



Instituto Politécnico de Leiria
Escola Superior de Tecnologia e Gestão

IPv6@ESTG-Leiria

Mobilidade IPv6 - Estudo das variantes de handover

Nélia Carina Gomes Grácio
Nelson de Jesus Sebastião

Leiria
Fevereiro de 2007



Instituto Politécnico de Leiria
Escola Superior de Tecnologia e Gestão

IPv6@ESTG-Leiria

Mobilidade IPv6 - Estudo das variantes de handover

Relatório final da disciplina de Projecto II, do curso de Licenciatura em
Engenharia Informática e Comunicações

Realizado entre Outubro de 2006 e Fevereiro de 2007

Autores

Nélia Carina Gomes Grácio – 9628

Nelson de Jesus Sebastião – 9053

Orientadores

Mário Antunes

Nuno Veiga

Leiria

Fevereiro de 2007

Agradecimentos

Os agradecimentos são endereçados especialmente às nossas famílias, por todo o apoio que nos deram ao longo de todos estes anos, mas em especial neste último, durante a realização deste projecto.

Agradecemos também aos nossos orientadores, professor Mário Antunes e professor Nuno Veiga, pela exigência e rigor que sempre nos inculcaram ao longo deste projecto, concedendo-nos também a liberdade de poder tomar decisões.

Aos nossos colegas, antecessores nos projectos IPv6@ESTG-Leiria, em especial ao Tiago Amado, antecessor no Projecto IPv6@ESTG-Leiria Mobilidade em IPv6, agradecemos a base de trabalho que nos proporcionaram e a ajuda que nos deram sempre que foi necessário.

Ao nosso colega Rui Silva, pela grande ajuda com o Linux, pois sem ele alguns dos nossos problemas não tinham sido resolvidos.

Por últimos agradecemos a todos aqueles com quem trocámos e-mails para esclarecimento de dúvidas.

A todos aqueles que directa ou indirectamente nos ajudaram, o nosso muito obrigado.

Um distinto obrigado à minha mãe, Esmeraldina, que sempre me apoiou e deu a força necessária para conseguisse chegar onde cheguei.

Ao meu colega de projecto Nelson Sebastião que sempre se mostrou dedicado e pelo alento prestado nos momentos mais difíceis da realização deste projecto.

Nélia Grácio

Um especial obrigado à Dra. Marta Gomes, Nuno Gomes e colegas de casa Rui Silva e Bruno Cristóvão pela força dada nos momentos mais difíceis do projecto.

À minha colega de projecto Nélia Grácio pela compreensão e motivação prestada durante as diversas etapas do projecto.

Nelson Sebastião

Resumo

A Mobilidade IP (MIP) permite que um utilizador se possa movimentar em redes IPv4 ou IPv6, sem perder a conectividade. Contudo, este projecto baseou-se apenas em mobilidade IPv6 e as suas variantes de handover. Este novo protocolo introduz três novos conceitos, o *Mobile Node* (MN), o *Home Agent* (HA) e o *Correspondent Node* (CN). Entre estes são trocadas mensagens, estabelecidas pelo protocolo, que são enviadas quando se verifica a deslocação do terminal móvel entre redes IPv6.

A acção de deslocação entre redes tem um processo base implícito, o *handover*. Este processo ocorre durante a transição entre redes e conduz, normalmente, à perda de vários pacotes. Este facto afecta as comunicações durante a transição, com especial ênfase nas comunicações em tempo real. Existem, no entanto, alguns mecanismos que podem acelerar este processo, minimizando o número de pacotes perdidos. Surge então um novo conceito de *handover* que reduz o tempo de duração do processo, o *Fast Handover for Mobile IPv6* (FMIPv6).

Com o objectivo de testar estas tecnologias, definiu-se um cenário geral de testes com suporte para ambos os mecanismos de *handover*. Os testes realizados comprovaram a eficácia do FMIPv6, que possui um *handover* claramente mais rápido, apresentando uma perda de pacotes mínima, quando comparada com o MIPv6.

Palavra-Chave: *Mobilidade IP; Mobilidade IPv6; MIPv6; Fast Mobility IPv6; FMIPv6; Handover; Handoff*

Abstract

IP Mobility (MIP) permits that a user can move between networks IPv4 or IPv6 without losing connectivity to the Internet. This project focuses mainly on IPv6 mobility and some of its handover variants. This new protocol introduces three new terms Mobile Node (MN), Home Agent (HA) and Correspondent Node (CN). Specified mobility messages are transferred between these three elements when the mobile node moves between IPv6 networks.

During the transition process also known as handover, packets are lost due to specific protocol issues. This affects communication during transition periods in particular real-time communications. To minimize packet loss some mechanisms were developed to accelerate the handover process. One of these mechanisms is Fast Handover for Mobile IPv6 (FMIPv6).

The objective of this project is to test these new technologies. A general scenario will be developed that supports both of these protocols MIPv6 and FMIPv6. Our tests will prove that FMIPv6 is more efficient, it reduces the handover time and this represents a very low packet loss in comparison with traditional MIPv6.

Key Word: *Mobile IP; Mobile IPv6; MIPv6; Fast Mobile IPv6; FMIPv6; Handover; Handoff*

Índice

<i>Acrónimos e Abreviaturas</i>	<i>xi</i>
<i>Lista de Figuras.....</i>	<i>xiii</i>
<i>Lista de Tabelas.....</i>	<i>xv</i>
1 Introdução	1
1.1 Objectivos	1
1.2 Estrutura do relatório.....	2
2 Mobilidade IP	5
2.1 Terminologia.....	6
2.2 Protocolo MIP	7
2.2.1 Principais intervenientes do MIP	9
2.2.2 Fases do MIP	9
3 Handover	11
3.1 Tipos de handover	12
3.1.1 Handovers entre diferentes tecnologias	12
3.1.1.1 Handover horizontal.....	13
3.1.1.2 Handover vertical	14
3.1.2 Handovers por áreas de abrangência.....	15
3.1.2.1 Macro-mobilidade	16
3.1.2.2 Micro-mobilidade.....	17
3.2 Decisão de Handover	18
3.2.1 Handover reactivo.....	18
3.2.2 Handover proactivo	19
4 MIPv6.....	21
4.1 Passos do processo de handover no MIPv6.....	23
4.1.1 Detecção de movimento	23
4.1.2 Descoberta de router vizinho	24

4.1.3	Configuração do Care-of-Address	25
4.1.4	Detecção de duplicação de endereços	25
4.1.5	Authentication, Authorization and Accounting e Qualidade de Serviço	26
4.1.5.1	AAA	26
4.1.5.2	QoS	26
4.1.6	Registar novo Care-of-Address	27
4.1.7	Fim do processo de handover	28
4.2	Latência do MIPv6.....	28
5	<i>Fast Handover</i>	31
5.1	Descrição	31
5.2	Troca de mensagens	32
5.3	Modos de operação.....	33
5.3.1	Modo predictivo	33
5.3.2	Modo reactivo.....	34
5.3.3	Comparação entre o modo preditivo e o modo reactivo	36
6	<i>Cenário de Teste</i>	37
6.1	Cenário Geral.....	37
6.2	Endereçamento.....	38
6.3	Hardware	38
6.4	Divisão de cenários.....	39
6.4.1	Cenário MIPv6	39
6.4.2	Cenário FMIPv6	40
6.5	Software	41
6.5.1	Sistemas operativos	41
6.5.1.1	Sistemas operativos comerciais.....	42
6.5.1.2	Sistemas operativos opensource.....	42
6.5.2	Implementações de mobilidade	43
6.5.2.1	MIPL.....	43
6.5.2.2	FMIPv6	43

6.5.3	Wireshark - Analisador de rede	44
6.5.4	Virtualização.....	45
6.5.5	Aplicações para testar a mobilidade	45
6.5.5.1	MIPv6 Tester.....	45
6.5.5.2	MIPv6 Analyzer	46
6.6	Objectivos dos Cenários	47
7	Testes	49
7.1	Cenário MIPv6	49
7.1.1	Debug do daemon mip6d.....	50
7.1.2	Tráfego icmp.....	52
7.1.3	Captura de pacotes	52
7.1.4	Handover	55
7.1.5	Resultados obtidos	58
7.1.6	Problemas detectados.....	59
7.2	Cenário FMIPv6.....	59
7.2.1	Debug do daemon fmipv6	59
7.2.2	Tráfego icmp.....	61
7.2.3	Captura de pacotes	62
7.2.4	Handover	65
7.2.5	Resultados obtidos	68
7.2.6	Problemas detectados.....	68
7.3	Comparação entre MIPv6 e FMIPv6	69
8	Conclusões.....	71
8.1	Objectivos realizados	71
8.2	Limitações	72
8.3	Trabalho futuro.....	73
8.3.1	Hierarchical Mobile IPv6 (HMIPv6) Handover	73
8.3.2	Heterogeneidade IPv4/IPv6	74
	Bibliografia	75

<i>Anexo A</i>	<i>Ficheiros de configuração.....</i>	<i>83</i>
<i>Anexo B</i>	<i>Planeamento do projecto</i>	<i>99</i>

Acrónimos e Abreviaturas

AAA	Authentication, Authorization and Accounting
AP	Access Point
API	Application Programming Interface
AR	Access Router
BSS	Basic Service Set
CN	Correspondent Node
CoA	Care-of-Address
DAD	Duplicate Address Detection
ESS	Extended Service Set
FHMIPv6	Fast Hierarchical Mobile IPv6
FMIPv6	Fast Handover for Mobile IPv6
HA	Home Agent
HMIPv6	Hierarchical Mobile IPv6
HoA	Home Address
IETF	Internet Engineering Task Force
IOS	Internetwork Operating System
ISATAP	Intra-Site Automatic Tunnel Addressing Protocol
ISP	Internet Service Provider
LAN	Local Area Network
MAP	Mobility Anchor Point
MIP	Mobile IP
MIPL	Mobile IPv6 for Linux
MIPv6	Mobile IPv6

MN	Mobile Node
NAR	New Access Router
NCoA	New Care-of-Address
PAR	Previous Access Router
PCoA	Previous Care-of-Address
QoS	Qualidade de Serviço
RADVD	Router Advertisement Daemon
RSVP	Resource ReSerVation Protocol
RTT	Round Trip Time
TCP/IP	Transmission Control Protocol/Internet Protocol
USAGI	UniverSAI playGround for Ipv6
VOIP	Voice over IP
WLAN	Wireless Local Area Network

Lista de Figuras

<i>Figura 2-1 Mobilidade possibilitada pelo protocolo IP.....</i>	<i>5</i>
<i>Figura 2-2 – Arquitectura MIP</i>	<i>6</i>
<i>Figura 2-3 – Encaminhamento triangular.....</i>	<i>8</i>
<i>Figura 3-1 – Handover horizontal</i>	<i>12</i>
<i>Figura 3-2 – Handover vertical.....</i>	<i>13</i>
<i>Figura 3-3 – Vertical Handover Inter-Technology</i>	<i>15</i>
<i>Figura 3-4 – Micro-Mobilidade Macro-Mobilidade</i>	<i>16</i>
<i>Figura 3-5 – Handover reactivo.....</i>	<i>18</i>
<i>Figura 3-6 – Handover proactivo.....</i>	<i>19</i>
<i>Figura 4-1 – Etapas do handover em MIPv6</i>	<i>22</i>
<i>Figura 4-2 – Processo de handover MIPv6.....</i>	<i>23</i>
<i>Figura 5-1 – Fast Handover IPv6 Handover Protocol</i>	<i>32</i>
<i>Figura 5-2 –Troca de mensagens no modo predictivo.....</i>	<i>34</i>
<i>Figura 5-3 – Troca de mensagens no modo reactivo</i>	<i>35</i>
<i>Figura 5-4 –Modo predictivo versus modo reactivo</i>	<i>36</i>
<i>Figura 6-1 – Cenário geral</i>	<i>37</i>
<i>Figura 6-2 – Cenário MIPv6.....</i>	<i>40</i>
<i>Figura 6-3 – Cenário FMIPv6</i>	<i>41</i>
<i>Figura 6-4 – MIPv6 Tester.....</i>	<i>46</i>
<i>Figura 6-5 – MIPv6 Analyzer.....</i>	<i>47</i>
<i>Figura 7-1 – Cenário de teste MIPv6.....</i>	<i>50</i>
<i>Figura 7-2 – Binding Update enviado pelo terminal móvel ao Home Agent</i>	<i>51</i>
<i>Figura 7-3 – Binding Update recebido no Home Agent.....</i>	<i>51</i>
<i>Figura 7-4 – Binding Update enviado pelo terminal móvel ao nó correspondente</i>	<i>51</i>
<i>Figura 7-5 – Eliminação do túnel criado entre o Home-Agent e o terminal móvel</i>	<i>52</i>
<i>Figura 7-6 – Comando ping6 (I).....</i>	<i>52</i>
<i>Figura 7-7 – Comando ping6 (II).....</i>	<i>52</i>
<i>Figura 7-8 – Captura de mensagens do processo de handover do MIPv6.....</i>	<i>53</i>

<i>Figura 7-9 – Análise gráfica da captura de mensagens do processo de handover do MIPv6</i>	<i>53</i>
<i>Figura 7-10 – Mensagem Binding Update enviada pelo terminal móvel ao Home Agent</i>	<i>54</i>
<i>Figura 7-11 – Mensagem Binding Acknowledgement enviada pelo Home Agent ao terminal móvel</i>	<i>55</i>
<i>Figura 7-12 – MIPv6 Tester</i>	<i>56</i>
<i>Figura 7-13 – Valores médios de handover para diferentes valores de Router Advertisements.....</i>	<i>58</i>
<i>Figura 7-14 – Cenário FMIPv6</i>	<i>59</i>
<i>Figura 7-15 – Arranque do daemon fmiprv6 do terminal móvel</i>	<i>60</i>
<i>Figura 7-16 – Interface eth1 do router de acesso AR1</i>	<i>61</i>
<i>Figura 7-17 – Arranque do daemon fmiprv6 do router de acesso AR1</i>	<i>61</i>
<i>Figura 7-18 – Comando ping6 com dois pacotes perdidos</i>	<i>62</i>
<i>Figura 7-19 – Comando ping6 com um pacote perdido.....</i>	<i>62</i>
<i>Figura 7-20 – Captura de pacotes: MinRtrAdvInterval 1 e MaxRtrAdvInterval 3</i>	<i>63</i>
<i>Figura 7-21 – Captura de pacotes: MinRtrAdvInterval 0.03 e MaxRtrAdvInterval 0.07.....</i>	<i>63</i>
<i>Figura 7-22 – Captura da mensagem PrRtAdv no router AR2</i>	<i>64</i>
<i>Figura 7-23 – Daemon do fmiprv6 no router AR2.....</i>	<i>65</i>
<i>Figura 7-24 – Handover entre a rede origem e a rede visitada</i>	<i>66</i>
<i>Figura 7-25 – Handover entre a rede visitada e a rede origem.....</i>	<i>66</i>
<i>Figura 7-26 – Handover entre a rede visitada e a rede origem e vice versa</i>	<i>68</i>

Lista de Tabelas

<i>Tabela 1 – Tempo de handover no MIPv6 com Router Advertisements a 1 e 3</i>	<i>56</i>
<i>Tabela 2 – Tempo de handover no MIPv6 com Router Advertisements a 0.03 e 0.07</i>	<i>57</i>
<i>Tabela 3 – Tempo de handover no FMIPv6</i>	<i>67</i>
<i>Tabela 4 – Comparação de handover entre MIPv6 e FMIPv6.....</i>	<i>70</i>

1 Introdução

As comunicações móveis e as redes IP, dois conceitos bastante conhecidos, actualmente surgem juntos, tornando-se na melhor combinação do século XXI, devido às características e vantagens de cada um.

As redes móveis são o tipo de tecnologia que pretende implementar a mobilidade nas redes TCP/IP (*Transmission Control Protocol/Internet Protocol*). Esta tecnologia resolve o problema das redes de área local (*Local Area Network - LAN*) ou seja, a rede deixa de ser fixa, proporcionando deste modo acessibilidade à Internet em qualquer lugar, a qualquer momento.

Deste modo surge o conceito de IP móvel onde um utilizador pode aceder à Internet, a partir de diferentes locais, utilizando um único endereço IP. Para que isto seja possível, o utilizador terá que utilizar um computador portátil, um telefone móvel, ou qualquer outro dispositivo móvel, que tenha atribuído um endereço IP.

Tal como este avanço na mobilidade IP, também na tecnologia IP foram efectuados avanços, surgindo o IPv6. Acerca desta tecnologia, remetemos para o trabalho anteriormente realizado, instalação de uma rede piloto em IPv6 [24].

Como foi acima referido, a mobilidade começa a ser um dos principais requisitos numa rede. Deste modo, e visto que o IPv6 está em permanente crescimento e é cada vez mais implementado, este projecto tem como objectivo principal o estudo da mobilidade em IPv6, mas em particular o processo de handover e as suas variantes.

1.1 Objectivos

Este projecto visa o estudo das variantes *handover* e *fast handover*, nos protocolos MIPv6 (*Mobile IPv6*) e FMIPv6 (*Fast Handover for Mobile IPv6*), respectivamente. Em cada uma destas variantes são estudadas as características que permitem estabelecer, no final deste projecto, um quadro comparativo de valores. Estas características são a latência (duração do *handover*) e o número de pacotes perdidos durante este processo.

Os objectivos deste projecto são:

- Estudar o conceito MIPv6, focando a nível de *handover*;
- Definir os possíveis cenários de teste onde possa ocorrer o processo de *handover*;
- Definir os diversos tipos de dispositivos (pda's, portáteis, telemóveis 3G, etc...) que possam ser utilizados nos possíveis cenários acima definidos.
- Escolher o(s) cenário(s) a implementar para a realização de testes, e a definição de *hardware* e *software* necessários à implementação;
- Implementar o(s) cenário(s) escolhido(s), de acordo com as definições do ponto anterior;
- Aferir resultados e comparar as variantes de *handover* testadas;
- Elaborar um relatório técnico sobre todo o trabalho desenvolvido e as conclusões dos resultados obtidos.

1.2 Estrutura do relatório

O texto apresenta-se organizado da seguinte forma:

- No capítulo 1 é apresentado o projecto e os seus objectivos;
- No capítulo 2 são apresentados conceitos básicos da mobilidade IP. É também apresentado um enquadramento do tema, as entidades funcionais e as fases envolvidas neste tipo de mobilidade;
- No capítulo 3 são apresentados os conceitos de *handover* necessários ao desenvolvimento deste projecto. Isto é, os tipos de *handover* existentes, se são de diferentes tecnologias ou áreas de abrangência, entre outros aspectos igualmente importantes;
- No capítulo 4 são apresentadas informações acerca do MIPv6, esclarecendo-se os conceitos inerentes à detecção de movimento, descoberta de rotas, configuração e duplicação de endereços. Também é apresentado um estudo acerca de autenticação e das técnicas de QoS que podem ser aplicadas neste protocolo;
- No capítulo 5 são referidas algumas variantes de *handover* existentes, nomeadamente o *handover* e o *fast handover*;

- No capítulo 6 são apresentados os cenários de teste utilizados para aferir resultados acerca do desempenho do *handover* e *fast handover*. Neste capítulo também são descritas as características dos cenários, tal como o endereçamento, *hardware* e *software* utilizados;
- No capítulo 7 são efectuados os testes e retiradas conclusões sobre estes. É efectuada a análise de resultados entre as diferentes variantes do *handover*.
- Por fim, o último capítulo apresenta as conclusões deste projecto. Aqui são expostos os objectivos realizados, se foram atingidos e superados os objectivos iniciais, com novos objectivos. Também são expostas algumas limitações do projecto, e o trabalho futuro que poderá ser desenvolvido numa possível continuação deste projecto.

2 Mobilidade IP

Um dos motivos que dificulta a adaptação das funções de mobilidade nas redes tradicionais é o facto dos protocolos de encaminhamento terem sido delineados e desenvolvidos tendo em conta que os terminais estariam sempre fixos, na sua rede de origem. No caso dos protocolos de rede utilizados na arquitectura TCP/IP, a mudança de rede significa a perda da ligação ou então exige uma forma muito complexa de encaminhamento por parte do *router*, o que poderia também levar a perdas de pacotes ou congestionamento da rede.

A tecnologia WLAN (*Wireless Local Area Network*), é o exemplo deste tipo de protocolos. Um utilizador pode estar ligado à Internet utilizando WLAN mas não pode mover-se. Isto é, não pode mudar o seu ponto de acesso (*Access Point - AP*) e manter suas ligações de rede estabelecidas (Figura 2-1).

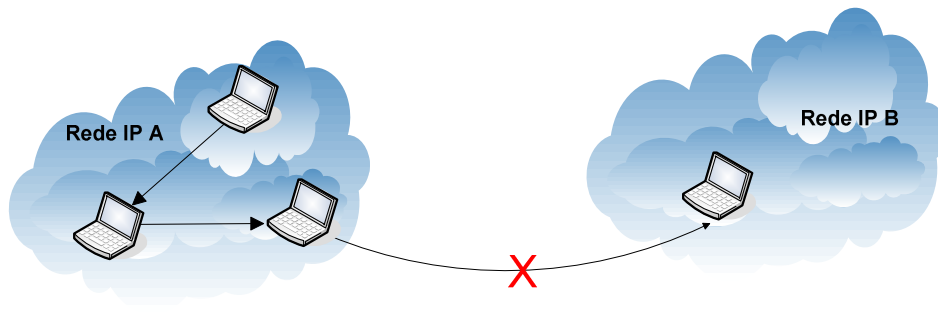


Figura 2-1 Mobilidade possibilitada pelo protocolo IP

Por esta mesma razão, o IETF (*Internet Engineering Task Force* [65]) desenvolveu o *Mobile IP* (MIP) que permite que um dispositivo móvel possa ser localizado, mesmo que esteja noutra rede, mantendo o seu endereço IP de origem (Figura 2-2). Com esta capacidade de mobilidade, o utilizador pode deslocar-se e alterar o seu ponto de acesso sem perder as ligações de rede. Isto deve-se ao facto de cada utilizador ter um IP fixo, independentemente da sua localização. [4]

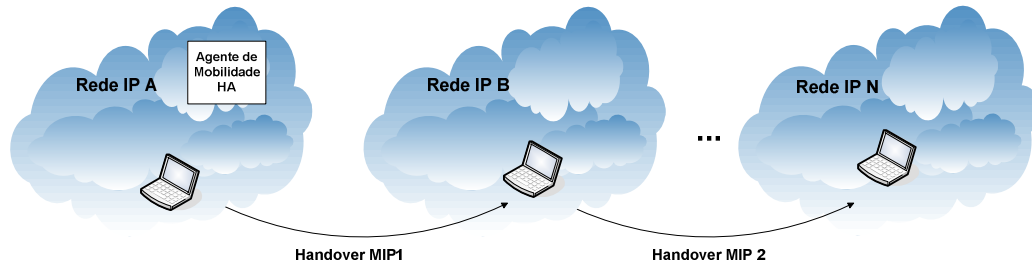


Figura 2-2 – Arquitetura MIP

Um estudo introdutório ao protocolo MIP pode encontrar-se em [1].

2.1 Terminologia

A seguinte terminologia é utilizada neste relatório.

- **Ponto de Acesso (AP)**

Equipamento de camada 2 ligado a uma rede IP, que oferece conectividade sem fios (*wireless*) ao terminal móvel.

- **Terminal Móvel (MN)**

Terminal que consegue alterar o seu ponto de acesso de uma localização para outra, sem perder as suas ligações, continuando contactável através do seu endereço origem.

- **Home Address (HoA)**

Endereço permanente, atribuído ao terminal móvel. Este endereço indica a sua ligação com a rede origem.

