Daniel Henrique Barboza

Acesso seguro à redes móveis IP

Dissertação de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica pela Universidade de Campinas. Área de concentração: Engenharia de Computação.

Orientador: Eleri Cardozo

Banca:
Marcos Rogério Salvador - CPqD
Paulo Lício de Geus - IC / UNICAMP

Campinas, SP
Julho de 2008

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

B234a

Barboza, Daniel Henrique
Acesso seguro à redes móveis IP / Daniel Henrique Barboza. --Campinas, SP: [s.n.], 2008.

Orientador: Eleri Cardozo
Dissertação (Mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Sistemas de comunicação sem fio. 2. Internet (Redes de computação). 3. Redes de computação – Medidas de segurança. 4. Redes de computação. I. Cardozo, Eleri. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

Título em Inglês: Secure access to mobile IP networks
Palavras-chave em Inglês: Wi-fi networks access, Network security, Handover
Área de concentração: Engenharia de Computação
Titulação: Mestre em Engenharia Elétrica
Banca examinadora: Marcos Rogério Salvador, Paulo Lício de Geus
Data da defesa: 07/07/2008
Programa de Pós-Graduação: Engenharia Eletrica

COMISSÃO JULGADORA - TESE DE MESTRADO

Candidato: Daniel Henrique Barboza

Data da Defesa: 7 de julho de 2008

Título da Tese: "Acesso Seguro à Redes Móveis IP"

Prof. Dr. Elen Cardozo (Presidente): _____

Prof. Dr. Marcos Rogério Salvador: _____

Prof. Dr. Paulo Licio De Geus: _____

Abstract

Wireless devices are becoming more popular. Almost any portable device nowadays has a wireless network interface and wireless access points are available at low cost for commercial or domestic use. However, the widespread of wireless networks is creating new challenges. The most critical challenge is related to the security in the access of wireless networks. The nature of wireless communications makes harder to guarantee privacy as data travels freely in electromagnetic waves. As such, wireless networks present many security breaches not found in cabled networks. Another challenge in wireless networks is to preserve the transport connections during handover. Handover is the process by which a mobile device replaces its current access point by a new one that presents a better signal to noise ratio. During handover the transport connections kept by the mobile device are compromised due to the procedures of disassociation and re-association among the access points and the updating of the mobile node's layer three parameters for its new location. This work has the objective of evaluating the impact of security in the handover overheads and to propose mechanisms by which the upper layers are notified when a handover occurs.

Keywords: Wi-Fi networks access, security, *handover*

Resumo

Os dispositivos de rede sem fio estão cada vez mais populares. Quase todos os dispositivos portáteis atuais possuem uma interface de rede sem fio e pontos de acesso estão acessíveis a baixo custo para uso corporativo e doméstico. No entanto, este aumento no uso de redes sem fio gera novos desafios. O mais crítico deles é com relação à segurança no acesso à rede sem fio. A natureza da comunicação sem fio dificulta a confidencialidade da informação que viaja livremente em forma de ondas eletromagnéticas. Deste modo, as redes sem fio apresentam novas brechas de segurança que não existem em redes cabeadas. Outro desafio em redes sem fio é preservar as conexões da rede de transporte durante o *handover*. *Handover* é o processo pelo qual um dispositivo móvel substitui seu ponto de acesso atual por um outro que possui melhor relação sinal-ruído. Durante o *handover*, as conexões da rede de transporte mantidas pelo dispositivo móvel são comprometidas. Isto ocorre devido aos procedimentos de desassociação e re-associação entre os pontos de acesso e a atualização dos parâmetros de camada três do nó móvel para sua nova localização. Este trabalho tem como objetivo avaliar o impacto da segurança no *overhead* de *handover* e propor mecanismos pelos quais as camadas superiores sejam notificadas quando um *handover* ocorrer.

Palavras-chave: Acesso a redes sem fio, segurança, *handover*

Agradecimentos

Agradeço o meu pai Nelson José Barboza, minha mãe Maria Idalici da Silva Barboza, meus irmãos: Débora Barboza e Douglas Nelson Barboza e minha noiva Raquel Nobrega da Silva pelo amor e apoio incondicional.

Agradeço o meu orientador, professor doutor Eleri Cardozo, por ter me orientado com paciência e dedicação neste trabalho de Mestrado.

Agradeço o meu orientador de iniciação científica, professor doutor Maurício Ferreira Magalhães, por ter me ensinado os conceitos básicos da área de rede de computadores e que me possibilitaram iniciar este trabalho de Mestrado.

Agradeço os meus amigos de graduação que sempre me apoiaram e estiveram presentes desde meu ingresso na universidade, especialmente André Paraense, Bruno Magna, César Costa e Danilo Tavares.

Agradeço todos os colegas de trabalho do projeto MPA, em especial Eduardo Zagari e Rodrigo Prado.

Agradeço todos os amigos do LCA, em especial Daniel "Rocco" Moraes, Rafael Pasquini, Rodolfo Villaca, Luciano "Ruivo" de Paula, Fábio Verdi, Paulo Coelho, Alex "Trustlix" Zanetti e Christian "Espanhol" Esteves.

Agradeço Walter Wong, companheiro de graduação e de mestrado, amigo fiel em todas as horas e todos os momentos.

Por fim, agradeço os membros da república "Suino's House" Gustavo Garcia, Henrique Padoves e Guilherme "Sushi" Borges pelas madrugadas memoráveis naquele apartamento inóspito.

Sumário

Lista de Figuras

Glossário

AAA *Authentication, Authorization and Accounting*

AES *Advanced Ecryption Standard*

AH *IP Authentication Header*

BU *Binding Update*

CCMP *Counter Mode with Cipher Block Chaining Message Authentication Code Protocol*

CN *Correspondent Node*

CoA *Care-of Address*

DAD *Duplicate Address Detection*

DES-CBC *Data Encryption Standard - Cipher Block Chaining*

DHCP *Dynamic Host Configuration Protocol*

DHT *Ditributed Hash Table*

DNS *Domain Name Server*

EAP *Extensible Authentication Protocol*

EAP-TLS *EAP - Tranport Layer Security*

ESP *IP Encapsulating Payload*

essid *Extended Service-Set Identification*

FBAck *Fast Binding Update Acknowledgment*

FBU *Fast Binding Update*

FEEC Faculdade de Engenharia Elétrica e Computação

FNA *Fast Neighbor Advertisement*

FTP *File Transfer Protocol*

GMPLS *Generalized Multiprotocol Label Switching*

HA *Home Agent*

HoA *Home Address*

HTTP *HyperText Transfer Protocol*

HTTPS *HyperText Transfer Protocol Secure*

IAPP *Inter Access Point Protocol*

IETF *Internet Engineering Task Force*

IP ou IPv4 *Internet Protocol version 4*

IPv6 *Internet Protocol version 6*

KVM *Keyboard Video and Mouse*

LAN *Local Area Network*

LCA Laboratório de Engenharia de Computação e Automação Industrial

LCoA *Local Care-of Address*

LSA *Link-State Advertisement*

MAP *Mobility Anchor Point*

MAR *Mobility Aware Router*

MD5 *Message Digest 5*

MIPv6 *Mobile Internet Protocol version 6*

MM *Mobility Manager*

MN *Mobile Node*

MPA *Mobility Plane Architecture*

MPLS *Multiprotocol Label Switching*

NAR *New Access Router*

NCoA *New Care-of Address*

NETLMM *Network-based Localized Mobility Management*

OSPF *Open Shortest Path First*

P2MP *Point to multi-point*

P2P *Peer to Peer*

PAR *Previous Access Router*

PCoA *Previous Care-of Address*

PrRtAdv *Proxy Router Advertisement*

RA *Router Advertisement*

RADIUS *Remote Authentication Dial In User Service*

RCoA *Regional Care-of Address*

RtSolPr *Router Solicitation for Proxy Advertisement*

SMTP *Simple Mail Transfer Protocol*

SSL *Secure Socket Layer*

TCP *Transmission Control Protocol*

TKIP *Temporal Key Integrity Protocol*

TLS *Transport Layer Security*

TLV *Type, Length, Value*

VPN *Virtual Private Network*

WECA *Wireless Ethernet Compatibility Alliance*

WEP *Wired Equivalent Privacy*

Wi-Fi *Wireless Fidelity*

WLAN *Wireless Local Area Network*

WPA *Wi-Fi Protected Access*

WPA2 WPA versão 2

WPA2 *Wi-Fi Protected Access 2*

Capítulo 1

Introdução

Neste primeiro capítulo faremos uma breve introdução à tecnologia Wi-Fi (*Wireless Fidelity*) e os desafios provenientes do uso da mesma que nos serviram de motivação para esta dissertação. A seguir discutiremos um sub-problema que foi o foco deste trabalho: o acesso a rede Wi-Fi, tanto no ponto de vista da segurança quanto no ponto de vista de configuração exigida. Este trabalho teve como base uma arquitetura de mobilidade desenvolvida pela equipe da qual o autor faz parte na Faculdade de Engenharia Elétrica e Computação - Unicamp. Na seção 1.1 apresentamos as motivações deste trabalho e na seção 1.2 as contribuições para pesquisas nesta área.

1.1 Motivações

A tecnologia Wi-Fi teve suas origens no século XIX quando Heinrich Rudolf Hertz em 1888 utilizou os conhecimentos produzidos anteriormente por James Clerk Maxwell e Michael Faraday para demonstrar a propagação de ondas eletro-magnéticas. Em seu trabalho, Hertz mostrou como ondas eletromagnéticas poderiam ser transmitidas pelo ar e recebidas por um equipamento receptor que estivesse ao alcance destas ondas. Mais tarde, Nikola Tesla implementou as aplicações práticas da comunicação sem fio [1].

O precursor do Wi-Fi surgiu cem anos depois, em 1991, com tecnologia desenvolvida pela NCR Corporation e AT&T. Estes primeiros produtos de comunicação sem fio eram chamados de WaveLAN e tinham velocidades de 1 e 2 Mbps. Porém, a tecnologia ficou comprometida pela falta de compatibilidade entre os diversos fabricantes. Para resolver este problema, um grupo de empresas formado pela 3Com, Aironet (agora Cisco), Harris Semiconductor (agora Intersil), Lucent, Nokia and Symbol Technologies criou a WECA (*Wireless Ethernet Compatibility Alliance*), uma organização que visava a padronização da tecnologia de redes sem fio. A tecnologia de rede sem fio que seguia o padrão IEEE 802.11b passou a ter o nome comercial de Wi-Fi e a ser utilizada como o padrão de todos os dispositivos de comunicação sem fio dos integrantes da WECA, a qual mudou de nome para Wi-Fi Alliance em 2003. Qualquer dispositivo que tenha a marca registrada Wi-Fi possui a garantia de interoperar com outros dispositivos Wi-Fi.

Com o problema da interoperabilidade resolvido, a tecnologia Wi-Fi passou a ficar mais acessível e mais utilizada. A grande maioria dos dispositivos móveis - *notebooks*, celulares, *palmtops* - possuem uma interface de rede Wi-Fi. A praticidade de ter acesso a rede cabeada sem usar cabos *Ethernet*

proporciona a oportunidade de se acessar a Internet dentro de um táxi, por exemplo. Junto com a popularização do uso novos problemas e desafios foram surgindo.

A confidencialidade dos dados que trafegam no ar é, por definição, quase nula. Pouco se pode fazer para impedir que uma comunicação Wi-Fi seja capturada por um receptor malicioso. Alguns mecanismos de segurança usados nas redes cabeadas podem ajudar neste problema, como cifragem de dados no nível de enlace. Outra falha de segurança é a personificação do ponto de acesso à rede Wi-Fi. Em seu uso mais comum, o dispositivo sem fio se conecta a uma rede de acesso[1] a partir de um ponto de acesso. Em muitos casos esta associação entre dispositivo Wi-Fi e ponto de acesso ocorre somente utilizando-se o nome do ponto de acesso, seu essid (*Extended Service-Set Identification*). Um ponto de acesso clonado com o mesmo essid, porém com sinal mais forte, terá prioridade sobre o original para ser acessado, o qual levará o dispositivo a se conectar em um ponto de acesso não confiável. Trata-se de um ataque similar ao ataque de homem do meio, famoso problema de segurança em redes cabeadas.

A performance é outra preocupação. A comunicação Wi-Fi não consegue alcançar as baixas taxas de erro da conexão cabeada, porém são satisfatórias para aplicações convencionais. O ponto comprometedor é quando levamos em consideração a questão da mobilidade. Os pontos de acesso comerciais possuem alcance restrito, girando em torno de cem metros sem nenhum tipo de obstáculo, como paredes e móveis. Não é preciso um deslocamento grande para observarmos o enfraquecimento do sinal e a necessidade de uma troca de ponto de acesso para outro com sinal mais forte. Essa troca involve uma perda de informação pelo fato de que o enfraquecimento do sinal acarreta o aumento da taxa de erros na comunicação.

A troca do ponto de acesso não é instantânea: o processo de desassociação do ponto de acesso antigo e a associação ao ponto de acesso novo pode levar centenas de milisegundos. Esse tempo de associação gera uma perda de pacotes que não serão entregues neste intervalo de "escuridão". Esse processo é chamado de *handover*. Porém, após a reassociação ao novo ponto de acesso devemos levar em consideração que a camada de rede deverá se adaptar a nova posição deste nó móvel[2]. Para aplicações que exigem requisitos de banda mínima, como aplicações de voz sobre IP e videoconferências, os *handovers* causarão perdas de informação que comprometerão essas aplicações. Muitas propostas surgiram com o objetivo de reduzir o tempo de *handover* na camada de enlace ou otimizar o reroteamento na camada de rede, sempre visando reduzir ao máximo essa perda de informação.

O que estes dois problemas possuem em comum é onde eles são tratados. O ato de acessar uma rede Wi-Fi pode incluir trocas de credenciais entre nó móvel e rede de acesso, o que aumenta a confiabilidade. O nó móvel pode usar um programa adicional para acessar a rede, provendo funcionalidades extras que auxiliarão no momento do *handover*. O modo com o qual o acesso ocorre e os eventos envolvidos durante esse processo são alvo de estudo das propostas que procuram resolver os problemas de segurança e performance. Mudanças superficiais podem não fazer diferença significativa, mudanças radicais podem dificultar o acesso à rede Wi-Fi para os nós móveis com baixo poder computacional. Deve-se buscar o equilíbrio entre facilidade de aplicação da solução de acesso e ganho adicional que esta irá prover.

Outro motivador deste trabalho é a arquitetura MPA (*Mobility Plane Architecture*). Ela foi o resultado de um esforço conjunto de nossa equipe de pesquisadores em cooperação com a Ericsson

[1] Rede de acesso consiste em uma rede cabeada que oferece serviços a dispositivos Wi-Fi.

[2] Chamaremos de nó móvel todo dispositivo Wi-Fi que se movimenta e muda sua posição na topologia da rede de acesso.

do Brasil. Esta arquitetura ataca o problema da micro-mobilidade, isto é, mobilidade dentro de um domínio, onde um nó móvel troca de ponto de acesso em uma mesma rede. Esta arquitetura tem parâmetros rigorosos de performance a serem obedecidos. Para tal, deve-se prover maneiras de diminuir ao máximo o tempo de *handover* de camada 3 e, consequentemente, a quantidade de informação perdida. Estudar uma maneira de diminuir o tempo de *handover* na arquitetura MPA, via maneiras de acessar uma rede Wi-Fi que não exijam software adicional ou configuração exaustiva no nó móvel, foi outra grande motivação.

1.2 Contribuições

Neste trabalho exploramos o acesso a redes Wi-Fi de maneira prática, utilizando protocolos bem estabelecidos e procurando isentar o nó móvel de qualquer processamento adicional no acesso. Houve a preocupação com a performance de *handover* e exploramos maneiras de notificar a camada de rede sobre a ocorrência de *handover*.

As contribuições deste trabalho são:

- Um estudo prático das formas mais comuns de acesso a uma rede Wi-Fi, analisando diferentes padrões de segurança e utilizando *hardware* e protocolos convencionais.
- Proposição de técnicas para notificação de eventos (*triggers*) para a camada de rede na ocorrência de *handovers*.

1.3 Organização do texto

No capítulo 2 apresentamos uma breve revisão bibliográfica, apresentando as tecnologias e propostas estudadas que auxiliaram na elaboração deste trabalho. No capítulo 3 apresentamos o procedimento de acesso a redes sem fio nas mais diferentes situações levando em consideração diversos fatores, tais como facilidade de configuração e segurança provida. No capítulo 4 o acesso na arquitetura MPA será abordado, tanto em termos de projeto quanto de implementação. No capítulo 5 detalharemos propostas que visam melhorar o acesso a redes IP móveis. Finalmente, no capítulo 6, apresentamos os resultados obtidos e trabalhos futuros.

Capítulo 2

Mobilidade em redes IP

Neste capítulo apresentaremos uma breve revisão bibliográfica dos conceitos pertinentes a esta dissertação de mestrado. Nas seções seguintes apresentaremos as propostas de mobilidade e de segurança estudadas. Deixaremos um estudo mais detalhado do acesso a rede sem fio para o próximo capítulo.

2.1 Soluções de mobilidade no nível de rede

Um dos primeiros trabalhos concretos com o objetivo de otimizar a comunicação entre dispositivos móveis foi o MIPv6 (*Mobile Internet Protocol version 6*). Outros trabalhos surgiram para tratar de aspectos específicos do próprio MIPv6, como otimização de rota e *handover* rápido (*fast handover*). Outros trabalhos tratam da mobilidade em um escopo menor do que o MIPv6, focando mobilidade inter-domínios. É o caso do NETLMM (*Network-based Localized Mobility Management*) e da MPA (*Mobility Plane Architecture*).

2.1.1 *Mobile* IPv6

Para as redes IPv6 (*Internet Protocol version 6*), o IETF (*Internet Engineering Task Force*) propôs um padrão para a mobilidade de dispositivos móveis. Este padrão é chamado de Mobile IPv6 Protocol, ou simplesmente MIPv6 [2] [3] . Este protocolo começou a ser desenvolvido na metade da década de 1990, mas só foi aprovado para ser publicado como RFC em 2003. Espera-se que o MIPv6 seja uma componente padrão de qualquer equipamento que suporte IPv6. As mensagens do MIPv6 são tranportadas atráves do cabeçalho de mobilidade, um dos cabeçalhos de extensão do IPv6.

O objetivo desse protocolo é promover mobilidade de maneira transparente às aplicações, utilizando-se dos recursos que o IPv6 proporciona. A idéia é que um dispositivo IPv6 seja alcançável sempre pelo mesmo endereço IPv6, não importa onde ele esteja localizado. A princípio isto soa como uma quebra do modelo de roteamento do próprio IPv6, pois não seria possível rotear pacotes para um dispositivo cujo endereço não pertence ao escopo de endereçamento de sua rede atual. Para contornar este problema, novos conceitos foram criados:

Nó móvel ou *Mobile Node* **(MN):** qualquer dispositivo que se movimenta dentro da topologia da Internet. Esta mobilidade não é causada exclusivamente por mudança de localização física; qualquer

evento que faça com que este dispositivo mude de link de acesso é caracterizado como mobilidade.

Nó correspondente ou *Correspondent Node* **(CN):** qualquer dispositivo que se comunique com um nó móvel. Ambos não são exclusivos: um nó móvel pode atuar também como CN, dependendo do contexto.

Endereço nativo ou *Home Address* **(HoA):** é o endereço estável e permanente por meio do qual os CNs vão se comunicar com os nós móveis.

Link nativo ou *home link***:** link no qual o dispositivo adquiriu seu HoA.

Agente nativo ou *Home Agent* **(HA):** roteador que atua como o representante do nó móvel na rede nativa. Ele redireciona todos os pacotes endereçados ao HoA de um nó móvel para o seu endereço atual (care-of address) usando tunelamento IP-IP.

Link estrangeiro ou *foreign link***:** qualquer link que não seja o link nativo onde o dispositivo móvel se associa.

Care-of address **(CoA):** endereço dado a um nó móvel quando visita um link estrangeiro. Trata-se de um endereço IPv6 como qualquer outro. Este endereço corresponde à atual posição do nó móvel na topologia da Internet.

O funcionamento do MIPv6 baseia-se na interação entre o nó móvel e o seu HA. O nó móvel tem um (ou mais) HoA, os quais serão usados pelos CNs que queiram se comunicar com este nó móvel. Enquanto o nó móvel estiver em seu link nativo o roteamento de pacotes entre este dispositivo e qualquer outro segue às vias do roteamento IP padrão.

Quando o nó móvel muda de localização e se associa a um link estrangeiro, este deverá necessariamente adquirir um endereço IPv6 válido nesta nova rede, caso contrário não terá conectividade com o resto da Internet. Para continar a ser alcançável pelo seu HoA ele deve avisar o seu HA de seu endereço atual na topologia da Internet, o seu CoA. O HA então associa o HoA deste nó móvel com o CoA enviado por ele, criando um túnel entre eles. A partir deste momento, qualquer pacote enviado para o endereço HoA será recebido por este HA, o qual os encaminhará pelo túnel criado para o nó móvel. De maneira similar, qualquer pacote enviado pelo nó móvel para este CN em questão passará pelo HA. Esta estratégia garante a mobilidade transparente para as aplicações fim a fim, pois os sockets que já estavam abertos durante a movimentação do nó móvel não precisam ser alterados.

