I7-196

UNIVERSIDADE EDUARDO MONDLANE

Faculdade de Ciências

Departamento de Matemática e Informática

TRABALHO DE LICENCIATURA

Internet Protocol v6 na UEM

Autora: Atanásia Amaral Mapapá

Maputo, Abril de 2005

UNIVERSIDADE EDUARDO MONDLANE

Faculdade de Ciências

Departamento de Matemática e Informática

TRABALHO DE LICENCIATURA

Internet Protocol v6 na UEM

Supervisor:
Engº Américo Muchanga

Co- Supervisor:
Engº Eneas Hunguana

Discente:
Atanásia Mapapá

Maputo, Abril de 2005

Dedicatória

À memória de minha mãe e de meu irmão Fai. Ao meu prezado pai, meu filho Yurí e meus irmãos Lucinda, Abílio, Necas, Linda e Mungi.

Agradecimentos

A Deus.

Aos meus Pais que sempre apoiaram-me nesta longa jornada estudantil, agradeço pelo incentivo, carinho e amor.

Aos meus supervisores, Eng. Américo Muchanga e Eng. Eneas Hunguana, por toda a paciência e perseverança. Muito obrigada!

Um especial agradecimento a vocês: Abílio Mapapá, Edson, Keogan, Mana Lucinda, Linda, Nina, Mungi, Melita, Necas, Alberto Muchanga, Sandro Issufo, Mimi, Judite Mandlate, Jacinto Muchine, Tsamba, Arnaldo Cumbe, Marcelo Munguanaze, Pedro Matusse, Quiquinha, Lurdes, Lola, Lúcia, Safry, Ana, Ndoca e todos os outros que não constam nesta lista e que contribuíram directa ou indirectamente para realização deste trabalho.

Por último mas não com menos importância, agradeço aos meus colegas, docentes, funcionários e amigos da faculdade, pela coragem e incentivo dados na realização deste trabalho.

Declaração de Honra

Declaro por minha honra, que este trabalho é fruto da minha profunda investigação e não foi submetido para um outro grau que não seja o indicado, Licenciatura em Informática na Universidade Eduardo Mondlane.

Maputo, Abril de 2005

A autora

Atanásia Amaral Mapapá

Atanásia Amaral Mapapá

Resumo

No início da década de 90, ocorreu uma grande explosão da Internet, tornando insuficientes os 32 bits usados no protocolo IP para endereçar cada máquina na rede. Com o tempo a tendência era de esgotamento de endereços IP. Para aliviar esse facto, a *Internet Engeneering Task Force* - IETF decidiu trabalhar na solução do problema que primeiramente seria a redistribuição dos endereços, aumentando o tempo de vida do IP e proporcionando tempo para criar e implementar uma nova solução. Após vários anos de trabalho pelo IETF, chegou-se a um novo protocolo, denominado *Internet Protocol Next Generation ou Internet Protocol Version 6* (IPng ou IPv6). O IPv6 para além de possuir um tamanho de IP maior que 32 bits, 128 bits, o seu cabeçalho foi optimizado, retirando-se alguns campos obsoletos e acrescentando novos campos e cabeçalhos de extensão. Outras mudanças serviram para tornar fácil o trabalho dos administradores de redes, como m, por exemplo, a capacidade de autoconfiguração das máquinas, que auxilia na instalação e configuração de uma rede com suporte ao IPv6.

O presente trabalho é constituído por uma descrição teórica do IPv6 e alguns conceitos básicos de redes relacionados com o mesmo e apresenta uma componente prática que consiste na configuração de uma rede IPv6 e aplicações principais da internet que suportam o IPv6 como: *Domain Name System* (DNS), *Web Server*, *File Transfer Protocol* (FTP), *Secure Shell* (SSH) usando uma rede experimental, estabelecida para o efeito, no Centro de Informática da Universidade Eduardo Mondlane.

LISTA DE FIGURAS

LISTA DE TABELAS

ÍNDICE

1 Introdução

Nos últimos anos tem ocorrido um grande desenvolvimento da rede mundial de computadores, designada Internet. A existência e facilidade de acesso a esta e a outras redes de computadores tem tido um grande impacto sobre a disponibilidade de todo o tipo de informação e sobre a forma como é utilizada na ciência em geral.

Com a explosão desta rede mundial tornou-se claro que os endereços disponíveis não seriam suficientes para satisfazer as exigências de um número cada vez maior de computadores ligados à Internet. É desta maneira que a comunidade de Internet, através da *Internet Engineering Task Force* (IETF), iniciou o desenvolvimento da Internet Protocol version 6 (IPv6).

A implementação de redes com base no IPv6 possibilita o crescimento de novas redes corporativas, educacionais, científicas, governamentais bem como a abertura de novas técnicas, formas de aplicações de outras tecnologias que influenciam o quotidiano de cada indivíduo. Alguns exemplos das inovações dessa tecnologia são aparelhos celulares com endereçamento IPv6 e a possibilidade de expansão dessa tecnologia para aparelhos de uso domésticos, possibilitando a sua operação via rede.

A Universidade Eduardo Mondlane (UEM) é uma instituição académica, cuja função principal é a educação, e possui uma vasta rede que interliga os seus diversos órgãos e fornece o acesso a Internet.

Tendo em conta a função que esta instituição exerce sobre a sua infra-estrutura de rede e de outras entidades a quem ela presta serviços através do Centro de Informática da Universidade Eduardo Mondlane (CIUEM), um dos maiores Internet Service Provider (ISP) em Moçambique, surge o interesse de se implementar nesta rede o suporte ao protocolo IPv6.

É neste sentido que o presente trabalho estará voltado, pretendendo-se com o mesmo implementar o protocolo IPv6 na rede da UEM permitindo, desta forma, que esta seja parte integrante da comunidade científica que está a contribuir para o desenvolvimento e implementação do protocolo assim como das aplicações de rede nele baseadas.

2 Descrição do problema

A utilização de novas tecnologias e a disponibilidade de novos serviços implicou uma explosão da adesão à Internet, em que o factor qualidade se tornou imperativo. Na concepção do protocolo IP, há meio da década 70, não foram previstas necessidades emergentes da sociedade e da sua relação com os sistemas de comunicação actuais, em particular com a Internet.

Características tais como: Espaço de Endereçamento, Auto-configuração, Mobilidade, Segurança, Qualidade de Serviço e Suporte de Aplicações de transmissão de dados em Tempo-Real não foram previstas no IPv4.

A rede da UEM funciona numa plataforma cujo suporte é o protocolo IPv4, podendo se verificar que o suporte a aplicações em tempo real, é uma das componentes que necessitam de ser revistas, pois nos sistemas de comunicação IPv4, principalmente na Internet, serviços como transmissão de áudio e vídeo em tempo-real estão a generalizar-se rapidamente. O IPv4 não prevê meios de reserva e gestão de largura de banda, de recursos e ainda de controlo do tempo de resposta, o que vem limitar o funcionamento de aplicações deste tipo sobre IPv4.

A necessidade de uma infra-estrutura segura é um imperativo para o desenvolvimento de actividades na UEM. Actualmente, é corrente a utilização de segurança ao nível da camada de Aplicação do modelo OSI[1]. No entanto, tal não assegura a integridade dos dados transmitidos a nível das camadas mais abaixo, o IPv4 deixa a segurança a cargo das aplicações. No entanto, é cada vez mais necessário ter mecanismos de autenticação, integridade e confidencialidade de dados, em particular na Internet.

Estes são alguns dos argumentos que justificam a implementação do IPv6 para uma instituição como a UEM, onde a comunicação é considerada a base para o seu funcionamento. É neste âmbito que pretende-se com este trabalho estabelecer na UEM uma rede baseada no IPv6, mostrando a sua importância, utilidade, suas diferenças com IPv4 e sobre tudo a sua integração.

[1] O modelo de referência Open systems interconection (OSI) foi desenvolvido pela ISO (International Organization for Standardization) como padrão de arquitetura aberta e baseado em camadas.

3 Objectivos

3.1 Geral

- ✓ Propor a implementação do suporte ao IPv6 na rede da UEM.

3.2 Específicos

- ✓ Analisar a tecnologia de ligação da rede Intranet e Internet na UEM;
- ✓ Identificar as necessidades do uso de IPv6 na rede da UEM;
- ✓ Desenhar um modelo de integração deste protocolo na rede;
- ✓ Efectuar testes de implementação do protocolo IPv6 numa rede experimental;
- ✓ Estabelecer os serviços básicos na Internet com base no IPv6, tais como, Webserver, DNS, FTP e SSH;
- ✓ Elaborar um guião de implementação de uma LAN IPv6;

4 Material e Métodos

Para que os objectivos propostos fossem concretizados seguiu-se a realização das seguintes actividades::

- Pesquisa e consulta bibliográfica;
- Testes laboratoriais usando sistemas operativos Unix;
- Entrevistas não estruturadas aos gestores e administradores da rede UEM.

A pesquisa e consulta bibliográfica, consulta a documentação existente na UEM, permitiu o fácil entendimento da arquitectura de rede na UEM, permitiu também o melhor entendimento dos objectivos da instituição com relação a rede onde tece sua adequação ao IPv6.

Os testes foram realizados no laboratório do CIUEM por forma a fazer uso dos recursos como PCs, sistemas Windows e Linux para a montagem da rede experimental que foi usada para o estudo.

As entrevistas aos administradores desta rede visaram reunir informações relevantes para a compreensão quanto a forma como a rede da UEM se encontra preparada face aos desafios que esta nova solução se propõe.

5 Fundamentos de Redes

Uma rede é um sistema inter conectado de dispositivos computacionais que fornece acesso compartilhado e económico a serviços de computadores [1].

Não restam dúvidas de que um dos maiores benefícios de uma rede é a partilha de recursos entre os usuários ou mesmo o fornecimento de um meio de armazenamento final superior ao que é utilizado sem a rede.

Basicamente pode-se considerar a constituição de uma rede em 2 computadores interligados com o objectivo de partilhar dados [2]. A ideia de dois computadores interligados por um cabo pode não parecer extraordinária mas no passado representou uma grande conquista nas comunicações [1].

Dois computadores ou nós seriam o número mínimo de dispositivos necessários para se formar uma rede. O número máximo não é predeterminado, teoricamente todos os computadores do mundo poderiam estar interligados.

5.1 Uso das Redes de Computadores

Na década de 1950, computadores eram máquinas grandes e complexas, operadas por pessoas altamente especializadas [2].

Avanços na década 60 possibilitaram o desenvolvimento dos primeiros terminais interactivos, permitindo aos usuários acesso ao computador central através de linha de comunicação onde passam a ter mecanismos de interacção directa com o computador.

A partir da década 1990 as redes começaram a oferecer serviços para pessoas individuais em suas próprias casas.

Muitas instituições têm um número significativo de computadores em operação, frequentemente instalados em locais distantes entre si e ainda cada computador têm uma tarefa específica. Por exemplo uma universidade com diversas faculdades, pode ter um computador em cada uma delas com sistemas para monitorar aproveitamento do estudante, pagamento de propinas etc. Inicialmente esses computadores poderiam funcionar de forma independente dos demais, mas, num determinado momento, pode-se decidir conecta-los para que seja possível coleccionar e extrair informação sobre toda instituição. Este caso pode ser tratado como **Partilha de recursos**, tendo como objectivo colocar todos os programas, equipamentos e especialmente dados ao alcance de todos utilizadores da rede, independentemente da localização física do recurso e do utilizador outro exemplo seria, uma instituição com trinta computadores em rede não precisaria de alocar exactamente trinta impressoras de serviço, uma única impressora pode servir a rede.

A rede também aumenta a **confiabilidade** do sistema, pois tem fontes alternativas de armazenamento. Por exemplo um mesmo arquivo pode ser colocado em diversos computadores da rede para no caso de ocorrer uma falha de hardware num dos computadores, poder-se recorrer ao seu backup.

Outra vantagem oferecida pelas redes é a **escalabilidade,** sendo a possibilidade de aumentar gradualmente o desempenho do sistema a medida que cresce o volume de carga, bastando para tal, que se adicionem mais processadores.

O **acesso a informação remotamente,** é uma área que é muito usada recorrendo a redes de computadores, uma das áreas em que ela já vem se destacando é o acesso a instituições financeiras. Muitas pessoas pagam suas contas, administram contas bancárias e gerênciam investimentos electronicamente.

Outra aplicação que pertence a esta categoria é o acesso a sistemas de informação como a World Wide Web, que contém, dados sobre artes, negócios, culinária, ciência e uma infinidade de outros assuntos.

Outra grande categoria do uso das redes é a interacção pessoa a pessoa, pode-se ver que o correio electrónico ou e-mail já é usado em larga escala por milhões de pessoas. O e-mail em tempo real permite que usuários remotos se comuniquem instantaneamente, vendo e ouvindo uns aos outros. Essa tecnologia possibilita a realização de reuniões virtuais, as chamadas **video-conferências,** entre pessoas separadas por uma grande distância.

No **entretenimento**, diversos filmes podem se tornar interactivos permitindo que o usuário altere o rumo da história, com cenários alternativos para todos os rumos.

Outras aplicações como **jogos de simulação** em tempo real do qual podem participar várias pessoas como simuladores de voo em que uma pessoa de uma equipa tenta acertar os adversários.

Resumindo a possibilidade de mesclar informação, comunicação e entretenimento certamente darão origem a uma nova e avançada indústria baseada nas redes de computadores [2].

6 Internet Protocol version 4 (IPv4)

Os protocolos são uma parte importante da rede. São eles que definem o conjunto de regras e procedimentos à seguir, para a transferência de dados através da rede. Portanto, para que se entenda a fundo como as redes funcionam, é importante perceber o funcionamento dos protocolos [4].

Protocolos de rede são basicamente a parte do sistema operativo da rede encarregue por ditar as normas para a comunicação entre os dispositivos, isto é, trocar informações entre os componentes de rede.

Para que todos os dispositivos da rede consigam comunicar-se, todos eles deverão usar a mesma linguagem, isto é, um mesmo protocolo ou então se disporem de um meio de interpretação que permita o entendimento entre dois dispositivos que implementem protocolos diferentes.

Uma rede pode usar diversos protocolos, embora cada um destes funcione de uma forma particular.

Mensagens geradas por uma máquina que não se adaptam a protocolos aceites não são reconhecidas por outras máquinas. Como acontece com os seres humanos essas mensagens são consideradas como se fossem ruídos.

Os protocolos são desenvolvidos em diversos níveis, as línguas, por exemplo, podem ser vistas como protocolos formulados para permitir que as pessoas se entendam.

Todas as regras e padrões que possibilitam a comunicação entre computadores são adequadamente denominados de protocolos.

O protocolo IP teve origem em 1970 no desenvolvimento da ARPANET[2], esta rede foi depois interligada a outras, formando em 1980 um vasto conjunto que passou a ser conhecido por Internet. Com a implementação do protocolo IP no UNIX, em 1982, um grande número de

[2] ARPANET: rede projectada pelo Departamento de Defesa Americano (DoD), para troca de informação entre bases militares Americanas.

universidades passou a formar as suas redes que por sua vez também foram ligadas à Internet [13].

Este protocolo integrado com o Transmission Control Protocol (TCP), constitui um dos elementos que mantém a Internet unida, o IP pertence a camada de rede, como tal fornece um serviço de transferência de dados independente da implementação da camada de ligação de dados (nível 2).

A tarefa do IP é de fornecer a melhor forma de transportar pacotes da origem para o destino, independentemente dessas máquinas encontrarem-se ou não na mesma rede.

O protocolo IP é complementado por outros protocolos, nomeadamente o Internet Control Message Protocol (ICMP) e vários protocolos auxiliares de controle de encaminhamento.

6.1 Formato do Pacote IPv4

O pacote IP é *a unidade básica de dados* no nível de rede e encontra-se dividido em duas áreas: cabeçalho e dados.

O cabeçalho tem uma parte fixa de 20 bytes e uma parte opcional de tamanho variável, contém toda a informação necessária que identifica o conteúdo do pacote, como pode ser visto na figura 6-1.

Na área de dados está encapsulado o datagrama do nível superior, ou seja, um datagrama TCP ou UDP.

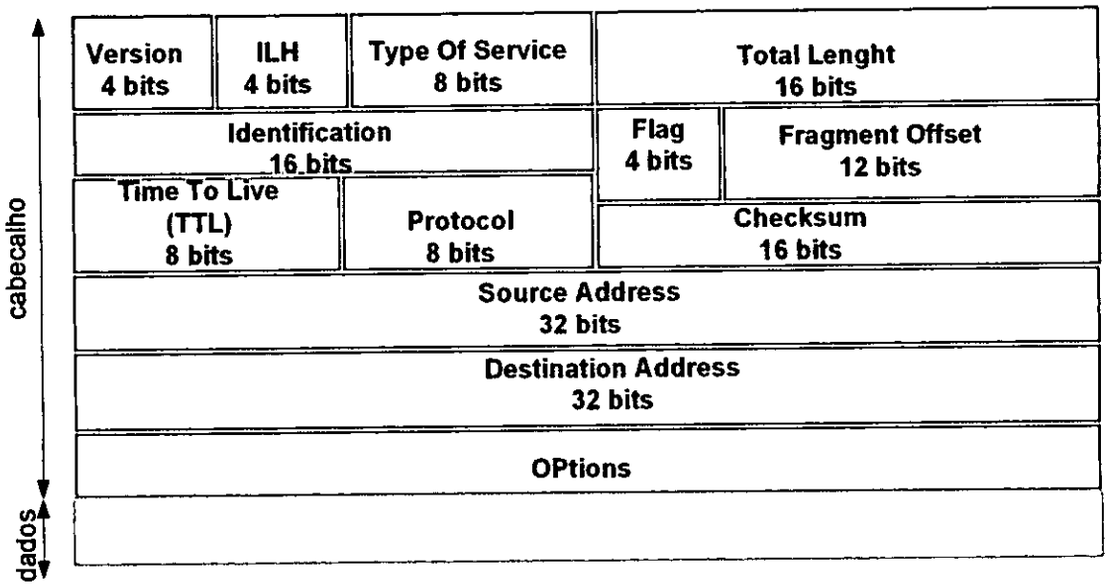

Figura 6-1: Cabeçalho Ipv4 [9].

O campo *VERSION* - Indica a versão do protocolo a que o pacote pertence, com isto é possível verificar as transições entre as versões que pode levar meses ou até anos.

O campo IP Header Length *(IHL)* - Informa o tamanho do cabeçalho (não sendo este constante)

TYPE OF SERVICE - Permite que o host informe a sub-rede, o tipo de serviço que deseja em outras palavras especifica como o pacote poderia ser manejado e dividido. São possíveis várias combinações de confiabilidade e velocidade. Um exemplo seria; para o caso em que se pretendesse transmitir voz digitalizada, a velocidade seria prioritária com relação à segurança. Outro exemplo seria de transferência de arquivos, uma transmissão sem erros seria mais importante do que uma transmissão rápida.

TOTAL LENGTH - Inclui tudo o que há no pacote, cabeçalho e dados. O tamanho máximo é de 65.535 bytes.

IDENTIFICATION, FLAGS e FRAGMENTS - Estes três campos controlam a fragmentação e a união dos pacotes. O campo de identification contém um único inteiro que identifica o pacote, é um campo muito importante porque quando um gateway fragmenta um pacote, ele copia a

maioria dos campos do cabeçalho do pacote em cada fragmento, então a identificação também deve ser copiada, com o propósito de que o destino saiba quais fragmentos pertencem a quais pacotes. Cada fragmento tem o mesmo formato que um pacote completo.

FRAGMENT OFFSET - Especifica o início do pacote original dos dados que estão sendo transportados no fragmento. É medido em unidades de 8 bytes, todos os fragmentos do pacote com excepção do último, devem ser múltiplos de 8 bytes, que é unidade elementar do fragmento.

TTL (Time To Live) – Número de saltos (hops)/links pelos quais o pacote pode ser roteado; decrementado em uma unidade por cada roteador por onde o pacote passa. Empregue para prevenir contra *loops* ou ciclos infinitos de roteamento causados por falhas nas ligações, ou roteadores mal configurados.

PROTOCOL - Especifica que protocolo de alto nível foi usado para criar a mensagem que está sendo transportada na área de dados do pacote.

HEADER-CHECKSUM – Verifica a integridade dos valores do cabeçalho. Pacotes com checksum inválidos são descartados pelos dispositivos de rede.

SOURCE AND DESTINATION IP ADDRESS - Especificam os endereços IP de 32 bits do remetente e destinatário respectivamente.

OPTIONS - Campo opcional. Este campo varia em comprimento dependendo de quais opções estão sendo usadas. Foi projetado para permitir que versões posteriores do protocolo incluam informações inexistentes, possibilitando a experimentação de novas idéias evitando a alocação de bits de cabeçalho para informações raramente necessárias. Existem opções de tamanho variáveis, cada uma começa com um código de 1 byte identificando a opção. Algumas opções são seguidas por um campo de tamanho de opção de 1 byte, e em seguida um ou mais bytes de dados. O campo *Options* é preenchido por um múltiplo de quatro bytes. No momento há cinco opções definidas, mas nem todos os roteadores as usam, a tabela 6-1 mostra essas opções.