- **Care-of-Address (CoA)**

Endereço associado ao terminal móvel, quando este se encontra numa rede visitada. O endereço tem um prefixo que indica a rede ou sub-rede visitada. Um terminal móvel poderá ter múltiplos *Care-of-Address*, mas com diferentes prefixos.

- **Home Agent (HA)**

Router localizado na rede origem do terminal móvel, onde este registou o seu *Care-of-Address* actual. Enquanto o terminal móvel estiver fora da rede origem, o *Home*

Agent intercepta os pacotes destinados ao endereço *Home Address* do terminal móvel, encapsula-os e envia-os através de um túnel para o *Care-of-Address* registado.

- **Nó Correspondente (CN)**

Nó distante com quem o terminal móvel comunica. Este nó pode ser móvel ou fixo.

- ***Access Router (AR)***

Router por omissão do terminal móvel, e agrega o tráfego proveniente deste terminal.

- ***New Access Router (NAR)***

Router atribuído ao terminal móvel, posterior ao processo de *handover*.

- ***New Care-of-Address (NCoA)***

Care-of-Address válido na sub rede do NAR.

- ***Duplicate Address Detection (DAD)***

Determina se o endereço que o terminal móvel pretende ainda não está ser utilizado.

- ***Previous Access Router (PAR)***

Router atribuído ao terminal móvel, antes do processo de *handover*.

- ***Previous Care-of-Address (PCoA)***

Care-of-Address válido na sub rede do *Previous Access Router*.

- ***Mobility Anchor Point (MAP)***

Router localizado na rede visitada. O *Mobility Anchor Point* é utilizado pelo terminal móvel como o *Home Agent* local.

2.2 Protocolo MIP

O MIP foi desenvolvido pelo IETF [65] em duas versões, *Mobile IPv4* (MIPv4 [49]) e *Mobile IPv6* (MIPv6 [59]). O objectivo principal deste protocolo é permitir que os terminais se possam movimentar sem que as suas comunicações e ligação à rede sejam quebradas. Este facto só é possível pois o terminal móvel está sempre acessível no seu *Home Address*, através do seu *Home Agent*. O *Home Address* consiste num endereço IP especial que permanece inalterado, independentemente da posição do

terminal móvel. Ou seja, quando o terminal móvel está na rede origem, é-lhe atribuído um *Home Address* mas, quando se movimentar é-lhe atribuído um endereço provisório, *Care-of-Address*.

Contudo, a comunicação entre o nó correspondente e o terminal móvel origina o chamado encaminhamento triangular. Este encaminhamento ocorre porque o nó correspondente comunica com o terminal móvel através do seu *Home Agent* e não directamente com o *Care-of-Address*. Mas, no sentido contrário, a comunicação entre o terminal móvel e o nó correspondente, é realizada directamente entre os dois intervenientes, como apresenta a Figura 2-3.

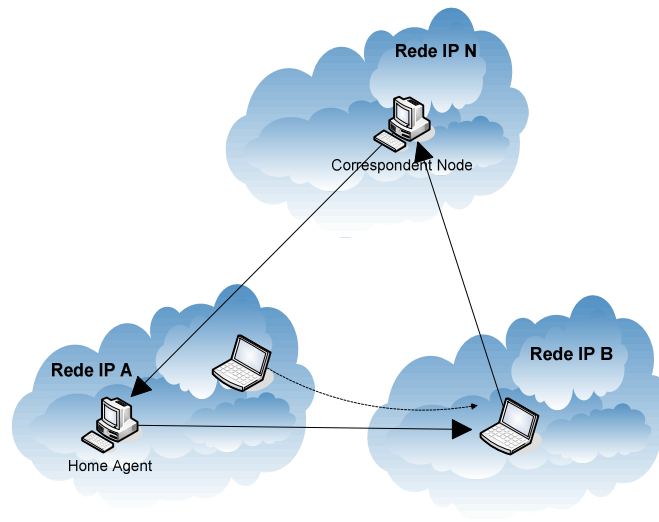


Figura 2-3 – Encaminhamento triangular

No entanto, o MIPv6 já previne este tipo de situações, recorrendo à utilização de optimização de rotas. Neste, o terminal móvel é responsável por enviar uma mensagem *Binding Update* ao nó correspondente, a informá-lo do seu novo endereço IP, o seu *Care-of-Address*. Quando o terminal móvel receber uma mensagem *Binding Acknowledgement* do nó correspondente, a comunicação passa a ser efectuada directamente entre o terminal móvel e o nó correspondente, sem necessitar de passar pelo *Home Agent*.

O *Home Agent* é responsável por controlar a localização do terminal móvel, ligando o seu *Home Address* ao *Care-of-Address*. Ou seja, assegura a conectividade entre as extremidades do túnel.

2.2.1 Principais intervenientes do MIP

O MIP tem três intervenientes:

- Terminal móvel que consiste num dispositivo móvel com um *wireless card* e protocolo IP;
- *Home Agent* que controla a localização do terminal móvel;
- Nó correspondente que consiste num nó fixo ou móvel que troca pacotes de dados com o terminal móvel.

2.2.2 Fases do MIP

Este protocolo tem quatro fases distintas.

- *Agent Discovery*, o terminal móvel tem de descobrir se está ligado à sua rede origem ou a uma rede visitada. Através de “*Router Advertisements*” (para IPv6) ou através de “*Agent Advertisements*” (para IPv4), o terminal móvel descobrirá a rede a que está agregado e por conseguinte, obterá um novo *Care-of-Address* caso não esteja na sua rede origem.
- *Registration*, o terminal móvel terá que registar o seu *Care-of-Address* no *Home Agent*, com o objectivo de o poder ligar com o *Home Address*. Este registo é realizado com mensagens “*Registration Request/Reply*” (para IPv4) ou com mensagens “*Binding Updates/Acknowledgement*” (para IPv6). Além disso, em IPv6 o terminal móvel pode enviar uma mensagem “*Binding Update*” para os nós correspondentes, com o objectivo de permitir comunicações directas, evitando que os pacotes sejam enviados para o *Home Agent*.
- *Routing and Tunneling*, o terminal móvel estabelece túneis (se necessário) com o *Home Agent* e nós correspondentes, a fim de enviar ou receber pacotes de dados. É importante referir que em IPv4 os nós correspondentes irão

continuar a enviar pacotes para o mesmo endereço IP de destino, o *Home Address*. Porém, em IPv6, os nós correspondentes enviarão os pacotes directamente para o *Care-of-Address*.

- *Handover*, o terminal móvel altera o seu ponto de acesso e necessita de:
 - 1) Descobrir a que rede se ligou, utilizando o *Agent Discovery*;
 - 2) Registar o seu novo *Care-of-Address*, através do *Registration*.

Durante esta fase, alguns pacotes de dados podem ser perdidos ou a sua entrega atrasada devido ao tempo de duração do processo de handover. [4]

3 Handover

A mobilidade IP, descrita no capítulo anterior, permite que um terminal móvel se desloque entre redes distintas (da rede origem para a rede visitada), sem que perca as suas ligações activas. Este processo é conhecido como *handover* ou *handoff*¹. Durante este processo, o terminal móvel desliga-se da rede origem antes de se ligar à rede visitada (especialmente se tiver apenas uma interface). Neste período de tempo, todas as ligações activas do terminal móvel serão quebradas. Assim, este não pode enviar nem receber pacotes provenientes das sessões estabelecidas.

Enquanto que a maioria das aplicações TCP são implementadas para gerir perdas de conectividade através do envio de pacotes de confirmação de recepção, as aplicações UDP não conseguem recuperar os pacotes perdidos. Como tal, as aplicações TCP e UDP em “tempo-real”, por exemplo aplicações de Voz sobre IP (VoIP) e aplicações de *streaming* de áudio/vídeo, são sensíveis ao tempo que o terminal móvel está sem conectividade, durante o processo de *handover*. Este tipo de aplicações necessita de *seamless handovers*, em que o processo de *handover* é suave e transparente para as camadas superiores.

Seamless handovers refere-se a *handovers* que são simultaneamente de dois tipos:

- *Smooth* – com muito poucos, ou mesmo nenhuns pacotes perdidos;
- *Fast* – com baixa latência.

Assim, dado que a Internet (como exemplo da maior rede global) necessita de suportar este tipo de aplicações, então o processo de *handover* nos protocolos MIP exige as duas qualidades acima apresentadas. Ou seja, é necessário que o processo de *handover* seja bastante rápido e com a perda mínima, ou mesmo nula, de pacotes. [11]

¹ Os termos *handoff* e *handover* são geralmente utilizados na literatura, tendo o mesmo significado.

3.1 Tipos de handover

O termo *handover* refere-se ao processo de deslocação do terminal móvel, entre a rede origem e a rede visitada.

Existem dois conceitos distintos no que se refere a *handover*, um diz respeito à tecnologia e outro à área de abrangência.

3.1.1 Handovers entre diferentes tecnologias

Em termos de tecnologia existem várias camadas do modelo OSI que são afectadas pelo *handover*. Normalmente, o *handover* que afecta apenas a camada de ligação, é designado por handover horizontal. Na Figura 3-1 podemos observar este tipo de *handover*, o terminal móvel desloca-se entre vários pontos de acesso, que são servidos pelo mesmo *router* de acesso (AR).

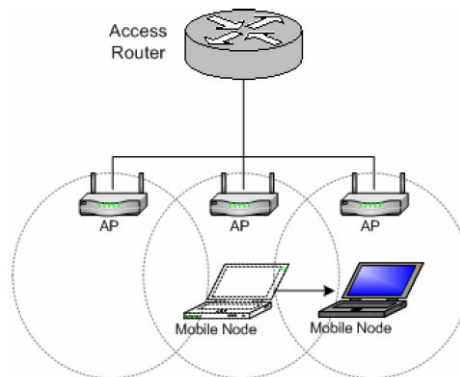


Figura 3-1 – Handover horizontal

No entanto, existe também o denominado handover vertical, que afecta a camada de ligação e a camada de rede. Neste, é atribuído um novo endereço IP ao terminal móvel, para que ele se possa deslocar entre redes diferentes, Figura 3-2.

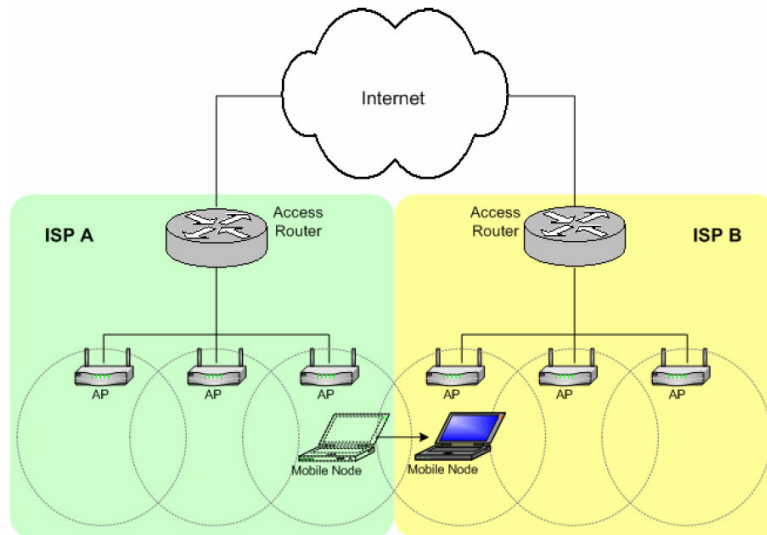


Figura 3-2 – Handover vertical

Todavia, cada tipo de handover acima apresentado pode ainda ser dividido em *inter-technology* ou *intra-technology*. Handover *inter-technology* ocorre entre redes de tecnologias diferentes, o que implica interfaces separadas no terminal móvel. Já o *handover intra-technology* ocorre entre redes da mesma tecnologia.

O *handover* horizontal deveria ser deste tipo. No entanto, tecnicamente, se a camada de rede não verificar nenhuma mudança de conectividade e de associação, pode ser considerado *inter-technology*. Por outro lado, o *handover* vertical pode ser de ambos os tipos, *inter-technology* ou *intra-technology*.

Se formos rigorosos, também podemos categorizar *handover* da camada física, que ocorre quando os pontos de acesso trocam de frequência ou de esquema de codificação da ligação actual.

É importante referir que em todos os cenários apresentados estão implícitos os protocolos de mobilidade.

3.1.1.1 Handover horizontal

A Figura 3-1 mostra um cenário simples, onde um terminal móvel se move entre pontos de acesso diferentes. Neste caso, os diferentes pontos de acesso são servidos pelo mesmo *router* de acesso, e pertencem à mesma *Extended Service Set* (ESS).

No *handover* horizontal, o terminal móvel ainda tem a mesma ligação do ponto de vista da camada de rede. Desta forma, os protocolos de MIP não são despoletados porque o terminal móvel pode utilizar o mesmo endereço IP.

Enquanto o terminal móvel se desloca entre o raio da alcance do ponto de acesso, o *Basic Service Set* (BSS) está continuamente a monitorizar a força do sinal de todos os diferentes BSSs dentro do mesmo ESS. É esta informação que o terminal móvel utiliza para decidir se se deve mover para outro ponto de acesso que apresente um sinal com maior intensidade.

3.1.1.2 Handover vertical

Como foi explicado anteriormente, o *handover* vertical ocorre quando existem alterações não só da camada de ligação mas também da camada de rede. Geralmente ocorre em cenários de *inter-technology*. No entanto, também pode ocorrer em cenários de *intra-technology*, caso o terminal móvel tenha duas interfaces da mesma tecnologia.

Na Figura 3-2, o terminal móvel alterou o seu *Internet Service Provider* (ISP). Embora os pontos de acesso possam estar geograficamente próximos, topologicamente podem estar muito distanciados. Um exemplo deste tipo de situações são os *hotspots* WLAN, de diferentes fornecedores de serviços de Internet, que se encontram fisicamente próximos.

Este tipo de *handover* pode também ocorrer em cenários onde os pontos de acesso pertencem ao mesmo domínio de administração, mas a ESSs diferentes. Ou seja, são servidos por diferentes *routers* de acesso.

A Figura 3-3 representa a situação mais complicada no *handover* vertical. Aqui o terminal móvel sai fora do alcance de uma rede WLAN, passando a estar coberto por uma rede 3G (GPRS, UMTS), onde tipicamente os fornecedores de serviço são diferentes.

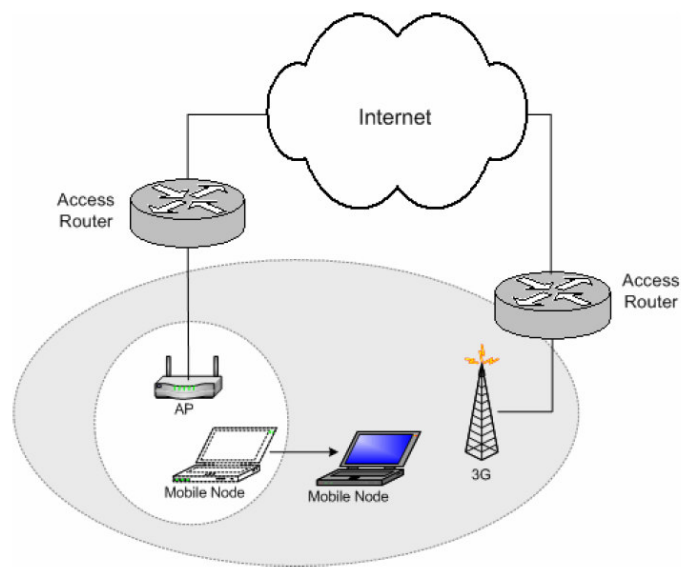


Figura 3-3 – Vertical Handover Inter-Technology

Tipicamente, em cenários de handover vertical, de fornecedores de serviços diferentes, a latência é maior. Isto deve-se não só à distância topológica mas também porque os fornecedores de serviços bloqueiam o tráfego de saída do terminal móvel até que estejam concluídos os procedimentos de autenticação e autorização. Desta forma o sucesso do *handover*, neste cenário, depende do sucesso da autenticação e autorização na rede visitada. [11]

3.1.2 Handovers por áreas de abrangência

Uma vez que a Internet já tem por si uma divisão bastante natural, da forma como está estruturada em domínios, redes e sub-redes, a divisão por áreas de abrangência vai implicar que se dividam os protocolos de mobilidade em dois níveis distintos, micro-mobilidade e macro-mobilidade. Esta divisão vai considerar características bastante diferentes da área de abrangência do protocolo, ao desempenho do mesmo e à sua complexidade, bem como da rapidez esperada das transições já referidas.

Independentemente das diferenças entre os dois níveis, existe sempre a preocupação de criar protocolos e mecanismos que sejam ajustados a cada nível. Contudo, é necessário ter sempre em vista a escalabilidade do mesmo, pois esse é um requisito

necessário para fornecer um serviço satisfatório, mesmo quando o número de utilizadores aumenta.

Os níveis macro-mobilidade e a micro-mobilidade estão representados na

Figura 3-4.

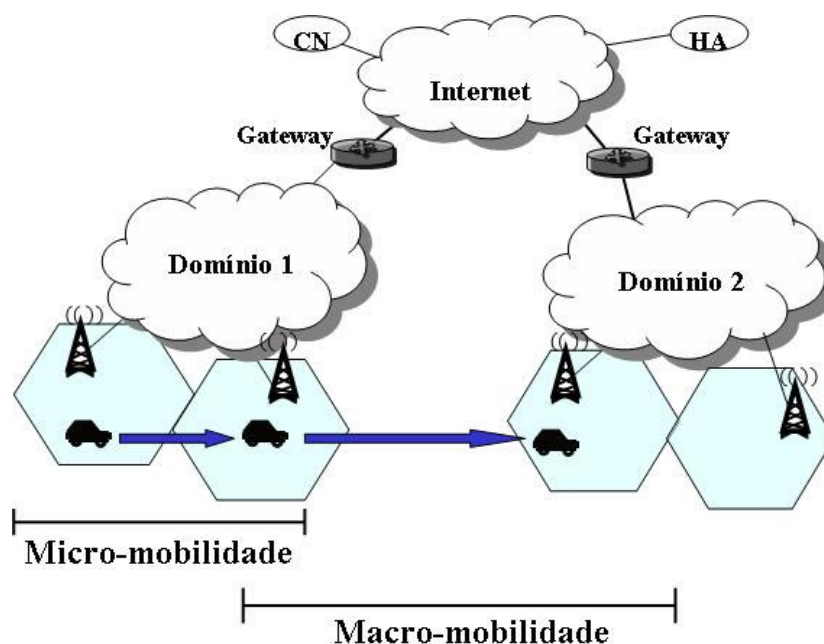


Figura 3-4 – Micro-Mobilidade Macro-Mobilidade

3.1.2.1 Macro-mobilidade

A macro-mobilidade considera o suporte da mobilidade a uma escala global, em que os terminais se movimentam para qualquer zona da Internet sem limitações. Nesta escala, o uso do protocolo são os domínios IP como um todo. Nestas condições, cada terminal móvel está localizado numa dada rede, sendo esta localização continuamente controlada pelo mecanismo de macro-mobilidade. No entanto, esta informação de localização é apenas relativa ao domínio onde o terminal móvel está. Ou seja, não se sabe a localização exacta deste no interior do domínio. A determinação desta localização será da responsabilidade do segundo nível de mobilidade.

Este protocolo só vai considerar a movimentação dos terminais móveis entre domínios IP, detectando estes movimentos e criando os mecanismos necessários para que os terminais móveis continuem a enviar e receber pacotes de dados normalmente, de forma a esconder dos outros nós da Internet a sua mobilidade.

Este tipo de movimentos, transição entre domínios, são raros e restringidos a acontecimentos isolados. Isto porque estes domínios são tipicamente geridos por entidades administrativas distintas e que estão geograficamente afastadas, em situações normais. Deste modo, esta classe de mobilidade já não tem requisitos tão apertados em relação à rapidez de transição entre redes.

3.1.2.2 Micro-mobilidade

A micro-mobilidade considera a mobilidade a uma escala local, que difere da anterior por ser limitada apenas ao interior de um domínio. Nesta escala, o uso do protocolo deixa de ser o domínio IP como um todo. Agora, cada equipamento da camada de rede constitui um único domínio IP, ficando o protocolo com a noção exacta de qual o elemento de rede concreto onde o terminal móvel está fisicamente localizado.

Neste tipo de área de abrangência são apenas consideradas as transições entre elementos de rede concretos, e não entre domínios.

Tal como os protocolos anteriores, também estes têm a função genérica de detectar a deslocação dos terminais móveis, e de criar os mecanismos necessários para que estes continuem a enviar e receber pacotes de dados normalmente, de forma a esconder aos outros nós da Internet a sua mobilidade no interior da rede local.

Este tipo de mobilidade é tipicamente gerido por uma única entidade administrativa central, ficando os seus constituintes localizados perto uns dos outros, e baseando toda a rede na mesma tecnologia. As transições dos terminais móveis entre os elementos são substancialmente mais frequentes, sendo também esperada conectividade constante, de forma a atingir o chamado *handover seamless* [14].

3.2 Decisão de Handover

Existem dois tipos de decisão que têm de ser considerados caso se pretenda um *handover seamless e lossless*, o *handover* reactivo e o *handover* proactivo. A diferença entre estes dois tipos reside no instante da decisão do *handover*. No caso do *handover* reactivo, não é estabelecida uma ligação activa com a nova rede de acesso, enquanto a anterior não quebrar. Só quando isto se verificar é que é estabelecida uma ligação para a nova rede de acesso. No caso do *handover* proactivo, é estabelecida a ligação com a nova rede de acesso mesmo antes da ligação à rede anterior quebrar, continuando esta ligação activa durante o processo de *handover* entre as duas redes.

3.2.1 Handover reactivo

Como foi acima referido, o *handover* reactivo depende da interrupção na camada de ligação. Por outro lado, a camada de rede detecta que a ligação já não está acessível pois deixa de receber *router advertisements* da ligação anterior. Deste modo, é despoletado o processo de *handover* reactivo, que conduz à interrupção da ligação durante o *handover* e consequentemente à perda de dados.

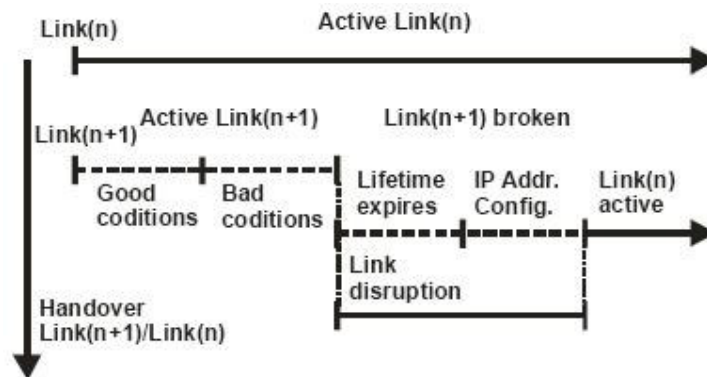


Figura 3-5 – Handover reactivo

A Figura 3-5 apresenta o processo de *handover* reactivo entre várias redes diferentes. A ligação n corresponde à ligação activa, enquanto que a ligação n+1 corresponde à ligação a outra rede de acesso.

Nesta figura podemos observar o facto do estabelecimento da ligação para a nova rede só ser efectuado após o tempo do último *router advertisement* ter expirado, dado a ligação com a rede anterior já não existir. É neste momento que é iniciado o processo de autoconfiguração do endereço IPv6, para a nova ligação.

A transição entre as duas ligações demora algum tempo, e à qual chamamos quebra de ligação. Esta é a razão da perda de pacotes durante o processo de *handover*.

O tempo da quebra de ligação deverá ser reduzido, com o objectivo de minimizar a perda de pacotes durante o *handover*.

