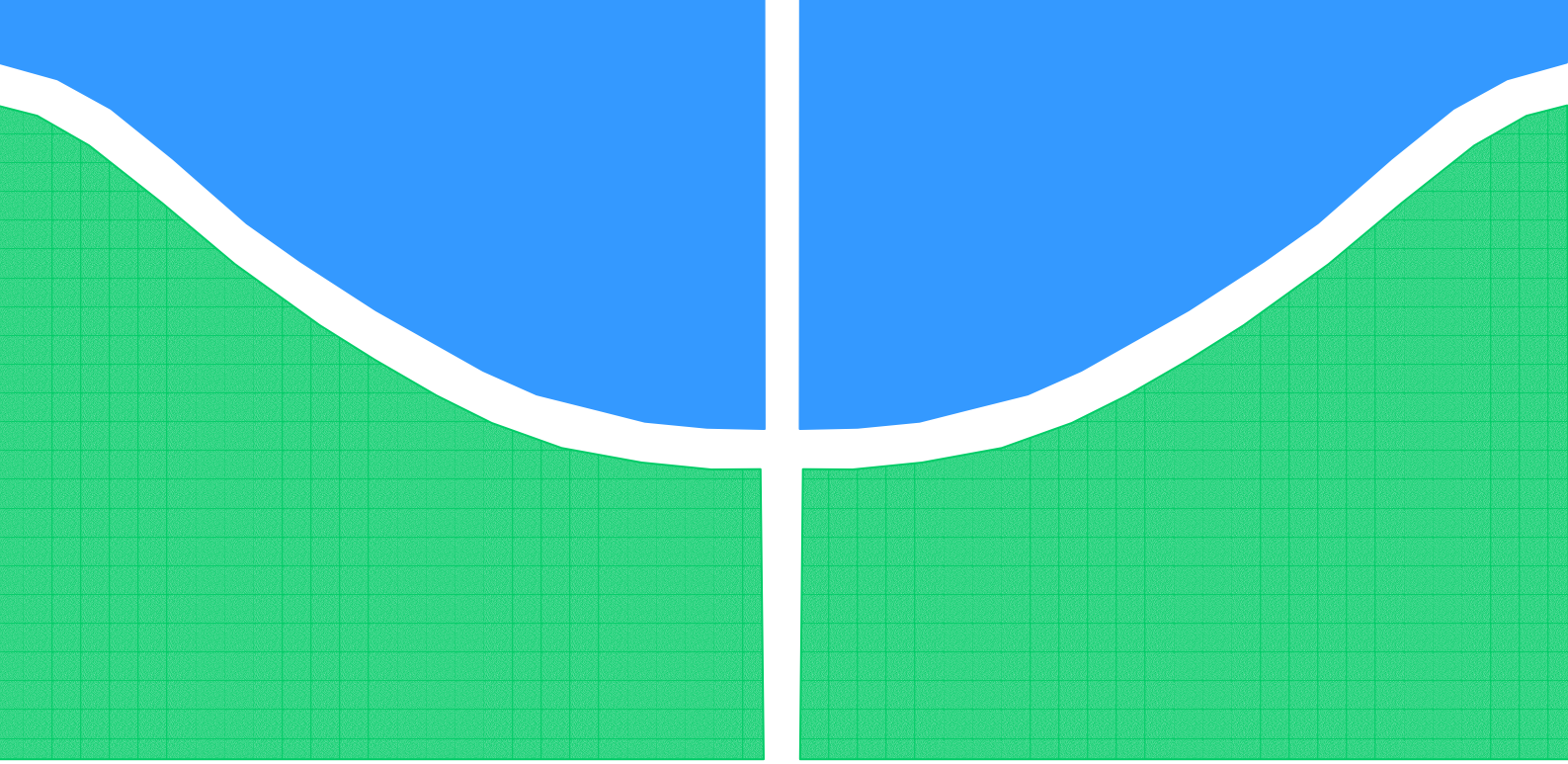


TRABALHO DE GRADUAÇÃO

APERFEIÇOAMENTO DO MOBILE IP COM REGISTRO REGIONAL E INTEGRAÇÃO COM PROTOCOLO MPLS

André Ricardo de Pinho Ronzani
Bruno Henrik Costa Faria

Orientador:
Alex Helder

Brasília, dezembro de 2007.

UNIVERSIDADE DE BRASILIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASILIA
Faculdade de Tecnologia
Departamento de Engenharia Elétrica

TRABALHO DE GRADUAÇÃO

Aperfeiçoamento do Mobile IP com Registro Regional e integração com Protocolo MPLS

André Ricardo de Pinho Ronzani
Bruno Henrik Costa Faria

Relatório submetido como requisito parcial para obtenção
do grau de Engenheiro Eletricista

Banca Examinadora

Dedicatória(s)

Ao meu pai, que me ensinou a nunca perder a alegria de viver e à minha mãe, que me ensinou a acreditar sempre na vida.

André R P Ronzani

Dedico este trabalho a minha família, exemplo sublime de amor incondicional.

Bruno H C Faria

Agradecimentos

Agradeço a Deus por ter me dado forças para enfrentar todas as barreiras impostas pela vida e pelo curso de Engenharia Elétrica no decorrer desses 5 anos. Agradeço a minha família por ter me ajudado em tudo que precisei. Agradeço ao professor Alex Helder por sua paciência e dedicação. Agradeço ao amigo Marco Aurélio pela ajuda nos momentos difíceis. Agradeço a namorada Carol pela dedicação e compreensão em todos os momentos. Agradeço de modo especial ao grande amigo Bruno, com quem compartilhei muitos momentos de trabalho e diversão. Agradeço aos meus irmãos Bento, Tales, Ítalo, Luiz, Ana, Pedro, Álvaro e todos os outros pelos momentos especiais que passamos juntos.

André R. P. Ronzani

Agradeço a Deus por ter me concedido o dom da sabedoria. À minha família que me acolhe e me incentiva em todas as etapas da minha vida. Ao meu grande amigo André Ricardo que faz as horas de mais árduo trabalho serem sempre divertidas e também pelo exemplo que ele é de pessoa mais querida por todos. À todos as pessoas especiais que fazem parte de minha vida e fazem de mim uma pessoa abençoada. Ao nosso paciente e companheiro orientador Alex Helder. Ao Maurélio por sempre se mostrar disposto em nos ajudar com o software.

Bruno H C Faria

RESUMO

APERFEIÇOAMENTO DO MOBILE IP COM REGISTRO REGIONAL E INTEGRAÇÃO COM PROTOCOLO MPLS

O presente trabalho tem como objetivo demonstrar aperfeiçoamentos do protocolo *Mobile IP* com o intuito de aumentar a eficiência de *handoff* e no transporte de pacotes enviados a um dispositivo móvel situado fora de sua rede de origem. Para isso, são demonstradas implementações físicas e implementações feitas por meio de softwares apropriados. No decorrer do trabalho, primeiramente, são descritos o Mobile IP e o MPLS para que se tenha a base teórica necessária ao entendimento das implementações. Em seguida, são demonstrados as implementações dos aperfeiçoamentos propostos e seus resultados.

ABSTRACT

MOBILE IP IMPROVEMENT WITH REGIONAL REGISTRATION AND INTEGRATION WITH MPLS PROTOCOL

This work aims to demonstrate improvements in Mobile IP protocol in order to increase efficiency in its handoff and transport of packages sent to a mobile device outside of their home network. To do so, are shown physical implementations and implementations made by using appropriate software. During the work, primarily, are described the Mobile IP and MPLS to take the theoretical basis necessary for the understanding of the implementations. Then, are demonstrated the implementations of the proposed improvements and their results.

SUMÁRIO

LISTA DE TABELAS

LISTA DE FIGURAS

LISTA DE SÍMBOLOS, NOMENCLATURA E ABREVIAÇÕES

AP	*Access point* — Ponto de acesso.
BGP	*Border gateway protocol* — Protocolo de gateway de borda
CN	*Correspondent node* — Nó correspondente
CoA	*Care-of address* — Endereço residente
DHCP	*Dynamic host configuration protocol* — Protocolo de configuração dinâmica de Servidor
FA	*Foreign agent* — Agente estrangeiro
FEC	*Forwarding equivalence class* — Classe de equivalência de encaminhamento
GFA	*Gateway foreign agent* — Agente estrangeiro *gateway*
GPL	*GNU general public license* — Licença pública geral GNU
HA	*Home agent* — Agente nativo
ICMP	*Internet control message protocol* — Protocolo de mensagem de controle da Internet
IETF	*Internet engineering task force* — Força-tarefa de engenharia da Internet
ILM	*Incoming label map* — Mapa de rótulos entrantes
IP	*Internet Protocol* — Protocolo de Internet
LAN	*Local area network* — Rede de área local
LDP	*Label distribution protocol* — Protocolo de distribuição de rótulos
LER	*Label edge router* — Roteador de bordas dos rótulos
LIB	*Label information base* — Base de informações de rótulos
LSP	*Label switched path* — Percurso comutado por rótulo
LSR	*Label switching router* — Roteador de comutação por rótulo
MIP	*Mobile IP* — IP móvel
MN	*Mobile node* — Nó móvel
MPLS	*Multiprotocol label switching* — Multi-protocolo de comutação de rótulos
NAT	*Network address translation* — Tradução de endereços de rede
OSPF	*Open shortest path first* — Caminho mais curto iniciado primeiro
QoS	*Quality of service* — Qualidade de serviço
RFA	*Regional foreign agent* — Agente estrangeiro regional
RFC	*Request for comments* — Chamada para comentários

RSVP	*Resource reservation protocol* — Protocolo de reserva de recursos
TCP	*Transmission control protocol* — Protocolo de controle de transmissão
UDP	*User datagram protocol* — Protocolo de datagrama de usuário
WAN	*Wide area network* — Redes de área ampla
W-LAN	*Wireless LAN* — Rede LAN sem fio

1 – INTRODUÇÃO

Nos dias de hoje, com o grande desenvolvimento e popularização dos computadores portáteis e a necessidade de mobilidade rápida, eficiente e descomplicada, cresce a demanda por recursos que maximizem a eficiência das redes sem fio. Atualmente, já convivemos com alguma mobilidade provida por protocolos das camadas física e de enlace de dados, como é o caso do já popular IEEE 802.11 (Ethernet sem fio). Entretanto, 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.

Na tentativa de resolver esse tipo de problema, o IETF (*Internet Engeneering Task Force*) propôs o protocolo Mobile IP - MIP RFC 3344 [22] (Mobilidade em IP) que, por sua vez, possibilita que um nó móvel passe de uma rede para outra sem que as sessões estabelecidas sejam interrompidas e permitindo que outras novas sejam criadas. Sendo assim, o protocolo permite que haja o deslocamento entre sub-redes e que não se perca o endereço IP utilizado. O grande desafio do Mobile IP é evitar que a conexão seja quebrada no evento da movimentação.

O Mobile IP é baseado em um mecanismo que utiliza agentes que administram a mobilidade de um dispositivo móvel, por exemplo, um notebook. Basicamente, esses agentes se dividem em agente nativo, o qual pertence a mesma rede que o dispositivo móvel e agente estrangeiro, o qual pertence a uma rede fora do domínio da rede nativa. Quando o nó móvel se desloca de sua rede nativa para uma rede estrangeira, faz um registro nesta rede, enviando ao agente estrangeiro informações sobre si e sobre seu agente nativo. O agente estrangeiro, por sua vez, envia uma requisição baseada nessas informações ao agente nativo requisitando que seja feito o registro do nó móvel em sua rede, conservando seu endereço IP. Se a resposta a essa requisição for positiva, um túnel IP-em-IP será estabelecido entre o dispositivo móvel e seu agente nativo. Isso faz com que toda mensagem que for destinada ao dispositivo, ao chegar à rede nativa, seja capturada pelo agente e reenviada através do túnel até que o nó móvel a receba.

No mecanismo do Mobile IP existem dois principais problemas que o tornam pouco eficiente no que diz respeito a velocidade e robustez. O primeiro deles está relacionado ao atraso causado pela movimentação do dispositivo móvel entre sub-redes de um mesmo domínio da rede estrangeira. No MIP tradicional, ao movimentar-se dentro de qualquer rede estrangeira trocando de sub-rede, o dispositivo móvel necessita fazer outro registro no

agente nativo para que seja restabelecida a conexão entre eles. O atraso causado por essa nova requisição é chamado de "atraso de *handoff*" que, dependendo da distância entre as redes nativa e estrangeira, pode ser demasiado grande, inviabilizando o uso do MIP para algumas aplicações críticas, como trafego de voz e vídeo.

O segundo principal problema do MIP está no fato de que todos os pacotes enviados ao nó móvel devem passar pelo agente nativo para, em seguida, serem encaminhados ao seu destino. Muitas vezes, o percurso fica muito maior do que se os pacotes fossem entregues diretamente da origem ao dispositivo móvel. Este grande percurso pode causar efeitos maléficos como perda de pacotes e atraso no envio deles.

Como solução para minimizar estes problemas e aumentar a eficiência do MIP, duas propostas são estudadas no presente trabalho.

Uma dessas propostas é a utilização do Registro Regional ou Mobile IP Hierárquico. Este mecanismo maximiza a eficiência do *handoff*, utilizando o gerenciamento da micromobilidade, ou seja, se o nó móvel se movimentar de uma sub-rede para outra, a autenticação só ocorre regionalmente, tornando-se desnecessário repetir a autenticação na rede nativa.

A outra proposta é a utilização do MPLS no lugar dos túneis IP-em-IP no transporte de pacotes entre a rede nativa e a rede estrangeira. Desta forma, seria melhorada a eficiência e conseqüentemente, a velocidade no transporte de pacotes entre as redes.

O foco deste trabalho é demonstrar por meios teóricos e práticos essas propostas afim de chegar a conclusões concretas a respeito de cada uma delas.

Na apresentação dos estudos e resultados das pesquisar e implementações feitas, este trabalho foi dividido em cinco capítulos. Após esta breve introdução, o capítulo seguinte explica o funcionamento do Mobile IP, dando ênfase a nova RFC 4857 [24] publicada em junho de 2007, que trata especificamente do Mobile IP Hierárquico.

No terceiro capítulo, é dada a teoria sobre o MPLS com intuito de se obter uma base completa para o entendimento das implementações feitas usando-se o protocolo.

O capítulo seguinte é onde são mostrados detalhadamente todos os testes, implementações e simulações feitas no trabalho. É nele que estão mostrados os meios utilizados para a obtenção dos resultados.

Finalmente, no quinto e último capítulo é onde são discutidos os resultados e feitas as conclusões sobre o trabalho.

2 - MOBILE IP e MOBILE IP HIERÁRQUICO

Neste capítulo serão teorizados todos os aspectos relevantes a respeito do Mobile IP, que vão desde seu funcionamento até seu principal aprimoramento, o Mobile IP Hierárquico, que é o foco do capítulo.

2.1 - FUNCIONAMENTO DO MOBILE IP

Para entender como funciona o IP Móvel, é preciso, inicialmente, entender como é a estrutura básica da rede e seus agentes. A estrutura é formada basicamente por duas redes distintas interconectadas por um roteador ou mesmo a Internet. Uma dessas redes é chamada de rede nativa (Home Network), que é a rede à qual o nó móvel pertence. A outra rede é chamada de rede estrangeira (Foreign Network), que é a rede na qual o nó móvel irá se conectar para comunicar-se com sua rede nativa. Essas duas redes são compostas de agentes que têm a capacidade de verificar a presença do nó móvel na rede e de manter ativa a conexão com este dispositivo. Segundo a RFC 3344 [22], os agentes e suas características básicas estão listados abaixo:

- Agente nativo (Home Agent - HA): Roteador da rede nativa com o qual são estabelecidos os túneis para entrega de pacotes do nó móvel quando este está em uma rede estrangeira. Também tem a função de mandar informações para que os Agentes Estrangeiros o localizem.
- Agente estrangeiro (Foreign Agent - FA): Roteador em uma rede estrangeira na qual o nó móvel está conectado que fornece serviços de roteamento enquanto este estiver registrado. Ele é o dispositivo que estabelece os túneis de comunicação com o agente nativo. Para o nó móvel, este agente pode servir simplesmente como um roteador padrão.
- Nó móvel (*Mobile Node* - MN): Dispositivo que se movimenta entre as diferentes redes e sub-redes sem que seu endereço IP seja modificado. Pode ser comparado com um notebook que se movimenta fisicamente de uma rede WI-FI para outra.

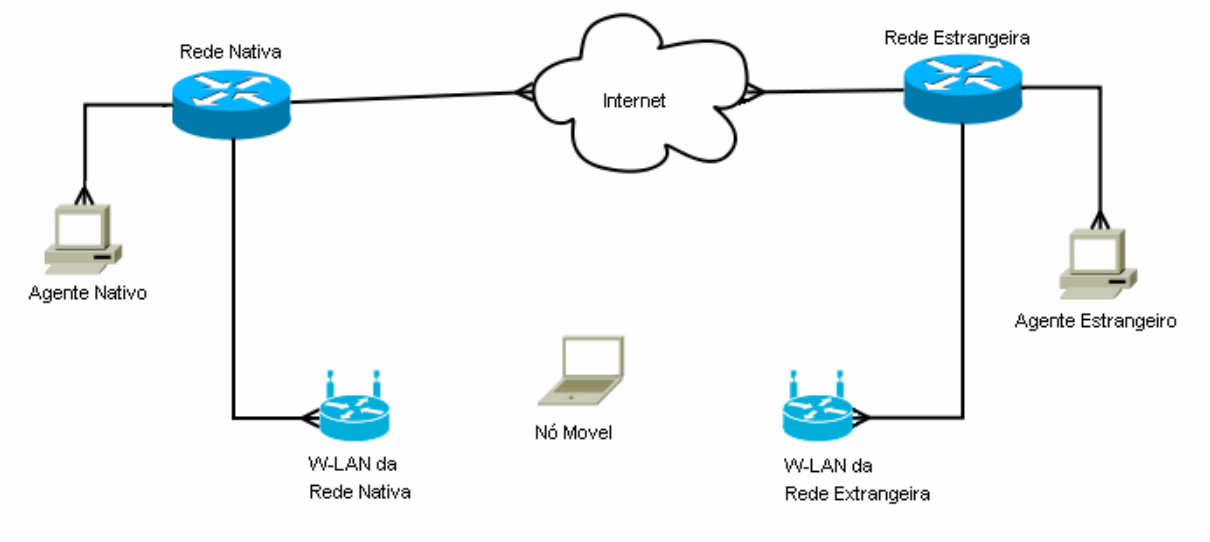


Figura 2.1 – Estrutura básica da rede Mobile IP

A figura acima representa um exemplo básico de uma rede que utiliza o Mobile IP. Nela, o nó móvel se conecta aos roteadores das redes Nativa e Estrangeira por meio de pontos de acesso (*access point*). Apesar do protocolo poder ser implementado em redes *ethernet* comuns (IEEE 802.3), suas funções têm mais utilidade se ele for usado em redes sem fio, pois assim, o usuário tem maior mobilidade, podendo passar rápido e descomplicadamante de uma rede para outra.

Basicamente, o funcionamento do Mobile IP ocorre da seguinte forma: Se o nó móvel está conectado diretamente a rede nativa, ele se autentica automaticamente e utiliza os recursos da rede normalmente, como em uma rede comum. Se o nó móvel está conectado em uma outra rede diferente da sua rede nativa, esta é considerada uma rede estrangeira. No momento da conexão, o nó móvel envia as informações do seu endereço e do endereço de seu agente nativo ao agente estrangeiro. Este, por meio dessas informações tenta entrar em contato com a rede nativa e fazer a autenticação do nó móvel. Se a autenticação for bem sucedida, se inicia um tunelamento de pacotes entre o agente estrangeiro e o agente nativo.

Os pacotes que chegam à rede nativa e são endereçados ao nó móvel, são capturados pelo agente nativo e enviados pelo túnel até o agente estrangeiro, o qual os envia ao nó móvel. Os pacotes gerados pelo nó móvel são enviados ao agente estrangeiro, o qual os envia ao destino de duas formas: por Tunelamento Triangular (regras comuns de TCP/IP) ou por Tunelamento Reverso (Passando pelo agente nativo). Ao analisar as duas formas de envio, a conclusão mais rápida a que se chega é que o Tunelamento Triangular teria um atraso muito menor que o Tunelamento Reverso, mas, em algumas redes, podem existir

roteadores que descartem pacotes com endereço de origem diferente do endereço da rede nativa, fazendo-se necessário o uso do Tunelamento Reverso.

A seguir, daremos uma explicação rápida dos três procedimentos mais importantes segundo a RFC 3344 [22]: Descoberta dos agentes, Registro e Roteamento.

2.1.1 Descoberta dos agentes

Ao conectar-se a alguma rede, o nó móvel necessita de um mecanismo que identifique em qual rede ele está, se é uma rede nativa ou uma rede estrangeira, além de anunciar quem ele é. Para isso são utilizadas mensagens de aviso tanto dos agentes da rede, quanto do próprio nó móvel. São eles o anúncio do agente (*agent advertisement*) e a solicitação de agente (*agent solicitation*). O *agent advertisement* é mandado para que o nó móvel detecte qual é o agente da rede na qual está conectado e a *agent solicitation* é mandada para que o agente da rede perceba que o nó móvel está conectado a ela.

Este método permite que o nó móvel receba as informações do endereço residente (*care-of-address* - CoA) oferecido pelo agente estrangeiro. O *care-of-address* é o endereço IP indicador de qual rede o agente nativo encontrará o nó móvel. Sendo assim, após o registro deste, o agente nativo se comunicará diretamente com ele através de um túnel IP-em-IP feito a partir desse endereço para o envio de informações ao nó móvel.

A descoberta dos agentes pode ser feita de duas formas, a primeira e mais usada é a que os agentes mandam anúncios periodicamente. Assim, ao entrar em uma nova rede, o nó móvel perceberá rapidamente em que rede está conectado e quais são os endereços residentes disponíveis. A segunda forma consiste em configurar os agentes para que só enviem um anúncio quando receberem uma solicitação do nó móvel. Desta forma, sobrecarrega-se menos a rede, mas a conexão se dá com um atraso um pouco maior.

Abaixo é mostrado um exemplo de mensagem de anúncio de um agente estrangeiro visto do nó móvel.

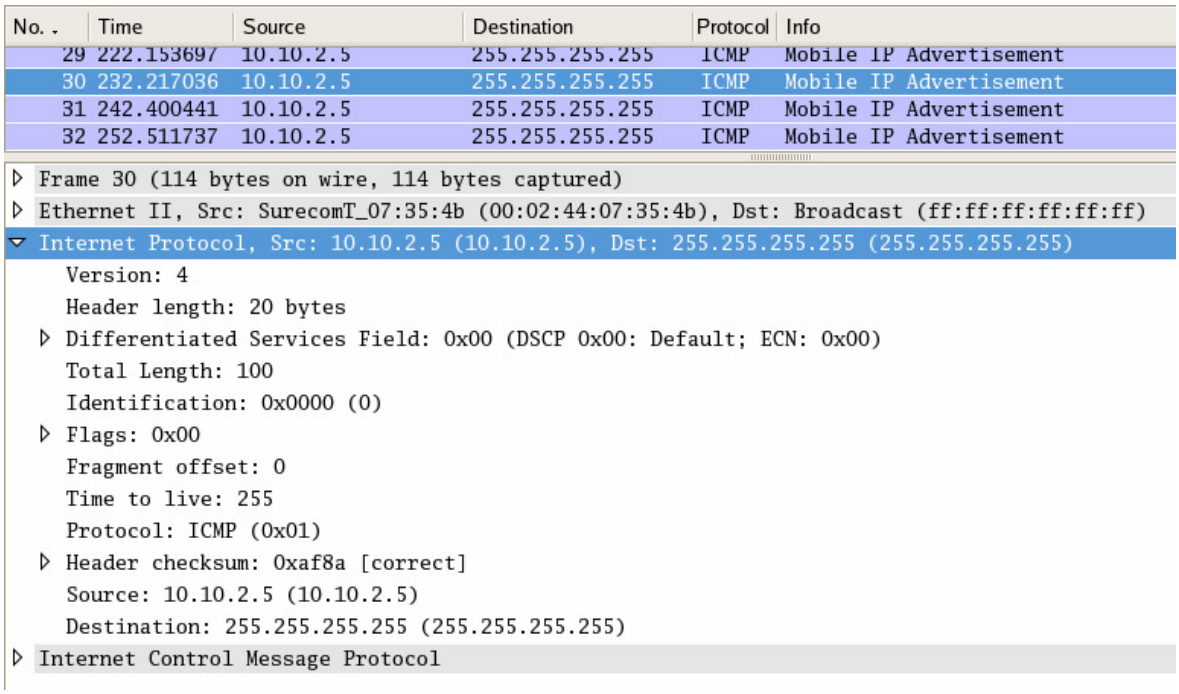


Figura 2.2 – Mensagem de anúncio de um agente estrangeiro.

Como podemos observar, a mensagem de anúncio (*advertisement*) é mandada em *broadcast* (*destination*: 255.255.255.255) pelo FA cujo endereço de rede é 10.10.2.5. Também podemos observar que este é um cabeçalho TCP-IP comum, o qual é usado neste tipo de mensagem.

2.1.2 Registro

Depois de descobertas a rede na qual o nó móvel está conectado e seu agente, o próximo passo é o registro nesta rede. É no registro que o dispositivo móvel fornace ao seu agente nativo informações de como alcançá-lo. Neste método e, de acordo com a RFC 3344 [22], o nó móvel pode executar as seguintes ações:

- Solicitar o encaminhamento de serviços quando visitando uma rede estrangeira;
- Informar ao seu agente nativo o seu *Care-of-Address* atual;
- Renovar seu registro que está prestes a expirar;
- Encerrar seu registro na rede estrangeira quando voltar a sua rede nativa.

O Mobile IP define dois procedimentos de registro diferentes, um para o registro do MN em uma rede estrangeira e outro para o registro em sua rede nativa.

No processo em que o nó móvel se conecta diretamente em sua própria rede nativa, são trocadas apenas duas mensagens:

1. A mensagem de requisição de registro do nó móvel no agente nativo;
2. A mensagem de resposta a esta requisição, permitindo ou não o registro do dispositivo a rede.

Figura 2.3 – Registro do MN quando conectado a rede nativa.

No processo em que o nó móvel se conecta a uma rede estrangeira, são seguidos quatro passos:

1. O nó móvel envia uma requisição de registro ao agente estrangeiro para começar o processo de registro;
2. O agente estrangeiro processa a requisição e a envia ao agente nativo;
3. O agente nativo manda uma resposta de registro para o agente estrangeiro permitindo ou não a conexão;
4. O agente estrangeiro processa a resposta de registro e a direciona ao nó móvel informando o resultado de sua requisição.