Opção	Descrição
Security	Especifica o nível de segurança do datagrama
Strict source routing	Mostra o caminho completo a ser seguido
Loose source rounting	Apresenta uma lista de roteadores que não devem ser esquecidos
Record route	Faz com que cada roteador anexe seu endereço IP
Timestamp	Faz com que cada roteador anexe seu endereço e seu timestamp

Tabela 6-1: Opções do cabeçalho IP

6.2 Endereços IP

Para que um sistema preste serviços de comunicação universalmente, é necessário estabelecer um método que seja aceite globalmente, para identificar os computadores. O mecanismo de endereçamento do protocolo IP utiliza apenas 4 octetos (32 bits) para designar de um modo universal um nó ou host.

Os quatro bytes que constituem um endereço IP são normalmente representados na notação decimal, separados por um ponto. Destes quatro bytes, alguns são usados para identificar a rede e os restantes para identificar o nó dentro dessa rede.

Na prática todos os computadores numa rede partilham o mesmo prefixo de rede, contudo a identificação do host é única, isto é, dois computadores na mesma rede têm o mesmo prefixo de rede e número de host diferente. Do mesmo jeito, dois computadores em redes diferentes têm que ter diferentes prefixos de rede, mas podem ter um mesmo número de host.

6.2.1 Classes de endereço IP

Para providenciar flexibilidade em suportar diferentes números de rede, os desenhadores decidiram que o endereço IP deveria ser dividido em três diferentes classes A, B e C, estas classes são primárias e muitas vezes são tidas como "classful". O formato dessas classes está ilustrado na figura 6-2.

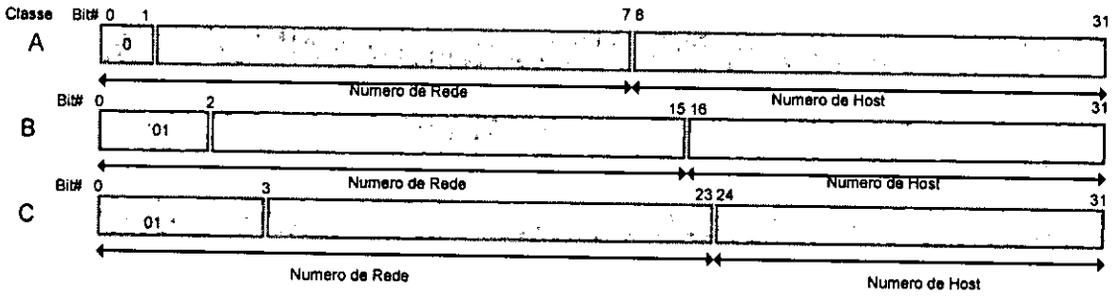

Tabela 6-2: Representação de endereços principais classful.

Por exemplo, se os dois primeiros bits forem 01, o ponto de divisão cai no 15° ou no 16° bit. Isto simplificava o sistema de roteamento no passado, pois os protocolos de roteamento originais não tinham chave para decifrar ou não tinham o "mask", cada roteador identificava o comprimento do prefixo da rede.

A classe A suporta 126 redes, para cada rede encontram-se alocados 16 milhões de hosts, no total o endereço /8 lê-se *slash 8*[3], contém 2.147.483.648 endereços individuais e o espaço de endereçamento IPv4 contém no máximo 4.294.967.296 endereços. Assim pode se dizer que os /8 classes de endereçamento constituem 50% do total de IPv4 endereços *unicast*[4][14].

A classe B tem 16 bits para o prefixo da rede, com os dois bits **1 0** mais significativos, os endereços da classe B têm no máximo 16.384 redes e para cada rede são definidos 65.534. O endereço /16, no total contém 1.073.741.824 endereços individuais e o espaço de endereçamento IPv4 contém no máximo 4.294.967.296 endereços o que quer dizer que os /16 classes de endereçamento constituem 25% do total de IPv4 endereços unicast.

A classe C tem 24 bits no prefixo de rede, com os bits **110** mais significativos, seguidos de 8 bits de host. Esta classe contém no máximo 2.097.152 redes, e para cada rede são 254 hosts. O endereço /24 contém no máximo 536.870.912 endereços individuais, tendo em conta que o espaço de endereçamento IPv4 contém no máximo 4.294.967.296 endereços. Os /24 classes de

[3] slash x ou /x, significa que o cumprimento da parte da rede do endereço IP é composta por x bits.
[4]Unicast: tipo de comunicação 1:1 nas redes de computadores onde participam 2 nodos de cada vez.

endereçamento constituem 12.5% do total de IPv4 endereços unicast [13]. Mais detalhes sobre a obtenção destes valores vide anexo B.

Outras classes são compostas pelas classes D e E onde a D, tem **1110** como o número de bits mais significativos e é reservada para multicast e a classe E, contém os bits mais significativos **1111**, que por sua vez esta classe é reservada para testes experimentais.

A tabela 6-2 apresenta um resumo dos endereços de rede e nó (host).

Classe da rede:	A	B	C
Valores para o primeiro byte	1 a 126	128 a 191	192 a 223
Mascara de rede (HEX)	FF.00.00.00	FF.FF.00.00	FF.FF.FF.00
Número de bits para rede	8	16	24
Primeira rede	1	128.1	192.0.1
Última rede	126	191.254	254.255.254
Número de redes possíveis	126	16382	2097150
Número de bits para host	24	16	8
Primeiro host de cada rede	0.0.1	0.1	1
Último host de cada rede	255.255.254	255.254	254
Número de hosts por rede	16777214	65534	254

Tabela 6-3: Resumo de endereços de rede.

O "broadcast" numa rede é referido ao número de "host" mais elevado. O broadcast apenas se refere a uma dada rede, sendo por isso usado em aplicações de rede local.

A tabela 6-4 apresenta um exemplo de brodcast para cada classe de rede:

Classe da rede:	A	B	C
Número de rede	120	180.10	194.120.135
Endereço de "*broadcast*"	120.255.255.255	180.10.255.255	194.120.135.255

Tabela 6-4: Exemplo de broadcast para cada classe de rede.

6.2.2 Limitações Não Previstas no Endereçamento de Classes

As classes A, B e C, foram facilmente entendidas e implementadas. Mas não foram forçadas a uma alocação eficiente dos espaços de endereçamento. Problemas resultam na falta de classes de rede que eram desenhadas para suportar pequenas e médias empresas. A classe /24 que suporta 254 hosts torna-se muito pequena, enquanto que a classe /16 que suporta 65. 534 hosts é muito grande. No passado a Internet atribuía a pequenas e médias empresas um /16 em vez de um conjunto de /24. Infelizmente isso resultou numa depreciação prematura de endereços /16.

6.2.3 Sub-Redes ou (CIDR- Classless Interdomain Routing)

Em 1985, o RFC 950, definiu um procedimento padrão para suportar a sub-rede, ou por outra, divisão de uma simples classe A, B ou C em pequenas partições. Sub-redes foram introduzidas para solucionar alguns dos problemas que essa parte da Internet começava a ter com as classes de dois níveis de hierarquia [11].

CIDR é um esquema de endereçamento para internet que permite uma alocação mais eficiente dos endereços IP em comparação a alocação por classes A, B e C.

A necessidade deste tipo de endereçamento surgiu devido ao problema de escassez de endereços IP e o crescimento das tabelas globais de roteamento.

Estes problemas foram resolvidos adicionando outro nível de hierarquia na estrutura de endereçamento IP. Além da estrutura de classes de dois níveis de hierarquia, sub-redes, suportam três níveis de hierarquia, a figura 6-2 mostra um exemplo.

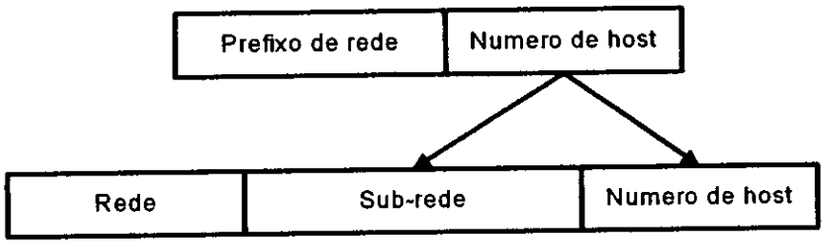

Figura 6-2: Sub-rede com hierarquia de 3 níveis

As sub-redes resolvem o problema de crescimento das tabelas de roteamento, garantido que a estrutura da sub-rede nunca seja visível fora da rede de uma organização privada.

O roteador dentro de uma organização privada precisa de diferenciação entre as sub-redes individuais, mas de acordo com os roteadores da Internet, todas as sub-redes da organização estão colectadas numa única entrada da tabela de roteamento. Isto permite que o administrador da rede introduza arbitrariamente uma complexidade na rede sem, contudo afectar o tamanho da tabela de roteamento IP.

Sub-redes solucionam os problemas com o registo de endereço IP, dando a cada organização um ou mais números de redes provenientes do IPv4.

A organização fica livre de atribuir um número de sub-redes desejado para cada rede interna, isto permite que a organização explore sub-redes adicionais sem que tenha a necessidade de obter um número de rede na Internet, a figura 6-3 sub-redes residentes na rede 130.5.0.0.

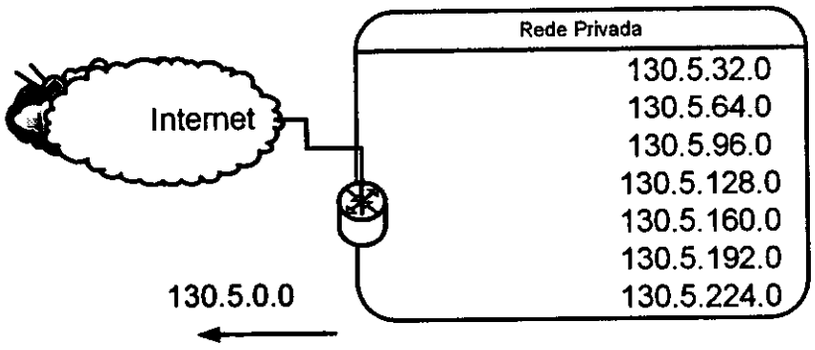

Figura 6-3: Exemplo de sub-redes [5].

Na figura pode se ver um *site* com muitos endereços de rede lógicos, usando endereços de sub-redes para cobrir-lhes com um simples /16. O roteador aceita todo o tráfego de Internet endereçado para rede 130.5.0.0, e encaminha todo tráfego para sub-redes internas baseando-se no 3° octecto de endereços de classes.

A exploração das sub-redes dentro de uma rede privada dá-nos muitos benefícios:

- O tamanho da tabela de roteamento de uma Internet global não cresce, i.é todas as sub-redes encontram-se combinadas numa única entrada da tabela de roteamento;
- O administrador local tem a possibilidade de adicionar sub-redes sem ter necessidade de obter um novo número de rede na Internet;
- *Route Flapping* (Alteração rápida de rotas) numa rede privada não afecta a tabela de roteamento, já que os roteadores da Internet não sabem como atingir uma sub-rede individual.

6.2.4 Considerações no Desenho de Sub-Redes

A exploração do plano de endereçamento requer um pensamento cuidadoso da parte do administrador da rede. Com isto, há palavras chaves que devem ser respondidas antes do desenho de uma rede:

- Quantas sub-redes no total são requeridas pela organização hoje;
- Quantas sub-redes a organização precisará no futuro;
- Quantos hosts existem na maior sub-rede ou site da organização;
- Quantas redes estarão na maior sub-rede da organização no futuro.

Para mais informação sobre o desenho de sub-redes vide Anexo B.

6.2.5 Solução Alternativa de Endereçamento

Network Address Translators (NATs), foram desenhadas para permitir que múltiplos hosts com endereço IP privado, tenham uma partilha dinâmica de um único endereço IP público.

Como tentativa de resolução do problema de endereçamento IP, instituições têm usado soluções alternativas, utilizando NAT. O problema deste tipo de solução é que ela incrementa complexidade na configuração, cria ponto único de falha na rede. Soluções NAT rompem modelos de conexão ponto a ponto, por consequência rompe o esquema de segurança ponto a ponto, serviços de voice over IP e mais.

6.3 Roteamento do Pacote IP

O roteamento IP consiste em decidir para onde enviar um pacote baseando-se no endereço IP destino contido no pacote.

Para entender o roteamento IP deve-se lembrar que a Internet é composta de múltiplas redes físicas interconetadas por computadores chamados Gateways ou Routers.

Quando um pacote é recebido, à parte de rede que compõe o endereço de destino é procurado na tabela de roteamento. Se o destino for uma outra rede, o pacote será encaminhado para o próximo router da interface fornecida na tabela. Caso o destino seja um host local, o pacote será enviado directamente. Esse algoritmo significa que cada roteador só precisa controlar as outras redes e hosts locais, o que reduz muito o tamanho da tabela de roteamento.

Quando a sub-rede é incluída, as tabelas de roteamento são alteradas acrescentando-se entradas do formato (esta rede, esta sub-rede, host). Sendo assim, um roteador da sub-rede k sabe como alcançar todas as outras sub-redes e, também, como chegar a todos hosts da sub-rede k. A sub-rede também reduz o espaço na tabela de roteamento ao criar uma hierarquia de três níveis.

O roteamento pode-se dividir em **directo** e **indirecto.**

6.3.1 Roteamento Directo

Neste tipo de roteamento a transmissão do datagrama é directa de uma máquina à outra. Duas máquinas podem trabalhar em roteamento directo somente se ambas estiverem na mesma rede (por exemplo um mesmo barramento ethernet).

A transmissão de um pacote IP entre duas máquinas numa mesma rede física não envolve gateways. O transmissor (remetente) encapsula o pacote num frame do nível de enlace, liga o endereço IP destino ao endereço físico (de *hardware*) correspondente, e envia o frame resultante diretamente ao destino.

Para saber se a máquina destino está na mesma rede se faz uma comparação entre os endereços IP fonte e destino, especificamente entre os campos que identificam a rede. Se os campos forem iguais significa que o pacote pode ser enviado directamente sem ter que passar por um gateway.

6.3.2 Roteamento Indirecto

Este tipo de roteamento é mais difícil, já que o remetente deve identificar um gateway ao qual o pacote pode ser enviado, depois o gateway deve enviar o pacote a rede destino.

Vamos supor que existam muitas redes interconetadas por gateways, mas que só tenham dois hosts em cada extremo da interconexão das redes figura 6-4, quando o host A quiser enviar ao host B, ele encapsula o pacote e o envia ao gateway mais próximo, neste caso R1. Quando o pacote chega ao gateway R1, o software de IP extrai o datagrama encapsulado, e a rotina do roteamento IP, seleciona o próximo gateway que formará parte do caminho que levará o datagrama ao host destino.

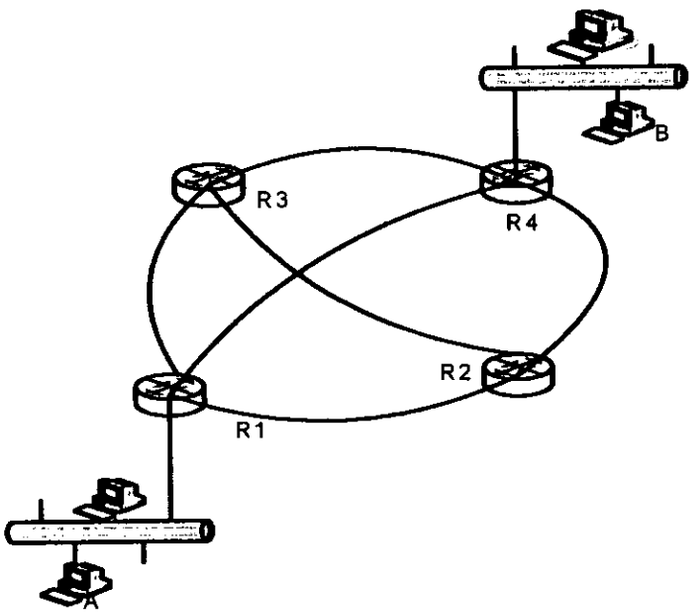

Figura 6-4: Exemplo de Roteamento Indirecto.

Para um gateway saber onde enviar um datagrama, e para um host saber qual o gateway a usar para um destino determinado, precisa-se de um algoritmo de roteamento que manuseie as tabelas de roteamento.

6.3.3 Tabelas de Roteamento

Uma tabela de roteamento é um conjunto de associações **(rede, rota 1, rota 2, ...)**, no qual cada associação regista várias rotas possíveis para atingir a rede indicada. Cada rota tem o formato **(próximo gateway, métrica)** que indica qual o "gateway" seguinte para onde deve ser enviado o "datagrama" e qual a métrica associada a essa rota, (a tabela 6-5 representa um exemplo de uma tabela gerada pelo roteador R1 da figura 6-5). A métrica é uma medição da eficiência do caminho até ao destino, pode ser definida com base em vários critérios tais como:

- Atraso na Transmissão;
- Número de "Hops" (nós intermédios);
- Capacidade das linhas;
- Preço da ligação.

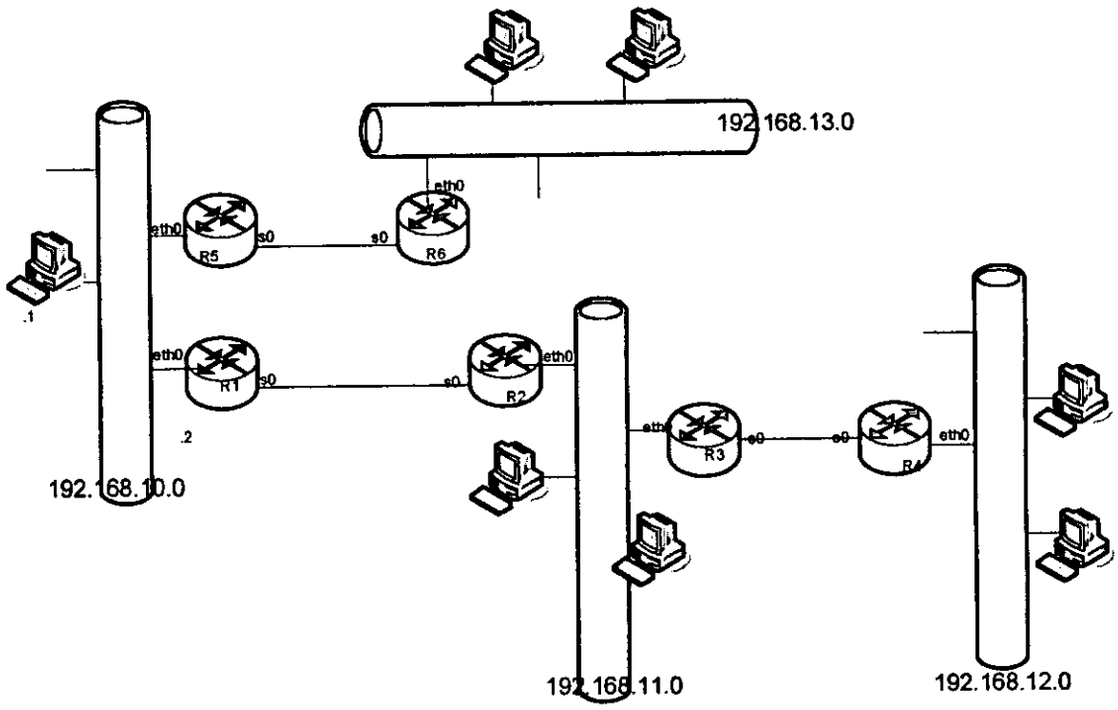

Figura 6-5: Exemplo de redes IPv4, para se gerar tabela de roteamento de R1.

Destino	Gateway	Netmask	Interface
192.168.10.0	R1	/24	eth0
192.168.11.0	R2	/24	s0
192.168.12.0	R2	/24	s0
192.168.13.0	R5	/24	Eth0

Tabela 6-5: Tabela de roteamento referente ao roteador R1.

Tanto os "hosts" como os "gateways" implementam geralmente tabelas de roteamento, as tabelas de roteamento podem ser estáticas ou dinâmicas. Uma tabela estática é definida pelo administrador da rede, sempre que se produzem alterações na topologia da rede as tabelas devem ser actualizadas manualmente. As informações de roteamento podem ser trocadas entre gateways de modo a actualizar dinamicamente as tabelas. Para o efeito usam-se protocolos de roteamento. Os protocolos de roteamento usados dentro das redes terminais são conhecidos por IGP (*Interior*

Gateway Protocols), sendo os mais comuns o RIP (*Routing Information Protocol*), OSPF (Open *Shortest Parth First*) e IS-IS (*Intermediate System-to Intermediate System*). Os protocolos usados nas redes de trânsito são conhecidos por EGP (*Exterior Gateway Protocols*).

Uma entrada importante nas tabelas de roteamento é a *default route* (rota padrão). É muitas vezes definida estaticamente, todos os "datagramas" cuja rede de destino não consta na tabela de roteamento são enviados para o gateway especificado na *default route*.

Designa-se "sistema autónomo" a um conjunto de redes administradas por uma única entidade ou por várias desde que apresente uma estratégia administrativa comum. A informação de roteamento é trocada entre os "gateways" usando um IGP comum, os "gateways" que asseguram a ligação ao exterior (gateways de fronteira) implementam o mesmo IGP do sistema autónomo mais um EGP para troca de informação com o exterior. A informação IGP nunca deve sair para o exterior, por exemplo, se o sistema autónomo usa sub-redes esse facto não transparece para o exterior, a figura 6-6 mostra um exemplo de um sistema autónomo.

