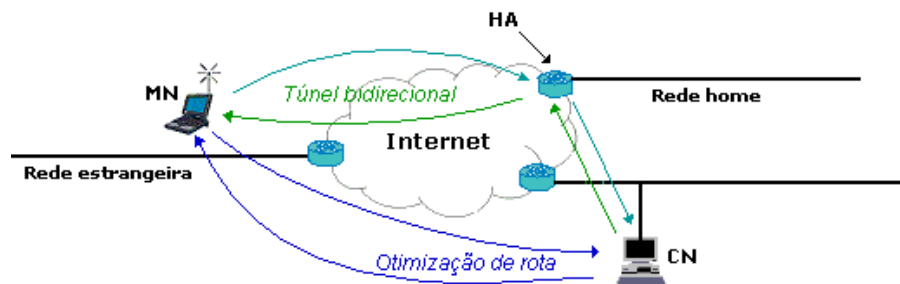


Figura 6.3 – Modos de comunicação entre CNe MN
(Fonte: Rede Nacional de Ensino e Pesquisa)

6.3 - Operação do Host Móvel

Quando em uma rede remota, o *host* móvel continua utilizando seu *home-address*, em adição aos COAs, podendo escolher entre estes endereços para utilizar como endereço de origem. Do ponto de vista das camadas de protocolo acima do *Mobile IP*, o *host* móvel geralmente vai utilizar seu *home-address* a fim de manter a transparência para estes aplicativos. Isso implica em transparência também para os *hosts* com os quais ele se comunica, como se o *host* móvel nunca tivesse saído de sua rede local.

Ao enviar tais pacotes, eles serão modificados de forma a mover o *home-address* do campo origem para a opção de *home-address* e utilizar um dos COAs como endereço de origem do pacote. Estas alterações serão revertidas no receptor do pacote, inserindo o *home-address* de volta no campo de origem, assim alcançando a desejada transparência para camadas superiores.

Para alguns tipos curtos de comunicação, particularmente as operações em que novas tentativas podem ser efetuadas com facilidade (ex.: DNS), o *host* móvel pode utilizar diretamente um de seus *care-of-address* no campo de origem do pacote, eliminando assim o processamento extra de utilizar a opção *home-address*. Para isso, no entanto, é necessário que a aplicação saiba tratar este tipo de comunicação, caso contrário recomenda-se utilizar o método mencionado anteriormente.

É importante reafirmar que pacotes enviados quando o *host* móvel está na sua rede local não precisam sofrer nenhuma alteração, sendo processados da mesma forma que pacotes gerados em *hosts* estacionários. Da mesma forma, pacotes enviados pelo *host* móvel de uma rede remota utilizando um IP da rede remota não são alterados. Um problema visto nesta forma de trabalhar é que *hosts* correspondentes que fazem algum tipo de verificação baseada no *home-address* do *host* móvel não irão reconhecê-lo, uma vez que o endereço de origem não é o *home-address*, e sim o endereço utilizado na rede remota.

Um *host* móvel, quando em rede remota, vai receber pacotes destinados ao seu *home-address* de uma das seguintes três maneiras:

1. Pacotes enviados por um correspondente que não tem uma entrada para o *host* móvel em seu *binding-cache* serão interceptados pelo HA na rede local do *host* móvel e tunelados para o COA do *host* móvel.
2. Pacotes enviados por um correspondente que tem uma entrada para o *host* móvel em seu *binding-cache* farão uso do cabeçalho de roteamento, e enviados diretamente para o COA do *host* móvel, recuperado do cache.
3. Pacotes enviados por um correspondente que tem uma entrada inválida para o *host* móvel em seu *binding-cache* serão enviados como no item 2.

O pacote, ao chegar na rede onde o COA utilizado reside, será descartado, pois o *host* móvel não se encontra mais nesta rede. Porém, se o *host* móvel tiver enviado um *binding-update* para o FA desta rede, o pacote será interceptado por este último, sendo tunelado para o novo COA do *host* móvel, que receberá o pacote com sucesso.

Para os casos 1 e 3 descritos acima, é recomendado que o *host* móvel envie um *binding-update* para o remetente do pacote, a fim de criar ou atualizar a sua entrada no *binding-cache* do correspondente.

O *host* móvel, visando descobrir a rede em que está, utiliza as facilidades implementadas pelo IPv6 *Neighbor Discovery*, além de ouvir os anúncios feitos pelos FA's locais. Baseado nestes métodos, o *host* móvel mantém uma lista com FA's e redes considerados ativos, além dos COA's disponíveis para seu uso, escolhendo uma das opções para ser o padrão, ou ainda, COA primário. Ao detectar que o FA em uso não está ativo, o *host* móvel deve escolher um novo de sua lista, e, devido à alteração do COA, fazer um *binding-update* com seu HA.

Esta lista também é utilizada quando o *host* móvel passa a utilizar outro COA; todos os FA's recebem um *binding-update*, para que possam tunelar pacotes enviados para o COA antigo. Ao selecionar um novo COA primário, o *host* móvel é obrigado a enviar um *binding-update* para seu HA com os seguintes requisitos:

- *bit* H (*home registration*) deve estar ligado;
- *bit* A (*acknowledge*) deve estar ligado;
- pacote tem que ter a opção de *home-address* preenchida;
- endereço de origem do pacote deve ser o COA a ser registrado, a menos que a sub- opção de COA alternativo esteja sendo utilizada;

é recomendado que o tempo de vida informado para este *binding* seja menor ou igual ao tempo de vida do COA.

Caso o *host* móvel tenha mais de um *home-address*, um *binding-update* deverá ser feito para cada um deles, obviamente em pacotes separados.