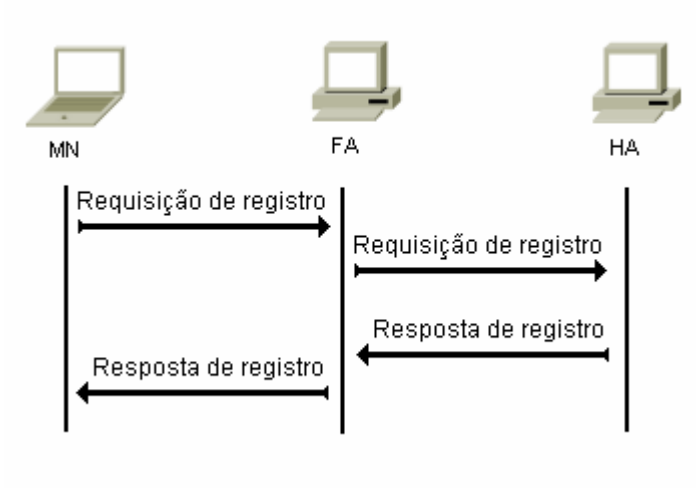


Figura 2.4 - Registro do MN quando conectado a uma rede estrangeira.

As mensagens de requisição são pacotes UDP que são direcionadas a porta 434 dos agentes. Já as mensagens de resposta, são pacotes UDP direcionados a uma porta variável, cabendo ao desenvolvedor defini-la. Os campos principais destas mensagens UDP são os seguintes:

- **Tipo:** Tamanho de um *byte*, indica se é uma mensagem de resposta ou requisição de registro.
- **Código:** Tamanho de um *byte*, indica o resultado do registro.
- **Tempo de vida:** Tamanho de dois *bytes*, indica o tempo que falta para o registro expirar.
- **Endereço nativo:** Tamanho de quatro *bytes*, indica o endereço IP do nó móvel.
- **Agente nativo:** Tamanho de quatro *bytes*, indica o endereço IP do Agente nativo.
- **Identificação:** Tamanho de oito *bytes*, usado na comparação de requisições com respostas de registro, para que haja uma proteção contra ataques de respostas a mensagens de registro. É baseado no campo "Identificação" da mensagem de requisição de registro e no tipo de proteção utilizada na configuração de segurança do sistema.

2.1.3 Roteamento

Na situação em que o nó móvel está em sua rede nativa e já fez a requisição de registro obtendo sucesso, ocorrerá o acesso à rede normalmente, sem que haja redução de desempenho. Já na situação em que o nó móvel está em uma rede estrangeira e já fez a requisição de registro obtendo permissão para usar os recursos da rede, a comunicação com o agente nativo se dará na forma de um Túnel IP-em-IP. Para isso, é importante que, tanto os agentes Nativos quanto os agentes Estrangeiros tenham suporte a tunelamento de datagramas usando encapsulamento IP-em-IP. O método de tunelamento padrão do Mobile IP é o encapsulamento IP-em-IP descrito na RFC 2003 [21].

A criação do túnel IP-em-IP é feita adicionando-se um novo cabeçalho IP no cabeçalho do pacote a ser transportado para permitir que este chegue um destino que não se alcançaria se fosse usado apenas o endereço IP original.

O tunelamento de um pacote a ser enviado se dá em cinco etapas básicas as quais estão enumeradas abaixo:

1. Primeiramente, o elemento gerador do pacote (um MN, por exemplo) o envia até um destino inalcançável com o próprio endereço IP do pacote;
2. Em seguida, o elemento do início do túnel (um FA, por exemplo) encapsula o pacote acrescentando um novo endereço IP no seu cabeçalho para mandá-lo para a outra extremidade do túnel a partir da qual se pode chegar ao destino com o cabeçalho original.
3. Feito isso, o pacote já encapsulado é roteado até o término do Túnel.
4. Nesta extremidade do Túnel o pacote é desencapsulado, retirando-se o novo cabeçalho que havia sido colocado no início do Túnel.
5. A partir daí, portando apenas o cabeçalho original, o pacote é entregue ao destinatário.

A figura abaixo apresenta uma ilustração das etapas descritas com suas respectivas numerações.

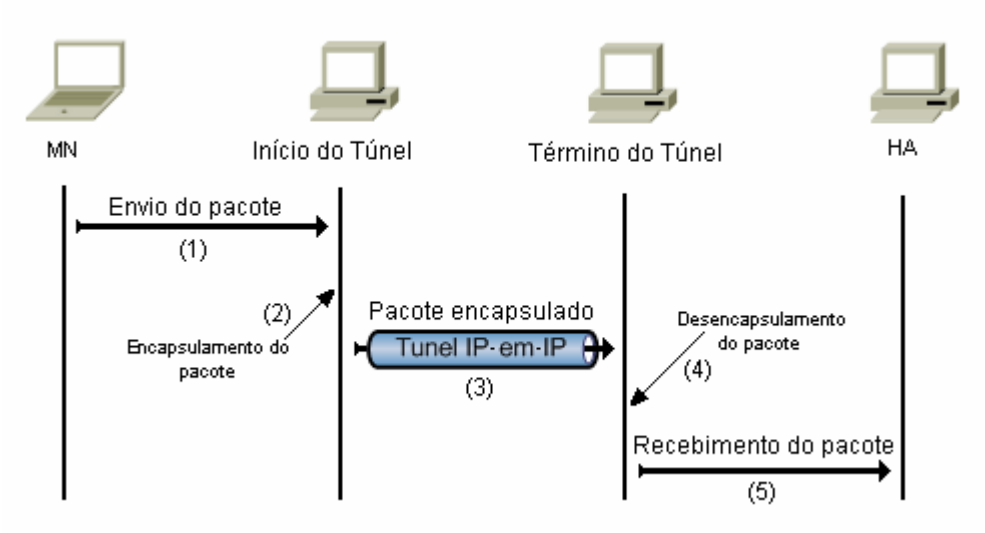


Figura 2.5 – Processo de tunelamento de pacotes.

Na estrutura mostrada acima, exemplificou-se como seria o envio de um pacote partindo do MN até sua chegada no HA por meio do túnel IP-em-IP. Um outro exemplo a ser dado é o envio do pacote partindo do HA para ser entregue ao MN. O mecanismo é o mesmo, o HA envia o pacote, este é encapsulado no agente nativo e enviado até o agente estrangeiro. Em seguida, o pacote é desencapsulado e enviado para o MN.

Até agora, falamos dos mais importantes mecanismos usados pelo *Mobile IP*, a descoberta dos agentes, o registro e o roteamento de pacotes. Dentro desses mecanismos, falamos sobre *handoff*, que é o registro do nó móvel em sua Rede nativa. Apesar de importante no quesito segurança e fundamental no envio e recebimento de pacotes, o *handoff* é também um ponto negativo no *Mobile IP*, pois é uma fonte de atraso durante o registro do nó móvel. Quando o este se movimenta de uma área de cobertura de rede para outra, mesmo que essas áreas se sobreponham, quando ele se desconectar da primeira rede, levará um certo tempo até que se conecte na segunda. Este tempo é o período gasto para que o nó móvel faça novamente sua autenticação no seu agente nativo, executando todas as requisições e recebendo todas as respostas novamente para que ocorra o registro. Portanto, quanto mais longe fisicamente o nó móvel estiver do seu agente nativo, mais tempo levará para que seja restabelecida a autenticação com sua Rede nativa.

Como proposta para a minimização desse problema, como citado no capítulo de introdução, apresenta-se a utilização do registro regional, ou IP Móvel Hierárquico. Este

mecanismo maximiza a eficiência do *handoff*, utilizando o conceito de micromobilidade, que consiste em criar domínios na rede estrangeira afim de que, se o nó móvel se movimentar de um domínio para outro, a autenticação só ocorre regionalmente, tornando-se desnecessário repetir a autenticação na Rede nativa. Este mecanismo e sua RFC serão explicados detalhadamente a seguir.

2.2 MOBILE IP HIERÁRQUICO - REGISTRO REGIONAL DO MOBILE IPV4 [RFC 4857]

Um dos grandes desafios no projeto de redes sem fio é criar uma forma eficaz de administrar a mobilidade. O protocolo Mobile IP é o padrão aceito para tratar da macromobilidade dos dispositivos móveis em sistemas sem fio baseados em pacotes. Ou seja, o MIP assegura que os usuários podem restabelecer a comunicação quando este se move em diferentes domínios administrativos em regiões geográficas diferentes. Através do MIPv4, um nó móvel é capaz de se registrar com seu agente nativo cada vez que muda seu endereço residente(CoA) como especificado na RFC 3344 [22]. De modo a manter serviços sem interrupção enquanto o usuário se move, o protocolo MIP requer que o nó móvel informe sua localização ao seu agente nativo a cada mudança de sub-rede de tal forma que o agente nativo intercepte os pacotes destinados ao nó móvel e envie estes pacotes através de um túnel em direção ao atual ponto de conexão dele. No entanto, o protocolo MIP não é muito eficiente para nós móveis com grande mobilidade. Seu mecanismo requer que todo nó móvel atualize seu novo endereço residente cada vez que ele muda de uma sub-rede para outra. Portanto, se um nó móvel está muito distante de sua rede nativa o tempo de sinalização para esses registros pode ser muito longo o que pode causar perda de pacotes *in-flight* e a degradação de QoS.

O Registro Regional para o MIP propõe uma solução para esse problema aplicando uma hierarquia entre os agentes estrangeiros de modo a localizar os registros dentro de um mesmo domínio. Se o nó móvel se mover de uma sub-rede para outra dentro do mesmo domínio este não deverá registrar-se novamente com seu agente nativo, ao invés, o nó móvel registra-se com uma nova entidade denominada Gateway Foreign Agent (GFA). Os registros regionais reduzem o número de mensagens de sinalização em direção ao agente nativo e diminuem o atraso de sinalização quando um nó móvel se move dentro de um

mesmo domínio. Assim, o MIP hierárquico propõe uma solução para administração da micromobilidade do nó móvel.

2.2.1 Descrição do Protocolo

A topologia onde o protocolo opera é ilustrada na figura abaixo. É assumida uma hierarquia simples, mas outros níveis de hierarquia podem ser implementados. Nesta figura podemos ver as entidades envolvidas, a rede nativa e o domínio sendo visitado.

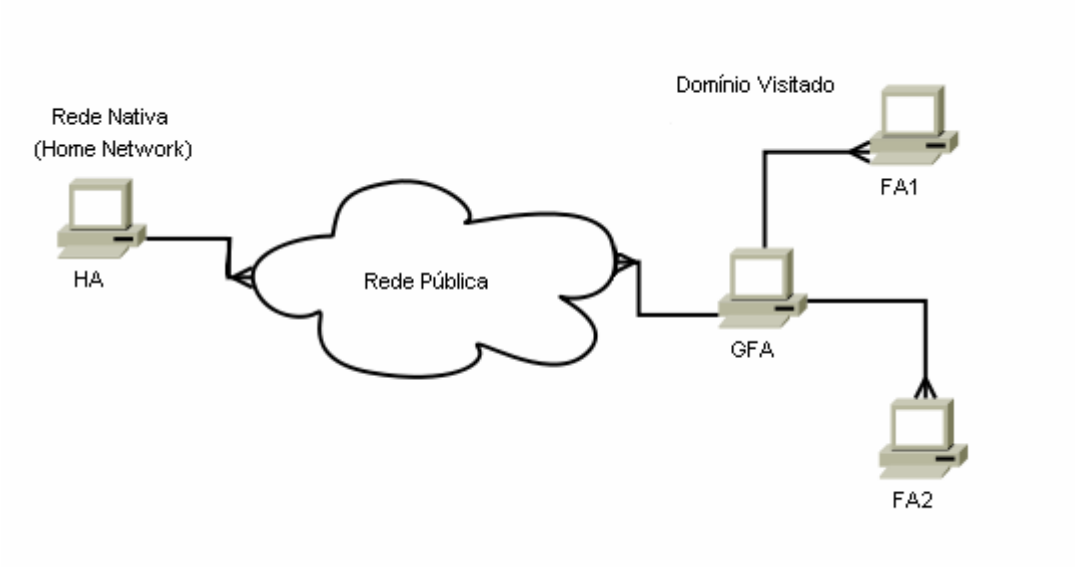


Figura 2.6 – Topologia básica do protocolo

Vemos na figura 1 que existem dois níveis de hierarquia de FAs no mesmo domínio. No topo da hierarquia fica o GFA(Gateway Foreign Agent) cuja diferença de um FA é seu endereço IP, que é publicamente roteável.

No MIP comum, se o MN se move do FA1 para o FA2, ele deveria realizar um novo registro com seu HA, o que pode ser demorado se a distância entre o domínio visitado e a rede nativa for grande. Com registro regional, o MN envia uma requisição de registro ao HA com o endereço IP do GFA como sendo seu CoA. O GFA mantém uma lista de visitantes que indica qual endereço dos FAs que será tratado como CoA do MN dentro do domínio. Forma-se, portanto, dois túneis IP-in-IP: um entre o HA e o GFA, e outro entre o GFA e o FA onde o MN está conectado. Assim, quando o MN se move entre sub-redes

dentro do mesmo domínio, ele precisa apenas interagir com o FA e GFA de modo a criar novos túneis apenas entre o GFA e o FA onde ele está conectado.

Nesse protocolo dois tipos de mensagens de registro são utilizados. A primeira, denominada Mensagens de Solicitação/Resposta de Registros (Registration Requests/Replies), ocorre quando o MN registra com o HA como descrito na RFC 3344 [22]. No entanto, essas mensagens agora são passadas pelo GFA e isso implica a implantação de novas extensões a estas mensagens. A outra é o novo par de Mensagens de Solicitação/Resposta de Registro denominada Mensagens de Solicitação/Registro de Registro Regional (Regional Registration Requests/Replies), nas quais são usadas dentro do domínio visitado para sinalização. O MN usa estas mensagens para comunicar ao GFA que está se movendo de um FA para outro dentro do domínio desse GFA.

Será descrito como esses registros ocorrem quando o MN chega em um novo domínio e quando ele se move dentro deste.

2.2.2 Registro com a Rede nativa

O registro com a rede nativa pode ocorrer na primeira vez que o nó móvel chega no domínio sendo visitado, ou quando o MN requer um novo HA ou quando ele muda para um domínio que tenha um novo GFA. Esse registro também pode ocorrer quando o tempo de vida de um registro anterior expira.

O fluxo de mensagens de sinalização para o registro com a Rede nativa é ilustrado na figura a seguir.

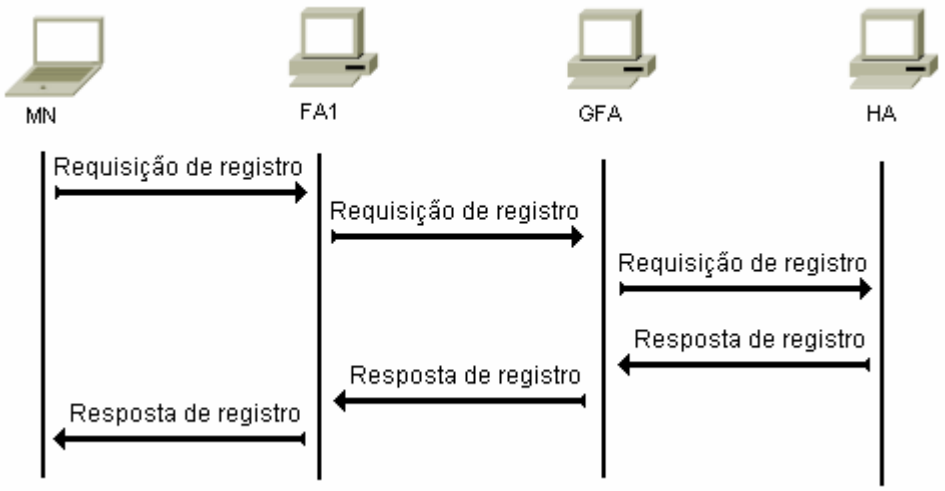
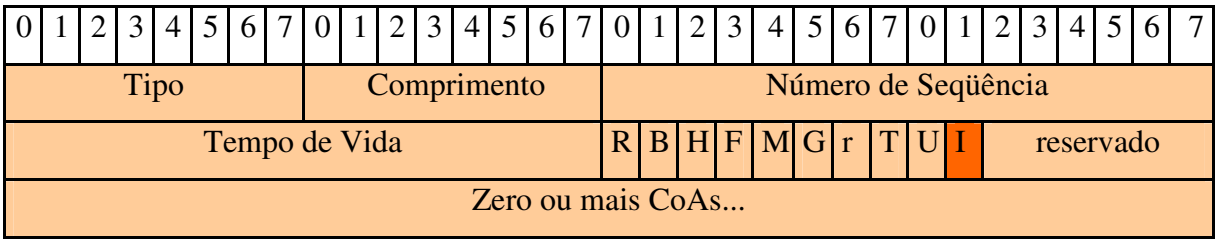


Figura 2.7 – Registro com a Rede nativa

Como ocorre no MIP comum, a descoberta de agentes pode se dar através de mensagens de solicitação de agentes ou através de mensagens de anúncio de agentes (Agent Request Message e Agent Advertisement Message respectivamente). Se o domínio do FA suporta Registro Regional, as mensagens de anúncio de agentes possuem uma alteração na extensão Mobility Agent Advertisement Extension definida na RFC 3344 [22]. Nesta extensão, é adicionado o bit "I" após todos as outros bits já definidos para essa extensão. Esse bit é definido para indicar que o domínio suporta Registro Regional.



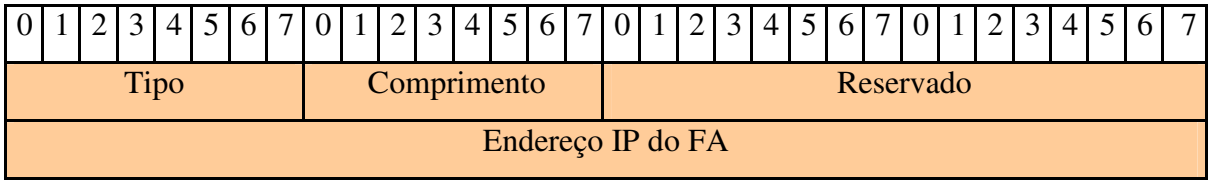
Tipo: 16 (Mobility Agent Advertisement)

I: Este domínio suporta registro regional.

Figura 2.8 – Formato da Extenção *Mobility Agent Advertisement Extension* com flag I.

Nas mensagens de anúncio de agentes enviadas pelo FA com o bit "I" com valor 1, pode haver o anúncio do endereço IP de um GFA ou alternativamente pode haver designação dinâmica de GFA que será discutido logo adiante.

No caso onde o endereço IP do GFA é anunciado pelo FA, a mensagem de solicitação de registro enviada ao FA pelo nó móvel possui o campo relativo ao CoA preenchido com o endereço do GFA anunciado. O FA adiciona a essa mensagem de solicitação de registro uma extensão denominada Hierarquical Foreign Agent Extension após todas extensões existentes no momento do recebimento dessa mensagem. Nessa extensão está o endereço IP do FA onde o MN está conectado. Quando essa mensagem é repassada ao GFA, essa extensão é retirada pelo GFA e este mantém, numa lista de conexões pendentes, o endereço do FA onde o MN está presente, ou seja, o CoA do MN dentro do domínio visitado.



Tipo: 140 (Hierarchical Foreign Agent Extension)

Figura 2.9 – Formato da extensão Hierarquical Foreign Agent onde o endereço do FA onde o MN está conectado é enviado ao GFA.

A mensagem de solicitação de registro, agora sem a extensão HFA, é repassada ao HA pelo GFA com seu endereço IP preenchendo o campo designado ao CoA. Como o CoA registrado no HA é o endereço do GFA, este não mudará quando o nó móvel se mover de um FA a outro sobre o mesmo GFA mantendo esse túnel IP-in-IP fixo enquanto o tempo de vida do registro ainda não expirou.

A outra forma de designação de GFA ao MN é a designação dinâmica. Para indicar que a designação dinâmica está sendo usada, o FA envia nas mensagens de anúncio de agentes o campo referente ao endereço do GFA preenchido com 1s. Se o MN suporta esse tipo de designação ele envia a mensagem de solicitação de registro com o campo referente ao CoA preenchido com 0s. Nesse sistema, não há como o MN saber com antecedência qual será o GFA pelo qual ele fará sua comunicação com seu agente nativo. Assim que o FA recebe a mensagem de solicitação de agente com o campo CoA preenchido com 0s, ele a repassa ao GFA apropriado para esse MN. Nesse momento surge uma diferença do modo de registro discutido anteriormente. No caso anterior, o MN é avisado da disponibilidade de um GFA nas mensagens de anúncio de agentes enviadas pelo FA e então envia uma mensagem de solicitação de registro com o campo CoA preenchido com o endereço IP do GFA. No caso da designação dinâmica o MN são saberia qual GFA seria designado, portanto, para que o endereço do GFA seja enviado ao HA, o GFA adiciona uma extensão à mensagem de solicitação de registro que será encaminhada ao HA. Essa extensão é denominada GFA IP Address Extension.

0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
Tipo								Comprimento								Reservado															
Endereço IP do GFA																															

Tipo: 46 (GFA IP Adress)

Figura 2.10 – GFA IP Address Extension

Ao receber a mensagem de solicitação de registro enviada pelo GFA, o HA percebe que o campo do CoA está preenchido com 0s, o que indica a presença de designação dinâmica de GFA. Portanto, o agente nativo deve registrar o endereço IP do GFA que está contido na extensão GFA IP *Address Extension* como sendo o CoA do MN. O HA deve inserir essa extensão no final da mensagem de resposta de registro encaminhada ao GFA. Isso ocorre

porque o MN deve saber o endereço do GFA e isso ocorre ao verificar a extensão GFA IP *Address* quando ele recebe a mensagem de resposta de registro.

Neste momento percebe-se que, para que a designação dinâmica de GFA funcione apropriadamente, o MN, o GFA e o HA devem suportar a extensão GFA IP *Adress Extension*. O FA, por sua vez, deve ser capaz de anunciar um endereço residente preenchido com 1s e de suportar uma solicitação de registro com endereço residente preenchido com 0s.

2.2.3 Registros Regionais

Nesta seção será descrito como o registro regional ocorre. Nessa descrição é assumido que um registro com a rede nativa já ocorreu como descrito na seção anterior. Uma vez que o agente nativo registrou o endereço do GFA como sendo o endereço residente do nó móvel, o nó móvel pode realizar registros regionais. Na ocasião do nó móvel realizar um registro regional, ele deve registrar o endereço do FA como sendo seu endereço residente dentro do domínio do GFA.

Referente a figura 1.5, supõe-se que o nó móvel realizou um registro com a rede nativa enquanto estava na no FA1. Num certo instante o nó móvel começa a se mover do FA1 em direção ao FA2 .

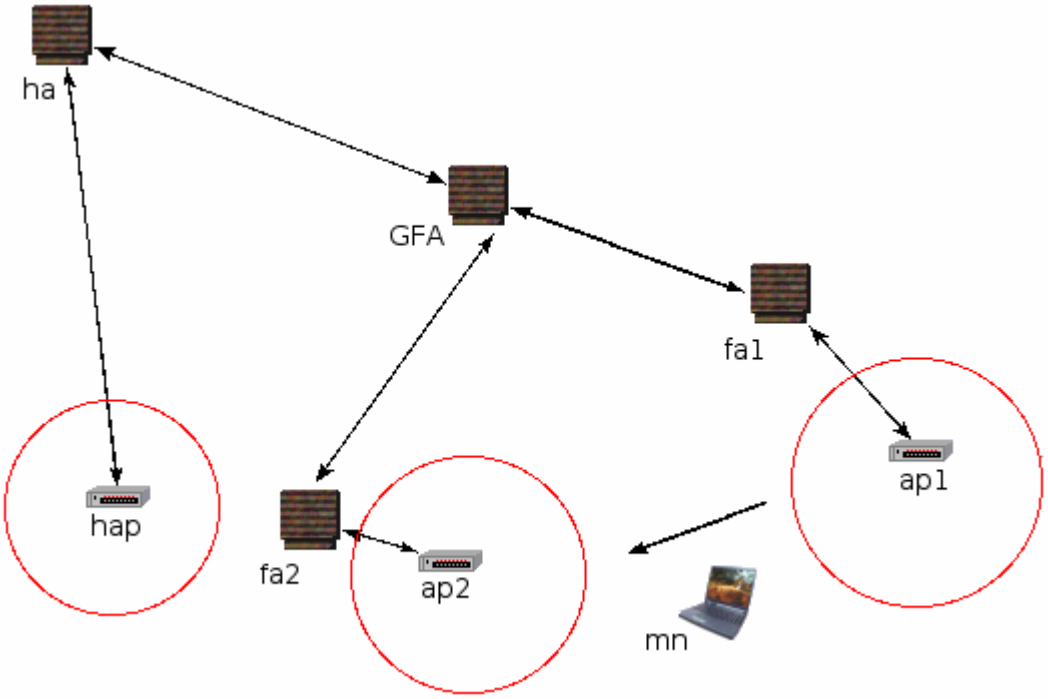


Figura 2.11 – Nó móvel movendo-se de um FA para outro.

Quando chega na área de cobertura do ponto de acesso “ap2” o nó móvel começa a receber mensagens de anúncio do FA2. Supõe-se que nessas mensagens o bit “I” está com valor 1, ou seja, este domínio suporta registro regional. Se o endereço do GFA que está sendo anunciado é o mesmo endereço do GFA que o nó móvel registrou como sendo seu endereço residente durante seu último registro com a rede nativa então, o nó móvel pode realizar registro regional com este novo FA e o GFA. O nó móvel solicita novo registro com o GFA através desse novo FA. A requisição é autenticada por uma associação de segurança existente entre o GFA e o nó móvel. Essa associação é necessária para poder evitar que algum *host* malicioso se registre como o nó móvel e desvie o tráfego simplesmente fazendo que com que o nó móvel pareça estar em outro lugar.

Finalmente o GFA registra o novo endereço do FA como sendo o novo CoA do nó móvel, e encaminha os pacotes direcionados a ele por esse destino.

Se por outro lado, o nó móvel enviar mensagens de solicitação de registro regional com o campo designado ao CoA preenchido com zeros o novo FA adiciona a extensão HFA (*Hierarchical Foreign Agent Extension*) à mensagem e a direciona ao GFA. O GFA usa a informação contida nessa extensão(ver figura 1.4) para renovar o endereço do ponto de ligação do nó móvel em sua lista de visitantes. O GFA envia uma mensagem de resposta de registro ao nó móvel via FA.

Pode acontecer também o fato de o endereço do GFA anunciado nas mensagens de anuncio enviadas pelo FA ser diferente do endereço que o nó móvel registrou como seu CoA em sua rede nativa na ocasião do seu último registro com ela. Se esse GFA anunciado pertencer ao mesmo domínio do GFA anterior o nó móvel pode tentar se registrar com seu GFA antigo, caso o FA não suporte registro com GFA diferente do anunciado este deve negar o registro com uma mensagem de erro *UNKNOWN_GFA*. Neste caso o nó móvel deve fazer um novo registro com sua rede nativa.

O nó móvel também pode fazer um registro regional quando o tempo de vida do Registro Regional está prestes a expirar.

É importante ressaltar que novos tipos de mensagens devem ser introduzidas nas Solicitações e Respostas de Registros Regionais. Isso se deve ao fato de que o nó móvel deve ser capaz de diferenciar Registros Regionais dos Registros com a rede nativa, pois no primeiro caso o registro é feito com o GFA e no segundo é feito com a rede nativa.