3.2.2 Handover proactivo

Neste tipo de decisão, o terminal móvel estabelece uma ligação com a nova rede de acesso, mesmo antes da ligação com a rede anterior quebrar. Consequentemente, a perda de pacotes é reduzida pois não existe interrupção durante o processo de *handover*.

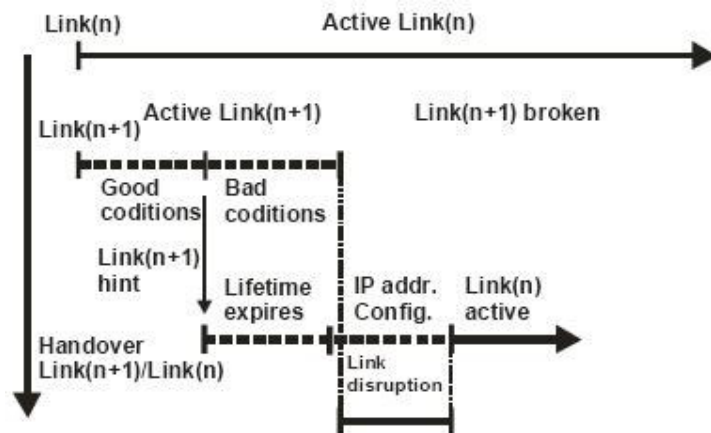


Figura 3-6 – Handover proactivo

A Figura 3-6 apresenta o processo de handover, da ligação (n) para a (n+1), tal como no *reactive handover*.

A decisão do *handover* proactivo é baseada numa sugestão, que contém informação da camada L2, sobre a ligação activa. Esta sugestão é lançada quando a ligação activa fica instável. Isto despoleta a configuração antecipada do endereço IPv6. Assim que este está disponível, é criada a ligação (n+1), entre o terminal móvel e a nova rede de acesso, e o processo de handover é estabelecido. A partir deste momento, a ligação já pode ser utilizada para transmissão de dados.

Neste tipo de decisão, a duração da quebra de ligação é reduzida, pois a nova ligação é estabelecida durante a utilização da ligação anterior.

Como já foi acima referido, o *handover* proactivo reduz a perda de pacotes durante o processo de handover. No entanto, também pode implicar uma redução do débito, caso a nova ligação escolhida suportar menos largura de banda que a anterior.

4 MIPv6

A especificação MIPv6 é uma norma proposta pelo IETF [65] para permitir a mobilidade transparente de terminais em IPv6. O protocolo permite que um terminal móvel se desloque de uma rede para outra sem necessitar de alterar o seu endereço IPv6. Quando o terminal se move para uma rede visitada, os pacotes são enviados para este através do seu *Home Address*. Este endereço é atribuído ao terminal móvel dentro da sua rede origem. Desta forma o movimento do terminal é completamente transparente à camada de transporte e a outros protocolos de camadas superiores.

Quando um terminal móvel altera o seu ponto de acesso à Internet, de uma rede IPv6 para outra, ocorre o processo de handover do MIPv6. Este processo é semelhante ao procedimento de auto-configuração que um terminal IPv6 sofre quando se liga a uma rede IPv6, embora tenha algumas diferenças importantes:

- O terminal móvel tem de detectar que se moveu para uma rede visitada;
- Uma vez configurado, o terminal móvel tem de informar o seu *Home Agent* e cada *Correspondent Node* da sua nova localização;
- Durante o processo de handover, as ligações de camadas superiores ainda se mantêm. Por este motivo, o handover deve ser realizado o mais rápido possível para minimizar a interrupção de pacotes perdidos e atrasados.

O processo de handover do MIPv6 é composto por sete passos, apresentados de seguida e ilustradas na Figura 4-1:

- Detecção de movimento;
- Descoberta de *router* vizinho;
- Configuração do endereço *Care-of-Address*;
- Detecção de duplicação de endereços;
- Autenticação, Autorização e Contabilização (*Authentication, Authorization and Accounting* (AAA)) e Qualidade de Serviço (QoS);
- Registrar o novo endereço *Care-of-Address*

- Envio de mensagem *Binding Update* para confirmar o fim do processo de handover;

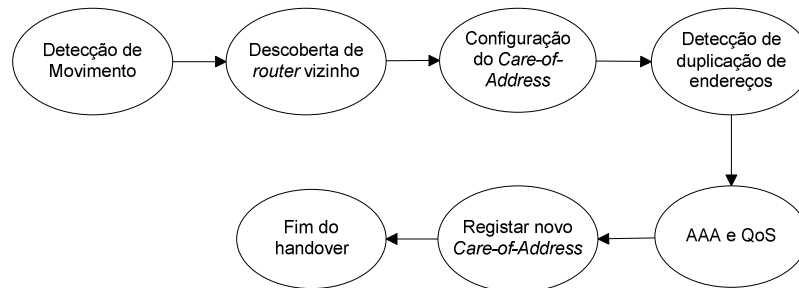


Figura 4-1 – Etapas do handover em MIPv6

Nas correntes especificações do IPv6, todos os passos acima apresentados são considerados indispensáveis, à excepção do *Authentication* e *Authorization*. Isto pelo facto de ser possível implementar o MIPv6 sem recorrer a mecanismos de autenticação e autorização, dado estes não interferirem directamente com o processo de mobilidade. São apenas mecanismos de segurança.

A Figura 4-2 representa o processo de handover no MIPv6 e as respectivas mensagens trocadas entre os intervenientes.

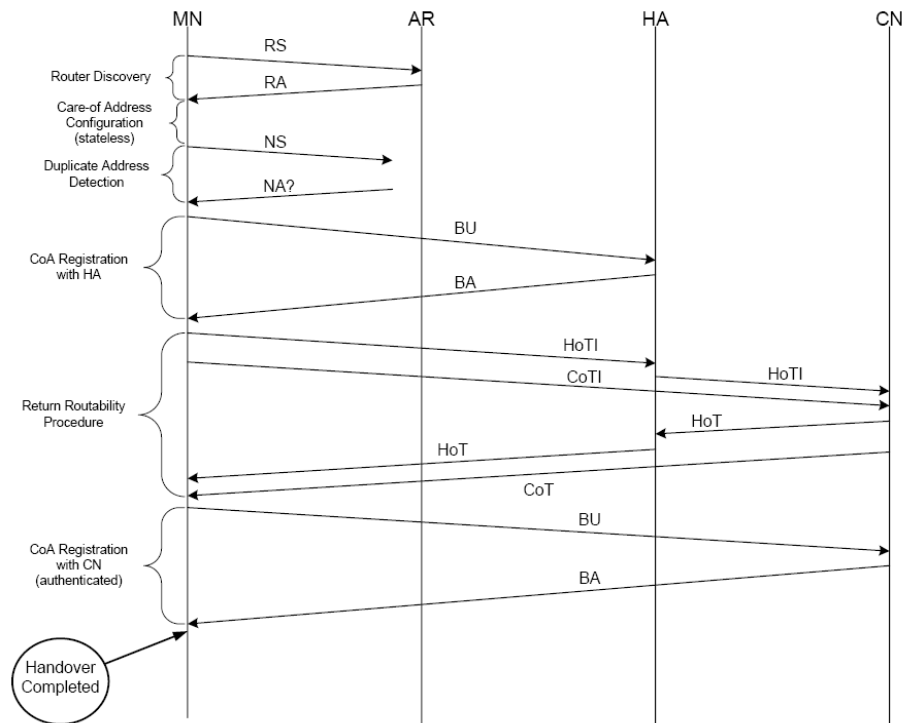


Figura 4-2 – Processo de handover MIPv6

4.1 Passos do processo de handover no MIPv6

4.1.1 Detecção de movimento

No protocolo MIPv6, é da responsabilidade do terminal móvel detectar que se mudou de rede. Esta tarefa nem sempre é simples, mas pode ser detectada quando:

- O *router* de acesso deixa de estar contactável;
- Está disponível um novo *router* de acesso.

Para determinar se o seu *router* de acesso ainda tem conectividade bi-direccional, o terminal móvel envia mensagens *Neighbor Unreachability Detection*. O envio destas mensagens funciona do seguinte modo: quando um terminal IPv6 tem um pacote para enviar, verifica a tabela de endereços IP dos seus vizinhos para determinar o endereço do próximo *router*. Esta tabela também tem associado o estado de cada vizinho, onde *REACHABLE* indica que o vizinho está contactável.

Um terminal IPv6 considera um vizinho contactável caso tenha recebido a confirmação da entrega dos pacotes enviados. Esta confirmação pode chegar ao

terminal móvel através da recepção de uma mensagem *Neighbor Advertisement* em resposta a um *Neighbor Solicitation* enviado pelo terminal móvel, ou então através de informação enviada por parte dos protocolos de camada superior.

Quando o terminal IPv6 pretende enviar um pacote e não existe nenhuma entrada na tabela dos endereços IP dos seus vizinhos definida como *REACHABLE*, então é enviada uma mensagem *Neighbor Solicitation*.

É de notar que esta funcionalidade só ocorre quando o terminal móvel tem pacotes para enviar. Assim, no pior caso, se o terminal móvel não estiver a enviar nenhum pacote, ele só vai detectar que se moveu quando receber uma mensagem *Unsolicited Router Advertisement* do novo *router* de acesso. Infelizmente pode ocorrer a perda de pacotes e elevada latência, caso o terminal móvel esteja a receber pacotes em tempo-real. Neste cenário o terminal móvel pode nem estar a enviar muitos pacotes. Podem ser apenas pacotes TCP ocasionais ou confirmações da camada de aplicação. Este tipo de pacotes não permite que o terminal móvel consiga detectar que o *router* de acesso não se encontra contactável.

No entanto, o facto do terminal móvel receber um aviso por parte do *router* não garante que se o terminal se deslocou. Pode ser só um aviso de um novo *router* a ligar-se à rede. E como é referenciado na RFC 2461 [49], este tipo de mensagem não pode servir como uma confirmação bi-direccional dado que só confirma a conectividade na direcção do terminal móvel.

4.1.2 Descoberta de router vizinho

Esta etapa ocorre quando o terminal móvel recebe uma mensagem *Router Advertisement* do *router* de acesso mais próximo.

O terminal móvel pode receber esta informação em duas situações diferentes:

- Quando o *router* envia uma mensagem *Router Advertisement* para o endereço *multicast* de todos os terminais;
- Em resposta a uma mensagem *Router Solicitation* enviada pelo terminal móvel.

Esta última mensagem é enviada caso o terminal móvel detecte que o seu *router* de acesso actual já não está contactável e receberá uma mensagem *Solicited Router Advertisement* ou *Unsolicited Router Advertisement* do novo *router* de acesso.

Não existe garantia de qual das mensagens vai receber. Isto é, depende muito das circunstâncias em que ocorre o handover: o período de envio de mensagens *Router Advertisement* por parte do novo *router* de acesso e os valores exactos dos temporizadores naquele instante.

Mas como foi referido anteriormente, mesmo que o terminal móvel receba uma mensagem *Router Advertisement*, isso não garante que se moveu, pelo que este pode querer confirmar se o seu *router* de acesso actual já não se encontra contactável, antes de utilizar o novo *router* de acesso.

4.1.3 Configuração do Care-of-Address

O terminal móvel terá que se configurar com um endereço IPv6 a utilizar na rede visitada. Este será o seu *New Care-of-Address*. Esta configuração poderá ser realizada de forma *statefull* ou *stateless*. [54] O método adoptado poderá se assinalado com a configuração de *flags* nas mensagens *Router Advertisement*.

Outra forma de configuração é a manual, embora qualquer configuração que necessite de intervenção humana não seja a melhor opção em cenários de MIPv6.

4.1.4 Detecção de duplicação de endereços

A detecção de duplicação de endereços, DAD, é realizada da mesma forma que quando um terminal móvel se liga a uma rede IPv6 para confirmar que os seus endereços são únicos na ligação. Um terminal móvel que se desloca terá que verificar se o endereço *Care-of-Address*, atribuído na fase de configuração, não existe. Este procedimento acontece independentemente da forma que o *Care-of-Address* foi obtido (*statefull*, *stateless* ou manualmente).

O procedimento da detecção e duplicação de endereços está especificado na RFC 2462 [54].

4.1.5 Authentication, Authorization and Accounting e Qualidade de Serviço

4.1.5.1 AAA

Se um terminal móvel atravessar diferentes domínios de administração é provável que se verifique alguma necessidade de negociação AAA antes de ter acesso à rede visitada.

Caso se verifique esta negociação, é necessário verificar as iterações de *handshake* entre o terminal móvel, o servidor local de AAA e o mesmo servidor localizado no *Home Agent*.

As credenciais são pedidas ao terminal móvel e serão transmitidas ao servidor local de AAA. Este verifica a identidade do terminal móvel no servidor do *Home Agent*, antes de conceder acesso à rede. Isto implica a ocorrência de pelo menos um *Round Trip Time* entre o servidor local e o servidor no *Home Agent*, e outro entre o servidor local e o terminal móvel. Este tempo poderá ser reduzido a apenas um *Round Trip Time*, se o servidor local de AAA estiver localizado no novo *router* de acesso.

A latência introduzida neste procedimento vai depender da localização física do:

- Terminal móvel;
- Servidor local de AAA;
- Servidor de AAA no *Home Agent*;
- E das especificações concretas de AAA implementadas.

Todavia, a latência introduzida vai ser superior à necessária para que se verifique um handover considerado *seamless*.

Caso as negociações AAA falharem devido à falta de credenciais, erros de rede, erros de protocolos de autenticação, entre outros, isso implica que o acesso à rede visitada seja negado e o processo de handover não é terminado.

4.1.5.2 QoS

Se o terminal móvel necessitar de parâmetros de QoS, existem dois tipos de abordagem que podem ser considerados:

- Utilização de serviços integrados, como o *Resource ReSerVation Protocol* [64]. Desta forma os parâmetros de QoS são enviados separadamente da transmissão de dados. Assim sendo, não é introduzida nenhuma latência ao handover, e mesmo que não seja estabelecido nenhum parâmetro de QoS, o handover poderá ser concluído com sucesso;
- Utilização de serviços diferenciados. Não é necessário estabelecer nenhuma iteração antes de ocorrer o handover.

No entanto, caso:

- Falhe algum parâmetro de QoS antes do handover terminar,
- Os parâmetros QoS sejam rejeitados pela rede visitada,
- Existam atrasos na negociação dos parâmetros de serviços diferenciados ou falhas nestes,

as sessões de aplicações que necessitem de um certo nível de QoS na rede, poderão ser prejudicadas. Um exemplo deste tipo de sessões é uma aplicação de VOIP.

4.1.6 Registrar novo Care-of-Address

Quando um terminal móvel detecta que se deslocou (obteve um novo *Care-of-Address* e tem acesso à rede visitada), tem de informar o *Home Agent* da sua nova localização.

Durante o período de tempo que perdeu na ligação com o *router* de acesso anterior, até informar o *Home Agent* da sua nova localização, os pacotes a si destinados são perdidos e não consegue enviar pacotes para os nós correspondentes.

O terminal móvel regista o seu novo *Care-of-Address* no *Home Agent* através de uma mensagem *Binding Update*. A confirmação do *Home Agent* é feita com uma mensagem *Binding Acknowledgement*. A partir deste momento o *Home Agent* está apto a enviar os pacotes destinados ao *Home Address* do terminal móvel, mas que serão enviados para o seu novo *Care-of-Address*.

4.1.7 Fim do processo de handover

Nesta etapa, o terminal móvel já pode enviar uma mensagem *Binding Update* a cada nó correspondente, a informar da sua nova localização e do seu novo *Care-of-Address*. Antes deste envio, é realizada ainda a verificação de *Return Routability* para garantir aos nós correspondentes que o *Binding Update* é autêntico.

Quando o terminal móvel recebe mensagens *Binding Acknowledgement* dos nós correspondentes, o processo de handover está completo, embora em alguma literatura o processo de handover seja considerado completo quando o terminal móvel regista o seu novo *Care-of-Address* no *Home Agent*.

4.2 Latência do MIPv6

Numa rede sem fios, a latência pode afectar significativamente o processo de handover. No MIPv6, os componentes apresentados de seguida influenciam o tempo do processo de handover:

- Tempo de detecção de movimento (td): período de tempo necessário ao terminal móvel para detectar que se deslocou para um novo ponto de acesso.
- Tempo de configuração do *Care-of-Address* (ta): intervalo de tempo desde que o terminal móvel detecta que se movimentou e o tempo de configuração do seu *Care-of-Address*. Este período de tempo inclui a detecção de duplicação de endereços.
- Tempo de estabelecimento de contexto (tc): período de tempo entre a atribuição do *Care-of-Address* e o estabelecimento do estado de contexto apropriado. Este estado pode ser qualquer parâmetro de AAA ou QoS.
- Tempo de registo (tr): período entre o envio de uma mensagem *Binding Update* ao *Home Agent* e a recepção do *Binding Acknowledgement* como confirmação da mensagem anterior. Este tempo ocorre após o estabelecimento de contexto.
- Tempo de optimização de rotas (to): período de tempo desde o registo do *Care-of-Address* no *Home Agent* até completar a etapa de optimização de rotas

para a lista de nós correspondentes. Este tempo inclui o procedimento de *return routability*.

Deste modo, pode-se determinar o valor total da latência, definido como a soma das latências explicadas anteriormente.[11]

5 Fast Handover

O protocolo *Fast Handover* para a mobilidade IPv6 (FMIPv6) [60] foi proposto com o objectivo de minimizar o tempo de interrupção do serviço quando um terminal móvel IPv6 se desloca para uma rede diferente da rede origem.

Sem este tipo de protocolo, o terminal móvel não consegue enviar nem receber pacotes, desde o ponto em que se desliga da rede origem, até que regista o novo endereço *Care-of-Address* na rede visitada. Contudo, este tipo de interrupção não é aceitável em serviços que correm em tempo real, como o VoIP.

A ideia do FMIPv6 é providenciar informação relativa à camada de ligação, com o objectivo de prever ou responder prontamente a um evento de handover. Assim, a conectividade IP no novo ponto de acesso é rapidamente estabelecida.

O facto de ser estabelecido um túnel entre o *router* actual e o novo *router* de acesso, torna possível a conectividade IP. Isto, mesmo antes de ser efectuado o registo do endereço no *Home Agent* ou no nó correspondente.

O FMIPv6 permite que os serviços em tempo real sejam restabelecidos mesmo antes do processo de registo de endereços estar concluído. Dado que o registo envolve algum tempo, este protocolo consegue valores mais baixos de interrupção de serviços em tempo real, comparado com o handover normal do MIPv6.

5.1 Descrição

O processo de *handover* do FMIPv6 é despoletado na camada de ligação. Aqui, os *triggers* são os intervenientes mais importantes deste processo. Todavia, também é necessário garantir que o atraso da camada de rede e o tempo em que a comunicação está interrompida sejam mínimos. Quando um terminal móvel se desloca entre dois *routers* de acesso, necessita para actualizar o seu *Home Agent*, através da utilização de mensagens *Binding Update*. Durante este período de tempo o terminal móvel não consegue comunicar, sendo este o tempo que se pretende minimizar no *fast handover*.

O protocolo FMIPv6 permite a antecipação do handover da camada de rede de tal modo que o tráfego de dados pode ser redireccionado para a nova localização do terminal móvel, mesmo antes de ele se deslocar para lá.

O terminal móvel, ou o *router* de acesso anterior, pode iniciar o procedimento de *fast handover*. Este processo é realizado através da utilização de informação *wireless* da camada de rede ou então dos *triggers*, que informam que o móvel irá ficar sem abrangência por parte do *router* de acesso anterior e do novo *router* de acesso. Se o *trigger* for recebido pelo terminal móvel, este iniciará o processo de *handover* enviando uma mensagem *Proxy Router Solicitation* para o *router* de acesso anterior. Por outro lado, se o *trigger* for recebido pelo *router* de acesso anterior, este transmitirá uma mensagem *Proxy Router Advertisement* para o terminal móvel adequado, sem a necessidade de solicitações. [61]

As trocas básicas da mensagem de *Fast Handover* estão ilustradas na Figura 5-1:

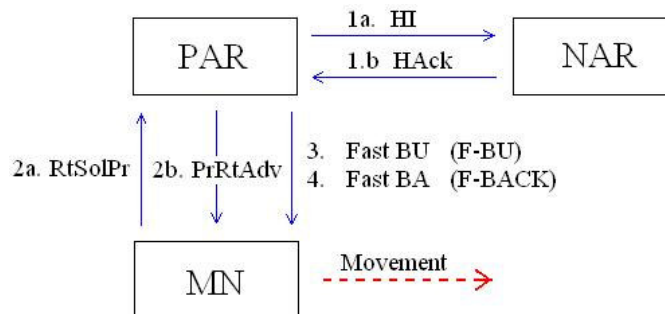


Figura 5-1 – Fast Handover IPv6 Handover Protocol

5.2 Troca de mensagens

O *handover* do protocolo FMIPv6 consiste na seguinte troca de mensagens:

- O terminal móvel envia uma mensagem *Router Solicitation for Proxy* para obter informação acerca dos *routers* vizinhos;
- O terminal móvel recebe uma mensagem *Proxy Router Advertisement* com informação relativa aos pontos de acesso;

- O terminal móvel envia uma mensagem *Fast Binding Update* para o *router* de acesso anterior;
- O *router* de acesso anterior envia uma mensagem *Handover Initiated* para o próximo *router* de acesso;
- O próximo *router* de acesso responde com uma mensagem *Handover Acknowledge*;
- O *router* de acesso anterior envia uma mensagem *Fast Binding Acknowledge* para o terminal móvel, através da nova ligação. Esta mensagem pode ser enviada através da ligação anterior caso o *Fast Binding Update* tenha sido enviado por aqui;
- O terminal móvel envia uma mensagem *Fast Neighbor Advertisement* para o próximo *router* de acesso, depois de se ter ligado a ele.

5.3 Modos de operação

Dependendo se o *router* de acesso anterior recebe ou não a mensagem *Fast Binding Acknowledge*, existem dois modos de operação. Estes são o modo predictivo e o modo reactivo. De seguida apresenta-se uma breve explicação do funcionamento de cada um deles.

5.3.1 Modo predictivo

Neste modo, o terminal móvel descobre os pontos de acesso que o rodeiam e o respectivo *router* de acesso que lhes corresponde. Quando o processo de associação entre o ponto de acesso e o terminal móvel está concluído, este já sabe as coordenadas do novo *router* de acesso, o prefixo e os endereços MAC e IP. Através das mensagens *Fast Binding Update* e *Fast Binding Acknowledge*, o terminal móvel consegue formular um novo *Care-of-Address*, que deverá ser aceite pelo novo *router* de acesso antes do terminal móvel se deslocar. Uma vez que este já efectuou a transição para o novo *router* de acesso, pode utilizar o seu novo *Care-of-Address* sem ter que descobrir o prefixo da sub rede. Doravante a latência é eliminada.

Logo que o terminal móvel estabeleça a ligação com o *router* de acesso, envia uma mensagem *Fast Neighbor Advertisement* anunciando sua presença. Além disso, o *router* de acesso anterior criará um túnel e enviará os pacotes para o novo *Care-of-Address* enquanto o terminal móvel não registar o novo *Care-of-Address* junto do *Home Agent* e dos nós correspondentes. Futuramente, todos os pacotes dessa ligação serão perdidos.

A

Figura 5-2 representa a troca de mensagens de *fast handover* no modo preditivo.