Em algumas situações o *host* móvel pode não saber o endereço de um *host* em sua rede local que pode servir de *home-agent* para ele. Neste caso o *host* móvel pode tentar descobrir o endereço de um *host*, como foi mencionado no tópico “Operação do *Home Agent*”. É obrigatório que o *host* móvel tente o *binding-update* antes com o HA que ele já vinha utilizando, caso exista.

O *host* móvel pode mandar *binding-updates* também para correspondentes, para que estes atualizem seu cache, evitando assim o overhead de tunelamento por parte do FA. O COA utilizado nesta operação pode ser escolhido livremente dentre os disponíveis para o *host* móvel; se o endereço utilizado for o *home-address*, o correspondente deve remover do *cache* a entrada correspondente ao *host* móvel.

É interessante notar que caso o *host* móvel não queira divulgar sua localização atual para um correspondente, basta não enviar *binding-updates* para este correspondente. Neste caso os pacotes serão tunelados pelo HA.

Para ativar o encaminhamento de pacotes por parte do FA, mencionado anteriormente, o *host* móvel deve mandar um *binding-update* para qualquer FA da rede remota onde ele estava anteriormente, utilizando como endereço de origem o COA utilizado nesta rede remota anterior e passando seu novo COA como o endereço a ser inserido no cache. É importante que o *bit H* (*home registration*) esteja ligado nesta mensagem, significando que o FA deve atuar temporariamente como HA do *host* móvel naquela rede. Um aspecto interessante é que este FA não sabe necessariamente o *home-address* do *host*

móvel, e nem precisa saber, pois tudo que ele tem que fazer é tunelar pacotes destinados ao COA antigo para o novo COA.

Correspondentes desejando saber o COA do *host* móvel podem enviar um *binding-request* para o mesmo. Ao receber este pedido o *host* móvel pode escolher entre responder ou não o pedido. Como mencionado anteriormente, o *host* móvel pode responder com um COA que é o seu próprio *home-address*, indicando que ele recebeu o pedido mas não quer divulgar sua localização. O *host* móvel pode também, é claro, processar o pedido normalmente, enviando um *binding-update* com seu COA atual.

A multiplicidade de COAs no *host* móvel auxilia o processo de *smooth handovers*, no sentido que ao se deslocar por células sobrepostas, o *host* móvel pode receber pacotes destinados a qualquer um dos COAs disponíveis. Assim, mesmo trocando de COA primário, o *host* móvel pode manter o COA primário anterior na lista de COAs disponíveis para uso. Outro benefício na disponibilidade de vários COA é a possibilidade do *host* móvel escolher como COA primário o que pertence à rede com melhor nível de comunicação, aumentando assim a eficiência de transmissão.

Ao retornar para sua rede local, o *host* móvel deve enviar um *binding-update* para seu HA, pedindo para que este deixe de interceptar e tunelar os pacotes destinados a ele. Uma situação que requer cuidado pode ocorrer neste ponto: caso o *host* móvel não saiba o endereço de camada *link* do *home-agent*, ele deve solicitar este endereço utilizando *unicast* e endereço de origem não especificado para evitar problemas com o algoritmo de detecção de endereços duplicados.

6.4 - Considerações de Segurança

O uso de *binding-updates* não autenticados é tido como problema de segurança conhecido pela comunidade. Como este recurso permite que pacotes destinados a um *host* móvel sejam enviados para um endereço remoto (COA), o uso não autenticado do mesmo poderia acarretar em *updates* maliciosos que levariam um correspondente a enviar pacotes destinados ao *host* móvel para um terceiro *host*.

Assim como o *binding-update*, as confirmações destes *updates* também devem ser autenticados. Alguém mal-intencionado poderia enganar o *host* móvel, fazendo este acreditar em um resultado falso de um *update* com seu HA, por exemplo. Ao contrário do *binding-update* e sua confirmação, o *binding-request* não oferece nenhum risco de segurança, pois seu uso não implica em mudança de estados no *host* móvel ou no correspondente.

Os números de seqüência utilizados nos *binding-updates* e suas confirmações também devem ser renegociados para evitar ataques de repetição de pacotes. A autenticação dos pacotes de *binding-update* é feita através da verificação do campo BSA (*binding security association*), existente conceitualmente em um campo do *binding-cache*. O processo para estabelecer tal associação não está definido no “Draft IETF *Mobile IPv6*”, e portanto está em aberto.

Um assunto já pendente no IPv4 que não foi resolvido no IPv6 é a questão do ARP gratuito, ou *Neighbor Advertisement* no IPv6, que tem como efeito associar um endereço IP a um determinado endereço de camada *link*. Um usuário malicioso conectado à rede local pode emitir um destes pacotes, anulando assim a proteção que o uso de *hubs*

chaveados traz. A técnica é um pouco mais complicada do que o explicado aqui, e está amplamente coberta em outros textos.

Além dessas questões, devemos considerar que o ambiente de computação móvel é bastante diferente do ordinário, estando as estações muitas vezes conectadas através de dispositivos sem fio (*wireless*). Tais conexões são muito vulneráveis a interceptação passiva dos dados, além de outros ataques ativos, como enviar pacotes especialmente preparados para causar algum tipo de confusão. Alguns destes problemas podem ser evitados utilizando-se criptografia. Estações móveis são ainda mais fáceis de serem roubadas, e neste caso, as chaves de autenticação ou de criptografia, bem como outras informações serão comprometidas.