Essas mensagens são protegidas por extensões de autenticação da mesma maneira que as mensagens de registro são protegidas no Mobile IPv4.

Como descrito anteriormente, as mensagens de Solicitação de Registro Regional são utilizadas pelo nó móvel para registrar-se com seu GFA corrente. Esta possui a estrutura descrita na figura 1.7.

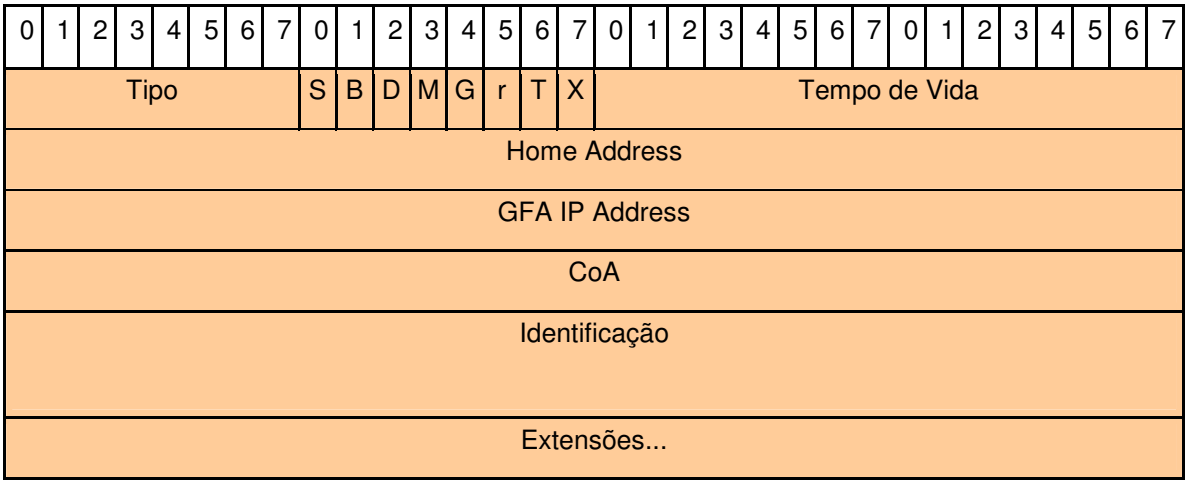


Figura 2.12 – Formato da mensagem de Solicitação de Registro Regional.

A mensagem de Solicitação de Registro Regional é definida como as mensagens de Solicitação de Registro definidas na RFC 3344 [22], no entanto ela apresenta algumas mudanças que são descritas a seguir:

Tipo: 18 (Solicitação de Registro Regional)

Tempo de vida: Número de segundos restantes antes do Registro Regional ser considerado expirado. O valor 0 quer dizer que o registro deve ser desfeito com o GFA. O valor 0xFFFF indica tempo de vida infinito.

Endereço IP do GFA: Endereço IP do *Gateway Foreign Agent*(GFA). Substitui o campo *Home Agent* na mensagem de Solicitação de registro definida na RFC 3344 [22].

Identificação: Número de 64 bits criado pelo nó móvel para relacionar uma mensagem de solicitação de registro com a respectiva mensagem de resposta.

Extensões: Extensões já existentes nas mensagens de Solicitação de Registro com a adição da extensão *Hierarchical Foreign Agent* (HFA).

As mensagens de Resposta de Registro Regional são o retorno da solicitação e dizem se o registro regional foi aceito ou foi negado ao nó móvel.
As mensagens de Resposta de Registro Regional possui os campos referentes ao Mobile IP como descritos a abaixo:

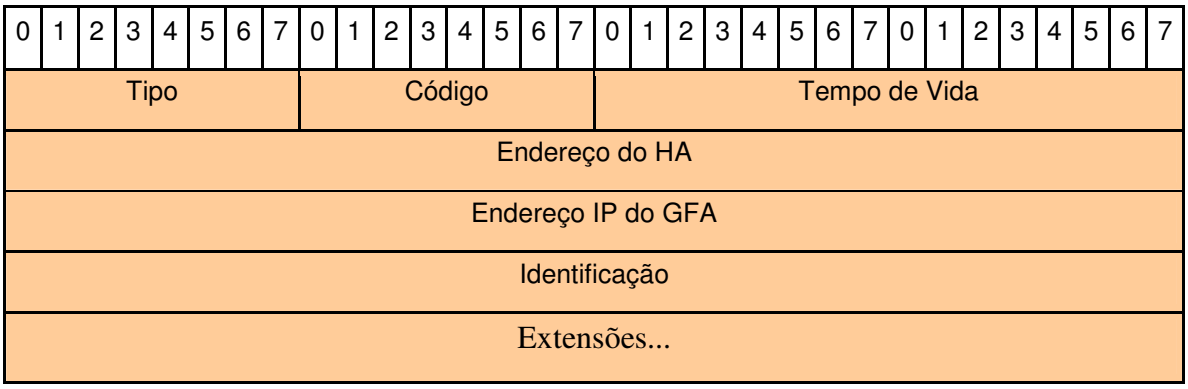


Figura 2.13 - Formato da mensagem de Resposta de Registro Regional.

As mensagens de Resposta de Registro Regional são definidas como as mensagens de Resposta de Registro na RFC 3344 [22] com as seguintes diferenças:

Tipo: 19 (Resposta de Registro Regional)

Código: Um valor indicando o resultado da Solicitação de Registro Regional. Esse campo pode ter os valores descritos na RFC 3344 [22] mais alguns valores adicionais como descreve a tabela abaixo.

Tabela 2.1 – Erros do campo Código Referentes aos Registros negados pelo FA

Erro	Valor
UNKNOWN_GFA	112
GFA_UNREACHABLE	113
GFA_HOST_UNREACHABLE	114
GFA_PORT_UNREACHABLE	115

Tabela 2.2 – Erros do campo Código Referentes aos Registros negados pelo GFA

Erro	Valor
NO_HOME_REG	193

Tempo de vida: Se o campo de código indicar que o registro regional foi aceitado, o campo Tempo de Vida é preenchido com o valor em segundos do tempo restante para que o registro regional seja considerado expirado. Um valor nulo indica que o registro com o GFA foi desfeito. O valor 0xFFFF indica infinito. Se o campo de código indicar que o registro foi negado então o conteúdo deste campo deve ser ignorado.

2.3 – CONCLUSÃO

Neste capítulo foram analisados os aspectos mais relevantes a respeito do Móbile IP. Foram também mostrados os principais problemas referentes ao protocolo, a latência no envio de pacotes pelo fato deles terem de passar pelo HÁ e o atraso de *handoff* ocasionado pelo registro no HÁ toda vez que o nó móvel troca de sub-rede.

Uma importante conclusão a que se chega neste capítulo é sobre o aprimoramento Mobile IP Hierárquico, que traz para o Mobile IP um gerenciamento para micromobilidade, ou seja, ele implementa uma forma de minimizar a latência de handoff quando o nó móvel está se movendo dentro de um mesmo domínio. Sendo assim, o Registro Regional contribui para o grande propósito do MIP, evitar que a conexão seja perdida por muito tempo no evento da movimentação.

Para o problema da latência no envio de pacotes, a solução proposta no trabalho é o uso do MPLS como protocolo de roteamento, o qual será tratado mais detalhadamente no próximo capítulo.

3 - MPLS

Neste capítulo serão explicados todos os aspectos relevantes a respeito do Protocolo MPLS, com o objetivo de se obter uma base completa de informações para que se possa fazer sua implementação com todos os seus parâmetros previamente explanados.

3.1 - FUNCIONAMENTO BÁSICO DO MPLS

Hoje em dia, com a grande necessidade de rapidez no transporte e roteamento de pacotes na Internet, cada vez mais necessita-se de mecanismos que maximizem a eficiência desses serviços. O Multiprotocol Label Switching (MPLS) é apresentado como uma solução que possibilita melhorar a velocidade de encaminhamento dos pacotes, sendo uma tecnologia de grande importância em redes IP. A Engenharia de Tráfego, que representa a habilidade de operadores de rede de ditar o trajeto pelo qual o tráfego segue, e o suporte a VPN são exemplos de duas aplicações onde o MPLS possui grande destaque.

O MPLS é baseado na geração de etiquetas para o transporte de pacotes na rede. Elas irão atuar como uma forma abreviada do cabeçalho IP. Este mesmo conceito é utilizado nos correios, onde a rua e a cidade são representadas por um endereço postal, de tamanho fixo e reduzido, que é o CEP. Através do uso dessas etiquetas criadas no MPLS são tomadas as decisões no encaminhamento do pacote. Os pacotes IP possuem um campo em seu cabeçalho que contém o endereço indicando para onde o pacote será encaminhado. No roteamento tradicional, este campo é verificado e processado em cada roteador da rede até que ele atinja seu destino.

No MPLS, os pacotes IP são encapsulados, através do uso de etiquetas, pelos dispositivos que se encontram na entrada da rede. O roteador de borda do MPLS analisa os índices do cabeçalho IP e seleciona uma etiqueta apropriada para colocar no pacote que será encapsulado. Grande parte da eficiência do MPLS vem do fato que, em contraste ao roteamento tradicional IP, esta análise pode ser baseada não apenas no endereço de destino que ele carrega dentro do cabeçalho, mas também pelo Qos requerido. Em todos os nós subseqüentes dentro da rede MPLS, a etiqueta é utilizada pelos roteadores para realizar a decisão de encaminhamento dos pacotes na rede. Em nenhum momento os roteadores pertencentes ao núcleo da rede analisam o cabeçalho IP. No final do processo, na medida em que os pacotes deixam a rede MPLS, as etiquetas são retiradas pelos roteadores de borda para que eles sejam entregues aos seus destinatários.

Os roteadores IP convencionais contêm tabelas de roteamento onde são feitas buscas referentes à informação do cabeçalho IP de um pacote para que a decisão de encaminhamento seja tomada. Estas tabelas são construídas pelos protocolos de distribuição do IP (por exemplo, RIP ou OSPF). O roteamento em geral engloba o plano de encaminhamento e o plano de controle. No roteamento IP, a análise do cabeçalho é feita no plano de encaminhamento, e, no plano de controle, é gerada a tabela de roteamento. Já no MPLS é possível separar o plano de encaminhamento do plano de controle. Com isso é possível modificar cada um separadamente. Devido a esta característica, não precisamos, por exemplo, mudar os dispositivos de encaminhamento caso se deseja mudar a estratégia de roteamento da rede.

O objetivo do MPLS não é abandonar a infra-estrutura de repasse de datagramas IP com base no destino em favor de rótulos de tamanho fixo e circuitos virtuais, mas aumentá-la rotulando datagramas seletivamente e permitindo que roteadores repassem datagramas com base em rótulos de tamanho fixo, (ao invés de endereços de destino IP) quando possível. Um ponto importante é que essas técnicas trabalham usando roteamento e endereçamento IP, tornando-se aplicável a uma grande quantidade de redes. A IETF reuniu esses esforços no protocolo MPLS RFC 3031 [13], mesclando efetivamente técnicas de circuitos virtuais em redes de datagramas com roteadores.

Os padrões do MPLS prometem oferecer a importante funcionalidade de inter-operabilidade entre diferentes tecnologias de redes. Uma dessas tecnologias é o *Mobile IP*, o qual pretende-se integrar com o MPLS para que seja aumentada sua eficiência.

Para se entender como o MPLS funciona, primeiramente é importante observar diversos conceitos básicos que se aplicam a tecnologia de comutação de pacotes. Abaixo são listadas algumas delas:

Componente de controle: Constrói e mantêm a tabela de encaminhamento do nó em uso. O MPLS trabalha com componentes de controle de outros nós para distribuir informação de roteamento de forma consistente e precisa, e também assegura procedimentos locais que são usados nas tabelas de encaminhamento. Protocolos de roteamento padrão (por exemplo, OSPF, BGP, e RIP) são usados na troca de informações de roteamento entre os componentes de controle. O componente de controle precisa reagir quando ocorrem mudanças na rede (tal como a falha de um link) mas não é envolvido no processamento individual dos pacotes.

Componentes de encaminhamento: Executam o encaminhamento dos pacotes. Eles usam as informações a partir da tabela de encaminhamento, que é mantida pelo roteador. Esta informação é carregada no próprio pacote e um conjunto de procedimentos locais faz a decisão de encaminhamento. No roteamento convencional, um algoritmo compara o endereço de destino no pacote com uma entrada na tabela de roteamento até que se encontre um valor vantajoso. Todo este processo é repetido em cada nó da fonte da origem até ao destino. Nos LSR (será explicado mais a frente), algoritmos de troca de etiquetas usam as etiquetas dos pacotes e a tabela de encaminhamento baseado em etiquetas para obter uma nova e a interface de saída para o pacote.

Tabela de encaminhamento: Conjunto de entradas em uma tabela que fornece informações que ajudam as componentes de encaminhamento executar as funções de comutação. A tabela de encaminhamento precisa associar cada pacote a uma entrada (tradicionalmente o endereço de destino) para fornecer instruções sobre para onde o pacote deverá ir.

LIB (*Label Information Base*): Tabela que indica aonde e como encaminhar os pacotes. Criada por equipamentos pertencentes a um domínio MPLS, a LIB contém uma lista de entradas que consistem de uma sub-entrada de ingresso e uma ou mais sub-entradas de egresso (etiqueta de saída, interface de saída, componentes de saída de nível de enlace). A LIB é construída baseada nas informações obtidas pelo LSR através da interação com os protocolos de roteamento.

FEC (*Forwarding Equivalence Class*): Definido como um grupo de pacotes que podem ser tratados de maneira equivalente. Um exemplo de uma FEC é um conjunto de pacotes *unicast* cujos endereços de destinos é compatível a um prefixo particular de endereço IP. Outro FEC é o conjunto de pacotes cujos endereços de fonte e destino são os mesmos.

Etiqueta (*label*): Identificador relativamente curto de comprimento fixo que é usado no processo de encaminhamento dos pacotes. As etiquetas são associadas a uma FEC em um processo obrigatório do MPLS. As etiquetas são normalmente locais em um link de dados e não têm nenhum significado global como um endereço. As etiquetas são restringidas pelas FEC, sendo resultado de alguns eventos explicados a seguir.

Na terminologia do MPLS, existem duas categorias de roteadores. Na borda da rede, os classificadores de pacote precisam de um elevado desempenho no processo de aplicação e retirada de etiquetas: eles são conhecidos como roteadores de borda, ou LER (*Label Edge Router*). A outra classe é composta pelos roteadores de núcleo, os quais são chamados de

LSR (*Label Switching Router*). Uma de suas características principais é encaminhar os pacotes já etiquetados de forma rápida, por isso a necessidade de um grau de processamento elevado.

No mecanismo de envio de pacotes MPLS, a cada pacote que entra na rede, primeiramente, é atribuído uma classe de envio FEC. Portanto, no cabeçalho, é inserido um rótulo identificador da FEC. Este rótulo colocado na borda da rede MPLS servirá de guia do pacote dentro da rede, sendo o único elemento a ser analisado no encaminhamento do pacote. Sendo assim, os próximos roteadores que receberem o pacote, irão repassá-lo apenas analisando esse rótulo. Esse processo de análise é chamado de permuta de rótulos (*label swapping*). O tamanho do rótulo analisado é de apenas 4 *bytes*, muito menor que um cabeçalho analisado em uma rede convencional, que é de 20 *bytes*.

No MPLS, diferentemente do roteamento IP tradicional, a atribuição de um pacote a uma determinada FEC é feita apenas uma vez, quando o pacote entra na rede MPLS. Uma FEC pode ser determinada por um ou mais parâmetros, dentre os quais podemos citar: endereço IP de origem ou destino, porta de origem ou destino, identificação do protocolo IP ou classe de serviço.

Um mecanismo muito importante na passagem do pacote de um roteador para outro é a troca de rótulos. Quando o pacote chega a um roteador, este verifica qual será o próximo nó que receberá o pacote e qual o próximo rótulo que será colocado no pacote. Sendo assim, o rótulo utilizado por uma determinada FEC muda de nó para nó. Esse mecanismo facilita a distribuição de rótulos em diferentes roteadores de borda da rede e permite a utilização de rótulos iguais em partes diferentes do túnel MPLS.

3.2 - ROTULAÇÃO

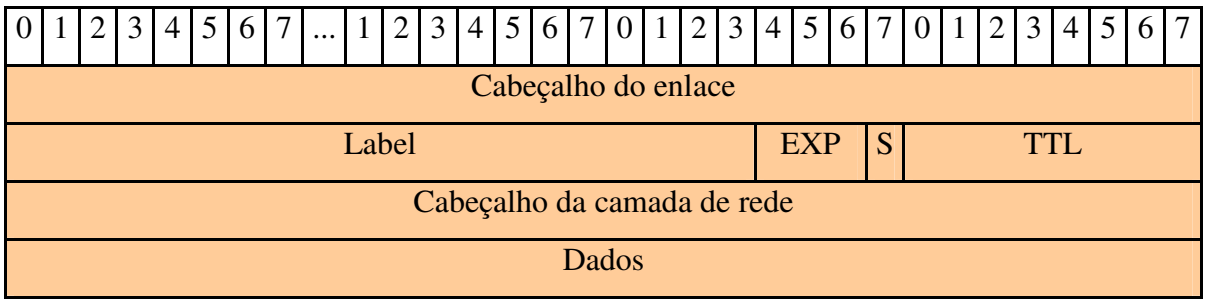


Figura 3.1 - Rótulo de um pacote MPLS

A figura 3.1 mostra o formato básico de um pacote rotulado em uma rede MPLS. Ele apresenta os seguintes campos importantes:

- O campo *Label* - 20 bits, carrega o valor atual da etiqueta MPLS;
- O campo EXP - 3 bits, Experimental bits. Utilizados para classe de serviço, eles podem afetar os algorítimos de enfileiramento e descarte aplicados ao pacote enquanto o mesmo é transmitido através da rede;
- O campo *S (Stack)* - 1 bit, suporta uma pilha hierárquica de etiquetas;
- O campo *TTL* - 8 bits, é o tempo de vida, que fornece funcionalidades de TTL IP convencional.

3.3 - CARACTERÍSTICAS DOS LSR E LER

Um LSR (*Label Switching Router*) é um dispositivo que suporta ao mesmo tempo os componentes de controle IP (isto é, protocolos de roteamento, RSVP, etc..) e os componentes de encaminhamento e troca de etiquetas. A figura a seguir exemplifica de forma simples uma rede de comutação de etiquetas e ilustra os roteadores de borda LERs (fornecem as funções de entrada e saída da rede) e de Núcleo LSRs (executam comutação em alta velocidade). Uma rede de comutação de etiquetas possui o mesmo objetivo das redes de roteamento tradicional: entregar o tráfego em um ou mais destinos.

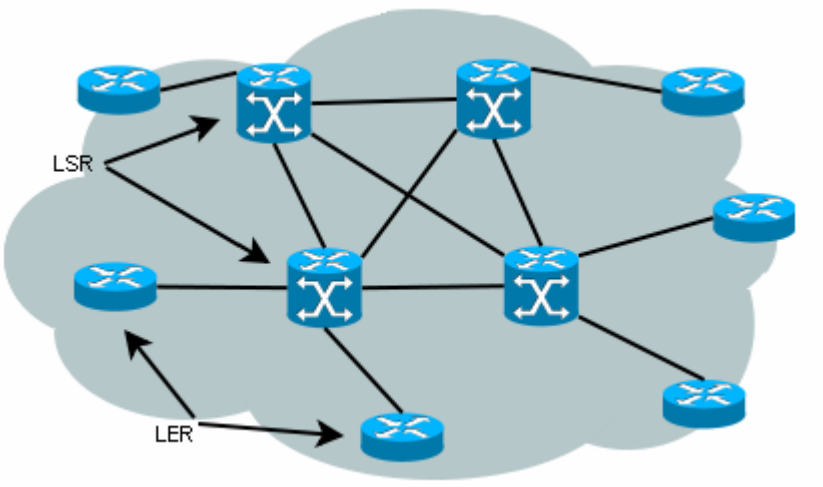


Figura 3.2: Exemplo de rede MPLS mostrando seus roteadores LER e LSR.

Quando um pacote entra em uma rede MPLS, como já foi dito, é criada uma FEC e esta é usada no roteamento desse pacote dentro do núcleo MPLS pelos roteadores LSR. Dentro desse contexto, o LSR emissor é considerado estar *upstream* na transmissão e o LSR

receptor é considerado estar *downstream.* Segundo a especificação do MPLS, o roteador que determina o rótulo é sempre o *downstream*. Os protocolos de encaminhamento de rótulos são responsáveis pela troca de pacotes no núcleo da rede. Ao se associar um rótulo a uma FEC, esta passa a ter características também determinadas pelo protocolo de encaminhamento.

É responsabilidade dos Roteadores de Borda, LER classificar o tráfego e aplicar ou remover etiquetas dos pacotes. As etiquetas podem ser atribuídas tomando como base requisitos de Qos, e não o seu endereço de destino como é feito no roteamento tradicional. O LER determina se o tráfego é um fluxo estável e implementa políticas de gerência e controle de acesso. Assim, a capacidade dos LER é a chave para o sucesso de um ambiente de comutação de etiquetas.

3.4 - O PATH LSP E AS TABELAS LIB

Um conjunto de dispositivos MPLS representa um domínio MPLS. Dentro desse domínio, um caminho é criado para um determinado pacote baseado em sua FEC. Este caminho é formado através de uma seqüência ordenada de LSRs estabelecida entre uma origem e um destino dentro do mesmo domínio. Este caminho formado é conhecido como LSP (*Label Switched Path*).

O LSP é ajustado antes da transmissão de dados. Ele é feito com protocolos de roteamento convencionais ou roteamento com restrições. É importante realçar que um LSP é unidirecional, portanto é preciso ter dois LSPs para uma comunicação entre duas entidades. Abaixo é ilustrado um LSP feito entre uma borda da rede e outra.

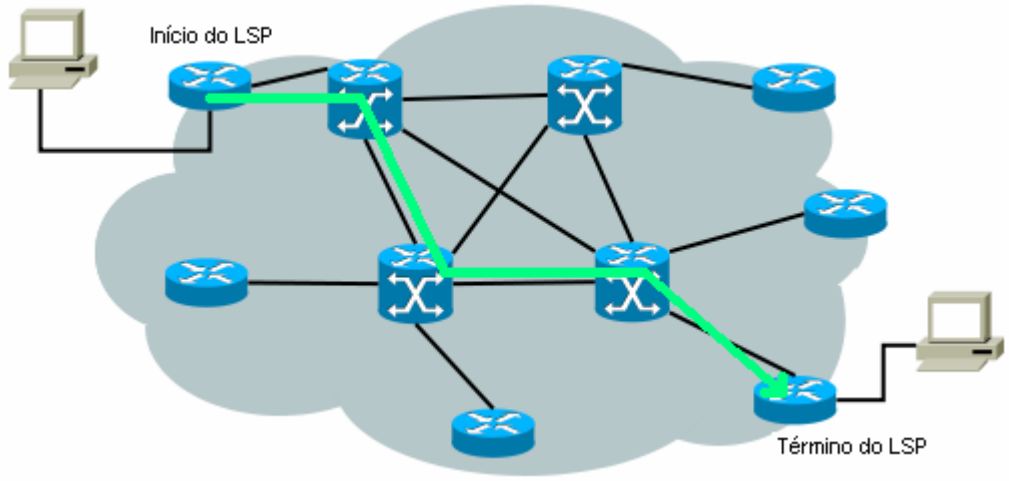


Figura 3.3: Exemplo de um LSP

Novos LSPs são criados apenas com novas FECs, evento que ocorre a partir da borda da rede, ou seja, nos LERs, os roteadores de borda. Sendo assim, os demais roteadores, os LSRs, apenas chavearão os rótulos, encaminhando os pacotes conforme a LSP já determinada, não havendo a necessidade de fazer um novo roteamento dos pacotes. Os rótulos são distribuídos assim que são estabelecidos os LSPs.

No envio de pacotes, após o início da criação de uma FEC, em cada LSR são criadas tabelas para especificar a forma como um pacote deve ser encaminhado. Essas tabelas, como já foi dito anteriormente, são as chamadas LIB (*Label Information Base*). Para ela ser criada, associa-se informações contidas nas FECs e nos rótulos já criados, dependendo da região da rede. A tabela abaixo mostra um exemplo de como pode ser disposta uma LIB.

In label	***Address Prefix***	***Out Interface***	***Out Label***
4	128.89	0	9
5	171.69	1	7
	...	...	

Figura 3.4: Exemplo de uma LIB.

A entrada de uma tabela de encaminhamento fornece, no mínimo, informação sobre a interface de saída e a nova etiqueta, mas também contém outras informações. Ela pode, por exemplo, indicar o método de enfileiramento na saída a ser aplicado ao pacote. Esse método é chamado de Protocolo de Distribuição de Rótulos.

3.5 - PROTOCOLOS DE DISTRIBUIÇÃO DE RÓTULOS

Um protocolo de distribuição de rótulos é o conjunto de procedimentos que permite a um roteador LSR distribuir rótulos para os outros roteadores. Os protocolos de distribuição contemplam todas as negociações entre LSRs necessárias para instruir a funcionalidade MPLS de cada nó. Sendo assim, para o estabelecimento de túneis LSP são utilizados esses procedimentos, pelos quais um LSR informa outro sobre as características da FEC que acarretou a criação de cada rótulo distribuído. Pelo fado do LSP ser unidirecional e de serem necessários dois LSP para a comunicação entre duas entidades, as trocas de informações rótulo/FEC através de um protocolo de distribuição de rótulo são chamadas pares de distribuição de rótulos.

Vários protocolos de distribuição de rótulos são suportados pelo MPLS, sua arquitetura, definida pela RFC 3031 [13] permite não apenas um único protocolo. Dos protocolos suportados pelo MPLS estão, por exemplo, o RSVP RFC 2205 [25], RSVP-TE RFC 3209 [03] (extensão padronizada do RSVP) e o OSPF RFC 2328 [18]. Além destes, foram desenvolvidos protocolos especialmente para o MPLS, como é o caso do LDP RFC 3036 [15] (*Label Distribution Protocol*) e do CR-LDP RFC 3212 [14] (*Constraint-based Routing LDP*), que é uma extensão do LDP já com melhoramentos relacionados a engenharia de tráfego e outros fatores. A seguir, daremos uma explicação do funcionamento do protocolo LDP para que se complete a base necessária para se entender por completo o funcionamento básico do MPLS. A partir daí, teremos condições de dar continuidade às implementações e comparações do MPLS e do HMIP no próximo capítulo.

3.5.1 Label Distribution Protocol – LDP

O protocolo LDP, como foi dito anteriormente, foi criado especialmente para atuar no MPLS. Com o intuito de fazer eficientemente a distribuição de rótulos entre os LSRs, ele é um conjunto de procedimentos e mensagens que mapeia os caminhos comutados na camada de enlace associando FECs a cada LSP que cria. Dois LSRs usando LDP para trocar informações para assim fazer o mapeamento de rótulos são chamados pares LDP. Existem quatro categorias de mensagens LDP:

- **Mensagens de descoberta:** São as mensagens usadas pelos LSRs para indicar sua presença na rede e fazer com que os outros roteadores ao seu redor o conheçam;
- **Mensagens de seção:** São as mensagens usadas para estabelecer, manter e terminar sessões entre pares LDP;
- **Mensagens de anúncio:** São as mensagens usadas para criar, alterar e excluir mapeamentos de rótulos, levando-se em conta as informações contidas nas FECs;
- **Mensagens de notificação:** São as mensagens usadas para fornecer informações sobre o andamento dos processos como, por exemplo, informações de erros.