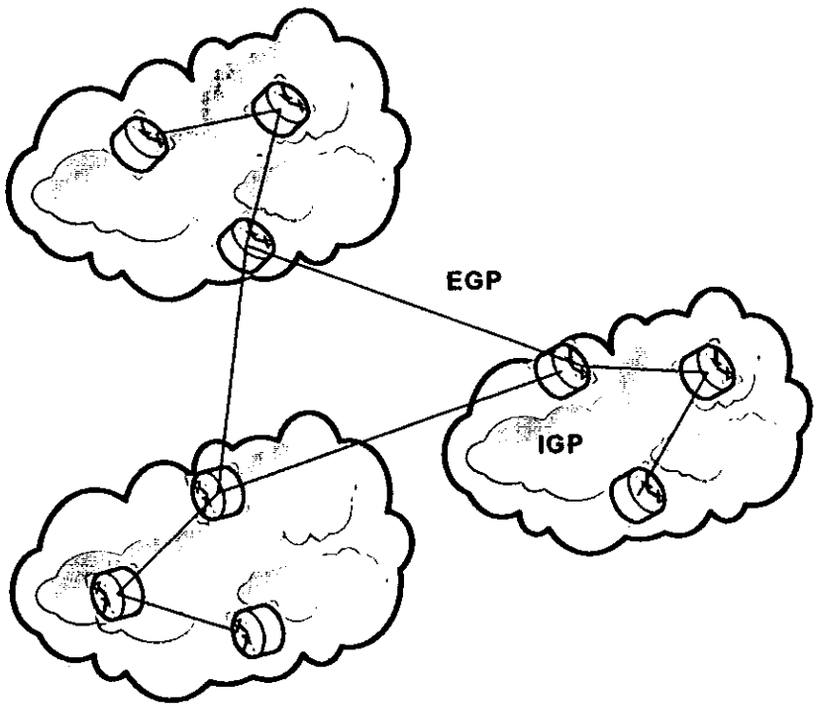

Figura 6-6: Exemplo de Sistema Autónomo

7 Internet Protocol v6

Na década de 80, quando foi descrito como seria o actual protocolo IP (*Internet Protocol*), tinha-se a ideia de que o número de hosts designado seria suficiente para atender uma grande demanda por muito tempo. Isto aconteceu porque, na teoria, se poderia ter até 4.3 biliões de hosts, embora este número possa parecer enorme ele demonstrou ser limitado para cada vez mais dispositivos que se ligavam a rede Internet. Isto eventualmente trouxe problemas como:

1° - Escassez de endereços IPv4;

2°- Habilidade de rotear o tráfego entre as redes em crescimento que compreendem a Internet e

3° - Necessidade de novas funcionalidades (segurança, qualidade de serviço etc.).

O primeiro problema é concernente a eventual escassez de endereços IP. A actual versão do IP (IPv4), define 32 bits de endereçamento, o que significa que existem apenas 2^{32} (4.294.967.296) hosts. Isto pode parecer um número enorme. Mas com abertura de novos mercados e uma porção significativa da população do mundo que torna-se candidata ao endereço IP, viu-se a proximidade de extinção de endereços IP para mais utilizadores que se ligavam à rede.

O problema de poucos endereços é agravado pelo facto de a porção de endereços IP, não ter sido eficientemente alocados, pois, o modelo tradicional de classes (A, B e C), não permite que o número de hosts pertencentes a uma dada classe seja usado no seu potencial máximo.

O segundo problema é causado pelo rápido crescimento das tabelas de roteamento. Os roteadores são requeridos a manter uma informação completa do roteamento para Internet.

Há 10 anos, roteadores apresentaram um crescimento exponencial, com o crescimento de utilizadores conectados a Internet, em Dezembro de 1990 haviam 2.190 roteadores, em Dezembro de 1992, o número cresceu para 8.500 e em Dezembro de 1995 já estavam ligados mais de 30.000 roteadores [7].

Infelizmente o problema de crescimento das tabelas de roteamento não pode ser resolvido simplesmente instalando mais memória no roteador.

O outro problema associado a esta categoria é que o IPv4 usa comprimentos de cabeçalhos variáveis, resultando no aumento do processo de roteamento.

O terceiro problema com IPv4 inclui a necessidade de novas funcionalidades que simplesmente não estão inclusas na versão corrente do protocolo.

Consequentemente ao amadurecimento da *Internet*, há cada vez mais interesse por serviços em tempo real, como a utilização de voz e vídeo, que não são devidamente suportados actualmente, surgindo então a necessidade do IP suportar o *multicasting*. O *multicasting* permite enviar um simples pacote a múltiplos nodos, (num dado grupo), em diferentes segmentos de rede. No modelo actual de unicast, se 5 hosts precisarem de receber um mesmo pacote de dados, o processo de envio do pacote é repetido 5 vezes a partir do pacote endereçado para cada host. No modelo de multicasting, para este caso apenas um pacote é enviado, reduzindo a carga nos roteadores e a largura de banda necessária para suportar a transmissão.

No IPv4, não foi prevista a indicação do nível de importância dos pacotes, como por exemplo a prioridade de um pacote de voz em relação à um pacote de dados ou vice-versa, contudo o uso deste campo nunca foi definido com precisão.

No IPv6 uma capacidade nova é adicionada para habilitar o etiquetamento de pacotes pertencendo a fluxos (flows) particulares para o qual, o remetente requisitou manipulação especial, como qualidade diferente do padrão do serviço ou serviço em tempo real.

A *Internet* não foi desenvolvida pensando-se em segurança. A segurança no IPv4 não é suportada a nível do protocolo, não havendo o suporte nativo à autenticação e à privacidade do tráfego de dados. Não há mecanismos em IPv4 para prevenir que um nodo personifique um outro, processo conhecido como *spoofing*, que representa uma ameaça a segurança em redes baseadas em IP [13].

No IPv6, a segurança foi endereçada para duas fontes: autenticação e encriptação. Os pacotes IPv6 podem ser configurados para autenticarem as suas origens. Isto previne que outros nodos façam spoofing do endereço IP em género de fraude e ganhem acesso a dados sensíveis.

Os pacotes IPv6 podem ser encriptados na camada de rede, eliminando as necessidades que cada protocolo de camada mais acima implemente as suas próprias metodologias de encriptação.

O desenvolvimento do IPv6 pela IETF iniciou em 1990. Ele consiste de um conjunto de protocolos e padrões conhecidos como IPv6, esta versão anteriormente chamada IPng, incorporou os conceitos de vários métodos propostos para actualizar o protocolo IPv4.

A natureza do próprio protocolo permite que este cresça, ou seja, escalado adaptando-se a novas necessidades. Precisamente a escalabilidade é a base fundamental de IPv6 frente ao IPv4.

7.1 História

Em 1991 (ano que começou a se definir uma proposta para o uso da Internet), já existiam 617.000 *hosts* conectados a Internet [16]. Desde 1990 já se sabia da necessidade de aumentar o endereçamento dos números IPs, devido ao grande aumento que estava ocorrendo na Internet e que em pouco tempo se esgotariam os números IPs. Este problema eminente foi resolvido em duas etapas, a primeira foi a redistribuição dos endereços através da proposta do CIDR. Isto deu um tempo a mais para a criação de uma nova proposta de endereçamento [13]. A segunda etapa foi a criação de um novo protocolo que aumentaria os endereços IPs. Então a *Internet Engineering Task Force* (IETF) criou um grupo de trabalho para desenvolver o novo protocolo chamado inicialmente de *Intenet Protocol New Generation* (IPng), o qual iria substituir o protocolo actual que passou a ser denominado *Internet Protocol version* 4 (IPv4). Então, em 1994, após várias discussões e revisões de propostas para o novo protocolo, o grupo de trabalho do IPng decidiu que caminho seguir. A proposta escolhida foi o *Simple IP Plus* (SIPP). O SIPP era uma proposta baseada na união do *Simple* IP (SIP) com o *Paul's* Internet *Protocol* (Pip). A ideia do SIP, proposta por Steve Deering, era de aumentar para 64 bits o endereçamento, além de deixar a fragmentação dos pacotes opcional e a retirada dos aspectos obsoletos do IPv4 [16]. Já a proposta PiP, criada por Paul Francis, tinha uma nova estratégia de roteamento baseada em listas directivas, permitindo uma melhor implantação de políticas de roteamento, facilitando inclusive a implantação de mobilidade. O grupo de trabalho do IPng resolveu também que, ao invés dos 64 bits propostos no SIPP, usaria 128 bits tendo assim 3,4 x 10^{38} endereços possíveis. Considerando

que a Terra possui 6371 quilómetros de raio, teremos então 6,6713599096 x 10^{19} endereços por centímetro quadrado [17].

7.2 Características de IPv6

De seguida são descritas as principais características de IPv6:

- Simplificação do formato do cabeçalho: o cabeçalho IPv6 tem um novo formato para manter o seu *overhead*[5] o mínimo possível, o que é alcançado com a remoção de alguns campos não essenciais e opcionais. Desta forma os pacotes são processados de forma mais eficiente pelos roteadores intermediários;
- Espaço de endereçamento expandido: IPv6 tem endereços IP de origem e destino de 128 bits ou 16 bytes;
- Suporte melhorado para extensões e opções: em IPv4 as opções eram integradas no cabeçalho base. Contudo em IPv6 as opções são consideradas cabeçalhos de extensão. Os cabeçalhos de extensão são inseridos apenas entre o cabeçalho base e a carga útil de dados (payload), se necessários;
- Extensibilidade: esta característica é consequência da anterior, pois o IPv6 pode facilmente incorporar novas funcionalidades com a adição de cabeçalhos de extensão após o cabeçalho base IPv6. Enquanto as opções no cabeçalho IPv4 podem suportar somente 40 bytes, o tamanho do cabeçalho de extensão é limitado apenas pelo tamanho do pacote IPv6;
- Configuração de endereços *stateful e stateless*: para tornar mais simples a configuração de máquinas, o IPv6 suporta configuração de endereços stateful, que necessita de um servidor DHCP e configuração de endereços stateless que ocorre na ausência de um DHCP[6]. Nesta última as máquinas no mesmo elance automaticamente se autoconfiguram

[5] Overhead: conteúdo de um pacote adicional em relação aos dados que se pretendem transmitir.
[6] DHCP (Dynamic Host Configuration Protocol): um protocolo defenido para facilitar autoconfiguração de nodos na rede.

com endereços IPv6 de elance, chamados endereços link-local, e com endereços derivados dos prefixos anunciados pelos roateadores locais;

- Suporte nativo a segurança: o suporte ao IPSec[7] é uma exigência de IPv6, através das extensões de autenticação e confidencialidade;
- Suporte nativo a mobilidade: IPv6 móvel permite roteamento transparente de pacotes IPv6 para nós moveis, tirando vantagens das oportunidades criadas pelo projecto da nova versão de IP.

7.3 Formato do Pacote IPv6

O IPv6 introduz um novo formato de cabeçalho (Figura 7-2). Em oposição à anterior (Figura 6-1) repetida na Figura 7-1 por conveniência, todos os campos deste novo cabeçalho possuem tamanho fixo, totalizando 64 bytes. O facto de esse possuir um tamanho fixo acelera bastante o processamento dos pacotes pelos roteadores, visto que não há necessidade de calcular a extensão de certos campos, e nem o tamanho do cabeçalho como um todo. Além disso, ocorreu uma redução do número de campos utilizados, por meio da exclusão de campos de pouca utilidade prática. Este facto também contribui para a diminuição do tempo gasto em processamento pelos roteadores.

[7] IPSec (***IP Security)***: é um conjunto de padrões utilizados para garantir uma comunicação segura entre dois computadores.

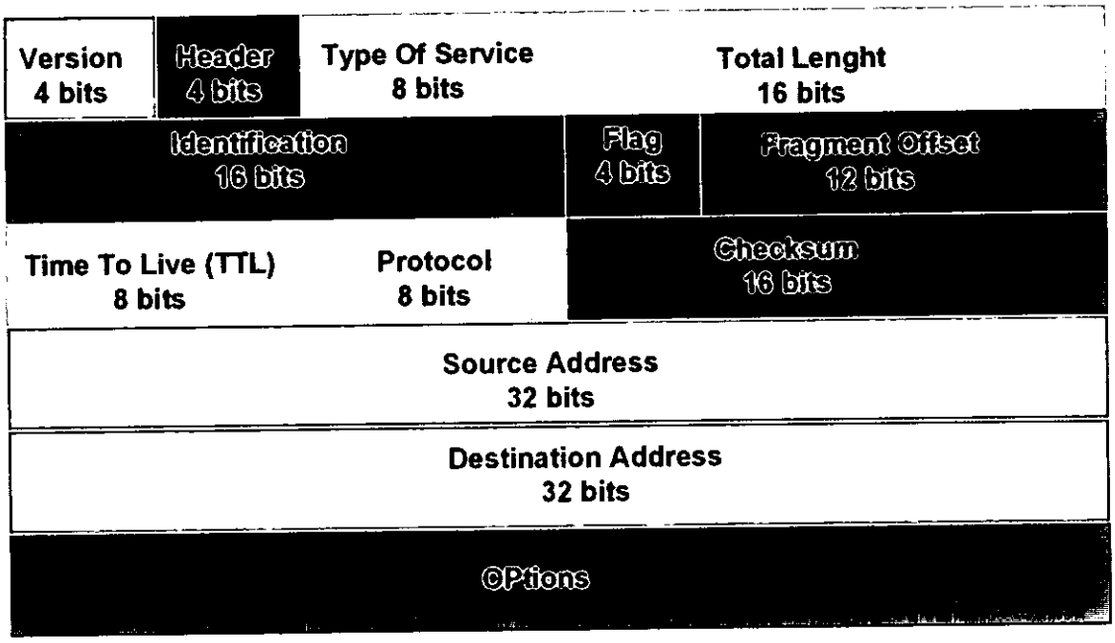

Figura 7-1: Cabeçalho IPv4 com indicação de campos modificados [9].

Na figura 7-1, são marcados os campos que desaparecem e os que são modificados em IPv6 mediante a côr de fundo, os que são modificados aparecem na cor amarela e os que desaparecem na cor vermelha.

O pacote IPv6 contém 8 campos contra 12 existentes no IPv4. O motivo fundamental pelo qual os campos foram eliminados foi a redundância desnecessária.

Em IPv4 mantém-se a mesma informação de várias maneiras. Um caso muito evidente é o Checksum, outros mecanismos de encapsulamento já realizam esta função (IEEE 802 MAC, framing PPP, etc.).

A função do campo *Checksum* era detectar erros que afectassem ao cabeçalho IP, sem detectar no entanto erros no restante pacote. Actualmente a maioria dos erros não são de transmissão, visto que os mecanismos de detenção de erros *Ethernet* e *Point to Point Protocol* (PPP) são bastante eficientes. Como os roteadores só alteram o campo *Hop Limit* (*Time-to-live* no IPv4), estes então terminam por recalcular o *Checksum* antes de retransmitir o pacote, o que pode causar a não detenção de possíveis erros. Além disso, vários roteadores, visando aumento de

performance, não verificavam mais este campo, terminando assim por torná-lo totalmente supérfluo.

No caso do campo Fragment Offset, é ligeiramente diferente, este foi excluído, pois se decidiu que pacotes não seriam mais fragmentados por roteadores. Caso um roteador receba um pacote com tamanho maior que o permitido, não o ingere transmitindo uma mensagem ao *host* que o enviou, comunicando o ocorrido. Este *host* deverá então retransmitir o pacote na forma de pacotes menores. Desta forma há um ganho de desempenho no roteamento, pois é eliminada a necessidade de um roteador fragmentar vários pacotes e, consequentemente, monta-los no destino.

Alguns dos campos foram renomeados:

- Total Length passa para Payload Length, que é definido pelos bytes que se seguem ao cabeçalho de 40 bytes. A razão da alteração deste campo deve-se a pequena modificação que ocorreu: os 40 bytes do cabeçalho deixam de ser contados como parte do tamanho, como acontecia até agora.
- Protocol passa para Next Header, dado que em vez de usar cabeçalhos de comprimento variável, usa sucessivos cabeçalhos encadeados, desse modo desaparece o campo options.
- Time to Live passa para Hop Limit, tem um tamanho de 8 bits (1 byte). Cada vez que passa por um nó é decrementado o valor do hop limit, o pacote é descartado quando o valor do hop limit for zero. O objectivo deste é evitar que os pacotes tenham vida eterna no seu percurso. No IPv4 o campo denotava o tempo em segundos, mas nenhum roteador o utilizou dessa forma. Portanto, seu nome foi alterado para reflectir o modo em que ele de facto é usado.

Os campos que então constituem o cabeçalho IPv6 são:

- Traffic Class, também denominado Priority, ou simplesmente Class. Poderia ser mais ou menos equivalente ao Type of Service em IPv4. A sua função é de distinguir os pacotes cuja a origem pode ter controle de fluxo daqueles que não podem ter.

Os valores de 0 a 7 são reservados para transmissões que podem ter sua velocidade reduzida diante de um congestionamento. Os valores de 8 a 15 são destinados ao tráfego em tempo real e cuja a taxa de transmissão é constante. Essa distinção permite que os roteadores ofereçam um melhor tratamento aos pacotes quando ocorre um congestionamento. Dentro de cada grupo, os pacotes com um número mais baixo são menos importantes do que os que têm números mais altos. O padrão sugerido para, por exemplo informação seria o uso de 1, 4 para FTP o uso de 6 para conexões Telnet, pois dificilmente percebe-se o retardo de alguns segundos sofridos por um pacote que transporta informações, ao contrário do que acontece a um retardo de um pacote de Telnet.

- *Flow Label,* para permitir tráfego com requisitos de serviços a tempo real. Tem comprimento de 20 bits. Identifica, com os campos *Source Address* e *Destination Address*, o fluxo ao qual o pacote pertence.

Estes campos são os que dão suporte à uma das características fundamentais e intrínsecas de IPv6: qualidade de serviço (QoS), Classe de Serviço (CoS), e por último um poderoso mecanismo de controle de fluxo, de acesso de prioridades diferenciadas de acordo com o tipo de serviço.

Portanto para o pacote IPv6 o cabeçalho tem o seguinte formato:

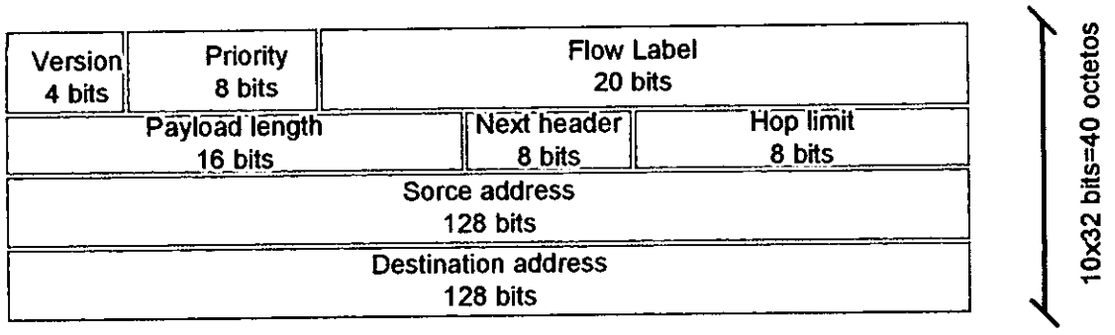

Figura 7-2: Cabeçalho IPv6 [17].

O comprimento do cabeçalho é de 40 bytes, constituindo o dobro de IPv4, no entanto levando muitas vantagens com a eliminação de campos redundantes.

O comprimento fixo do cabeçalho implica uma maior facilidade para o seu processamento em roteadores e comutadores. O facto dos campos estarem alienados a 64 bits, permite que novas gerações de processadores de 64 bits possam processar duma maneira mais eficaz o cabeçalho IP.

7.3.1 Cabeçalhos de Extensão

Todo o *pacote* IPv6 começa com um cabeçalho básico, como foi visto anteriormente. Nem sempre a informação contida no cabeçalho é suficiente para transportar um pacote, por isso é necessário adicionar um ou mais cabeçalhos de extensão. A finalidade de um cabeçalho de extensão é de acrescentar maior informação sobre o pacote, para que seja processado adequadamente, tanto no destino como nos roteadores [15]. Todo o cabeçalho de extensão possui um identificador, como pode ser visto na tabela 7-1.

Este identificador é referenciado no campo *Next Header* do cabeçalho IPv6, sendo que ele contém o valor do primeiro cabeçalho que irá aparecer após o cabeçalho IPv6. Caso não exista nenhuma extensão é usado no campo *next header* o valor 59, indicando que não existe nenhum cabeçalho de extensão[15].

Identificador	Cabeçalho de Extensão
0	Hop-by-Hop Options
43	Routing Information
44	Fragment
51	Authentication Header
50	Encapsulating Security Payload Header
59	No Next Header
60	Destination Options Header

Tabela 7-1: Cabeçalho de extensão.