Usuários preocupados em manter secreta a localidade do *host* móvel, podem usar uma técnica que consiste em tunelar para o HA os pacotes a serem enviados. Desta forma, os pacotes vão parecer estar saindo da rede onde o *home-agent* reside, mantendo assim o local verdadeiro de transmissão secreto.

6.5- Futuro do MIPv6

O HMIPv6 (Hierarchical Mobile IPv6) foi proposto como uma melhoria do MIPv6. Ele foi desenhado com o propósito de reduzir a quantidade de sinalização requerida e para melhorar a velocidade de *handoff* para conexões móveis. O HMIPv6 foi proposto pelo IETF (*Internet Engineering Task Force*).

O MIPv6 define meios de gerenciar mobilidade global, mas não enfoca o gerenciamento de mobilidade local separadamente. Ele utiliza-se do mesmo mecanismo em ambos os casos, o que causa uma ineficiência na utilização de recursos no caso de mobilidade local.

O HMIPv6 adiciona um novo nível, baseado em MIPv6, que separa mobilidade local de global. No HMIPv6, a mobilidade global é gerenciada pelos protocolos do MIPv6, enquanto *handoffs* locais são gerenciados localmente.

O novo nó em HMIPv6 é chamado de MAP (*Mobility Anchor Point*). O MAP, que substitui o *foreign agent* do MIPv4, ao contrário do mesmo, não exige estar presente em cada subnet. O MAP também ajuda a diminuir a latência referente aos *handoffs* devido ao fato de ele poder se atualizar mais rapidamente que o remoto *home agent*.

6.6 - Mobile IPv6 x Mobile IPv4

O *Mobile IPv6* é uma combinação natural das experiências adquiridas com o desenvolvimento do suporte para *hosts* móveis no IPv4 com as oportunidades oferecidas pelo desenvolvimento de uma nova versão do IP e suas propriedades. Neste tópico é apresentado um resumo das maiores diferenças entre o *Mobile IPv4* e o *Mobile IPv6*.

A otimização conhecida no *Mobile IPv4* como “*Route Optimization*” (otimização de rota) foi incorporada ao IPv6, ao invés de fazer parte de um conjunto de extensões opcionais. Esta otimização consiste em armazenar em um *cache* o COA do *host* móvel, permitindo ao correspondente enviar pacotes sem precisar do tunelamento feito pelo HA.

O problema das regras de *firewall* que impediam o MH de usar o seu *home-address* como source dos pacotes foi resolvido permitindo que os MH's usem o COA para enviar os pacotes. O *home address* é enviado opcionalmente no pacote, e a habilidade de processar este campo é quesito obrigatório em qualquer implementação do IPv6.

Enquanto que no IPv4 pacotes UDP eram utilizados para enviar mensagens de controle, no IPv6 estas mensagens podem ir nos pacotes já existentes, fazendo o que é conhecido como “*piggybacking*”.

A possibilidade de utilizar o COA para enviar pacotes permite simplificar o roteamento de pacotes *multicast*. No IPv4, estes pacotes tinham que ser enviados ao HA através de *tunneling*, para que o MH pudesse utilizar seu *home address*.

Não é mais necessário o uso de roteadores especiais para atuar como FA; todos os roteadores IPv6 vão ter esta funcionalidade. O mecanismo de detecção de movimento no IPv6 permite que, ao perder o *link* com o antigo FA, o *mobile host* passe a utilizar um novo FA e um novo COA.

O novo formato do pacote do IP permite que os pacotes para o MH não precisem ser encapsulados. Ao invés disso, é utilizado o cabeçalho de roteamento, reduzindo assim o *overhead* da transmissão. A implementação do *Neighbor Discovery* permite também que o processo de interceptação dos pacotes destinados ao MH por parte do HA não dependa da camada inferior (data *link*), simplificando o protocolo e aumentando sua robustez.

Limitações na definição do ICMP no IPv4 exigiam processamento especial para mensagens ICMP, através do tunnel soft state. Esta limitação foi eliminada na nova definição do ICMP.

A busca automática do HA por parte do MH agora é feita utilizando o *anycast*. Desta forma, o MH só receberá a resposta de um HA, ao invés de

respostas de todos os HA da sua área permanente. Os *binding updates*, no IPv4 feitos pelo HA e FA, agora são feitos pelo próprio *host* móvel, permitindo que ele decida se vai ou não divulgar seu COA.

É permitido ao MH ter mais de um COA, ajudando assim a implementar o que é chamado de *Smooth Handover*, onde o MH não perde as conexões estabelecidas ao trocar de área.

7- SEGURANÇA

A especificação do protocolo IPv6 inclui segurança na camada de rede através de uma arquitetura criada para tal, a IPsec. Esta arquitetura é bastante flexível, de modo a evitar problemas relacionados com restrições de exportação de criptografia e não impede a utilização de outros mecanismos de segurança por parte das aplicações.

Tendo em conta que a utilização generalizada do IPv6 não é um processo imediato, a IPsec foi concebida de modo a poder ser utilizada com o protocolo IPv4. Existem já implementações de sistemas *Unix* que englobam a arquitetura de segurança para o protocolo IP.

7.1 - IP Security Protocol (IPsec)

Segundo Karn *et al.* (1998), o projeto IPsec representa um esforço desenvolvido pelo *Working Group* IPsec da IETF para desenvolver uma arquitetura de segurança para o protocolo IP e tem como objetivos:

- a) Criar uma infra-estrutura de rede segura providenciando proteção nos cabeçalhos de dados e de chaves;
- b) Reduzir a preocupação de implementar mecanismos de segurança nas aplicações;
- c) Compatibilizar o seu funcionamento com mecanismos de segurança já existentes e utilizados por aplicações;
- d) Evitar problemas de exportação de criptografia;

- e) Ser parte integrante do protocolo IPv6 e poder ser aplicável ao IPv4.