O funcionamento básico do MIPv6, descrito acima, é composto a partir de três funcionalidades: configuração de endereço e encaminhamento de mensagens, detecção de movimento e otimização de rota. A combinação destas compõem praticamente todo o protocolo MIPv6.

Configuração de endereço e encaminhamento de mensagens

A configuração de endereços no IPv6 pode ser sem estado (*stateless*) ou com estado (*stateful*). No caso *stateless*, o nó móvel se autoconfigura usando informações que recebe do link onde se encontra. No caso *stateful* este tem que buscar parâmetros de configuração em um servidor. Devemos observar que qualquer um destes métodos de configuração de endereço é válido para o MIPv6. Entraremos em mais detalhes destes dois métodos de configuração de endereço mais adiante.

O processo de autoconfiguração de endereço do IPv6 consiste em usar o identificador da interface de rede do nó móvel com o prefixo de sub-rede divulgado pelo roteador daquele domínio. Isto é feito através de uma mensagem chamada Divulgação de Roteador, ou *Router Advertisement* (RA). Esta mensagem carrega, dentre outros parâmetros, informações sobre o prefixo da sub-rede e roteador padrão. O identificador da interface de rede nada mais é do que um tipo de endereço IPv6 não-roteável

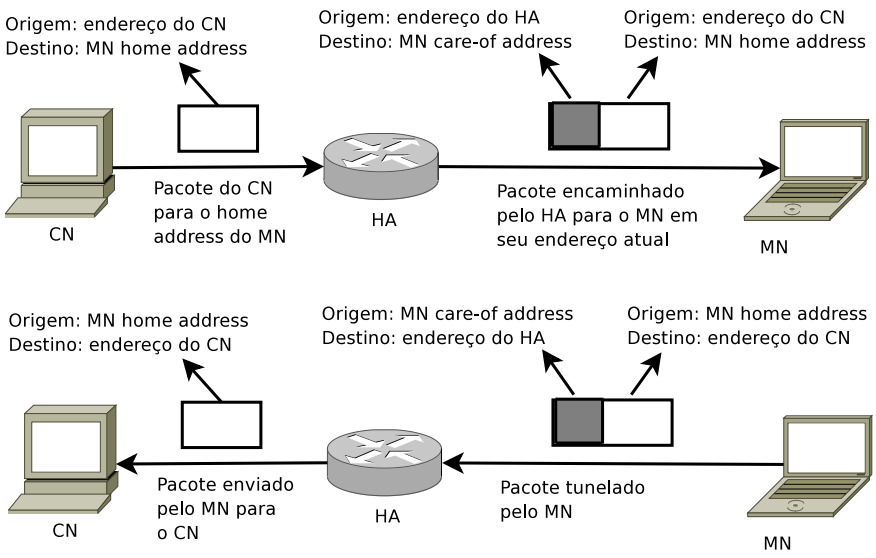

Figura 2.1: Encaminhamento de pacotes via tunelamento no MIPv6

chamado link-local address, o qual é formado a partir do MAC da interface de rede. Este processo de autoconfiguração é completamente transparente ao usuário. Nas distribuições Linux, o módulo IPv6 configura automaticamente um endereço para qualquer interface de rede se o computador receber uma mensagem de RA na mesma.

Quando o nó móvel está em seu link nativo ele atua como um dispositivo IPv6 padrão, recebendo os pacotes endereçados a ele via roteamento padrão. Se ele se encontra em um link estrangeiro, este configura um CoA baseando-se no prefixo divulgado no link. Após a configuração, é necessário avisar o seu respectivo HA de seu novo endereço. Isto é feito através de uma mensagem de *Binding Update* (BU). Ao receber esta mensagem, o HA passará a defender o endereço do nó móvel, agindo como um proxy. Ele divulga no link nativo que vai representar o HoA deste nó móvel através de uma mensagem de Divulgação de Vizinho Proxy, ou *Proxy Neighbor Advertisement*. Desta forma, qualquer pacote que for endereçado ao IP do nó móvel será encaminhado para o HA, o qual cuidará de encaminhar o pacote para o endereço atual do nó móvel.

Enquanto estiver em um link estrangeiro, o nó móvel deverá, de tempos em tempos, mandar uma mensagem de BU para o seu HA, atualizando o estado de seu mapeamento no seu link nativo. A mensagem de BU possui um parâmetro de tempo de vida, ou tempo de validade deste estado. O nó móvel deve mandar novas mensagens de BU para o HA em um intervalo de tempo menor do que esta validade, caso contrário o seu agente nativo poderá invalidar este estado e parar de encaminhar os pacotes para o nó móvel. O HA também pode forçar uma atualização deste estado enviando um Binding Refresh Request para o nó móvel.

Este encaminhamento de mensagens entre o nó móvel e o seu respectivo HA é feito através de um túnel bidirecional cujas pontas são, respectivamente, o endereço do HA e o endereço CoA do nó móvel. Este mecanismo oculta a mudança de endereço do nó móvel aos seus respectivos CNs, que continuarão a se comunicar com o mesmo através de seu endereço HoA. A figura 2.1 ilustra como que o encaminhamento de pacotes ocorre de maneira transparente para o CN.

Detecção de movimento

O nó móvel utiliza as mensagens de RA que são divulgadas nos links pelos roteadores de acesso para detectar uma mudança de ponto de acesso. Se nesta mensagem contiver um prefixo de sub-rede diferente do anterior, houve uma mudança de link. A partir deste momento o nó móvel deve configurar o CoA e enviar uma mensagem de BU notificando-o de sua mudança de endereço. Este é o método clássico de detecção de movimento previsto pelo MIPv6.

Devemos observar que esta detecção de movimento é feita em camada 3, pois houve uma mensagem IPv6 que chegou no nó móvel que o notificou da mudança de link. Outros métodos de detecção são possíveis se considerarmos a camada 2, a qual cuida de eventos do enlace aéreo. Porém não é o escopo do MIPv6 utilizar eventos ou recursos das camadas inferiores para antecipar a detecção de movimento. Entretanto, o MIPv6 proporciona algumas alternativas para a redução deste tempo de handover.

Quando um nó móvel está associado a um determinado link, mas possui acesso a diferentes links de diferentes sub-redes, este pode configurar antecipadamente os CoAs para cada um destes links. Se o nó móvel detectar que mudou para um destes links, o endereço já está configurado, ganhando desta forma o tempo que seria usado para a configuração do endereço.

Para diminuir a perda de pacotes, o nó móvel pode enviar um BU para o HA do link anterior para que encaminhe o tráfego destinado ao seu CoA antigo para o seu CoA novo. Seria como usar o roteador da rede anterior como seu HA temporário. Uma outra alternativa é o que chamamos de bicasting. O nó móvel se registra em mais de um roteador de acesso. Assim, qualquer tráfego com destino ao nó móvel será replicado em diversos roteadores de acesso diferentes. Ao mudar de link, o nó móvel já estará recebendo tráfego se ele já havia se registrado naquele roteador de acesso. Mesmo sendo eficiente e simples, esta opção gera muito desperdício de banda, uma vez que o tráfego será replicado e só um dos roteadores de acesso entregará os pacotes, pois o nó móvel só estará associado a um roteador de acesso por vez.

Outras propostas apresentadas a seguir tentam reduzir o tempo e a perda de pacotes relativa ao handover no MIPv6.

Otimização de rota

O encaminhamento de pacotes para o nó móvel atráves do HA pode implicar em grandes atrasos. Conforme visto na figura 2.1, há o encaminhamento do pacote do CN para o HA e depois do HA para o nó móvel. Existem situações onde esta rota pode ser severamente ineficiente. Considere o caso onde o CN e o nó móvel encontram-se na mesma sub-rede. Os pacotes que sairem do CN com destino ao HoA do nó móvel vão sair da sub-rede, chegar até o HA e serão encaminhados de volta para a mesma sub-rede.

A otimização de rota proposta pelo MIPv6 consiste em informar o CN do CoA do nó móvel, tirando assim o HA do processo de encaminhamento. Desta forma, o CN atuaria de maneira similar ao HA - usando um mapeamento HoA - CoA, este enviará os pacotes diretamente para o CoA do nó móvel via tunelamento, porém os pacotes enviados devem conter o endereço HoA do nó móvel. Desta forma, a mudança de endereço continuará sendo transparente às camadas superiores.

2.1.2 HMIPv6 - *Hierarchical* MIPv6

O MIPv6 [2] [3] tem como proposta resolver o problema da mobilidade quando um nó se movimenta entre diferentes redes de acesso, mudando de IP na topologia da internet. Neste caso, a alcançabilidade do mesmo pelo seu HA muda, pois o gateway da outra rede de acesso será diferente.

Porém nem todos os *handovers* serão feitos de uma rede de acesso a outra. É comum em redes de grande porte mais de um ponto de acesso servir à mesma rede, com mudança de endereço apenas dentro da mesma. O nó móvel continua a ser alcançável externamente pelo mesmo gateway. No MIPv6, este caso é um *handover* como outro qualquer: existe um tráfego de controle entre o nó e o seu HA, onde o nó registra o seu novo CoA no HA e há um novo tunelamento de tráfego. Neste caso em especial, não há mudanças no roteamento do HA para alcançar o nó móvel; ele está na mesma rede, portanto o gateway continua o mesmo. Foi gerado um tráfego para fora da atual rede de acesso do nó móvel que poderia ser evitado, além da latência gerada pelo processo de se comunicar com o HA que se encontra em uma outra rede. Com o crescimento do uso das redes sem fio, será cada vez mais comum este cenário de termos mais de um roteador de acesso em uma mesma rede, aumentando o número de *handovers* executados dentro da mesma rede de acesso. Para estes *handovers* em especial, um tratamento local de mobilidade seria mais apropriado.

Pensando nesta otimização, novas extensões do MIPv6 surgiram com o intuito de melhorar o tratamento dado a estes casos, os quais são denominados micro-mobilidade. Eles coexistem com o MIPv6, uma vez que este resolve o problema da macro-mobilidade[1]. Uma destas propostas é o HMIPv6, ou *Hierarchical* MIPv6.

No HMIPv6 existem algumas alterações no MIPv6, em especial no cliente (nó móvel) e na rede de acesso. O nó móvel passa a ter dois endereços, RCoA (*Regional Care-of-Address*) e LCoA (*on-Link Care-of-Address*). RCoA é o endereço que o nó móvel recebe ao entrar na nova rede de acesso e, consequentemente, o endereço que será enviado pela mensagem de *Binding Update* para o HA. LCoA é o endereço que é configurado no nó móvel usando o prefixo do seu roteador de acesso atual, de maneira similar ao que acontece no MIPv6. Assim que o nó móvel chegar na rede, RCoA será igual ao LCoA. Quando o nó móvel realizar *handover* para um outro roteador de acesso desta mesma rede, ele configurará um novo LCoA, porém seu RCoA permanecerá o mesmo, não havendo nenhuma comunicação com o HA. Todavia, o nó móvel receberá pacotes em LCoA, e não em RCoA. Existe a necessidade de algum mecanismo que permita que os pacotes que endereçados à RCoA sejam encaminhados ao nó móvel com endereço atual LCoA.

Neste contexto surge a figura do MAP (*Mobility Anchor Point*). Introduzido pelo HMIPv6 na rede de acesso, este elemento fará o mapeamento entre RCoA e LCoA localmente[2]. Ao mudar de roteador de acesso, o nó móvel dispara uma mensagem de *Binding Update*, assim como no MIPv6, entretanto com algumas modificações. A mensagem possui uma nova *flag* adicionada pelo HMIPv6, a qual indica que a mensagem de BU trata-se de um registro em um MAP e não deve ser encaminhada para fora da rede. O campo de HoA (*Home Address*) contém o endereço RCoA e o campo CoA o endereço LCoA. Quando este BU chega no MAP correspondente, este fará o mapeamento RCoA - LCoA e garantirá que todos os pacotes endereçado a RCoA sejam tunelados por ele para o nó móvel no endereço LCoA. Este comportamento é similar ao comportamento do HA no MIPv6, o que nos permite chamar o MAP de *Home Agent* local. Uma vez que o mapeamento e tunelamento é feito pelo

[1] Handover entre redes de acesso diferentes

[2] Dentro da atual rede de acesso do nó móvel.

MAP, não há mudanças no HA, economizando tráfego e diminuindo o tempo de handover. Esta é uma das vantagens do HMIPv6: não há mudanças na implementação do MIPv6 no HA e nos CNs. A desvantagem é a necessidade de adicionar funcionalidades à implementação do nó móvel.

Deste funcionamento básico, outras funcionalidades do HMIPv6 estão concentradas na descoberta e seleção dos MAPs. Faremos uma breve descrição destas funcionalidades. Maiores detalhes podem ser encontrados em [4].

Descoberta de MAPs

Assim que um nó móvel entra em um domínio com o suporte a MAPs, este deve ser notificado a respeito dos MAPs locais que servem aquele domínio. Esta notificação é feita basicamente de duas formas. Uma delas é usar as mensagens de *Neighbor Discovery*[3] do IPv6 para propagar dados sobre o MAP. Isto é feito inserindo-se nesta mensagem uma extensão com o formato TLV (*Type, Length, Value*) com o endereço global do MAP. A outra forma é utilizar mensagens de RA (*Router Advertisement*) para informar os MAPs presentes no domínio.

Seleção de MAPs

O HMIPv6 permite que um nó móvel se registre em mais de um MAP, de forma que as mensagens entre um *correspondent node* e o nó móvel percorram sempre o caminho mais curto possível. Em um cenário onde a rede de acesso é grande, mais de um *gateway* de entrada é usado pelos CNs para se comunicar com os nós móveis na rede e, dependendo da proximidade e da topologia, pode-se usar diferentes gateways. Um nó móvel que possui CNs diferentes pode se registrar em MAPs próximos aos gateways de saída, de maneira que o caminho entre o CN e o nó móvel seja o mais curto possível.

2.1.3 FMIPv6 - *Fast handover for* MIPv6

O *Fast handover for* MIPv6 (FMIPv6) [2] é uma extensão do MIPv6 que tenta diminuir a perda de pacotes durante o processo de *handover*. Se o nó móvel conseguir manter suas antigas conexões enquanto o seu processo de *handover* não foi concluído, a perda de pacotes é minimizada. Para tanto, é necessário que o nó móvel receba pacotes no seu endereço antigo, chamado de PCoA (*Previous Care-of Address*), enquanto o seu novo endereço, NCoA (*New Care-of Address*), não está totalmente configurado.

Para que isto seja possível, uma das alternativas é a criação de um túnel entre o antigo roteador de acesso do nó móvel denominado *Previous Access Router* (PAR) e o novo, denominado *New Access Router* (NAR). A partir deste túnel, o PAR pode encaminhar os pacotes endereçados em PCoA para o nó móvel em NCoA.

O FMIPv6 inicia sua operação com o nó móvel pedindo informações sobre os outros roteadores de acesso vizinhos através de uma mensagem RtSolPr (*Router Solicitation for Proxy Advertisement*). Em resposta, o roteador de acesso (futuro PAR) envia um PrRtAdv (*Proxy Router Advertisement*), uma mensagem que contém dados sobre os links vizinhos. As informações fornecidas são, para cada link vizinho:

[3] Esta mensagem faz parte do procedimento de descoberta de vizinhança do IPv6, o qual não entraremos em detalhes. Para maiores informações consultar [2].

- Endereço L2 do roteador de acesso;
- Endereço IPv6 do roteador de acesso;
- Prefixo de sub-rede válido.

A partir destas informações, o nó móvel consegue prever qual será o seu endereço IPv6 caso ele migre para aquele link. Ao perceber que está se movimentando em direção a um link específico[4], este configura um endereço NCoA a partir dos dados que ele conseguiu de seu PAR. Logo após, o nó móvel envia uma mensagem de FBU (*Fast Binding Update*) para o PAR, de maneira que este associe PCoA e NCoA. Desta forma, o roteador de acesso antigo pode encaminhar os pacotes para o nó móvel em sua nova localização.

O PAR, ao receber o FBU, realiza a associação e envia uma mensagem de FBAck (*Fast Binding Update Acknowledgment*) para o nó móvel, confirmando o processamento do FBU. Todavia o nó móvel estava no processo de *handover*, e há chances deste não receber o FBAck antes de ter saído do link. O nó móvel deve estar preparado para agir de maneira adequada a qualquer uma das 2 situações:

- Se o FBAck foi recebido ainda no link antigo, então o encaminhamento de pacotes já estará ativo quando o nó móvel fizer a migração. Neste caso, assim que se associar ao NAR, o nó móvel envia uma mensagem de FNA (*Fast Neighbor Advertisement*)[5] para confirmar o uso de seu endereço NCoA. Assim, os pacotes armazenados em buffer no NAR ou que estiverem chegando poderão ser entregues ao nó móvel.
- Se o FBAck não foi recebido pelo nó móvel, este não pode considerar que o processamento do FBU foi feito de maneira correta[6]. Nesta situação, o nó móvel deve enviar a mensagem de FBAck para o PAR assim que sua associação com o NAR estiver completa.

Em qualquer uma das situações, a probabilidade de perda de pacotes no PAR ou no NAR é grande. No primeiro, quando a mensagem de FBAck não foi processada antes do nó móvel sair do link do PAR; no segundo, quando os pacotes são encaminhados para o NAR antes do nó móvel estar apto para recebê-los. Nos dois casos, a bufferização dos pacotes nos roteadores de acesso é necessária para evitar tais perdas.

O FMIPv6, assim como o HMIPv6, não exige mudanças na implementação do MIPv6 no HA e nos CNs, mas requer alterações no nó móvel e nos roteadores de acesso.

[4]Essa predição pode ser feita usando várias heurísticas, como por exemplo o nível de potência do sinal das antenas ao redor do nó móvel

[5]A mensagem de *Neighbor Advertisement* no IPv6 tem como função divulgar o endereço recém-configurado pelo nó móvel pelo link, de maneira que se possa detectar endereços duplicados. O FNA nada mais é do que uma adaptação desta mensagem para o FMIPv6.

[6]Pode ser o caso da mensagem de FBU ter sido enviada tarde demais, quando não havia mais conectividade entre o nó móvel e o PAR.

2.1.4 NETLMM - *Network-based Localized Mobility Management*

As duas extensões do MIPv6 descritas anteriormente (HMIPv6 e FMIPv6) são claras evidências de que tratar a mobilidade como um único processo em qualquer situação (como o MIPv6 faz) é ineficiente e de difícil implantação. Além de tornar o *handover* local ineficiente, devemos levar em consideração que as políticas administrativas de domínios geograficamente e topologicamente separados são diferentes, o que gera problemas na aceitação de um único modelo global a ser usado por todos. A separação das soluções de mobilidade entre soluções para a mobilidade global e mobilidade local permite uma evolução mais fácil de ambas, além de facilitar a adoção dos mesmos - um modelo de mobilidade local deve ser independente e coexistir com qualquer modelo de mobilidade global, e vice-versa.

A adoção generalizada de switches WLAN (*Wireless Local Area Network*), os quais resolvem o problema da mobilidade e sem envolvimento do nó móvel prova que um modelo de mobilidade local que não necessite de software adicional nos dispositivos móveis é uma solução que seria aceita e de fácil implantação nas redes sem fio atuais. Uma das alternativas para cumprir este objetivo é fazer com que a mobilidade local seja resolvida pela própria rede de acesso. Neste contexto surgiu o grupo de trabalho do NETLMM: desenvolver um protocolo para o gerenciamento de mobilidade local, inicialmente em redes IPv6, que cumpra com estes requisitos.

A arquitetura concebida pelo NETLMM propõe um novo elemento na rede de acesso chamado MAP (*Mobility Anchor Point*), o qual será responsável por manter um conjunto de rotas para os dispositivos móveis. Estas rotas indicam em qual roteador de acesso o nó móvel se encontra no momento. Pacotes endereçados para os nós móveis são encaminhados para o MAP correspondente, da mesma forma que este encaminha os pacotes com origem nos nós móveis para outros endereços.

Em um *handover*, o roteador de acesso que receber o nó móvel envia uma mensagem de atualização de rota para o MAP correspondente, o qual vai redirecionar os pacotes para o nó móvel em questão para este novo roteador de acesso. A participação do nó móvel neste processo é mínima; não existe nenhum protocolo especial para suportar esta mobilidade local. O nó móvel obrigatoriamente deverá ser detectado pelo novo roteador de acesso, seja por um processo de autenticação, configuração de endereço ou outros meios[7].

O protocolo e arquitetura propostos pelo NETLMM têm similaridades com a proposta do HMIPv6, com algumas diferenças. Ambos definem um elemento chamado ponto de âncora de mobilidade, ou MAP, que resolve o problema da mobilidade local. Porém, a proposta do NETLMM é o desacoplamento da mobilidade local e global, o que resulta em independência do protocolo de mobilidade global. Isto não acontece com o HMIPv6, que depende do MIPv6. Outro ponto interessante é a carga no nó móvel: enquanto o HMIPv6 requer alteração na implementação do MIPv6 nos nós móveis, o NETLMM não vai exigir nenhuma carga ou software adicional nos dispositivos móveis - o que é um caminho a mais para a adoção e implantação da arquitetura.