Na primeira coluna da tabela 7-1 é apresentado o valor que identifica o cabeçalho estendido e na segunda o cabeçalho que é representado. A tabela mostra ainda a sequência preferencial de uso dos cabeçalhos, não sendo obrigatório o uso nesta ordem, a não ser o *Hop-by-Hop Options* que é indispensável aparecer logo após o cabeçalho IPv6, quando este for necessário [15]. A importância de seguir a ordem é para deixar mais eficiente o transporte do pacote, Uma das consequências de seguir esta ordem é facilitar o processamento dos roteadores, evitando que cada roteador tenha que analisar todos os cabeçalhos até encontrar um cabeçalho que contenha informações úteis para o processamento do pacote no roteador. Os dois cabeçalhos que importam para um roteador são o *Hop-by-Hop Options* e o de roteamento [16].

Fica claro que não é necessário que se tenha todos os cabeçalhos presentes em um datagrama IPv6, e eles podem aparecer mais de uma vez no pacote. O único cabeçalho que pode aparecer no máximo duas vezes é o *destination option*, que aparece em dois lugares distintos, depois do *hop-by-hop* e antes do protocolo acima ao IPv6. Exemplos destes protocolos são: de transporte, como o TCP e o UDP, de controle, como o ICMP, e o de rotemento como o OSPF [5].

7.3.2 Hop-By-Hop Options

O cabeçalho *hop-by-hop* é utilizado para carregar informações opcionais, e deverá ser examinado por todos os roteadores pelo qual o pacote passa. Quando houver a existência deste cabeçalho, o

mesmo deverá aparecer logo após o cabeçalho IPv6 [15]. Já que ele é o único cabeçalho a ser examinado por cada nó intermediário.

7.3.3 Routing Information (Informação de Roteamento)

É usado para fornecer uma lista de um ou mais nós intermediários que devem ser vistos no caminho do pacote até ao destino.

7.3.4 Fragment

É usado para serviços de fragmentação e remontagem de pacotes. No caso do pacote a ser enviado ser maior que o MTU (Maximum *Transmission Unit*) suportado, então o nó origem fragmenta o pacote, já que em IPv6 somente os nós origem podem fragmentar.

7.3.5 Destination Option

É usado para especificar partes opcionais que são examinadas pelos nós intermediários ou pelo destino final, o que dependerá da posição onde ele estiver na ordem dos cabeçalhos de extensão.

Os campos **Authentication** e **Encapsulation Security Payload** serão explicados no capítulo da segurança.

7.4 Endereçamento em IPv6

O espaço de endereçamento do IPv6 é de 2^{128}, o que equivale a $3{,}40 \times 10^{38}$ (340.282.366.920.938.463.463.374.607.431.768.221.456) endereços possíveis. Isto pode ser dito por outras palavras na medida em que este valor é impronunciável, que por

centímetro quadrado da superfície terrestre teria-se nada mais nada menos do que 6,6713599096 x 10^{19} endereços IPv6.

O IPv6 é representado por 8 campos de endereço de 16 bits (em forma hexadecimal), separados por dois pontos, como mostra o exemplo: X:X:X:X:X:X:X:X onde X representa um valor hexadecimal de 16 bits da porção correspondente ao endereço IPv6. Os zeros que aparecem à esquerda de cada campo podem não ser escritos. dado o número de endereços existentes, poderiam existir largas cadeias de bits zero, deste modo a sua representação é abreviada para "::", representando múltiplos grupos consecutivos de 16 bits zero. Este campo deve aparecer uma só vez no bloco de endereço IPv6.

Exemplo os endereços `1080:0:0:0:8:800:200C:417A`, representando um endereço unicast, `0:0:0:0:0:0:0:1`, representando o loopback e `0:0:0:0:0:0:0:0`, um endereço não especificado, podem ser representados como: `1080::8:800:200C:417A`, `::1` e `::` respectivamente.

Uma outra forma alternativa e conveniente, é quando se trata de um ambiente misto IPv6 e IPv4, o que seria: x:x:x:x:x:x:d.d.d.d, onde x representa valores hexadecimais de 16 bits e o d representa valores decimais de 8 bits cada (representação padrão de IPv4).

Esta forma é usada preferencialmente por nodos com IPv6 e que recebem pacotes IPv4, ou transmitem pacotes IPv4 para nodos que só tenham o protocolo da versão 4. Esta concatenação é formada por um prefixo de 96 bits de tamanho, com representação do IPv6 mais 32 bits da forma de representação do IPv4 [6]. Um exemplo seria:

os endereços

`0:0:0:0:0:0:13.1.68.3`

`0:0:0:0:0:FFF:129.144.52.38`

também podem ser representados como:

`::13.1.68.3`

`:: FFF:129.144.52.38`

A representação de prefixos em IPv6 segue a seguinte estrutura

Endereço IPv6/ comprimento do prefixo

Onde:

- Endereço IPv6 corresponde ao endereço IPv6 propriamente dito em qualquer notação válida;
- Comprimento do prefixo, valor decimal indicando quantos bits contíguos da parte a esquerda do endereço compõem o prefixo.

Por exemplo a representações válidas de prefixo de 60 bits `12AB00000000CD3` serão:

`12AB:0000:0000:CD30:0000:0000:0000:0000/60`

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

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

7.4.1 Arquitectura de Endereçamento

A figura 7-3 representa a arquitectura do endereço IPv6 onde:

FP= 001 (Format Prefix)- é usado para identificar os endereços globais unicast.

TLA ID= 0x1FFE (Top-Level Aggregation Indentifier)- Este campo é obtido através de IANA[8] para efeitos de teste de backbone.

O uso deste campo é temporário e todos os utilizadores deste endereço deverão renomeia-lo em algum tempo do futuro.

NLA ID= Next Level Aggregation Identifier - é tido através do administrador TLA ID na hierarquia de endereçamento suficiente para identificar redes em trânsito e os utilizadores finais

[8] IANA (Internet Assigned Numbers Authority) : Organização responsável para fornecer blocos de endereço IP aos ISPs, outros exemplos deste tipo de organizações sao: (RNP, NIC, NCC etc).

com arquitectura e topologia de backbone, providencia um trânsito com múltiplos níveis com objectivos de backbone para testes completos de IPv6.

SLA ID- (Site Level Aggregation Identifier) - é usado para organizações individuais com vista a criarem sua conexão local (hierarquicamente) e para identificar sub-redes, o acesso a este endereço é responsabilidade de cada organização individual.

Interface ID- É o identificador de interface de ligação sendo definido num apropriado IPv6 over Link como ethernet, FDDI etc.

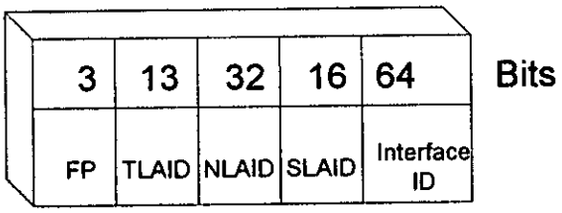

Figura 7-3: Arquitetura de endereçamento IPv6 [17].

7.4.2 Diferenças com IPv4

Há algumas diferenças importantes no endereçamento IPv6 com relação ao IPv4 que é importante referir:

- Não há endereços broadcast (sua função é substituída por endereços multicast);
- Qualquer campo pode conter só zeros ou só uns, salvo que explicitamente se prove o contrário [16];
- Todas interfaces têm que ter, pelo menos um endereço unicast *link-local*;
- Uma única interface pode ter vários endereços IPv6 de qualquer tipo (unicast, anycast e multicast);
- No IPv4 associa-se um prefixo de sub-rede com um endereço MAC,no IPv6 podem se associar múltiplos prefixos de sub-rede a um único MAC.

7.4.3 Endereços especiais em IPv6

Os endereços especiais ou reservados são endereços para algum tipo de função necessária para o controle e ou a manutenção da rede. Portanto, ficou definido que todos os endereços que iniciem com a sequência de bits "00000000" ou com oito bits mais significativos do endereço IPv6 zeros, são reservados ou especiais. Assim, existem 120 bits reservados para alguma funcionalidade especial. As funções de alguns destes endereços serão especificados a seguir.

7.4.3.1 Loopback

No IPv4, existe o endereço `127.0.0.1`, que é chamado de endereço de *loopback*. Sua função é fazer com que seja transmitido algo para si mesmo. Este tipo de endereço serve para realizar testes de softwares. No IPv6, o endereço reservado para esta função é `0:0:0:0:0:0:0:1` ou `(::1)`, não é associado a uma interface físico; trata-se de uma interface virtual, pode ser usado para testes de verificação de correcta inicialização de um protocolo numa máquina.

7.4.3.2 Endereço não especificado

Quando um endereço for 0:0:0:0:0:0:0:0 ou ::, ele é chamado de endereço não especificado ou inválido. Uma das finalidades deste tipo de endereço é seu uso em transmissões *multicast*, por causa de algumas restrições onde não se deve especificar o endereço origem da transmissão *multicast*. Um exemplo disto é o cabeçalho estendido de roteamento do tipo zero. Nunca deve ser atribuído à nenhum nodo, pois se usa para indicar a ausência de endereços, trata-se de host que está a ser inicializado antes de ter o seu próprio endereço.

7.4.3.3 Endereço de Translação para IPv4

Como existe a ideia do IPv6 substituir aos poucos as redes IPv4, foi criado um mecanismo para fazer dinamicamente, o tunelamento de pacotes IPv6 para pacotes IPv4. Os nodos que usam esta técnica têm um endereço *unicast* especial, que carrega nos 32-bits menos significativos um endereço IPv4. Os restantes dos 96 bits são representados por zeros. Esta técnica é chamada de IPv4, compatível com endereço IPv6 [1]. O endereço `0:0:0:0:0:0:10.16.169.1` (ou `::10.16.169.1`) é um exemplo dessa técnica. Existe também um segundo tipo de endereço de translação que é o mapeamento do IPv4 para endereço IPv6. Ou seja, serve para representar nodos que não suportam endereços IPv6. Foi definido que este tipo de endereço possui os 80 primeiros bits mais significativos valores zero. Os 16 bits seguintes têm o valor um e os restantes 32 bits representam um endereço IPv4 [16]. Um exemplo deste tipo de mapeamento é representado pelo endereço `0:0:0:0:0:FFFF:10.16.169.1`, ou

simplificando `::FFFF:10.16.169.1`.

7.4.3.4 Endereço de Agregação Global

Para se ter uma melhor organização na distribuição do endereçamento, por parte dos órgãos responsáveis (como o IANA, RNP, NIC, NCC), foi adoptado então algo semelhante ao CIDR (*Classless Interdomain Routing*) do IPv4 [14]. A ideia do CIDR era de eliminar a separação entre a parte do endereço IP que identifica a rede e a parte que identifica a máquina na rede, passando a ser necessário que em cada endereço seja informada a quantidade de bits usados para cada finalidade, ou seja, é atribuído um prefixo para a rede [14]. Já a proposta criada para a distribuição de endereços IPv6 designa um prefixo, ou uma faixa de endereços, para um provedor base. Este prefixo é a base para fazer o roteamento na Internet, ou seja, a decisão de repassar um pacote para uma rede é baseada no prefixo do endereço. Desta forma não existe a necessidade de se manter grandes tabelas de rotas e nem de conhecer a topologia da rede [14]. A organização do endereço de agregação golobal possui três níveis de hierarquias, que são:

- Topologia pública: formada por provedores que fazem a troca de informações entre si;
- Topologia local: onde existe a troca de tráfego dentro de um provedor;
- Identificador da interface: é a identificação da máquina, enlace.

7.4.3.5 Endereços Unicast

Um endereço unicast é tido como um identificador para um único interface. Um pacote enviado a um endereço unicast é entregue a apenas um interface identificado com esse endereço.

Os endereços unicast são agregáveis com máscaras de bits contíguos similares ao caso IPv4 com CIDR. Há várias formas de atribuição de endereços unicast, e algumas podem ser definidas no futuro [17].

7.4.3.6 Endereço anycast

Um endereço IPv6 anycast é atribuído a mais de um interface, tipicamente pertencendo a hosts diferentes, sendo que um pacote enviado a esse endereço será entregue a interface mais próxima, de acordo com os protocolos de roteamento.

Este tipo de endereçamento pode ser usado por um nodo para determinar a rota pela qual ele quer que seus pacotes trafeguèm. Por exemplo, ele poderia seleccionar por quais provedores seus pacotes podem passar. Essa capacidade pode ser implementada através da configuração de endereços anycast que identifiquem um conjunto de roteadores pertencentes a esses provedores. Assim os endereços dos provedores confiáveis podem ser citados como endereços intermediários no cabeçalho de roteamento.

Os endereços anycast são alocados a partir dos endereços unicast, o unicast é atribuído a mais de um nodo, que deve ser configurado para ser anycast.

7.4.3.7 Endereço multicast

Os endereços Multicast identificam um grupo de interfaces. Uma interface pode pertencer a qualquer número do grupo multicast. Endereços deste tipo têm o formato mostrado na tabela 7-2.

8 bits	4 bits	4 bits	112 bits
11111111	FLAGS	SCOP	GROUP ID

Tabela 7-2: Formato de endereços multicast

11111111 identifica o prefixo. Os bits que seguem são para flags. Actualmente os 3 primeiros bits estão reservados para o uso futuro, estando os bits menos significativos com as seguintes funções:

- 0 – indica um endereço permanente;
- 1 – indica um endereço provisório.

O SCOP é um valor de 4 bits usado para limitar o escopo E identifica todo o planeta Terra. O escopo F já está reservado para o escopo que identifica a galáxia, ou sistema solar, dentro de uma futura expansão da Internet. Os valores podem ser vistos na tabela 7-3.

0	reservado
1	Nodo local
2	link-local
3	não usado
4	não usado
5	site-local
6	não usado
7	não usado

8	organização local
9	não usado
A	não usado
B	não usado
C	não usado
D	não usado
E	global
F	reservado

Tabela 7-3: Endereços IPv6 reservados.

GROUP ID identifica o grupo *multicast* sendo permanente ou não, dentro do escopo dado

A missão deste tipo de endereçamento é adequado a transmissões múltiplas (*broadcast*).

7.4.3.8 Endereços de Uso Local

Para o uso de redes locais, foram definidos dois tipos de endereços *unicast*: um chamado de enlace local ou *link-local*, e outro chamado de *site-local*. Estes dois tipos de endereços não são conectados à Internet. Eles só servem para uso local, ou seja, corporações que não estão conectadas à Internet. O primeiro tipo de endereço (enlace local ou *link-local*) tem como propósito o uso em pequenas redes que não possuem roteadores. É usada também para a autoconfiguração de endereços, O segundo tipo de endereço, o *site-local* é utilizado para redes de computadores locais que possuem roteadores.

7.5 Autoconfiguração em IPv6

A capacidade de autoconfiguração de endereços IPv6 foi projectada para assegurar que a configuração manual de máquinas, antes de conecta-las à rede, não seja necessária. Autoconfiguração será uma característica chave de IPv6 quando todo tipo de equipamento, tais como televisores, DVD players, refrigeradores e telefones móveis usarem um endereço IPv6 [6].

A auto configuração é um conjunto de passos pelos quais um host decide como configurar seus interfaces em IPv6. Esta operação é também conhecida como operação plug-and-play de máquinas na Internet.

Quando uma máquina é ligada, deve automaticamente associar um endereço IP á sua interface de rede. No caso de IPv4, esta associação é manual ou é através de um servidor DHCP e o default gateway.

Pode-se imaginar uma rápida e fácil transição de uma rede a outra ao se tratar de dispositivos wireless.

São duas as formas de um host se auto-configurar com IPv6 (com estado-stateful ou e sem estado- stateless) e seja qual for a forma usada para autoconfiguração o host logo que inicializa as suas interfaces obtém um endereço link-local para cada interface IPv6.

Para o host se autoconfigurar ele irá necessitar de um prefixo ou seja o prefixo de link-local, fe80::/64 e também um valor único de 64 bits para identificar a interface. O maior problema de se conseguir este valor é que a máquina ainda não possui informação sobre quem está conectado na sua rede e por isso deve-se usar algum método para definir este valor e que garanta que o mesmo não seja repetido.

Para resolver este problema foi proposto o uso de EUI-64[9] um padrão definido pela IEEE que supostamente é um valor único e é obtido através dos 48 bits do padrão de endereçamento IEEE 802 que formam o endereço MAC.

Autoconfiguração Stateless: A autoconfiguração sem estado (*stateless*), é utilizada em redes onde não exista a necessidade de distribuir endereços de forma exacta ou padronizada para cada *host,* não há necessidade de se instalar um servidor DHCP na rede para fins de configuração. Para uma máquina se autoconfigurar, ela inicialmente irá obter um endereço de link-local, que é obtido combinando o prefixo fe80:: com o endereço MAC do interface de rede. Após obter este endereço, irá enviar uma mensagem de *router solicitation* (solicitação de roteador), ou seja, irá

[9] EUI-64: endereço de 8 bytes defenido pela IEEE , consiste na transformação do endereço MAC de 48 bits para 64 bits.

enviar um pacote *multicast* para o endereço FF02::2. Este é o endereço no qual todos os roteadores da rede devem escutar. Quando um roteador recebe esta mensagem, ele deverá responder com uma mensagem de *router advertisement*, que conterá o prefixo da rede.

Auto configuração Stateful: este tipo de configuração requer alguma configuração na máquina e também necessita do uso de servidor um exemplo deste tipo de configuração é através do servidor DHCP.

7.6 Qualidade de Serviço

No cabeçalho IPv6 os campos Flow Label e Priority, são usados para identificar aqueles pacotes que necessitam de "cuidados especiais". São pacotes originados em aplicações multimédia ou de tempo real, por exemplo.

Flow Label: são 24 bits que devem ser usados para identificar um tipo de fluxo de dados (uma conexão ou um circuito virtual).

O uso deste campo não é explicitamente definido, mas imagina-se que um fluxo orientado, necessita de uma atenção maior que o fluxo não orientado. Deixa-se a cargo dos roteadores a decisão sobre que medida tomar.

Dentro de cada categoria (orientada ou não), haveria um identificador de fluxo que iria dar a sugestão no tratamento daquele caso. Quando os roteadores recebessem um pacote com determinado identificador de fluxo, consultariam a uma tabela onde se iria recuperar o tipo de tratamento.

Prioridade: este campo determina a prioridade do pacote com relação a outros. Todos os pacotes de um determinado fluxo devem ter a mesma prioridade, portanto estes são dois campos usados em conjunto. Espera-se que esse campo identifique e priorize aplicações interactivas, como sessão remota.

O uso efectivo dá-se quando o pacote enfrenta um tráfego congestionado. Valores de 0 a 7 neste campo lidam com transmissões (geralmente TCP) que podem ser retardadas no caso de um

congestionamento. Valores de 8 a 15 referem-se a aplicações cujo o tráfego é constante e um atraso implicaria em perda de informação, como vídeo e áudio.

7.7 Mobilidade

Actualmente, já se convive com alguma mobilidade provida por protocolos das camadas físicas e de enlace de dados. Um bom exemplo disto é o já popular IEEE 802.11 (Ethernet sem fio). Contudo, a mobilidade, neste caso, existe apenas em âmbito local, sendo impossível que uma unidade móvel se desloque entre redes diferentes, conservando, portanto, sua configuração de rede inalterada durante a movimentação.

Mobilidade IP (MIP), por outro lado, possibilita que um nó móvel passe de uma rede para outra sem que as conexões/sessões estabelecidas sejam interrompidas, permitindo que outras novas sejam estabelecidas.

Como resultado do desenvolvimento dos diferentes padrões para comunicação em redes *wireless*, com o surgimento de equipamentos portáteis com mais recursos computacionais e com utilização de técnicas de compressão e transmissão capazes de trazer um significativo aumento na banda disponível na interface aérea, há uma forte pressão de usuários e fornecedores de serviços de telecomunicações por recursos que ofereçam suporte nativo na camada de rede à mobilidade. Além disto, mobilidade IP está sendo vista como a melhor forma de interconectar as diferentes tecnologias de redes *wireless* (IEEE 802.11, GPRS, HiperLan, etc.), entre si e com as tradicionais redes cabeadas [16].

7.7.1 Mobilidade IP e seu funcionamento

Imagine que um dispositivo móvel inicie, por exemplo, uma conexão FTP e, no meio da transmissão, o nó móvel muda de rede. Para manter a conexão do FTP na camada de transporte, é preciso manter o mesmo endereço IP. Mudando o endereço IP, a conexão é desfeita.

Por outro lado, a entrega de pacotes para o ponto de conexão corrente do nó móvel depende do número de rede contido em seu endereço IP. Quando o nó móvel muda de rede, receberá um novo endereço IP e isto significa que haverá uma mudança no roteamento dos pacotes enviados a ele.

O Mobile IP (MIP ou MIPv4) foi projectado para resolver este problema, permitindo que um nó móvel tenha dois endereços IP, denominados *home address* e *care-of address*. O *home address* é estático e referenciado, por exemplo, para identificar conexões da camada de transporte (por exemplo, TCP). O *care-of address* muda a cada novo ponto de conexão e pode ser visto como endereço de significado topológico do nó móvel. O *care-of address* indica o novo ponto de conexão do nó móvel.

A solução Mobile IP para IPv4 conta com dois elementos:

- *home agent* (HA): é um roteador na rede de origem do nó móvel;
- *foreign agent* (FA): é um roteador na rede onde o nó móvel está momentaneamente conectado.