Através dos seus componentes, a IPSec usa este conceito para permitir a implementação de redes virtuais privadas e seguras através de redes públicas tais como a *Internet*.

7.1.1 - IPSec - Características

Conforme Sofia (1998) o IPSec representa uma arquitetura para o protocolo IP, integrando mecanismos de autenticação, gestão e distribuição de chaves que podem ser usados com qualquer das versões do protocolo IP. O IPSec utiliza como mecanismos de autenticação dois cabeçalhos de extensão específicos do protocolo IPv6: o cabeçalho de autenticação (*Authentication Header*) e o cabeçalho de encapsulamento de dados de segurança (*Encapsulating Security Payload Header*).

Além destes dois cabeçalhos, o IPSec define também o conceito de associação de segurança - conjunto de diretivas que permite negociar algoritmos de cifra a utilizar. Uma associação de segurança representa uma relação entre duas ou mais entidades comunicantes e descreve quais os mecanismos de segurança a utilizar para estabelecer uma comunicação segura. A associação de segurança permite negociar protocolos, algoritmos de cifra e chaves a usar, e contém informação sobre:

- a) Algoritmo e modo de autenticação a aplicar ao cabeçalho de autenticação;
- b) Chaves usadas no algoritmo de autenticação;

- c) Algoritmo, modo e transformada de cifra utilizados no cabeçalho de encapsulamento de dados de segurança, ESP;
- d) Chaves usadas no algoritmo de cifra do cabeçalho de encapsulamento de dados;
- e) Chaves de autenticação usadas com o algoritmo que faz parte da transformada ESP;
- f) Tempo de vida da chave;
- g) Tempo de vida da associação de segurança;
- h) Endereço(s) fonte da associação de segurança;
- i) Nível de sensibilidade dos dados protegidos.

Na prática, uma associação de segurança é representada por um índice de parâmetros de segurança - *Security Parameter Index (SPI)* - com um endereço IP destino. O SPI é um campo que surge nos cabeçalhos de segurança IPv6 (AH e ESP), que não é cifrado na transmissão, já que a sua informação é essencial para decifrar a informação transmitida.

Quando uma entidade quiser estabelecer uma associação de segurança, utiliza um SPI e um endereço destino (pertencente à entidade com que deseja estabelecer comunicação segura) e envia essa informação à entidade com que quer estabelecer o canal seguro. Assim, para cada sessão de comunicação autenticada entre dois nodos, são necessários dois SPI - um para cada sentido, dado que cada associação de segurança é unidirecional.

O IPSec apresenta uma estrutura bastante flexível, que não obriga à utilização de algoritmos de autenticação ou cifra específicos. Assim, o IPSec pode interagir com as normas mais recentes. No entanto, dada a necessidade de segurança, a IETF definiu como algoritmos a usar:

- a) HMAC-MD5 e HMAC-SHA-1 para autenticação (quer no cabeçalho AH, quer no ESP);
- b) DES-CBC, para a cifra usada no cabeçalho ESP.

O IPSec integra gestão manual de chaves. A gestão é da responsabilidade de protocolos criados para este fim, tais como o SKIP, da *Sun Microsystems*, ou o *Photuris*, (acrónimo em latim para desenvolvido por *Phil Karn*), ou ainda o protocolo *Internet Key Exchange, IKE*.

Na medida em que estes cabeçalhos são cabeçalhos de extensão que irão ser adicionados a um cabeçalho IP, os encaminhadores podem interpretá-los como fazendo parte integrante dos dados, o que permite a compatibilidade destes mecanismos com equipamento que compreende o protocolo IP mas não o IPSec.

Os componentes da IPSec são:

- a) Cabeçalho de Autenticação (AH)
- b) Cabeçalho de Encapsulamento de Dados de Segurança (ESP)
- c) Mecanismos de Gestão de Chaves (SOFIA, 1998).

7.1.2 - Cabeçalhos de Autenticação (AH)

O cabeçalho de autenticação, Figura 7.1, representa um cabeçalho de extensão do protocolo IPv6 e foi criado para validar a identidade de entidades comunicantes. Ele tem a função de identificar o emissor e destino corretos, verificar se o emissor que afirma ter enviado os dados é a mesma pessoa e providenciar mecanismos de autenticação aos datagramas IP.

Próximo Cabeçalho	Tamanho do Módulo	Reservado
Índice de Parâmetros de Segurança (SPI)		
Número de seqüência		
Dados de Autenticação		

Figura 7.1 - Cabeçalho de Autenticação

Na medida em que alguns dos campos do datagrama IP são alterados durante a transmissão e dado que o cabeçalho de autenticação é adicionado na fonte do datagrama. Este cabeçalho por si só não fornece proteção contra ataques de análise de tráfego ou confidencialidade, sendo para tal usado normalmente em conjunto com o cabeçalho de encapsulamento de dados (SILVA, 2003).

7.1.3 - Cabeçalho de Encapsulamento de Dados de Segurança

O cabeçalho de encapsulamento de dados de segurança (ESP), Figura 7.2, é um cabeçalho de extensão pertencente ao protocolo IPv6 que fornece integridade e confidencialidade aos datagramas IP através da cifra dos dados contidos no datagrama.