[7]No IPv6, por exemplo, o nó pode enviar uma mensagem de *Router Solicitation* ao migrar para outro roteador de acesso. Esta mensagem pode ser aproveitada pelo roteador de acesso como evidência de que o nó móvel realizou *handover*.

2.1.5 MPA - *Mobility Plane Architecture*

A MPA é uma arquitetura que foi resultado de várias pesquisas realizadas em cooperação entre a Ericsson e a FEEC (Faculdade de Engenharia Elétrica e Computação). Essas pesquisas consistiram em estudar soluções para mobilidade, as quais as mais relevantes para este trabalho já foram apresentadas. Todas elas apresentam deficiências em comum, o que nos levou à proposição da MPA.

Problemas com as soluções existentes

Uma das características que as soluções existentes têm em comum é a sobrecarga de *software* necessária no dispositivo móvel. Embora *notebooks* tenham capacidade de memória e processamento para realizar estas tarefas adicionais, muitos outros dispositivos com menor capacidade não dispõem de tal capacidade. Com o passar do tempo o hardware destes dispositivos evoluirão, até chegar em um ponto onde essa limitação será desconsiderada. Todavia, ainda existirá o trabalho de instalar e configurar um *software* adicional no dispositivo. Nenhuma das soluções de micro-mobilidade apresentadas teve a preocupação real de aliviar o processamento por parte do nó móvel, de maneira a minimizar a quantidade de configuração necessária no mesmo.

A outra característica é o suporte ao IPv6. Há alguns anos atrás acreditava-se que o IPv4 seria substituído pelo IPv6 devido ao esgotamento dos endereços disponíveis. O IPv6 veio resolver este problema, com um espaço de endereçamento de 128 bits ao invés dos 32 bits do IPv4, além de resolver outros problemas encontrados no IPv4, em especial em relação a segurança. O que vemos hoje é que o IPv4 continua em crescimento, graças a soluções como o CIDR (*Classless Inter Domain Routing*) e NAT (*Network Address Translation*), que ampliaram o espaço de endereços do IPv4. É consenso que as redes ainda utilizarão IPv4 por algum tempo, pois não há hoje motivação que justifique o custo elevado de uma transição para IPv6 de toda a infraestrutura existente hoje[8].

Estes dois problemas somados tornam as propostas de micro-mobilidade existentes hoje de difícil implantação nas redes móveis atuais, seja por não suportarem IPv6, seja exigindo que cada nó móvel que acesse a rede instale *software* adicional para usufruir totalmente dos serviços da mesma.

Princípios básicos da MPA

Nosso foco em uma solução de micro-mobilidade ao invés de macro-mobilidade deve-se a dois motivos principais. O primeiro é a dificuldade de propor um modelo inter-domínios único para a mobilidade. Cada domínio possui suas próprias políticas de acesso e segurança, o que dificulta um esquema que se adeque a todos de maneira satisfatória. O segundo é o tamanho de cada domínio. Com o barateamento dos pontos de acesso, será cada vez mais comum o nó móvel passar por uma quantidade maior de *handovers* dentro do mesmo domínio, o que torna crítico o suporte à micro-mobilidade. Devemos lembrar que a utilização de um esquema de micro-mobilidade não descarta o uso de um esquema de macro-mobilidade no caso de *handovers* entre domínios.

Outro objetivo alcançado foi a independência do protocolo IPv6. A partir do nosso modelo funcional, abstraímos a tecnologia de rede usada, flexibilizando nossa arquitetura. MPA fornece serviços de

[8] O continente norte-americano possui uma grande porção espaço de endereçamento do IPv4, enquanto que no caso da Ásia, por exemplo, a fatia é escassa. Acredita-se que os países emergentes da Ásia vão impulsionar a adoção do IPv6, obrigando que outros países adotem o IPv6.

micro-mobilidade tanto para redes IPv4 quanto para MPLS (*Multiprotocol Label Switching*), GMPLS (*Generalized Multi Protocol Label Switching*) e até mesmo IPv6.

Nos estudos preliminares, observamos que poucas soluções de mobilidade se preocuparam com a quantidade de configuração e de *software* nos dispositivos móveis. No projeto da MPA optamos por distribuir a inteligência e carga de processamento da arquitetura nos elementos de rede, poupando o nó móvel de *softwares* adicionais e de configurações exaustivas.

MPA consiste na utilização de uma rede *overlay* sobre a rede de acesso, na qual todo o tráfego para os nós móveis é direcionado para os respectivos roteadores de acesso através de túneis ponto-multiponto. O uso de túneis ponto-multiponto permite o que chamamos de *bicasting*: duplicação do tráfego com destino ao nó móvel para mais de um roteador de acesso, de forma a minimizar a perda de pacotes no momento do *handover*.

Modelo funcional

A figura 2.2 mostra os elementos básicos do modelo funcional da MPA.

- **Rede de transporte:** Rede IPv4, IPv6, MPLS ou GMPLS na qual deseja-se prover serviços de micro-mobilidade.
- **Nó móvel:** Qualquer dispositivo Wi-Fi que muda de ponto de acesso com o passar do tempo.
- **Mobility Aware Router (MAR):** Roteador da rede de transporte com funções relacionadas a mobilidade, como redirecionamento de tráfego destinado ao nó móvel.
- **MAR de ingresso:** MAR que também faz o papel de *gateway* da rede de transporte com o mundo externo. É necessário que ele sempre seja o *gateway* da rede para que este redirecione o tráfego para o nó móvel em caso de mudança de ponto de acesso.
- **MAR de egresso:** MAR que faz a comunicação entre a rede cabeada e a rede Wi-Fi. É necessário que todos os roteadores de acesso sejam MARs para que a detecção de movimento do nó móvel aconteça.
- **MAR de ramificação:** MAR que se localiza no ponto de ramificação entre a rota antiga e a nova rota que o tráfego em direção ao nó móvel deve seguir na rede de acesso.
- **Túnel ponto-multiponto (*Point to multi-point tunnel*, ou P2MP *tunnel*):** Túnel de encapsulamento de tráfego para os nós móveis. Na MPA ele é formado a partir de um MAR raiz, pode possuir um ou mais MARs de ramificação, e possui um ou mais MARs de egresso. Assemelha-se ao formato de uma árvore com a raiz no MAR de ingresso, alguns MARs de ramificação como caules e os MARs de egresso sendo as folhas.
- **Rede móvel overlay:** Rede lógica construída a partir de um ou mais túneis ponto-multiponto, a qual é responsável pelo encaminhamento do tráfego com destino aos nós móveis.

Além dos elementos básicos, definimos quatro blocos funcionais que compõem a arquitetura:

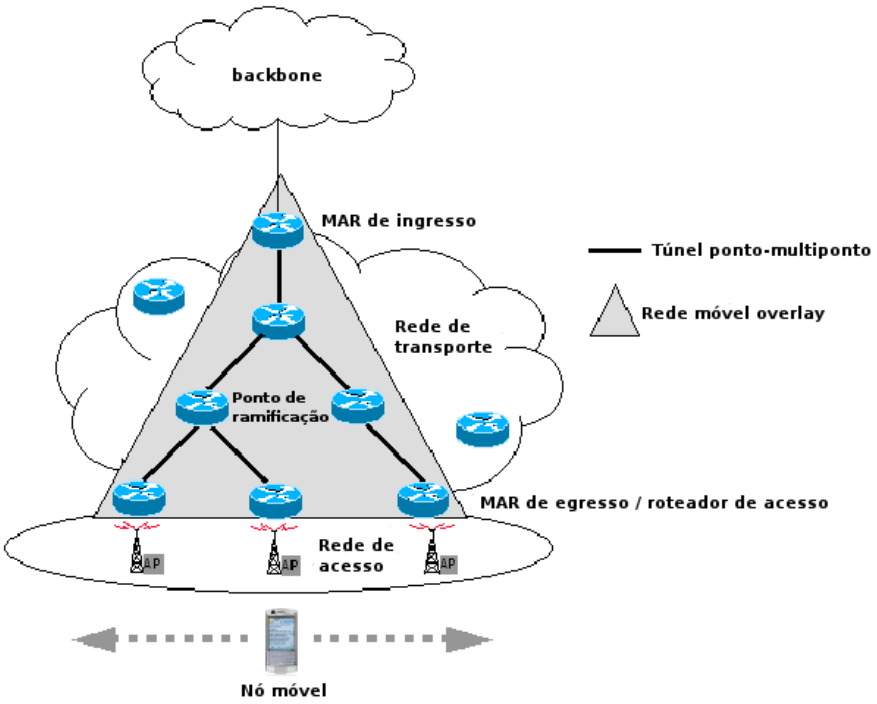

Figura 2.2: Elementos básicos do modelo funcional da MPA

- **Gerenciamento de Túneis (GT):** Este bloco funcional é responsável pelo estabelecimento, desativação e re-roteamento de túneis P2MP. Para tal, é necessário que este disponibilize interfaces de interação com o sistema de gerenciamento de rede e também com operadores humanos. Gerenciamento de túneis é uma atividade desempenhada pelos MARs.

- **Roteamento Móvel (RM):** O RM rastreia a posição atual do nó móvel (respectivo roteador de acesso) e atua sobre os MARs de maneira que o tráfego seja redirecionado para a nova posição do nó móvel. Função é desempenhada exclusivamente pelos MAR.

- **Configuração de Endereço (CE):** Esta entidade é responsável por configurar um endereço de camada 3 para o nó móvel no momento que ele se conecta a rede ou quando muda de rede de acesso. A configuração de endereço é um procedimento no qual tanto a rede em si (MARs) quanto o nó móvel cooperam.

- **Facilitador de *Handover* (FH):** Este bloco funcional tem como objetivo prover serviços que facilite o processo de *handover* do nó móvel dentro da mesma rede de acesso. Isso inclui notificação de associação ao MAR de egresso, o que chamamos de *trigger*, sinalização referente ao *handover* e associação segura ao ponto de acesso. Note que este bloco funcional envolve MARs, pontos de acesso, servidores de autenticação e o próprio nó móvel.

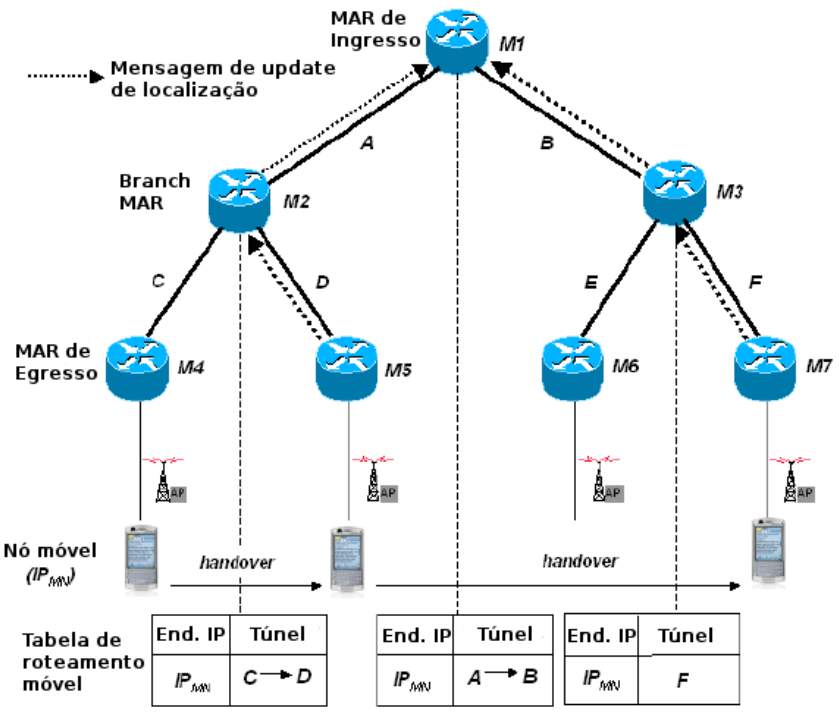

Figura 2.3: Ilustração do funcionamento básico da MPA

Funcionamento básico da arquitetura

A figura 2.3 ilustra o cenário básico de funcionamento da arquitetura. Consideremos o caso de um nó móvel que acabou de se conectar na rede de acesso. Ele realiza a associação com o ponto de acesso e configura seu endereço de camada 3 - atividades dos blocos lógicos HH e CE respectivamente - para estabelecer conexões e receber tráfego. Nesse momento, o MAR de egresso M4 envia uma mensagem de *update* de localização a qual notifica os MARs de nível hierárquico maior na rede overlay que o nó móvel de endereço L3 IPmn é acessível via M4. Esta mensagem fará a atualização do bloco funcional RM, atualizando as tabelas de roteamento móvel dos MARs que a receberam, de maneira a encaminhar o tráfego com destino ao nó móvel corretamente.

No primeiro *handover* (M4-M5) o novo MAR de egresso do nó móvel envia a mensagem de notificação para os MARs de nível superior, similar ao que o MAR M4 fez anteriormente. Neste *handover* em especial, o MAR M2 faz o papel de MAR de ramificação, pois é o exato ponto de ramificação entre M4 e M5. Ao receber a mensagem de notificação, M2 atualiza sua tabela de roteamento móvel e encaminha a notificação para M1, o qual não necessita atualizar sua tabela.

No segundo *handover*, M7 passa a ser o roteador de acesso do nó móvel. De maneira similar aos dois casos anteriores, a mensagem de notificação é disparada para atualização do bloco RM. Porém desta vez o MAR de ramificação passa a ser M1, o qual deve atualizar sua tabela de roteamento móvel apropriadamente.

No caso do MAR M2, ponto de ramificação do *handover* anterior, as funcionalidades do bloco funcional GT vão eliminar a entrada para IPmn na tabela de M2. As entradas na tabela de roteamento móvel são do tipo *soft-state*, ou seja, são apagadas se ocorrer a ausência de mensagens de *update* de

localização após um certo período de tempo pré-estabelecido. Uma maneira de forçar que o nó móvel tenha esse comportamento é estabelecer endereços de camada 3 de duração curta, o que forçará o nó móvel a renovar o seu endereço e, por conta disso, disparar mensagens de *update* de localização.

Observe que a partir deste funcionamento básico diversas extensões são possíveis. Por exemplo, estratégias de engenharia de tráfego para determinar a melhor topologia para a rede overlay, políticas para controlar o *soft-state* das entradas da tabela de roteamento móvel e otimizações no processo de *handover*. Embora algumas dessas extensões possa exigir algum *software* adicional no nó móvel, o funcionamento básico da arquitetura concentra toda a inteligência na rede de acesso, exigindo o mínimo possível dos clientes móveis.

Possibilidades de protocolos para os blocos funcionais

A partir da descrição funcional da arquitetura podemos definir quais os requisitos para a implementação de cada bloco funcional e exemplificar alguns protocolos que cumprem tais requisitos.

- **Protocolo de gerenciamento de túneis:** O protocolo deve permitir o estabelecimento de túneis P2MP pela rede de acesso. Alguns protocolos que suportam esse tipo de operação são o RSVP-TE (*Resource Reservation Protocol - Traffic Engineering*) e o SNMP (*Simple Network Management Protocol*).

- **Protocolo de roteamento móvel:** É necessário ter um protocolo exclusivo para o roteamento da rede overlay, o qual difere do roteamento usado na rede de acesso. O objetivo deste protocolo é atualizar a tabela de roteamento móvel de todos os MARs com a posição atual do nó móvel.

  Uma solução é divulgar a posição atual do nó móvel como um atributo de enlace de um protocolo de roteamento convencional, como o OSPF (*Open Shortest Path First*). Pode-se usar um LSA (*Link-State Advertisement*) opaco para divulgar essa informação, a qual deverá ser interpretado de maneira correta pelo MAR.

  Uma outra possibilidade é aproveitar a sinalização do protocolo de gerenciamento de túneis. No caso do RSVP-TE, as mensagens de RESV[9] podem carregar um objeto opaco que contém informações sobre a localização dos nós móveis.

- **Protocolo de configuração de endereço:** A configuração de endereço pode ser tanto sem guardar estado (*stateless*) quanto guardando estado (*stateful*)[10]. O protocolo de configuração de endereço *stateful* mais usado é o DHCP (*Dynamic Host Configuration Protocol*). DHCP permite configurar um conjunto de endereços os quais só serão usados pelos agentes móveis, bem como o tempo de validade do endereço (também chamado *lease time*) pode ser ajustado de maneira que o nó móvel tenha que renovar o seu endereço em um intervalo de tempo determinado, gerando o evento (*trigger*) de camada 3 que será usado para atualizar o RM.

- **Protocolo para sinalização de *handover*:** Qualquer protocolo de *fast handover* pode ser empregado no bloco FH. No caso do IPv6, a RFC 4068 (*Fast Handovers for Mobile IPv6*) propõe

[9] A mensagem de RESV é responsável pela reserva de recursos para o fluxo. Maiores detalhes sobre as mensagens e o funcionamento do RSVP em [9].

[10] Veremos mais detalhes sobre esses dois tipos de configuração de endereço no capítulo 3

um protocolo de sinalização de *handover* na camada 3. Outra alternativa é usar *triggers* de camada 2 capazes de sinalizar os eventos da camada de enlace Wi-Fi para a rede de acesso.

2.2 Modelos de segurança

Para prover segurança no acesso a uma rede, diversos modelos de segurança podem ser aplicados em diversos níveis. Cada um possui vantagens e desvantagens que são convenientes ou não para situações específicas.

Falaremos brevemente a seguir sobre este modelos, classificando-os de acordo em qual nível do modelo OSI ele atua. Daremos maior atenção à camada 2, uma vez que o modelo de segurança de camada 2 é um dos temas centrais deste trabalho.

2.2.1 Segurança na camada 2

Os modelos que são utilizados na camada 2 para a segurança de redes cabeadas não são aplicáveis para redes sem fio, devido à natureza completamente distinta da informação e como ela é transmitida. Em uma rede cabeada é muito mais difícil de interceptar uma comunicação entre dois computadores do que em uma rede sem fio, onde a informação se propaga livremente no espaço. Como consequência, a adulteração de pacotes de autenticação é prática comum nas redes sem fio, porém não nas redes cabeadas.

Outra diferença é na procura e localização da rede. Não há outra maneira de saber se existe uma rede cabeada em uma residência a não ser conectando o cabo de rede em um computador. Para isso, no entanto, foi necessário ter acesso às dependências internas da construção e ter acesso ao cabeamento da rede. Já com redes sem fio o acesso é mais simples. Os pontos de acesso 802.11 devem anunciar a sua presença. Este alcance deve ser suficientemente grande para que dispositivos Wi-Fi possam se associar a este ponto de acesso sem necessariamente estarem próximos fisicamente do mesmo. Este anúncio é feito através de quadros de *beacon*. Estes quadros não só carregam informações sobre o ponto de acesso, mas também parâmetros da rede. Os *beacons* se propagam sem cifragem para facilitar o acesso dos dispositivos a ele. Portanto, qualquer atacante precisa somente de uma antena para detectar a existência de uma rede sem fio.

Ao encontrar uma rede, o próximo passo é associar-se a esta. Em uma rede cabeada a localização e associação são praticamente o mesmo processo. Nas redes sem fio existem diversas maneiras de fazer a associação. O padrão 802.11 a princípio só provia uma maneira de restringir o acesso à uma rede sem fio, o qual era feito implementando-se a especificação WEP (*Wired Equivalent Privacy*). Em meados de 1998[11], havia uma grande preocupação de se disponibilizar o mais rápido possível o hardware necessário para implantar uma rede sem fio; muitos fornecedores de hardware deixaram de implementar o WEP e partiram para uma outra solução: filtragem de endereços MAC.

Como todo dispositivo de rede, as placas de rede sem fio possuem um identificador de 48 bits (MAC) nos quadros enviados por elas. Este identificador é chamado de endereço MAC. Partindo do princípio onde os dispositivos possuem MACs exclusivos, a filtragem de endereços MAC tem como objetivo permitir associação ao ponto de acesso somente às interfaces de rede que possuam seu

[11] O WEP se tornou padrão IEEE 802.11 em 1997

endereço MAC cadastrado no mesmo. Este procedimento não é suficiente para controlar o acesso, uma vez que existem diversas formas de mudar o endereço MAC de uma interface de rede sem fio. Não é possível assumir que, se um dispositivo tem seu MAC cadastrado no ponto de acesso, este é um dispositivo confiável.

Tendo em vista aumentar a confiabilidade das redes sem fio, os fabricantes de hardware passaram a suportar o WEP. Após alguns anos, diversas falhas de segurança foram encontradas no WEP, em especial em sua criptografia, que poderia ser facilmente quebrada. Para suprir as deficiências do WEP, foi criado o WPA (*Wi-Fi Protected Access*). O WPA nada mais é do que uma preparação para o padrão IEEE 802.11i, considerado o mais seguro até hoje. A segunda versão do WPA, WPA2, é considerada a implementação do padrão IEEE 802.11i.