Estando o nó móvel fora de sua rede, teremos quatro possíveis participantes em uma comunicação: o nó móvel (MN); o nó correspondente (CN); o home agent (HA); e o foreign agent (FA). Neste caso, o nó móvel deve adquirir um care-of address (possivelmente com o foreign agent) e registrá-lo com o Home Agent. O HA realiza o tunelamento de mensagens enviadas pelo nó correspondente ao nó móvel, enquanto este estiver fora de sua rede. A figura 16 apresenta a arquitectura Mobile IP.

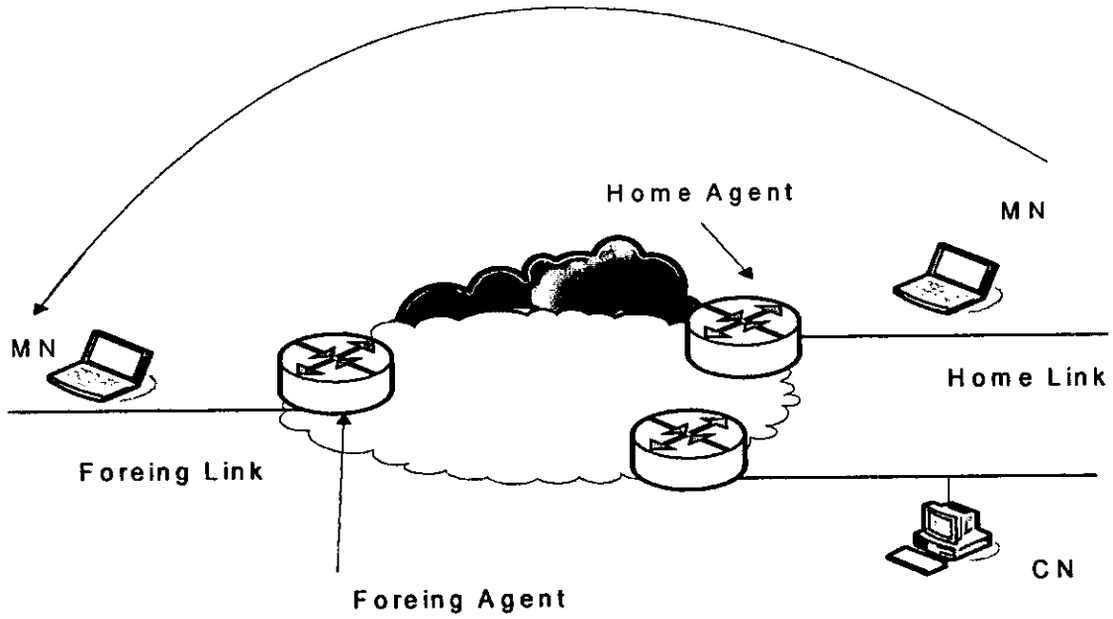

Figura 7-4: Arquitectura Mobile IP [13].

Percebe-se, com a descrição acima, que o funcionamento do *Mobile IP* (MIP) gera um "roteamento triangular", ou seja, um nó correspondente, conhecendo apenas o *home address* do nó móvel, enviará os pacotes para a rede original do nó móvel. Porém, como o nó móvel se moveu, o *home agent* intercepta os pacotes e "tunela" para o nó móvel em seu *care-of address*, ou seja, envia o pacote para a rede em que o nó móvel está momentaneamente.

Este facto é um dos problemas do MIP, já que todos os pacotes serão tunelados para o nó móvel em outra rede, o que gera sobrecarga de processamento no *Home Agent* (HA). O MIPv6 soluciona este problema através de optimização de rota (algumas soluções para optimização de rota no MIPv4 também foram propostas, mas actualmente, este item não é de interesse do IETF [4].

A possibilidade que um nodo tem de manter o mesmo endereço IP, independentemente do ponto onde ele estiver, é outra motivação de IPv6. Em IPv4 foram iniciados trabalhos para este efeito, mas as complicações para usar a mobilidade têm sido enormes [4].

A ideia básica permite identificar com seu endereço (*home address*), independentemente do seu ponto de conexão à Internet. Quando não estiver no seu ponto de origem ou de partida, também estará associado a informação que permite identificar sua posição do endereço actual (*care of address*). Os pacotes enviados a um nodo móvel são transparentemente encaminhados ao seu endereço actual.

O exemplo que se pode imaginar para se entender este tipo de redes é a (habitual rede de telefonia móvel), onde o nó móvel neste caso pode estar conectado simultaneamente a várias redes, e deve ser alcançável por qualquer uma delas.

Existe um protocolo que permite que os nodos IPv6 alcancem a informação de vínculo entre o endereço de partida e a posição actual, neste caso são capazes de enviar os pacotes destinados, ao nó móvel directamente ao seu endereço actual.

7.7.2 Segurança (autenticação e encriptaçao)

Actualmente, com a crescente utilização da Internet para fins financeiros, a preocupação com segurança é cada vez maior. Cada vez mais, bancos disponibilizam serviços de *Home Banking*, e empresas vendem seus produtos on-line. Além disso, as pessoas querem ter privacidade ao utilizar a Internet.

Baseados nestes e outros problemas relativos a segurança, os desenvolvedores do IPv6 resolveram incluir facilidades de segurança neste protocolo, implementando então segurança a nível da camada de rede. Isto elimina a necessidade de implementação de mecanismos de segurança nas camadas superiores, em particular na camada aplicação.

Esta segurança é implementada por dois mecanismos:

- Authentication Header (AH) - Com este método, o cabeçalho é autenticado, garantindo assim a identidade do rementente, e que o pacote não foi alterado em tráfego. A informação pertinente é armazenada em um *Authentication Header*, que é um dos possíveis tipos de cabeçalho de extensão.

- Encrypted Security Payload (ESP) - Este método criptógrafa os dados enviados (todo o *payload*), e armazena as informações pertinentes em um ESP (outro tipo de cabeçalho de extensão). Assim, é possível garantir que caso a informação transmitida seja interceptada por pessoas não autorizadas, estas serão incapazes de compreendê-la, garantindo assim privacidade.

Estes dois métodos podem ser utilizados em conjunto, a fim de fornecer tanto autenticidade quanto privacidade.

Deste modo, garante-se segurança em três aspectos muito importantes, tendo a certeza de:

- Quem nos enviou determinada informação;
- Que está não foi adulterada por terceiros, correspondendo a informação realmente enviada;
- Que a informação não foi lida por terceiros.

Estes mecanismos de segurança podem ser adicionados ao IPv4, através de protocolos de segurança como o IPSec (*IP Security*), no entanto o IPv6 apresenta a grande vantagem de já possuí-los nativamente.

Além do que já foi exposto, este protocolo é independente do algoritmo utilizado para criptografia, permitindo uma maior flexibilidade e segurança, visto que pode-se periodicamente evoluir para um método criptográfico mais seguro.

A utilização deste protocolo propicia em muito a implementação de Redes Virtuais Privadas ou VPN's (*Virtual Private Networks*), que são basicamente a criação de redes lógicas, utilizando as infra-estruturas de redes físicas já existentes. Estas redes apresentam muitas vantagens em termos de custo e praticidade de uso. No entanto, para a efectiva implementação destas é indispensável que haja segurança, por exemplo, a nível de privacidade de dados e controle de acesso.

A utilização do IPv6 propcia a criação destas redes na medida em que uma vez que os dados "abaixo" da camada IP são criptografados e autenticados, praticamente todo o problema de privacidade de dados que as VPN's exigem já está resolvido.

8 UEM

A Universidade Eduardo Mondlane (UEM) foi fundada a 1 de Maio de 1976, pelo então presidente Samora Machel. Ela resulta de Estudos gerais Universitários fundada a 21 de Agosto de 1962, que foram promovidos á Universidade de Lourenço Marques em 1968.

A Universidade Eduardo Mondlane é uma instituição do ensino superior, público que tem se empenhado em ser uma instituição de excelência no contexto da educação, da ciência, da cultura e da tecnologia, educando para a vida os profissionais que capacita e assumindo responsabilidades no processo de inovação e transferência de conhecimento e no desenvolvimento sustentado [8].

A UEM, é a maior Universidade do País, sediada na cidade de Maputo, actualmente com cerca de 17 unidades académicas incluindo (ESHTI[10]) tendo cerca de 12 faculdades, e ainda com cerca de 2500 funcionários e 12 000 estudantes [18].

Fazem parte desta Universidade O Campus Universitário, Reitoria, O Campus das Engenharias, e outras faculdades que se encontram localizadas fora do Campus principal, como a faculdade de Arquitectura, Faculdade de Medicina, Faculdade de Veterinária, Faculdade de Direito.

O Centro de Informática da Universidade Eduardo Mondlane (CIUEM) é a instituição, dentro da Universidade que vela pelas comunicações, sistemas de informação e providencia alguns serviços aos funcionários da UEM, estudantes e alguns particulares. Os serviços disponibilizados são o acesso a Internet, Web hosting, E-mails, DNS, Webmail, Sistemas de informação e desenho de redes assim como sua instalação.

8.1 Caracteristicas de hardware/ Software e comunicações

O campus Universitário e algumas unidades no Campus Universitário tém as suas redes locais seguindo uma topologia estrela (10/100), interligadas por meio de uma fibra óptica formando

[10] ESHTI: Escola Superior de Hotelaria e Turismo de Inhambane.

uma topologia em árvore, actualmente com capacidade de 100 Mbps em fase de migração para 1Gbps. A figura 8-1 ilustra o esquema da rede de dados da UEM.

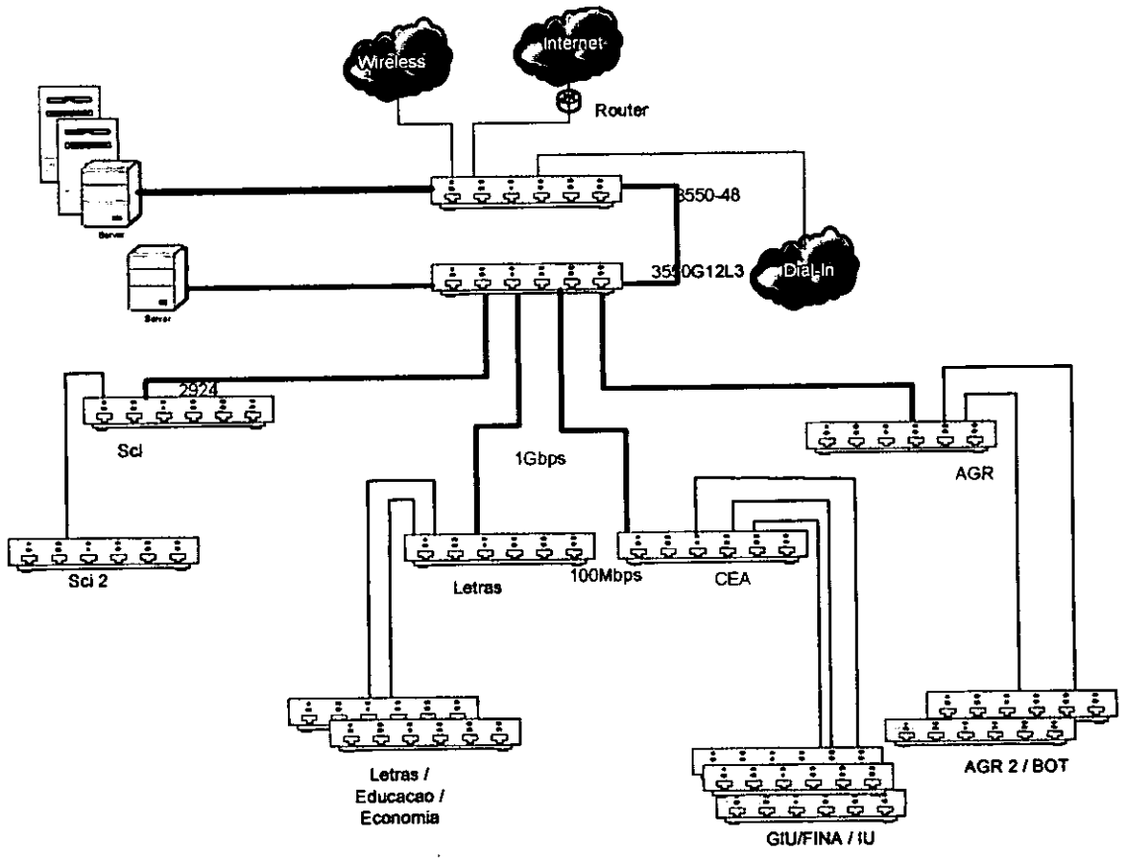

Figura 8-1: Rede no Campus Principal.

No Campus das Engenharias as LANs seguindo uma topologia estrela (10/100), encontram-se interligados por meio de uma fibra óptica seguindo a mesma topologia e capacidade do Campus Principal figura 8-2.

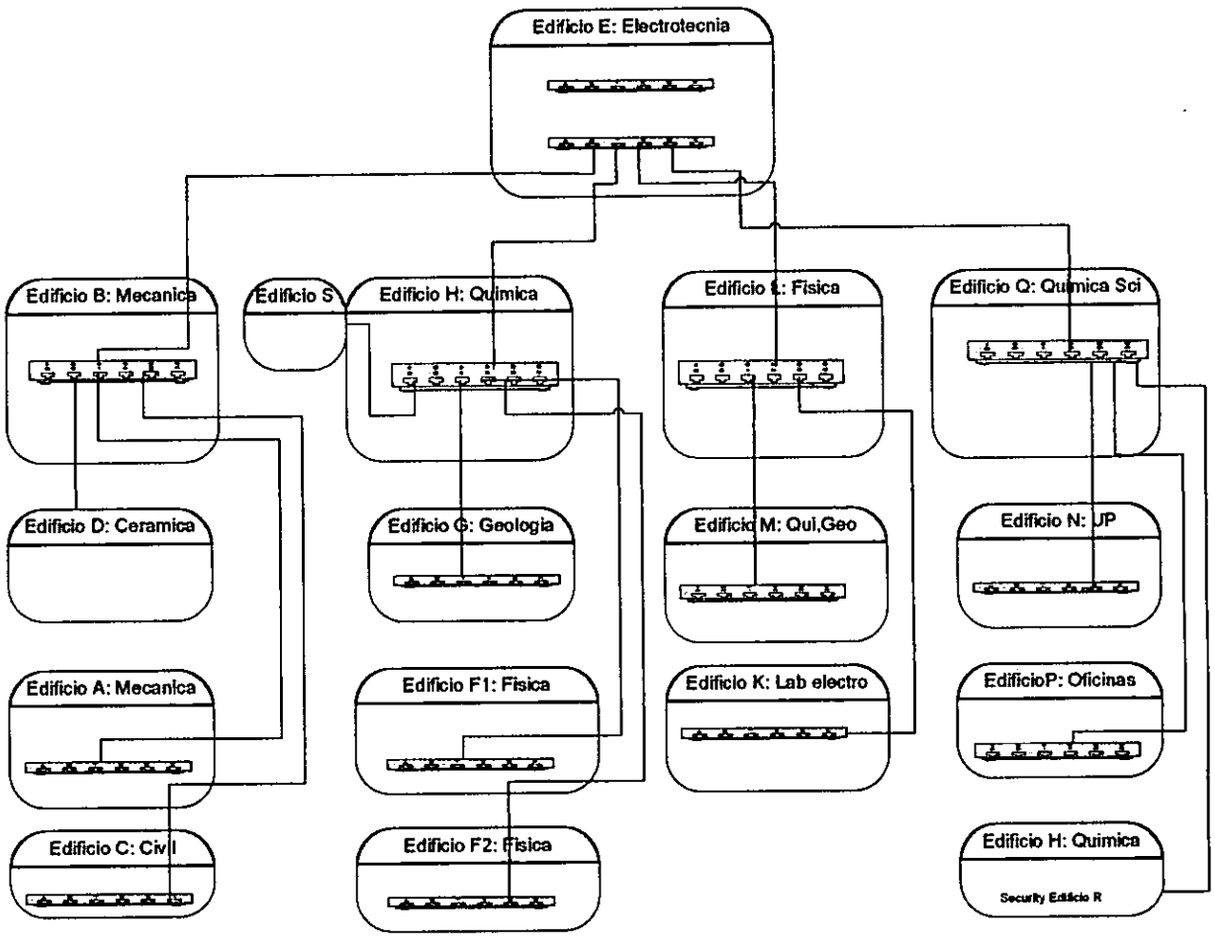

Figura 8-2: Rede no Campus das Engenharias.

Todas as unidades fora do campus Universitário estão ligadas por meio de Links wireless. O CIUEM é o Hub Principal para a ligação Internacional com 512 kbps uplink e 2 Mbits downlink para EUA via PanAmSat à MCI. O CIUEM providencia Link internacional a todas as faculdades, combinando tecnologias de Wireless e dial-up. Estações fora do campus usam tecnologia wireless. A tecnologia wireless é uma rede privada e pertence a universidade. A conexão dial-up é feita através da linha publica da TDM figura 8-3.

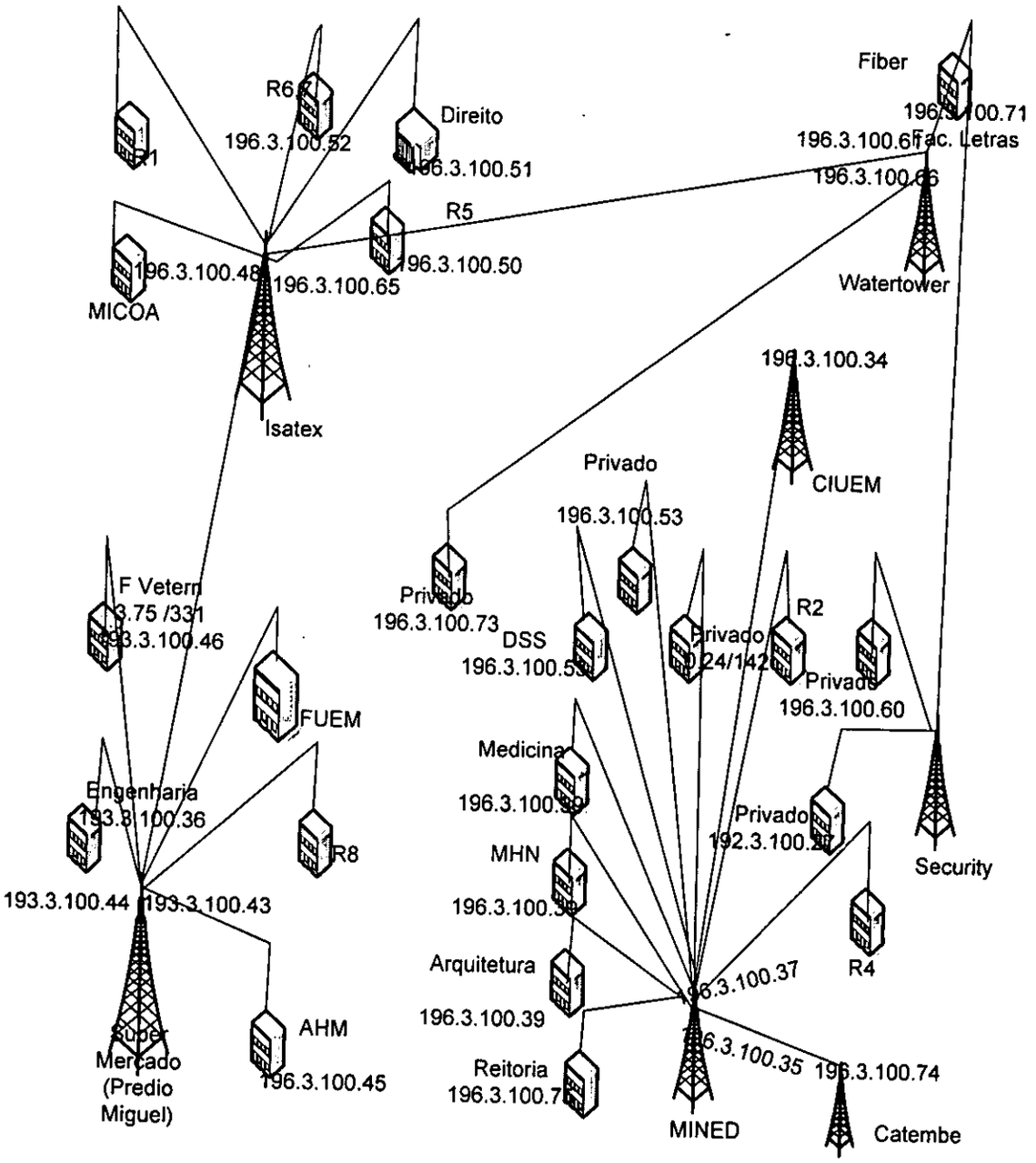

Figura 8-3: Rede MAN Wireless da UEM.

Em resumo Existem na UEM redes Locais (LANs) que interligam PCs e outros dispositivos como servidores ao, nível local para a partilha de recursos e informação. Estas redes locais estão

por sua vez interligadas numa rede sem fio (wireless). Que constitui assim a rede global da UEM. A tabela 8-1 mostra a distribuição da rede global da UEM.

	Tipo	Topologia	Capacidade	Beneficiários
Principal Backbone nos Campus	Fibra Óptica	árvore	100 Mbps por migrar para 1Gbps	Campus principal & Engenharias
Rede Local (LAN)	UTP	estrela	10/100 Mbps	Faculdades
Backbone (MAN)	Wireless 802.11. DSSS	árvore	11 Mbps	UEM
Site Links	FHSS	Ponto a Ponto E estrela radial	3 Mbps	UEM

Tabela 8-1: Distribuição da rede da UEM.