Índice de Parâmetros de Segurança (SPI) Dados Transformada, Tamanho Variável

Figura 7.2 - Cabeçalho de Encapsulamento de Dados de Segurança

A utilização do ESP pode ser efetuada de dois modos:

- a) Modo de Transporte (*transport-mode*). Providencia proteção principalmente no respeitante aos protocolos da camada superior. É utilizado majoritariamente em comunicação

extremo a extremo entre dois nodos, por exemplo, um cliente e um servidor. Este modo cifra informação do protocolo da camada de transporte, adicionando-lhe em seguida um novo cabeçalho IP não-cifrado, pelo que se torna vantajoso em redes relativamente pequenas, nas quais o(s) servidor(es) e nodo implementam a IPSec ;

- b) Modo de Túnel (*tunnel-mode*). Providencia proteção ao pacote IP. Para tal, após a adição dos campos ESP ao pacote IP, todo o pacote é tratado como o módulo de dados de um novo pacote IP. Assim, pode ser usado para enviar dados cifrados através de um túnel, o que permite enviar dados independentemente da infra-estrutura utilizada. Um exemplo é o envio de pacotes IP através de canais virtuais criados numa rede IP pública, como a *Internet*. Através deste modo, pode ser fornecida segurança a um grupo de nodos que não implementem o IPSec.

7.1.4 - Mecanismos de Gestão de Chaves

Além dos mecanismos de autenticação e validação da informação a IPSec necessita de um mecanismo eficiente de gestão de chaves. A gestão de chaves diz respeito à criação, eliminação e alteração das chaves. Embora a IPSec não integre um mecanismo de gestão de chaves, a IETF definiu como norma de gestão o protocolo híbrido ISAKMP/*Oakley* também denominado IKE, *Internet Key Exchange*, que se encontra baseado nos documentos:

- a) ISAKMP - *Internet Security Association and Key Management Protocol*. Protocolo que descreve uma infra-estrutura para a gestão de associações de segurança;
- b) *Oakley* - protocolo que define material de chaves para cifra, *hashing* e autenticação e é compatível com a gestão de associações de segurança ISAKMP;
- c) *Internet Domain Of Interpretation* - define parâmetros ISAKMP para as associações de segurança IPsec no domínio Internet;
- d) Resolução ISAKMP/*Oakley* - define o perfil do protocolo híbrido ISAKMP/*Oakley*, escolhido como norma de gestão de chaves criptográficas pela *Internet Engineering Task Force*;
- e) IKE - *Internet Key Exchange*.

O IKE usa a porta 500 do protocolo UDP e interage com os restantes mecanismos de segurança IPsec através de associações de segurança. Assim, o IKE proporciona a possibilidade de estabelecer associações de segurança para diversos protocolos e aplicações de segurança, tendo assim um método transparente e aberto de associar diferentes mecanismos de segurança, sem envolver as entidades participantes na comunicação. O IKE agrupa funcionalidades dos protocolos ISAKMP e *Oakley*:

- Mensagens ISAKMP. Mensagens pré-definidas e compostas de módulos. São usadas para trocar informação nas trocas de chaves;
- Modos *Oakley*. O protocolo *Oakley* define trocas de chaves a que chama modos e que utilizam mensagens ISAKMP.

Quando uma entidade pretende estabelecer comunicação segura, passa pelas fases IKE:

Fase 1: esta fase ocorre num meio inseguro. Tem o objetivo de estabelecer um canal seguro que irá proteger as trocas da Fase 2. É executada uma vez para várias fases 2;

Fase 2: esta fase ocorre no canal seguro criado na fase 1. As suas negociações têm o objetivo de estabelecer as associações de segurança que irão proteger a comunicação.

Após estas duas fases, encontra-se estabelecido um canal seguro através do qual se pode efetuar comunicação segura (SOFIA, 1998).

8 – IMPLEMENTAÇÕES IPv6 – IPv4

O objetivo desta implementação foi realizar a instalação de dois nós experimentais IPV6, apresentando compatibilidades, funcionabilidades e interoperabilidade entre os protocolos IPV6 com o protocolo IPv4. O trabalho compreendeu os seguintes componentes:

- a) Implementação de duas máquinas com suporte nativo Ipv6;
- b) Análise e compatibilidade funcional de sistema operacional;
- c) Instalação de servidores de DNS IPv6;
- d) Resolução de problemas de endereçamento;
- e) Definição, implementação e análise de problemas de roteamento;
- f) Análise de compatibilidade e interoperacionalidade IPv4/IPv6.

8.1 - Implementação de duas máquinas com suporte nativo IPv6

Foi implementado 2 (dois) computadores nas dependências Eletronorte para teste, sendo configurações mínimas:

Computador Cliente – Pentium III , 1100 MHz, 256 MB RAM, 20 GB HD, 1 Placa de Rede.

Computador Servidor – Pentium III , 1100 MHz, 512 MB RAM, 20 GB HD, 2 Placas de Rede.

A figura 8.1 demonstra topologia utilizada:

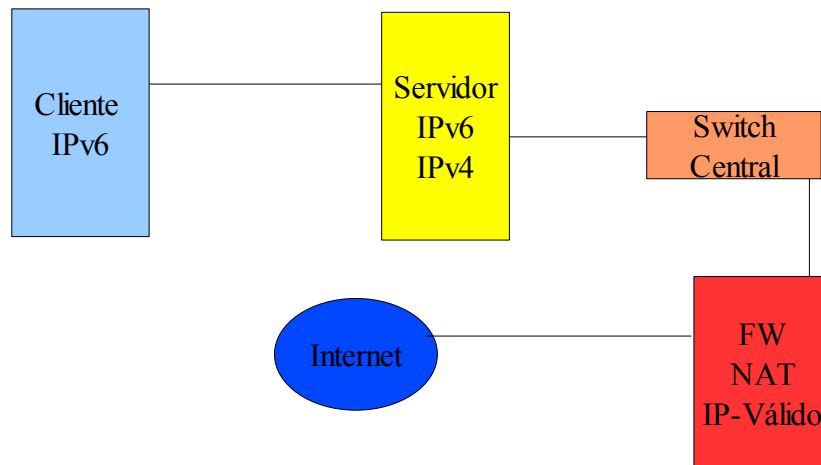


Figura 8.1 – Topologia de Implementação.