WEP - *Wired Equivalent Privacy*

WEP foi desenvolvido com o intuito de garantir a privacidade da comunicação entre os clientes e os pontos de acesso. Além disso, levou-se em consideração o custo necessário para a implementação do protocolo no hardware.

A base de funcionamento do WEP é a utilização do algoritmo criptográfico RC4 com chaves de 40 bits para cifrar os pacotes trocados entre o ponto de acesso e o cliente, de maneira que a informação não pudesse ser recebida por outro cliente qualquer que não tivesse a chave criptográfica. Para evitar que o pacote fosse adulterado antes de chegar ao destino, o WEP usa uma função detectora de erros chamada CRC-32. Ao fazer o checksum do pacote, utiliza-se esta função para gerar um outro identificador chamado ICV (*Integrity Check Value*) que é conferido no destino do pacote.

Como uma primeira tentativa de fornecer segurança para as comunicações sem fio, o WEP foi bem vindo e é largamente utilizado até hoje. Todavia, suas falhas de segurança fizeram com que o WEP perdesse a credibilidade para ser usado em redes onde os requisitos de segurança fossem exigentes. Suas falhas mais críticas estão no seu algoritmo de cifragem e na função detectora de erros. O algoritmo RC4 mostrou-se fraco [12], podendo ser quebrado facilmente com o hardware disponível hoje. A sua função de detecção de erros é linear, tornando possível adulterar pacotes cifrados sem saber a chave RC4. Além disso existem outras características que comprometem a segurança: todos os clientes usam a mesma chave criptográfica, aumentando assim o volume de dados para um atacante realizar ataques sobre a mesma chave. Finalmente, diversas ferramentas conseguem quebrar a segurança de uma rede WEP em cerca de minutos, explorando as falhas de segurança que o protocolo apresenta.

Uma tentativa de aprimorar o WEP foi o WEP2. Entretanto este nada faz com relação às falhas do WEP original, apenas dificulta o processo de quebra utilizando, por exemplo, chaves criptográficas de 104 bits ao invés de 40 bits.

Diversos guias e ferramentas livres para quebrar e explorar as deficiências de uma rede protegida pelo WEP podem ser consultados na Internet [11].

WPA - *Wi-Fi Protected Access*

Criado pela Wi-Fi Alliance, o grupo que detém os direitos da marca Wi-Fi e responsável por certificar todos os dispositivos compatíveis com este padrão, WPA nada mais é do que um esforço para tentar suprimir as falhas de segurança encontradas no WEP. Ele foi baseado no modelo IEEE

802.11i, já contendo a maioria desta especificação. Sua versão mais recente, WPA2 (WPA versão 2), é 100% compatível com o 802.11i. WPA é compatível com todos os adaptadores de rede sem fio, porém requer atualização dos pontos de acesso mais antigos. Já para implementar o WPA2 pode ser necessário atualizar o firmware tanto do ponto de acesso quanto dos adaptadores Wi-Fi. Este padrão foi desenvolvido para a utilização em conjunto com um servidor de autenticação IEEE 802.1x, também chamado de servidor AAA (*Authentication, Authorization and Accounting*), porém pode ser usado no modo chave pré-compartilhada, onde todos os dispositivos Wi-Fi usam a mesma palavra-chave para o acesso.

Dentre os diversos avanços do WPA em relação ao WEP, destacamos o TKIP (*Temporal Key Integrity Protocol*). Este protocolo foi elaborado com o intuito de ser o mais compatível possível com o WEP, de maneira que não fosse necessária a troca dos dispositivos de rede Wi-Fi. Portanto ele continua a utilizar o RC4, usado no WEP e comprovadamente inseguro nos padrões de hoje. O que o torna mais seguro do que o WEP, dentre outros atributos, é a checagem de integridade da mensagem e redistribuição de chaves. No WEP, é possível alterar o conteúdo de um pacote cujo conteúdo fosse conhecido, mesmo sem decifrá-lo. Isto não acontece com o TKIP, pois sua verificação de integridade de mensagens cobre este caso. A redistribuição de chaves faz com que um atacante tenha menos dados decifrados com uma mesma chave para tentar algum tipo de ataque, pois a chave de cifragem usada por um dispositivo móvel é trocada periodicamente.

O WPA2 contém todos os atributos do WPA, além de algumas características adicionais previstas no modelo 802.11i. Em particular, foi adicionado um novo algoritmo de cifragem baseado no AES (*Advanced Encryption Standart*)[12], chamado CCMP (*Counter Mode with Cipher Block Chaining Message Authentication Code Protocol*), o qual é considerado totalmente seguro até o momento. Hoje, todos os dispositivos novos que quiserem ter a certificação Wi-Fi devem ter a certificação WPA2.

A autenticação no WPA e WPA2 pode ser feita de duas maneiras. Uma delas é o uso de chave pré-compartilhada, geralmente usado quando não há disponibilidade de um servidor AAA. O outro é uma combinação de autenticação de sistema aberto e autenticação 802.1x, ou seja, usando um servidor AAA. O dispositivo móvel é autorizado a enviar pacotes de autenticação para o ponto de acesso, o qual encaminhará a informação para o servidor AAA. Se o servidor AAA aceitar as credenciais do nó móvel, o ponto de acesso fornece ao dispositivo uma chave criptográfica para enviar e receber dados. Caso contrário, o ponto de acesso informa ao nó móvel que o acesso foi negado. Há um avanço considerável na segurança do acesso em usar um servidor AAA, pois toda a inteligência do controle de admissão é feito por ele, o qual só divulga o resultado para o ponto de acesso; há menos informações de controle trafegando pelo ar, o que dificulta uma série de ataques.

Outro atributo importante do WPA é o uso do EAP (*Extensible Authentication Protocol*) [14], um framework para a criação de mecanismos de autenticação os quais chamamos de métodos EAP. A utilização dos métodos EAP permite que cada fabricante possa implementar o seu próprio mecanismo de autenticação sem comprometer a interoperabilidade com outros hardwares, desde que estes também usem algum método EAP. Originalmente o WPA certificava somente um método EAP, chamado EAP-TLS (*EAP - Tranport Layer Security*). Para aumentar a interoperabilidade entre os hardwares Wi-Fi, outros métodos EAP passaram a ser certificados pelo WPA.

Mesmo com o WPA2, uma lição foi aprendida ao se observar o caso do WEP. Os protocolos e

[12]O AES é um padrão de criptografia adotado pelo governo dos Estados Unidos e usado largamente.

métodos de segurança na camada 2 dos dispositivos sem fio estarão sempre em constante mudança. Não é possível afirmar que o WPA2, mesmo sendo o mais seguro até agora apesar de conter algumas falhas [15], será sempre seguro. Quanto mais utilizado, mais chances de novas falhas serem descobertas e exploradas e maior a necessidade de se continuar a pesquisar maneiras mais eficientes e seguras de trafegar dados em redes sem fio.

2.2.2 Segurança na camada 3

No caso da camada 3 e superiores, os mecanismos de segurança usados independem da camada de enlace. Tanto em redes cabeadas quanto em redes sem fio, estes modelos podem ser aplicados da mesma forma. Isto porque a maioria deles têm como objetivo proteger o protocolo mais usado de camada 3, o IP (*Internet Protocol*). Diversos tipos de ataques podem ser efetuados sobre este protocolo, os quais em sua maioria não podem ser detectados ou evitados por segurança de camada 2.

O IP foi elaborado em uma época onde a Internet consistia de pequenas redes universitárias e de centros de pesquisa. Enquanto nessas redes era possível confiar no usuário, hoje centenas de milhões de pessoas usam a Internet em suas próprias residências, no trabalho e em locais públicos como *cyber* cafés ou LAN *houses*. E estas realizam as mais diversas operações: desde ler correio eletrônico até fazer compras e transações bancárias. Neste contexto, o número de criminosos virtuais começou a crescer rapidamente. Para quase toda falha encontrada por *hackers* no protocolo IP é disponibilizado na própria Internet no que consiste a falha, como explorá-la e muitas vezes como resolvê-la. Boa parte dos chamados *crackers* não sabem no que consistem as falhas; simplesmente se aproveitam das descobertas dos *hackers*[13] para a realização de ataques. Alguns *crackers* que possuem conhecimento técnico disponibilizam ferramentas gratuitas para que outros *crackers* as utilizem.

Neste ambiente hostil, soluções para a segurança da Internet foram tomadas. Um dos caminhos tomados foi melhorar e incrementar o próprio protocolo IP. O outro foi, através do IPv6, rever todas as falhas fundamentais de seu antecessor e desenvolver um protocolo que possua soluções de segurança nativos[14]. O IPsec foi uma das alternativas encontradas para aumentar a segurança e confiabilidade do IPv4 e obrigatório no IPv6.

IPsec - IP Security

O IPsec é um conjunto de protocolos que visam melhorar a segurança do IP. Eles adicionam as seguintes funcionalidades ao protocolo IP:

- Cifragem de tráfego;
- Validação de integridade;

[13] Existe uma grande confusão entre estes dois termos. *Hacker* é o nome dado para um especialista em informática e computação. Como este termo passou a ser usado para representar criminosos virtuais, os próprios *hackers* criaram o termo *cracker*, este sim, uma pessoa de má fé que quase sempre usa o conhecimento dos *hackers* para crimes virtuais. *Crackers* que possuem conhecimentos de *hackers* são extremamente perigosos, todavia são em menor número.

[14] Este foi apenas um dos motivadores para a formulação do IPv6. Um motivo que foi levado em consideração é o espaço de endereçamento do IPv4 que, devido ao número crescente de dispositivos com acesso a Internet, está se esgotando.

- Autenticação dos nós.

Dois modos de operação foram previstos no IPsec. Um deles é chamado modo de transporte: provisão de segurança fim a fim, onde o processamento relativo a segurança é feito nas duas pontas do fluxo de pacotes. Neste modo, apenas o conteúdo do pacote (*payload*) é cifrado, preservando-se os demais cabeçalhos. A vantagem deste modo de operação é que o roteamento IP padrão continua válido, uma vez que não houve modificação no cabeçalho do pacote, podendo ser encaminhado normalmente pelos roteadores. Uma desvantagem é que, como o hash do pacote não é recalculado após a cifragem, ele não consegue passar por NATs. Entretanto existem maneiras de contornar esta deficiência. Este modo é mais utilizado em conexões ponto a ponto.

O outro modo é chamado modo túnel. Ao invés de cifrar somente o conteúdo, todo o pacote é cifrado. Para ser transmitido, este pacote cifrado é colocado como *payload* de um outro pacote IP. Este modo é normalmente utilizado para estabelecer túneis entre roteadores.

Em ambos os modos, o conceito atrás do IPsec é o mesmo: utilizar associações seguras entre os nós da rede. Para prover segurança no IPv4 e também no IPv6, onde se tornou componente obrigatório, foram definidos dois cabeçalhos: AH (*IP Authentication Header*) e ESP (*IP Encapsulating Payload*). O cabeçalho AH fornece autenticação e integridade ao pacote, dada a escolha adequada do algoritmo de cifragem. Já o cabeçalho ESP provê confidencialidade.

2.2.3 Segurança na camada 4 e superiores

Na camada de transporte, os protocolos mais usados são o TCP (*Transmission Control Protocol*) e o UDP (*User Datagram Protocol*). Uma vez que estes foram desenvolvidos junto com o IP, as premissas de segurança destes não diferem das usadas no IP: não houve preocupação na cifragem, confidencialidade e consistência dos dados. Com o passar dos anos, muitas falhas de segurança relacionados a más implementações ou a falhas conceituais foram sendo descobertas e usadas por *crackers* para ações criminosas. No caso do TCP/UDP, muitas aplicações já tinham sido desenvolvidas a partir destes protocolos e uma reestruturação ou mesmo reformulação destes não era (e continua não sendo) viável. Observamos que a segurança de camada 4 e superiores independe totalmente do tipo de enlace físico, portanto todas as considerações feitas são aplicáveis tanto a redes cabeadas quanto a redes sem fio.

A abordagem portanto prevê melhorias na segurança destes dois protocolos, porém sem a necessidade de uma reestruturação dos mesmos. Uma das soluções para a segurança da camada 3, o IPSec, cobre a camada 4, uma vez que suas funções criptográficas são utilizadas nas seções TCP. O uso de *firewalls* também ajuda a evitar diversos problemas de segurança do TCP e UDP e mesmo das aplicações.

Acima da camada 4 e antes da camada 7 encontra-se outro método que se tornou muito usado para garantir segurança: o SSL (*Secure Socket Layer*) e o TLS (*Transport Layer Security*).

SSL/TLS - Secure Socket Layer / Transport Layer Security

O SSL e o seu sucessor, o TLS, são protocolos de criptografia que oferecem serviços de segurança fim a fim para as aplicações, com comunicação segura e autenticação.

O SSL versão 3.0, lançado pela Netscape em 1996, sofreu pequenas alterações e se tornou o TLS 1.0, protocolo definido pela IETF em 1999 na RFC 2246. Desconsiderando algumas diferenças que

não vamos comentar aqui, tratam-se do mesmo protocolo e as considerações a seguir valem para ambos.

Para comunicação cliente e servidor, o modo mais comum de usar o TLS é com autenticação somente do lado do servidor. O servidor possui um certificado que garante sua identidade e autenticidade, enquanto que o cliente permanece anônimo. Desta forma, o cliente terá certeza da identidade do servidor, através do certificado emitido pelo mesmo. Este modelo de segurança é largamente utilizado em sistemas de banco via internet, onde é essencial ter certeza de que as informações de sua conta estão realmente sendo enviadas para o banco.

Para comunicações onde é necessário saber a identidade do cliente, usa-se o esquema de autenticação mútua. Neste caso, o cliente deve apresentar um certificado que autentique sua identidade. Embora muito mais seguro, este método exige esforço de configuração do lado do cliente.

Este é um dos motivos da autenticação somente do servidor ser tão popular. Um complicador é a falta de flexibilidade de acesso - se o sistema de banco de internet usasse autenticação mútua deveríamos instalar o certificado respectivo que autentica cada cliente que utilizarmos. O nível de segurança obtido com o mecanismo de autenticação somente do lado do servidor é na maioria dos casos suficiente para as aplicações.

O modo de funcionamento do TLS é divido em três partes. A primeira é a negociação do algoritmo de cifragem que será usado. A segunda parte trata da troca de chaves públicas para a comunicação cifrada. A terceira parte é a transmissão dos dados cifrados, utilizando algoritmos de chave simétrica - algoritmos que utilizam a mesma chave tanto para a cifragem como a decifragem.

TLS é usado em protocolos de aplicações como HTTP (*HyperText Transfer Protocol*), SMTP (*Simple Mail Transfer Protocol*), FTP (*File Transfer Protocol*) e acima de protocolos de camada de transporte, como o TCP. Seu uso mais notável é em conjunto com o HTTP, formando o HTTPS (*HyperText Transfer Protocol Secure*). HTTPS é o padrão para transações seguras na Internet, em especial transações financeiras. Outro uso do TLS é no tunelamento do tráfego da rede formando VPNs (*Virtual Private Network*), uma outra alternativa às VPNs construídas sobre IPSec.

Camada de aplicação

Já a camada de aplicação (camadas 5, 6 e 7) é a mais problemática no aspecto da segurança. Um dos motivos é a existência de uma grande variedade de protocolos de aplicação. Em uma máquina, diversas aplicações conectadas na internet rodam ao mesmo tempo, tornando extremamente difícil uma ferramenta ou modelo que consiga garantir a segurança de todas. O outro motivo é que a maioria dos ataques de segurança visam acesso a algum tipo de informação de alguma aplicação. Para um *cracker*, é mais vantajoso conseguir executar um ataque sobre a aplicação em si do que algum ataque nas camadas inferiores, onde ele precisa de mais passos para conseguir acesso aos dados da aplicação.

A questão se resume em uma pergunta simples: a aplicação é capaz de se proteger por si só, sem ajuda de outros recursos de outras camadas de rede? A reposta é: deveria. Softwares seguros devem tratar a entrada e saída a partir do pressuposto que tudo aquilo o qual ela não controla é inseguro e deve ser tratado como tal. Nem todos os softwares são desenvolvidos com esta concepção, e as falhas de segurança na própria aplicação surgem.

Alguns protocolos tentam aumentar o nível de segurança das aplicações, oferendo-lhes serviços de autenticidade, privacidade e consistência.

- *S/MIME - Secure Multipurpose Internet Mail Extensions*

MIME é um protocolo que permite que código binário seja anexado junto ao corpo de texto de mensagens de correio eletrônico. Este código binário pode ser um arquivo de áudio, vídeo ou mesmo um arquivo binário executável. Como o MIME não prevê nenhum tipo de criptografia, o correio eletrônico e todo o seu conteúdo é passível de ataques de *sniffing*, onde o tráfego da rede é monitorado de maneira passiva e transparente e o conteúdo de cada pacote pode ser capturado para análise posterior. Além disso, é possível adulterar o conteúdo do email, corropendo seu conteúdo. O S/MIME é um protocolo com a mesma funcionalidade do MIME, todavia com suporte a cifragem de dados. Isso garante que a mensagem chegue ao seu destino com a garantia de não ter sido alterado no meio do caminho.

- *HTTPS / S-HTTP - Secure HyperText Transfer Protocol*

HTTP é o protocolo adotado na *World Wide Web* para a transferência de dados entre servidores e clientes. O uso do HTTP cresceu com a Internet, passando a ser cada vez mais exposto e, com isso, suas falhas de segurança foram sendo expostas e exploradas. Por ser responsável por grande parte do tráfego da rede, qualquer vulnerabilidade ou deficiência neste protocolo compromete a confiabilidade da rede. O S-HTTP (também é conhecido por HTTPS) é um conjunto de políticas que permite que o tráfego HTTP seja encapsulado e cifrado de diversas maneiras, entretanto não especificando nenhum tipo de algoritmo de cifragem ou estrutura de chave criptográfica, deixando a cargo de cada implentação do S-HTTP decidir qual mecanismo é mais adequado para cada funcionalidade. A implementação mais comum do S-HTTP é em conjunto com o TLS, conforme citado anteriormente.

2.3 Trabalhos relacionados

As duas extensões do MIPv6 discutidas em anteriormente, HMIPv6 (seção 2.1.2) e FMIPv6 (seção 2.1.3), são exemplos de trabalhos relacionados à micro-mobilidade de redes IPv6.

Em [6] uma proposta de micro-mobilidade é apresentada tendo como base o funcionamento de redes de celulares. Esta proposta se chama Cellular IP. Uma rede de acesso que utiliza Cellular IP se conecta à Internet através de um roteador chamado *gateway*. Este gateway provê endereços IP locais aos nó móveis - endereço chamado *Care-of Address* como vimos no cenário do MIPv6. Outros roteadores, chamados estações base, são responsáveis por todas as funções de mobilidade, funcionando como um roteador *wireless* e como roteador IP. Dois tipos de *handovers* são suportados: *hard handover*, onde o nó móvel decide mudar de estação base se desassociando completamente da estação base antiga e *semisoft handover*, onde um nó móvel, no processo de mudança de estação base, recebe pacotes das duas estações base. Cellular IP requer *software* adicional no nó móvel para funcionar.

Outra proposta de micro-mobilidade é o HAWAII (*Handover-Aware Wireless Access Internet Infrastructure*) [7]. O objetivo do HAWAII é estender o MIPv6, diminuindo a perda de tráfego durante o *handover* e a alta quantidade de mensagens de controle entre nó móvel e *Home Agent*. A proposta do HAWAII é utilizar o mecanismo do MIPv6 para macro-mobilidade e os seus mecanismos para prover micro-mobilidade com maior eficiência do que o MIPv6. Todavia, pelo fato do HAWAII contar com mecanismos do IPv6, existem as mesmas restrições de uso: protocolo de tranporte tem que ser, obrigatoriamente, IPv6, e *software* adicional é necessário no nó móvel.

Capítulo 3

Soluções de acesso

O acesso a uma rede Wi-Fi pode ser feito de diferentes formas. Dependendo do tipo de uso e do nível de segurança desejado, temos diferentes níveis de dificuldade na configuração da rede. Redes para uso doméstico geralmente não necessitam de segurança rigorosa, enquanto redes corporativas necessitam de um esquema mais rígido. Um dos motivos é a quantidade de tráfego: ataques de segurança podem ser feitos mais facilmente quanto maior for o tráfego pela rede. Em uma empresa a quantidade de dispositivos móveis acessando os diversos pontos de acesso gera um tráfego muito atrativo para esse tipo de ataque.

Outro aspecto no acesso à rede é a configuração do nó móvel. O mais comum, herança do modelo de acesso das redes cabeadas, é a configuração de endereço com estado (*stateful*) usando o protocolo DHCP. Com o IPv6, o modo de configuração de endereço sem estado (*stateless*) ganha destaque.