8.2 Necessidades de IPv6 na UEM

O ensino e investigação são as actividades principais da Universidade. Sendo assim, impõe-se que a procura da excelência e qualidade privilegie, em primeiro lugar, uma maior eficiência nas tais actividades [18].

As tecnologias de informação irão contribuir para melhorar o processo de ensino, contribuindo tanto para o aumento do ingresso de estudantes "virtuais", como para o fomento da investigação [18].

Deste modo a UEM juntando aquilo que é parte do seu plano estratégico pensa em realizar cursos adequados a realidade nacional, introduzindo novos métodos de ensino, usando-se das tecnologias de informação que poderão ser por exemplo a introdução de métodos de ensino a distância.

A técnica de ensino a distância, requer aplicações de voz e vídeo em tempo real, vendo-se deste modo a necessidade de se integrar o IPv6, sendo este a base para suporte eficiente a este tipo de aplicações.

O IPv4 foi concebido não prevendo a necessidade de fornecer segurança (autenticidade e privacidade) dos elementos que se comunicam. O IPv6 tendo sido desenvolvido a posterior incorpora estas facilidades para além de possuir um espaço de endereçamento maior.

Como qualquer outra instituição, a UEM tem se preocupado com a segurança da informação na sua rede, tendo diversos sistemas que precisam de protecção da informação que fluí na rede. Neste momento a segurança é implementada com recursos à diversas aplicações criptográficas requerendo custos de implementação elevados e necessitando de muito tempo para serem desenvolvidas. O que se verifica até então é que a UEM não faz o uso deste tipo de aplicações. Com o IPv6 não haverá mais preocupação da segurança no nível de aplicação, pois este implementa as medidas de segurança ao nível do núcleo do protocolo o que irá ajudar a UEM na resolução do problema descrito acima.

A mobilidade é um outro factor importante, olhando para a tecnologia wireless que a UEM implementa, os utilizadores terão a possibilidade de se mover de uma rede a outra sem quebrar a sessão/conexão estabelecida.

A autoconfiguração também é uma característica fundamental requerida na rede da UEM dada a dispersão em que se encontram os recursos e o plano de expansão da mesma. Esta pode constituir largos passos para facilitar o trabalho dos administradores, dando-lhes a possibilidade de se preocuparem apenas com a gestão dos elementos centrais da rede.

O IPv6 não só trará estas novas facilidades como também irá permitir que a UEM faça parte da comunidade internacional que usa o IPv6, facilitando o intercâmbio de aplicações que suportam o IPv6.

9 Modelo de implementação IPv6 na UEM

O IPv6 provavelmente estará presente, daqui a alguns anos, em máquinas onde hoje possuem o IPv4, e será tão comum quanto o IPv4 é actualmente. Mas para que isto aconteça, é necessário criar mecanismos para que se possa testar o protocolo e suas aplicações, bem como novas propostas de protocolos. Um dos mecanismos, que foi considerado essencial para a implementação do IPv6 é a criação de uma rede de testes ou experimental que será discutida no capítulo 10.

O modelo proposto para a implementação do protocolo IPv6 na UEM e também no desenvolvimento e teste pode ser visto na figura 9-1.

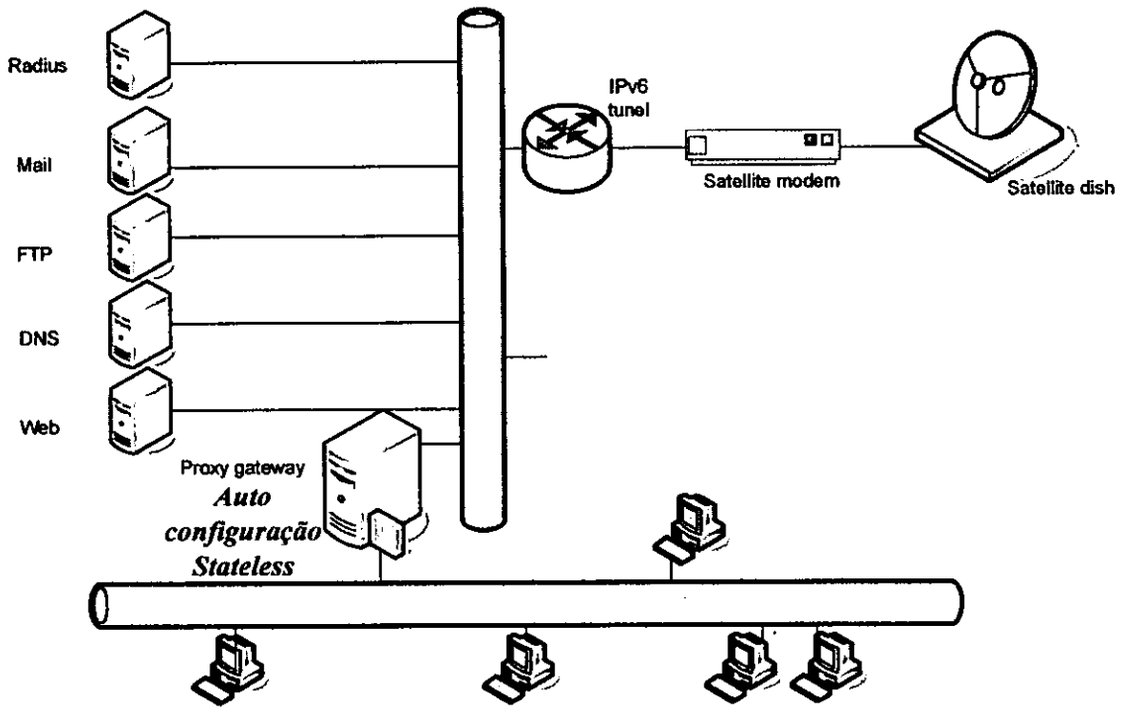

Figura 9-1: Modelo de Implementação da rede IPv6 na UEM.

O IPv6 será implementado gradualmente, de modo que ambas as versões do IP possam coexistir por alguns anos, até que o período de transição seja completado. Assim, este modelo deverá apresentar compatibilidade com a versão anterior. *Hosts* IPv6 são então capazes de se comunicar tanto com *hosts e servers* IPv6 e IPv4, através do túnel IPv6 mostrado na figura 9-1.

Este cenário funciona da seguinte maneira: Caso esta rede com *hosts* IPv6 queira se comunicar com outra IPv6, e só hajam entre eles *hosts* IPv4, poderão utilizar a técnica de tunelamento usando a entrada IPv6 túnel, ilustrada da Figura 9-1, que consiste em "reempacotar" pacotes IPv6 no formato IPv4, ou seja, ter um pacote IPv6 dentro de um IPv4, enviá-los pelos *hosts* IPv4, e "desempacotá-los" quando alcançarem o outro *host* IPv6 (router).

O modelo apresentado representa a rede do CIUEM, como sendo o centro de distribuição da rede nas diferentes partições, e também como o ponto escolhido para primeira fase na implementação do protocolo.

O cenário de rede para o modelo proposto será constituído por um router Ipv6 (Ipv6 túnel), ligado a rede local (CIUEM) e será ligado ao backbone da Internet. Isto irá permitir que ambos os pacotes IPv4 e IPv6 trafeguem nesta rede.

Os servidores de nomes (DNS), FTP, Web, serão configurados para terem suporte ao protocolo Ipv6.

10 Rede de Testes do Ipv6

Para fins de testes com o IPv6 foram usados 4 PCs, correndo os sistemas operativos Redhat Linux 9 (verões 7.2 e 9.0) e MS windows (2000 e XP). A figura 9-2 mostra o esquema.

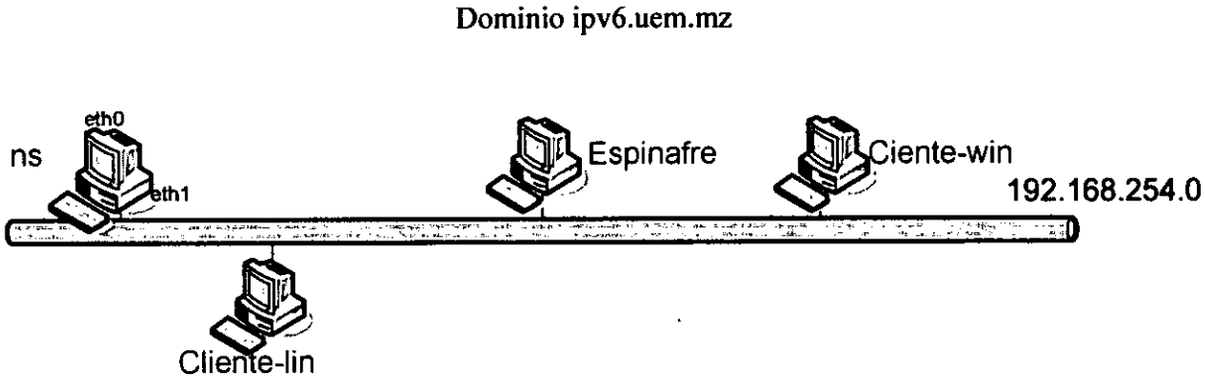

Figure 10-1: Rede estabelecida para testes IPv6.

A figura 10-1 mostra um segmento de rede estabelecido para testes, onde à cada máquina é associado um nome: (ns,espinafre,cliente-win, cliente-lin e guest), estando ligados a uma rede interna IPv4 (192.168.254.0) do CIUEM. O linux constituirá a plataforma de suporte para os diferentes serviços para a rede em causa.

A versão do kernel nas máquinas com Redhat Linux (9.0) é 2.4.20 e na máquina com Redhat Linux (7.2) é 2.4.7, já vêm com suporte ao IPv6, precizando-se apenas fazer as devidas configurações, para activar os respectivos módulos.

Cada máquina na rede desempenha uma função específica. A máquina **ns** correndo apenas o Linux 9.0 tem o papel de servidor é onde foram configurados os serviços de DNS, RADVD, Web server e FTP. As restantes máquinas combinam os sistemas operativos Linux e Windows onde desempenham o papel de clientes respectivamente.

10.1 Habilitando o IPv6 nas interfaces de rede

Para se habilitar o módulo de suporte para o IPv6 manualmente no Linux, deve-se digitar na linha de comandos do sistema operativo o seguinte comando:

```
#modprobe ipv6
```

Após a execussão do comando, as interfaces, adquirem um endereço de link-local com prefixo fe80::/64, onde a parte que compõe o host é resultado da transformação EUI-64 no endereço MAC da interface. Para visualizar as configurações actuais após a execução do comando `modprobe` executa-se o comando:

```
#ifconfig
```

O resultado a seguir é referente a máquina **ns,** na figura 9-3 são apresentados alguns endereços adquiridos pelas máquinas após a execução do comando `modprobe ipv6` :

```
eth0      Link encap:Ethernet  HWaddr 00:30:F1:39:A2:56
          inet addr:192.168.254.92  Bcast:192.168.254.255
Mask:255.255.255.0
          inet6 addr: fe80::230:f1ff:fe39:a256/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:576 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:65355 (63.8 Kb)  TX bytes:1144 (1.1 Kb)
          Interrupt:11 Base address:0xf000

eth1      Link encap:Ethernet  HWaddr 00:30:F1:39:A3:52
          inet addr:192.168.10.3  Bcast:192.168.10.255  Mask:255.255.255.0
          inet6 addr: fe80::230:f1ff:fe39:a352/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:10 dropped:0 overruns:0 carrier:20
          collisions:0 txqueuelen:100
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
          Interrupt:5 Base address:0x1000

lo        Link encap: Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:10 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:700 (700.0 b)  TX bytes:700 (700.0 b)
```

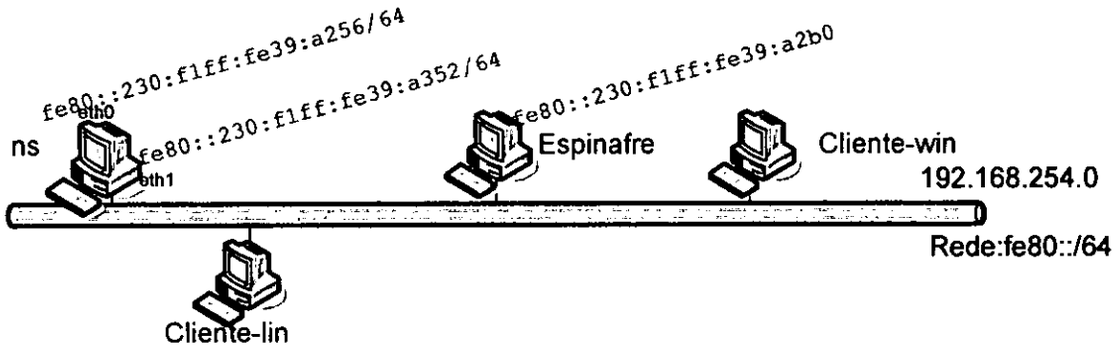

Figura 10-2: Rede de testes com os endereços IPv6 após o comando `modprobe ipv6`

É importante lembrar que ao se desligar ou ao se reinicializar o sistema o comando modprobe ipv6 deverá ser executado novamente ou por outra para que o endereço permaneça, mesmo quando se reinicializa o sistema será necessário acrescentar no ficheiro `/etc/modules.conf` a seguinte linha: `alias net-pf-10 ipv6` para habilitar automaticamente. Caso pretenda-se desabilitar deve-se substituir a linha acima pela: `alias net-pf-10 off`.

Exemplo do ficheiro `/etc/modules.conf`:

```
alias eth0 tulip
alias eth1 tulip
alias sound-slot-0 i810_audio
post-install sound-slot-0 /bin/aumix-minimal -f /etc/.aumixrc -L >/dev/null
2>&1 || :
pre-remove sound-slot-0 /bin/aumix-minimal -f /etc/.aumixrc -S >/dev/null
2>&1 || :
alias usb-controller usb-uhci
alias net-pf-10 ipv6
```

O comando `#ifconfig <interface> inet6 add <endereço ipv6>` pode ser usado para adiccionar um endereço IPv6 manualmente.
Para este caso, o comando `# ifconfig eth0 inet6 add fe3f::1:2:3:4:5/64`, adiciona o endereço `fe3f::1:2:3:4:5/64` na interface eth0 da máquina denominada **ns** e como resultado do comando `ifconfig` digitado de seguida teremos:

```
eth0      Link encap:Ethernet  HWaddr 00:30:F1:39:A2:56
          inet addr:192.168.254.92  Bcast:192.168.254.255  Mask:255.255.255.0
          inet6 addr: fe80::230:f1ff:fe39:a256/64 Scope:Link
```

```
          inet6 addr: fe3f::1:2:3:4:5/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3866 errors:0 dropped:0 overruns:0 frame:0
          TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:518708 (506.5 Kb)  TX bytes:1771 (1.7 Kb)
          Interrupt:11 Base address:0xf000

eth1      Link encap:Ethernet  HWaddr 00:30:F1:39:A3:52
          inet addr:192.168.10.3  Bcast:192.168.10.255  Mask:255.255.255.0
          inet6 addr: fe80::230:f1ff:fe39:a352/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:10 dropped:0 overruns:0 carrier:20
          collisions:0 txqueuelen:100
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
          Interrupt:5 Base address:0x1000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:10 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:700 (700.0 b)  TX bytes:700 (700.0 b)
```

Os testes de comunicação entre máquinas são executados através do comando `ping`, quando o `ping` for direccionado para um endereço linklocal o camando será:

`#ping6 -I  <divice> <endereço IPv6>` ou por outra `#ping6 <endereço ipv6>` quando a direcção do ping for um endereço diferente do linklocal. Onde: <device> é o nome da interface por onde os pacotes deverão ser enviados.

Para efeitos de teste observa-se o ping efectuado da máquina **ns** para máquina **espinafre:**

```
# ping6-I eth1 fe80::230:f1ff:fe39:a2b0

PING fe80::230:f1ff:fe39:a2b0 (fe80::230:f1ff:fe39:a2b0) from
fe80::230:f1ff:fe39:a256 eth0: 56 data bytes
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=1 ttl=64 time=0.323 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=2 ttl=64 time=0.363 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=3 ttl=64 time=0.312 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=4 ttl=64 time=0.348 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=5 ttl=64 time=0.319 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=6 ttl=64 time=0.630 ms
```

```
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=7 ttl=64 time=0.308 ms
```

o camando ping pode ser executado da máquina **cliente-lin** para máquina **ns** usando o endereço global do modo seguinte:

`#ping6 fe3f:1:2:3:4:5` e o resultado deste ping é:

```
PING fe3f::1:2:3:4:5(fe3f::1:2:3:4:5) 56 data bytes
64 bytes from fe3f::1:2:3:4:5: icmp_seq=1 ttl=64 time=0.708 ms
64 bytes from fe3f::1:2:3:4:5: icmp_seq=2 ttl=64 time=0.315 ms
64 bytes from fe3f::1:2:3:4:5: icmp_seq=3 ttl=64 time=0.319 ms
64 bytes from fe3f::1:2:3:4:5: icmp_seq=4 ttl=64 time=0.313 ms
64 bytes from fe3f::1:2:3:4:5: icmp_seq=5 ttl=64 time=0.309 ms
64 bytes from fe3f::1:2:3:4:5: icmp_seq=6 ttl=64 time=0.287 ms
64 bytes from fe3f::1:2:3:4:5: icmp_seq=7 ttl=64 time=0.311 ms
64 bytes from fe3f::1:2:3:4:5: icmp_seq=8 ttl=64 time=0.313 ms
64 bytes from fe3f::1:2:3:4:5: icmp_seq=9 ttl=64 time=0.320 ms

--- fe3f::1:2:3:4:5 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8002ms
rtt min/avg/max/mdev = 0.287/0.355/0.708/0.125 ms
```

10.2 Login Remoto Seguro/Secure Shell (SSH)

o **ssh** é usado para aceder de uma forma segura a uma máquina remota; os testes mostraram que para o caso de endereço de link-local o **ssh** não funciona, respondendo para endereços diferentes deste sendo o comando executado do seguinte modo:

`#ssh -6 <endereço ipv6>`, o exemplo asseguir mostra a execução do comando a partir da máquina **cliente-lin**, com o fim a aceder remotamente a máquina **ns**.

```
#ssh -6 fe3f::1:2:3:4:1

[root@cliente1 root]# ssh -6 fe3f::1:2:3:4:5
The authenticity of host 'fe3f::1:2:3:4:5 (fe3f::1:2:3:4:5)' can't be
established.
RSA key fingerprint is f0:34:ad:84:0e:ce:70:e6:3d:38:bb:0b:60:d7:e9:00.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'fe3f::1:2:3:4:5' (RSA) to the list of known
hosts.
root@fe3f::1:2:3:4:5's password:
Last login: Mon Nov 15 11:27:38 2004
```

10.3 Autoconfiguração de endereços IPv6 pelo modo stateless

Algo interessante em IPv6 é a autoconfiguração. Ela permite que o host ao se conectar a rede IPv6, seja atribuido seu endereço e o respectivo gateway sem ter que usar o DHCP.

A autoconfiguração referida funciona do modo seguinte:

- O nó usa o prefixo do endereço de link-local e o estende para um endereço de 128 bits adiccionando seu endereço MAC com base no método EUI-64.
- O nó envia mensagens de neighbour solicitation para verificar se um outro nó usa o mesmo endereço, em caso afirmativo esse nó responde com mensagem de neighbour advertisement e a autoconfiguração é abortada;
- Caso não o prefixo de link-local é adiccionado ao seu interface. Neste momento o nó já possui conectividade IPv6 com os seus vizinhos;
- Sendo assim o nó envia mensagens de *router advertisement* para todos os roteadores do grupo multicast para verificar a presença ou não destes;
- O roteador responde com mensagem de *router advertisement* indicando o prefixo de rede;
- O prefixo da rede (x/64) é estendido para endereço de 128 bits com base no método EUI-64;

O linux habilita a autoconfiguração por defeito, sendo apenas preciso configurar o roteador com o prefixo que vai ser atribuído a rede. Existindo duas maneiras para isso: usando o radvd ou zebra.

10.3.1 Configuração do "Router Advertisement Daemon"(RADVD)

Router Advertisement Daemon **(RADVD)** é uma feramenta usada para enviar "router advertisements" indicando o prefixo de rede. Não suporta outros protocolos de roteamento como o zebra mas permite um controle para mensagens de router advertisements.