8.2 - Análise e compatibilidade funcional de sistema operacional

A escolha do Sistema Operacional foi voltado ao software livre *Linux*, pois o mesmo atende todas as características e compatibilidades funcionais mediante ao projeto IPv6.

Características: *Linux Fedora 4.1, Kernel 2.5.16.*

Todos pacotes necessários para execução desse projeto foram instalados junto ao sistema operacional em modo servidor:

Rede

DNS

Roteamento

8.3 - Instalação de servidores de DNS IPv6

A instalação do DNS foi atribuída junto ao sistema operacional. O BIND utilizado para o DNS foi o 9.2.1 – 16.

Foi configurado o /etc/named.conf , /var/named e /etc/resolv.conf e foi modificado:

No arquivo etc/named.conf (No final do arquivo)

```
Zone " ipv6.proj " IN {  
    Type master;  
    File " dom.dns";  
};
```

/var/named

No arquivo dom.dns (adicionado o arquivo)

```
$TTL 86400  
$ORIGIN    ipv6.proj.  
@          IN      SOA   ipv6ipv4.ipv6.proj  
root.ipv6.proj.(  
  
                    1      ;      Serial  
                    3H    ;      Refresh
```

```
15M ; Retry
1W ; Expire
1D ; minimum
```

```
IN NS ipv6ipv4.ipv6.proj.
ipv6ipv4 IN AAAA 3ffe:2b00:100:f102::1
```

Configurações atribuídas no /etc/resolv.conf

Máquina Cliente

```
nameserver 192.168.1.1
nameserver [3ffe:2b00:100:f102::1]
```

Máquina Servidor

```
nameserver 192.168.1.1
nameserver [3ffe:2b00:100:f102::1]
```

Após as configurações atribuídas, foi reinicializado o serviço de rede e de DNS:

```
service network restart
service named start
```

Para verificação do funcionamento DNS foi utilizado o comando ping para as duas estações Cliente e Servidor:

```
ping ipv6ipv4.ipv6.proj
```

8.4 - Resolução de problemas de endereçamento

Para inserção de endereços IPv6 foi utilizado a atribuição designada para RNP, 3ffe:2b00..., foi criado as redes aleatoriamente dentro do escopo RNP. No caso IPv4 foi utilizado os endereços privados 192.168.1.x classe C. Segue abaixo as configurações das placas no servidor e cliente.

Servidor

No arquivo /etc/sysconfig/network-scripts/eth1

```
DEVICE=ETH1
ONBOOT=yes
IPADDR=10.5.0.201
NETMASK=255.255.255.0
NETWORK=10.5.0.0
BROADCAST=10.5.255.255
IPV6INIT=yes
IPV6_AUTOCONFIG=no
IPV6ADDR=3ffe:2b00:101:f101::1/64
```

```
DEVICE=ETH0
ONBOOT=yes
IPADDR=192.168.1.1
NETMASK=255.255.255.0
NETWORK=192.168.1.0
BROADCAST=192.168.0.255
IPV6INIT=yes
```

```
IPV6_AUTOCONFIG=no
IPV6ADDR=3ffe:2b00:100:f102::1/64
```

Cliente

```
DEVICE=ETH0
ONBOOT=yes
IPADDR=192.168.1.2
NETMASK=255.255.255.0
NETWORK=192.168.1.0
BROADCAST=192.168.0.255
```

```
IPV6INIT=yes
IPV6_AUTOCONFIG=no
IPV6ADDR=3ffe:2b00:100:f101::1/64
```

Para verificação do funcionamento foi utilizado o comando *ifconfig* e *ping* nas duas estações, Cliente e Servidor: Com *ifconfig* pode-se verificar a configuração atribuída nas placas de redes, atenção para o escopo global e escopo de *link*. O escopo global apresenta-se como o endereçamento atribuído na placa e o escopo de *link* é uma atribuição gerada automaticamente identificando o *link*.

ifconfig (Servidor)

```
Eth1  Endereço inet6: fe80::202:44ff:fe19:6d9f/64 Escopo: Link
      Endereço inet6: 3ffe:2b00:101:f101::1/64 Escopo Global
Eth0  Endereço inet6: fe80::202:44ff:fe19:6d89/64 Escopo: Link
      Endereço inet6: 3ffe:2b00:100:f102::1/64 Escopo Global
```

ifconfig (Cliente)

Eth0 Endereço inet6: fe80::250:bfff:fed2:2bdd /64 Escopo: Link
Endereço inet6: 3ffe:2b00:100:f101::1/64 Escopo Global

ping 3ffe:2b00:100:f101::1
ping 3ffe:2b00:101:f101::1
ping 3ffe:2b00:100:f102::1

ping 192.168.1.1
ping 192.168.1.2
ping 10.5.0.201
ping 10.5.0.200

ping 200.252.155.21

8.5 - Definição, implementação e análise de políticas de roteamento.

Foi adotado a política de roteamento mediante ao *iptables* ativando módulo *modprobe* IPv4 e IPv6 como demonstra abaixo:

No arquivo `/etc/rc.local`

```
modprobe iptable_nat  
iptables -t nat -A POSTROUTING -o eth1 -j MASQUERADE  
echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
/sbin/modprobe ipv6
```

```
touch /var/lock/subsys/local
```

Foi inserido a configuração para o funcionamento de comunicação.