No mecanismo usado pelo LDP, primeiramente, são distribuídos os rótulos de acordo com as FECs para que, assim os pacotes sejam roteados através da rede MPLS. Na etapa de distribuição de rótulos e mapeamento da rede, são enviadas mensagens de descoberta partindo da borda da rede MPLS. Essas mensagens passam de LSR em LSR até chegar na

borda de saída da rede. Nessas mensagens são enviadas requisições de rótulos entre os roteadores para que sejam estabelecidos os pares LDP entre cada um deles. Esse mapeamento é feito para cada FEC requerida. Após esta montagem da rede MPLS, que é feita usando-se mensagens TCP padrão, é iniciado o envio dos pacotes já rotulados, usando o protocolo UDP. Sendo assim, o atrazo no envio dos pacotes diminui bastante, pois, agora, são lidos apenas os rótulos (*labels*) no transporte dos pacotes de uma borda a outra da rede.

O LDP pode operar em dois modos de distribuição de rótulos, o Modo de Distribuição Sob Demanda, no qual os LSRs só distribuem etiquetas ao par LDP se este solicitar e o Modo de Distribuição Não-solicitada, onde os LSRs distribuem etiquetas de tempos em tempos sem que haja uma solicitação explícita. Este último modo tem a vantagem de deixar as LIBs dos roteadores sempre atualizada, mas tem a grande desvantagem de, em vários casos, sobrecarregar a rede.

Abaixo, é mostrado um esquema de distribuição dos rótulos em uma rede MPLS onde o LDP atua sob demanda. Nele, duas FECs são representadas pelos endereço IP 128.89.-.- e 171.69.-.-, as quais são utilizadas para criar os LSPs.

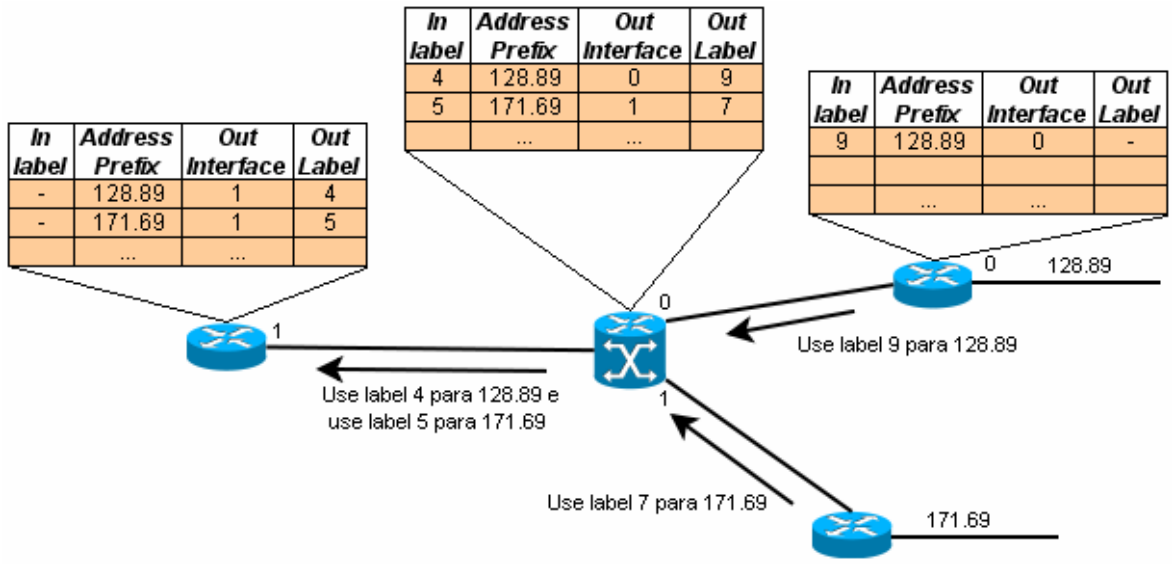


Figura 3.5: Distribuição dos rótulos

Após a distribuição dos rótulos, e conseqüente mapeamento da rede MPLS, os pacotes de dados são transportados apenas fazendo-se a troca dos *labels* em cada roteador de acordo com as LIBs já criadas. A seguir é mostrado o esquema do encaminhamento de pacotes, complementando o exemplo anterior.

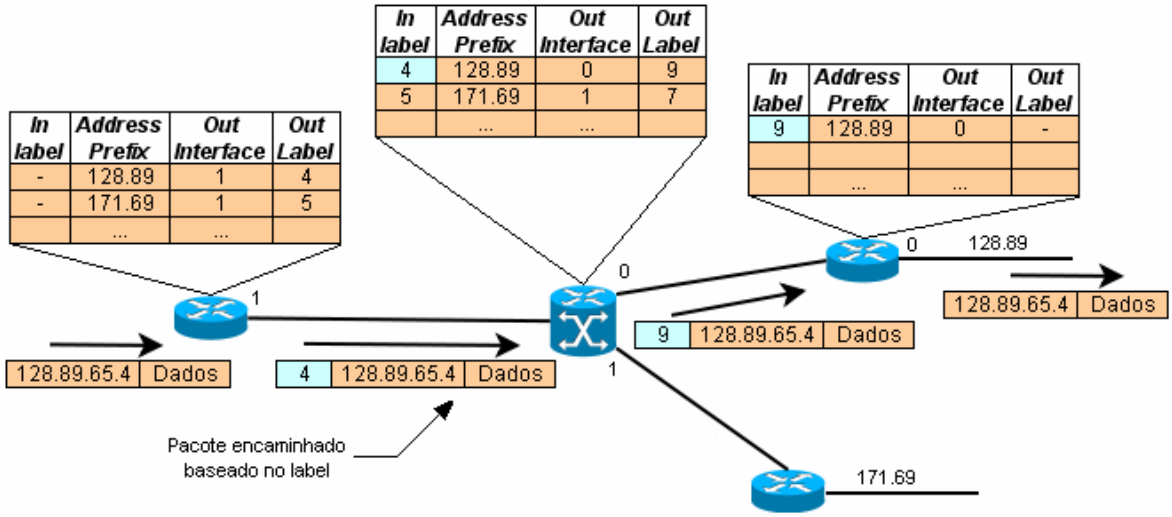


Figura 3.6: Encaminhamento do pacote

3.6 - CONCLUSÃO

O MPLS proporciona uma melhora significativa do processo de encaminhamento de pacote devido a sua simplicidade, evitando a necessidade de realizar análise do cabeçalho IP ao longo do caminho, e criando um ambiente de suporte controlado de Qos. Vários fabricantes desenvolveram diferentes técnicas de comutação de etiquetas e o MPLS surgiu como um padrão que unificou estas tecnologias. O MPLS permite a integração do IP com ATM e diversas outras tecnologias de camada 2 e camada 3; suporta a convergência de serviços (voz , dados e vídeo) ; oferece novas oportunidades à Engenharia de Tráfego e suporte a VPN.

Adicionando etiquetas de tamanho fixo reduzido, da mesma forma que o CEP ajuda na classificação das cartas, a performance no processamento dos pacotes é melhorado e o controle de Qos pode ser facilmente aplicado.

O MPLS é uma tecnologia flexível, que permite sua aplicação em várias tecnologias de rede. Sua característica de interligar roteadores IP o torna referência para construção de redes que proporcionam a convergência de telefone, vídeo e serviços computacionais.

Dentro da realidade do *Mobile IP*, que é o foco deste projeto e, tendo-se em vista todas as suas qualidades, o MPLS demonstra ser uma boa solução para o problema do atraso causado pela necessidade dos pacotes enviados ao nó móvel terem de passar pelo agente nativo. Além disso, ele também pode proporcionar uma diminuição no tempo gasto para serem feitos os *handoffs*, que resulta em um curto período no qual o nó móvel permanece sem comunicação. Portanto, a integração do MIP com o MPLS aumentaria a eficiência no

envio de pacotes de dados. No próximo capítulo são apresentadas as implementações e informações correlatas sobre a tecnologia *Mobile IP* e suas possíveis melhorias.

4 - IMPLEMENTAÇÕES HMIP E MPLS

A tecnologia Mobile IP, como já citado anteriormente, apresenta algumas deficiências. Estas deficiências, em sua maioria dizem respeito a rapidez no transporte de dados e no *handoff*. No que tange ao transporte de dados, o problema está no fato de quando o nó móvel está fora de sua rede nativa, ou seja, em uma rede estrangeira, os pacotes recebidos ou enviados por ele terem que, necessariamente, passar pelo agente nativo. Isso pode causar um grande atraso, dependendo da distância entre a rede estrangeira e a rede nativa.

No que diz respeito ao atraso de *handoff*, o problema está no tempo que se gasta quando o nó móvel se desloca de uma rede estrangeira para outra. No Mobile IP tradicional, quando isso ocorre, o nó móvel faz outra autenticação em sua rede nativa, ficando um tempo sem comunicação até que a conexão seja restabelecida na nova rede estrangeira. Dependendo da distância entre a rede estrangeira e a rede nativa, esse tempo de espera pode ser muito grande, o que pode causar conseqüências indesejadas ao usuário.

O foco deste capítulo é demonstrar por meios práticos algumas propostas que maximizem da eficiência do Mobile IP. O primeiro dos testes realizados foi a implementação física do HMIP. O objetivo deste teste foi demonstrar fisicamente como funciona o Mobile IP Hierárquico.

Os testes seguintes foram feitos por meio do software simulador OMNeT++, o qual nos deu uma gama maior de recursos para demonstrar importantes comparações entre o Mobile IP tradicional e o Mobile IP integrado às propostas de eficiência. Foram feitos testes comparativos usando as Tecnologias HMIP e MPLS.

A seguir, estão dispostos detalhadamente os testes realizados no trabalho.

4.1 - IMPLEMENTAÇÃO FÍSICA DO HMIP

Como já mencionado, um dos problemas do Mobile IP é o tempo gasto no *handoff* quando o nó móvel passa de uma rede estrangeira para outra. Isso ocorre pois, quando se passa de uma rede para outra, é necessária uma nova autenticação na rede nativa para que seja restabelecida a conexão. Como explicado detalhadamente no capítulo 2 deste trabalho, o Mobile IP Hierárquico representa uma boa alternativa para este problema, pois usa o conceito de micromobilidade para não haver a necessidade de uma nova autenticação do nó móvel em sua rede nativa. O intuito desta implementação foi demonstrar detalhadamente como funciona o HMIP visto fisicamente. A seguir, são explicadas

algumas das características do *Dynamics Mobile IP*, o software usado na implementação do Mobile IP Hierárquico.

4.1.1 Dynamics Mobile IP

Para este teste foi usado o *Dynamics* por alguns motivos, dentre eles estão: ele se baseia em IPv4, que é a tecnologia de rede utilizada no laboratório no qual foi realizado este teste; possui licença GPL e não é restrito a distribuições ou *kerneis* exclusivamente UNIX.
O *Dynamics Mobile IP* foi inicialmente desenvolvido na Universidade de Tecnologia de Helsinki (HUT – *Helsinki University of Technology*) e, desde julho de 2003, tornou-se um projeto aberto no *SourceForge.Net*. Ele é um software que, ao ser instalado e configurado em uma rede onde se necessita a utilização do Mobile IP, implementa o protocolo. O *Dynamics*, além de implementar a especificação contida na RFC 3344 [22], também implementa o registro regional RFC 4857 [24], que possibilita o IP Móvel Hierárquico. Portanto, neste projeto, foi escolhido trabalhar com a versão de linux do nó móvel.
Abaixo são dados os passos para a instalação do *Dynamics Mobile IP*:

- Passo 1: Instala-se o pacote apropriado do Dynamics para cada componente da rede, ou seja, no HA, instala-se o pacote dynamics-ha; no GFA e nos Faz instala-se o pacote dynamics-fa e no MN instala-se o pacote dynamics-mn.
- Passo 2: Configura-se os arquivos ".conf" de cada componente da rede de acordo com a topologia da rede.
- Passo 3: Depois de todos os componentes de rede devidamente configurados dá-se o *start* no dynamics, iniciando o serviço em cada um deles.

No teste feito, foi utilizada a versão 0.8.1 do *Dynamics* e o sistema operacional utilizado em todas as máquinas da rede foi o *Fedora Core 5*. Este sistema operacional foi escolhido pelo fato do Dynamics funcionar bem em tal ambiente.

4.1.2 Características da rede implementada

No teste realizado, implementou-se a topologia básica de uma rede que utilizasse o Mobile IP Hierárquico. A topologia da rede com os endereços IP das interfaces de rede utilizados está representada a seguir.

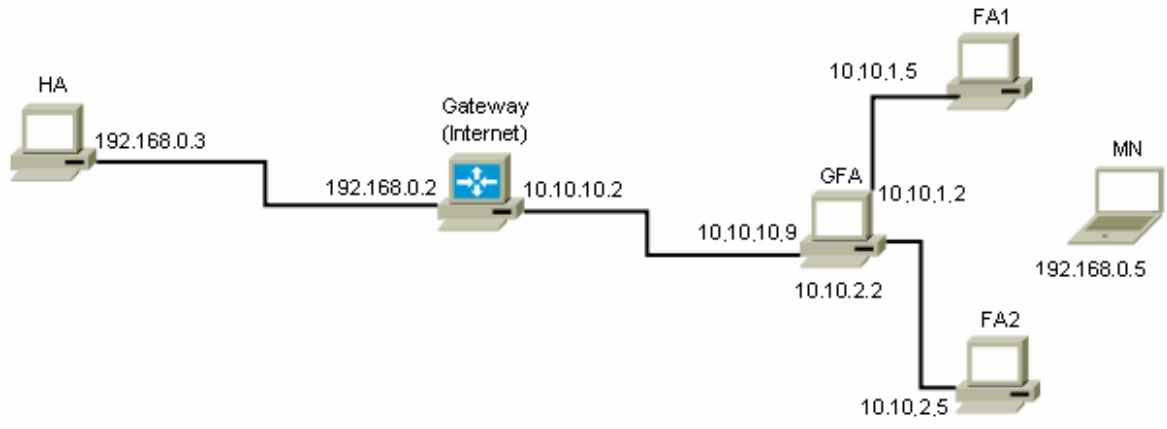


Figura 4.1 – Topologia da rede com os endereços de todas as interfaces de rede utilizadas.

Na etapa de configuração da rede, configurou-se o arquivo ".conf" de cada um dos componentes da rede. Em cada um deles, as configurações principais seguem a seguir:

- **Configuração do "dynhad.conf", o arquivo de configuração do HA:**

Neste arquivo, em seu início, configura-se o modo de descoberta do HA, se ele manda ou não mensagens em *broadcast*. Além disso, configura-se o envio automático de anúncios enviados e o tempo entre um e outro. O endereço do agente também pode ser configurado Abaixo, é mostrado trecho do arquivo:

```
# Interfaces to be used for Mobile IP services. Note that you have to
#configure
# each interface that may receive or send registration messages.
# interface: name of the interface, e.g. eth0
# ha_disc:
#    0 = do not allow dynamic HA discovery
#    1 = allow dynamic HA discovery with broadcast messages
# agentadv:
#    0 = do not send agent advertisements without agent solicitation
#    1 = send agent advertisements regularly
#   -1 = do not send any (even solicited) agent advertisements
# interval: number of seconds to wait between two agentadvs
#           (if allowed for this interface)
# force_IP_addr: local address to be forced for this interface
#            (can be used to select one of the multiple virtual
#            addresses); if not entered, the primary address of the
#            interface is used
INTERFACES_BEGIN
# interface  ha_disc  agentadv  interval  force_IP_addr
eth0         1        1         20
#eth1        1        1         20        192.168.240.2
INTERFACES_END
```

Além disso, configura-se a porta de comunicação para receber pedidos de registros em:

```
# UDP port to listen for registration requests
# The default is 434
UDPPort 434
```

- **Configuração do "dynmnd.conf", o arquivo de configuração do MN:**

Neste arquivo, em seu início, configura-se o endereço IP do MN:

```
# The Mobile Nodes's IP address in the Home Network.
# If using AAA (see UseAAA below), home address can be set to 0.0.0.0 in
#order to request a home address from the AAA infrastructure. This
#requires that also MN NAI is configured.
MNHomeIPAddress 192.168.0.5
```

Em seguida, deve-se configurar o endereço IP da rede nativa no trecho:

```
# The IP address of Mobile Node's Home Agent. In case of a private HA
#address
# this is the address of the surrogate HA. If the HA address is unknown,
#set
# this to 0.0.0.0 and make sure that HomeNetPrefix is correct for dynamic
# HA address resolution or use AAA to discover HA address. If the HA has
# multiple interfaces, this should be the address of the "public"
#interface,
# i.e., the one toward default gateway (it has to be reachable from the
#foreign networks).
HAIPAddress 192.168.0.3
```

Neste arquivo, ainda podem ser configurados o prefixo da rede nativa e o gateway usado pela sua rede nativa:

```
# Network address of home network (CIDR format: a.b.c.d/prefix_length)
# This is used with FA decapsulation and dynamics HA address resolution.
#If
# commented, the routing entry is not removed nor added. The home net
#entry
# may optionally be used with MN decapsulation - see
#MNDecapsRouteHandling
# option below.
#
# Example: 192.168.242.0/24
HomeNetPrefix 192.168.0.0/24

# Home net default gateway
# This entry can be used to force a gateway that the MN uses when it is
# at home. If this is left commented, the MN tries to use the default
#route
# that was in use when the program was started.
#
HomeNetGateway 192.168.0.2
```

Assim como no arquivo de configuração do HA, no MN também pode-se configurar a porta de comunicação:

```
# UDP port to be used for sending registration requests
# Port 434 is allocated for Mobile IP signaling and this should not be
#changed unless the network is known to use some other port (i.e. all the
#FAs and HAs must have the same port configured).
UDPPort 434
```

- **Configuração do "dynfad.conf" no GFA:**

Neste arquivo, em seu início há uma importante configuração que difere o GFA de um FA. Nesta parte do arquivo, configura-se as interfaces de rede do GFA. Nela, configura-se o tipo de interface, o envio automático de anúncios, além do tempo entre os anúncios:

```
# Interfaces to be used for Mobile IP services
# interface: name of the interface, e.g. eth0
# type:
#    1 = both upper and lower direction
#    2 = only upper direction (to upper FA / HA)
#    3 = only lower direction (to lower FA / MN)
#    Note: Only one interface can be used for upper direction, but
#    multiple interfaces can be used for lower direction.
# agentadv:
#    0 = do not send agent advertisements without agent solicitation
#    1 = send agent advertisements regularly
#   -1 = do not send any (even solicited) agent advertisements
# interval: number of seconds to wait between two agentadvs
#           (if allowed for this interface)
# force_IP_addr: local address to be forced for this interface
#            (can be used to select one of the multiple virtual
#            addresses); if not entered, the primary address of the
#            interface is used
#
# In the example below, interface "eth0" can be used for both upper and
#lower directions, Agent Advertisements are sent regularly with the
#interval of 30 s, and the primary address of the eth0 interface is used.
# Correspondingly, interface "eth1" would allow only lower direction
# connections (connections with MNs or lower FAs) with periodical
# Agent Advertisements with an interval of 20 s, and a specific IP
#address would be forced to the interface "eth1".
#
# The last entry that is of type upper (or both upper and lower), is
# used to send messages in the upper direction.
#
INTERFACES_BEGIN
#  interface  type  agentadv  interval  force_IP_addr
eth0  2     -1    10
eth1  3     -1    20
#eth2 3     -1    10
INTERFACES_END
```

No trecho acima, ao colocar o *type* da interface "eth0" como sendo "2", estamos afirmando que essa interface só se comunica na direção do HA. Ao colocar o *type* da interface "eth1" como sendo "3", estamos afirmando que essa interface só se comunica na direção dos outros FAs e, por conseqüência, do MN. Ao colocar o parâmetro *agentadv* como sendo "-1", estamos afirmando que o GFA não deve mandar anúncios. Isso ocorre pois, quem irá mandar os anúncios serão os FAs.

No trecho a seguir configura-se a interface do FA de mais alta hierarquia, ou seja, o próprio GFA.

```
# Address of the highest FA
# This address is used in the communication with the HA and it is
#advertised
# in agent advertisement messages. This should be from the "public side"
# interface of the FA (i.e., the interface that is toward HA or the
#default
# gateway).
# If this FA is the highest FA for some organization, use its address
#here.
# In this case, the address would most probably be from the interface
#that
# is configured for upper direction (type 1 or 2 in the interface list
#above).
HighestFAIPAddress 10.10.10.9
```

No trecho abaixo configura-se a interface de comunicação do FA de hierarquia acima do FA em questão. Como este já é o FA de mais alta hierarquia, coloca-se o endereço do próprio GFA:

```
# Address of the upper FA
# This is the address of the FA to which the requests are forwarded
# on they way to the Home Agent.
# If this is the same as the FA's own IP address,
# then this FA is really the highest FA and the requests are forwarded
# directly to the Home Agent.
UpperFAIPAddress 10.10.10.9
```

No trecho abaixo, coloca-se TRUE para que o componente da rede seja o FA de mais alta hierarquia, pois ele é o GFA:

```
# HighestFA < TRUE | FALSE >
# Whether this FA is the highest FA (i.e. it does not have any upper FAs
# and it communicates directly with the Home Agents).
HighestFA TRUE
```

- **Configuração do "dynfad.conf" no FA1:**

Os arquivos de configuração de todos os FAs são os mesmos dos GFAs, o que muda são os parâmetros configurados.

No início do arquivo, atribui-se os parâmetros das interfaces de rede do FA, colocando-se o tipo de interface, o envio automático de anúncios, além do tempo entre os anúncios:

```
INTERFACES_BEGIN
# interface  type  agentadv  interval  force_IP_addr
eth0  1     1     10
#eth0         1      1         30
#eth1        3     1         20           192.168.240.2
INTERFACES_END
```

No trecho acima, ao colocar o *type* da interface "eth0" como sendo "1", estamos afirmando que essa interface se comunica tanto na direção do HA quanto na direção do MN. Ao colocar o *type* da interface "eth1" como sendo "1", estamos afirmando que essa interface só se comunica na direção dos outros FAs e, por conseqüência, do MN. Ao colocar o

parâmetro *agentadv* como sendo “1”, estamos afirmando que o FA deve mandar anúncios regularmente.

No trecho abaixo configura-se a interface do FA de mais alta hierarquia. No caso, o FA de mais alta hierarquia é o GFA.

```
# Address of the highest FA
# This address is used in the communication with the HA and it is
advertised
# in agent advertisement messages. This should be from the "public side"
# interface of the FA (i.e., the interface that is toward HA or the
default
# gateway).
# If this FA is the highest FA for some organization, use its address
here.
# In this case, the address would most probably be from the interface
that
# is configured for upper direction (type 1 or 2 in the interface list
above).
HighestFAIPAddress 10.10.10.9
```

No trecho abaixo configura-se a interface de comunicação do FA de hierarquia acima do FA em questão. No caso, o FA de hierarquia acima deste é o GFA, portanto, coloca-se o endereço da interface de comunicação do GFA com o FA1.

```
# Address of the upper FA
# This is the address of the FA to which the requests are forwarded
# on they way to the Home Agent.
# If this is the same as the FA's own IP address,
# then this FA is really the highest FA and the requests are forwarded
# directly to the Home Agent.
UpperFAIPAddress 10.10.1.2
```

No trecho abaixo, coloca-se FALSE para indicar que o FA em questão não é o componente de mais alta hierarquia.

```
# HighestFA < TRUE | FALSE >
# Whether this FA is the highest FA (i.e. it does not have any upper FAs
and
# it communicates directly with the Home Agents).
HighestFA FALSE
```

- **Configuração do “dynfad.conf” no FA2:**

Este arquivo de configuração terá suas atribuições quase todas iguais às do FA1. O único ponto que muda é a configuração do *Address of the upper FA*:

No trecho abaixo configura-se a interface de comunicação do FA de hierarquia acima do FA em questão. No caso, o FA de hierarquia acima deste é o GFA, portanto, coloca-se o endereço da interface de comunicação do GFA com o FA2.

```
# Address of the upper FA
# This is the address of the FA to which the requests are forwarded
# on they way to the Home Agent.
# If this is the same as the FA's own IP address,
# then this FA is really the highest FA and the requests are forwarded
# directly to the Home Agent.
UpperFAIPAddress 10.10.2.2
```

4.1.3 Resultados da implementação do HMIP usando o Dynamics

Na apresentação dos resultados mostraremos as mensagens de pedido de autenticação, as respostas de autorização a esses pedidos, assim como os cabeçalhos que são colocados no tunelamento IP-em-IP feito entre os FAs e o HA para demonstrar o encapsulamento e o desencapsulamento feito entre os agentes.

No teste, primeiramente, foi feita a conexão do MN no FA1 para capturarmos os resultados referentes ao FA1. Em seguida, conectou-se o MN no FA2 para capturarmos os resultados referentes ao FA2. Foi usado o software de análise de protocolos de rede *Ethereal* versão 0.99 na captura dos resultados da implementação.

Para ficarem bem claros os resultados do teste, vamos dá-los na visão de cada um dos componentes da rede.

- **Resultados vistos a partir do HA**

Como explicado no capítulo 2, quando o MN está fora de sua rede nativa, ou seja, em uma rede estrangeira, ele manda pedidos de autenticação através dos FAs até chegar ao HA. Este pedido enviado pelo MN chegou ao HA da seguinte forma:

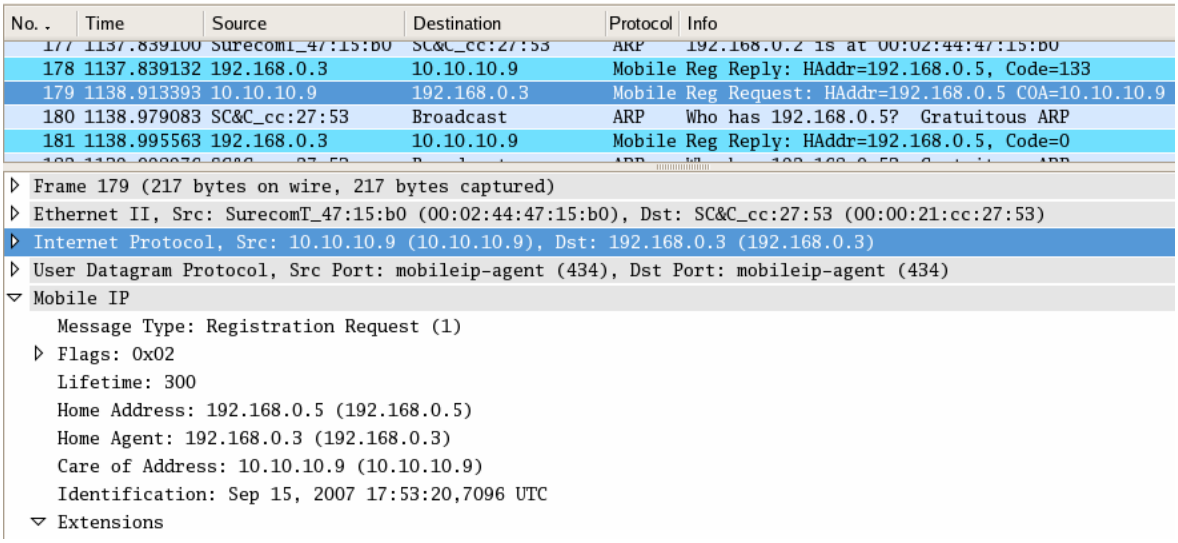


Figura 4.2 – Pedido de autenticação visto do HA

No sorfware *Ethereal* a mensagem aberta é sempre a que está selecionada em um tom te azul mais escuro na caixa mais acima.