Na seção 3.1 apresentamos algumas das ameaças mais comuns à segurança de redes Wi-Fi. Na seção 3.2 apresentamos os modos de acesso a uma rede sem fio, considerando tanto praticidade de configuração quanto segurança provida. Na seção 3.3 discutiremos quais modelos de configuração são mais adequados para usos específicos.

3.1 Ataques de segurança mais comuns

Partimos sempre do pressuposto de que uma rede Wi-Fi é perceptível por qualquer dispositivo móvel. Todos os ataques descritos a seguir só são possíveis pois informações sobre a rede trafegam livremente pelo enlace aéreo e podem ser capturadas, mesmo que criptografadas, por qualquer hardware que tenha uma interface Wi-Fi monitorando o meio. Eles variam de acordo com o tipo de informação e como que a mesma trafega pelo ar. Uma rede Wi-Fi que não é perceptível e nem tem tráfego exposto pelo ar se assemelha a uma rede cabeada convencional, não sendo objeto deste trabalho.

3.1.1 Ataques passivos

Usando algum dispositivo com interface sem fio, este fica monitorando o tráfego e buscando informações que permitem quebrar a segurança da rede. A partir deste tipo de ataque pode-se mapear um certo padrão no tráfego que passa por aquele ponto de acesso, o que, mesmo que não seja sufici-

ente para entrar na rede, já se caracteriza como um monitoramento de tráfego por um agente externo. Damos o nome de ataque passivo pois o invasor não tenta quebrar a segurança da rede, fica apenas monitorando o tráfego.

3.1.2 Ataques de dicionário

Trata-se de um ataque passivo, porém aplicado em redes corporativas ou redes de maior volume de tráfego. Capturando-se o tráfego destas redes e correlacionando-os por um longo período de tempo, pode-se quebrar chaves e senhas usadas para a segurança do acesso.

3.1.3 Ataque homem do meio

Com o objetivo de capturar informações sobre o acesso à rede, o atacante personifica o ponto de acesso que deseja atacar. Alguns clientes móveis se conectam aos pontos de acesso usando o identificador ESSID do mesmo. Este identificador pode ser divulgado abertamente, junto com a identificação do ponto de acesso, se este estiver configurado para tal. Configurando-se um outro ponto de acesso com este mesmo identificador, este receberá requisições de autenticação de clientes que pensam que este é o ponto de acesso verdadeiro. O falso ponto de acesso receberá então informações que são enviadas pelos nós móveis ao ponto de acesso verdadeiro. O atacante então pode usar um cliente móvel convencional e usar a informação capturada para acessar o ponto de acesso verdadeiro.

Uma forma de evitar este tipo de ataque é configurar os pontos de acesso de maneira que os seus respectivos ESSIDs não sejam divulgados livremente pelo ar, impossibilitando a clonagem descrita acima.

3.1.4 Ataques de jamming

São ataques do tipo negação de serviço. A freqüência padrão de operação da tecnologia Wi-Fi 802.11 b/g é de 2,4GHz. Todavia, esta faixa de freqüência é dividida por outros equipamentos eletrônicos, por exemplo telefones sem fio. Um ruído de alta intensidade nesta freqüência próximo a uma rede sem fio causa distúrbios na comunicação de dados, podendo torná-la inoperante.

Uma solução para este tipo de ataque é utilizar uma outra faixa de freqüência para a tecnologia Wi-Fi. Obviamente isto implica em grandes esforços para atualizar ou trocar todo o hardware que utiliza a freqüência de 2,4GHz.

3.2 Modos de acesso

Um dispositivo Wi-Fi no sistema operacional Linux possui os seguintes modos de operação: Ad-Hoc, *Managed*, *Master*, *Repeater*, *Secondary* e Monitor. O modo *Master* permite que o dispositivo móvel atue como um ponto de acesso, ou seja, como um gateway de conexão entre a rede a qual este tem acesso e outros nós Wi-Fi. No modo *Repeater* o nó móvel atua como um roteador, encaminhando pacotes que chegam a ele. Já no modo *Secondary* o dispositivo atua como um ponto de acesso ou roteador de backup, no caso do principal não puder atuar. Por fim, o dispositivo móvel fica em estado

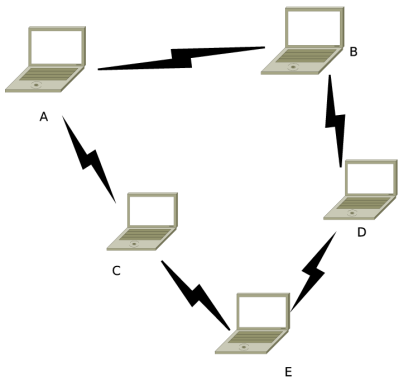

Figura 3.1: Exemplo de uma rede Ad-Hoc de 5 nós.

passivo, vigiando o tráfego ao seu alcance através do modo Monitor. Note que em todos estes modos que detalhamos o nó móvel não acessa uma rede Wi-Fi.

Os outros dois modos já permitem o acesso a uma rede Wi-Fi. No modo Ad-Hoc o dispositivo móvel não procura por pontos de acesso, atuando ponto-a-ponto com outros nós móveis. Já no modo *Managed* o nó se associa a um ponto de acesso. A associação atráves de um ponto de acesso é o modo mais comum, porém o Ad-Hoc tem suas aplicações, por exemplo, em redes *peer-to-peer*.

3.2.1 Wi-Fi modo Ad-Hoc

Ad-Hoc em latim significa "para este propósito", alguma alternativa ou atitude tomada para uma solução imediata. Redes Wi-Fi do tipo Ad-Hoc tem como objetivo estabelecer uma conectividade entre os dispositivos presentes na mesma área, a qual é cessada no término da comunicação.

A comunicação é feita ponto-a-ponto entre todos os nós que pertencem à rede. Para isso, estes devem divulgar sua conectividade para os nós próximos, de maneira que seja possível identificar uma rota de um nó para qualquer outro, uma vez que estes não possuem nenhum conhecimento prévio da topologia da rede. Na rede Ad-Hoc da figura 3.1, por exemplo, o nó A só se comunica com o nó E atráves do encaminhamento da informação pelos nós B e D ou pelo nó C.

Para descoberta de topologia diversos protocolos de roteamento podem ser utilizados os quais são classificados como pró-ativos ou reativos. Os pró-ativos mantêm periodicamente em todos os nós tabelas de roteamento com as rotas para todos os nós atualizadas. Os reativos procuram uma rota sob demanda, somente quando necessitam se comunicar com um nó específico. Um exemplo de protocolo de roteamento pró-ativo é o AWDS (*Ad-Hoc Wireless Distribution Service*) e um de reativo é o DSR (*Dynamic Source Routing*). Ambos possuem vantagens e desvantagens as quais não serão discutidas nesta dissertação.

Redes do tipo Ad-Hoc são especialmente utilizadas em situações onde se necessita de uma rede de comunicação volátil de rápida implantação e mínima configuração, onde a preocupação com segurança, sigilo e autenticidade é deixada de lado graças ao caráter temporário da comunicação. O uso dessas redes nas forças armadas é bastante comum, onde deseja-se transferir dados entre dois ou mais dispositivos móveis temporariamente. Alguns videogames portáteis como o Playstation Portable e o Nintendo DS utilizam redes Ad-Hoc para partidas com mais de um jogador *multiplayer*.

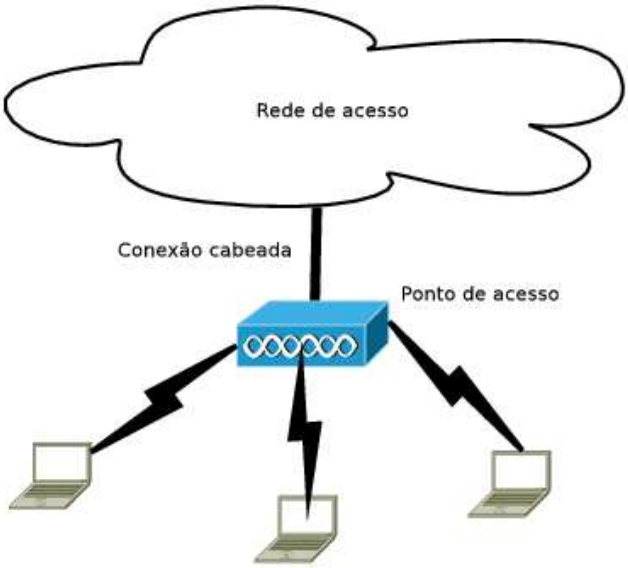

Figura 3.2: Exemplo de uma rede com ponto de acesso.

3.2.2 Wi-Fi modo *Managed*

O baixo custo de pontos de acessos Wi-Fi contribuiu para a grande disseminação e utilização das redes sem fio. Diversos estabelecimentos comerciais disponibilizam pontos de acesso para seus clientes se conectarem a internet. A comodidade de não usar cabos para montar LANs e a facilidade de configuração de uma placa sem fio trouxe as redes Wi-Fi para as residências. Qualquer notebook moderno possui uma placa de rede Ethernet e uma Wi-Fi. O acesso via ponto de acesso é o mais comumente usado e no qual daremos mais atenção.

O ponto de acesso consiste em um meio de comunicação entre os dispositivos Wi-Fi e a rede cabeada a qual este está ligado. Os pontos de acesso funcionam como *hubs* em redes Ethernet, entretanto ao invés de cabos são utilizadas conexões de rádio. A figura 3.2 mostra a arquitetura geral de uma rede com um ponto de acesso, onde este provê conectividade dos nós móveis à rede de acesso.

O acesso a uma rede sem fio é caracterizado pelo maneira a qual o nó móvel se conecta a mesma. Este acesso é dependente exclusivamente dos requisitos de segurança exigidos pela rede e pela configuração do nó que antecede a associação com o ponto de acesso.

Tipos de configuração de nós móveis

Caracterizamos três tipos diferentes de configuração para um nó móvel em uma rede IP qualquer. Um deles é a configuração manual, outro é a configuração via o protocolo de configuração, tipicamente DHCP, e a configuração de endereço *stateless*, nativa do IPv6.

A configuração manual consiste na configuração de cada nó de forma independente por um administrador ou usuário. Para cada nó inserido ou retirado da rede é necessário um esforço de configuração para configurar o nó e, em certos casos, reconfigurar as rotas da rede e sua topologia. Para redes de pequeno porte, onde a inserção ou remoção de nós é rara, a configuração manual de endereço é

uma alternativa viável.

A configuração de endereço a partir do protocolo DHCP fornece uma maneira de concentrar a configuração dos nós em uma entidade central, o servidor de configuração. Qualquer nó que deseja acessar a rede deve solicitar a este servidor os parâmetros de configuração necessários. O esforço do administrador de rede está em configurar o servidor de configuração de maneira apropriada. O DHCP é largamente utilizado pois é incluído em praticamente qualquer sistema operacional moderno. Qualquer distribuição Linux possui cliente e servidor DHCP na instalação, enquanto que os sistemas operacionais da Microsoft, mesmo de maneira trasparente para o usuário, utilizam o DHCP para a configuração automática de endereço. A deficiência mais grave do DHCP é a centralização de toda a configuração no servidor, a qual pode ser contornada utilizando-se servidores redundantes caso o principal falhe.

A configuração *stateless* do IPv6 é uma alternativa de se eliminar o controle centralizado da configuração dos nós, permitindo com que cada nó seja responsável pela sua própria configuração. Uma rede típica IPv6 divulga com certa freqüência mensagens de divulgação de rota, chamadas de RA (*Router Advertisement*), para todos os nós conectados no *link*. Ao conectar um nó neste link, o nó receberá esta mensagem e configurará seu endereço IPv6 e outros parâmetros, como *gateway* padrão e DNS (*Domain Name Server*) da rede. O endereço é configurado fazendo-se uma junção do prefixo da sub-rede divulgado na mensagem de RA com o seu endereço MAC. O endereço resultante passa por uma verificação de redundância chamada DAD (*Duplicate Address Detection*) antes de ser usado. A vantagem deste modelo de configuração é a descentralização da configuração dos nós. A desvantagem é que não existe um controle dos endereços que podem ser utilizados.

Estes 3 modelos de configuração de endereço podem ser aplicados a qualquer tipo de rede, tanto cabeada quanto Wi-Fi. Todavia, para redes Wi-Fi existe um padrão *de facto* para o uso da configuração via DHCP. Um dos motivos é a facilidade de configuração. Redes móveis são por natureza dinâmicas, podendo ter diferentes nós se associando ao ponto de acesso a cada instante; a configuração manual está descartada por ser muito custosa nesse cenário. Outro motivo é o fato que a configuração do tipo *stateless* é exclusiva do IPv6, não sendo utilizada no IPv4. Como a maioria das redes ainda utilizam IPv4, podemos desconsiderar este modo de configuração também.

Portanto, assumiremos que toda a configuração de endereço para os casos que estudaremos adiante é feita através do protocolo DHCP.

Níveis de segurança

Conforme mencionado no começo deste capítulo, quanto maior o nível de segurança exigido maior o esforço de configuração e manutenção da rede Wi-Fi e em certos casos da rede de acesso cabeada. Um maior nível de segurança necessita também configuração extra dos nós móveis para o acesso.

O primeiro nível de segurança é o nível onde não há segurança alguma. O ponto de acesso fica exposto ao mundo externo, sem exigir nenhum tipo de credencial para liberar o acesso a seu *link*. Para isso, deve-se considerar que todos os nós móveis que se conectarem neste ponto de acesso não são confiáveis. Isto implica que estes não devem ser tratados como os outros nós da rede de acesso. Um exemplo de como isto pode ser possível é a utilização de um *firewall* filtrando todo o tráfego do ponto de acesso, controlando a maneira com a qual os nós móveis associados ao mesmo se comunicam com a parte interna da rede. A utilização do DHCP neste cenário torna-se atrativa, pois se pode estabelecer

configurações especiais para os nós que se associarem aos pontos de acesso inseguros, facilitando a configuração do filtro de tráfego.

O segundo nível é a utilização de um esquema de segurança no ponto de acesso, por exemplo, WEP. Para acessar a rede, é necessária a configuração da chave WEP no nó móvel. Atualmente o WEP é largamente utilizado devido a facilidade de configuração do ponto de acesso e do nó móvel. Entretanto, diversos estudos já comprovaram que o WEP e WEPv2 podem ser facilmente quebrados. Como citado no capítulo 2, existem ferramentas livres na Internet que descobrem a chave do ponto de acesso WEP vigiando o tráfego no link sem fio. É um erro comum hoje acreditar que o WEP seja suficiente para que todos os nós móveis sejam tratados como confiáveis e tenham acesso à rede cabeada. A solução para este problema é adotar um mecanismo de segurança mais forte ou modelar a segurança da rede cabeada considerando cada nó móvel como não confiável.

O terceiro nível é a adoção de um esquema de segurança forte no ponto de acesso. Para redes sem fio isso implica no uso de WPA e WPA2 com um servidor de autenticação. Neste cenário o acesso ao meio é custoso, podendo até implicar em troca de certificados entre o ponto de acesso e o nó móvel para comprovar autenticidade. Todavia, as chances de um ataque de segurança a partir da camada de enlace são reduzidas, de maneira que podemos aliviar a carga de configuração da rede cabeada para se proteger dos dispositivos móveis não-seguros. Este nível necessita de configuração do nó móvel para o acesso a rede, não sendo recomendado para pontos de acesso onde se deseja fornecer serviços esporadicamente para acessos rápidos.

A tabela 3.1 compara os 3 níveis de segurança de camada de enlace descritos levando em consideração a configuração tanto da rede de acesso quanto do nó móvel.

Nível	Dificuldade configuração rede acesso	Dificuldade configuração nó móvel
1	baixa	baixa
2	baixa	media
3	alta	alta

Tabela 3.1: Comparação entre os diferentes tipos de configuração de segurança.

3.3 Configuração de acesso em redes Wi-Fi

Tomando como base as características descritas acima, é possível estabelecer qual tipo de configuração representa bom custo-benefício para um fim específico.

3.3.1 Redes Wi-Fi residenciais

Com o custo cada vez mais baixo de provedores com alta velocidade[1], muitas residências passaram a ter uma conexão veloz à internet. Ao mesmo tempo, o preço de um computador de mesa ou *desktop* cai drasticamente a cada dia. Boa parte desta queda se dá devido a queda de preços dos *notebooks*. É possível encontrarmos pessoas que utilizem apenas seu computador móvel tanto no trabalho

[1] Velocidades acima de 256kbps, também conhecido como banda-larga.

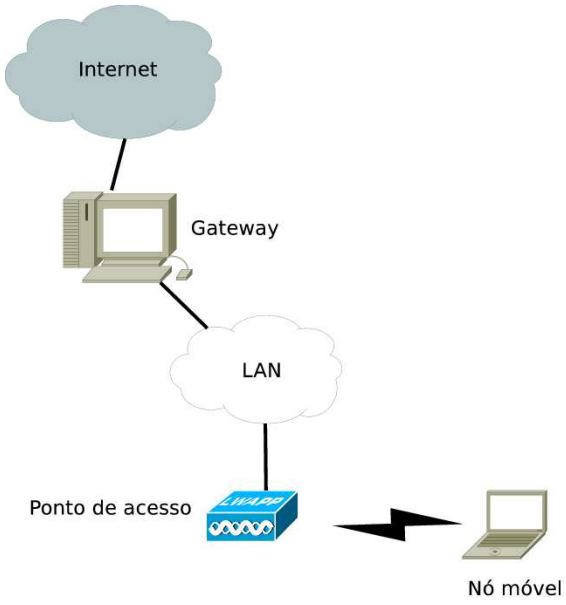

Figura 3.3: Topologia comum encontrada em residências com internet banda-larga e dispositivos móveis.

quanto em casa, fazendo o *desktop* cair em um desuso. Para continuar sobrevivendo, os computadores de mesa têm que oferecer desempenho alto a preços cada vez mais baixos.

Com este cenário, é comum encontrarmos redes domiciliares de mais de um computador de mesa e com um ponto de acesso Wi-Fi para os dispositivos móveis. Uma topologia para este tipo de rede é apresentada na figura 3.3. A conexão a Internet é feita via modem, o qual é conectado a um *desktop*. Este computador que recebe a conexão é responsável por compartilhar esta com os outros elementos da rede, fazendo o papel de *gateway* da rede. Boa parte dos sistemas operacionais possuem facilidades para configurar a máquina para ser o *gateway* da rede. Este computador é conectado a uma rede local cabeada na qual podemos encontrar mais computadores de mesa. Conectado a esta LAN (*Local Area Network*) temos um ponto de acesso que permite que nós móveis acessem a LAN e, consequentemente, a conexão à internet.

Outra topologia que está ficando cada vez mais frequente com o barateamento de roteadores Wi-Fi é ilustrada na figura 3.4. O roteador faz o papel de ponto de acesso e de *gateway*, centralizando toda a configuração de compartilhamento de conexão com a Internet e de acesso Wi-Fi em um só elemento. Embora seja de configuração mais simples do que a exposta na figura 3.3 ambas possuem as mesmas características: redes com poucos *desktops* que prevêm o acesso de alguns nós móveis a mesma.

Os dois modelos de segurança mais usados são os níveis 1 e 2, ambos por serem de configuração menos trabalhosa da rede de acesso. Particularmente o nível 2 é o mais usado neste caso, mesmo exigindo uma configuração de chave criptográfica no nó móvel. Como não se espera que diversos nós móveis de diferentes origens acessem uma rede Wi-Fi residencial, o esforço adicional de configuração compensa a segurança adicional.

Além disso, o ponto de acesso não fica completamente aberto para a conexão de qualquer nó móvel ao alcance. É cada vez mais comum ao ligar um *notebook* em nossas residências encontrarmos uma rede Wi-Fi alcançável e sem chave, provavelmente do vizinho ao lado. Em edifícios é ainda mais

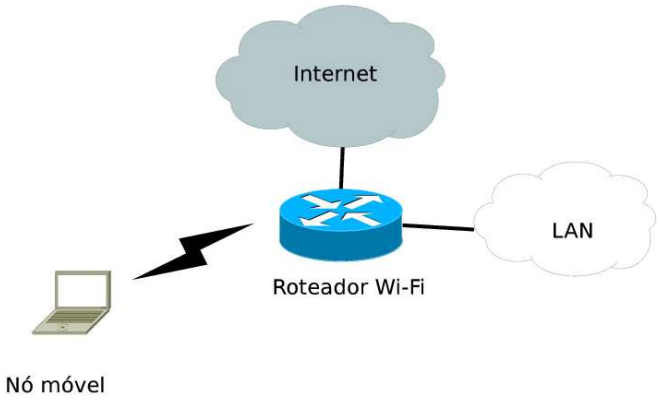

Figura 3.4: Topologia encontrada em residências usando roteador Wi-Fi.

comum, pois as ondas de rádio do ponto de acesso podem atravessar paredes e teto, sendo perceptíveis nos apartamentos vizinhos. Obviamente se o vizinho acessar o seu ponto de acesso, sua conexão de internet será divida com ele também, causando perda de performance.

Conforme discutido na seção anterior, o nível 2 possui segurança inferior ao nível 3, onde já se utiliza WPA e WPA2. Porque não se usa o nível 3 para redes domésticas? Se o número de nós móveis é restrito, um esforço maior de configuração pode prover uma segurança no nível de enlace muito mais alta, pois não se usaria mais WEP.