No arquivo /etc/sysconfig/network

```
NETWORKING=yes  
HOSTNAME=IPV6IPV4  
NETWORKING_IPV6=yes
```

#IPV6FORWARDING=yes (se acionado, somente funciona o encaminhamento de endereços Ipv6).

Foi adicionado manualmente a rota e o gateway para acesso fora da Eltronorte.

```
route add -net 0.0.0.0 netmask 0.0.0.0 dev eth1  
route add default gw 10.5.0.1
```

Foi atribuído uma NAT para conectar fora da rede Eletronorte pelo endereço IP válido 200.252.155.9.

8.6 - Análise de compatibilidade e interoperacionalidade IPv4/IPv6

Mediante todas implementações e análises feitas e apresentado no decorrer da implementação do IPv6 no sistema operacional linux Fedora 4.1

pôde-se verificar que o IPv4 e IPv6 trabalham em conjunto e são compatíveis em relação ao endereçamento, roteamento e DNS, e o IPv6 apresenta completa integração com sistema baseado em software livre quando esse vem atribuído com suporte nativo ao protocolo IPv6, bastando compreender o funcionamento e sua configuração.

9 - CONCLUSÃO

A flexibilidade do protocolo IP e sua natureza aberta o tornaram sucesso mundial. Atualmente, mesmo após anos de existência, percebe-se que o mesmo está apto às novas exigências.

A IETF e mais especificamente o grupo de trabalho IPng tem hoje a oportunidade de oferecer ao mundo uma solução para várias restrições do IP atual (Ipv4).

De acordo pesquisas realizadas no desenvolvimento dessa monografia identificou-se que o IPv6 surgiu para resolver problemas a médio e longo prazo na falta de endereços IP. Além disso, apresenta suporte nativo à segurança e ao *multicast*, assim como a possibilidade de oferecer tratamento diferenciado para diferentes tipos de serviços.

Foi implementado nesse trabalho a instalação do *software* livre com IPv6, configurado o endereçamento, resolução de nomes, roteamento e características específicas de integração do IPv4 com o IPv6, onde apresentou normal funcionalidades diante dos testes executados.

Atinge-se uma nova era, onde tudo e todos passam a obter um ponto em comum de comunicação e esse trabalho de monografia indica que temos formas de dar continuidade ao crescimento e integração no protocolo IP.

BIBLIOGRAFIA

- ALMEIDA, Rubens Queiroz de. *Protocolo Ipv6*. [on-line]. Disponível na Internet via [www. url: http://www.dicas-l.unicamp.br/Treinamentos/tcpip](http://www.dicas-l.unicamp.br/Treinamentos/tcpip). Arquivo capturado em 18 de Março de 2005.
- ANTON, Eric Ricardo. *Arquitetura TCP/IP*. [on-line]. Disponível na Internet via [www. url: http://www.gta.ufrj.br/grad/99_2/eric/index.htm#IPv6](http://www.gta.ufrj.br/grad/99_2/eric/index.htm#IPv6). Arquivo capturado em 18 de Novembro de 2005.
- ARTOLA, Esmilda Saens. *PROTOCOLOS IP, TCP E UDP*. [on-line]. Disponível na Internet via [www. url: http://penta.ufrgs.br/hometcp.html](http://penta.ufrgs.br/hometcp.html). Arquivo capturado em 19 de Março de 2005.
- BRADNER, Scott O., *et al.* RFC 1752 *IPNG, Internet Protocol Next Generation*, Addison- Wesley Publishing Company, 1996.
- 6BONE. *Testes para o desenvolvimento do Ipv6*. [on-line]. Disponível na Internet via [www. url: http://www.6bone.net/](http://www.6bone.net/) Arquivo capturado em 18 de Março de 2005.
- COMER, Douglas E. *Interligacao em rede com TCP/IP – volume 1 Principios, protocolos e arquitetura*. Rio de Janeiro: Campus, 1998.
- DEERING, S. & HINDEN, R. *RFC 1883 - Internet Protocol, Version 6 (IPv6) Specification*. [on-line]. Disponível na Internet via [www. url: http://www.faqs.org/rfcs/rfc1883.html](http://www.faqs.org/rfcs/rfc1883.html) . Arquivo capturado em 18 de setembro de 2005.
- FCCN - Fundação para Computação Científica Nacional. *Piloto IPv6*. [on-line]. Disponível na Internet via [www. url: http://www.fccn.pt/rccn/projectos/ipv6](http://www.fccn.pt/rccn/projectos/ipv6). Arquivo capturado em 15 de Novembro de 2005.