Como podemos ver, esta mensagem de pedido de registro (*Reg request*) foi enviada ao HA a partir do GFA com o *Care of address* sendo o endereço do próprio GFA. Como podemos observar, além do COA, nesta mensagem constam ainda os endereços do HA e do MN.

A resposta ao pedido é enviada a partir do HA da seguinte forma:

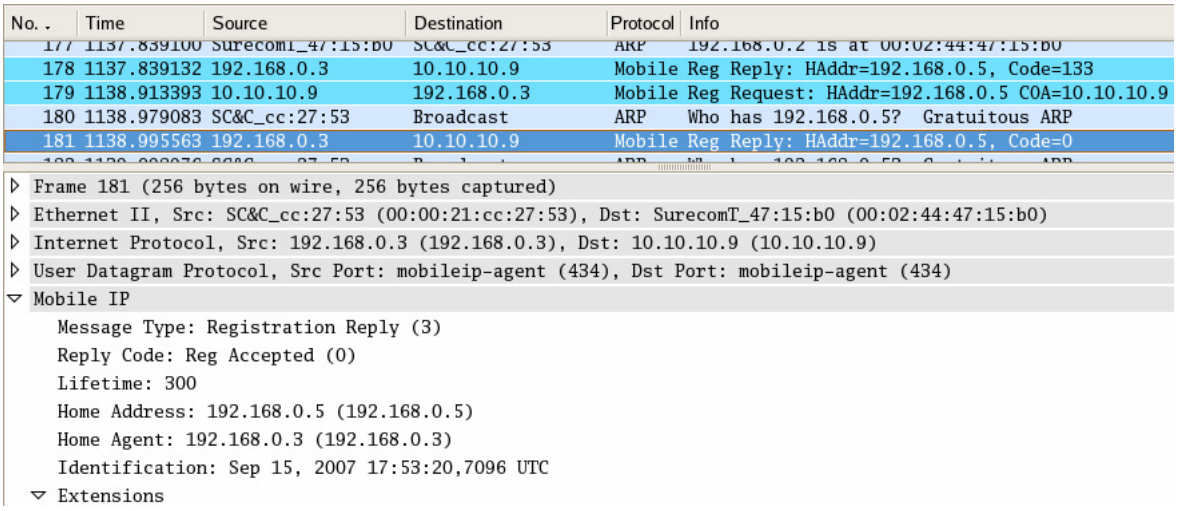


Figura 4.3 – Resposta ao pedido de autenticação visto do HA

A resposta positiva ao pedido se dá com o “Code=0”, que é o código de aceitação da autenticação. Esta é a mensagem de retorno, que é endereçada ao GFA e este, por sua vez, a reenvia até chegar ao MN.

Como intuito de observar o encapsulamento de pacotes, executou-se um comando “ping” do MN para o HA para confirmar a comunicação entre eles. Como resultado deste comando, podemos observar o encapsulamento e desencapsulamento do pacote no HA. A seguir, podemos observar o cabeçalho que encapsula o pacote até este chegar ao HA:

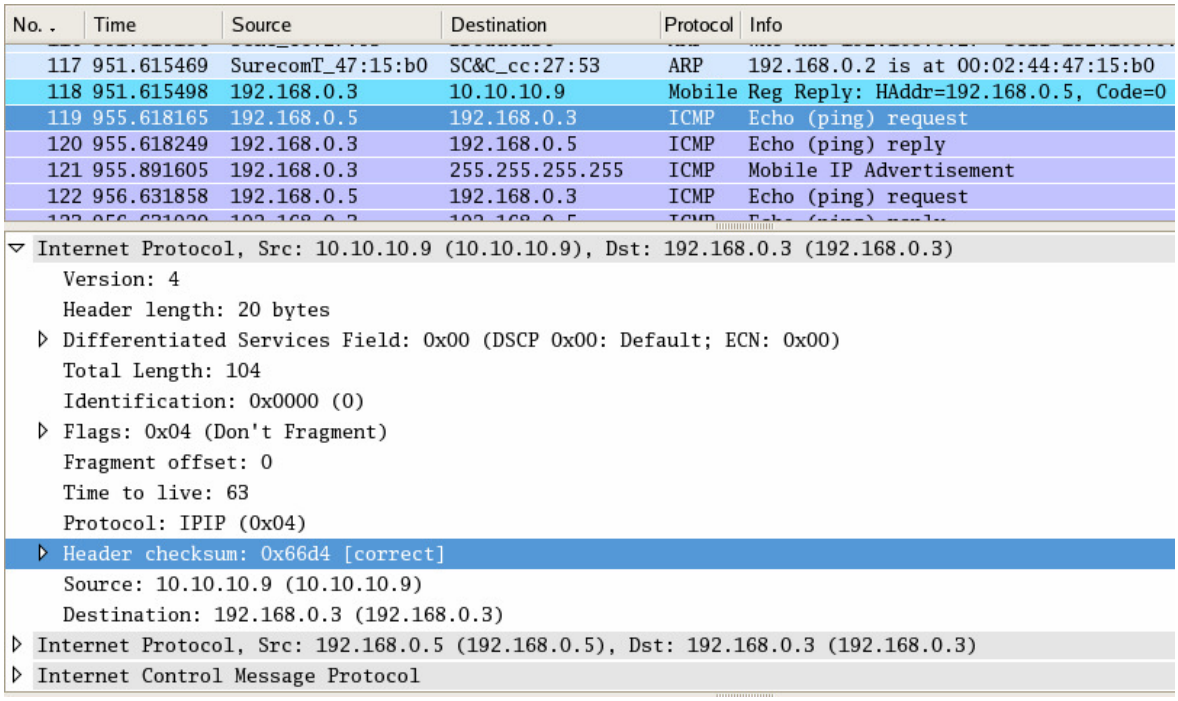


Figura 4.4 – Requisição do ”ping” feita pelo MN e enviada ao HA.

Em resposta ao “ping” o HA encapsula o pacote de resposta para enviá-lo de volta ao MN:

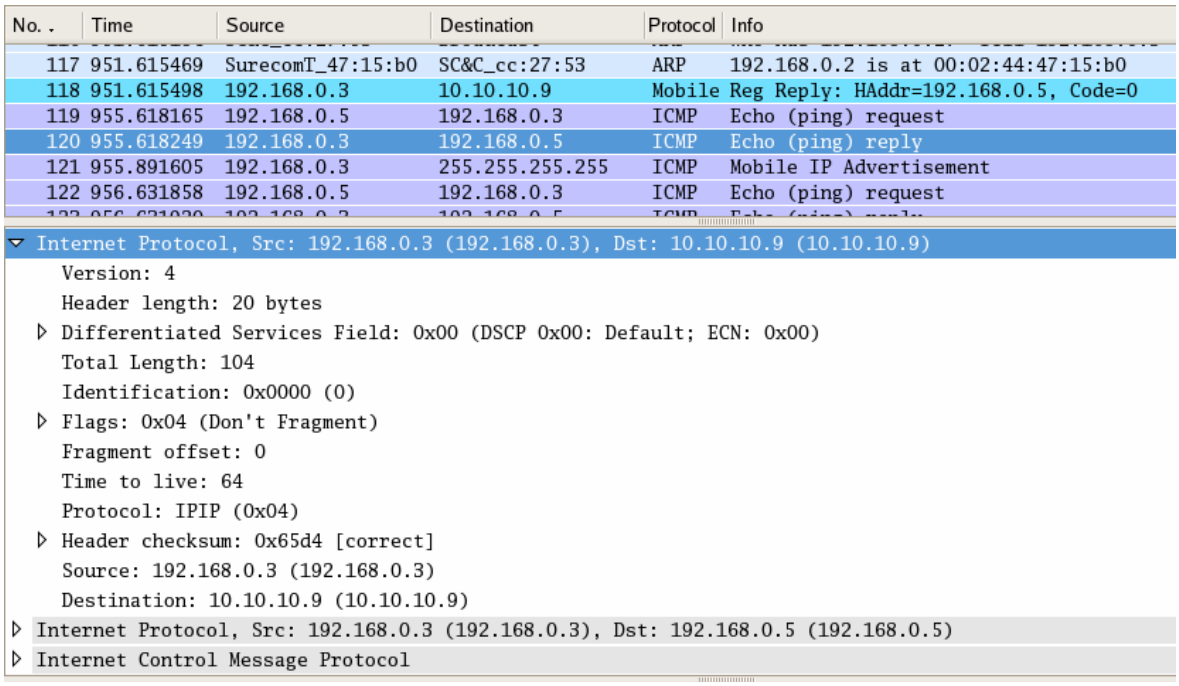


Figura 4.5 – Resposta do HA ao ”ping”.

O Dynamics ainda oferece uma ferramenta chamada *dynha_tool* que nos dá o status da conexão e uma listagem dos túneis estabelecidos, como vemos abaixo:

```
[root@Espanha ~]# /usr/sbin/dynha_tool
Dynamics Home Agent Control Tool v0.8.1
Using agent path "/var/run/dynamics_ha_admin"
> status
Home Agent status:
version            0.8.1
tunnels            1
request rejected   1
request accepted   7
discard(unk. ext)  0
discard(malformed) 0
discard(vendor)    0
advertisement sent 306
apicalls(admin)    1
apicalls(read)     0
> list
1 tunnels:
192.168.0.5
```

Como se pode observar, o número de registros rejeitados foi “1” e o número de registros aceitos foi “7”. Ainda podemos observar que o número de túneis estabelecidos foi “1” e este túnel foi feito até o MN, cujo endereço é mostrado.

Dentro dos resultados dos testes na visão do HA não há diferenças conectando-se o MN no FA1 e no FA2, pois, usando o conceito de micromobilidade, não chegaria mensagem de requisição de registro novamente ao HA na mudança entre o FA1 e o FA2. Além disso,

as mensagens que se vê no HA são provenientes de um só GFA, não importando o endereço do FA onde o MN está conectado. Portanto, mostrou-se apenas os resultados da conexão do MN no FA1.

- **Resultados vistos a partir do GFA**

Quando o MN está em uma rede estrangeira, no exemplo, no FA1, ele manda uma requisição de registro para o FA1 e este, por sua vez, manda esta requisição de registro ao GFA, como é mostrado abaixo:

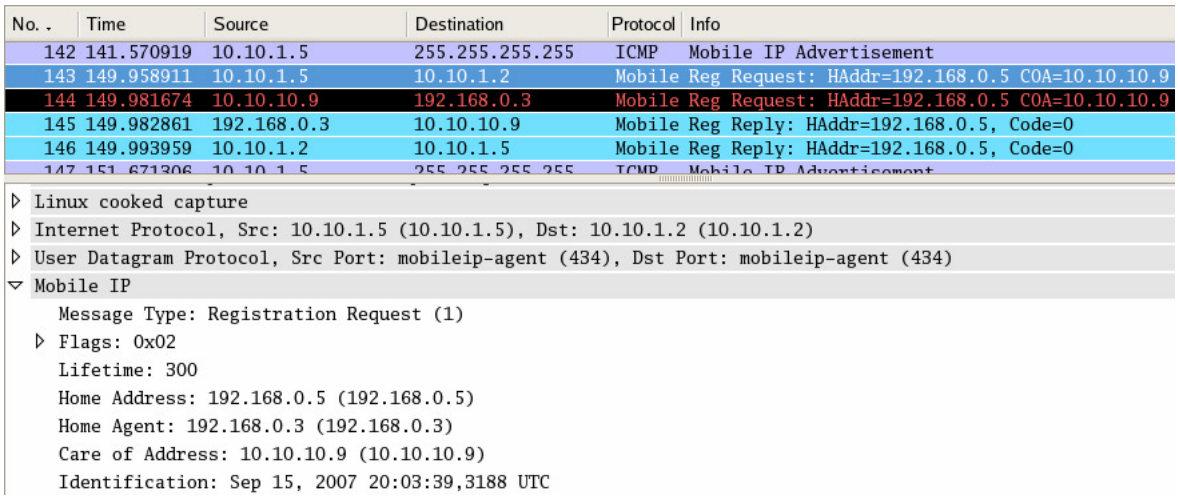


Figura 4.6 – Requisição de registro enviado do FA1 para o GFA.

Ao receber este pedido de registro, o GFA o envia para o HA da seguinte forma:

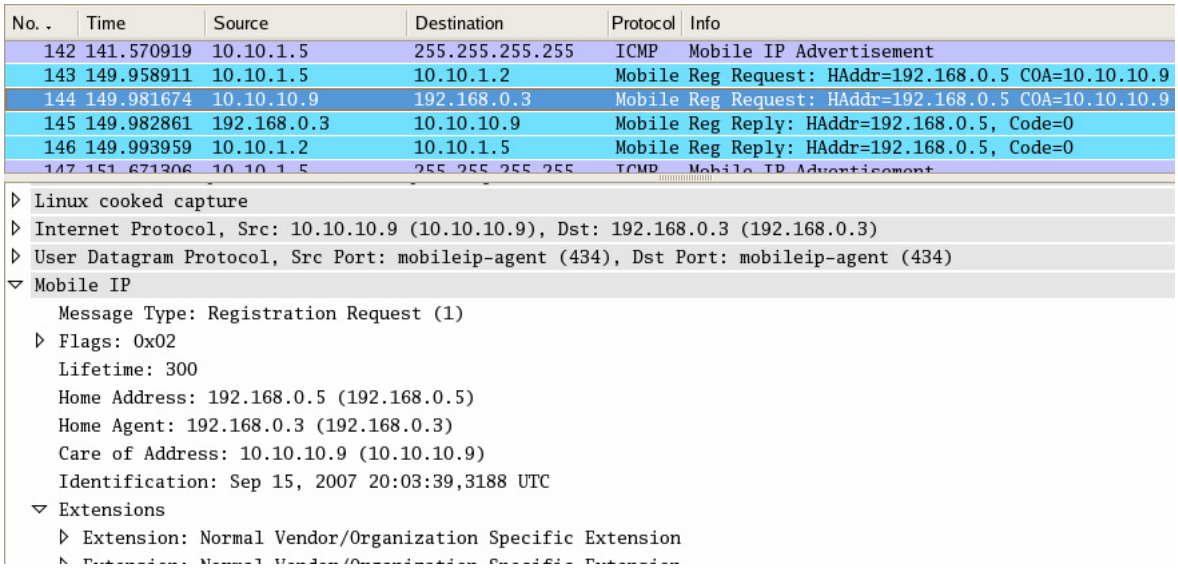


Figura 4.7 – Requisição de registro enviado do GFA para o HA.

Quando o HA recebe o pedido de registro e o aceita, retorna a seguinte mensagem para o GFA:

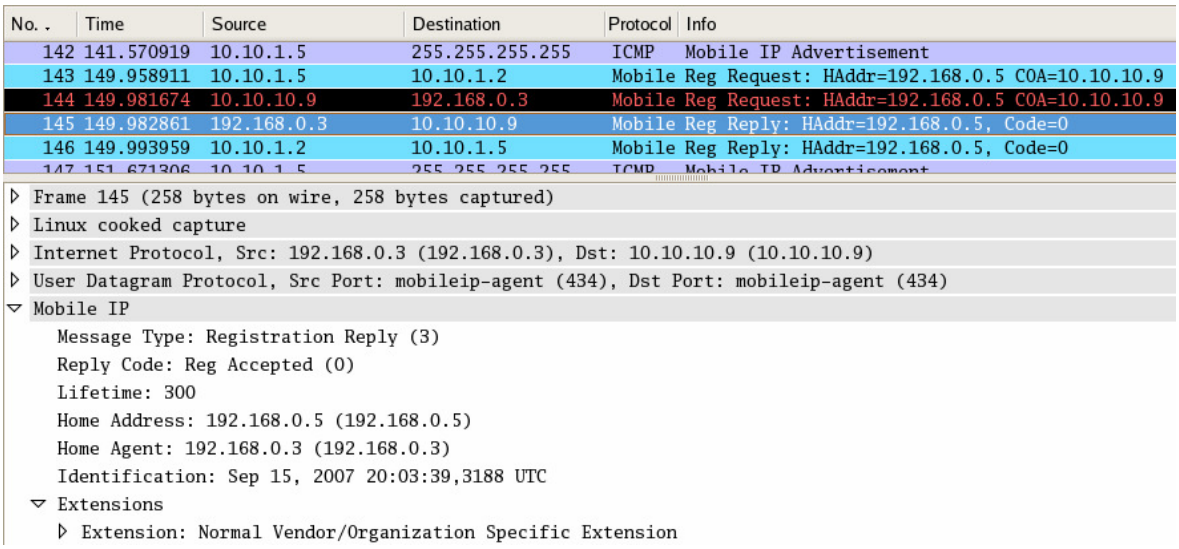


Figura 4.8 – Autorização de registro enviado do HA para o GFA.

Como vemos acima, o “Code=0” indica que a requisição de registro foi aceita.

O GFA, ao receber esta resposta positiva, a manda para o FA1 para que este a mande para o MN.

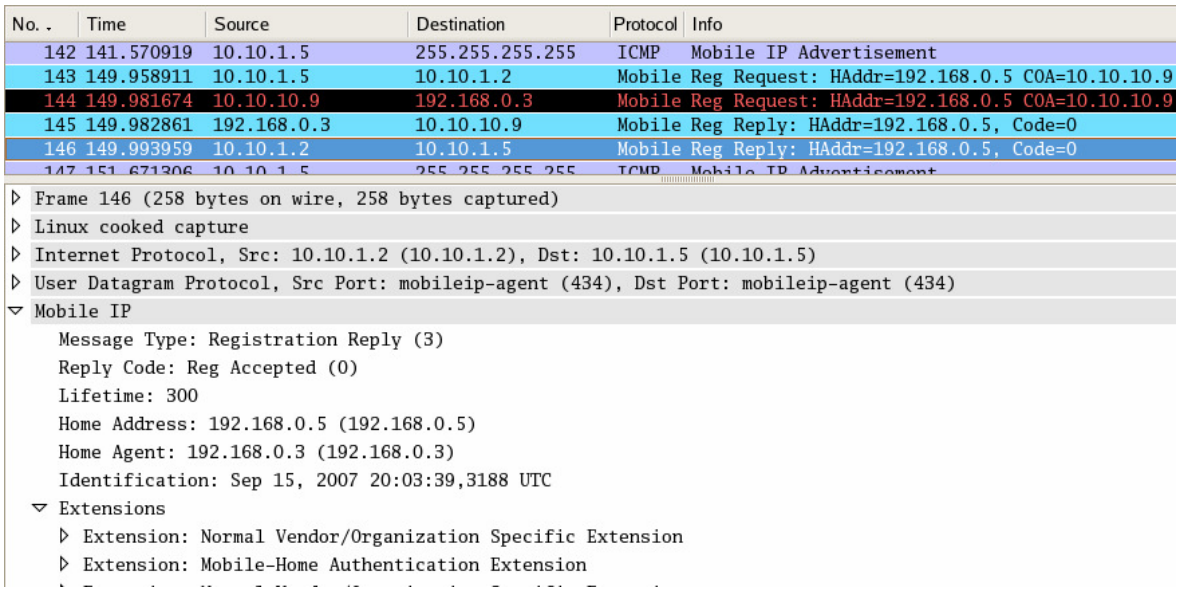


Figura 4.9 – Autorização de registro enviado do GFA para o FA1.

Executando-se um comando “ping” do MN para o HA para confirmar a comunicação entre eles, podemos observar que o pacote já encapsulado vem do HA, passa pelo GFA e é enviado ao FA1. Nas figuras a seguir, podemos observar a requisição do “ping” indo em direção ao HA e a resposta ao “ping” voltando do HA. No cabeçalho desses pacotes, podemos notar o encapsulamento na ida e na volta:

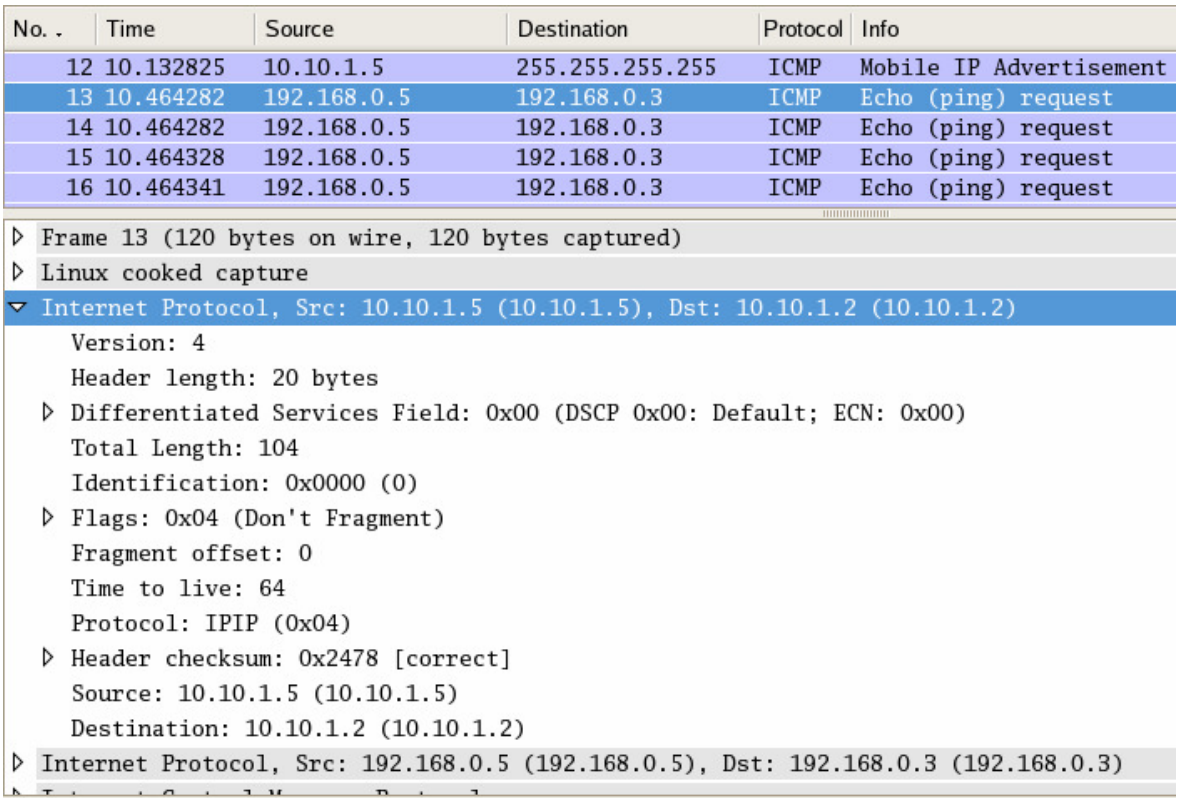


Figura 4.10 – Requisição do ”ping” feita pelo MN e enviada ao HA.

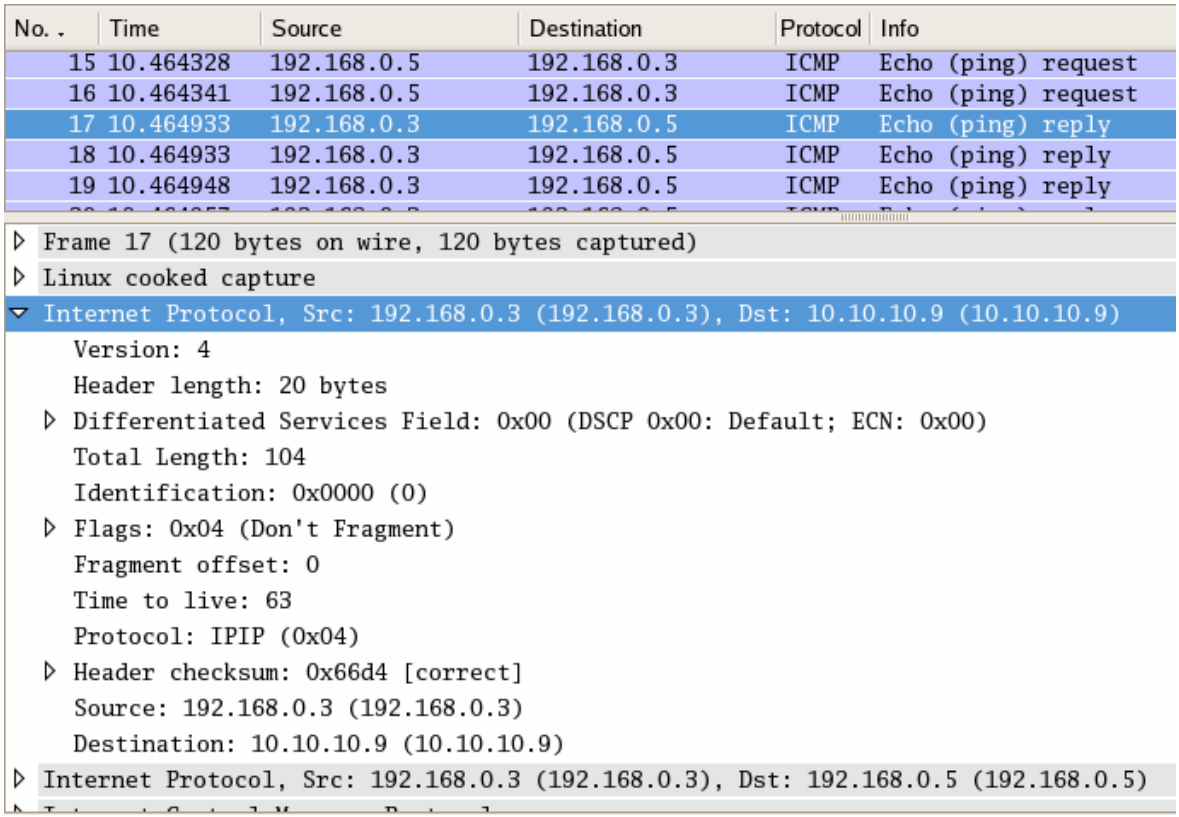


Figura 4.11 – Resposta do HA ao ”ping”.