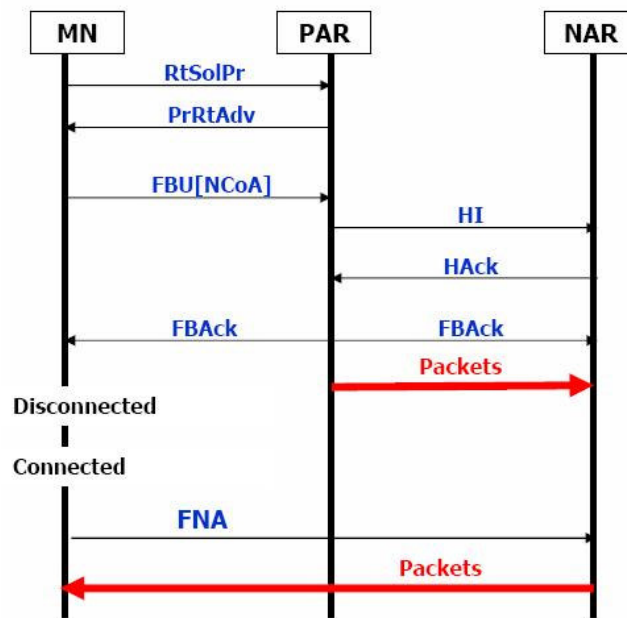


Figura 5-2 –Troca de mensagens no modo preditivo

5.3.2 Modo reactivo

Neste modo o terminal móvel não recebe a mensagem *Fast Binding Acknowledge* através da ligação com o *router* de acesso anterior. Isto deve-se ao facto do terminal móvel se deslocar antes de enviar a mensagem *Fast Binding Update* ou então realiza o *handover* de camada de ligação imediatamente após ter enviado esta mensagem, mas antes de receber o *Fast Binding Acknowledge*. De qualquer modo, o terminal móvel não consegue garantir que a mensagem *Fast Binding Update* foi correctamente processada.

O terminal móvel envia uma mensagem *Fast Neighbor Advertisement* quando se liga directamente ao novo *router* de acesso. Este troca mensagens de *Fast Binding Update* e *Acknowledge* com o *router* de acesso anterior. A partir deste momento é estabelecida uma ligação directamente entre o novo *router* de acesso e o terminal móvel, tal como no modo preditivo.

No modo reactivo é esperado um pior desempenho face ao modo preditivo, devido ao terminal móvel realizar o processo de detecção e duplicação de endereços depois de se ligar ao novo *router* de acesso [18].

A Figura 5-3 representa a troca de mensagens de *fast handover* no modo reactivo.

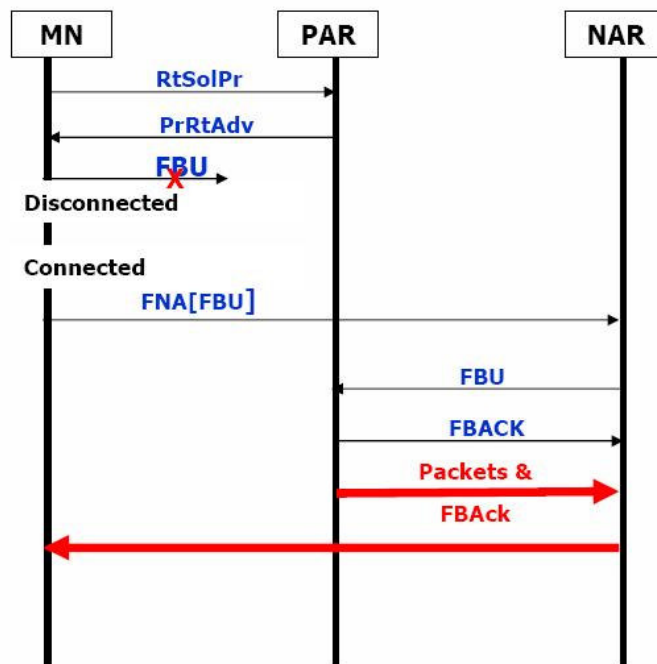


Figura 5-3 – Troca de mensagens no modo reactivo

5.3.3 Comparação entre o modo preditivo e o modo reactivo

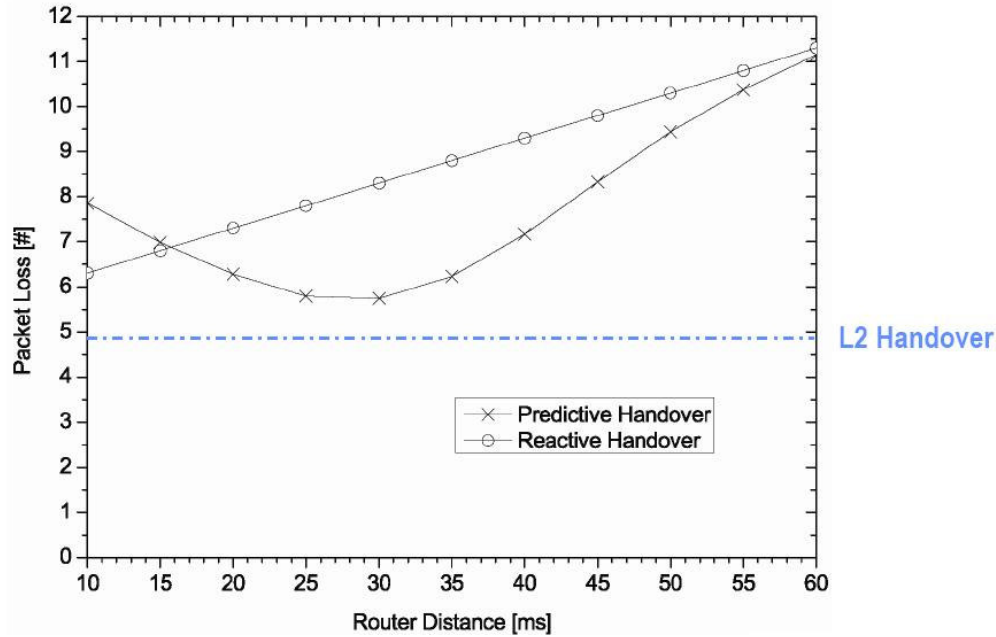


Figura 5-4 –Modo preditivo versus modo reactivo

A Figura 5-4 apresenta a comparação de *fast handover* no modo preditivo e no modo reactivo, relativamente à perda de pacotes. Esta comparação é realizada em função da distância entre *routers*. Podemos observar que, tal como foi acima documentado, o modo preditivo tem muito menor perda de pacotes que o modo reactivo.

6 Cenário de Teste

Neste capítulo é apresentado o cenário de testes. Este é composto por um cenário geral e dois que derivam deste. O cenário geral contempla os equipamentos com as funcionalidades necessárias para efectuar os testes de handover e comprovar o conceito dos protocolos em estudo.

6.1 Cenário Geral

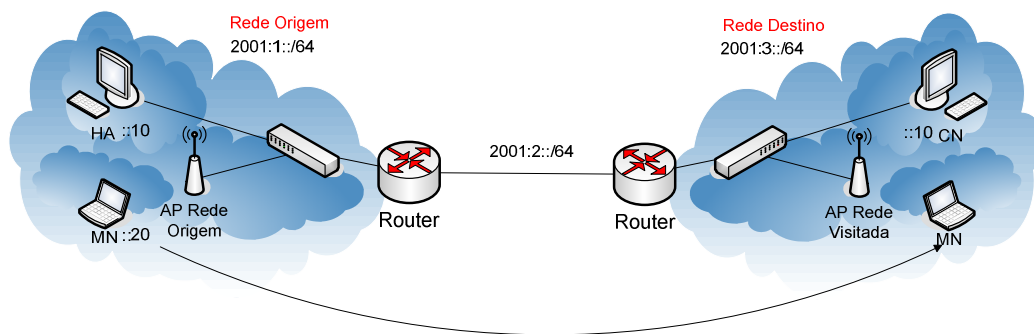


Figura 6-1 – Cenário geral

O cenário apresentado na Figura 6-1, composto por três redes distintas, onde uma delas é a rede de *backbone*, permite analisar o processo de mobilidade de um terminal móvel entre redes IPv6.

Ao comprovar o conceito de MIPv6 e de FMIPv6 entre duas redes distintas, é possível adaptar a um cenário mais real, onde estão envolvidas diversas redes. Na eventualidade de se pretender realizar testes numa rede mais extensa, apenas seria necessário efectuar algumas alterações na configuração para abranger todas as redes existentes. Estas alterações passariam pela adição dos equipamentos necessários e a sua respectiva configuração nos ficheiros correspondentes.

No cenário apresentado, o nó correspondente encontra-se localizado na rede visitada. Porém, poderia estar localizado na rede origem ou em qualquer ponto da Internet.

No cenário de testes, estabeleceu-se que o terminal móvel apenas comunica com um nó correspondente. Contudo, na realidade poderia comunicar com vários nós em simultâneo.

Testar a mobilidade não implica obrigatoriamente a utilização de pontos de acesso *wireless*. É possível executar estes testes em ambiente *wired*. Contudo, num ambiente desta natureza existe sempre a quebra da ligação e perda de pacotes devido à mudança física do cabo de rede para a rede visitada. Assim, de forma a minimizar ao máximo os parâmetros que contribuem para o aumento do *handover* e para tornar os resultados dos testes os mais fidedignos possíveis, apenas será considerado o ambiente *wireless*.

6.2 Endereçamento

O endereçamento utilizado no cenário é *site-global*, no entanto poderiam ser utilizados endereços *site-local*. É importante referir que os endereços *link-local* não funcionam para este tipo de testes.

Deste modo, o endereçamento é o seguinte:

- 2001:1::/64, para a rede IPv6 origem;
- 2001:2::/64 para a rede *backbone*;
- 2001:3::/64, para a rede IPv6 visitada.

6.3 Hardware

Nesta secção são descritos os equipamentos e as características principais para implementar um cenário desta natureza.

Para produzir o cenário, é necessário o seguinte equipamento:

- 2 Router com suporte de mobilidade
- 2 Switchs;
- 2 computadores desktop, cada um com uma placa de rede ethernet;
- 1 computador portátil com uma placa de rede wireless;

- 2 pontos de acesso.

6.4 Divisão de cenários

Quando se estabelece um cenário para efectuar testes numa rede, determinam-se também os equipamentos necessários a esses mesmos testes. No entanto, quando não existem equipamentos disponíveis para a realização dos testes, e a sua aquisição se torna demasiado dispendiosa, existem algumas formas para contornar essa falta de recursos.

Neste caso, optou-se por recorrer à utilização de computadores para realizar as mesmas funções que um *router* IPv6. Isto só é possível pois estes mesmos computadores estão configurados para fazer o encaminhamento entre as redes que lhe estão associadas.

Desta forma, de seguida são apresentados individualmente cada cenário e o respectivo equipamento.

6.4.1 Cenário MIPv6

Na Figura 6-2 estão representados os equipamentos necessários para a realização de testes sobre a mobilidade em IPv6:

- 1 computador portátil com uma placa de rede *wireless*;
- 2 pontos de acesso;
- 2 computadores *desktop*, cada um com duas placa de rede *ethernet*.

Este cenário é idêntico ao disponível no *How To* publicado pelo MIPL. No entanto, é importante referir o agrupamento entre o *Home Agent* e o *router* da rede origem, visto que o MIPv6 suporta esta associação dentro do mesmo router, que neste caso é substituído por um computador *desktop*. O mesmo se verifica na rede visitada. O router é agrupado com o nó correspondente.

O encaminhamento será realizado através de rotas estáticas nos ficheiros de configuração, embora também se possa recorrer à utilização de protocolos de

encaminhamento, como por exemplo o ripng ou o Zebra2. Este último pelo facto de se utilizar o sistema operativo Linux.

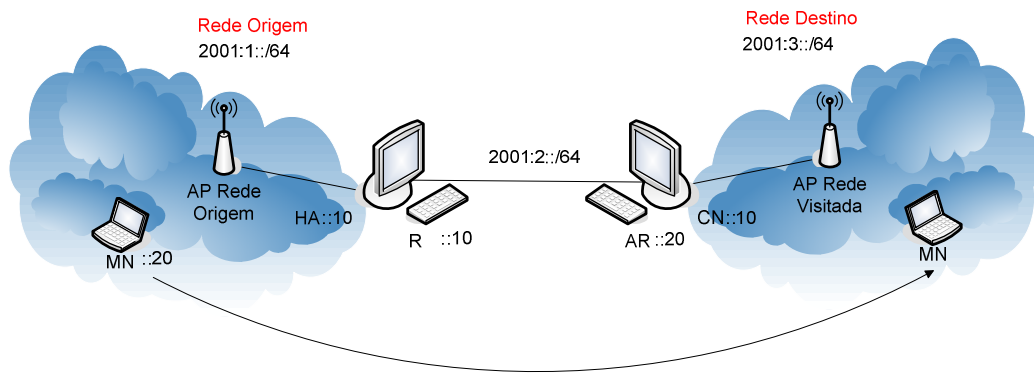


Figura 6-2 – Cenário MIPv6

6.4.2 Cenário FMIPv6

Na Figura 6-3 estão representados os equipamentos necessários para a realização de testes sobre o FMIPv6:

- 1 computador portátil com uma placa de rede *wireless*;
- 2 pontos de acesso;
- 2 computadores *desktop*, cada um com duas placa de rede *ethernet*;
- 1 computador *desktop*, cada uma placa de rede *ethernet*;
- 1 switch.

O cenário de teste a utilizar baseia-se no apresentado no *testbed* do projecto fmipv6.org. Utilizar-se-á este cenário devido ao facto de ainda não existirem testes realizados com o cenário implementado para o MIPv6, onde o *Home Agent* é utilizado

² Aplicação de encaminhamento de protocolos TCP/IP.

também como *router*. Esta informação foi recolhida através do repositório de e-mails do projecto fmipv6.org que aconselhavam deste cenário.

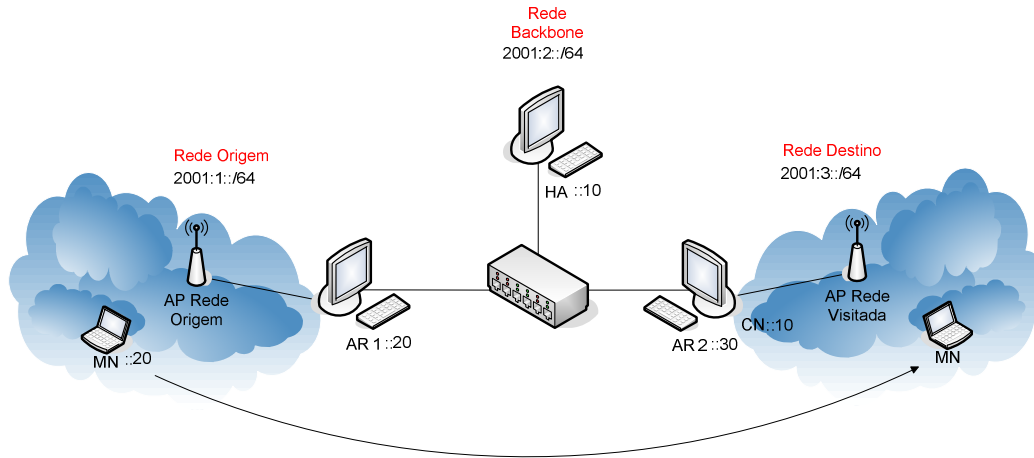


Figura 6-3 – Cenário FMIPv6

Tal como no cenário anterior o encaminhamento será feito através de rotas estáticas mas poderá ser utilizado um protocolo de encaminhamento. Na implementação do cenário apresentado no projecto fmipv6.org foi utilizada a aplicação Quagga.

6.5 Software

Nesta secção são apresentados os softwares necessários. Foram analisados diversos sistemas operativos, analisadores de rede, aplicações de teste, implementações existentes, entre outros.

6.5.1 Sistemas operativos

Na análise de sistemas operativos, foi necessário verificar quais os sistemas que contém suporte para protocolos de mobilidade. Desta forma foram analisados alguns sistemas operativos *opensource* e alguns comerciais. Destes últimos constam o Windows XP, o Windows Vista e o sistema operativo IOS da Cisco. Relativamente

aos sistemas *opensource*, apenas foi analisado o Linux, em algumas das suas diversas distribuições, nomeadamente o Fedora Core e o Ubuntu.

6.5.1.1 Sistemas operativos comerciais

Em relação à Cisco e ao seu IOS, as versões que contém suporte para mobilidade são a 12.3(14)T, 12.4 e 12.4(T). Estas versões têm suporte para as funcionalidades de *Home Agent*, e nó correspondente.

Por outro lado, o Windows XP apenas tem suporte para a funcionalidade de nó correspondente. Já esteve disponível uma aplicação desenvolvida pela Microsoft, com as funcionalidades de *Home Agent*, terminal móvel e nó correspondente, mas neste momento já não se encontra disponível.

O Windows Vista tem suporte para protocolos de mobilidade, mas até à data da escrita deste relatório não foi possível identificar quais as funcionalidades incorporadas neste sistema operativo.

6.5.1.2 Sistemas operativos opensource

O sistema operativo Linux apresenta mais vantagens, relativamente aos sistemas operativos anteriormente descritos, pelo facto de ser *opensource* e apresentar um vasto leque de distribuições. Outra grande vantagem deste sistema operativo deve-se ao facto da existência de implementações *opensource* dos diversos protocolos de mobilidade. Desta forma, e como a distribuição Fedora Core já foi anteriormente utilizada em testes de mobilidade, nomeadamente em [1], decidiu-se utilizar esta distribuição, com o Core 6 (o mais recente).

As implementações mais recentes existentes dos protocolos MIPv6 e FMIPv6 foram implementadas para o *kernel* 2.6.16. Contudo, o Fedora Core 6 traz de origem o *kernel* 2.6.18. Deste modo, será necessário compilar o *kernel* que contém suporte para os protocolos de mobilidade.

Na implementação do FMIPv6 utilizar-se-á uma máquina com a distribuição Ubuntu, com o *kernel* 2.6.16, apenas para efectuar testes de compatibilidade entre diferentes distribuições.

6.5.2 Implementações de mobilidade

Nesta secção são descritas as *packages* necessárias ao correcto funcionamento dos protocolos de mobilidade IPv6 no Linux.

6.5.2.1 MIPL

O *Mobile IPv6* para Linux (MIPL) consiste num módulo para o *kernel* que, como o próprio nome indica, permite a implementação de mobilidade IPv6 para Linux.

O MIPL, até à versão 1.1, foi desenvolvido por um projecto denominado GO-Core, do Instituto de Engenharia de Software e Telecomunicações, na Universidade de Tecnologia de Helsínquia. Esta versão é uma implementação incluída no *kernel* da especificação Mobilidade IPv6 que não contém suporte IPSec.

No entanto, todas as versões posteriores à 2.0 (inclusivé), tiveram a cooperação de outro projecto, o USAGI/WIDE. Os projectos GO-Core e USAGI desenvolveram um conjunto de extensões para a pilha protocolar IPv6 do Linux, de forma a suportar a mobilidade IP. O objectivo principal é tornar estas modificações pertencentes ao *kernel*.

O MIPL é baseado na RFC 3775 [59], e tem como última a versão 2.0.2, datada de 14 de Junho de 2006. As versões anteriores à 2.0 não suportavam IPSec.

Em [72] é possível consultar toda a informação referente ao MIPL.

O projecto USAGI (UniverSAl playGround for Ipv6) é composto por voluntários Japoneses que pretendem contribuir para a comunidade Linux e IPv6 através de melhoramentos da pilha protocolar IPv6 para sistemas Linux.

6.5.2.2 FMIPv6

O FMIPv6 é considerado com um complemento ao MIPv6. Como foi explicado em capítulos anteriores, o MIPv6 permite que um terminal móvel se mova entre diversas redes sem perder a conectividade à rede/Internet. Este projecto tem como objectivo reduzir o tempo em que o terminal móvel se encontra incontactável.

Durante a transição existe um período de tempo em que o terminal móvel deixa de receber e enviar pacotes. O atraso de *handover*, resultante de procedimentos *standard* do MIPv6, nomeadamente de detecção de movimento, configuração do novo *Care-of-Address* e *Binding Update*, é por vezes inaceitável para aplicações de tempo real como aplicações de VoIP.

Este projecto visa a implementação do protocolo FMIPv6 especificado na RFC 4068 [60] em plataformas Linux.

Em [73] é possível consultar toda a informação referente ao FMIPv6.org.

6.5.3 Wireshark - Analisador de rede

O Wireshark é um analisador de protocolos, ou “*packet sniffer*”, utilizado para fazer *troubleshooting* de redes, análise, desenvolvimento de protocolos/aplicações e também para fins educacionais.

O funcionamento do Wireshark é semelhante ao tcpdump³, com a diferença de ter um interface mais amigável, pois possui um ambiente gráfico bastante apelativo e possui funções de ordenação e filtragem da informação. Com esta aplicação é possível visualizar todo o tráfego que circula na rede, ao colocar a placa de rede em modo promíscuo.

Para além destas funções, a aplicação permite ainda obter informações acerca do encapsulamento e de diversos campos específicos de cada protocolo, já que este analisador compreende a estrutura dos diversos protocolos de rede e consegue interpretar o seu significado.

O Wireshark é uma aplicação disponibilizada com a licença GPL *opensource* e utiliza a plataforma GTK+. Está disponível para ambientes Windows, Unix e derivados deste último incluindo Linux, Solaris, FreeBSD, NetBSD, OpenBSD e MAC OS X.

O Wireshark utiliza a biblioteca pcap⁴ para realizar a captura dos pacotes, sendo assim necessário que a rede suporte esta API.

³ tcpdump é uma ferramenta de depuração que corre em linha de comandos.

⁴ pcap é uma API para a captura de pacotes.

6.5.4 Virtualização

O VMware é um *software* que permite a existência de várias imagens de sistemas operativos na mesma máquina física. Ele permite a instalação e utilização de um sistema operacional dentro de outro, dando suporte real ao *software* de outros sistemas.

A aplicação VMware foi desenvolvida pela VMware Inc., localizada em Palo Alto, California, Estados Unidos. O nome, VMware, surge de um jogo de palavras com Virtual Machine (máquina virtual), que é o nome técnico dado a um sistema operativo que corre sobre o VMware.

Ao contrário de muitos mitos divulgados pela Internet, o VMware não é um emulador. Este *software* desce a um nível mais baixo, onde o processador chega por vezes a executar o código da máquina virtual.

O produto disponibilizado pelo VMware que será utilizado para a virtualização neste projecto será o VMware Server. Este é de licença livre ou seja, está disponível mediante um registo, sem custos, na página dos produtos VMware [74], e permite criar máquinas virtuais, editá-las e reproduzi-las.

6.5.5 Aplicações para testar a mobilidade

6.5.5.1 MIPv6 Tester

O MIPv6 Tester (Figura 6-4) é uma aplicação que simplifica os testes de mobilidade em ambientes IPv6. Isto é conseguido através de uma ligação bidireccional TCP e duas ligações unidireccionais UDP entre duas máquinas na rede, designadas por cliente e servidor. A aplicação permite testar a conectividade entre as máquinas e medir o tempo em que a ligação está interrompida.

Esta aplicação está disponível sobre uma licença GNU em [75].

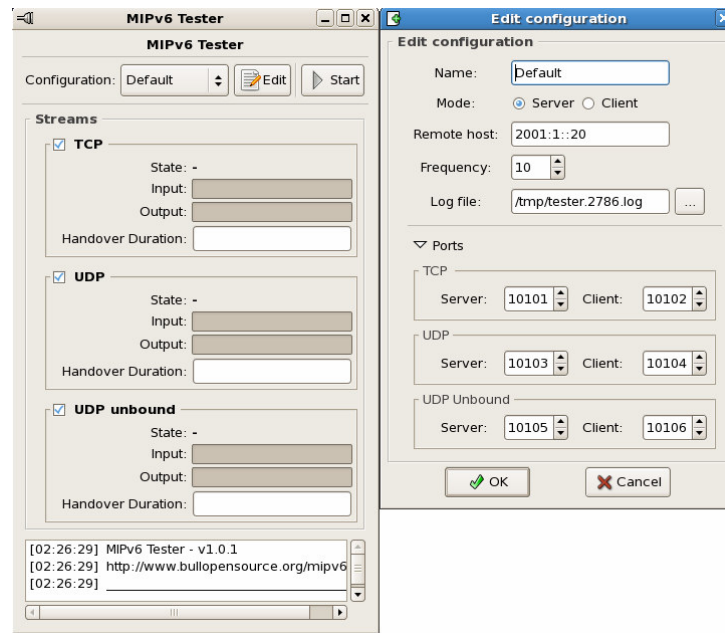


Figura 6-4 – MIPv6 Tester

6.5.5.2 MIPv6 Analyzer

O MIPv6 Analyzer é uma ferramenta *web* desenvolvida por Paolo Cavone [70], para o estudo de mobilidade IPv6 em Linux. Baseia-se na base temporal da informação fornecida pelos ficheiros de *log* do tcpdump, em particular, obtido através do seguinte filtro:

- `tcpdump -i <interface> -vv ipv6;`

A aplicação permite analisar ficheiros de *log* através da opção:

- Summary >> New Log.