- HINDEN, Robert. *Informações sobre o IPng*. [on-line]. Disponível na Internet via www. url: <http://playground.sun.com/pub/ipng/html/ipng-main.html>. Arquivo capturado em 20 de setembro de 2005.
- HINDEN, R. *et al. RFC 2373 - IP Version 6 Addressing Architecture*. [on-line]. Disponível na Internet via www. url: <http://www.ietf.org/rfc/rfc2373.txt>. Arquivo capturado em 18 de setembro de 2005.
- HINDEN, R. *et al. RFC 1884 - IP Version 6 Addressing Architecture*. [on-line]. Disponível na Internet via www. url: <http://rfc.net/rfc1884.html>. Arquivo capturado em 18 de setembro de 2005.
- HINDEN, R. & POSTEN, J. *RFC 1897 - IPv6 Testing Address Allocation*. [on-line]. Disponível na Internet via www. url: <http://rfc.net/rfc1897.html>. Arquivo capturado em 18 de setembro de 2005.
- HUITEMA, Christian. *IPv6: The New Internet Protocol, 2/e.*: Prentice Hall PTR, 1997.
- ICANN, Internet Corporation for Assigned Names and Numbers. 2005 . Disponível em <http://www.icann.org.br/general>. Arquivo capturado em 15 de abril de 2006.
- IETF - Internet Engineering Task Force. *Informações oficiais relacionadas ao IPng*. [on-line]. Disponível na Internet via www. url: <http://www.ietf.org/ids.by.wg/ipngwg.html> . Arquivo capturado em 20 de outubro de 2005.
- KARN, P. ; METZGER, P. ; SIMPSON, W. *RFC 1829 - The ESP-DES CBC Transform*. [on-line]. Disponível na Internet via www. url: <http://rfc.net/rfc1829.html> . Arquivo capturado em 11 de junho de 2005.
- NACAO, Joao Paulo Gonsiro. *IPng a.k.a. IPv6*. UFRJ - Universidade Federal do Rio de Janeiro - Coordenação dos Programas de Pós-Graduação em Engenharia, 1994.
- PERKINS, C. *RFC 2002 - IP Mobility Support*. [on-line]. Disponível na Internet via www. url: <http://www.ietf.org/rfc/rfc2002.txt>. Arquivo capturado em 15 de novembro de 2005.

- PORTUGAL IPV6 TASK FORCE. *Como Ativar o Protocolo Ipv6 em Linux* [on-line]. Disponível na Internet via www. url: <http://www.ipv6-tf.com.pt/home.htm> . Arquivo capturado em 20 de Maio de 2006.
- REIS, Rui; SOFIA, Helena R. E. C. *Internet Protocol Next Generation*. [on-line]. Disponível na Internet via www. url.: <http://planeta.clix.pt/rute/apts/ipv695/index.htm>. Arquivo capturado em 18 de Abril de 2005.
- REKHTER, Y. *et al. RFC 2073 - DNS Extensions to support IP version 6* [on-line]. Disponível na Internet via www. url: <http://www.ietf.org/rfc/rfc2073.txt> . Arquivo capturado em 18 de Abril de 2005.
- RNP - REDE NACIONAL DE ENSINO E PESQUISA. *A Nova Geração de Protocolos IP* [on-line]. Disponível na Internet via www. url: <http://www.rnp.br/newsgen/9811/intr-ipv6.html>. Arquivo capturado em 18 de setembro de 2005.
- ROBERTO, Paulo. *O Protocolo TCP/IP*. [on-line]. Disponível na Internet via www. url: <http://www.dicas-l.unicamp.br/Treinamentos/tcpip/02.html> Arquivo capturado em 18 de julho de 2005.
- ROSA, Miguel. *IPv6 - IP Next Generation - Estudos sobre o protocolo IP de Nova Geração*. [on-line]. Disponível na Internet via www. url: <http://www.ip6.fc.ul.pt/> . Arquivo capturado em 03 de abril de 2005.
- SILVA, Eduardo S. Machado da. *A Geração Futura de Internet: IPng e IPv6*. [on-line]. Disponível na Internet via www. url: <http://www.lcmi.ufsc.br/~esms/redes/ipv6/> . Arquivo capturado em 04 de abril de 2005.
- SILVA , Lino Sarlo da – *Virtual Private Network (VPN)*. São Paulo: ED. Novatec, 2003.
- SOARES, Luiz Fernando G.; LEMOS, Guido; COLCHER, Sérgio. *Redes de Computadores: das LANs, MANs e WANs*. Rio de Janeiro: Campus, 1995.

SOFIA, Helena R. E. C. *Piloto IPv6 na RCCN*. [on-line]. Disponível na Internet via www. url: <http://planeta.clix.pt/rute/aps/ipv698/index.htm> Arquivo capturado em 05 de maio de 2005.

SOFIA, Helena R. E. C. *Piloto IPv6*. [on-line]. Disponível na Internet via www. url: <http://planeta.clix.pt/rute/aps/ipv699/index.htm> . Arquivo capturado em 05 de maio de 2005.

SOFIA, Helena R. E. C. *Estudo do Protocolo ISAKMP/Oakley como Norma de Gestão de Chaves da Arquitectura de Segurança IPsec*. [on-line]. Disponível na Internet via www. url: <http://planeta.clix.pt/rute/aps/tese/index.htm> . Arquivo capturado em 05 de maio de 2005.

SOFIA, Helena R. E. C. *IPsec*. [on-line]. Disponível na Internet via www. url: <http://planeta.clix.pt/rute/aps/ipsecjan/index.htm> . Arquivo capturado em 08 de junho de 2005.

TANENBAUM, A. S. *Sistemas Operacionais Modernos*. Rio de Janeiro: Prentice-Hall do Brasil, 1995.

TANENBAUM, A. S. *Computer Network* 3ª Edição. Rio de Janeiro: Prentice-Hall do Brasil, 1996.

THOMSON, S.& HUITEMA, C. *RFC 1886 - DNS Extensions to support IP version 6* [on-line]. Disponível na Internet via www. url: <http://www.rnp.br/ipv6/rfc1886.txt> . Arquivo capturado em 18 de Abril de 2005.

THOMSON, S. & NARTEN, T. *RFC 2462 - IPv6 Stateless Address Auto configuration..* [on-line]. Disponível na Internet via www. url: <http://www.faqs.org/rfcs/rfc2462.html> . Arquivo capturado em 19 de Abril de 2005.