A configuração é feita inserindo-se as seguintes linhas no ficheiro: /etc/radvd.conf
Exemplo:

```
# especifica a interface por onde serão enviadas as mensagens de router
interface eth0
advertisement{
# habilita a autoconfiguração
    AdvSendAdvert on;
# especifica o prefixo de rede que sera atribuido aos hosts na rede
prefix 3ffe:306:11:2/64 {
        AdvOnLink;
    };
};
```

Apois a execução do comando `radvd -d 1` já se pode ver o endereço na máquina **ns** com o prefixo atribuido pelo radvd:

```
#ifconfig eth0

eth0      Link encap:Ethernet  HWaddr 00:30:F1:39:A2:56
          inet addr:192.168.254.92  Bcast:192.168.254.255  Mask:255.255.255.0
          inet6 addr: fe80::230:f1ff:fe39:a256/64 Scope:Link
          inet6 addr: 3ffe:306:11:2:230:f1ff:fe39:a256/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1404039 errors:0 dropped:0 overruns:0 frame:0
          TX packets:129087 errors:2 dropped:0 overruns:0 carrier:2
          collisions:70989 txqueuelen:100
          RX bytes:146541393 (139.7 Mb)  TX bytes:107374797 (102.4 Mb)
          Interrupt:11 Base address:0xf000
```

Endereços adquiridos pelas restantes máquinas podem ser vistos na figura 9-4:

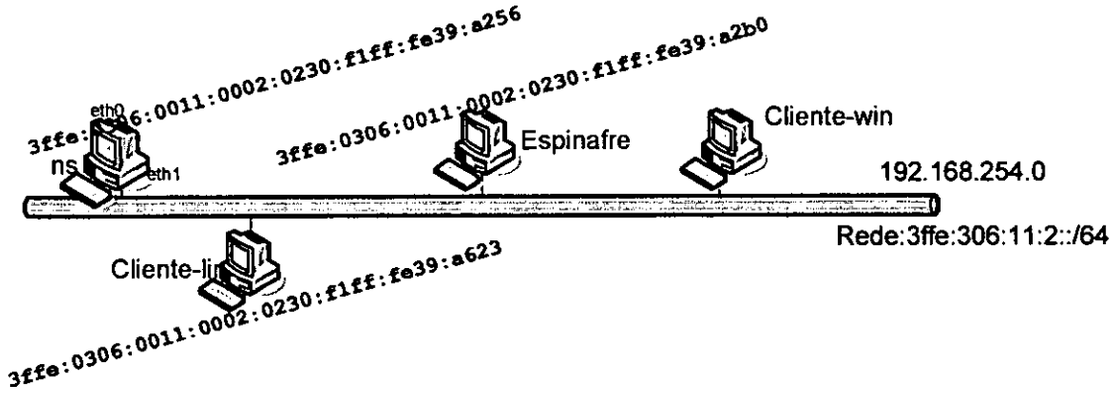

Figura 10-3: Endereços adquiridos pelas máquinas pelo radvd.

10.3.2 Configuração do Zebra prefix advertisement

O zebra é um software de roteamento, para roteadores baseados em PCs, mais completo em termos de suporte à vários protocolos de roteamento.

A única coisa que se deve fazer é a configuração do prefixo para cada interface por onde se pretende fazer o envio de mensagens de *router advertisements*, devendo-se inserir as configurações no seguinte ficheiro: /etc/zebra

O exemplo abaixo mostra como é feita a configuração para enviar o prefixo 3ffe:306:11:2/64 para a rede através da interface eth0:

Exemplo:

```
interface eth0
ipv6 nd send-ra
ipv6 nd prefix-advertisement 3ffe:306:11:2/64
```

10.4 Configuração DNS

Domain Name System (DNS), faz o mapeamento entre os nomes das máquinas numa dada rede e os respectivos endereços IP, sendo este mapeamento feito de nome para endereço IP ou de endereço IP para nome.

O mapeamento é feito com base na associação do nome que para este caso seria ns.ipv6.uem.mz que indica o nome do sevidor onde irá correr o DNS com o endereço IP da própria máquina `3ffe:306:11:2:230:f1ff:fe39:a256`. O DNS também contém o mapeamento do endereço IP para o nome da máquina, este mapeamento é conhecido como mapeamento reverso ou *reverse mapping*.

Os ficheiros requeridos para esta configuração serão:

1. `/etc/named.conf ( ficheiro de configuração de zonas na rede);`
2. `/etc/resolv.conf ( ficheiro que especifica o endereço do servidor para onde o host deverá apontar);`
3. `/var/named/ ipv6.uem.mz.zone6 ( ficheiro de mapeamento directo );`
4. `/var/named/ipv6.uem.mz.reverse6.2 (mapeamento reverso)`
5. `/var/named/ipv6.uem.mz.reverse6.3 (mapeamento reverso)`
6. `/var/named/named.ca (ficheiro que contêm configuração de zonas na Internet, ou guarda informação sobre os root name servers)`
7. `/var/named/localhost.zone.v2 (mapeamento directo para localhost)`
8. `/var/named/localhost.zone.reverse (mapeamento reverso para localhost).`

Os ficheiros que serão mostrados nesta fase serão o primeiro e os dois últimos, visto que os outros não apresentam diferença com os do IPv4.

O ficheiro asseguir /etc/named.conf, é de configuração de zonas, dentro do domínio ipv6.uem.mz, é configurado na máquina **ns**:

```
options {
      directory "/var/named"; # a linha especifica onde serão guardados os
ficheiros de configuração#
      listen-on-v6 {on}; # habilita a escuta pelo ficheiro resolv.conf pelo
endereço Ipv6#
};

zone "." IN  # DNS tem estrutura de árvore o "." Ou root é o topo ou raiz
abaixo deste Top Level Domain, responde pelas zonas for a do domínio defenido
como por exemplo:(org, com. Edu, net etc)
{
      type hint;
         file "named.ca";
};

#abaixo os registos indicam as definições de zonas, ambos indicando
mapeamentos directos ou reversos e os devidos fichiros de configuração na
linha file "nome do ficheiro";#

zone "localhost" IN {
      type master;
      file "localhost.zone.v2";
//    allow-update { none; };
};

zone "0.0.127.in-addr.arpa" IN {
      type master;
      file "named.local.v2";
//    allow-update { none; };
};
zone "0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.e.f.ip6.int" IN {
      type master;
      file "ipv6.uem.mz.zone.reverse6.2";
      allow-update { none; };
};
zone "2.0.0.0.1.1.0.0.6.0.3.0.e.f.f.3.ip6.int" IN {
      type master;
      file "ipv6.uem.mz.zone.reverse6.3";
      allow-update { none; };
};


zone "ipv6.uem.mz" IN {
      type master;
  //    notify no;
      file "ipv6.uem.mz.zone6";
};
include "/etc/rndc.key";
```

Os registos abaixo podem ser encontrados uns nos ficheiros de mapeamento reverso outros no de mapeamento directo:

Start for Authority (SOA)- o primeiro nome depois do SOA ou por outra `ns.ipv6.uem.mz` é o nome do servidor DNS, o Segundo nome `root.ns.ipv6.uem.mz`, indica o *mail address* do reponsavel pelo domínio.

NS- lista os name server para esta zona.

AAAA- nome para mapeamento de endereço;

PTR (Domain name pointer) - endereço para mapeamento de nomes;

CNAME- nome canónico para aliase

O exemplo asseguir representa a configuração do ficheiro /var/named/ipv6.uem.zone6 de mapeamento directo:

```
$TTL  86400
@       IN      SOA  ns.ipv6.uem.mz. root.ns.ipv6.uem.mz. (   .
num)                                    1997022700  ; Serial number (yyyymmdd-

                                         3H ;  Refresh
                                         15M ; Retry
                                         1W  ; Expire
                                         1D) ; Minimum

                        IN NS ns.ipv6.uem.mz.
ns 1D IN AAAA fe80:0000:0000:0000:0230:f1ff:fe39:a352
ns 1D IN AAAA fe80:0000:0000:0000:0230:f1ff:fe39:a256
ns 1D IN AAAA fe3f:0000:0000:0001:0002:0003:0004:0005
ns 1D IN AAAA 3ffe:0306:0011:0002:0230:f1ff:fe39:a256
cliente-lin 1D  IN AAAA fe80:0000:0000:0000:0230:f1ff:fe39:a623
cliente-lin 1D  IN AAAA 3ffe:0306:0011:0002:0230:f1ff:fe39:a623
espinafre   1D  IN AAAA fe80:0000:0000:0000:0230:f1ff:fe39:a2b0
espinafre   1D  IN AAAA 3ffe:0306:0011:0002:0230:f1ff:fe39:a2b0
www         CNAME                   ns
```

ficheiro de mapeamento reverso /var/named/ ipv6.uem.mz.reverse6:

```
$TTL  86400

$ORIGIN 0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.e.f.ip6.int.
@       IN SOA ns.ipv6.uem.mz.   root.ns.ipv6.uem.mz. (

             1997022700   ; Serial
                       3H ; Refresh
                     15M ; Retry
                      1W ; Expire
                     1D) ; Minimum

               NS   ns.ipv6.uem.mz.
6.5.2.a.9.3.e.f.f.f.1.f.0.3.2.0   IN   PTR   ns.ipv6.uem.mz.
2.3.5.a.9.3.e.f.f.f.1.f.0.3.2.0   IN   PTR   ns.ipv6.uem.mz.
3.2.6.a.9.3.e.f.f.f.1.f.0.3.2.0   IN   PTR   cliente-lin.ipv6.uem.mz.
0.b.2.a.9.3.e.f.f.f.1.f.0.3.2.0   IN   PTR   espinafre.ipv6.uem.mz.
```

Outro ficheiro de mapeamento reverso é o : ipv6.uem.mz.zone.reverse6.3, corresponde a zona do prefixo atribuido pelo radvd:

```
$TTL  86400

$ORIGIN 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.f.3.e.f.ip6.int.
@       IN SOA ns.ipv6.uem.mz.   root.ns.ipv6.uem.mz. (

             1997022700   ; Serial
                       3H ; Refresh
                     15M ; Retry
                      1W ; Expire
                     1D) ; Minimum

               NS   ns.ipv6.uem.mz.
6.5.2.a.9.3.e.f.f.f.1.f.0.3.2.0   IN   PTR   ns.ipv6.uem.mz.
3.2.6.a.9.3.e.f.f.f.1.f.0.3.2.0   IN   PTR   ns.ipv6.uem.mz.
3.2.6.a.9.3.e.f.f.f.f.1.f.0.3.2.0 IN   PTR   cliente-lin.ipv6.uem.mz.
0.b.2.a.9.3.e.f.f.f.1.f.0.3.2.0 IN   PTR   espinafre.ipv6.uem.mz.
```

As ferramentas usadas para fins de teste de corecta funcionalidade (lookup) do DNS são: o `dig` e o `host` os exemplos asseguir monstram como se devem usar os camandos para o efeito:

`# host -t aaaa espinafre.ipv6.uem.mz` esta linha pode-se comparar com uma pergunta como: " qual é o endereço do host espinafre.ipv6.uem.mz":

```
espinafre.ipv6.uem.mz has AAAA address fe80::230:f1ff:fe39:a2b0
```

```
#host -n fe80:0000:0000:0000:0230:f1ff:fe39:a256
```

```
6.5.2.a.9.3.e.f.f.f.1.f.0.3.2.0.0.0.0.0.0.0.0.0.0.0.0.0.8.e.f.ip6.int
domain name pointer ns.ipv6.uem.mz.
```

```
#host -n fe80:0000:0000:0000:0230:f1ff:fe39:a352
```

```
2.5.3.a.9.3.e.f.f.f.1.f.0.3.2.0.0.0.0.0.0.0.0.0.0.0.0.0.8.e.f.ip6.int
domain name pointer ns.ipv6.uem.mz.
```

`#dig ipv6.uem.mz` resume a informação sobre o DNS.

```
; <<>> DiG 9.2.1 <<>> ipv6.uem.mz
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48489
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;ipv6.uem.mz.                   IN    A

;; AUTHORITY SECTION:
ipv6.uem.mz.            86400 IN    SOA   ns.ipv6.uem.mz.
root.ns.ipv6.uem.mz. 1997022700 10800 900 604800 86400

;; Query time: 1 msec
;; SERVER:
3ffe:306:11:2:230:f1ff:fe39:a256#53(3ffe:306:11:2:230:f1ff:fe39:a256)
;; WHEN: Mon Jan 24 17:23:34 2005
;; MSG SIZE  rcvd: 73
```

Nesta fase, como já existe um servidor DNS o ping pode ser efectuado especificando o endereço.

Exempolo:

```
#Ping6 -I eth0 espinafre.ipv6.uem.mz

PING espinafre.ipv6.uem.mz(fe80::230:f1ff:fe39:a2b0) from
fe80::230:f1ff:fe39:a256 eth0: 56 data bytes
```

```
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=1 ttl=64 time=0.323 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=2 ttl=64 time=0.363 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=3 ttl=64 time=0.312 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=4 ttl=64 time=0.348 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=5 ttl=64 time=0.319 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=6 ttl=64 time=0.630 ms
64 bytes from fe80::230:f1ff:fe39:a2b0: icmp_seq=7 ttl=64 time=0.308 ms
--- fe3f::1:2:3:4:5 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8002ms
rtt min/avg/max/mdev = 0.287/0.355/0.708/0.125 ms
```

10.5 Configuração de Web server

O servidor de páginas web é configurado no ficheiro `/etc/httpd/conf/httpd.conf`, são apenas mostradas as linhas básicas de configurações necessárias para o efeito:

`DocumentRoot "/var/testeipv6"`- Esta linha especifica o directório onde serão hospedados os conteúdos das paginas web.

`Listen [fe3f::1:2:3:4:5]:80`- Esta linha especifica a porta de escuta e o endereço IPv6 do servidor web.

`DirectoryIndex doc1.html index.html.var`- Mostra a página que deve aparecer como primeira página ao se aceder ao serviço web; para este caso o doc1.html, será a página que aparecerá por defeito e o directório onde ela se encontra é `/var/testeipv6/doc1`.

```
<VirtualHost [3ffe:0306:0011:0002:0230:f1ff:fe39:a256]:80>
</VirtualHost>
```

Por fim o registo composto pelas *tags*

`<VirtualHost [3ffe:0306:0011:0002:0230:f1ff:fe39:a256]:80>` e `</VirtualHost>`, deverá ser editado com o endereço do servidor web correspondente e a sua respectiva porta.

Para efeitos de teste deve-se aceder a página com base na segunte linha:

http://[3ffe:0306:0011:0002:0230:f1ff:fe39:a256]

Com o servidor de nomes DNS pode-se associar este endereço ao nome usual, isto poderia se consiguir acrescentando a linha www CNAME ns no ficheiro:

`/var/named/ipv6.uem.mz.zone6`, desta forma a página já se pode aceder usando a seguinte linha: http://www.ipv6.uem.mz .

10.6 File Transfer Protocol (FTP)

FTP é um protocolo Cliente/Servidor que permite aos utilizadores transfirirem ficheiros de e para uma máquina remota. Trabalha com auxílio do TCP e é comum ser usado nas redes de computadores locais, assim como nas amplas, como a Internet.

Existem várias aplicações FTP para servidores tendo sido usado para este trabalho moftpd, por defeito a versão do linux usada não traz este ficheiro, sendo assim foi necessário fazer o dowload do mesmo. Os comandos usados para configuração do mesmo foram:

`# ./ configure`

`# make` (criar executavel)

`# make install` (colocar o executavel nos ficheiros do linux padrão)

Em seguida é mostrada a localização do ficheiro de configuração e o seu conteúdo /usr/local/etc/moftpd.conf: , algumas linhas foram alteradas para refletir as configurações da rede de testes, onde foram usados os dados do servidor FTP que para este caso foi a máquina **ns.**

O ficheiro asseguir mostra essas linhas e acima de cada uma delas vem a devida explicação.

```
#porta de onde deve-se escutar o ftp
Port 21
# especifica o endereço do servidor ftp que deve ser escutado
Bind 3ffe:0306:0011:0002:0230:f1ff:fe39:a256;
# qualquer host IPv4 pode aceder a este servidor
Range 0.0.0.0/0;
# qualquerhost Ipv6 pode aceder a este servidor
Range ::/0;
```

```
# assemelha-se a pergunta podemos permitir acesso de ou #para redes fora do
nosso control?

 AllowForeign Yes

# pode-se pedir password se um usuário não conhecido tentar aceder ao
#serviço

PassIfInvalid no

AllowLogin no

# localização por defeito atribuida aos usuários quando acessam ao serviço

Chroot /var/ftp

# o tempo maximo dado ao usuário em cada conexão pode se desabilitar
#colocando valor 0

MaxIdle 5m

# adiciona aliase, o primeiro argumento refere-se ao usuário e o Segundo ao
#aliase

UserAlias "anonymous", "ftp";

DictoryMsgFile "README";

<user anonymous>

Alias ftp;

Anonymous yes;

Uid nobody;

Home "/var/ftp";

<directory "/">

Deny all;

     Allow reading, signed;

      Require encrypted;

FakeUser ftp;

FakeDirMode "drwxr-xr-x"

<directory incoming>

       Allow storing;
      Deny listing;
         </directory>
```

```
</directory>
  </user>
 </server>
```

10.7 O protocolo Ipv6 e a Microsoft (windows 2000 e XP)

O windows 2000 por default não possui instalado o suporte ao protocolo Ipv6, para tal deve ser feito o download do arquivo "tpiv6-001205.exe" fornecido pela Microsoft no site http://msdn.microsoft.com/downlads/sdks/platform/tpipv6/download.asp.

Para a instalação do suporte ao protocolo é necessário atender os pré-requisitos mínimos abaixo indicados:

- Windows 2000service pack 1;
- Adaptador de rede;
- Protocolo IPv4 instalado, tendoem conta que o IPv6 substituirá o IPv4 num futuro proximo é necessário que se integre com IPv4 e trabalhe por cima dele até sua migração total.

Os únicos sistemas operativos da Microsoft que até agora suportam Ipv6 são windows 2000 e XP, que para o caso do XP para que o endereço Ipv6 seja habilitado basta executar no prompt do DOS o seguinte comando:

```
C:\>ipv6 install
```

Mais detalhes sobre a instação e configuração do Ipv6 no windows vide em Anexo C.

11 Estratégias para Transição

Sendo o CIUEM responsavél pela rede da UEM, vai ser a primeira, na sua rede local a introduzir serviços com IPv6 e será responsável pela sua atribuição nas restantes unidades.

A transição no que diz respeito ás restantes repartições será feita olhando para aquilo que é estratégia da UEM, no que diz respeito ás necessidades prioritárias que cada repartição concorre às mudanças que o IPv6 trás. Inicialmente todos os servidores de nomes deverão ser configurados de modo a suportar ambos os protocolos IPv6 e IPv4 de modo a permitir uma transição suave.

A estratégia principal exigida pelo grupo de trabalho IPv6 é a implementação do TCP/IP com pilha dupla (IPv6 e IPv4) nos hosts e roteadores da rede [15].

Há várias outras questões e observações sobre o processo de transição que devem ser consideradas como: planeamento de alocação de endereços, requisitos de software (sistemas operativos e aplicações), requisitos de hardware (memória e CPU), velocidade dos links, recursos financeiros, etc [17].

É importante notar que IPv6 já possui representação de endereços prevendo sua utilização com mecanismos de transição, foi incluido um recurso para que pacotes IPv4 trafeguem em redes puramente IPv6.

12 Conclusões e Recomendações

O processo de desenvolvimento do IPv6 levou cerca de 10 anos. Enquanto se desenvolvia este standard, a indústria foi encontrando mecanismos de mitigar o problema de escassez de IPs. Um dos mecanismos mais divulgados é o uso do NAT. O NAT de facto resolve o problema de escassez de IPs. Contudo o NAT não é solução completa se comparada com a solução que o IPv6 oferece, pois o NAT não permite que qualquer host na rede tenha unicidade de endereço o que torna impossível que aplicações que exigem que o endereço seja permanente durante a sessão funcione.

A implementação de mecanismos de autoconfiguração presente no IPv6 é outra vantagem evidente que permite a mobilidade, eliminando problemas de configuração manual, um único prefixo de rede pode ser atribuido a N máquinas, sem necessidade de se especificar o intervalo de computadores na rede, usando-se para este efeito anúncios de prefixo feitos pelos roteadores, o que fornece uma vantagem enorme de escalabilidade que este protocolo possui.. O IPv6 prové meios alternativos de configuração a partir do DHCP se assim for pretendido.

O DNS é um serviço extremamente importante para o IPv6, a resolução de IPs em nomes facilita o uso destes endereços que pela sua natureza são extensos e difícies de se fixarem. Os ficheiros de configuração seguem o mesmo principio do IPv4, podendo manter registos que permitem a coexistência de ambos no mesmo DNS.

As ferramentas de open-source utilizadas no trabalho se mostraram apropriadas para implementações em âmbito de pesquisa, podendo por isso serem usadas em diversos projetos e aplicações.

Para que a transição dos protocolos não seja problemática, é necessário que os sistemas operativos já venham preparados para usar o IPv6, como já ocorre com as versões mais recentes do Linux e windows.

O uso de técnicas de integração é extremamente importante para o sucesso do protocolo IPv6 na rede em causa, pois fazer uma migração de um ambiente IPv4 para um ambiente IPv6 leva algum tempo a implementar. Deste modo é importante que na fase inicial da implementação deste protocolo não abranja toda a rede da UEM, propõe-se uma implementação faseada, escolhendo-se uma subrede como por exemplo a do CIUEM, num periodo mínimo de seis

meses, isto irá permitir uma fácil gestão e monitoramento da rede em causa, reportando erros que possam surgir e ainda criar competência necessária para uma implementação de larga escala.

Todo o estudo feito sobre IPv6 ajudou a entendê-lo e ver que existe bastante trabalho e possibilidade de criar novos projectos explorando minuciosamente as facilidades maravilhosas e possibilidades de ser usado numa área universal.