Usando a ferramenta *dynfa_tool*, que nos dá o status da conexão e uma listagem dos túneis estabelecidos, obtemos os resultados abaixo:

```
[root@localhost ~]# /usr/sbin/dynfa_tool
Dynamics Foreign Agent Control Tool v0.8.1
Using agent path "/var/run/dynamics_fa_admin"
> status
Foreign Agent status:
version            0.8.1
tunnels            1
pending reg.req.   0
request rejected   0
request accepted   5
reply rejected     0
reply accepted     5
discard(unk. ext)  0
discard(malformed) 0
discard(vendor)    0
advertisement sent 0
apicalls(admin)    1
apicalls(read)     0
> list
1 tunnels:
192.168.0.5 192.168.0.3
```

Como vemos, estão indicados os endereços dos componentes que dão início e término do túnel IP-em-IP, representados pelos endereços IP `192.168.0.5` e `192.168.0.3`, ou seja, os endereços do MN e do HA.

Dentro dos resultados dos testes na visão do GFA, no processo de envio e recebimento de pacotes, a única diferença entre conectar o MN no FA1 e no FA2 é o endereço do FA que se comunica com o GFA. Já a autenticação na movimentação do MN de um agente para o outro se daria apenas dentro do domínio da rede estrangeira, usando o conceito da micromobilidade. O resultado dessa mudança e problemas ocorridos por deficiências no software Dynamics serão discutidos na sessão de discussão dos resultados.

- **Resultados vistos a partir do FA1**

Ao conectar-se ao FA1, o MN envia a ele a mensagem de requisição de autenticação destinada ao HA. A partir daí, esta mensagem segue pelo GFA até seu destino como já foi mostrado anteriormente. Se a resposta à requisição for positiva, a mensagem que sai do FA1 e chega ao MN é da seguinte forma:

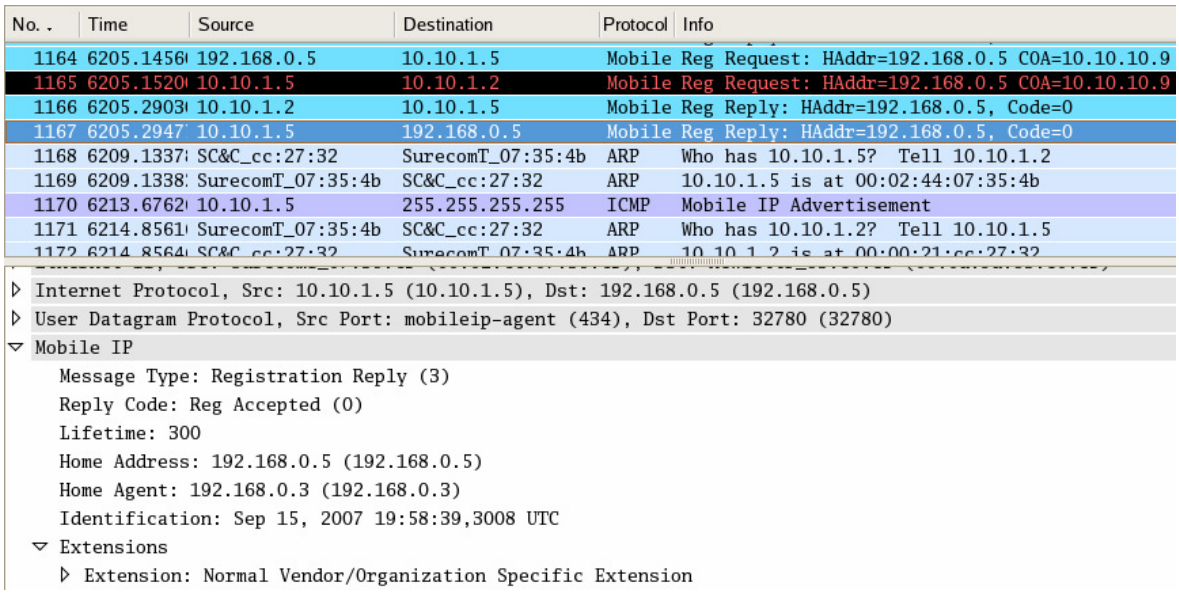


Figura 4.12 – Autorização de registro enviado do FA1 para o MN.

Como podemos observar, o “code=0” está presente em todas as mensagens positivas de requisição de registro.

Executando-se um comando “ping” do MN para o HA, como já sabemos, na volta (*reply*), o pacote já encapsulado vem do HA, passa pelo GFA e chega ao FA1. No FA1, o pacote é desencapsulado e enviado normalmente ao MN.

Na figura a seguir, podemos observar a requisição do “ping” indo em direção ao HA por meio do GFA enviado pelo FA1. A resposta ao “ping” já desencapsulada será mostrada no item que mostra dos resultados vistos do MN.

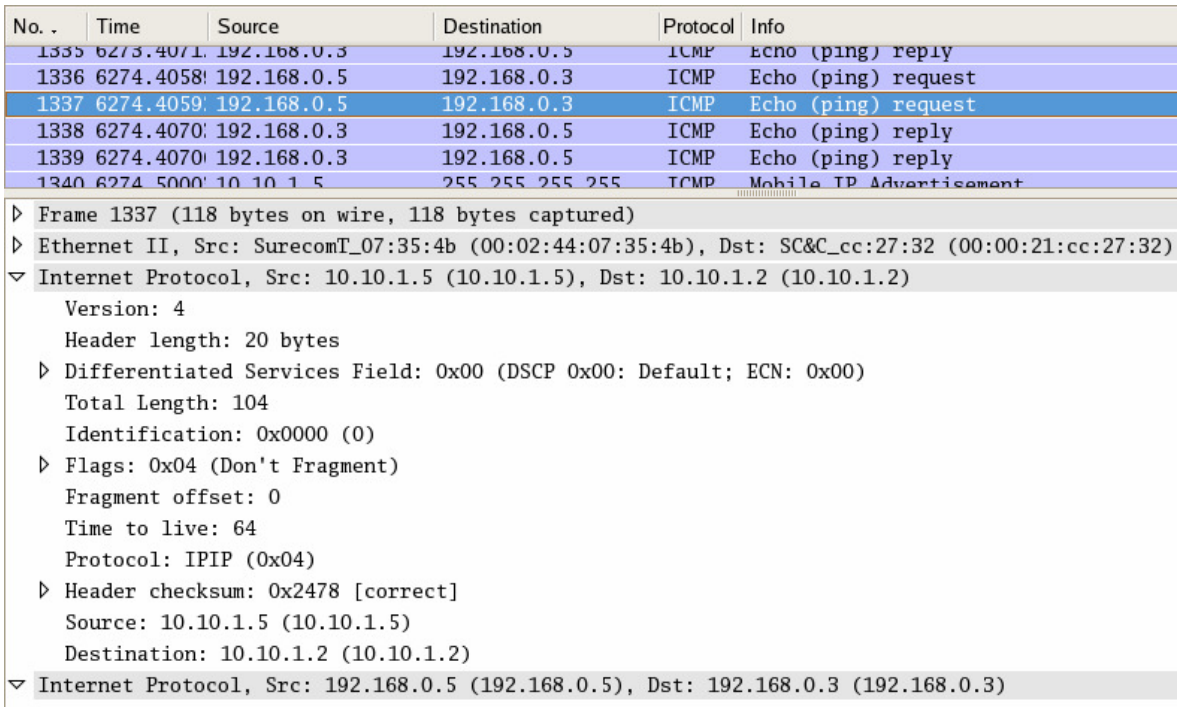


Figura 4.13 – Requisição do ”ping” feita pelo MN e enviada ao HA.

Usando a ferramenta *dynfa_tool*, que nos dá o status da conexão e uma listagem dos túneis estabelecidos, obtemos os resultados abaixo:

```
[root@localhost media]# /usr/sbin/dynfa_tool
Dynamics Foreign Agent Control Tool v0.8.1
Using agent path "/var/run/dynamics_fa_admin"
> status
Foreign Agent status:
version             0.8.1
tunnels             1
pending reg.req.    0
request rejected    0
request accepted    109
reply rejected      0
reply accepted      7
discard(unk. ext)   0
discard(malformed)  0
discard(vendor)     0
advertisement sent  690
apicalls(admin)     1
apicalls(read)      0
> list
1 tunnels:
192.168.0.5 192.168.0.3
```

Como podemos observar, novamente mostra-se um túnel estabelecido entre os endereços 192.168.0.5 e 192.168.0.3.

➢ **Resultados vistos a partir do FA2**

Quando o MN se movimenta do FA1 para o FA2, ele se desconecta do primeiro e se reconecta ao segundo, restabelecendo a comunicação direta com o HA. Ao fazer esta movimentação, para diminuir o tempo de *handoff*, o MN faz a requisição ao FA2, este faz a requisição ao GFA e este último já faz a autenticação do MN sem requerer nenhuma autenticação do HA. Sendo assim, a resposta positiva a requisição vem em pouco tempo e tem forma equivalente à resposta obtida quando se faz a autenticação passando pelo HA. As figuras abaixo mostram a mensagem de resposta positiva à requisição de autenticação recebida pelo FA2 e enviada por ele ao MN.

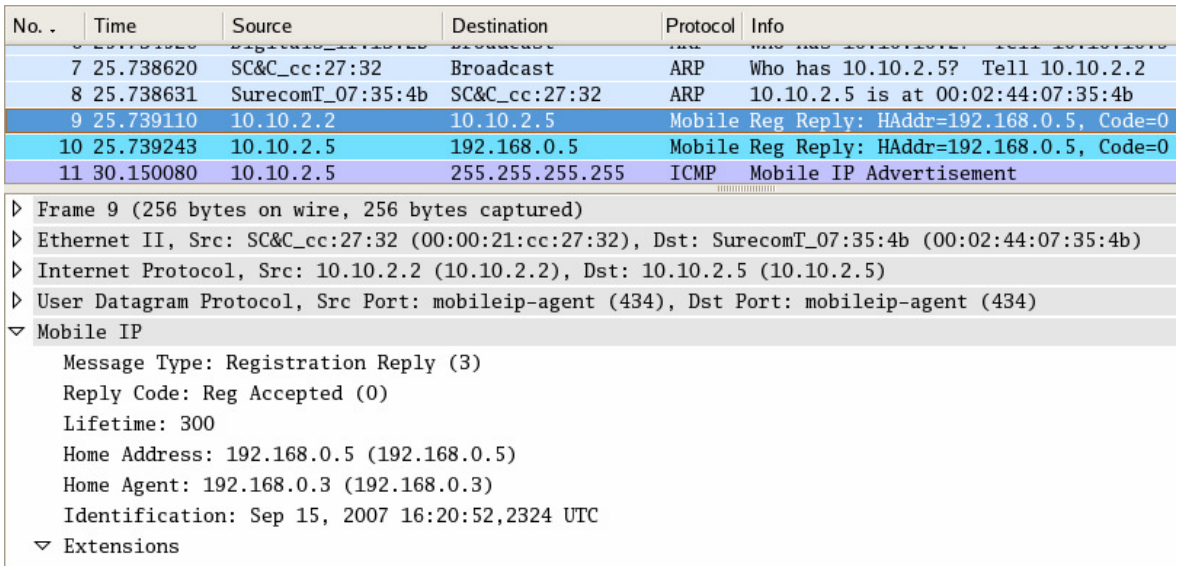


Figura 4.14 – Autorização de registro enviado do GFA para o FA2.

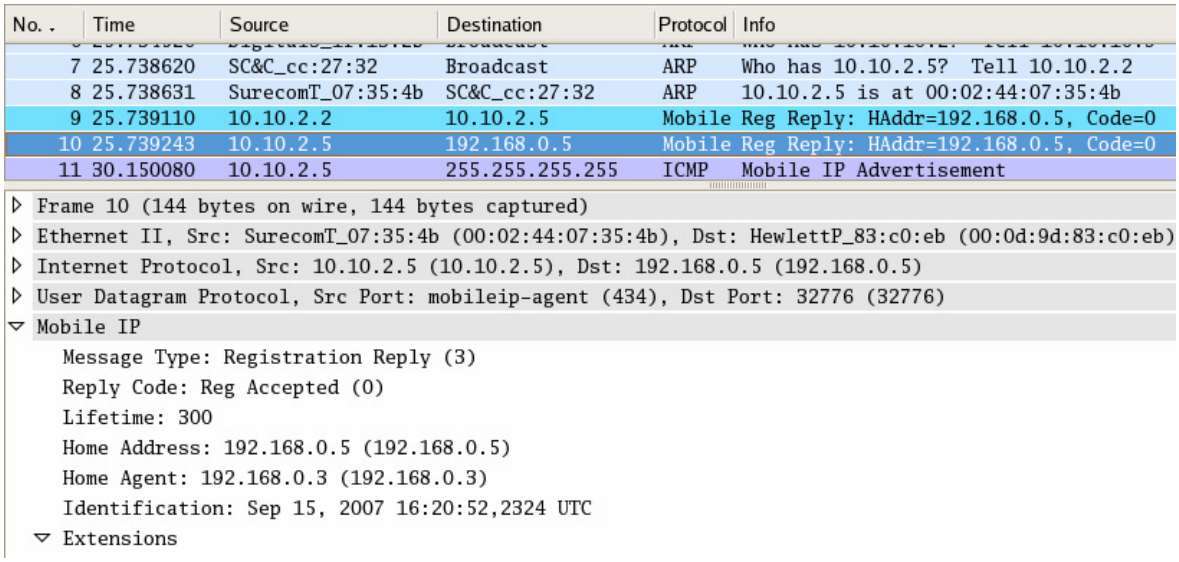


Figura 4.15 – Autorização de registro enviado do FA2 para o MN.

Ao executar o “ping” nesta situação, na qual o MN está conectado ao FA2, as mensagens de requisição e resposta são equivalentes às mostradas anteriormente referindo-se ao FA1.
A ferramenta *dynfa_tool*, assim como no FA1, também nos mostra um túnel estabelecido entre os endereços 192.168.0.5 e 192.168.0.3.

- **Resultados vistos a partir do MN**

Finalmente, após fazer a requisição de autenticação e esta passar por toda a trajetória já mostrada, o MN a recebe com resposta positiva para sua conexão direta com o HA. O pedido de requisição e a resposta a ele estão mostrados a seguir.

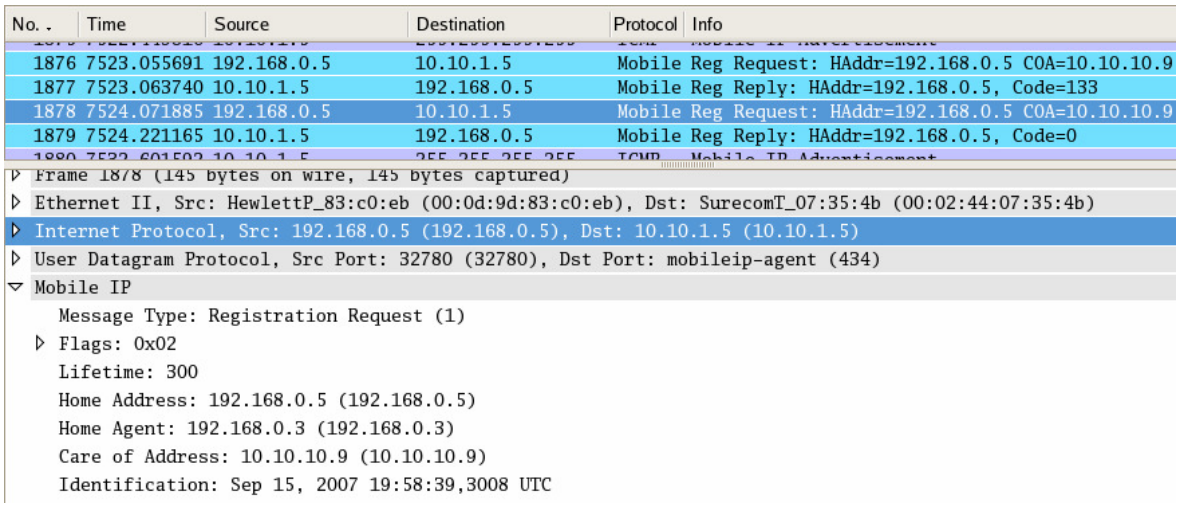


Figura 4.16 – Requisição de registro enviado do MN para o FA1.

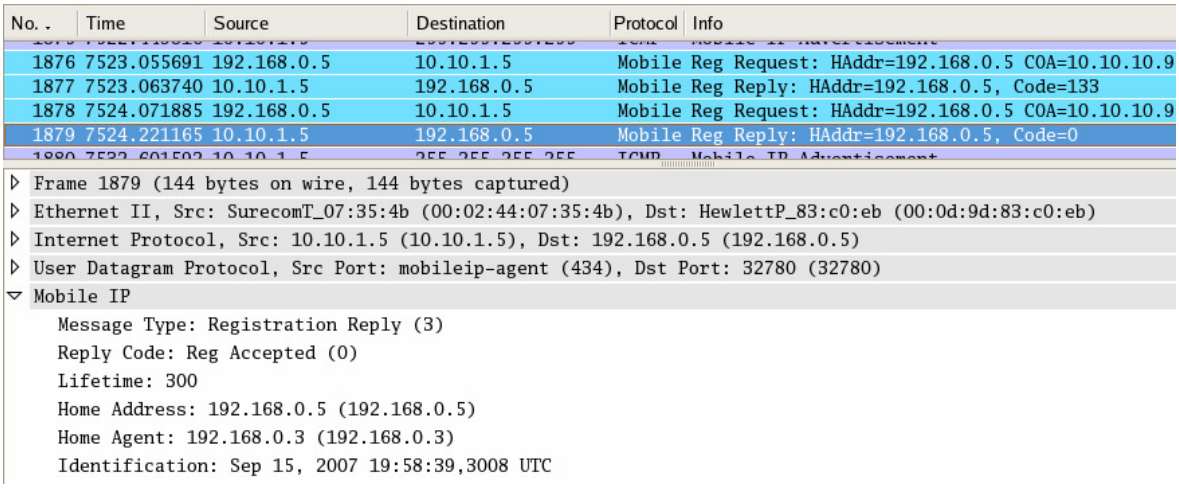


Figura 4.17 – Autorização de registro enviado do FA1 para o MN.

Ao executar o “ping” no MN, podemos observar que os cabeçalhos dos pacotes de requisição e dos pacotes de resposta estão todos em sua forma normal, ou seja os pacotes não estão encapsulados, como se pode ver a seguir.

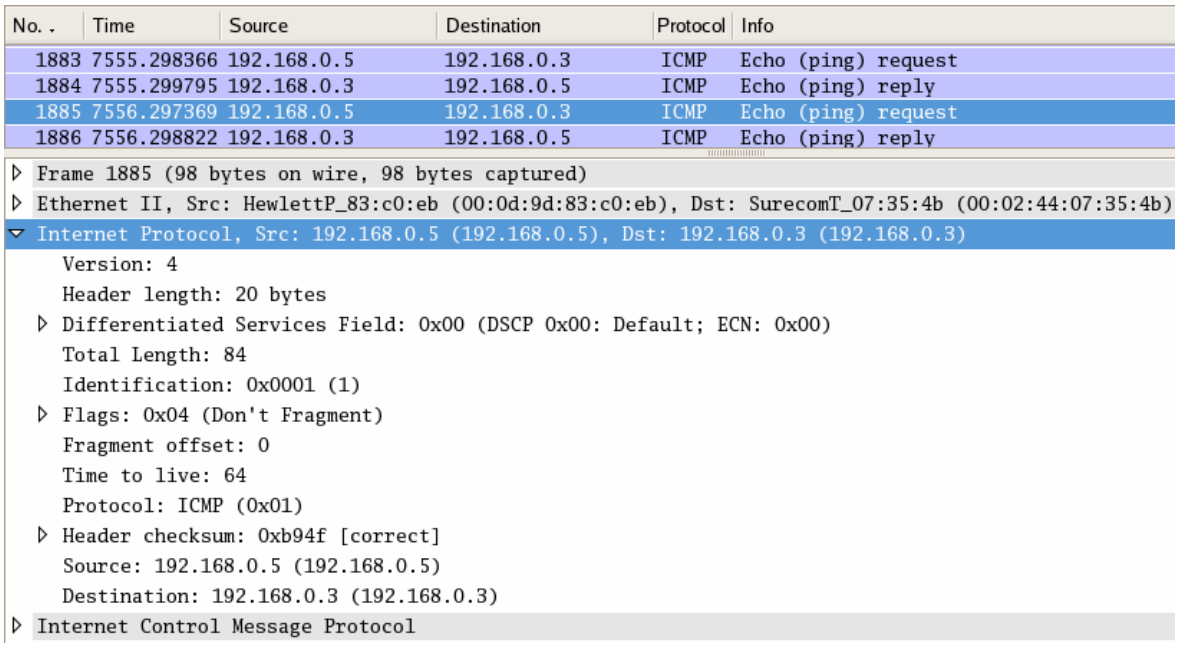


Figura 4.18 – Requisição do ”ping” feita pelo MN e enviada ao HA.

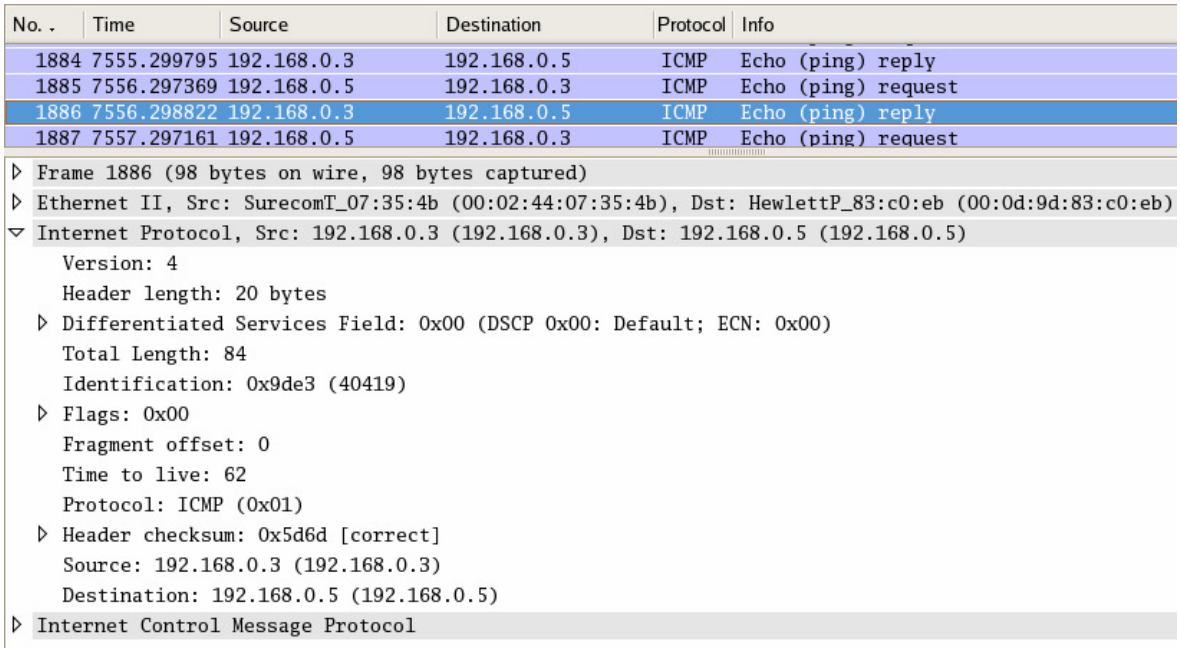


Figura 4.19 – Resposta ao ”ping”.

A conclusão a que chegamos é que o processo de encapsulamento e desencapsulamento de pacotes ocorrem no FA. Sendo assim, a partir do FA até o HA, os pacotes vão encapsulados seguindo o túnel IP-em-IP até chegarem ao seu destino.

A ferramenta *dynmn_tool*, ao ser usada no MN no fornece uma rica lista de informações que nos permite mapear a rede MIP. Como mostrado a seguir, podemos visualizar o

endereço local, o Care-of-adress, o FA e o HA. Pode-se também observar no item *tunneling mode* que o túnel foi formado com sucesso em todo o caminho do MN ao HA.

```
> status
Mobile status:
        state           Connected
        local addr      192.168.0.5
        co-addr         10.10.10.9
        FA-addr         10.10.1.5
        HA-addr         192.168.0.3
        Home addr       192.168.0.5
        tunnel is       up
        lifetime left   297s
        tunneling mode  full tunnel
        last request    3s ago; Sat Sep 15 16:03:53 2007
        last reply      3s ago; Sat Sep 15 16:03:53 2007
        reply code      0 - registration accepted
        info text       connection established
        active devices  1
        discarded msgs  0
> list
List of heard mobility agents:
10.10.1.5        eth0  prio 100 (-  0%), age  4s FA DYN IN-USE CURRENT
```

4.1.4 Discussão dos resultados da implementação do HMIP usando o Dynamics

A partir do software Dynamics Mobile IP e os testes realizados em laboratório, pôde-se entender exatamente como é implementado o Mobile IP Hierárquico. Informações muito importantes como o mecanismo de autenticação no HA e como essa autenticação passa de agente em agente foram demonstradas detalhadamente no teste. Além disso, pôde-se observar como são encapsulados e desencapsulados os pacotes enviados e recebidos pelo MN quando este está em uma rede estrangeira. Pôde-se observar também como o GFA trabalha como centralizador dos pacotes enviados ou recebidos no domínio da rede estrangeira.

Além das observações feitas no teste e mostradas no trabalho, algumas observações não puderam, em primeiro momento, ser feitas perfeitamente usando o Dynamics. O software, apesar de implementar de forma adequada a maioria das características do Protocolo Mobile IP, apresenta deficiências. A primeira delas é a instabilidade da conexão quando é feita a implementação do Mobile IP Hierárquico. Foi constatado em laboratório que, passado algum tempo de conexão do MN com o HA, esta conexão se perde, havendo necessidade de fazer o registro novamente.

Outra deficiência do software implementador do Mobile IP é na movimentação do MN entre um FA e outro usando o registro regional. Percebemos em laboratório que, ao mudar de um FA para outro, o MN demorava um período muito longo para começar a tentar refazer a conexão com o HA. Esta deficiência é de grande importância na escolha ou não do uso do software em situações onde devem ocorrer registros regionais.

Essas deficiências apresentadas pelo software foram atribuídas principalmente a falta de atualizações e pesquisas incorporadas ao software, visto que sua última atualização foi feita no ano de 2001. Isso pode causar uma série de conseqüências maléficas como incompatibilidades e falta de suporte. Constatou-se também que o mecanismo de registro regional não foi plenamente estruturado, implementado e testado no software, ficando algumas importantes lacunas a serem preenchidas, como é o caso da resposta rápido do GFA quando ocorre a micromobilidade do MN dentro do domínio da rede estrangeira.

Para obtermos resultados mais precisos traçando-se um paralelo entre o uso do Mobile IP tradicional e o Hierárquico, ainda foram feitas simulações usando o software OMNeT++, o qual nos deu resultados mais satisfatórios. Com este software ainda foram feitos testes do roteamento MPLS, que complementou ainda mais os resultados.

4.2 - ANÁLISE DO DESEMPENHO DO HMIP E DO MPLS ATRAVÉS DE SIMULAÇÕES

A seguir serão demonstrados alguns resultados de simulações a respeito do protocolo Mobile IP e seu aprimoramento com Registro Regional ou Mobile IP Hierárquico. Também foram realizadas simulações para comparação do desempenho usando o protocolo MPLS e um protocolo de roteamento IP denominado OSPF.