A configuração é muito mais complexa do que a segurança nível 2. Primeiro temos o servidor de autenticação, o qual deve ser configurado um serviço ativo em uma das máquinas da rede local para que o ponto de acesso consiga autenticar cada tentativa de acesso a rede Wi-Fi. Em segundo lugar temos a configuração do nó móvel, o qual não é configurado somente com a chave do ponto de acesso, mas com um usuário previamente configurado no servidor AAA. Dependendo do método de autenticação utilizado é necessário gerar certificados de autenticidade para uso do nó móvel e do servidor usando SSL. Por fim, a quantidade de tráfego de uma rede domiciliar Wi-Fi não é tão grande, o que dificulta boa parte dos ataques de segurança de camada 2.

3.3.2 Redes Wi-Fi comerciais

A flexibilidade de ser capaz de conectar à internet de qualquer lugar é um dos atrativos de se adquirir um *notebook*. E isso vem ocorrendo com mais freqüência pois muitas lojas, cafeterias, cantinas e afins passaram a prover acesso Wi-Fi gratuito a seus clientes para acesso à internet. Muitas pessoas acabam entrando nesses estabelecimentos somente por este acesso e podem vir a consumir algum produto ou serviço.

A topologia dessas redes comerciais não difere em modo geral das que presenciamos nas redes domésticas. Para fornecer o serviço mais simples possível aos seus clientes, a escolha mais comum é o nível 1 de segurança Wi-Fi, ou seja, sem segurança de camada de enlace e com configuração nula do ponto de acesso. Uma vez que não existe a preocupação de proibir acessos simultâneos aos recursos da rede, a adoção de um esquema de segurança de nível 2 com WEP seria apenas uma configuração adicional a ser feita no nó móvel.

Não é recomendado acessar serviços de banco online ou qualquer outro tipo de serviço que en-

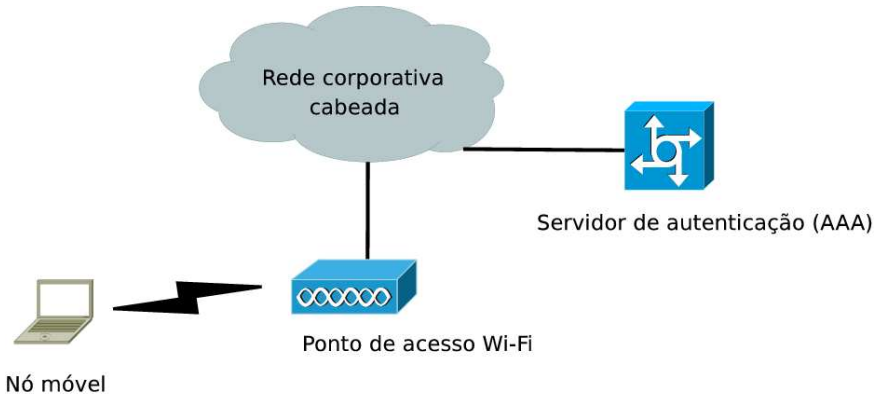

Figura 3.5: Topologia geral encontrada em redes Wi-Fi corporativas.

volva transações confidenciais a partir deste tipo de rede, uma vez que o sigilo do acesso não é garantido e é facilmente monitorado. Todavia, admite-se que o acesso a esse tipo de rede seja breve - de 10 a 30 minutos - e que o maior uso seja para navegar portais de notícias e consultar correio eletrônico.

3.3.3 Redes Wi-Fi corporativas

Os requisitos de redes corporativas e empresariais possuem requisitos rigorosos de segurança. A adoção do WPA e WPA2 permitiram que redes Wi-Fi pudessem ser utilizadas nesses ambientes. A configuração da rede de acesso é feita pelo administrador de rede, o qual geralmente é responsável pela configuração de cada nó móvel que acessará esta rede Wi-Fi.

A figura 3.5 exemplifica a topologia usada por esse tipo de rede Wi-Fi. A grande diferença entre esse tipo de topologia e as das figuras 3.3 e 3.4 é a presença do servidor AAA, responsável pela autenticação do acesso. A partir deste é possível configurar diferentes credenciais para diferentes dispositivos móveis, além de ter controle de quem terá acesso a rede Wi-Fi.

Devemos ressaltar que nada impede que em outras redes Wi-Fi dentro da mesma empresa não se possa utilizar outros modelos de segurança. Em áreas de acesso de visitantes é interessante adotar um esquema de segurança menos rígido que exija menos configuração do nó móvel e seja de uso mais prático.

Capítulo 4

Acesso na arquitetura MPA

Neste capítulo discutiremos como os modelos de acesso apresentados no capítulo 3 podem ser aplicados na arquitetura MPA. Na seção 4.1 discutiremos uma implementação da arquitetura MPA em termos de protocolos usados e ambiente físico da rede de acesso. Na seção 4.2 analisaremos a performance de *handover* dos nós móveis sem utilizar a arquitetura, considerando somente a camada de enlace. Por fim, na seção 4.3, apresentaremos os experimentos envolvendo *handover* na arquitetura MPA.

4.1 Implementação da arquitetura MPA

No capítulo 2 apresentamos a descrição funcional da MPA com as características dos protocolos candidatos a compor cada bloco funcional. Em nossa implementação instanciamos os blocos funcionais utilizando os protocolos a seguir.

4.1.1 RSVP-TE

RSVP-TE proporciona uma solução simples e eficaz tanto para o gerenciamento de túneis quanto para o roteamento móvel. Essa facilidade deve-se ao uso de um objeto opaco chamado objeto de localização de nós móveis que os roteadores que atuam como MAR são capazes de interpretar.

O gerenciamento de túneis da rede *overlay* móvel é feita através do RSVP-TE. Cria-se uma topologia ponto-multiponto e a reserva destes túneis é feita com mensagens de PATH gerados pelo MAR de ingresso e encaminhadas em direção ao MAR de egresso (sentido *downstream*). O estado dos túneis é mantido através de mensagens de RESV gerados pelo MAR de egresso e encaminhadas em direção ao MAR de ingresso (sentido *upstream*). Os túneis são mantidos em *soft-state*, sendo inativados depois de um intervalo de tempo onde não houve mensagens de RESV que atualizassem o estado dos mesmos.

O objeto de localização de nós móveis representado na figura 4.1 carrega no campo *Mobile Node ID* o identificador único do nó móvel, na maioria dos casos o endereço MAC da sua placa da rede, o campo *Mobile Node Address Type* indica o tipo de endereçamento de camada de rede (IPv4 ou IPv6 por exemplo), *Mobile Node Prefix Length* representa o tamanho do prefixo, *Mobile Node Prefix* o prefixo do nó móvel e *Egress MAR ID* o identificador do MAR de egresso.

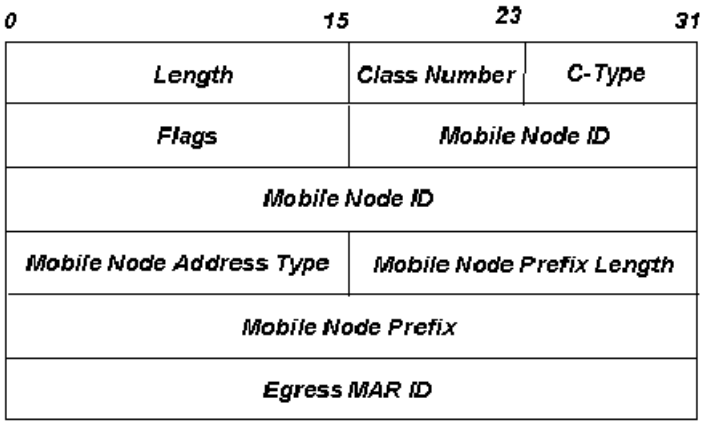

Figura 4.1: Ilustração do objeto de localização de nós móveis

O uso do prefixo do nó móvel, ao invés do endereço completo, permite que um conjunto de nós móveis sejam referenciados em um mesmo objeto no caso de mobilidade agregada, onde um bloco de nós executam *handover* simultaneamente. Este objeto é incorporado a uma mensagem de RESV em sentido *upstream*, seguindo a operação normal do protocolo RSVP-TE. Esta mensagem é gerada no MAR de egresso assim que este reconhece que um nó móvel se associou a um ponto de acesso conectado neste MAR. Ao receber esta mensagem, o MAR atualiza sua tabela de roteamento móvel com a posição atual do nó móvel. Neste contexto, as mensagens de RESV que carregam o objeto opaco descrito acima são usadas como mensagens de roteamento, pois notificam a rede de acesso de uma mudança de MAR de egresso de um ou mais nó móveis, a qual deve realizar as atividades de reroteamento de túneis e tráfego de maneira adequada.

Esta abordagem permite gerar eventos de camada 2 e 3 sem a necessidade de implementar um protocolo específico para a sinalização de mobilidade.

4.1.2 DHCPv4 / DHCPv6

Dentre os tipos de configuração de endereço *stateless* e *stateful*, optamos por usar o último. O fator determinante é a existência de uma obrigatoriedade do nó móvel em tentar uma comunicação com a rede de acesso após realizar cada *handover*. No caso da configuração *stateless* do IPv6, por exemplo, o nó móvel configura seu endereço IPv6 sem a necessidade de enviar mensagens para a rede de acesso, o que tornaria necessário *software* adicional no nó móvel para a notificação de associação a um MAR, evento necessário para a geração do *trigger*.

Na figura 4.2 temos uma ilustração do funcionamento padrão do DHCP para a geração destes eventos. Esta operação padrão prevê que um nó móvel deve enviar uma mensagem em *broadcast* buscando um servidor de endereços, a qual é denominada DHCP DISCOVER. Se houver um servidor disponível, este responde com uma mensagem DHCP OFFER, já oferecendo um endereço para o nó móvel. Este ao receber o DHCP OFFER envia um DHCP REQUEST, um pedido de endereço para o servidor DHCP. A mensagem DHCP ACK confirma a reserva deste endereço para este nó móvel, o qual a partir deste momento configura sua interface de rede com os parâmetros enviados na mensagem DHCP ACK, como roteador padrão e endereço do servidor.

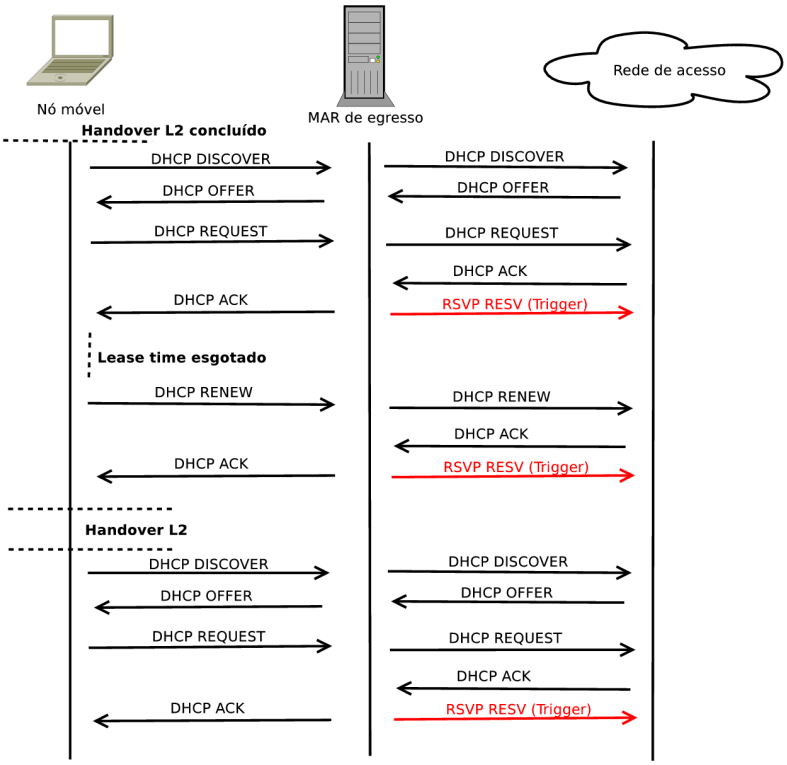

Figura 4.2: Operação do DHCP gerando *triggers* com o RSVP-TE.

A mensagem de DHCP ACK obrigatoriamente passará pelo MAR de egresso, que é o roteador de acesso deste nó móvel. Este MAR está atento a estas mensagens de DHCP ACK e, ao ver que uma delas foi enviada, um objeto RSVP-TE opaco é criado com as informações vindas desta mensagem e uma mensagem de RESV é enviada *upstream*, mensagem esta que atualizará o estado das tabelas de roteamento móvel dos MARs envolvidos. As mensagens de interesse podem ser capturadas através de um filtro de tráfego que encaminha os pacotes solicitados para o espaço de usuário, onde uma aplicação pode manipulá-los da maneira que desejar. Em nosso caso utilizamos o utilitário *iptables* para capturar as mensagens DHCP que passam pelo MAR de egresso e uma aplicação em linguagem C dispara ou não o *trigger* de acordo com a mensagem recebida.

Periodicamente este nó móvel deve se comunicar com a rede de acesso para renovar o seu endereço, o qual é válido por um intervalo de tempo determinado pelo servidor DHCP (*lease time*). Após este intervalo, uma mensagem DHCP RENEW é gerada do cliente para o servidor DHCP, o qual responde novamente com um DHCP ACK e, ao receber esta mensagem, o nó móvel renova o seu endereço para uso. Este ato de renovar o endereço causa uma nova mensagem RESV, a qual atua como um *trigger* no MAR de egresso, uma vez que uma nova mensagem DHCP ACK foi gerada. Este *trigger* atualiza o estado dos túneis e do roteamento móvel. Observe que com este mecanismo é possível gerar *triggers* periódicos através da operação normal do protocolo DHCP.

Devemos observar que o comportamento acima descrito supõe que todos os elementos da rede de acesso fazem parte da mesma rede lógica ou VLAN, de maneira que a mensagem de DHCP DISCOVER enviada pelo nó móvel pode ser recebida pelo servidor DHCP. Em muitos casos isto não é possível pois o roteador de acesso está em uma VLAN diferente da VLAN deste servidor. Para estes

casos, o protocolo DHCP prevê o uso de um software adicional em cada roteador de acesso chamado agente de relay ou *relay agent*. Ele encaminha a mensagem de *broadcast* do nó móvel diretamente para o endereço do servidor DHCP, o qual responde para o *relay agent* e este encaminha a resposta para o nó móvel. A partir do momento que o nó móvel conhece o endereço do servidor DHCP e já possui um endereço de rede válido a comunicação entre ambos passa a ser direta e o *relay agent* não é mais utilizado.

4.1.3 Tunelamento IP/IP

A rede de transporte pode suportar, dentre outros protocolos, IPv4 ou IPv6. O tunelamento de tráfego nestes dois casos é feito através do GRE - *Generic Routing Encapsulation*. O GRE permite encapsulamento de IPv4 sobre IPv4 ou IPv6 e IPv6 sobre IPv6 ou IPv4.

Os túneis são estabelecidos através das mensagens PATH e RESV do RSVP-TE. Cada túnel é representado por uma interface virtual, a qual pode ser tratada como uma interface de rede comum. Assim, as entradas da tabela de roteamento móvel mapeiam prefixos dos nós móveis a interfaces virtuais, no caso os túneis IP/IP. Esta abordagem de tunelamento IP/IP torna o roteamento móvel idêntico ao roteamento convencional, uma vez que ambas compartilham do mesmo mecanismo de encaminhamento presente no sistema operacional.

4.1.4 Segurança

Diversos mecanismos de segurança foram empregados nesta implementação. Optamos inicialmente pelo segundo modelo citado no capítulo 3 - acesso usando WEP - e depois migramos para o tipo 3 - acesso via servidor de autenticação.

Em um primeiro momento a preocupação com a autenticidade e privacidade foram deixadas de lado em favor de outros tipos de testes da arquitetura, como performance de *handover*. O uso de uma autenticação do tipo WEP, neste cenário, elimina o *overhead* adicional que uma autenticação WPA com servidor AAA adiciona. Por sua vez, o uso do servidor de autenticação (em nosso caso um servidor RADIUS) permitiu novas possibilidades de *trigger*, desta vez com eventos de camada de enlace, que acontecem antes ainda dos *triggers* disparados pelas mensagens do DHCP. Devemos observar, no entanto, que um tipo de notificação não elimina a outra.

4.1.5 Ambiente físico

Uma rede com 16 nós foi utilizada para avaliar a arquitetura MPA. Cada um dos 16 nós é um computador com processador Pentium 4 de 3 Ghz, 160 Gigabytes de disco rígido e 1 Gigabyte de memória RAM. A disposição é em 2 grupos de 8, no qual cada grupo é acessado via um monitor, mouse e teclado a partir de dois dispositivos KVM (*Keyboard, Video and Mouse*) que alternam o sinal de teclado, vídeo e mouse entre as oito máquinas de cada grupo.

O sistema operacional que utilizamos em cada uma delas é o Linux Slackware 10.2, com kernel 2.6.13.3, configurado com funções de roteamento e de captura de pacotes para IPv4 e IPv6. Tivemos o cuidado de configurar cada máquina de maneira igual, de forma a eliminar possíveis fontes de problema devido a configurações conflitantes. Desta forma temos um ambiente homogêneo para simular nossa rede de acesso, onde cada máquina representa um roteador da mesma.

Figura 4.3: Rack de 16 nós que foi usado como *testbed*.

Cada nó possui quatro interfaces Ethernet, sendo que uma delas está ligada a um roteador Extreme Summit 200 para acesso externo e as outras 3 estão ligadas em um *switch* D-Link DES 3559 de 48 portas. O uso do *switch* limita-se somente a criação de enlaces ponto-a-ponto por meio de VLANs.

4.2 Análise de performance de *handover* na camada de enlace

Antes de avaliarmos a performance da arquitetura MPA é necessário estimar o tempo de *handover* de camada de enlace. Como não temos controle direto sobre este tempo, devemos parametrizá-lo adequadamente para analisarmos os resultados adquiridos na arquitetura de maneira adequada.

Para fazer o papel de nó móvel testamos diversos tipos de *laptops* com diferentes configurações e placas de rede Wi-Fi. O objetivo era estimar a influência do *hardware* nos resultados dos testes. O que concluímos foi exatamente os mesmo resultados encontrados no estudo feito em [19]. Não existe uma padronização no funcionamento destes dispositivos, embora todos declarem ser Wi-Fi. Estas diferenças de funcionalidade acarretam em tempos diferentes de *handover* usando placas de rede Wi-Fi de diferentes fabricantes.

O esquema de segurança utilizado foi o mais simples possível, no caso o WEP. A utilização de um servidor RADIUS adiciona uma latência extra no sistema a qual será discutida no próximo capítulo.

A medição do tempo de *handover* de camada de enlace foi feita a partir de um cenário simples mostrado na figura 4.4. Usamos uma máquina para gerar tráfego UDP para o nó móvel em um endereço e porta específico. A configuração do endereço de camada de rede do nó móvel é feita manualmente e de maneira a não ser necessária uma renovação de endereço a cada mudança de ponto de acesso, eliminando assim qualquer latência de camada de rede que pudesse interferir na medição. De maneira similar, avaliamos também a perda de pacotes para um fluxo em sentido *upstream*, ou seja, do nó móvel para a mesma máquina na rede de acesso. Os pontos de acesso utilizados são dois Linksys WAP54G idênticos.

Os quatro equipamentos utilizados como nós móveis foram:

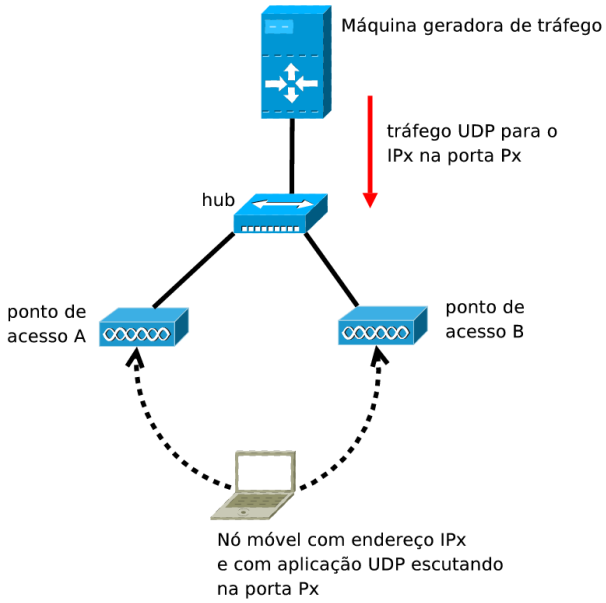

Figura 4.4: Cenário de teste para medição de *handover* de camada de enlace.