A Figura 6-5 mostra a imagem da janela principal da aplicação *web*, que está acessível em [70].

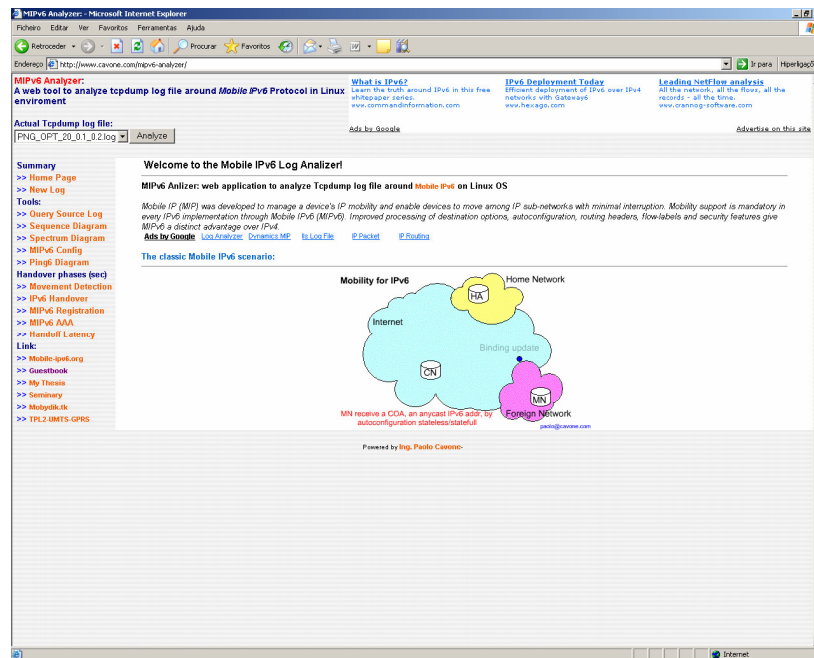


Figura 6-5 – MIPv6 Analyzer

6.6 Objectivos dos Cenários

O cenário anteriormente apresentado tem um conjunto de objectivos de estudo. Como já foi anteriormente referido, este projecto visa o estudo e análise de redes IPv6 em relação à mobilidade.

Será efectuada a comparação entre *handover* e *fast handover*, a nível de perda de pacotes e duração do tempo do processo de *handover*. Os estudos teóricos indicam que o *fast handover* apresenta melhorias significativas, a todos os níveis, em comparação com o *handover* tradicional. [5] [11]

7 Testes

Neste capítulo são apresentados os testes que foram realizados em ambos os cenários referidos no capítulo anterior.

Serão expostos os resultados dos testes relativos à deslocação do terminal móvel, o *debug* do respectivo *daemon* e as mensagens capturadas durante o *handover*. Serão também analisados os resultados obtidos em cada cenário e os principais problemas que ocorreram durante a fase de testes.

É importante referir que para testar a mobilidade, além do respectivo *daemon*, também é necessário recorrer à utilização de um outro *daemon* (*radvd*) que controla toda a informação referente aos *routers* envolvidos na mobilidade. Aqui é possível configurar diversos parâmetros, de entre os quais se destacam os valores correspondentes à frequência de envio de mensagens *Router Advertisement*, o *MinRtrAdvInterval* e o *MaxRtrAdvInterval*. Na RFC 3775 [59] os valores mínimos especificados correspondem a 0.03 e 0.07 segundos. Quanto menores forem estes valores, menor será a periodicidade de envio de *Router Advertisements*, logo a deslocação será detectada mais rapidamente, mas será maior o *overhead* gerado na rede.

Para a realização dos testes foi utilizado o Fedora Core 6, com o *kernel* 2.6.16, com todas as *packages* e *patches* necessários à mobilidade IPv6, quer em MIPv6 [72] quer em FMIPv6 [73].

7.1 Cenário MIPv6

A Figura 7-1 apresenta o cenário de teste utilizado para a realização de testes sobre o *handover* em MIPv6.

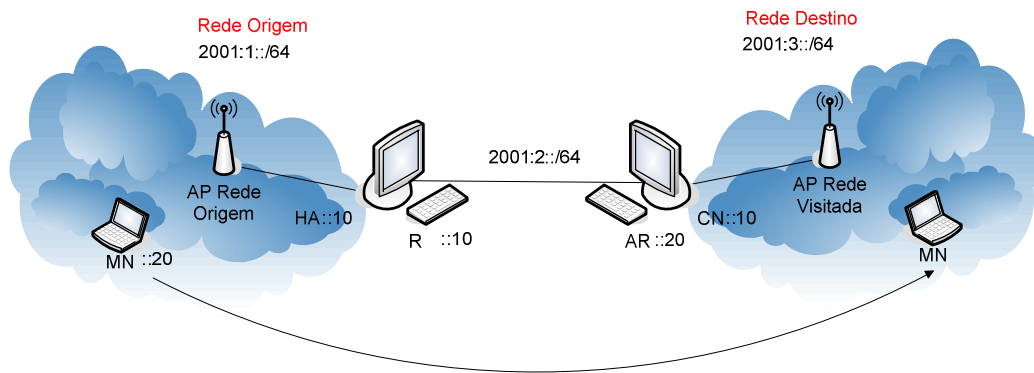


Figura 7-1 – Cenário de teste MIPv6

7.1.1 Debug do daemon mip6d

O *daemon* do MIPv6, responsável pelo processo de mobilidade, apresenta toda a informação relativa a este processo, em forma de *debug*. Todo o processo pode ser verificado através da troca de mensagens entre o terminal móvel, o *Home Agent* e o nó correspondente, pode ser verificado.

A Figura 7-2 apresenta a primeira mensagem trocada entre o terminal móvel e o *Home Agent*. Esta mensagem indica que o terminal móvel já se encontra na rede visitada (1) e que lhe foi atribuído um novo *Care-of-Address* (2). O terminal móvel pretende registar este endereço no seu *Home Agent* para que a comunicação passe a ser realizada directamente para esse endereço (3). Por sua vez, na Figura 7-3 o *Home Agent* responde com uma mensagem *Binding Acknowledgement* a confirmar o registo do novo *Care-of-Address*. A partir deste momento, todos os pacotes destinados ao endereço *Home Address* do terminal móvel serão reencaminhados para o seu novo endereço *Care-of-Address*, através de um túnel criado entre estes dois pontos.

```

mn move: in foreign net 1
mv_hoa: move HoA 2001:1:0:0:0:0:20/128 from iface 6 to 9
mn_send_home_bu: 682
mn_get_home_lifetime: CoA lifetime 2591998 s, HoA lifetime 581 s, BU lifetime 580 s
process_first_home_bu: New bule for HA
bul_add: Adding bule
== BUL_ENTRY ==
Home address 2001:1:0:0:0:0:20 2
Care-of address 2001:3:0:0:211:21ff:fec2:eae
CN address 2001:1:0:0:0:0:10
lifetime = 580, delay = 1500
flags: IP6 MH BU HOME IP6 MH BU ACK
mn_send_home_bu: New bule for HA
mh_send: sending MH type 5
from 2001:1:0:0:0:0:20
to 2001:1:0:0:0:0:10 3
mh send: local CoA 2001:3:0:0:211:21ff:fec2:eae

```

Figura 7-2 – Binding Update enviado pelo terminal móvel ao Home Agent

```

xfrm_ha_init: Adding policies and states for HA
ha_if_addr_setup: Joined anycast group 2001:1:0:0:fdff:ffff:ffff:fffe on iface 4
mh_bu_parse: Binding Update Received
ndisc_do_dad: Dad success
tunnel add: created tunnel ip6tnl1 (6) from 2001:1:0:0:0:0:10 to 2001:3:0:0:211:21ff:fe3d:128e
mh_send_ba: status 0

```

Figura 7-3 – Binding Update recebido no Home Agent

Depois de registrado o endereço no *Home Agent*, o terminal móvel envia um *Binding Update* ao nó correspondente a avisá-lo do seu novo endereço *Care-of-Address*, tal como mostra a Figura 7-4.

```

mh_bu_parse: Binding Update Received
mh_bu_parse: src 2001:1:0:0:0:0:20
mh_bu_parse: coa 2001:3:0:0:211:21ff:fe3d:128e
mh_send_ba: status 0
mh_send: sending MH type 6
from 2001:3:0:0:0:0:10
to 2001:1:0:0:0:0:20

```

Figura 7-4 – Binding Update enviado pelo terminal móvel ao nó correspondente

A Figura 7-5 apresenta a eliminação do túnel criado entre o *Home Agent* e o novo *Care-of-Address* do terminal móvel. Esta eliminação ocorre quando o terminal móvel regressa à rede origem.

```

mh_send: local CoA 2001:3:0:0:211:21ff:fec2:eae
mn_pol_ext_cleanup:
bul_delete: Deleting bule
== BUL_ENTRY ==
Home address      2001:1:0:0:0:0:0:20
Care-of address   2001:1:0:0:0:0:0:20
CN address        2001:1:0:0:0:0:0:10
lifetime = 0, delay = 32000
flags: IP6_MH_BU_HOME IP6_MH_BU_ACK
__tunnel_del: tunnel ip6tnl1 (9) from 2001:3:0:0:211:21ff:fec2:eae to 2001:1:0:0:0:0:0:10 user count
decreased to 0
__tunnel_del: tunnel deleted

```

Figura 7-5 – Eliminação do túnel criado entre o Home-Agent e o terminal móvel

7.1.2 Tráfego icmp

Nesta secção são apresentadas a Figura 7-6 e a Figura 7-7, que ilustram o resultado do comando *ping6* em diferentes momentos. É possível verificar que o handover não tem um tempo definido ou seja, a sua duração não é constante.

```

64 bytes from 2001:3::10: icmp_seq=24 ttl=63 time=13.2 ms
64 bytes from 2001:3::10: icmp_seq=25 ttl=63 time=32.1 ms
64 bytes from 2001:3::10: icmp_seq=26 ttl=63 time=70.3 ms
64 bytes from 2001:3::10: icmp_seq=27 ttl=63 time=11.5 ms
ping: sendmsg: Invalid argument
ping: sendmsg: Invalid argument
ping: sendmsg: Invalid argument
From 2001:1::20 icmp_seq=31 Destination unreachable: Address unreachable
From 2001:1::20 icmp_seq=32 Destination unreachable: Address unreachable
64 bytes from 2001:3::10: icmp_seq=33 ttl=63 time=3.46 ms
64 bytes from 2001:3::10: icmp_seq=34 ttl=63 time=3.04 ms
64 bytes from 2001:3::10: icmp_seq=35 ttl=63 time=3.14 ms
64 bytes from 2001:3::10: icmp_seq=36 ttl=63 time=3.03 ms

```

Figura 7-6 – Comando ping6 (I)

```

64 bytes from 2001:3::10: icmp_seq=173 ttl=63 time=2.82 ms
64 bytes from 2001:3::10: icmp_seq=174 ttl=63 time=2.60 ms
64 bytes from 2001:3::10: icmp_seq=175 ttl=63 time=2.58 ms
64 bytes from 2001:3::10: icmp_seq=176 ttl=63 time=2.68 ms
ping: sendmsg: Invalid argument
ping: sendmsg: Invalid argument
ping: sendmsg: Invalid argument
From 2001:1::20 icmp_seq=180 Destination unreachable: Address unreachable
64 bytes from 2001:3::10: icmp_seq=181 ttl=63 time=6.83 ms
64 bytes from 2001:3::10: icmp_seq=182 ttl=63 time=3.85 ms
64 bytes from 2001:3::10: icmp_seq=183 ttl=63 time=3.11 ms
64 bytes from 2001:3::10: icmp_seq=184 ttl=63 time=3.27 ms

```

Figura 7-7 – Comando ping6 (II)

7.1.3 Captura de pacotes

Em relação à captura de pacotes, as figuras seguintes apresentam a troca de mensagens entre o terminal móvel, o *Home Agent* e o nó correspondente.

A Figura 7-8 e a Figura 7-9 apresentam as mensagens de iniciação do processo de *handover*, *Binding Update*, *Home Teste Init*, *Care-of Test Init* e *Care-of Test*.

No. -	Time	Source	Destination	Protocol	Info
3625	68.050005	2001:3::211:21ff:f	2001:1::10	MIPv6	Binding Update
3760	73.554311	2001:1::20	2001:3::10	MIPv6	Home Test Init
3834	81.464598	2001:3::211:21ff:f	2001:3::10	MIPv6	Care-of Test Init
3835	81.467171	2001:3::10	2001:3::211:21ff:f	MIPv6	Care-of Test

Figura 7-8 – Captura de mensagens do processo de *handover* do MIPv6

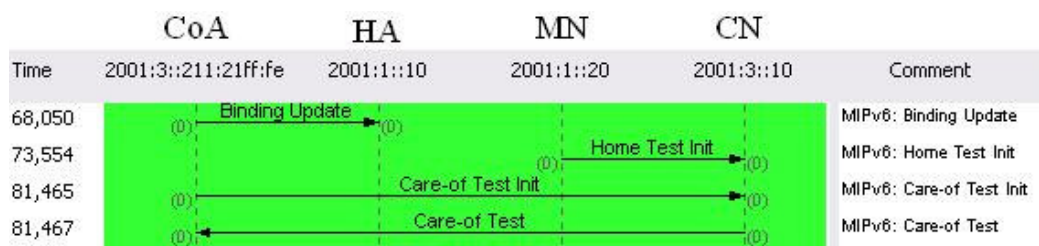


Figura 7-9 – Análise gráfica da captura de mensagens do processo de *handover* do MIPv6

A Figura 7-8 apresenta todos os detalhes da mensagem *Binding Update* que o terminal móvel envia ao *Home Agent*. Em (1) podemos verificar que esta mensagem foi enviada quando o terminal móvel se encontrava na rede visitada, embora continue a ter o seu *Home Address* (2). Esta mensagem é enviada quando o terminal móvel pede ao *Home Agent* que registe o seu novo *Care-of-Address* (3) (4).

No.	Time	Source	Destination	Protocol	Info
197	176.19419	::	ff02::1:fffc2:eae	ICMPv6	Neighbor solicitation
198	177.19457	2001:3::211:21ff:f	2001:1::10	MIpV6	Binding Update
200	177.20209	fe80::211:21ff:fec	ff02::16	ICMPv6	Multicast Listener Report Message v2

Internet Protocol Version 6 Version: 6 Traffic class: 0x00 Flowlabel: 0x00000 Payload length: 56 Next header: IPv6 destination option (0x3c) Hop limit: 64 Source address: 2001:3::211:21ff:fec2:eae Destination address: 2001:1::10	1
Destination option Header Next header: Mobile IPv6 (0x87) Length: 2 (24 bytes) PadN: 4 bytes option Type: 201 (0xc9) – Home Address option option Length : 16 Home Address : 2001:1::20 (2001:1::20)	2
Mobile IPv6 / Network Mobility Payload protocol: IPv6 no next header (0x3b) Header length: 3 (32 bytes) Mobility Header Type: Binding Update (5) Reserved: 0x00 Checksum: 0xf7fe	
Binding Update Sequence number: 28923 1... = Acknowledge (A) flag: Binding Acknowledgement requested .1.. = Home Registration (H) flag: Home Registration ..0. = Link-Local Compatibility (L) flag: No Link-Local Address Compatibility ...0 = Key Management Compatibility (K) flag: No Key Management Mobility Compatibility 0... = MAP Registration Compatibility (M) flag: No MAP Registration Compatibility Lifetime: 145 (580 seconds)	3
Mobility Options PadN: 2 bytes Alternate care-of address: 2001:3::211:21ff:fec2:eae (2001:3::211:21ff:fec2:eae)	4

Figura 7-10 – Mensagem Binding Update enviada pelo terminal móvel ao Home Agent

Na Figura 7-11 podemos observar a confirmação do pedido de registo do *Care-of-Address*, por parte do *Home Agent*.

No. -	Time	Source	Destination	Protocol	Info
280	198.42659	2001:1::10	ff02::1:ff00:20	ICMPv6	Neighbor solicitation
281	198.42663	2001:1::20	2001:1::10	ICMPv6	Neighbor advertisement
282	198.42910	2001:1::10	2001:1::20	MIPv6	Binding Acknowledgement
283	198.42924	2001:1::20	ff02::1	ICMPv6	Neighbor advertisement
284	198.43959	::	ff02::1:ffc2:ea	ICMPv6	Neighbor solicitation

[-] Frame 282 (70 bytes on wire, 70 bytes captured)
[-] Ethernet II, Src: Vmware_a8:60:38 (00:0c:29:a8:60:38), Dst: Cisco_c2:0e:ae (00:11:21:c2:0e:ae)
[-] Internet Protocol Version 6
Version: 6
Traffic class: 0x00
Flowlabel: 0x00000
Payload length: 16
Next header: Mobile IPv6 (0x87)
Hop limit: 64
Source address: 2001:1::10
Destination address: 2001:1::20
[-] Mobile IPv6 / Network Mobility
Payload protocol: IPv6 no next header (0x3b)
Header length: 1 (16 bytes)
Mobility Header Type: Binding Acknowledgement (6)
Reserved: 0x00
Checksum: 0x0c32
[-] Binding Acknowledgement
Status: Binding Update accepted (0)

Figura 7-11 – Mensagem Binding Acknowledgement enviada pelo Home Agent ao terminal móvel

As figuras anteriores apresentam as mensagens mais importantes do *handover* em MIPv6.

7.1.4 Handover

Para determinar o tempo que envolve o processo de *handover*, foi utilizada a aplicação MIPv6 Tester. Como já foi anteriormente referido, esta aplicação gera tráfego e determina o tempo em que uma ligação está interrompida, neste caso o tempo de *handover* que decorre durante a transição do terminal móvel entre redes.

O funcionamento da aplicação passa pela definição de um cliente e um servidor, entre os quais serão criadas ligações TCP e UDP. Também é possível definir o número de pacotes por segundo (frequência) que serão enviados.

Com o MIPv6 Tester configurado no terminal móvel como servidor e no *Home Agent* como cliente, e com uma frequência de 100 pacotes por segundo obtiveram-se os valores de *handover* apresentados na Figura 7-12.

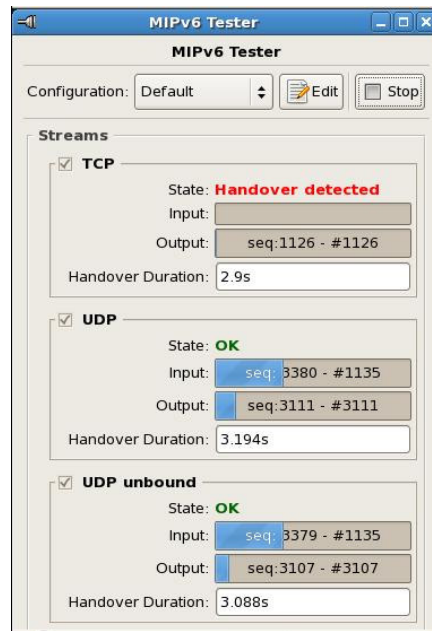


Figura 7-12 – MIPv6 Tester

Após realizados alguns testes com a configuração acima referida, e com os valores de *Router Advertisements* mínimo e máximo configurados a 1 e 3, respectivamente, obtiveram-se os resultados expostos na Tabela 1.

Teste	Origem → Visitada	Visitada → Origem
1	7.931	3.722
2	4.348	2.873
3	5.742	4.025
4	11.156	3.819
5	6.936	3.019
Média	7.223	3.492

Tabela 1 – Tempo de handover no MIPv6 com Router Advertisements a 1 e 3

Para verificar qual a influência dos *Router Advertisements* no processo de *handover*, foram alterados os seus valores para 0.03 e 0.07 (valores mínimos especificados na

RFC 3775 [59]) e realizados os mesmos testes. A Tabela 2 apresenta os resultados obtidos nos cinco testes realizados.

Teste	Origem → Visitada	Visitada → Origem
1	4.925	3.768
2	4.214	3.160
3	3.980	2.983
4	4.461	3.317
5	3.842	3.094
Média	4.284	3.264

Tabela 2 – Tempo de handover no MIPv6 com Router Advertisements a 0.03 e 0.07

Ao analisar a Figura 7-13 verifica-se que o *handover* no sentido da rede origem para a rede visitada continua ser mais demorado do que no sentido contrário. No entanto é possível concluir que o valor dos *Router Advertisements* influenciam o tempo de *handover*. Isto deve-se ao facto das mensagens *Router Advertisement* estarem a ser enviadas com maior frequência, logo o *handover* é detectado mais rapidamente, o que reduz significativamente a sua duração.

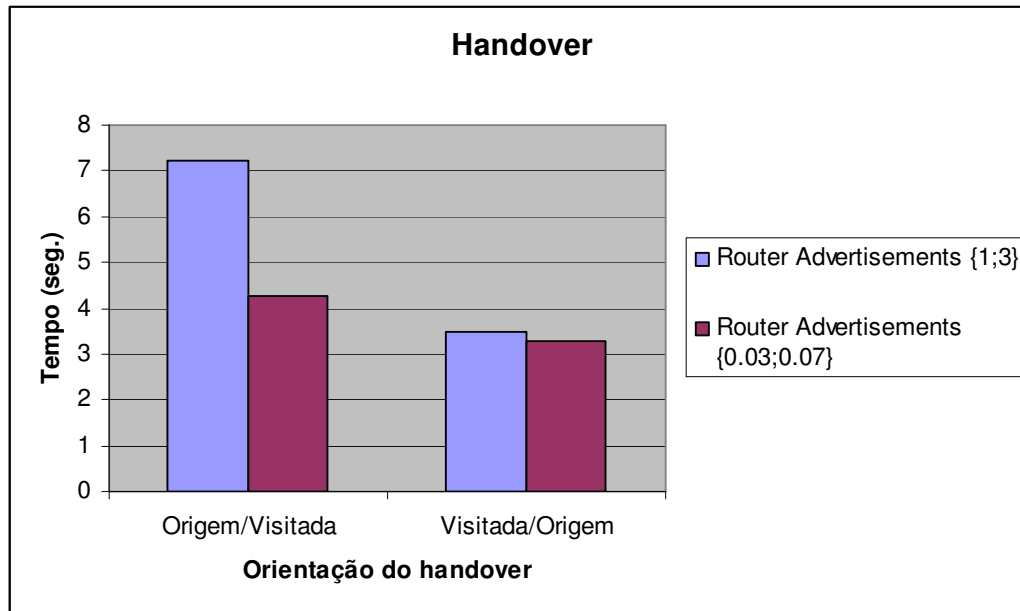


Figura 7-13 – Valores médios de handover para diferentes valores de Router Advertisements

7.1.5 Resultados obtidos

Na implementação deste cenário pretendeu-se estudar o processo de *handover* no protocolo MIPv6. Retirados os resultados de diversos testes realizados é possível concluir que a duração do processo de *handover* está relacionada com o tempo de envio de mensagens *Router Advertisement*. Quando estas estão configuradas no seu valor mínimo, o tempo de *handover* é visivelmente menor. Isto deve-se ao facto de serem enviados mais *Router Advertisements*, logo o terminal móvel detecta mais rapidamente que se está a deslocar e realiza o *handover*.