A opção que melhor atendia as necessidades deste trabalho foi o software de simulação OMNET++[2]. Essa ferramenta já possuía módulos disponíveis que descreviam os protocolos estudados neste trabalho além de ter uma linguagem de alto nível para descrição de topologias que o torna bastante versátil. Nesta seção sobre simulações, segue uma descrição de como o software funciona seguido da descrição do cenário para cada simulação. Concluímos o capítulo comparando resultados de desempenho do HMIP e do MIP e como uma rede MPLS poderia apresentar aperfeiçoamentos para esses protocolos.

4.2.1 - Software OMNET++

O simulador OMNET++ é um ambiente de simulação de evento-discreto orientado a objetos. Sua sigla quer dizer *Objective Modular Network Testbed in C++*. Por ser um software de código-aberto sua distribuição é livre para uso acadêmico. Sua área de aplicação é a simulação de redes de comunicação. Com essa ferramenta podemos simular:

- modelagem de tráfego em redes de telecomunicação;
- modelagem de protocolos;
- modelagem de redes de fila de espera (*queueing networks*);
- validação de arquiteturas de hardware;
- avaliação de performance de sistemas complexos de software.

O software oferece uma arquitetura de componente para os modelos. Os componentes (ou módulos) são programados em C++ e então são reunidos por uma linguagem de alto nível denominada NED, a qual serve para descrever a topologia da rede (Network Descriptor).

Uma das grandes vantagens do OMNET++ é que ele permite que o usuário implemente arbitrariamente novas funcionalidades além das fornecidas pelo próprio programa. Isso não acontece com outras ferramentas de simulação, como o NS ou COMNET, nas quais o usuário fica limitado as implementações fornecidas.

No entanto, existem vários modelos de simulação disponibilizados para o Omnet++, dentre eles temos modelos para IP, TCP, Ethernet , OSPF e outros. O comportamento de cada entidade definida nesses protocolos é realizado em códigos na linguem C++. Neste trabalho utilizamos os modelos HMIP e MIP que fazem parte da linha de trabalho "IPv6SuiteWithINET" e do modelo MPLS/RSVP_TE pertencente a linha de trabalho "INET". Para realizar comparações com o MPLS, foi usado um modelo de roteamento IP que neste caso é o OSPF.

Basicamente, para realizar a simulação de um modelo em particular, devemos descrever a topologia da rede usando a linguagem NED. Nessa linguagem descrevemos como as entidades de uma simulação irão se comunicar com outras entidades ou com elas mesmas. As entidades usam "mensagens" para se comunicar e é no arquivo NED que as portas de entrada e saída dessas mensagens são declaradas e como essas portas se conectam.

O comportamento de cada módulo (ou entidade) é feito através de um código em C++. Esse código que é responsável por gerar as mensagens, criar novos eventos, tratar mensagens recebidas dentre outras funcionalidades.

A configuração da simulação é descrita em um arquivo nomeDaTopologia.ini. Nesse arquivo podemos determinar a duração da simulação, qual topologia está sendo usada, em qual arquivo no formato xml está informações sobre tabelas de roteamento, inicializações de aplicações UDP nas entidades das redes, links TCPs e outros.

4.2.2 - Modelos de simulação utilizados

O simulador OMNET++ permite que se façam modelos de simulação arbitrariamente de modo que seja possível a implementação de um modelo para qualquer que seja o protocolo. No entanto, existem inúmeros modelos *source-free* de protocolos como IP, IPv6, OSPF, MPLS e outros. Neste trabalho foram utilizados os modelos para MIPv6, HMIPv6, MPLS e OSPF.

4.2.3 - Modelos para MIP e HMIP

Foram utilizados os modelos para IPv6[25] porque não havia disponível nenhum modelo que simulassem o comportamento do MIPv4 e HMIPv4. Criar novos programas leva bastante tempo e seria necessário um código complexo para modelar um sistema que usasse esses protocolos. A opção por usar um modelo já pronto aponta ser a mais viável e segura, no entanto a simulação seria feito sobre IPv6.

As principais diferenças entre o IPv4 e o IPv6 foram levantadas para traçar um paralelo entre MIP e HMIP IPv4 e IPv6.

A primeira diferença, e seguramente a mais marcante, está no tamanho dos endereços. O IPv6 possui endereços formados com 128 bits o que totaliza 2^{128} endereços distintos enquanto que o IPv4 possui endereços de 32 bits o que totaliza 2^{32} endereços.

O cabeçanho do IPv6 também é distinto. O IPv6 define diversos tipos de extensões de cabeçalho (*extension headers)* que podem ser usadas para incluir informações adicionais logo após o cabeçalho de um pacote IPv6. As *extension headers* definidas são:

- Opção de destino (*Destination Option header)*
- Opção de salto-a-salto *(HopbyHop Options header)*
- Roteamento (*Routing header)*
- Autenticação *(Authentication header)*

O cabeçalho de opção de destino pode conter informações que apenas serão processadas no destino final.

Similarmente, o cabeçalho de opção de salto-a-salto pode ser incluído para carregar opções. Essas opções são processadas em cada roteador intermediário que recebe e direciona o pacote e no destino final.

O Cabeçalho de roteamento é particularmente útil para o MIPv6, e é similar ao *Source Router Options,* no entanto no MIPv6 o pacote não é processado a cada salto. Além disso, o nó no destino final que recebe o pacote não é obrigado a rotear pacotes à fonte, isso evita o tunelamento triangular onde pacotes enviados de um nó correspondente para o nó móvel devem passar primeiramente no agente nativo para depois ser encaminhado para o nó móvel. Isso é possível no MIPv4 usando-se otimização de roteamento.

O cabeçalho de autenticação permite o pacote incluir informações adicionais de autenticação.

No IPv4, cada opção de IP é tratada como uma opção salto-a-salto, causando assim uma diminuição na performance da rede por ter que ser processado a cada roteador intermediário, mesmo não pertencendo a esses roteadores.

A estrutura de cabeçalho do IPv6 é feita para diminuir o que se denomina *header overhead* colocando campos opcionais ou sem uso após o cabeçalhos IP.

A tabela abaixo resume as principais diferenças entre o IPv6 e o IPv4.

Tabela 4.1 - Diferenças básicas entre IPv6 e IPv4.

	IPv6	IPv4
Espaço para endereço	128bits	32bits
Segurança	Cabeçalhos para segurança	Não possui
Autoconfiguração	Padrão na versão	Não existe

Os mecanismos de registro com os agentes de mobilidade não mudam de uma versão pra outra. O IPv4 utiliza ARP e ICMP enquanto o IPv6 pode utilizar NDE (*Neighbour Discovery Protocol*) para que os nós de uma rede se identifiquem. Os protocolos MIP e HMIP em ambas as versões apresentam os mesmos mecanismos para a descoberta de agentes, para tratamento do handover e para criação de túneis IP-em-IP. Portanto, para a análise de latência de *handover* no MIPv4 e HMIPv4, os modelos MIPv6 e HMIPv6 são válidos.

Apesar dos agentes de mobilidade terem as mesmas funções em ambas as versões a nomenclatura

muda para alguns agentes inseridos no HMIP. O GFA, agente com maior nível de hierarquia no HMIPv4, é denominado MAP(*Mobility Anchor Point*). Os agentes estrangeiros de mais baixa hierarquia (*FA – Foreign Agent*) são denominados AR (*Access Router*) no IPv6.

Como os módulos são baseados em IPv6, as simulações estarão com a nomenclatura para IPv6, no entanto, será feita a analogia com IPv4.

4.2.4 - Simulação do MIP e do HMIP

Antes de prosseguir com os dados obtidos com a simulação, uma descrição é feita a respeito das topologias usadas, da configuração das entidades participantes da rede e das aplicações que estão sendo executadas nos hosts.

Para as duas simulações foram usados roteadores idênticos, espaçados com a mesma distância, interconectados com links de mesma velocidade e delay. Além da configuração dos roteadores ser idêntica, a velocidade que o nó móvel se movimenta e a área de cobertura dos Pontos de Acesso são as mesmas em ambas simulações. Isso acontece porque é desejável que a diferença de tempo de latência medida para cada caso seja decorrente apenas das diferenças peculiares entre os dois protocolos. Com isso é possível mostrar o aperfeiçoamento do MIP usando-se o HMIP.

As figuras a seguir mostram as topologias usadas para o MIP e HMIP respectivamente.

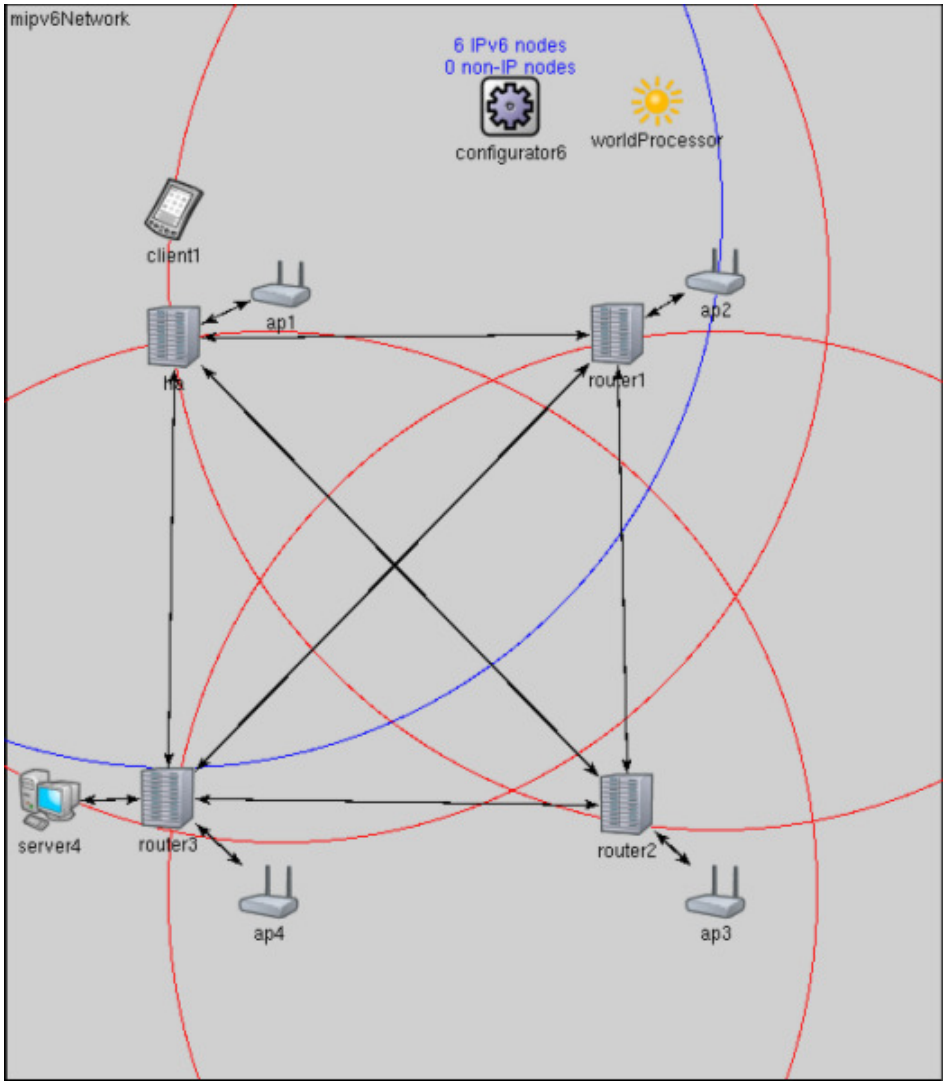


Figura 4.20 - Topologia usada para a simulação do protocolo MIP

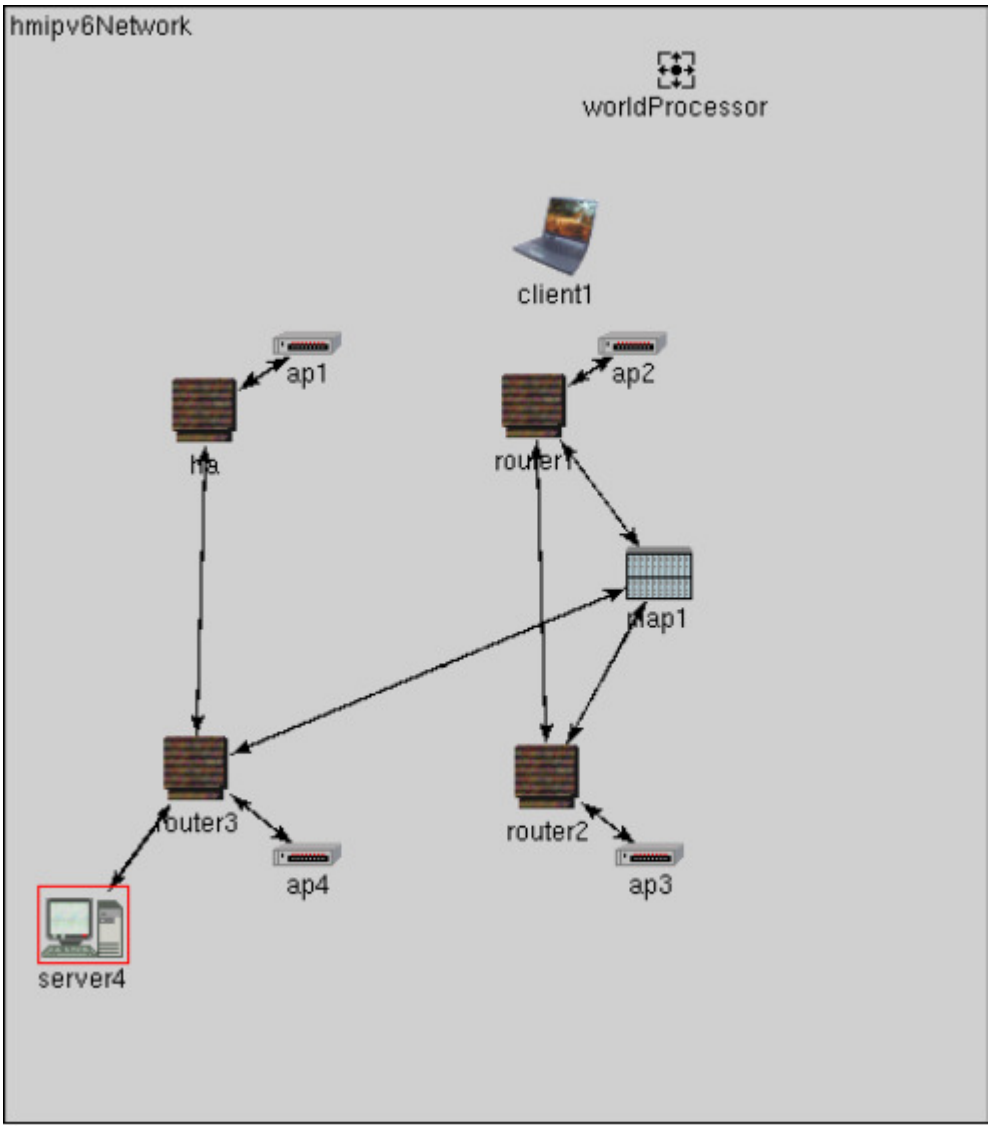


Figura 4.21 – Topologia utilizada para a simulação do protocolo HMIP

O roteamento nessa rede é feito utilizando-se o protocolo RIPv2 RFC 1723. Este é um protocolo baseado no roteamento vetor-distância. O RIPv usa contagem de saltos como métrica de roteamento.

A configuração dos pontos de acesso é a seguinte:

- Protocolo de transmissão (camada física): IEEE 802.11b
- Máxima potência de transmissão: 10dBm
- Limiar de potência(para Rx confiável): -96dBm
- Perdas de propagação: 2.6 dBm/m^2

O servidor 4 pode ser um servidor de aplicativo UDP ou TCP. Com relação aos protocolos MIP e HMIP este pode se comportar como um nó correspondente, no entanto, nesta simulação ele não está gerando tráfego na rede.

O nó móvel está descrito na topologia como "*client1*". Ele usa o modelo *WirelessHost*. Possui uma interface wireless 802.11b no modo Infra-estrutura.

O OMNeT++ permite que se faça modelos dos canais. O canal entre roteadores foi modelado da seguinte maneira:

```
channel internetCable
      delay 0.012
      datarate 10e9
endchannel
```

Como podemos observar desse trecho de código, o atraso associado a esse link é de 12ms e a taxa de transmissão é de 10Gbps.

Para as conexões entre os roteadores e nós de sua rede é usado o canal "*intranetCable*". A taxa é de 100Mbps (Fast Ethernet) com delay nulo.

Comparando as figuras acima vemos que a topologia para o HMIP possui um nó distinto denominado

MAP (*Mobility Anchor Point*). Como descrito anteriormente, esse nó é equivalente em funcionalidade ao GFA do HMIPv4. Os roteadores que pertencem ao domínio do MAP são *router1* e *router2*. O roteador *router3* pertence a outro domínio.

O movimento do nó móvel segue o percurso ha-router1-router2-router3.

Nesse percurso existem três *handovers*. Do ha para o *router1*, do *router1* para o *router2* e do *router2* para o *router3*. A figura 4 mostra o tempo de latência para cada *handover*.

Como o MIP gerencia apenas a macromobilidade do nó móvel, todos os registros foram feitos com o agente nativo. A variação desses registros é causada pelas diferentes rotas que o tráfego de mensagens de registro seguem até chegar ao HA.

No caso do HMIP vemos uma queda na latência de *handover* no segundo registro. Voltando ao percurso que o nó móvel faz, vemos que nos *handovers* de ha para *router1* e de *router2* para *router3* o nó móvel precisa registrar-se com seu agente nativo, pois é nessas transições que o nó móvel está passando de um domínio para o outro.

No caso do *handover* de *router1* para o *router2* o tempo de latência do HMIP é menor que o tempo de latência do MIP. Como descrito anteriormente, os roteadores *router1* e *router2* fazem parte do domínio do MAP, em outras palavras, estes nós estão disponibilizados de forma hierárquica. Portanto, quando o nó se move da área de cobertura do Ap2 para o Ap3

ele realiza um registro regional com o roteador MAP. Durante o *handover*, o túnel entre o ha e o MAP se mantém e é feito um novo *binding* entre o MPA(GFA) e o nó móvel.

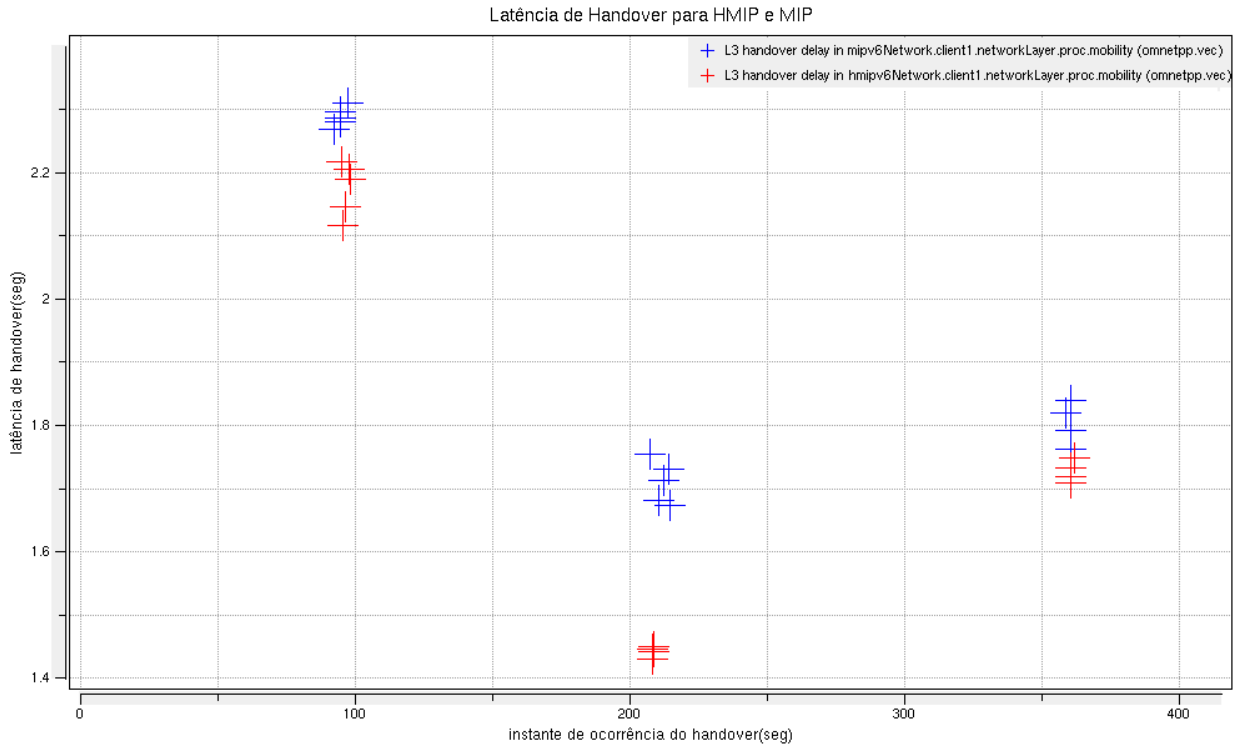


Figura 4.22 - Latência de handover para o MIP(em azul) e para o HMIP(em vermelho).

Para verificar o desempenho de cada protocolo, foi iniciado um ping a partir do nó móvel com destino ao agente nativo(ha). A figura abaixo mostra o RTT(*Round Trip Delay*) dos pings obtidos.

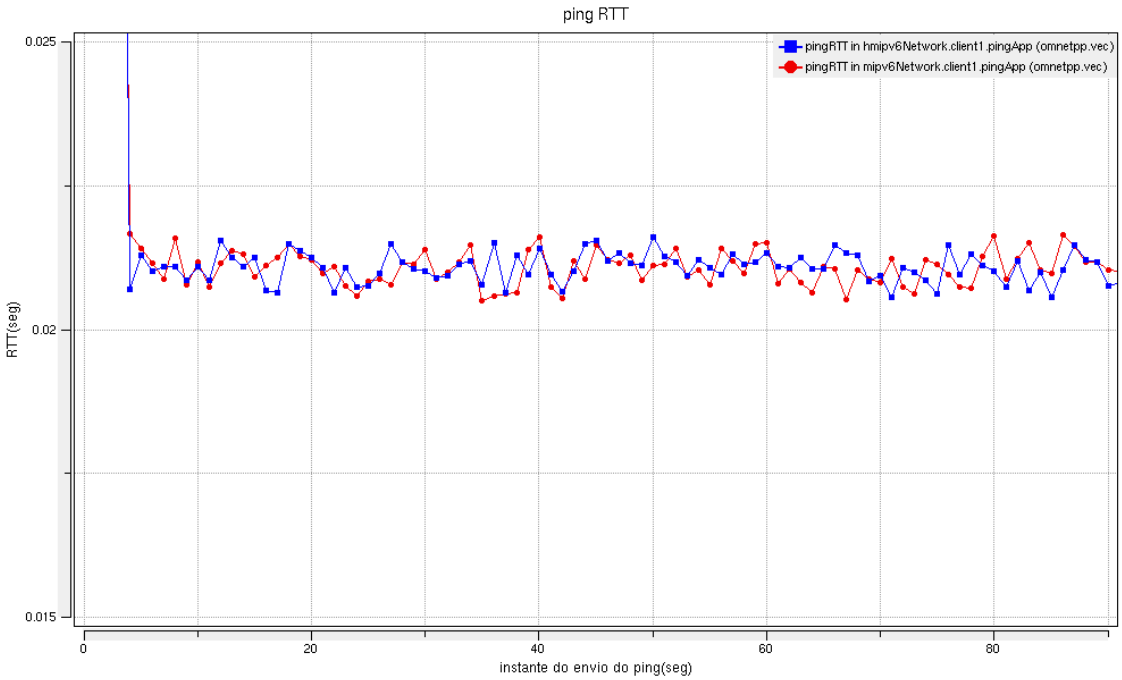


Figura 4.23 – Ping RTT na simulação do MIP e do HMIP.

O gráfico em azul mostra o RTT para o HMIP enquanto que o vermelho mostra o RTT para o MIP.

Durante o intervalo de observação do gráfico, o túnel entre os nós móveis está ativo pois há resposta de todos os "pings". Vemos que o tempo de RTT é praticamente mesmo entre os protocolos. Isso acontece porque, uma vez que o registro é feito o túnel IP-em-IP entre o nó móvel e o HA, nenhum registro precisa ser feito até que o tempo de vida do túnel expire ou o nó móvel migre para a região administrativa de outro roteador. Vemos com a figura 3 que a utilização de registros regionais não afeta no desempenho da rede com relação a velocidade do tráfego ou a desempenho de roteamento.

Com esse resultado vemos o aperfeiçoamento que o HMIP traz ao MIP com a adição de gerencia de micromobilidade. Latência de *handover* menor é crucial para aplicações de tempo real. Essas aplicações requerem garantias rigorosas de que vão chegar a tempo em seu destino, anulando a perda de pacotes e reenvio de informação. Serviços de multimídia necessitam de requisitos de qualidade de serviço(QoS) e são especialmente sensíveis a latência. Portanto, a aplicação de registros regionais melhoram a QoS das redes por onde esse tipo de informação trafega.

Um menor atraso dos pacotes enviados entre os agentes pode ser obtido integrando-se o HMIP com formas de roteamento que apresentem maior desempenho. Neste trabalho discutiu-se as redes MPLS e foi mostrado que essas redes mostram um aumento de desempenho com relação a uma rede IP convencional. A próxima seção consiste na

simulação de uma rede MPLS e uma rede OSPF. Serão comparados alguns parâmetros como RTT de "pings" entre dois *hosts* para análise de desempenho de cada uma.

4.2.5 - Simulação MPLS e OSPF

Para verificar que o MPLS apresenta um desempenho melhor para ambientes com MIP ou HMIP se fez necessário uma comparação com algum protocolo de roteamento IP. Para este trabalho foi utilizado o OSPF(*Open Shortest Path First*). Este é um tipo de roteamento *link-state*. Neste esquema as rotas são criadas baseadas no menor esforço e não na quantidade de saltos(*hops*). Como diz seu nome, a tabela de roteamento é construída com o base no menor caminho(*SPF- Shortest Path First*). Os roteadores trocam informações denominadas *Link State Advertisements*, esses anúncios carregam informações como métrica e outras varáveis. Os roteadores vão acumulando essas informações e através delas montam a topologia da rede em forma de árvore. O algoritmo que calcula o menor caminho é denominado Algoritmo de Dijkstra.

Como foi dito anteriormente, diversos modelos de protocolos já foram desenvolvidos para o OMNET++ e estão disponíveis como *source-free* para utilização. Para esta simulação foi utilizado o modelo MPLS usando o protocolo de sinalização RSVP-TE. O protocolo RSVP-TE (*Resource Reservation Protocol with Traffic Engineering*) é um protocolo da camada de transporte e é utilizado para suportar engenharia de tráfego em ambientes MPLS.

O modelo para roteamento OSPF também estava disponível. O modelo usa OSPFv3 definida na RFC 2740 no qual suporta tanto IPv6 quanto IPv4.

Para as duas simulações, o arranjo físico dos roteadores é o mesmo, ou seja, as características do enlace físico e a distância entre os roteadores nas duas topologias são equivalentes. O que difere de uma para outra é obviamente o protocolo de roteamento que os roteadores utilizam.