- *Notebook* Dell 510 com placa Wi-Fi Intel 2200
- *Notebook* Dell 520 com placa Wi-Fi Intel 3945
- *Notebook* Acer Aspire 3003LCi com placa Wi-Fi Broadcom 4318
- *Desktop* Dell OptiPlex com placa Wi-Fi Linksys

A perda de pacotes durante o *handover* é igual ao intervalo de tempo o qual o nó móvel recebeu ou enviou o último pacote na antena antiga e recebeu ou enviou o primeiro pacote em sua antena atual. Os pacotes são gerados em intervalos de 20ms, portanto temos este erro em cada medida feita. Realizando 10 medidas para cada equipamento diminuímos o erro em um fator de 10, ficando com 2ms de imprecisão.

Para realização deste experimento cada equipamento ficou entre os dois pontos de acesso, de maneira que não fosse necessário deslocar o equipamento para associar-se ao outro ponto de acesso. Todos os equipamentos possuem sistema operacional Linux, portanto foi possível usar o mesmo arquivo de *batch* ou script para mudança de ponto de acesso. Usando o comando *iwconfig* é possível manipular o driver de rede Wi-Fi para fazer associação, desassociação, fazer varredura procurando pontos de acesso entre outras funções.

Um cuidado especial para este teste é evitar fazer mudanças de ponto de acesso em um curto intervalo de tempo após uma mudança anterior. É previsto no IEEE 802.11 que as antenas devem guardar em cache o estado do nó móvel que acabou de se desassociar. Calculamos empiricamente que o modelo de ponto de acesso que utilizamos guarda em cache este estado por um minuto. Qualquer *handover* feito nesse intervalo de tempo leva cerca de 50ms somente. As medidas foram obtidas fazendo um *handover*, aguardando três minutos e só depois fazendo outro *handover*.

A tabela 4.1 apresenta a média de pacotes perdidos com o respectivo tempo de *handover* associado. Mesmo sendo as quatro placas de rede Wi-Fi testadas compatíveis com IEEE 802.11g, os tempos são drasticamente diferentes. Tanto em [19] como em nosso caso o modelo de *hardware* é fator decisivo para a performance de *handover*. Outro fator interessante é a diferença da perda de pacotes para *download* e *upload* que aconteceu nos quatro casos. No caso do *notebook* Acer, por exemplo, o recebimento de pacotes é restabelecido 380 milisegundos após a mudança de antena, porém o envio de pacotes para a rede de acesso requer 900 milisegundos.

Este experimento nos permite concluir que devemos evitar mudanças de equipamento para cada teste de acesso e segurança na MPA, parametrizando nossas medidas adequadamente para cada *hardware* testado.

Nó móvel	Perda de pacotes no *download*	Perda de pacotes no *upload*
Dell 510 com Intel 2200	14 (280 ms)	42 (840 ms)
Dell 520 com Intel 3495	75 (1500 ms)	83 (1660 ms)
Acer Aspire com Broadcom 4318	19 (380 ms)	45 (900 ms)
Dell OptiPlex com Linksys	20 (400 ms)	28 (560 ms)

Tabela 4.1: Perda de pacotes em download e upload para diferentes equipamentos

4.3 Implementação da Arquitetura MPA

A implementação da MPA pode ser dividida em duas partes. Na primeira parte implementamos uma rede de acesso em IPv6 com o objetivo de avaliar como a nova versão do IP atuaria diante da nossa arquitetura. Na segunda parte a MPA foi implantada em uma rede IPv4 convencional.

4.3.1 Implementação em rede IPv6

O kernel 2.6.13.3 usado nos nós de nosso rack possui suporte nativo ao IPv6, sendo necessário somente a ativação dos módulos correspondentes na configuração do kernel. A figura 4.5 representa a topologia usada para esta implementação.

Utilizamos metade do nosso rack para simular uma rede de transporte com oito roteadores com suporte a MPA (MARs). Cada *link* entre os roteadores representa uma VLAN, de maneira a isolar o tráfego de broadcast de outros *links*. Devemos pensar nos *links* como subredes entre dois roteadores, e cada subrede deve ter suas próprias políticas de tráfego e segurança e, portanto, não pode sofrer influência de nenhuma outra subrede do mesmo domínio. Cada VLAN foi nomeada VLAN1, VLAN2 até VLAN8 de acordo com o número acima de cada *link* na figura 4.5. Foram utilizados quatro pontos de acesso Linksys WAP54G dispostos um em cada MAR de egresso, no caso as máquinas DCA05, DCA06, DCA07 e DCA08.

Mesmo com o suporte nativo ao IPv6, alguns *softwares* adicionais foram necessários para realizar nossos testes. Trata-se do agente RSVP, DHCPv6 (servidor, cliente e agente de *relay*) e do agente de *router advertisement*.

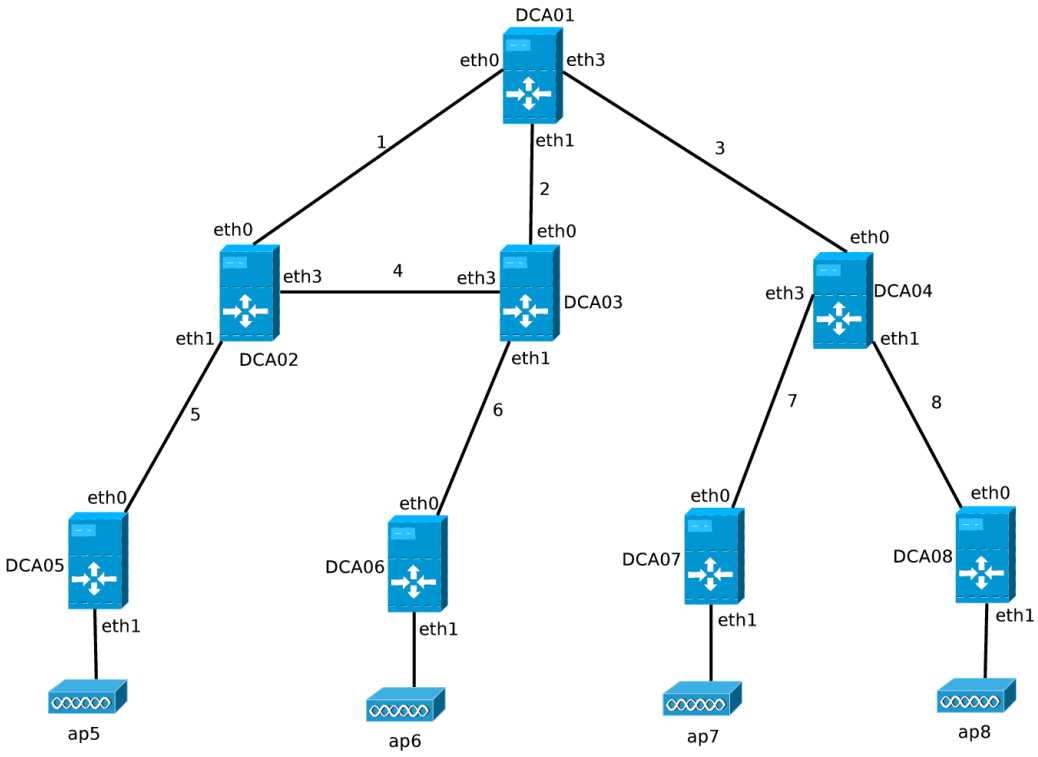

Figura 4.5: Topologia física usada para a implementação da MPA no IPv6.

- **Agente RSVP (rsvpd)**: Para nossos experimentos desenvolvemos um agente RSVP que segue as normas da RFC 3209 - *RSVP-TE: Extensions to RSVP for LSP Tunnels*. Este agente funciona como um *daemon*[1] que espera conexões TCP em uma porta específica para realizar tarefas de gerência. Cada tarefa é executada a partir de um comando em XML que descreve qual ação deve ser tomada. Estas ações estão relacionadas ao estabelecimento e rompimento de túneis P2MP da rede overlay móvel.

- **DHCPv6**: O Slackware 10.2 não possui uma suíte DHCPv6 nativa, sendo necessário a instalação de uma. Utilizamos a suíte Dibbler [16] que foi a que mais se adequou às nossas necessidades e segue a RFC 3315 - *Dynamic Host Configuration Protocol for IPv6*. O ponto crítico da utilização desta implementação em relação a outras testadas é que esta contém, além da implementação do cliente, um *relay agent* para encaminhamento de requisições DHCP e um servidor que interpreta corretamente as mensagens deste agente (mensagens *relay-forward* e *relay-reply*). Na época em que fizemos estes testes (metade de 2006) outras implementações do DHCPv6 não continham um *relay agent* ou o servidor não tinha suporte as mensagens do mesmo.

- *Router Advertisement Daemon* - **radvd**: Segundo a própria RFC 3315, o funcionamento do DHCPv6 não é exatamente igual ao funcionamento do DHCPv4. No IPv4 não existe nenhum

[1] Chamamos de *daemon* todo o processo que roda no plano secundário do sistema operacional e que reage a eventos específicos.

mecanismo de autoconfiguração, portanto o protocolo de configuração de endereço é responsável por configurar cada parâmetro, dentre eles o roteador (*gateway*) padrão. No caso do IPv6 este parâmetro não é configurado pelo DHCP pois o IPv6 prevê que esta informação é divulgada nas mensagens de *Router Advertisement*, conforme vimos no capítulo dois.

Para que os nós consigam configurar o *gateway* padrão instalamos o radvd [17]. Este *daemon* é configurado para divulgar o *gateway* padrão no *link* em que se encontra. Podemos usá-lo também para que os nós se autoconfigurem se o radvd divulgar o prefixo do roteador de acesso, o qual é utilizado pelo nó em conjunto com o seu endereço de *link* para formar o seu endereço IPv6 válido. Como não utilizamos autoconfiguração dos nós, o radvd é empregado apenas para divulgar o roteador padrão, complementando a operação do DHCPv6.

A instalação de cada um destes *softwares*, com exceção do rsvpd que deve estar presente em todos os roteadores da rede de acesso, varia de acordo com o papel que cada roteador possui na topologia atual. Em nossa topologia da figura 4.5 a máquina DCA03 hospeda o servidor DHCPv6; DCA05, DCA06, DCA07 e DCA08, nossos roteadores de acesso ou MARs de egresso, executam o *relay agent* para encaminhar as requisições de endereço para o servidor DHCPv6 e também o radvd para divulgar o roteador padrão para os nós associados nos pontos de acesso.

Execução dos experimentos

Os testes foram feitos utilizando-se a topologia de túneis e os endereços IPv6 de acordo com a figura 4.6. Os endereços IPv6 de cada interface foram estabelecidos utilizando o prefixo **fec0:a0a** mais o número da VLAN correspondente, que varia de 1 a 9, mais o identificador do roteador. No caso do roteador DCA01, sua interface que está conectada com DCA03 possui endereço fec0:a0a:2::1/24, onde o 2 é o número da vlan e o 1 é o identificador do roteador, no caso DCA0**1**. A rede de acesso Wi-Fi possui mesmo prefixo em toda a sua extensão, no caso prefixo 9. A rede overlay móvel consiste em um túnel P2MP mostrado na figura em linha tracejada. O tráfego em direção aos nós móveis será encaminhado a partir destes túneis. Note que os *branching points* de tráfego ocorrem nos roteadores DCA01, DCA03 e DCA04.

Um ponto de destaque neste cenário é a configuração do servidor DHCP para reconhecer de qual roteador de acesso a mensagem de *relay* saiu. Na configuração padrão de um servidor DHCP é natural configurar uma única faixa de endereços para cada interface utilizada e, em caso geral, uma faixa na qual o endereço da própria interface está contida. Tomemos uma máquina qualquer que possui três interfaces de rede da seguinte forma:

- interface eth0 com endereço IPv6 fec0:a0a::5/64
- interface eth1 com endereço IPv6 2000::5/64
- interface eth2 com endereço IPv6 2500::6/64

Uma requisição DHCP que chegue ao servidor pela interface eth0 tende a pertencer a rede fec0:a0a::/64, enquanto que pela interface eth1 pertence a rede 2000::/64 e pela eth2 pertence a 2500::/64. Não é um caso comum onde uma requisição de uma rede com prefixo 2000::/64 possa chegar por outra interface que não seja a eth1. Observando novamente a figura 4.6 verificamos que as requisições DHCP de

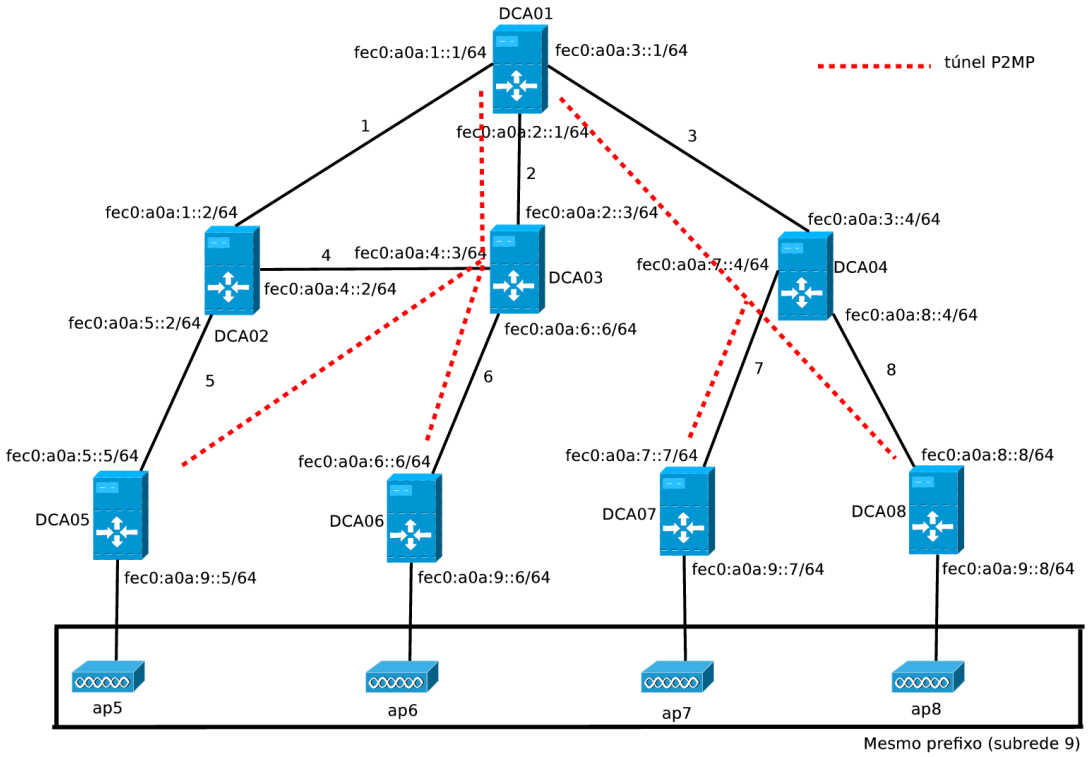

Figura 4.6: Topologia da implementação utilizando IPv6.

todos os nós móveis devem sempre devolver endereços pertencentes à rede fec0:a0a:9::/64, enquanto que nenhuma interface do nosso servidor na máquina DCA03 possui um endereço IPv6 nesta faixa.

Um outro agravante diz respeito aos MARs de egresso DCA07 e DCA08. Ambos têm como próximo *hop* em direção ao servidor DHCP a máquina DCA04 a qual encaminhará as requisições para DCA01 e finalmente DCA03 pela interface eth2 no endereço fec0:a0a:2::3/64. Isto torna impossível a distinção de qual MAR de egresso a mensagem partiu, uma vez que a faixa de endereços é a mesma (fec0:a0a:9::/64) e ambas alcançam o servidor na mesma interface.

Este problema é solucionado pelo DHCP através de um atributo na mensagem de *relay* denominada *circuit-id*. Este campo adicional é configurável nos *relay agents* de cada MAR de egresso e pode ser utilizado como parâmetro na atribuição de endereços pelo servidor DHCP, possibilitando a associação de diferentes faixas de endereços de acordo com o valor do *circuit-id* e, finalmente, saber o ponto de acesso no qual o nó móvel se associou. Embora não seja crítico para redes IPv6[2] em um cenário onde o prefixo das redes de acesso móvel é diferente esta distinção pelo *circuit-id* é crucial para o funcionamento adequado do servidor DHCP. Portanto, fizemos a configuração do servidor DHCP desta forma para tornar futuras mudanças na topologia o mais simples possível.

O teste consistiu em enviar um tráfego com origem do roteador DCA01 para o nó móvel. Uma vez que toda a rede Wi-Fi possui o mesmo prefixo, o nó móvel mantém o mesmo endereço IPv6 por toda a extensão da rede, independente de onde ele está em um dado instante. Basta o nó móvel se associar a um ponto de acesso desta rede que o endereço IPv6 adquirido será válido para qualquer

[2]Como todos os endereços atribuídos pertencem a mesma rede fec0:a0a:9::/64, seria possível atribuir este único espaço de endereços para todas as interfaces de rede do servidor.

outro ponto de acesso desta mesma rede.

Para este cenário foi utilizado o *notebook* Acer Aspire 3003LCi, o qual possui 512Mb de memória RAM, instalação do Slackware Linux 10.2 e placa Wi-Fi Broadcom 4318. Em um primeiro momento o único *software* adicional no nó móvel foi o cliente DHCPv6, o qual não é nativo do Linux instalado no nó móvel. Esta abordagem não produziu resultados promissores. Obtivemos resultados com tempos de *handover* na casa de dois segundos, o que considerando o tempo de camada de enlace que o equipamento leva (ver tabela 4.1) implica em um tempo cerca de dois segundos somente para atualizar o estado do endereço IPv6 que já era válido.

A explicação encontra-se no próprio funcionamento do radvd. Pela RFC 2461 (*Neighbor Discovery for IP Version 6 (IPv6)*) as mensagens de *Router Advertisement*(RA) não solicitadas devem ter um espaçamento mínimo de três segundos entre uma e outra, alegando que não se deve confiar na ausência desta mensagem para detectar falhas de roteador pois outras mensagens e algoritmos realizariam esta tarefa. No pior caso, um nó móvel realizará *handover* e esperará três segundos para receber a mensagem de RA com o *gateway* padrão, o que impossibilita a renovação de seu endereço prévio com o servidor DHCP.

Para tentar contornar este problema fizemos algumas modificações tanto no radvd quanto no nó móvel. Primeiramente, tomamos como base a RFC 3775 (*Mobility Support in IPv6*) que prevê novos intervalos mínimos para o envio da mensagem de RA. O tempo mínimo passou a ser 0,03 segundos e o máximo 0,07 segundos, dando em média 50 ms para o envio de uma mensagem de RA não solicitada. Como já havíamos adicionado o cliente Dibbler no nó móvel, colocamos um pequeno script de mudança de ponto de acesso com um pequeno trecho de código que envia uma mensagem de *Router Solicitation* para o novo *link* assim que o *handover* de camada de enlace é realizado. O radvd, ao receber uma mensagem de *Router Solicitation*, deve responder imediatamente com uma mensagem de RA.

Junto com esse script colocamos também um programa que tenta simular o comportamento que se observa com o Wi-Fi no Windows XP. Neste sistema operacional a placa de rede Wi-Fi se associa ao ponto de acesso com sinal mais forte e muda para outro ponto de acesso conforme a potência dos sinais começa a oscilar. Esse programa usa o *iwtools* do Linux para tentar simular este comportamento. Ele realiza a varredura dos pontos de acesso alcançáveis, faz o parser da resposta para obter a potência atual dos pontos de acesso e, baseando-se em uma certa heurística (potência do ponto de acesso atual pelo menos 10% menor do que a do ponto de acesso mais potente, por exemplo), o programa dispara o script de *handover*.

Este nosso *software* que muda automaticamente de ponto de acesso no Linux trouxe implicações interessantes nos resultados obtidos, as quais veremos a seguir.

Resultados obtidos

Foram efetuados diversos testes de *handovers* com o nó móvel agindo como receptor de um tráfego oriundo do roteador DCA01 como mostra a figura 4.6. O gerador de tráfego utilizado foi o MGEN [20].

Primeiramente testamos a performance do nosso programa para mudança automática de antena. Realizando o rastreamento de pontos de acesso a cada dois segundos, a heurística para mudança de antena é quando um dos pontos de acesso mostrar-se por quatro rastreamentos consecutivos mais forte do que os demais e o nó móvel não estar associado a este ponto de acesso. Percebemos que

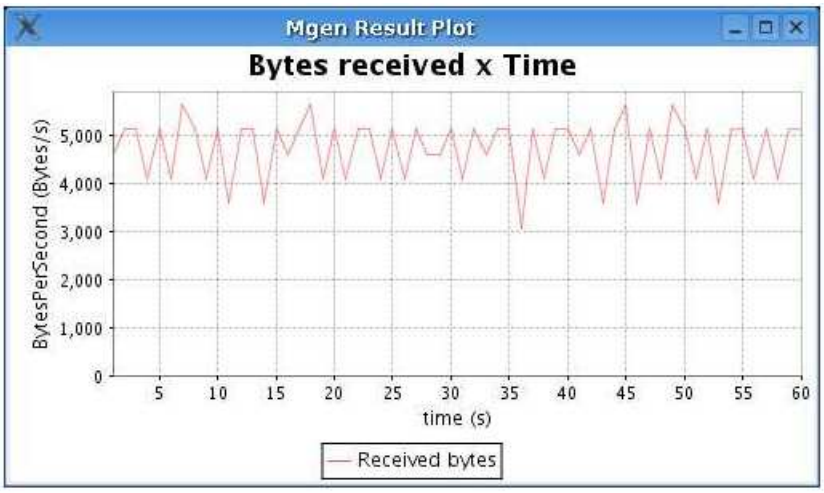

Figura 4.7: Teste de recepção de um fluxo de dados com o programa de mudança de antena ativado.

em alguns destes testes o tráfego recebido pelo nó móvel continha muitas perdas de pacote, mesmo quando o *handover* não acontecia naquele instante. Devemos esperar uma perda de informação em uma comunicação Wi-Fi, a qual é afetada pelos mais diversos ruídos do meio, ao contrário de uma comunicação por cabo cujo enlace é isolado do meio externo. A questão passou a ser se a perda que estávamos presenciando era natural ou se nosso programa estava influenciando de alguma forma.