A utilização de endereços de `linklocal` para efeitos de teste mostrou-se limitada para alguns serviços como ssh -6, ping6, web etc, para o caso do ping é necessária a especificação da interface e o host destino fica sem saber que endereço solicitou o acesso. Sendo assim, para trabalhos futuros seria aconselhável que se usassem endereços globais para que os testes sejam eficazes.

O projecto proposto a UEM, trata-se de uma Reengenharia, existindo portanto recursos de base para suportar o protocolo em estudo, além de que os sistemas usados para configuração são gratuitos, esperando-se apenas por custos que advem da àquisição dos endereços IPv6 e formação dos admnistradores da rede.

Há ainda nesta área muitos requisitos por se explorarem, por exemplo neste trabalho foram apenas analizados aspectos de comunicaçao numa rede fixa é importante que para futuros trabalhos se explorem os testes de mobilidade, segurança e de qualidade de serviço que serão certamente necessários para o suporte de aplicações multimedia como o VoiP, video conferência, IP TV entre outros.

13 Bibliografia

[1] Tanenbaum, Andrew S, **Redes de Computadores**,Campus Ltda, 1997,923pp;

[2] Zacker, Craig; Doyele, Paul, **Upgranding and Repairing Networks**, Que Corporation, 1998, 1056pp;

[3] Sportack, Mark A, **Networking Essentials**, first edition, Sams Publishing, 1998, 575pp;

[4] Solomon, James D , **Mobile IP The Internet Unplugged**, Prentice Hall PTR & Upper Saddle River, 1998, 350pp;

[5] Kaufman, Charlie; Perlman, Radia; Speciner, Mike, **Network Security PRIVATE Communication in a PUBLIC World**, second edition, Prentice Hall PTR, 2002, 713pp;

[6] Hunt, Craig, **TCP/ IP Network Administration**, second edition, O`Reilly, United States of America, 1998, 612pp.

[7] Goldman, James E; Rawles Philip T, **Applied Data Communication A Business-Oriented Approach**, Third edition, John Wiley & Sons, Inc, United States of America, 2001, 692pp.

[8] Massingue, V.S, **Buiding Awareness and Supporting. African Universites in ICT Managment. The big ICT five (Strategy, Development, Acquisition, Implementation, Utilization, Service Managment)**, Central Impressora Editora de Maputo, 2003, 315 pp;

[9] Wright Gary R. , Stevens Richard W., **TCP/IP Illustrated The Implementation**, Volime 2, Addison-Wesley Longman, Inc, England, 1995, 1174pp;

[10] Marques José Alves, Guedes Paulo, **Tecnologia de Sistemas Distribuidos**,FCA-Editora de Informática, Porto,, 1998, 501pp;

[11] Cisco System at al, **Internet Working Technologies Hand book**, second edition, 2001;

[12] Política de Informatica, http://www.infopol.gov.mz/proj_pol/estado.htm; (consulta 17/03/04);

[13] Renato M.E. Sabbatini, **Aplicações na Internet**, Março de 2003 http://www.epub.org.br/informed/intern1.htm (consulta 17/03/04);

[14] RFC 15 19, **CIDR An Address Assignment And Aggregation Strategy**, Setembro de 1993, http://www.faqs.org/rfcs/rfc1519.html , (consulta no dia 26/07/04);

[15] RFC2460, **Internet Protocol Version 6 (IPv6),** Dezembro de 1998, http://www.faqs.org/rfcs/rfc2460.html, (consulta no dia 26/7/04);

[16] BR6Bone. **IPv6 Networks**, Março de 2003, http://www.6bone/, (consulta no dia 03/03/2004);

[17] Deering S, Hinden R, **Internet Protocol Version 6 (IPv6)** , fev de 2002, http://www.ietf.org/, (consulta dia 01/03/04);

[18] UEM, **Plano Estratégico**, Março de 2003, http://www.uem.mz (consulta no dia 06/07/04);

[19] RFC 2428, **FTP extensions for IPv6 and NATs,** Setembro de 1998, http://www.faqs.org/rfcs/rfc2428.html, (consulta no dia 23/12/04)

14 Glossàrio

Anycast: É um tipo de comunicação que participam um determinado grupo de nós, dos quais o nó origem envia seus pacotes para o nó mais proximo.

Backbone ou 6bone: É um conjunto de caminhos disponíveis para as redes locais ou regionais consiguirem interconeção a longas distâncias com outras redes.

Broadcast : Tipo de comunicação em que o nó origem envia seus pacotes para todos os outros na mesma rede.

Datagrama : Cabeçalho e dados no nível de transporte.

Firewall : Consiste de demarcação de um perímetro de segurança, visando estabelecer que a área interna por este delimitada deve ser protegida do que se situa fora de suas fronteiras.

General Packet Radio Service (GPRS) : É um padrão para comunicações wireless com velocidades acima de 115 Kilobits por segundo.

Multicast: Comunicação estabelecida num grupo em que o nó origem envia seus pacotes para múltiplos receptores.

Overhead : Conteúdo de um pacote adicional em relação aos dados que se pretendem transmitir.

Request for coments (RFC). Um documento técnico que define ou formaliza um padrão dentro da rede Internet.

Socket: Tipo especial de *file handler* que é usado pelo processo para solicitar serviços de rede ao sistema operativo.

Spoofing : Criação de pacotes IP com imitação do endereço de origem.

Unicast : Tipo de comunicação 1:1, onde participam dois nodos de cada vez.

Anexo A-Obtenção dos valores de redes e hosts possíveis para classes A, B e C

A classe A suporta 126 ($2^7 - 2$)/8 redes. A subtração de 2 é devido a rede 0.0.0.0 que é um roteador default e 127/8, que é reservado para " loopback " onde para cada rede podem ser alocados ($2^{24} - 2$), isto é 16 milhões de hosts, a subtração dos 2 hosts é devido a todos zeros e todos uns para " broadcast ".

O endereço /8 contém 2^{31} (2.147.483.648) endereços individuais e o espaço de endereçamento IPv4 contém no máximo 2^{32} (4.294.967.296) endereços. Assim pode se dizer que os /8 classes de endereçamento constituem 50% do total de IPv4 endereços unicast[13].

A classe B, tem 16 bits para o prefixo da rede, com os dois bits 10 mais significativos, mais os outros 14, seguidos de 16 bits para host.

Os endereços da classe B têm no máximo 2^{14} ^14 redes e para cada rede são definidos 65.534 ($2^{16} - 2$) hosts por rede. O endereço /16 contém 2^{30} (1.073.741.824) endereços individuais e o espaço de endereçamento IPv4 contém no máximo 2^{32} (4.294.967.296) endereços. Os /16 classes de endereçamento constituem 25% do total de IPv4 endereços unicast.

A classe C, tem 24 bits no prefixo de rede, com os bits 110 mais significativos, seguidos de 8 bits de host.

A classe contém no máximo 2.097.152 (2^{21}) redes, e para cada rede são 254 ($2^8 - 2$) hosts. O endereço /24 contém 2^{29} (536.870.912) endereços individuais, tendo em conta que o espaço de endereçamento IPv4 contém no máximo 2^{32} (4.294.967.296) endereços. Os /8 classes de endereçamento constituem 12.5% do total de IPv4 endereços unicast [13].

Como $8=2^3$, serão necessários 3 bits para enumerar as 8 sub-redes, neste exemplo a organização está a criar sub-redes no /24 então precisará ainda de mais 3 bits ou /27, este prefixo de rede estendido pode ser expresso em notação decimal como 255.255.255.224, figura B1

	Prefixo de rede				
193.1.1.0 =	11000001	00000001	00000001	000	00000000
	Prefixo da rede estendida				
255.255.255.224	11111111	11111111	11111111	111	00000
	27 bits				

Figura B1 : Exemplo de subnet mask para comprimento extendido

Os 27 bits do prefixo de rede estendido deixa 5 bits para se definir o endereço dos hosts em cada sub-rede, isto significa que cada sub-rede com 27 bits de prefixo, representa um bloco contíguo de 2^5 (32) endereços IP, contudo todos os zeros e todos os uns não podem ser alocados, então há 30 ($2^5 - 2$) possíveis hosts para cada sub-rede.

O segundo passo seria a **definição do número de sub-redes,** 8 sub-redes serão enumeradas de 0 a 7. Em geral para definir o número de sub-redes, são alocados os bits correspondentes aos números decimais de 0 a 7, no lugar dos 3 bits na extensão exemplo para a sub-rede # 6 a representação binária de 6 (110) é alocada nos 3 bits do campo da sub-rede.

As sub-redes correspondentes a este exemplo são dadas a baixo:

Rede base: 11000001.00000001.00000001.00000000 = 193.1.1.0/24

Subnet # 0: 11000001.00000001.00000001.**000**00000 = 193.1.1.0/27

Subnet # 1: 11000001.00000001.00000001.**001**00000 = 193.1.1.32/27

Subnet # 2: 11000001.00000001.00000001.**010**00000 = 193.1.1.64/27

Subnet # 3: 11000001.00000001.00000001.**011**00000 = 193.1.1.96/27

Subnet # 4: 11000001.00000001.00000001.**100**00000 = 193.1.1.128/27

Subnet # 5: 11000001.00000001.00000001.**101**00000 = 193.1.1.160/27

Subnet # 6: 11000001.00000001.00000001.**110**00000 = 193.1.1.192/27

Subnet # 7: 11000001.00000001.00000001.**111**00000 = 193.1.1.224/27

Anexo B-Considerações no Desenho de Sub-Redes

O primeiro passo no processo do planeamento é levar o número máximo de sub-redes requeridas e transformá-los na potência de base 2.

Por exemplo se a organização precisa de 9 sub-redes, 2^3 ou 8 não nos daria o número de sub-redes suficientes, nesse caso precisaríamos de 2^4 ou 16. para isto é necessário que o administrador tenha em mente o espaço futuro de crescimento.

O 2° passo é assegurar que há endereços de hosts suficientes para a sub-rede maior da organização. Se por exemplo a maior sub-rede necessitar de 50 endereços de host hoje 2^5 ou 32 não seriam suficientes, neste caso precisaríamos de 2^6 ou 64 hosts.

O último passo iria assegurar que a organização tivesse a alocação dos endereços providenciando bits suficientes para explorar o plano de endereçamento requerido.

A motivação da definição de várias classes de redes suportando quantidades diversas de nós (hosts) é evidente: permitir uma melhor adaptação à realidade das organizações. Na prática verifica-se contudo que três classes se revelam insuficientes para todo o tipo de situações.

Quando um organização necessita de uma grande quantidade de redes IP a solução consiste na subdivisão de uma rede atribuída em várias sub-redes. O processo é meramente interno e não deve transparecer para o exterior, consiste em reservar alguns dos bits mais significativos da parte de "host" para identificar a sub-rede. Por exemplo uma rede de classe C utiliza 8 bits para o host, destes oito podemos reservar alguns dos mais significativos para sub-rede e os restantes para host.

Exemplo : uma organização foi atribuída um número de rede igual a 193.1.1.0/24 e precisa de definir 6 sub-redes, tendo em conta que a maior sub-rede da organização tem 24 hosts. Pretende-se que se indique o grupo de sub-redes e de hosts que irão constituir a rede.

O primeiro passo é determinar o número de bits necessários para definir 6 sub-redes. Tendo em conta que com o endereço de rede só se podem criar sub-redes através de limites binários, as sub-redes serão criadas em blocos de potência de base 2 ex. [$2(2^0)$ $4(2^2)$ $8(2^3)$ $16(2^4)$ ....]. Neste caso seria impossível definir um bloco de endereço IP que contenha exatamente 6 sub-redes, ficando assim definido um bloco de 8 (2^3) e teria nesse caso duas sub-redes para uso futuro.

Uma maneira simples de verificar se as sub-redes estão corretas é verificar se todas as sub-redes são múltiplos da sub-rede #1, neste caso os múltiplos de 32 são 0, 32, 64...

Do meso modo são definidos os hosts válidos para cada sub-rede, o exemplo asseguir mostra endereços possíveis para a sub-rede # 2

Subnet # 2: 11000001.00000001.00000001.**010**00000 = 193.1.1.64/27

Host # 1: 11000001.00000001.00000001.001**00000** = 193.1.1.65/27

Host # 2: 11000001.00000001.00000001.010**00010** = 193.1.1.66/27

Host # 3: 11000001.00000001.00000001.011**00011** = 193.1.1.67/27

Host # 4: 11000001.00000001.00000001.100**00100** = 193.1.1.68/27

Host # 5: 11000001.00000001.00000001.101**00101** = 193.1.1.69/27

.

.

.

Host # 27: 11000001.00000001.00000001.001**11011** = 193.1.1.91/27

Host # 28: 11000001.00000001.00000001.010**11100** = 193.1.1.92/27

Host # 29: 11000001.00000001.00000001.011**11101** = 193.1.1.93/27

Host # 30: 11000001.00000001.00000001.100**11110** = 193.1.1.94/27

Anexo C-IPv6 no Windows

Após feito o download do tpiv6-001205.exe é necessário seguir aos passos para instalação:

1) Faz-se o logon no Windows 2000 com um usuário que tenha privilégios de administrador local.

2) Salva-se o ficheiro obtido via download (tpipv6-001205.exe) em uma pasta local.

3) Usando o Windows Explorer deve-se executar o arquivo setup.exe localizado no diretório onde foi descompactado o pacote IPv6 que neste caso é c:\IPV6Kit, porém há situações em que deverá ser precizo abrir uma janela MS-DOS entrar no diretório c:\IPV6kit e executar o seguinte comando: "c:\IPv6kit\setup.exe -x files".

Esse comando irá descompactar a instalação do pacote dentro de um sub-diretório chamado files. De seguida será necessário abrir o ficheiro hotfix.inf no diretório c:\IPV6Kit\files em um editor de texto. Dentro deste arquivo existe uma secção chamada [Version] e nessa secção existe uma linha com a seguinte variável NTServicePackVer=256 que deve ser alterada NTServicePackVer=512. Após este procedimento salva-se o arquivo.

4) No Diretório c:\IPV6Kit\files executa-se o arquivo hotfix.exe usando o Windows Explorer e faz-se o restart do computador depois de instalado o pacote.

5) Após o restart do computador é necessário adicionar o serviço a rede, clicando em start, settings, Networks and Dial-up Connections, com o botão diteiro do mouse selecciona-se a conexão baseada em ethernet, que geralmente tem o nome "*Local Area Connection*" e escolhe-se a opção properties.

6) Na janela que aparecer deve-se clicar em install, depois clicar em protocol e depois em add, se o pacote do IPV6 foi instalado correctamente irá aparecer na lista de protocolos "Microsoft IPv6 Protocol" .

Seguindo os passos descritos acima, nesse ponto o computador já estará com o suporte ao protocolo IPv6 funcionando corretamente. A instalação desse pacote irá atualizar alguns programas, dll`s e acrescentar comandos para gerenciar e configurar o protocolo ipv6. O driver do protocolo ipv6 (tcpip6.sys) é instalado no diretório \winnt\system32\drivers.

Abiblioteca Winsock para endereços da família INTE6 (wship6.dll) e todas as aplicações e ferramentas são Instaladas no diretório \winnt\system32.

As ferramentas instaladas são:

- IPV6.exe: o comando ipv6.exe, com excepção da configuração de IPsec, é o

responsável por todas as configurações do protocolo ipv6, pode ser usado para fazer query's,configurar interfaces, endereços e rotas.

- PING6.exe: este comando é o conhecido ping utilizado para IPv4, só que suporta apenas endereços ipv6.
- TRACERT6.exe: este comando é para traçar as rotas através de roteadores IPV6.
- TTCP.exe: esta ferramenta é usada para mandar pacotes TCP e pacotes UDP entre dois pontos, ttcp.exe também suporta ipv4 e ipv6.
- 6TO4CFG.exe: ferramenta usada para configurar a conectividade do protocolo ipv6 em cima de uma rede ipv4.
- IPSEC6.exe, Ferramenta Utilizada para configurar as políticas e configurações

relacionadas a segurança do protocolo.

Alguns aplicativos foram actualizados para suportar tanto ipv4 quanto ipv6. Com as DLL`s atualizadas é possível agora acessar sites HTTP utilizando ipv6, com o novo ftp client é possível fazer transferencias em cima de ipv6, fazer conexões telnet com servidores ipv6, e também através da nova versão do Telnet Server é possível fornecer portas de conexão para clientes ipv6.

C.1-Configuração do IPv6 no Windows 2000

O suporte ao protocolo IPv6 está instalado, porém, não está configurado adequadamente, abaixo serão mostradas algumas queries para ver Como está a configuração do protocolo e posteriormente será demostrado Como configurá-lo.

O comando ipv6.exe como já mencionado acima é o comando para realizar as configurações do protocolo ipv6, abaixo é executada uma query com o comando ipv6 if, essa opção extrairá informações sobre as interfaces de rede ipv6.

```
C:\>ipv6 if

Interface 5 (site 1): 6-over-4 Virtual Interface

  uses Neighbor Discovery

  link-level address: 192.168.254.131

    preferred address fe80::c0a8:fe83, infinite/infinite

    multicast address ff02::1, 1 refs, not reportable

    multicast address ff02::1:ffa8:fe83, 1 refs, last reporter

  link MTU 1280 (true link MTU 65515)

  current hop limit 128

  reachable time 23000ms (base 30000ms)

  retransmission interval 1000ms

  DAD transmits 1

Interface 4 (site 1): Local Area Connection

  cable unplugged

  uses Neighbor Discovery

  link-level address: 00-30-f1-32-a2-ca

    preferred address fe80::230:f1ff:fe32:a2ca, infinite/infinite

    multicast address ff02::1, 1 refs, not reportable

    multicast address ff02::1:ff32:a2ca, 1 refs, last reporter

  link MTU 1500 (true link MTU 1500)

  current hop limit 128

  reachable time 16000ms (base 30000ms)

  retransmission interval 1000ms

  DAD transmits 1

Interface 3 (site 1): Local Area Connection 2

  uses Neighbor Discovery

  link-level address: 00-30-f1-39-a2-b0

    preferred address 3ffe:306:11:2:230:f1ff:fe39:a2b0, 2591994s/604794s
(addrco
```

```
nf)
    preferred address fe80::230:f1ff:fe39:a2b0, infinite/infinite
    multicast address ff02::1, 1 refs, not reportable
    multicast address ff02::1:ff39:a2b0, 2 refs, last reporter
  link MTU 1500 (true link MTU 1500)
  current hop limit 64
  reachable time 35500ms (base 30000ms)
  retransmission interval 1000ms
  DAD transmits 1
Interface 2 (site 0): Tunnel Pseudo-Interface
  does not use Neighbor Discovery
  link-level address: 0.0.0.0
    preferred address ::192.168.254.131, infinite/infinite
  link MTU 1280 (true link MTU 65515)
  current hop limit 128
  reachable time 0ms (base 0ms)
  retransmission interval 0ms
  DAD transmits 0
```

Por defeito o computador que tiver apenas uma placa de rede serão criadas quarto interfaces ipv6.

A interface 1 com o nome de loopback Pseudo-Interface vem com o endereço ::1, esta interface tem a mesma função do loopback do ipv4, isto é, o endereço 127.0.0.1 em ipv4 é equivalente ao endereço ::1.

A interface 2 também denominada Tunnel Pseudo-interface é responsável pela tradução do endereço ipv4 na interface para que a mesma possa ser acessível pelo protocolo

IPv6, com isso possibilita a implementação do tunelamento sobre o ipv4. Sem tal artificio, não é possível que um endereço ipv6 seja visto diretamente por um endereço IPv4. É responsável pela criação deste pseudo túnel para cada endereço ipv4 que seja utilizado.

A interface 3 também denominada 6-over-4 virtual interface é responsável pela criação de uma interface virtual com um endereço ipv6 associado ao seu ipv4.

A interface 3 também cria automaticamente os endereços para multicast os quais correspondem aos endereços ipv4 da classe D. Neste tipo de interface o link-level address corresponde a um dos endereços ipv4 do computador. Nesta interface corre um serviço denominado neighbor discovery o qual descobre outros endereços ipv6 na rede.

A interface 4, também denominada Local Area Connection (varia de acordo com o

nome das interfaces ipv4) possui como link-level address o endereço físico da placa de rede (MAC address), de acordo com a quantidade de endereços MAC (placas de rede) disponíveis, são criadas para cada uma novas interfaces. Na mesma também corre o serviço de neighbor discovery e os endereços para multicast.

Por default é criada uma rota para tunelamento automático. Para a visualização das

rotas é utilizado o comando ipv6 rt conforme abaixo:

```
C:\>ipv6 rt
::/96 -> 2 pref 0 (lifetime infinite)
```

A partir deste momento tem-se o suporte para ipv6 instalado já com sua configuração

Anexo D- Entrevista

Objectivo da entrevista: obter uma visão sobre o estágio actual da rede da UEM, etender a estrutura da rede e levanter necessidades que justifiquem a implementação do IPv6 na rede da UEM, tendo em linha de orientação o propósito do trabalho Implementação de uma rede com suporte ao IPv6 na mesma rede.

1-Como é que a rede encontra-se constituida (número de redes locais e tipo)?

2-Como é feita a ligação entre as diferentes redes locais na UEM?

3-Quais as dificuldades enfrentadas na gestão da rede actual?

4-Quais os servicos de rede e equipamentos utilizadados?

5-Existe suporte a aplicações a tempo real?

6-Quais os objectivos da instituição com relação a estes services?