Os links entre os roteadores possuem uma velocidade de 10Mbps e delay de 2ms até 5ms. Os hosts e os roteadores estão conectados com links de 2.5Mbps com delays de 0.2ms.

As figuras a seguir mostram as topologias utilizadas para a simulação da rede OSPF e MPLS respectivamente.

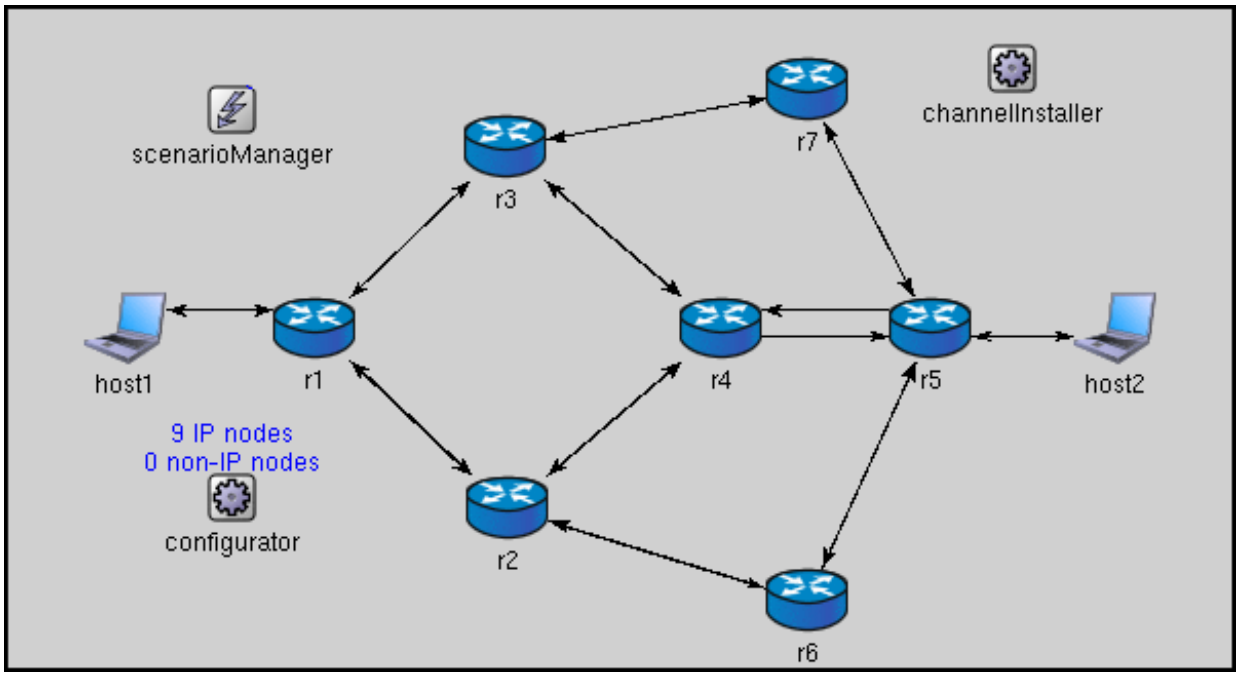


Figura 4.24 - Topologia usada na simulação da rede OSPF

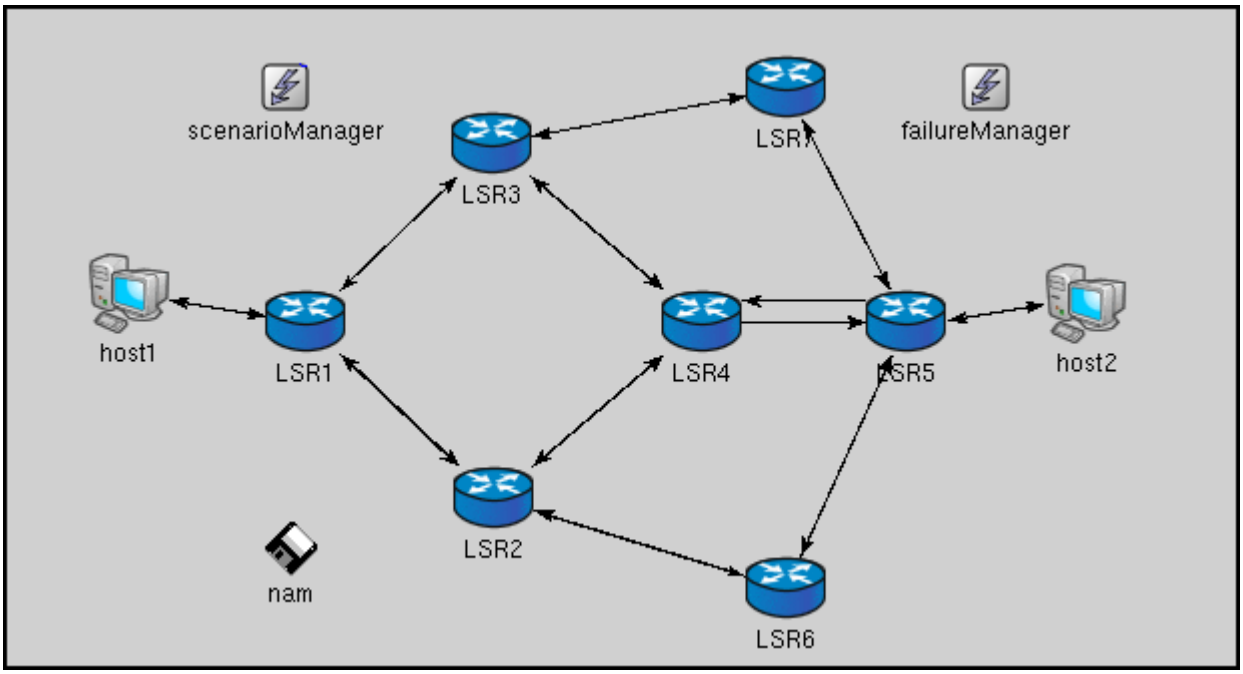


Figura 4.25 - Topologia usada na simulação da rede MPLS

Para avaliar a capacidade do MPLS em aumentar a eficiência com relação a latência foi plotado o gráfico do RTT(*Round Trip Time*) dos "pings" enviados do *host1* em direção ao *host2* em ambas as simulações. As simulações duraram 500seg e a taxa de "pings" enviados era de 0.1 ping/seg. A figura 6 mostra que o tempo de retorno dos "pings" na rede MPLS (vermelho) é inferior ao OSPF(azul).

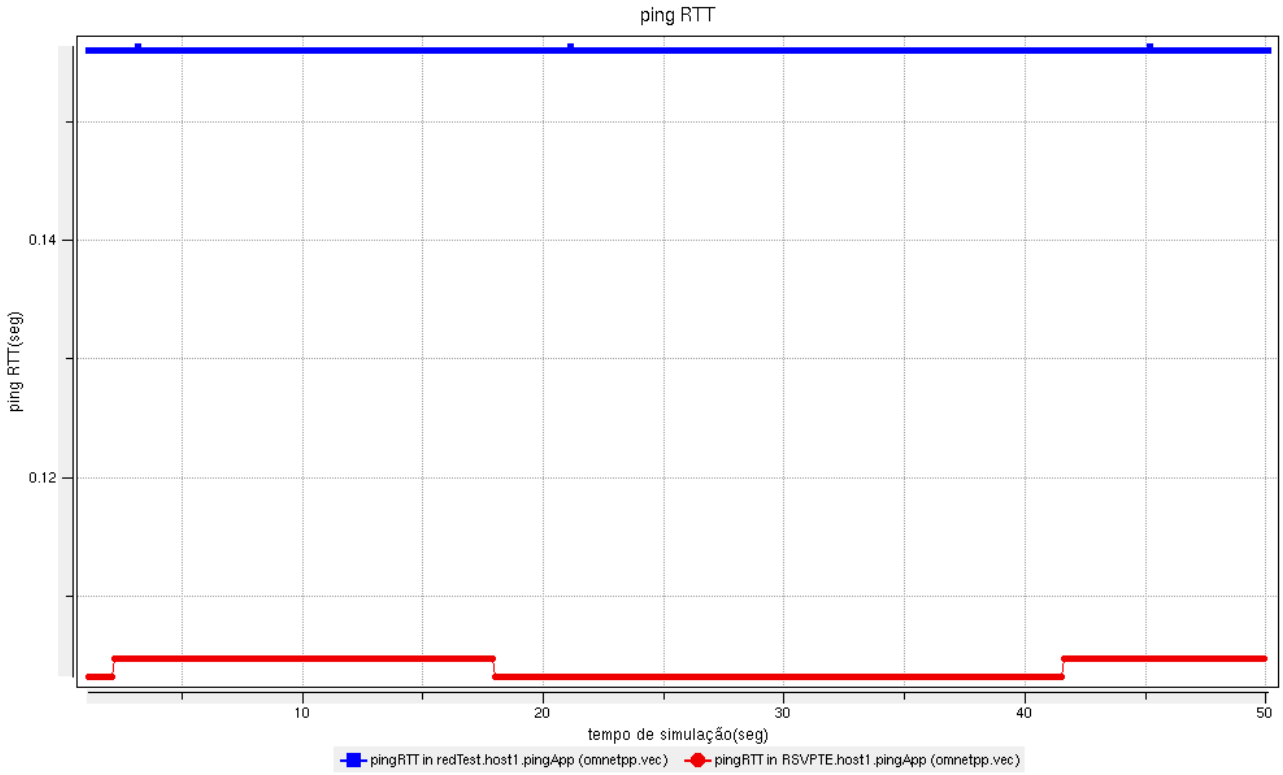


Figura 4.26 - Ping RTT para rede MPLS(vermelho) e para rede OSPF(azul).

A tabela a seguir mostra resultados estatísticos do envio de pings pelo host1.

Tabela 4.2 – Estatísticas de ping RTT.

	OSPF	MPLS
Qtd de pings enviados	490	490
Pacotes pedidos(%)	0	0
RTT mínimo	156,036 ms	103,259ms
RTT máximo	156,287ms	104,707ms
RTT médio	156,038ms	103,976ms
Desvio padrão	0,0195735ms	0.725104ms
Variância	3.83123e-10ms	5.25776e-07ms

Outro parâmetro utilizado na comparação dos dois protocolos foi o tamanho da fila de pacotes em um dos roteadores. O algoritmo utilizado é o Tail Drop. Esse algoritmo é bastante simples pois não diferencia o tráfego. Quando o tamanho da fila alcança sua capacidade máxima os pacotes que estão chegando vão sendo descartados até que a fila

tenha um tamanho suficiente para comportar os pacotes que estão chegando. O roteador tomado como referência é o roteador no qual o host1 está conectado. Para a topologia OSPF seria o roteador "r1" e para a topologia MPLS seria o "LSR1". A medida do tamanho da fila foi medida na interface ponto a ponto voltada ao host1.

O tamanho máximo para o buffer foi configurado para 10 pacotes de 56Kb. Lembrando que o payload dos pacotes ping tinham esse tamanho. As figuras 6 e 7 mostram os resultados obtidos para o OSPF e para o MPLS respectivamente.

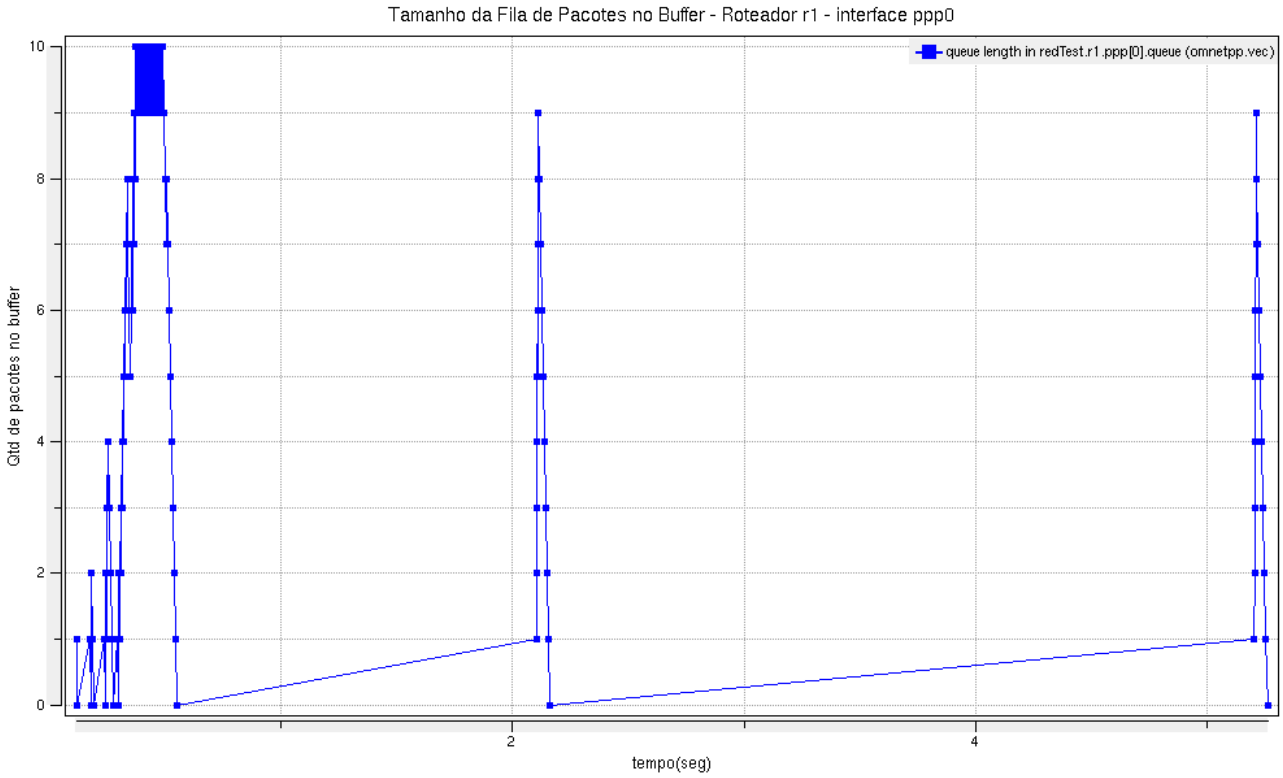


Figura 4.27 - Tamanho da fila de Pacotes no roteador r1 (OSPF).

Verifica-se desta figura que o buffer do roteador chega ao limite máximo no início da simulação e tende a ficar próximo de 10 na rede OSPF.

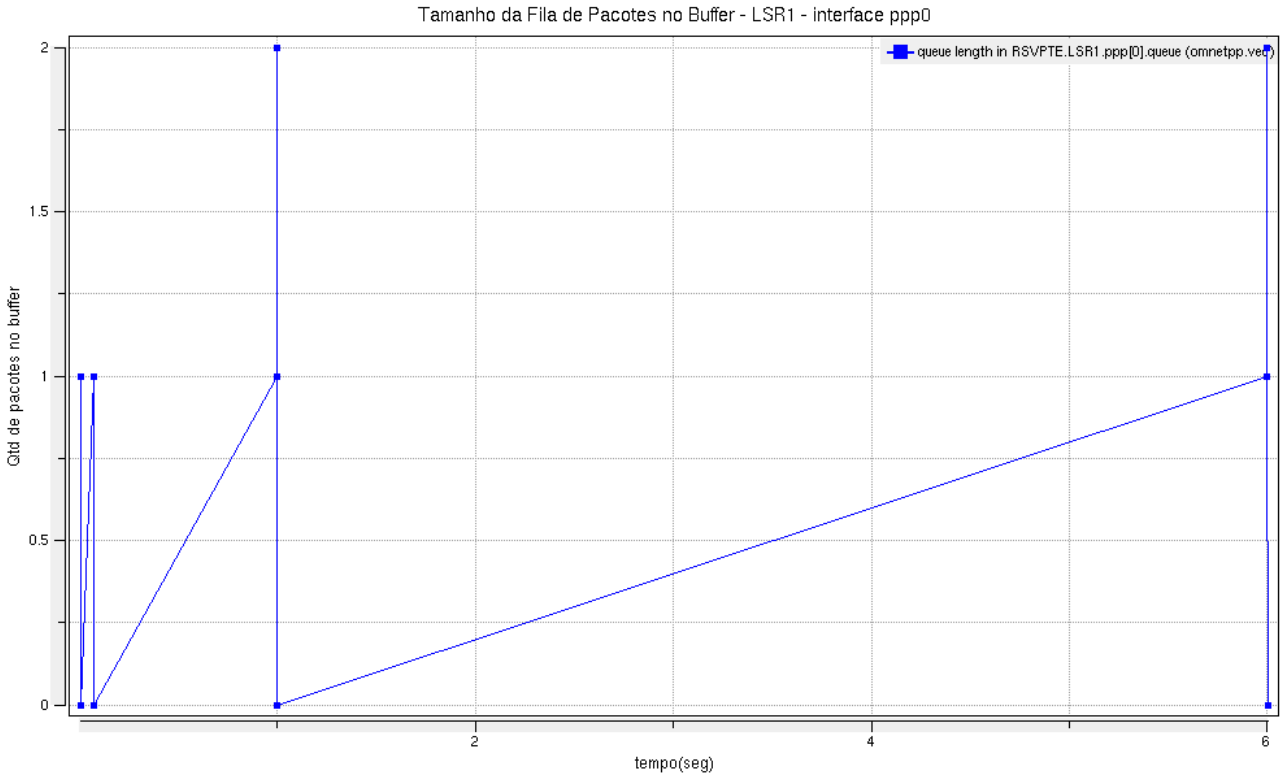


Figura 4.28 - Tamanho da fila de Pacotes no roteador LSR1 (MPLS).

A partir dessa figura é possível ver que o tamanho da fila para o caso do MPLS é bastante inferior ao OSPF e em nenhum instante o máximo da fila é alcançado. Portanto, com MPLS a probabilidade de ocorrer perda de pacotes é reduzida. Deve-se levar em conta que o esforço computacional é menor no encaminhamento de pacotes no MPLS. Isso ocorre porque quando os roteadores possuem a base de informações de rótulo não é mais necessário analisar os 20 bytes do cabeçalho IP, somente o rótulo que é constituído de 4 bytes. Por outro lado, grupos de pacotes que compartilham as mesmas requisições de encaminhamento recebem uma designação de classe de envio equivalente ou FEC (*Forwarding Equivalence Class*). Pacotes com a mesma FEC recebem os mesmos rótulos e são encaminhados de forma equivalente dentro da rede. Dessa maneira os roteadores só precisam encaminhar os pacotes com base em seu rótulo diferentemente do roteamento IP onde é necessário grande esforço de processamento para encaminhar um pacote pois é necessário verificar toda a tabela de roteamento que chega a ter um tamanho considerável . Com os resultados obtidos com o *host1* gerando pacotes “ping” verifica-se também que esse esquema de encaminhamento se mostra mais eficiente que com o roteamento feito em uma rede IP utilizando OSPF. Esse acréscimo no desempenho da comutação dos pacotes é particularmente desejável em aplicações em tempo real.

No âmbito do Mobile IP, o núcleo MPLS pode ser usado para diminuir a latência no envio de pacotes. Por exemplo, pode ser usado o protocolo RSVP que permite reservar recursos para certos tipos de tráfego e promover um serviço diferenciado ao Mobile IP. No entanto é importante ressaltar que, no caso do HMIP, o agente nativo e o GFA devem ser roteadores de borda (LER – Label Edge Routers) pois dessa forma é possível manter um LSP entre esses agentes de mobilidade.

5 - CONCLUSÃO

Neste trabalho foram apresentadas tecnologias que dão suporte ao mecanismo de mobilidade na Internet, o Mobile IP. Pesquisas realizadas nesta área se fazem necessárias pelo grande crescimento das redes sem fio e da necessidade de mobilidade rápida, fácil e descomplicada.

Como foi mostrado, inicialmente em teoria com o detalhamento da RFC 4857 [24] e depois, na prática, o Registro Regional ou Mobile IP Hierárquico demonstra ser uma boa alternativa ao problema do "atraso de *handoff*" ocasionado pela movimentação de um MN de uma sub-rede para outra no domínio da rede estrangeira.

Neste trabalho foi analisado e implementado o aprimoramento Mobile IP Hierárquico que traz para o Mobile IP um gerenciamento para micromobilidade, ou seja, ele implementa uma forma de minimizar a latência de handoff quando o nó móvel está se movendo dentro de um mesmo domínio. Sendo assim, o Registro Regional contribui para o grande propósito do MIP, evitar que a conexão seja perdida por muito tempo no evento da movimentação.

Além disso, foi mostrado por meio de simulações e análises estatísticas que o MPLS demonstra ser uma boa alternativa ao fato de que, no MIP, todos os pacotes que se destinam ao nó móvel têm de passar pelo agente nativo. Portanto, para o problema de latência de envio dos pacotes ao nó móvel, foi proposto nesse trabalho a integração dos protocolos de mobilidade com o protocolo de alto desempenho de roteamento MPLS. Foi comprovado que a utilização do roteamento MPLS, ao invés do tunelamento IP-em-IP, traz ganho no desempenho do sistema no que diz respeito envio e recebimento de pacotes. Portanto, a proposta é que, ao invés de se utilizar túneis IP-em-IP baseados no roteamento IP convencional, sejam utilizados túneis LSP, os quais apresentam maior desempenho na comutação de pacotes que trafegam na rede.

Fazendo-se uma análise geral de todo estudo teórico e prático feito neste trabalho, os resultados das propostas de aperfeiçoamento do Mobile IP foram de grande valia, pois conseguiu-se traçar um paralelo entre o desempenho no *handoff* utilizando o HMIP e o desempenho no roteamento de pacotes utilizando MPLS. Portanto, foi demonstrado neste estudo que essa integração mostrou-se bastante promissora no âmbito de simulações e implementações físicas em pequena escala, podendo ser transportada para o uso redes de complexidade maior.

Para trabalhos futuros, uma boa sugestão é a implementação dos protocolos HMIP e MPLS trabalhando juntos, de forma que os roteadores de borda da rede MPLS sejam o agente nativo e o gateway do agente estrangeiro. Desta forma, ao invés de usar o túnel IP-em-IP entre os agentes, seria usado um túnel MPLS.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] Alwayn, V.,(2002). “Advanced MPLS Design and Implementation” , Cisco Press, IN EUA.

[2] András Varga. OMNeT++ - Portable Simulation Environment in C++. Technical University of Budapest, 2006. In Hungarian.

[3] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V. e Swallow, G. (2001). “*RSVP-TE: Extensions to RSVP for LSP tunnels*.” RFC 3209, 61p.

[4] Björn, A., Forsberg, D., Hautio, J., Malinen, J.K., Mustonen, K., Weckström, T., Malinen, J. e Kari, H. “Dynamics Mobile IP.” Acessado em novembro de 2007 no endereço http://dynamics.sourceforge.net/

[5] Braden, R., Zhang, L., Berson, S., Herzog, S. e Jamin, S. (1997). “*Resource ReSerVation Protocol (RSVP).*” RFC 2205, ISI, UCLA, IBM Research e Universidade de Michigan, 76p.

[6] Calhoun, P.R. e Perkins, C.E. (2001). “*Diameter Mobile IP extensions.*” IETF-Draft, Sun Laboratories e Nokia Research Center, 32p.

[7] Chaskar, H. e Koodli, R. (2001). “*A Framework for QoS Support in Mobile IPv6*.” IETF-Draft, Nokia Research Center, 11p.

[8] Chen W.T. e Huang, L.C. (2000). “*RSVP Mobility Support: A Signaling Protocol for Integrated Services Internet with Mobile Hosts*.” Em: *Proceedings IEEE INFOCOM 2000*, Tel Aviv, Israel, 1283-1292.

[9] Choi, J.K. (2003). “*Draft new Recommendation Y.MIPoMPLS (Mobile IP Services over MPLS).*” *International Telecommunication Union ITU, Telecommunication Standardization Sector*, 38p.

[10] Choi, J.K., Kim, M.H. e Lee, Y.J. (2001). “*Mobile IPv6 support in MPLS Network.*” IETF-Draft, ICU e ETRI, 12p.

[11] Choi, J.K., Um, T.W., Lee, Y.K. e Yang, S.H. (2001). “*Extension of LDP for Mobile IP Service through the MPS Network.*” IETF-Draft, ICU e ETRI, 28p.

[12] Deering, E. (1991). “*ICMP Router Discovery Messages*.“ RFC 1256, Xerox PARC, 19p.

[13] E. Rosen, A. Viswanathan, e R. Callon. “Multiprotocol Label Switching Architecture.”, RFC 3031, jan. 2001.

[14] Jamoussi, B., Andersson, L., Callon, R., Dantu, R., Wu, L., Doolan, P., Worster, T., Feldman, N., Fredette, A., Girish, M., Gray, E., Heinanen, J., Kilty, T. e Malis, A. (2002). "*Constraint-based LSP setup using LDP*." RFC 3212, 42p.

[15] L. Anderson, P. Doolan, N. Feldman, A. Fredette, B. Thomas. "LDP Specification". RFC 3036, jan. 2001.

[16] Leu, J.R. "*SourceForge.net: MPLS for Linux.*" Acessado em fevereiro de 2006 no endereço http://sourceforge.net/projects/mpls-linux/.

[17] Malki, K.E. (2005). "*Low latency handoffs in Mobile IPv4.*" IETF-Draft, Athonet, 56p

[18] Moy, J. (1998). "*OSPF version 2*." RFC 2328, Proteon Inc., 189p.

[19] Oliveira, A. H. C. de, (2006). "*Gerenciamento de túneis em ambiente MPLS integrado a Mobile IP.*" Dissertação de mestrado, Universidade de Brasília, Brasil, 136p.

[20] Palmieri, F. (2005). "*An MPLS-based architecture for scalable QoS and traffic engineering in converged multiservice mobile IP networks*." Em: *Computer Networks*, 47, 257-269.

[21] Perkins, C. (1996). "*IP Encapsulation within IP*." RFC 2003, IBM, 14p.

[22] Perkins, C. (2002). "*IP Mobility Support for IPv4*." RFC 3344, Nokia Research Center, 99p.

[23] Perkins, C. e Johnson, D.B. (2001). "*Route Optimization in Mobile IP.*" IETF-Draft, Nokia Research Center e Carnegie Mellon University, 25p.

[24] Perkins, C., Fogelstroem,E. e Jonsson, A. (2007). "Mobile Ipv4 Regional Registration" RFC 4857, Nokia Siemens Networks, 32p.

[25] R. Braden(ed.), L. Zhang, S. Berson, S. Herzong e S. Jamin. "Resource Reservation Protocol(RSVP)-Vertion 1 Functional Especification", RFC 2205, set. 1997.

[26] S. Deering e R. Hiden. "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, dez. 1998.

[27] Rosen, E., Viswanathan, A. e Callon, R. (2001). "*Multiprotocol label switching*

[28] Xie, K., Wong, V.W.S. e Leung, V.C.M. (2003). "*Support of micro-mobility in MPLS-based wireless access networks.*" Em: *WCNC 2003 - IEEE Wireless Communications and Networking Conference*, New Orleans, E.U.A., 4, 1, 1242-1247.