É possível verificar que o *handover* no sentido da rede origem é mais rápido. Isto deve-se ao facto de não ser necessário efectuar todo o processo de registo de endereços e a comunicação com o *Home Agent* e nó correspondente. Quando o terminal móvel se desloca para a rede origem, o único processo que ocorre é a eliminação do túnel que foi previamente criado para estabelecer a comunicação com o novo *Care-of-Address*.

7.1.6 Problemas detectados

Em relação aos problemas detectados, apenas os testes realizados com a aplicação *web MIPv6 Analyzer* tornaram-se infrutíferos dado que a aplicação *web* foi desenvolvida com base na versão MIPL 1.1, não estando ainda disponível para a versão utilizada do MIPL, a versão mais recente MIPL 2.0.2. Ao retirar o resultado do comando *tcpdump*, nesta última versão, e ao colocá-lo como entrada da aplicação *web MIPv6 Analyzer*, verificou-se que a aplicação não estava preparada para o mesmo.

7.2 Cenário FMIPv6

A Figura 7-14 apresenta o cenário de teste utilizado para a realização de testes sobre o *handover* em MIPv6.

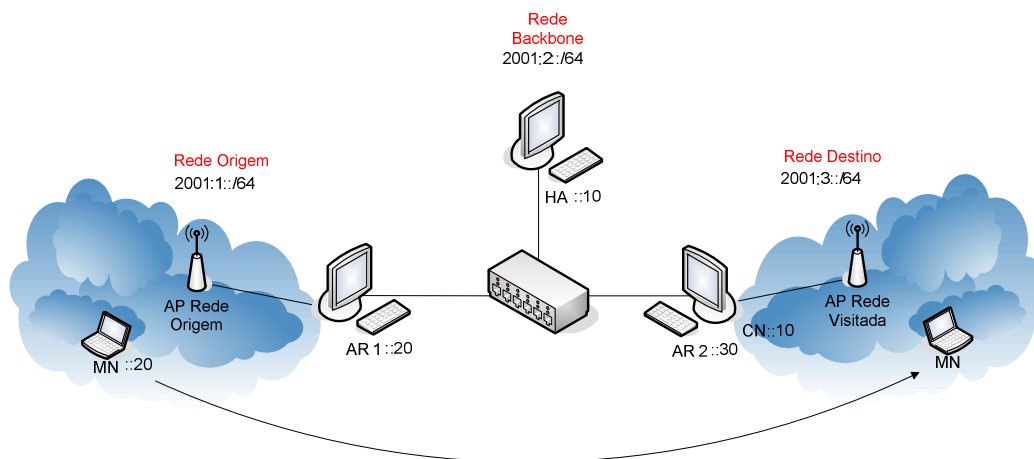


Figura 7-14 – Cenário FMIPv6

7.2.1 Debug do daemon fmip6

O *daemon fmip6* é utilizado com a finalidade de acompanhar a transição do terminal móvel entre as diferentes redes de acesso. No entanto, também é muito útil pelo facto de que apresenta mensagens de *debug* ao longo de todo este procedimento. Com estas mensagens é possível acompanhar o que ocorre na rede, desde o envio de mensagens de controlo de *handover*, tal como *Router Solicitation Proxy* e *Proxy Router*

Advertisement, até à aquisição dos endereços *Care-of-Address*, não esquecendo o envio de *Router Advertisements* entre os *routers* de acesso e o terminal móvel.

É importante referir que quando o terminal móvel se encontra na rede origem, antes de efectuar a mobilidade, já tem atribuído um endereço *Care-of-Address* da rede origem. Este endereço é-lhe atribuído através dos *Router Advertisements* que têm origem no *router* da rede origem.

Na Figura 7-15, destacado com (1), está saliente o instante em que o *daemon* inicia o processo de análise da rede. A partir deste instante é possível visualizar a actividade que ocorre no terminal móvel, quer ele esteja parado ou em movimento. Em (2) e (3) estão destacados, respectivamente, o instante em que o terminal móvel recebe uma mensagem *Router Advertisement* e o instante em que lhe é atribuído um *Care-of-Address* da rede origem.

```
17:37:29.529467|main.242: fmip6-mn v1.0-rc1 started
17:37:29.529839|fmip6_mn_main.1712: FMIPv6 MN Main.
17:37:29.531641|homng_init.747: Starting HOMNG INIT!
17:37:29.531948|process_nlmsg.1047: Movement detection ignored event on a non primary iface!
17:37:29.532025|process_new_inet6_iface.852: adding iface eth1 (4)
17:37:29.532549|md_link_up.718: link up on iface eth1 (4)
17:37:29.532969|__md_discover_router.688: discover link on iface eth1 (4)
17:37:29.533784|fmip6_mn_mh_listen.200: Starting mn_mh_listen!
17:37:32.590791|md_recv_ra.1883: received RA from fe80:0:0:0:250:fcff:fe40:ed46 on iface 4
17:37:32.591095|md_check_default_router.1748: looking for existing routers on iface eth1 (4)
17:37:32.591143|md_change_default_router.1594: add new router fe80:0:0:0:250:fcff:fe40:ed46 on inte
17:37:32.591277|__md_new_link.729: new link on iface eth1 (4)
17:37:32.591328|md_update_router_stats.1421: adding default route via fe80:0:0:0:250:fcff:fe40:ed46
17:37:32.591375|md_update_router_stats.1423: !!!!!!!!!!!!!UPDATING NEIGHBOUR CACHE!!!!!!
17:37:32.591512|md_update_router_stats.1438: adding prefix 2001:1:0:0:0:0:0:64
17:37:32.590951|process_addr.662: .
17:37:32.592046|process_new_addr.585: new address 2001:1:0:0:211:21ff:fec2:eae on iface 4
17:37:32.592103|update_coa.541: adding CoA 2001:1:0:0:211:21ff:fec2:eae on iface eth1 (4)
```

Figura 7-15 – Arranque do daemon fmip6 do terminal móvel

No *Router Advertisement* é possível observar que este foi enviado com o endereço *link-local* da respectiva placa de rede, tal como mostra a Figura 7-16.

```

eth1      Link encap:Ethernet  HWaddr 00:50:FC:40:ED:46
          inet addr:10.0.0.2  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::250:fcff:fe40:ed46/64 Scope:Link
          inet6 addr: 2001::1::10/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:426 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10099 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:52941 (51.7 KiB)  TX bytes:1191654 (1.1 MiB)
          Interrupt:11 Base address:0x8800

```

Figura 7-16 – Interface *eth1* do router de acesso AR1

A Figura 7-17 mostra o início do processo de análise da rede no *router* AR1, destacado com (1), e a troca de mensagens entre o AR1 e o terminal móvel. Esta troca, visualizada em (2) e (3), corresponde à recepção de uma mensagem *RtSolPr* e ao envio de uma mensagem *PrRtAdv*. Esta mensagem é enviada como resposta a um *RtSolPr*. O seu conteúdo contém informação acerca de todos os *routers* vizinhos.

```

22:37:37.122733|main.243: fmip6-ar v1.0-rc1 started
22:37:37.124195|fmip6_xfrm_pol_add.444: Adding Home Address handling
22:37:37.130569|fmip6_ar_icmp6_listen.172: Starting fmip6_ar_icmp6_listen 1
22:37:37.130693|fmip6_ar_mh_listen.240: Starting fmip6_ar_mh_listen
22:40:10.646287|fmip6_ar_icmp6_listen.195: Received a message:
        type: 2
        from: 2001::1::211:21ff:fec2:eae
        to: ff02::2
        iif: eth1
22:40:10.646488|fmip6_ar_recv_rtsolpr.315: RtSolPr Received 2
22:40:10.646772|do_ni_hover.1821: Doing a Network Initiated Handover
22:40:10.646829|do_ni_hover.1822: MN'll have to move to AP with LLA 00:40:96:54:1A:47
22:40:10.647375|fmip6_ar_send_prtradv.672: PrRtAdv sent 3
22:40:10.647467|fmip6_ar_recv_rtsolpr.451: Exit!
22:57:08.133039|graceful_exit.109: =====

```

Figura 7-17 – Arranque do daemon *fmip6* do router de acesso AR1

7.2.2 Tráfego icmp

Como já foi acima referido, os parâmetros *MinRtrAdvInterval* e *MaxRtrAdvInterval*, presentes no ficheiro de configuração *radvd.conf* influenciam o *handover*. Aqui serão apresentadas duas figuras que mostram a diferença no resultado do comando *ping6* para valores de *Router Advertisements* de {0.03;0.07} e {1;3}.

A Figura 7-18 mostra que foram perdidos dois pacotes durante a ligação entre o terminal móvel e o nó correspondente, com os valores de *MinRtrAdvInterval* e *MaxRtrAdvInterval* configurados com 1 e 3 respectivamente.

```
64 bytes from 2001:3::10: icmp_seq=16 ttl=63 time=2.53 ms
64 bytes from 2001:3::10: icmp_seq=17 ttl=63 time=2.55 ms
64 bytes from 2001:3::10: icmp_seq=18 ttl=63 time=2.64 ms
64 bytes from 2001:3::10: icmp_seq=19 ttl=63 time=2.70 ms
64 bytes from 2001:3::10: icmp_seq=22 ttl=64 time=16.0 ms
64 bytes from 2001:3::10: icmp_seq=23 ttl=64 time=2.96 ms
64 bytes from 2001:3::10: icmp_seq=24 ttl=64 time=2.44 ms
```

Figura 7-18 – Comando ping6 com dois pacotes perdidos

Já a Figura 7-19, onde os parâmetros foram configurados com os valores mínimos, comprova que apenas se perdeu um pacote.

```
64 bytes from 2001:3::10: icmp_seq=132 ttl=63 time=2.59 ms
64 bytes from 2001:3::10: icmp_seq=133 ttl=63 time=2.28 ms
64 bytes from 2001:3::10: icmp_seq=134 ttl=63 time=2.50 ms
64 bytes from 2001:3::10: icmp_seq=135 ttl=63 time=2.33 ms
64 bytes from 2001:3::10: icmp_seq=137 ttl=64 time=15.0 ms
64 bytes from 2001:3::10: icmp_seq=138 ttl=64 time=4.39 ms
64 bytes from 2001:3::10: icmp_seq=139 ttl=64 time=2.15 ms
```

Figura 7-19 – Comando ping6 com um pacote perdido

7.2.3 Captura de pacotes

Tal como foi referido na secção anterior, os valores dos *Router Advertisements* influenciam a frequência de envio destas mensagens. As figuras seguintes apresentam a captura de pacotes no momento do *handover*, com diferentes valores de *MinRtrAdvInterval* e *MaxRtrAdvInterval*.

Na Figura 7-20 é possível verificar em (1) o envio de um *ping6* (*Echo request*) para o nó correspondente, a partir do *Care-of-Address* da rede origem, e a respectiva resposta. Entretanto o terminal móvel desloca-se para a rede visitada, e deixa de receber *Echo reply*, até que seja finalizado o *handover* e lhe seja atribuído um novo *Care-of-Address* na rede visitada (2). Neste momento o *ping6* volta a ser estabelecido ou seja, volta a existir comunicação entre o terminal móvel e o nó correspondente. É importante referir que neste momento os valores mínimo e máximo dos *Router Advertisements* estão com 1 e 3.

147	67.174401	fe80::250:fcff:fe40:ed46	ff02::1	ICMPv6 Router advertisement
148	68.059903	2001:1::211:21ff:fec2:eae	2001:3::10	ICMPv6 Echo request
149	68.062427	2001:3::10	2001:1::211:21ff:fec2:eae	ICMPv6 Echo reply
150	68.696196	fe80::250:fcff:fe40:ed46	ff02::1	ICMPv6 Router advertisement
151	69.059277	2001:1::211:21ff:fec2:eae	2001:3::10	ICMPv6 Echo request
152	70.059357	2001:1::211:21ff:fec2:eae	2001:3::10	ICMPv6 Echo request
153	70.116054	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
160	71.309760	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
162	72.063416	2001:3::211:21ff:fec2:eae	ff02::1:ff00:10	ICMPv6 Neighbor solicitation
163	72.070613	2001:3::10	2001:3::211:21ff:fec2:eae	ICMPv6 Neighbor advertisement
164	72.070641	2001:3::211:21ff:fec2:eae	2001:3::10	ICMPv6 Echo request
165	72.072980	2001:3::10	2001:3::211:21ff:fec2:eae	ICMPv6 Echo reply
166	72.083994	2001:3::211:21ff:fec2:eae	2001:3::10	ICMPv6 Echo request
167	72.087008	2001:3::10	2001:3::211:21ff:fec2:eae	ICMPv6 Echo reply

Figura 7-20 – Captura de pacotes: *MinRtrAdvInterval 1 e MaxRtrAdvInterval 3*

Já a Figura 7-21 mostra o *handover* nas mesmas condições que anteriormente, mas com os valores de *Router Advertisements* no mínimo, respectivamente 0.03 e 0.07.

7130	364.503130	fe80::250:fcff:fe40:ed46	ff02::1	ICMPv6 Router advertisement
7131	364.599480	2001:1::211:21ff:fec2:eae	2001:3::10	ICMPv6 Echo request
7132	364.601781	2001:3::10	2001:1::211:21ff:fec2:eae	ICMPv6 Echo reply
7133	364.620281	fe80::250:fcff:fe40:ed46	ff02::1	ICMPv6 Router advertisement
7134	364.659766	fe80::250:fcff:fe40:ed46	ff02::1	ICMPv6 Router advertisement
7135	364.726903	fe80::250:fcff:fe40:ed46	ff02::1	ICMPv6 Router advertisement
7136	364.759464	fe80::250:fcff:fe40:ed46	ff02::1	ICMPv6 Router advertisement
7137	365.064811	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7138	365.108500	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7139	365.146472	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7140	365.181052	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7141	365.218927	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7142	365.273303	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7143	365.329295	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7144	365.371479	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7145	365.446408	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7146	365.504887	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7149	365.554920	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7150	365.591930	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7151	365.663176	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7152	365.713004	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7154	365.770666	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7155	365.816440	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7156	365.864541	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7157	365.923008	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7158	365.957396	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7159	366.013091	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7162	366.066762	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7163	366.114411	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7164	366.179140	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7165	366.216702	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7167	366.268554	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7168	366.282582	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7169	366.334998	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7170	366.362740	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7171	366.413089	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7172	366.474730	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7173	366.520754	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7174	366.600016	fe80::20c:29ff:feda:8052	ff02::1	ICMPv6 Router advertisement
7175	366.605453	2001:3::211:21ff:fec2:eae	ff02::1:ff00:10	ICMPv6 Neighbor solicitation
7176	366.625448	2001:3::10	2001:3::211:21ff:fec2:eae	ICMPv6 Neighbor advertisement
7177	366.625492	2001:3::211:21ff:fec2:eae	2001:3::10	ICMPv6 Echo request
7178	366.625736	2001:3::211:21ff:fec2:eae	2001:3::10	ICMPv6 Echo request
7179	366.628866	2001:3::10	2001:3::211:21ff:fec2:eae	ICMPv6 Echo reply

Figura 7-21 – Captura de pacotes: *MinRtrAdvInterval 0.03 e MaxRtrAdvInterval 0.07*

No Wireshark também é possível analisar detalhadamente os pacotes que são capturados. A Figura 7-22 apresenta os detalhes correspondentes ao pacote ICMPv6, capturado no *router* de acesso AR2, com a mensagem *PrRtAdv*:

1. Indicação do tipo de mensagem: *PrRtAdv*;
2. Endereço físico do *router* de acesso que envia a mensagem;
3. Endereço físico do novo ponto de acesso;
4. Endereço físico do novo *router* de acesso. Neste caso o *router* AR1;
5. Endereço IP do novo *router* de acesso;
6. Prefixo do novo *router* de acesso.

No. -	Time	Source	Destination	Protocol	Info
1	0.000000	fe80::20c:29ff:feda:805c	ff02::1:ff00:10	ICMPv6	Neighbor solicitation
2	0.000488	2001:2::10	fe80::20c:29ff:feda:8	ICMPv6	Neighbor advertisement
3	0.000664	fe80::20c:29ff:feda:8052	2001:1::211:21ff:fec2	ICMPv6	PrRtAdv (ICMPv6 Experimental Mobility)
7	4.583352	fe80::20c:29ff:feda:805c	ff02::16	ICMPv6	Multicast Listener Report Message v2
8	4.979779	fe80::204:5aff:fe49:8482	fe80::20c:29ff:feda:8	ICMPv6	Neighbor solicitation

Internet Control Message Protocol v6

Type: 150 (PrRtAdv (ICMPv6 Experimental Mobility)) 1

Code: 1 (Network Initiated Handover trigger)

Checksum: 0xd1d9 [correct]

Subtype: Proxy Router Advertisement

Identifier: 34283

ICMPv6 options

Type: 19 (Link-layer Address)

Length: 16 bytes (2)

Option-Code: Link-layer Address of the source 2

Link-layer address: 00:0c:29:da:80:52:00:00:00:00:00:00:00:00:00:00

ICMPv6 options

Type: 19 (Link-layer Address)

Length: 16 bytes (2)

Option-Code: Link-layer Address of the New Access Point 3

Link-layer address: 00:40:96:54:8e:c6:00:00:00:00:00:00:00:00:00:00

ICMPv6 options

Type: 19 (Link-layer Address)

Length: 16 bytes (2)

Option-Code: Link-layer Address of the NAR 4

Link-layer address: 00:50:fc:40:ed:46:00:00:00:00:00:00:00:00:00:00

ICMPv6 options

Type: 17 (IP Address)

Length: 24 bytes (3)

Option-Code: NAR's IP address 5

Prefix length: 64

IPv6 Address: 2001:1::10

ICMPv6 options

Type: 18 (New Router Prefix Information) 6

Length: 24 bytes (3)

Option-Code: 0

Prefix length: 64

Prefix: 2001:1::

Figura 7-22 – Captura da mensagem *PrRtAdv* no *router* AR2

Também no *router* de acesso AR2 podemos comprovar esta troca de pacotes com o terminal móvel, Figura 7-23.

```
[root@localhost fmipv6-ar]# ./fmipv6-ar -c /usr/local/etc/fmipv6_ar.conf
02:14:24.976708|main.243: fmipv6-ar v1.0-rc1 started

*****
Current Candidate Access Router Information. (List Size=1)
*****
*****
New AP's Link Layer Address      : 00:40:96:54:8E:C6
New AP's Link Layer Address Code : 0
New Router's Link Layer Address  : 00:50:FC:40:ED:46
New Router's IP Address          : 2001:1::10/64
New Router Prefix Information    : 2001:1::/64
NCoA (MN only)                  : ::
PCoA (MN only)                  : ::
PAR (MN only)                   : ::
*****

*****
Interface list
*****
*****
Interface (index)                : eth1 (4)
Interface's prefix               : 2001:3::/64
*****
02:14:24.982702|fmipv6_xfrm_pol_add.444: Adding Home Address handling
02:14:24.991215|fmipv6_ar_icmp6_listen.172: Starting fmipv6_ar_icmp6_listen
02:14:24.991911|fmipv6_ar_mh_listen.240: Starting fmipv6_ar_mh_listen
02:28:30.683263|fmipv6_ar_icmp6_listen.195: Received a message:
                                     type: 2
                                     from: 2001:1::211:21ff:fec2:eae
                                     to: ff02::2
                                     iif: eth1
02:28:30.684272|fmipv6_ar_rcv_rtsolpr.315: RtSolPr Received
02:28:30.724757|do_ni_hover.1821: Doing a Network Initiated Handover
02:28:30.724822|do_ni_hover.1822: MN'll have to move to AP with LLA 00:40:96:54:
BE:C6
02:28:30.725832|fmipv6_ar_send_prtradv.672: PrRtAdv sent
02:28:30.725879|fmipv6_ar_rcv_rtsolpr.451: Exit!
```

Figura 7-23 – Daemon do fmipv6 no router AR2

7.2.4 Handover

O processo de *handover* de um terminal móvel pode ser determinado através da troca de mensagens entre este e os *routers* de acesso.

As figuras seguintes mostram a deslocação do terminal móvel entre a rede origem e a rede visitada, e vice-versa.

A Figura 7-24 apresenta o início e o término do processo de *handover*. O primeiro corresponde ao envio de um *Router Solicitation Proxy* para o *router* de acesso. Ao receber esta mensagem, o *router* de acesso responde com informação relativa aos seus vizinhos para onde o terminal móvel se poderá deslocar, neste caso o novo *router* de acesso AR2. Deste modo, inicia o processo através da aquisição de informação relativa ao novo *router* e à rede que lhe está associada (1). Após terminar esta fase, são eliminadas as rotas por omissão, os prefixos e o endereço *Care-of-Address* da rede origem (2) e criadas novas entradas para a rede visitada (3). Estas entradas consistem numa rota por omissão, no prefixo da rede e na atribuição do novo *Care-of-Address*.

```

17:37:33.646856|car_disc_send_rtsolpr.576: AP list size(3) - sending contents in an RtsolPr 1
17:37:33.647012|car_disc_send_rtsolpr.609: Sending RtsolPr with iov_ind=5, src=2001:1::211:21ff:fec2:eae
17:37:33.647062|car_disc_send_rtsolpr.610: dst=ff02::2
4 17:37:34.381812|md_recv_ra.1883: received RA from fe80:0:0:0:20c:29ff:feda:8052 on iface 4
17:37:34.382049|md_check_default_router.1748: looking for existing routers on iface eth1 (4)
17:37:34.382101|md_change_default_router.1594: add new router fe80:0:0:0:20c:29ff:feda:8052 on interface
17:37:34.382149|prefix_list_get.41: looking for 2001:1:0:0:0:0:0:0
17:37:34.382267|md_expire_router.337: expiring router fe80:0:0:0:250:fcff:fe40:ed46 on iface eth1 (4)
17:37:34.382320|prefix_list_get.41: looking for 2001:3:0:0:0:0:0:0
17:37:34.382367|md_expire_coa.314: expiring CoA 2001:1:0:0:211:21ff:fec2:eae on iface eth1 (4)
17:37:34.382415|md_new_link.729: new link on iface eth1 (4)
17:37:34.382458|md_free_router.260: deleting default route via fe80:0:0:0:250:fcff:fe40:ed46 2
17:37:34.382601|md_free_router.272: deleting prefix route 2001:1:0:0:0:0:0:0/64
17:37:34.382707|md_flush_coa.222: deleting CoA 2001:1:0:0:211:21ff:fec2:eae on iface 4
17:37:34.382760|md_update_router_stats.1421: adding default route via fe80:0:0:0:20c:29ff:feda:8052
17:37:34.382807|md_update_router_stats.1423: !!!!!!!!!!!UPDATING NEIGHBOUR CACHE!!!!!!
17:37:34.382937|md_update_router_stats.1438: adding prefix 2001:3:0:0:0:0:0:0/64
17:37:34.448892|fmip6_transaction_fail.532: Calling failure handler for transaction 15646
17:37:34.449039|car_disc_rtsolpr_failed.627: Entry 3
17:37:34.449083|fmip6_transaction_fail.538: Removing transaction 15646@iface(4)...
17:37:34.449128|fmip6_transaction_fail.542: Success (0 transactions left)!
17:37:34.476906|process_addr.662: .
17:37:34.476964|process_new_addr.585: new address 2001:3:0:0:211:21ff:fec2:eae on iface 4
17:37:34.477019|update_coa.541: adding CoA 2001:3:0:0:211:21ff:fec2:eae on iface eth1 (4)
    
```

Figura 7-24 – Handover entre a rede origem e a rede visitada

A Figura 7-25 mostra o processo inverso, ou seja, o *handover* no regresso à rede origem. O procedimento é análogo ao anteriormente descrito.

```

17:40:21.499310|md_router_timeout.1389: expire router fe80:0:0:0:20c:29ff:feda:8052 without NUD on inter
17:40:21.499523|md_expire_router.337: expiring router fe80:0:0:0:20c:29ff:feda:8052 on iface eth1 (4)
17:40:21.499575|md_expire_coa.314: expiring CoA 2001:3:0:0:211:21ff:fec2:eae on iface eth1 (4)
17:40:21.499622|md_discover_router.688: discover link on iface eth1 (4)
17:40:21.499984|md_trigger_movement_event.182: #####
17:40:21.500035|md_trigger_movement_event.187: strategy 0 type 6 iface eth1 (4) CoA 0:0:0:0:0:0:0:0
17:40:21.500082|md_trigger_movement_event.188: #####
17:40:22.249609|md_recv_ra.1883: received RA from fe80:0:0:0:250:fcff:fe40:ed46 on iface 4 1
17:40:22.249962|md_check_default_router.1748: looking for existing routers on iface eth1 (4)
17:40:22.250010|md_change_default_router.1594: add new router fe80:0:0:0:250:fcff:fe40:ed46 on interface
17:40:22.250123|md_new_link.729: new link on iface eth1 (4)
17:40:22.250173|md_free_router.260: deleting default route via fe80:0:0:0:20c:29ff:feda:8052 2
17:40:22.250316|md_free_router.272: deleting prefix route 2001:3:0:0:0:0:0:0/64
17:40:22.250429|md_flush_coa.222: deleting CoA 2001:3:0:0:211:21ff:fec2:eae on iface 4
17:40:22.250483|md_update_router_stats.1421: adding default route via fe80:0:0:0:250:fcff:fe40:ed46
17:40:22.250530|md_update_router_stats.1423: !!!!!!!!!!!UPDATING NEIGHBOUR CACHE!!!!!!
17:40:22.250661|md_update_router_stats.1438: adding prefix 2001:1:0:0:0:0:0:0/64 3
17:40:22.249827|process_addr.662: .
17:40:22.251190|process_new_addr.585: new address 2001:1:0:0:211:21ff:fec2:eae on iface 4
17:40:22.251304|update_coa.541: adding CoA 2001:1:0:0:211:21ff:fec2:eae on iface eth1 (4)
    
```

Figura 7-25 – Handover entre a rede visitada e a rede origem

Com base nos dados presentes nas figuras acima apresentadas, é possível retirar informação relativa ao tempo de *handover*.

A primeira coluna (4), da Figura 7-24, indica a hora exacta em que ocorre cada iteração do *daemon* do fmip6. Assim sendo, é possível fazer o cálculo do tempo que decorre entre o envio do *Router Advertisement* por parte do novo *router* de acesso, e a atribuição do novo *Care-of-Address* da rede visitada.

Com base nas figuras acima apresentadas, obteve-se um tempo de *handover* de 0.83 segundos no sentido da rede origem para a rede visitada. No sentido inverso obteve-se 0.002 segundos.

Foram realizados mais alguns testes, com o objectivo de determinar um valor médio para o tempo de *handover* em ambos os sentidos. Em alguns desses testes não foi possível voltar à rede origem, ou seja, não se conseguiu executar o *handover* nesse sentido. A Tabela 3 apresenta os valores determinados em ambos os sentidos, da rede origem para a rede destino e vice-versa.

Teste	Origem → Visitada	Visitada → Origem
1	0.04	0.0005
2	0.97	—
3	0.83	0.002
4	0.001	—
5	0.06	0.001
Média	0.38	0.001

Tabela 3 – Tempo de handover no FMIPv6

É possível verificar que o *handover* apresenta valores bastante pequenos. O FMIPv6 é consideravelmente mais rápido. No entanto, não é possível justificar a discrepância de valores nos testes realizados.

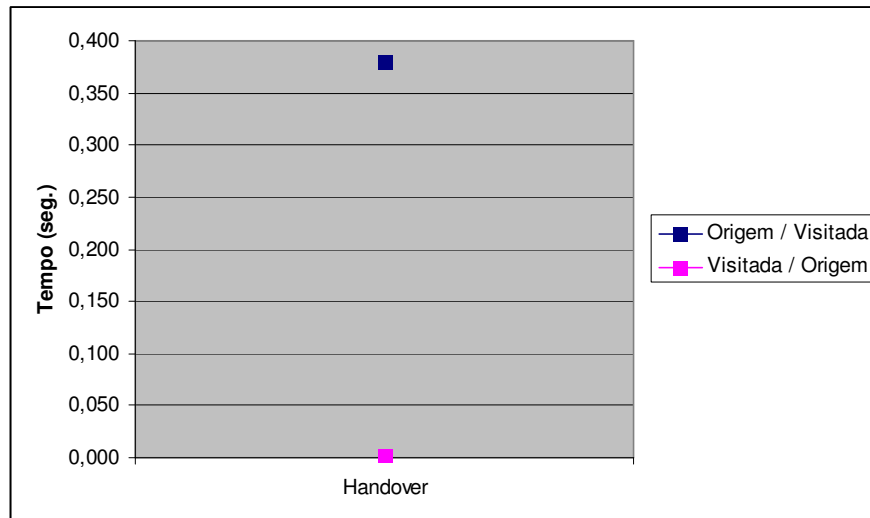


Figura 7-26 – Handover entre a rede visitada e a rede origem e vice versa

7.2.5 Resultados obtidos

Na implementação deste cenário pretendeu-se demonstrar o processo de handover do terminal móvel, da rede origem para a rede visitada e vice-versa.

Foram efectuados diversos testes fazendo vários handovers. Nestes testes, na utilização do comando *ping6* a partir do terminal móvel para o nó correspondente conseguimos observar que os valores do *radvd* configurados contribuem de uma forma significativa para os resultados. Ao configurar estes valores para os valores sugeridos na RFC 3775 [59] que são de 0.03 para o *MinRtrAdvInterval* e 0.07 para o *MaxRtrAdvInterval*, observou-se que foram perdidos muito menos, ou mesmo nenhuns pacotes *ping6*. No entanto, com os valores por omissão, mais elevados, respectivamente 1 e 3, observamos que são perdidos em média dois pacotes *ping6* em cada *handover*. Este facto acontece porque é partir dos *Router Advertisements* recebidos que é configurado o novo *Care-of-Address*.

7.2.6 Problemas detectados

Dado o FMIPv6 ser uma tecnologia ainda muito recente, existiram muitos factores que nos impediram de colocar o cenário a funcionar na plenitude. Surgiram muitos

problemas na implementação, nomeadamente na instalação, na visualização das mensagens, e na obtenção do tempo que demora o *handover* a ocorrer. No entanto, conseguimos visualizar algumas dessas mensagens apenas através do *daemon* fmipv6. No repositório de e-mails oficial do FMIPv6 [73], estão identificados diversos problemas na implementação que ainda não foram resolvidos, porque a implementação ainda é experimental. Além disso não são dadas nenhuma garantia que está a funcionar na plenitude para todos os cenários e equipamentos.

Existem anomalias com algumas placas de rede que se comportam de maneira diferente no processo de *handover*.

Em termos de visualização de mensagens, não foi possível determinar as mensagens que circulavam na rede, pois o Wireshark e o *tcpdump* não devolvem nenhum resultado de mensagens fmipv6. No repositório de e-mail estão colocadas diversas questões neste sentido, onde a solução passaria pela colocação de diversos ficheiros ao Wireshark. A aplicação destes ficheiros não foi possível em tempo útil, sendo assim o único resultado conseguido foi através do *debug* do respectivo *daemon*.

É importante referir que não existe nenhum How To da implementação do *daemon* fmipv6. A documentação existente sobre o fmipv6 em cenários reais é muito escassa e pouco esclarecedora.

7.3 Comparação entre MIPv6 e FMIPv6

Nesta secção pretende-se realizar uma comparação entre os resultados obtidos nos dois pontos anteriores. Esta comparação passa pela análise dos valores de *handover* em cada um dos cenários, MIPv6 e FMIPv6.

Com já foi acima referido, no caso do MIPv6, os testes realizados com os parâmetros *MinRtrAdvInterval* e *MaxRtrAdvInterval* configurados com os seu valores mínimos, 0.03 e 0.07, respectivamente, apresentaram tempos de *handover* mais breves quando comparados com os mesmos testes com os *Router Advertisements* configurados a 1 e 3. No entanto, o FMIPv6, para os mesmos valores de *Router Advertisements* apresenta ainda melhores resultados, como era esperado.

A Tabela 4 mostra a comparação entre o *handover* do MIPv6 e do FMIPv6.

Teste	Origem → Visitada	Visitada → Origem
MIPv6	4.284	3.264
FMIPv6	0.38	0.001

Tabela 4 – Comparação de handover entre MIPv6 e FMIPv6

Ao analisar a tabela é possível verificar que o *handover* no sentido da rede origem é mais rápido para ambos os testes. Isto deve-se ao facto de não ser necessário efectuar todo o processo de registo de endereços e a comunicação com o *Home Agent* e nó correspondente. Quando o terminal móvel se desloca para a rede origem, o único processo que ocorre é a eliminação do túnel que foi previamente criado para estabelecer a comunicação com o novo *Care-of-Address*.

Em relação à perda de pacotes, também é importante referir que o FMIPv6 perde muito menos pacotes do que o MIP. Isto deve-se não só ao facto não só dos *Router Advertisements* estarem configurados com os valores mínimos mas também porque o *fast handover* é conhecido por detectar previamente a deslocação para uma nova rede, logo começa o processo de handover mesmo antes de quebrar completamente a ligação com a rede origem.

Finalizando esta comparação, e tal como foi anteriormente referido, o protocolo FMIPv6 apresenta melhores resultados em tempo de handover do que o MIPv6. Assim sendo, para todo o tipo de aplicações, mas principalmente aqueles que são em tempo real e que utilizem a mobilidade, recomenda-se a utilização deste protocolo pois apresenta uma menor perda de pacotes em relação ao MIPv6.

8 Conclusões

Neste capítulo são apresentados os objectivos realizados, as limitações, a apreciação final e o trabalho futuro.

8.1 Objectivos realizados

A realização deste projecto foi gratificante pelo facto de que nos permitiu ter contacto com esta nova tecnologia. Pudemos perceber o funcionamento da mobilidade que tanto está presente nos nossos dias.

Ao longo da realização deste projecto o objectivo em mente foi a comparação entre o *handover* e o *fast handover*. Para atingir este objectivo foi necessário a realização de um estudo aprofundado sobre os conceitos envolvidos nos protocolos MIPv6 e FMIPv6 e foram definidos os possíveis cenários a partir das implementações de cada protocolo. Na definição dos cenários a implementar foi ainda colocado a possibilidade de alterar o cenário fornecido pela implementação do FMIPv6, a alteração consistia em colocar o *router* de acesso a executar também funções de *Home Agent*. A alteração foi desaconselhada pela equipa de desenvolvimento do projecto FMIPv6 e por este motivo não foi implementada.

A implementação dos cenários baseou-se numa rede *wireless* de forma a maximizar o uso da tecnologia de mobilidade. No entanto apenas foi utilizado um computador portátil, não sendo testado em equipamentos pda's e telemóveis 3G por não existir ainda implementações estáveis destes protocolos implementáveis em tais dispositivos.

A implementação dos cenários de teste envolveu diversos procedimentos para ambos os protocolos. No cenário de teste do protocolo MIPv6 foram utilizados manuais de configuração e instalação disponíveis pela equipa de desenvolvimento da implementação MIPv6. No caso do FMIPv6 a implementação foi conseguida com grande esforço, dada a inexistência de documentação sobre a implementação e configuração do cenário apresentado pelo grupo de trabalho do FMIPv6.

O resultado final deste projecto foi conseguido, embora com poucos parâmetros de comparação devido aos problemas que surgiram na fase de implementação que tornou o tempo muito escasso para a execução de mais testes de forma a solidificar a ideia

que o FMIPv6 tem tempos de *handover* muito menores que o MIPv6, e que o seu uso em aplicações de tempo real se torna bastante aceitável.

8.2 Limitações

Ao longo deste estudo foi possível verificar que a tecnologia ainda se encontra bastante imatura. As implementações disponíveis ainda não atingem todas as funcionalidades especificadas pelos protocolos e são experimentais.

As implementações que existem são ainda difíceis de configurar e não apresentam grandes soluções quando ocorrem erros na sua configuração.

No caso da implementação do MIPL, não existe grandes novidades desde a altura do último projecto de mobilidade realizado na ESTG [1]. Continuam a existir alguns problemas na sua configuração e não foram adicionadas novas funcionalidades.

No caso da implementação do FMIPv6 as configurações ainda não estão documentadas, sendo apenas dados alguns passos no ficheiro de instalação do *daemon*. Desta forma a instalação do *daemon* e sua utilização tornou-se difícil, porque foi necessário adaptar o Linux de forma a compilar correctamente o *daemon*.

Os requisitos de material na implementação do FMIPv6 também não estão documentados. Não são referidos equipamentos específicos embora tenham sido detectadas algumas anomalias em algumas placas de rede *wireless* que se comportam de maneiras diferentes durante o processo de *handover*. Esta situação está a ser revista pela equipa de desenvolvimento do projecto FMIPv6. Esta situação poderia ter causado algumas anomalias no decorrer dos testes realizados neste projecto. Apenas foi testado com uma placa de rede *wireless* e por vezes a placa desligava-se e deixava de ter conectividade a ambas as redes. Não foi detectada nenhuma explicação para a ocorrência desta situação.

A fase de implementação e configuração do cenário foi muito longa, retirando assim tempo para a realização de testes exaustivos. A ideia inicial passou por fazer comparações com diversos tipos de tráfego, e com a transferência de ficheiros entre o terminal móvel e o nó correspondente. Todavia, não foi possível testar o FMIPv6 com outro tráfego além do tradicional icmp, embora este não seja muito fiável, mas foi a

única forma possível de realizar testes no curto espaço de tempo que restava para a conclusão do projecto.

8.3 Trabalho futuro

Nesta secção são descritas algumas propostas de trabalho futuro que podem contribuir para o projecto IPv6@ESTG-IPLEIRIA. Estas propostas passam pela implementação FMIPv6 com o protocolo HMIPv6 e testar a heterogeneidade da mobilidade entre IPv4/IPv6.

8.3.1 Hierarchical Mobile IPv6 (HMIPv6) Handover

O HMIPv6 foi criado com o objectivo de reduzir a quantidade de mensagens de sinalização transmitidas entre o terminal móvel, os nós correspondentes e o seu *Home Agent*. Para este efeito, o *Mobility Anchor Point* (MAP), pode ser utilizado para melhorar o desempenho do MIPv6, relativamente à velocidade do handover.

O MAP é essencialmente um *Home Agent* local que irá limitar a quantidade de mensagens de sinalização MIPv6 com origem de fora do domínio local.

O objectivo de introduzir o HMIPv6 é realçar o desempenho do MIPv6 enquanto que se minimiza o impacto no MIPv6 ou nos protocolos a ele associados. Eliminar o atraso adicional provocado pela troca de mensagens pode melhorar significativamente o desempenho do MIPv6.

O HMIPv6 também suporta *Fast Mobile IPv6 Handovers* para ajudar o terminal móvel a conseguir mobilidade *seamless*. Existem duas formas de integrar os mecanismos HMIPv6 e *Fast Handover*. O primeiro refere a colocação de MAPs em vez de *routers* de acesso. O Segundo cenário refere a colocação do MAP num router de agregação, “acima” dos *routers* de acesso.

A combinação de *Fast Handover* com HMIPv6 permite a antecipação do *handover* de camada 3, de tal forma que o tráfego de dados pode ser redireccionado para a nova localização do móvel, mesma antes de ele se mover para lá.

Poderá ser feito um estudo sobre o funcionamento, requisitos e restrições deste protocolo em conjunto com o FMIPv6 designado por FHMIPv6.

8.3.2 Heterogeneidade IPv4/IPv6

A mobilidade IP tem especificações para os protocolos IPv4 e IPv6. O conceito e as necessidades são semelhantes mas existem algumas melhorias significativas no IPv6 relativamente ao standard IPv4.

A adaptação da mobilidade IP em cenários reais está em crescimento, mas os mecanismos de transição assumem que os dispositivos são estáticos dentro de uma rede IPv4 ou IPv6. Mas, será que existe alguma solução de mobilidade IP em cenários de transição? Existem vários drafts que debatem este tópico, na grande maioria, referem a necessidade de *Dual Stack* no MN (DS-MN). No entanto, existem diversos mecanismos de transição entre IPv4/IPv6.

Poderá ser feito um levantamento dos diversos tipos de mecanismos de transição IPv4/IPv6 que permitem que a mobilidade seja transparente para a versão IP.

Bibliografia

Documentos:

- [1] Amado, Tiago Mira; “IPv6@ESTG-Leiria Testes de Mobilidade IPv6”; Relatório final da disciplina de Projecto II, do curso de Engenharia Informática e Comunicações; ESTG-Leiria, Fevereiro 2006
<http://www.ipv6.estg.ipleiria.pt/>
- [2] Bask, Joni; “Heterogeneous Handoff Algorithm for Mobile IPv4 without Foreign Agents”, Maio 2003
- [3] Bellavista, Paolo; Corradi, António; Giannelli, Carlo; “Adaptative Buffering-based on Handoff Prediction for Wireless Internet Continuous Services”
<http://zeus.elet.polimi.it/is-manet/Documenti/pap-deis-38.pdf>
- [4] Cabellos-Aparicio, Albert; Julian-Bertomeu, Hector; Núñez-Martínez, Jose; Jakab, Loránd; Serral-Gracià, René; Domingo-Pascual, Jordi; “Measurement-Based Comparison of IPv4/IPv6 Mobility Protocols on a WLAN Scenario”
<http://www.comp.brad.ac.uk/het-net/HET-NETs05/ReadCamera05/P16.pdf>
- [5] Cabellos-Aparicio, Albert; Núñez-Martínez, Jose; Julian-Bertomeu, Hector; Jakab, Loránd; Serral-Gracià, René; Domingo-Pascual, Jordi; “Evaluation of the Fast Handover Implementation for Mobile IPv6 in a real Testbed”
<http://people.ac.upc.edu/ljakab/fasthandover.pdf>
- [6] Cabellos-Aparicio, Albert; Serral-Gracià, René; Jakab, Loránd; Domingo-Pascual, Jordi; “Measurement Based Analysis of the Handover in a WLAN MIPv6 Scenario”
- [7] Chakravorty, Rajiv; Vidales, Pablo; Patanapongpibul, Leo; Subramanian, Kavitha; Pratt, Ian; Crowcroft, Jon; “On Inter-network Handover Performance using Mobile IPv6”
<http://www.cl.cam.ac.uk/~rc277/handovers.pdf>
- [8] Colombari, Roberto; Fessia, Flávio; “QoS (Quality of Service) issues in Mobile IP Problem Analyses and Sugestions”, Março 2001
- [9] Costa, Xavier Pérez; Schmitz, Ralf; Hartenstein, Hannes; Liebsch, Marco; “A MIPv6, FMIPv6 and HMIPv6 handover latency study: analytical

approach”

http://www.ist-mobydick.org/publications/MIPv6_latency_study.pdf

- [10] Duff, K.; Clayton, P.; Terzoli, A.; “Facilitating Seamless Macro-Mobility across Heterogeneous Wireless IP Network”
<http://www.fester.rucus.net/documents/duff-satnac05.pdf>
- [11] Dunmore, Martin; Pagtziz, Theo; Edwards, Chris; Deliverable D4.1.3 “Mobile IPv6 Handovers: Performance Analysis and Evaluation”, Junho 2005
<http://www.6net.org/publications/deliverables/D4.1.3v1.pdf>
- [12] Dunmore, Martin; Ruppelt, Reinhard; Saarinen, Tommi; Edwards, Chris; Deliverable D4.1.4 “Final Mobile IPv6 Support Guide”, Fevereiro 2005
- [13] Estrela, Pedro Vale; “Mobilidade em IP – Estado da Arte”, trabalho da disciplina de Tópicos Avançados em Conectividade e Sistemas Distribuídos, do Instituto Superior Técnico, Abril 2002
- [14] Estrela, Pedro Vale; “Protocolos de Mobilidade para Terminais IP”, Dissertação para a obtenção do Grau de Mestre em Engenharia Informática e de Computadores, Instituto Superior Técnico, Janeiro 2003
http://tagus.inesc-id.pt/~pestrela/timip/Estrela_MSC_-_Protocolos_Mobilidade_para_Terminais_IP.pdf
- [15] Fathi, Hanane; Chakraborty, Shyam; Prasad, Ramjee; “Mobility management for VoIP: Evaluation of Mobile IP-based protocols”
<http://ieeexplore.ieee.org/iel5/9996/32113/01495021.pdf>
- [16] Hautala, Mikko; “Mobile IPv6 performance in 802.11 networks: handover optimizations on the link and network layer”, Março 2006
http://www.netlab.tkk.fi/opetus/s383310/05-06/Kalvot%2005-06/hautala_160306.pdf
- [17] Hernandez, Edwin; Helal, Sumi; “Predictive Mobile IP for Rapid Mobility”
<http://www.icta.ufl.edu/projects/publications/Helal-Hernandez-LCN04-FINAL.pdf>

- [18] Jang, Hee Jing; “Mobile IPv6 Fast Handovers over IEEE 802.16 Networks”
<http://www.vsix.net/other/summit/Korea2006/www.ipv6.or.kr/summit2006/proceeding/TS4-1.pdf>
- [19] Krob, Andréa Collin; Bruno, Gaspare Giuliana E.; “Análise de desempenho do handoff no Mobile IP”
<http://www.upf.br/errc2006/download/Analise.de.desempenho.do.handoff.no.Mobile.IP.21969.pdf>
- [20] Li, Ruidong; Li, Jie; “Na Enhanced Handover Scheme for Mobile IPv6”
- [21] Nagamuta, Vera; “Protocolos de Micro-Mobilidade IP”
gsd.ime.usp.br/seminars/2005/nagamuta.ppt
- [22] Petander, Henrik; Perera, Eranga; Seneviratne, Aruna; “Multicasting with selective delivery: A SafetyNet for vertical handoffs”
<http://arxiv.org/ftp/cs/papers/0607/0607137.pdf>
- [23] Riva, Oriana; “IP Mobility Solutions”, Dezembro 2003
http://www.cs.helsinki.fi/u/jmanner/Courses/seminar_papers/mobility.pdf
- [24] Serafim, David L. S.; Santos, Vítor A. C. ; “IPv6@ESTG-Leiria – Instalação de uma Rede Piloto”; Relatório final da disciplina de Projecto I, do curso de Engenharia Informática e Comunicações; ESTG Leiria, Julho 2005
<http://www.ipv6.estg.ipleiria.pt/>
- [25] Schmidt, Thomas C.; Wahlisch, Mathias; “Performance Analysis of Multicast Mobility in a H-MIP Proxy Environment”
<http://www.6net.org/events/workshop-2004/schmidt.pdf>
- [26] Schmidt, Thomas C.; “Mobility Optimization in IPv6 Networks”
<http://users.informatik.haw-hamburg.de/~schmidt/it/MIPv6.pdf>
- [27] Schmidt, Thomas C.; Wahlisch, Mathias; “Seamless Multicast Handover in a Hierarchical Mobile IPv6 Environment (M-HMIPv6)”, 2005
<http://www3.ietf.org/proceedings/05aug/slides/MobOpts-5.pdf>
- [28] Stefan; Gorg Carmelita Aust; Pampu, Cornel; “Proactive Handover Decision for Mobile IP based on Link Layer Information”
http://www.comnets.uni-bremen.de/~aust/Aust_WOCN2004.pdf

- [29] Schmidt, Thomas C.; Wahlisch, Mathias; “Performance Analysis of Multicast Mobility in a Hierarchical Mobile IP Proxy Environment”, 2004
<http://www.terena.nl/publications/tnc2004-proceedings/papers/schmidt.pdf>
- [30] Schmidt, Thomas C.; Wahlisch, Mathias; “Predictive versus Reactive – Analysis of Handover Performance and its Implications on IPv6 and Multicast Mobility”
<http://www.rz.fhtw-berlin.de/data/rz.web/content/Materialien/Projekte/mipv6/telesys05.pdf>
- [31] Schotanus, H. A.; Verkoelen, C. A. A.; “Description of na IPv6 testbed”, Dezembro 2000
- [32] Sousa, Tiago; Monteiro, Edmundo; Boavida, Fernando; “Estudo do IPv6 móvel em Linux”
<http://eden.dei.uc.pt/~edmundo/Conf%20Nacionais/CN26%202003%20CRC%20Sousa.pdf>
- [33] Su-xiang, Weng; Shu-fen, Liu; Zhi-lin, Yao; An Implementation of Mobile IP’s Tunnel Technology in Linux Kernel
<http://ieeexplore.ieee.org/iel5/9313/29613/01349066.pdf?arnumber=1349066>
- [34] Thomas, Joseph; Luo, Lan; “Efficient IP-Based UMTS Networks”
- [35] Ylianttila, Mika; “Vertical Handoff and Mobility – System Architecture and Transition Analysis”, 2005
<http://herkules.oulu.fi/isbn9514276922/isbn9514276922.pdf>

Drafts:

- [36] Aust, S.; Gorg, C.; Pampu, C.; “Policy Based Mobile IPv6 Handover Decision (POLIMAND), Fevereiro 2005
<http://www.comnets.uni-bremen.de/~aust/draft-iponair-dna-polimand-02.txt>
- [37] Engelstad, P.; “Transitional Integration of Mobile IPv4 and Mobile IPv6”, Agosto 2001
<http://www.join.uni-muenster.de/Dokumente/drafts/draft-engelstad-ngtrans-mipv4-over-mipv6-01.txt>

- [38] IPv6 Forum, “Mobile IPv6 Test Profile version 3.3.1”, 2005
http://cert.v6pc.jp/mip6/doc-3.3/mip6_profile_v3_3_1.pdf
- [39] Koodli, Rajeev; Perkins, Charles E.; “Adapting Mobile IPv6 Fast Handovers for IPv4”, Outubro 2004
<http://tools.ietf.org/wg/mip4/draft-jee-mip4-fh80216e-00.txt>
- [40] Koodli, Rajeev; Perkins, Charles E.; “Mobile IPv4 Fast Handovers”, Outubro 2006
<http://www.ietf.org/internet-drafts/draft-ietf-mip4-fmipv4-02.txt>
- [41] Manner, J.; Kojo, M.; Suihko, T.; Eardley, P.; Wisely, D.; Hancock, R.; Georganopoulos, N.; “Mobility Related Terminology”, Janeiro 2001, <draft-manner-seamoby-terms-00.txt>
- [42] Tsirtsis, George; Soliman, Hesham; “Mobility management for Dual Stack mobile nodes - A problem Statement”, Junho 2006
<http://www.ietf.org/internet-drafts/draft-ietf-mip6-dsmip-problem-02.txt>
- [43] Tsirtsis, George; Soliman, Hesham; Park, V.; “Dual Stack Mobile IPv4”, Julho 2006
<http://www.ietf.org/internet-drafts/draft-ietf-mip4-dsmipv4-00.txt>
- [44] Xia, F.; Sarikaya, B.; “FMIPv6 extension for Multicast Handover”, Setembro 2006
<http://tools.ietf.org/wg/mipshop/draft-xia-mipshop-fmip-multicast-00.txt>

HOW TO:

- [45] Linux Mobile IPv6 HOW TO – Lars Strand
- [46] Linux IPv6 HOW TO (en) – Peter Bieringer

Livros:

- [47] Popoviciu, Ciprian; Levy-Abegnoli, Eric; Grossetete, Patrick; Deploying IPv6 Networks, Cisco Press, 2006 ISBN 1-58705-210-5

- [48] Soliman, Heshman; Mobile IPv6 - Mobility in a Wireless Internet, Addison Wesley, 2004 ISBN 0-201-78897-7

RFCs:

- [49] RFC 2002 – Perkins, C.; “IP Mobility Support”, Outubro 1996
- [50] RFC 2104 – Kraczynk, H.; Bellare, M.; Canetti, R.; “HMAC: Keyed-Hashing for Message Authentication”, Fevereiro 1997
- [51] RFC 2401 – Kent, S.; Atkinson, R.; “Security Architecture for Internet Protocol”, Novembro 1998
- [52] RFC 2460 – Deering, S.; Hinden, R.; “Internet Protocol, Version 6 (IPv6) Specification”, Dezembro 1998
- [53] RFC 2461 – Narten, T.; Nordmark, E.; Simpson, W.; “Neighbor Discovery for IP Version 6 (IPv6)”, Dezembro 1998
- [54] RFC 2462 – Thomson, S.; Narten, T.; “IPv6 Stateless Address Autoconfiguration”, Dezembro 1998
- [55] RFC 2765 – Nordmark, E.; " Stateless IP/ICMP Translation Algorithm (SIIT)", Fevereiro 2000
- [56] RFC 2766 – Tsirtsis, G.; Srisuresh, P.; "Network Address Translation - Protocol Translation (NAT-PT)", Fevereiro 2000
- [57] RFC 3220 – Perkins, C.; “IP Mobility Support for IPv4”, Janeiro 2002
- [58] RFC 3344 – Perkins, C.; “IP Mobility Support for IPv4”, Agosto 2002
- [59] RFC 3775 – Johnson, D.; Perkins, C.; Arkko, J.; “Mobility Support in IPv6”, Junho 2004
- [60] RFC 4068 – Koodli, R.; “Fast Handovers for Mobile IPv6”, Julho 2005
- [61] RFC 4140 – Soliman, H.; Castelluccia, C.; Malki, K. El, Bellier, L.; “Hierarchical Mobile IPv6 Mobility Management (HMIPv6)”, Agosto 2005
- [62] RFC 4214 – Templin, F.; Gleeson, T.; Talwar, M.; Thaler, D.; "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", Outubro 2005

- [63] RFC 4260 – McCann, P.; “Mobile IPv6 Fast Handovers for 802.11 Networks”, Novembro 2005
- [64] RFC 4495 – Polk, J.; Dhesikan, S.; “A Resource Reservation Protocol (RSVP) Extension for the Reduction of Bandwidth of a Reservation Flow”, Maio 2006

Websites:

- [65] IETF, Active IETF Working Groups – Internet Area
<http://www.ietf.org/html.charters/wg-dir.html#Internet%20Area>
- [66] IEEE, Wireless Working Group
<http://www.ieee802.org/11/>
- [67] The IPv6 Portal
<http://www.ipv6tf.org>
- [68] Nautilus6
<http://www.nautilus6.org>
- [69] 6 NET
<http://www.6net.org>
- [70] Site do MIPv6 Analyzer
<http://www.cavone.com/mipv6-analyzer/>
- [71] Moby Dick
http://www.ist-mobydick.org/publications/MIPv6_latency_study.pdf
- [72] Site oficial do MIPv6
<http://mobile-ipv6.org>
- [73] Site oficial do FMIPv6
<http://www.fmipv6.org/>

- [74] Site oficial do VMWare
<http://www.vmware.com>
- [75] Site do mipv6tester
<http://www.bullopen-source.org/mipv6/tester.php>

Anexo A Ficheiros de configuração

Ficheiros de configuração do *Home Agent* para o MIPv6

Interfaces, encaminhamento e iniciação de serviços

```
#interface eth0

/sbin/ifconfig eth0 down

/sbin/ifconfig eth0 up

/sbin/ip -6 addr add 2001:2::10/64 dev eth0

#interface eth1

/sbin/ifconfig eth1 down

/sbin/ifconfig eth1 up

/sbin/ip -6 addr add 2001:1::10/64 dev eth1

#Activate IPForwarding

/sbin/sysctl -w net.ipv6.conf.all.forwarding=1

/sbin/sysctl -w net.ipv6.conf.all.autoconf=0

/sbin/sysctl -w net.ipv6.conf.all.accept_ra=0

/sbin/sysctl -w net.ipv6.conf.all.accept_redirects=0

#Routes

/sbin/ip -6 route flush all

/sbin/ip -6 route add 2001:3::/64 via 2001:2::20

#Inicializar os router advertisements

/usr/local/sbin/radvd-1.0/radvd -C /usr/local/etc/radvd.conf

#Inicializar o mip6

mip6d -d 1 -c /usr/local/etc/mip6d.conf
```

Daemon mip6d

```
## This is an example of mip6d Home Agent configuration file

#### Common file

## Set the mode witch the deamon will run
NodeConfig HA;

# If set to > 0 will not detach from tty
DebugLevel 10;

## Suport route optimization with other MNs
DoRouteOptimizationCN enabled;

#### Specific options to Home Agent

## Specifies the interface and options. If no options "Interface
"eth0"
Interface "eth1";

## We won't use IPsec
UseMnHaIPsec disabled;
OptimisticHandoff enabled;

##END
```

Daemon radvd

```
interface eth1
{
    AdvSendAdvert on;
    AdvIntervalOpt on;

    # These settings cause advertisements to be sent every 1-3 seconds.
    MinRtrAdvInterval 0.03;
    MaxRtrAdvInterval 0.07;
    AdvHomeAgentFlag on;

    HomeAgentLifetime 10000;
    HomeAgentPreference 20;
    AdvHomeAgentInfo on;

    prefix 2001:1::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
        AdvRouterAddr on;
        AdvPreferredLifetime 500;
        AdvValidLifetime 600;
    };
};
```

Ficheiros de configuração do terminal móvel para o MIPv6

Interfaces, encaminhamento e iniciação de serviços

```
#interface eth1

/sbin/ifconfig eth1 down

/sbin/ifconfig eth1 up

/sbin/ip -6 addr add 2001:1::20/64 dev eth1


#Activate IPForwarding

/sbin/sysctl -w net.ipv6.conf.all.forwarding=0

/sbin/sysctl -w net.ipv6.conf.all.autoconf=1

/sbin/sysctl -w net.ipv6.conf.all.accept_ra=1

/sbin/sysctl -w net.ipv6.conf.all.accept_redirects=1


#Routes

/sbin/ip -6 route flush all

/sbin/ip -6 route add ::/0 via 2001:1::10


iwconfig eth1 mode managed essid "MIPv6" enc off


#Inicializar o mip6

mip6d -c /usr/local/etc/mip6d.conf
```

Daemon mip6d

```
## This is an example of mip6d Mobile Node configuration file

#### Common file

## Set the mode witch the deamon will run

NodeConfig MN;

# If set to > 0 will not detach from tty

DebugLevel 10;

## Suport route optimization with other MNs

DoRouteOptimizationCN enabled;

## Specifies the interface and options. If no options "Interface
"eth0"

Interface "eth1";

## We won't use IPsec

UseMnHaIPsec disabled;

OptimisticHandoff enabled;

SendMobPfxSols enabled;

DoRouteOptimizationMN enabled;

UseCnBuAck enabled;

MnHomeLink "eth1"{

    HomeAgentAddress 2001:1::10;

    HomeAddress 2001:1::20/64;

}

##END
```

Ficheiros de configuração do *router* acesso/nó correspondente para o MIPv6

Interfaces, encaminhamento e iniciação de serviços

```
#interface eth0

/sbin/ifconfig eth0 down

/sbin/ifconfig eth0 up

/sbin/ip -6 addr add 2001:2::20/64 dev eth0


#interface eth1

/sbin/ifconfig eth1 down

/sbin/ifconfig eth1 up

/sbin/ip -6 addr add 2001:3::10/64 dev eth1


#Activate IPForwarding

/sbin/sysctl -w net.ipv6.conf.all.forwarding=1

/sbin/sysctl -w net.ipv6.conf.all.autoconf=0

/sbin/sysctl -w net.ipv6.conf.all.accept_ra=0

/sbin/sysctl -w net.ipv6.conf.all.accept_redirects=0


#Routes

/sbin/ip -6 route flush all

/sbin/ip -6 route add 2001:1::/64 via 2001:2::10


#Inicializar os router advertisements

/usr/local/sbin/radvd-1.0/radvd -C /usr/local/etc/radvd.conf

#Inicializar o mip6

mip6d -d 1 -c /usr/local/etc/mip6d.conf
```

Daemon mip6d

```
## This is an example of mip6d Home Agent configuration file
```

```
#### Common file
```

```
## Set the mode witch the deamon will run
```

```
NodeConfig CN;
```

```
# If set to > 0 will not detach from tty
```

```
DebugLevel 10;
```

```
## Suport route optimization with other MNs
```

```
DoRouteOptimizationCN enabled;
```

```
##END
```

Daemon radvd

```
interface eth1
{
    AdvSendAdvert on;
    AdvIntervalOpt on;

    # These settings cause advertisements to be sent every 1-3 seconds.
    MinRtrAdvInterval 0.03;
    MaxRtrAdvInterval 0.07;
    AdvHomeAgentFlag off;

    prefix 2001:3::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
        AdvRouterAddr on;
    };
};
```

Ficheiros de configuração do AR1 para o FMIPv6

Interfaces, encaminhamento e iniciação de serviços

```
#interface eth0 backbone
/sbin/ifconfig eth0 down
/sbin/ifconfig eth0 up
/sbin/ip -6 addr add 2001:2::10/64 dev eth0

#interface eth1 wireless
/sbin/ifconfig eth1 down
/sbin/ifconfig eth1 up
/sbin/ip -6 addr add 2001:1::10/64 dev eth1

#Activate IPForwarding
/sbin/sysctl -w net.ipv6.conf.all.forwarding=1
/sbin/sysctl -w net.ipv6.conf.all.autoconf=0
/sbin/sysctl -w net.ipv6.conf.all.accept_ra=0
/sbin/sysctl -w net.ipv6.conf.all.accept_redirects=0

#Routes
/sbin/ip -6 route flush all
/sbin/ip -6 route add 2001:3::/64 via 2001:2::20

#Inicializar os router advertisements
radvd -C /usr/local/etc/radvd.conf

#FMIPv6
fmipv6-ar -C /usr/local/etc/fmipv6_ar.conf
```

Daemon fmipv6

```
DetachFromTTY = false

DebugLevel = 3

AskSupportForBuffering = true

BufferLength = 5

Interface{

    IfName = "eth1"

    preference = 1

}

NIHover{

    do_ni_hover = false

    ap_lla = 00:40:96:54:1a:47 # endereço MAC do AP da rede
    visitada

}

nap_fmipar1{

    nap_lla = 00:40:96:54:1a:47

    nr_lla = 00:0c:29:da:80:52

    #nr_lla = 00:40:96:54:1a:47

    nr_addr = 2001:3::10/64

    nr_pfx = 2001:3::/64

}
```

Daemon radvd

```
interface eth1
{
    AdvSendAdvert on;
    AdvIntervalOpt off;

    # These settings cause advertisements to be sent every 1-3 seconds.
    MinRtrAdvInterval 1;
    MaxRtrAdvInterval 3;
    AdvHomeAgentFlag on;

    HomeAgentLifetime 10000;
    HomeAgentPreference 20;
    AdvHomeAgentInfo on;

    prefix 2001:1::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
        AdvRouterAddr on;
    };
};
```

Ficheiros de configuração do AR2 para o FMIPv6

Interfaces, encaminhamento e iniciação de serviços

```
#interface eth0

/sbin/ifconfig eth0 down

/sbin/ifconfig eth0 up

/sbin/ip -6 addr add 2001:2::30/64 dev eth0

#interface eth1

/sbin/ifconfig eth1 down

/sbin/ifconfig eth1 up

/sbin/ip -6 addr add 2001:3::10/64 dev eth1

#Activate IPForwarding

/sbin/sysctl -w net.ipv6.conf.all.forwarding=1

/sbin/sysctl -w net.ipv6.conf.all.autoconf=0

/sbin/sysctl -w net.ipv6.conf.all.accept_ra=0

/sbin/sysctl -w net.ipv6.conf.all.accept_redirects=0


#Routes

/sbin/ip -6 route flush all

/sbin/ip -6 route add 2001:1::/64 via 2001:2::10


#Radvd

/usr/local/sbin/radvd-1.0/radvd -C /etc/radvd_fmip.conf


#FMIPv6

fmipv6-ar -C /usr/local/etc/fmipv6_ar.conf
```

Deamon fmipv6

```
DetachFromTTY = false
```

```
DebugLevel = 3
```

```
AskSupportForBuffering = true
```

```
BufferLength = 5
```

```
Interface{
```

```
    IfName = "eth1"
```

```
    preference = 1
```

```
}
```

```
NIHover{
```

```
    do_ni_hover = false
```

```
    ap_lla = 00:40:96:54:8e:c6 # endereço MAC do AP da rede  
    visitada
```

```
}
```

```
nap fmipar1{
```

```
    nap_lla = 00:40:96:54:8e:c6
```

```
    nr_lla = 00:50:fc:40:ed:46 #00:40:96:54:8e:c6 #  
    00:50:fc:40:ed:46
```

```
    nr_addr = 2001:1::10/64
```

```
    nr_pfx = 2001:1::/64
```

```
}
```

Daemon radvd

```
interface eth1
{
    AdvSendAdvert on;
    AdvIntervalOpt on;

    #These settings cause advertisements to be sent every 3-1 seconds.
    MinRtrAdvInterval 1; #0.03;
    MaxRtrAdvInterval 3; #0.07;

    prefix 2001:3::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
        AdvRouterAddr on;
    };
};
```

Ficheiros de configuração do terminal móvel para o FMIPv6

Interfaces, encaminhamento e iniciação de serviços

```
#interface eth1

/sbin/ifconfig eth1 down

/sbin/ifconfig eth1 up

/sbin/ip -6 addr add 2001:1::20/64 dev eth1


#Activate IPForwarding

/sbin/sysctl -w net.ipv6.conf.all.forwarding=0

/sbin/sysctl -w net.ipv6.conf.all.autoconf=1

/sbin/sysctl -w net.ipv6.conf.all.accept_ra=1

/sbin/sysctl -w net.ipv6.conf.all.accept_redirects=1


#Routes

/sbin/ip -6 route flush all

/sbin/ip -6 route add ::/0 via 2001:1::10


iwconfig eth1 mode managed essid "MIPv6" enc off


#FMIPv6

fmipv6-mn -C /usr/local/etc/fmipv6_mn.conf
```

Deamon fmipv6

```
DetachFromTTY = false
```

```
DebugLevel = 3
```

```
PrimaryInterface {  
    IfName = "eth1"  
}
```

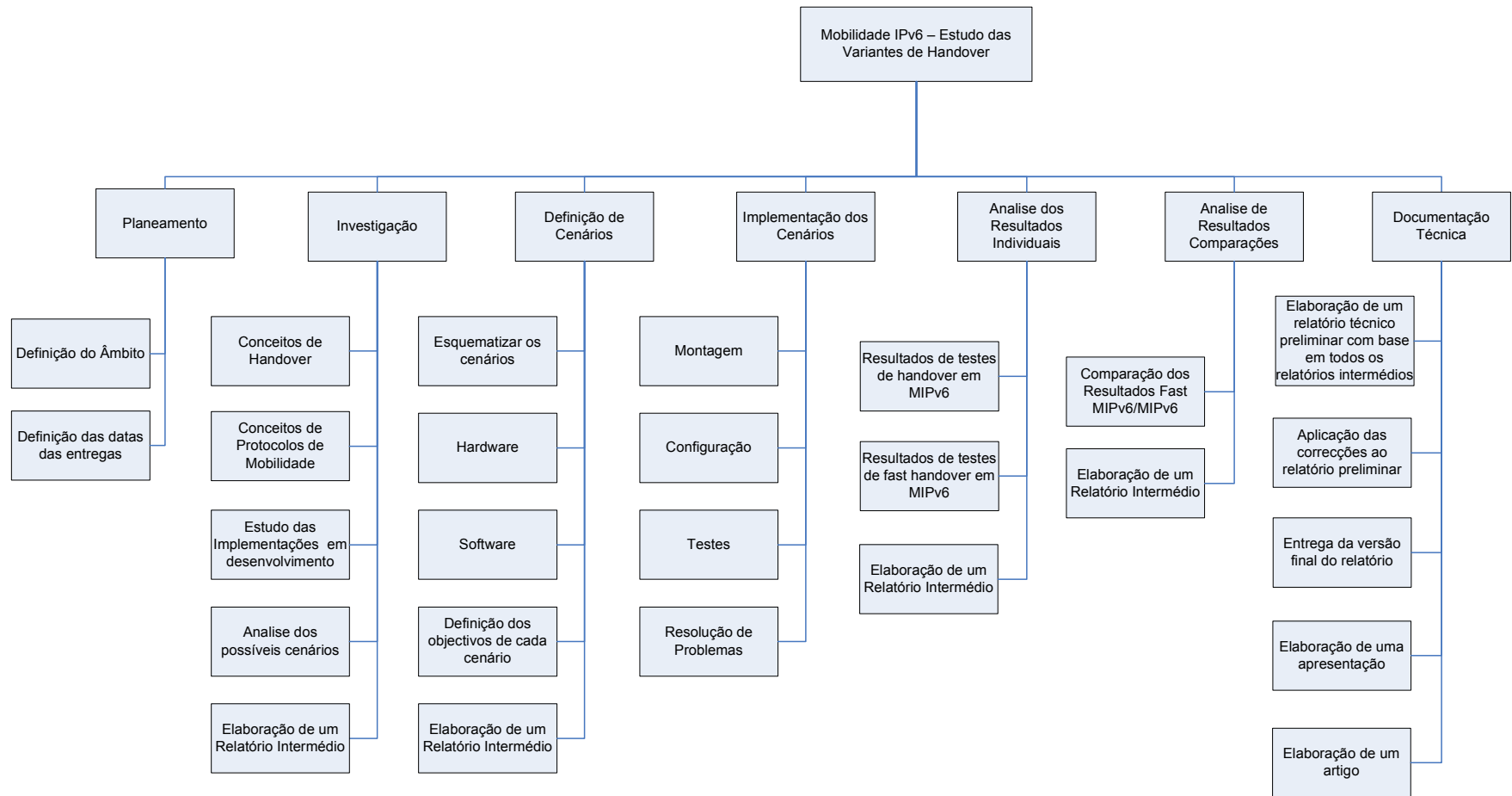
```
#ScanningInterface {  
    IfName = "eth1"  
}
```

```
LinkQuality {  
    UseLinkQualityTriggers = true  
    Threshold = -50  
}
```

Anexo B Planeamento do projecto

A definição do âmbito tem por objectivo ser uma linha orientadora para todos os elementos envolvidos no projecto. Assim sendo é necessário definir claramente os pacotes de trabalho e as tarefas a realizar. A melhor forma para explicar as tarefas a realizar é através do desenvolvimento de um organograma técnico. O organograma técnico deste projecto é apresentado na figura seguinte.

As componentes envolvidas no desenvolvimento do organograma técnico pretendem ser por si só claras e a sua nomenclatura deve ser identificativa dos seus objectivos..



O método utilizado para estabelecer a sequência e o relacionamento entre as actividades foi o disponibilizado pela ferramenta *Microsoft Project*, nomeadamente a vista de gráfico de *Gantt*. Este gráfico permite o estabelecimento de diferentes tipos de relação entre as actividades (início-início, início-fim, fim-início e fim-fim).

O diagrama de precedência permite identificar as tarefas iniciais assim como as dependências importantes entre as tarefas a realizar.