A resposta pode ser vista observando-se as figuras 4.7 e 4.8. Enviando um fluxo UDP contínuo para o nó móvel estacionário a a partir do roteador DCA01, portanto sem realizar nenhum *handover*, capturamos quantos pacotes foram perdidos em dois cenários. Em um deles o nosso programa de mudança de antena estava ativado, enquanto que no outro este programa estava desativado. Os resultados mostram que nosso programa estava causando uma perturbação no recebimento de pacotes do nó móvel e, coincidentemente, em tempos condizentes com o intervalo de rastreamento de antenas configurado. Concluímos que o comando que realiza varredura a procura de ponto de acesso faz com que a placa de rede Wi-Fi não seja capaz de receber todos os pacotes enviados para a mesma, uma vez que o dispositivo está realizando o rastreamento de pontos de acesso.

Como o objetivo de nossos testes é verificar o efeito do *handover* em nossa arquitetura e não de otimizar esta mudança automática de antena, os demais testes foram feitos sem este *software* para mudança automática pois não encontramos uma outra maneira de realizar esta tarefa sem utilizar o comando que efetua varredura.

Realizamos seis experimentos. Em cada um deles um *handover* é feito do ponto de acesso do MAR de egresso DCA05 para o MAR de egresso DCA06. O tráfego, conforme dito anteriormente, é gerado da máquina DCA01 para o nó móvel, o qual, após receber um endereço IPv6 válido em sua primeira associação com DCA05, mantém este mesmo endereço após o *handover* para DCA06. Os experimentos diferem entre si pela característica do tráfego enviado e pelo protocolo de camada de transporte utilizado. A duração do fluxo é fixa em 60 segundos e o tamanho do pacote é fixo em 512 bytes. Variamos a taxa de pacotes por segundo, testando fluxos de 10, 100 e 1000 pacotes / segundo, o que nos dá uma banda de 5, 50 e 500 kbytes/s.

As medidas tomadas nesses experimentos foram:

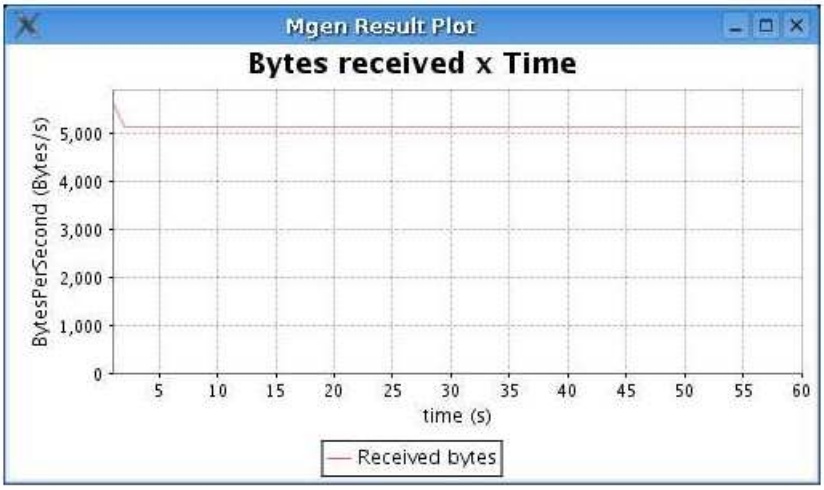

Figura 4.8: Mesmo teste da figura 4.7, porém sem o programa de mudança ativado.

- **TL3** - Tempo total de *handover* de camada 3: intervalo de tempo que começa desde o instante onde o nó inicia o processo de reassociação a um outro ponto de acesso e termina quando o nó móvel configura o *gateway* padrão já no outro ponto de acesso, após processar a mensagem de *Router Advertisement*.

- **Tmpa**: O tempo que a arquitetura necessita para sinalizar a associação do nó móvel neste novo ponto de acesso e redirecionar o tráfego para esta nova localização. Para calcular este tempo medimos o intervalo de tempo entre o recebimento do último pacote no ponto de acesso antigo e o recebimento do primeiro pacote após o *handover*, descontando TL3.

- Perda de pacotes no *handover*: quantos pacotes foram perdidos no intervalo de tempo que o nó trocou de ponto de acesso.

- Perda total de pacotes: quantos pacotes foram perdidos no período de 60 segundos. Esta perda inclui a perda pelo *handover* e a perda pelo enlace aéreo.

A seguir apresentamos os resultados obtidos. Os experimentos 1, 2 e 3 foram feitos com tráfego UDP, enquanto que os outros três foram feitos com tráfego TCP. No caso dos experimentos com TCP a quantidade de perda de pacotes não foi medida, simplesmente pois o TCP possui mecanismos para evitar a mesma, retransmitindo o pacote quantas vezes forem necessárias. Todavia a perda pode ser visualizada pelas figuras 4.12, 4.13 e 4.14.

- **Experimento 1**: Tráfego UDP de 10 pacotes por segundo, totalizando 5Kbytes por segundo de banda. Esta taxa de pacotes é baixa o suficiente para não haver perda de pacotes pelo ar e somente uma pequena perda de pacotes no *handover*. As medidas tiradas foram:

  - TL3 = 1054 ms
  - Tmpa = 190 ms
  - Tempo total de *handover* = 1244 ms

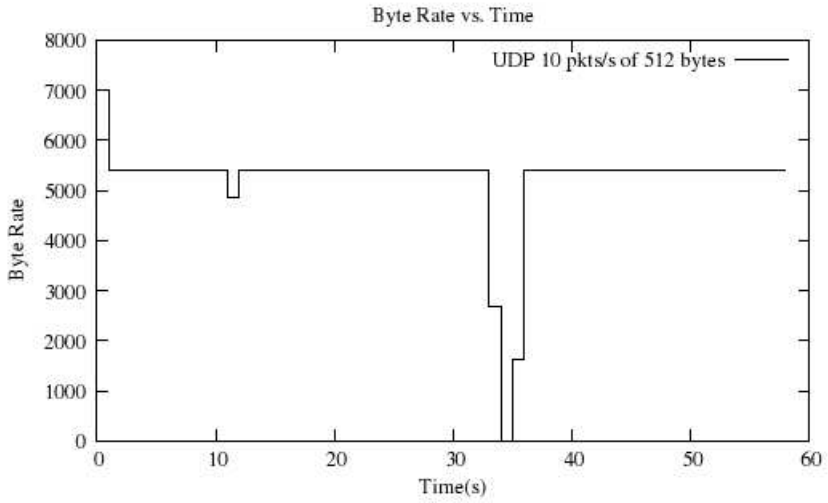

Figura 4.9: Tráfego UDP recebido pelo nó móvel no experimento 1.

  - Perda de pacotes devido ao *handover* = 12 (2% do total)
  - Perda total de pacotes = 12

- **Experimento 2**: Tráfego UDP de 100 pacotes/segundo, totalizando 50 Kbytes/s de banda. Os resultados deste experimento estão abaixo. Note que estes foram similares ao experimento 1, com exceção da quantidade de pacotes perdida no enlace aéreo que já começa a aparecer (uma vez que a perda de pacotes total é maior do que a perda devido ao *handover*).

  - TL3 = 1080 ms
  - Tmpa = 191 ms
  - Tempo total de *handover* = 1271 ms
  - Perda de pacotes devido ao *handover* = 125 (2,1% do total)
  - Perda total de pacotes = 146

- **Experimento 3**: Tráfego UDP de 1000 pacotes/segundo, totalizando 500 Kbytes/s de banda. Esta banda mostra-se alta o suficiente para provocar uma perda de pacotes no enlace aéreo compatível com a perda de pacotes causada pela troca de antena. O que é interessante observar é que, mesmo com uma banda relativamente alta, o tempo Tmpa permanece praticamente constante, assim como a porcentagem de perda de pacotes devido ao *handover*.

  - TL3 = 1126 ms
  - Tmpa = 191 ms
  - Tempo total de *handover* = 1219 ms
  - Perda de pacotes devido ao *handover* = 1502 (2,5% do total)

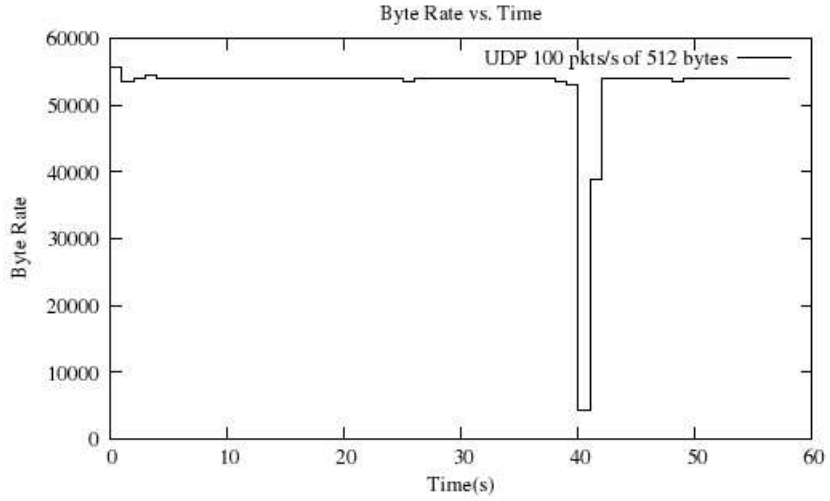

Figura 4.10: Tráfego UDP recebido pelo nó móvel no experimento 2.

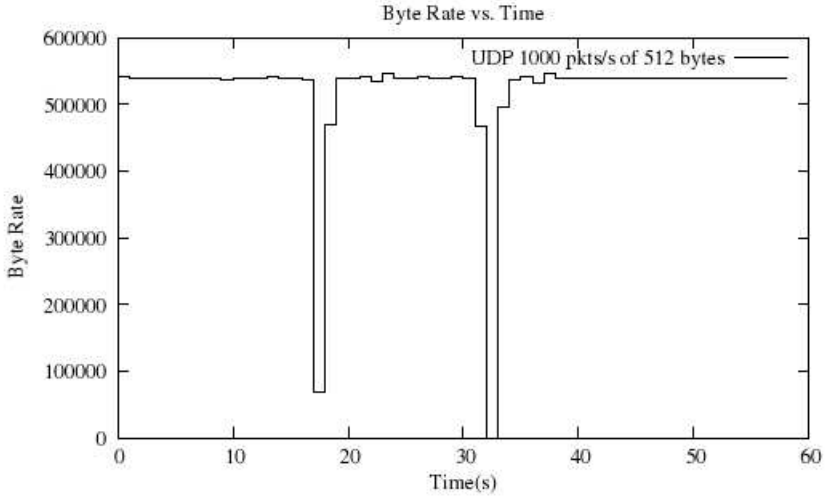

Figura 4.11: Tráfego UDP recebido pelo nó móvel no experimento 3.

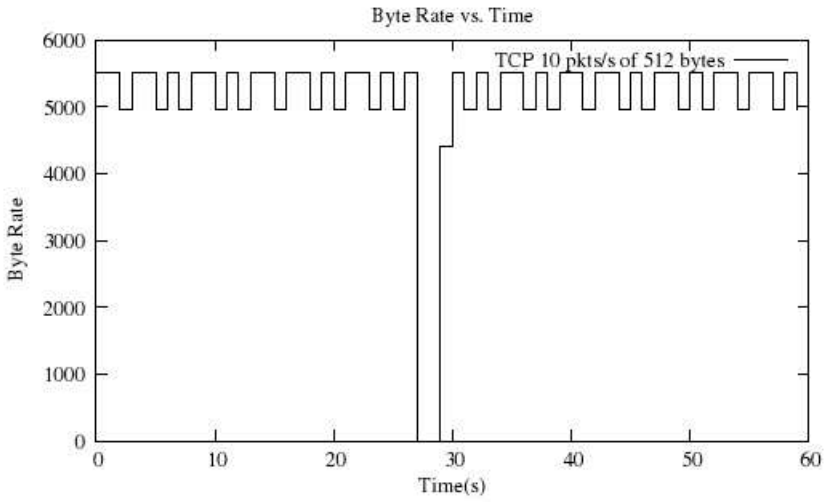

Figura 4.12: Tráfego TCP recebido pelo nó móvel no experimento 4.

  - Perda total de pacotes = 2445

- **Experimento 4**: Tráfego TCP de 10 pacotes/segundo, totalizando 5 Kbytes/s de banda. Para este teste com TCP o tempo Tmpa já aumenta significativamente em comparação com os valores obtidos nos testes 1, 2 e 3. A banda recebida se manteve constante, porém com oscilações.

  - TL3 = 954 ms
  - Tmpa = 623 ms
  - Tempo total de *handover* = 1577 ms

- **Experimento 5**: Tráfego TCP de 100 pacotes/segundo, totalizando 50 Kbytes/s de banda. A banda real recebida pelo nó móvel, no entanto, não passa de 35 Kbytes/s, devido a perda de pacotes no enlace aéreo em conjunto com o algoritmo de retransmissão do TCP. O tempo Tmpa, por sua vez, manteve-se na mesma ordem de grandeza que foi medida no experimento 4.

  - TL3 = 1116 ms
  - Tmpa = 497 ms
  - Tempo total de *handover* = 1613 ms

- **Experimento 6**: Tráfego TCP de 1000 pacotes/segundo, totalizando 500 Kbytes/s de banda. Entretanto o nó móvel recebe não mais de 70 Kbytes/s, mostrando a quantidade altíssima de pacotes que deixaram de ser transmitidos graças a combinação retransmissão do TCP e aumento das perdas de pacotes no enlace aéreo. Curiosamente o tempo Tmpa obtido foi mais baixo do que nos outros dois testes com TCP.

  - TL3 = 1152 ms

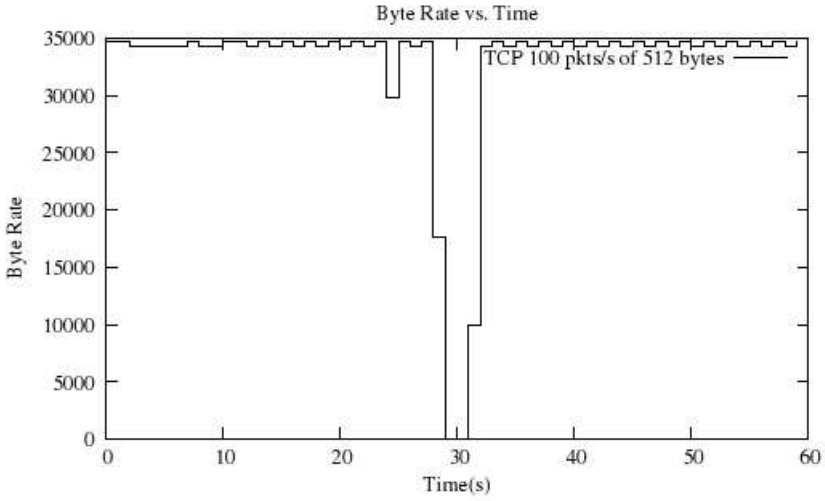

Figura 4.13: Tráfego TCP recebido pelo nó móvel no experimento 5.

- Tmpa = 298 ms
- Tempo total de *handover* = 1450 ms

Comentários sobre os resultados obtidos

O tempo que chamamos de TL3 possui valor médio de 1100 milisegundos, sendo de 700 a 900 milisegundos o tempo do *handover* de camada de enlace, e o restante do tempo consiste em 20 milisegundos para configurar a interface Wi-Fi e por volta de 200 milisegundos para o processamento das mensagens de descoberta de vizinhança do IPv6, ou seja, as mensagens de *Router Solicitation* e *Router Advertisement*. Os tempos medidos em nosso experimentos foram 1054, 1080, 1126, 954, 1116 e 1152 milisegundos, medidas factíveis com as medidas preliminares feitas na tabela 4.1, uma vez que o envio da mensagem de *Router Solicitation* não deixa de ser um *upload* e o tempo de *upload* medido para o *notebook* usado foi de 900 milisegundos.

Para os testes com UDP, a perda de pacotes devido a uma troca de antena foi de 2 a 2,5% com um tempo de sinalização da MPA constante de 200 milisegundos, o que mostra que o *overhead* da nossa arquitetura e modelo de acesso independe do volume de tráfego que é transmitido.

Já os testes com TCP mostraram que o algoritmo de *slow start* para a retransmissão de pacotes perdidos é fator decisivo na degradação da performance. No teste 6, por exemplo, a banda recebida pelo nó móvel é 88% menor do que a esperada, fato que não ocorreu nos testes com UDP. Este mecanismo de *slow start* influenciou também no tempo Tmpa, que no UDP girou em média de 200 milisegundos e no TCP 400 milisegundos.

4.3.2 Implementação em redes IPv4

A implantação da arquitetura MPA para uma rede de acesso com IPv4 é similar à anterior com IPv6, porém com algumas mudanças:

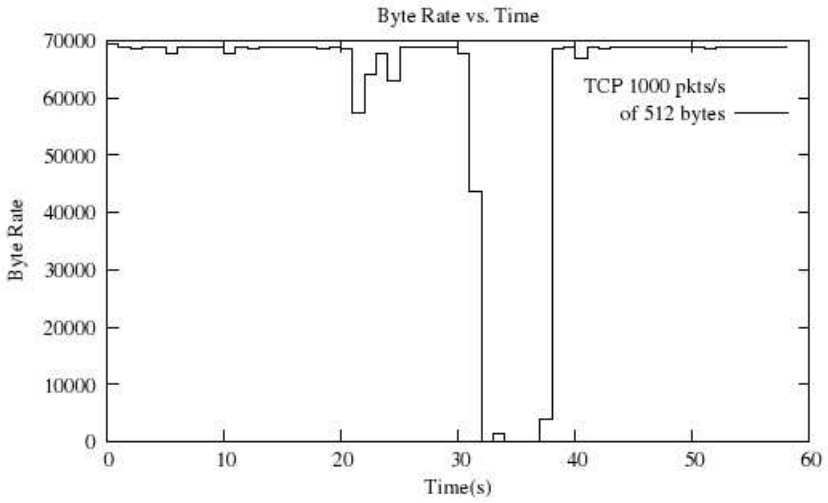

Figura 4.14: Tráfego TCP recebido pelo nó móvel no experimento 6.

- Não foi adicionado nenhum *software* adicional no nó móvel. No caso do IPv6 foi necessário inserir uma aplicação que enviava uma mensagem de *Router Solicitation* para que o processo de configuração de *gateway* padrão não dependesse da aleatoriedade das mensagens de *Router Advertisement* enviadas pelo radvd. Isto foi necessário pois o DHCPv6 não se responsabiliza por configurar este parâmetro no nó móvel. Não é o caso do DHCP para IPv4, ou DHCPv4. Este configura o *gateway* padrão no cliente DHCP, permitindo que ele seja suficiente para a conectividade do nó na camada de rede.

- Não foi necessário instalar uma nova suite DHCP. O sistema operacional Slackware 10.2 possui um servidor DHCP e um cliente, os quais foram utilizados. O único *software* adicional necessário foi um outro *relay agent* [18] nos roteadores de acesso, pois o disponível na distribuição não permitia a mudança do *circuit-id* da mensagem de *relay*.

- Os equipamentos utilizados foram um *notebook* Dell 510 e um *desktop* Dell OptiPlex com placa Wi-Fi Linksys.

Neste cenário com IPv4 medimos o tempo de camada de rede no *handover* o qual não temos controle e é dependente das rotinas do sistema operacional. Desta forma verificamos qual o atraso real da nossa arquitetura no acesso a rede.

Testes de tempo de *handover* na camada de rede

A topologia para estas medidas é mais simples do que a usada na figura 4.16. No cenário da figura 4.15 duas subredes de prefixos diferentes foram configuradas e o mesmo tráfego CBR (*Constant Bit Rate*, tráfego de fluxo contínuo e regular) é replicado pelas duas subredes a partir do gerador de tráfego MGEN, o mesmo utilizado nos experimentos com IPv4. O nó móvel ao realizar a mudança de ponto de acesso deve configurar endereço de rede e *gateway* padrão. Nas duas máquinas que atuam como roteadores de acesso foram utilizados *relay agents* DHCP. As medições foram realizadas em

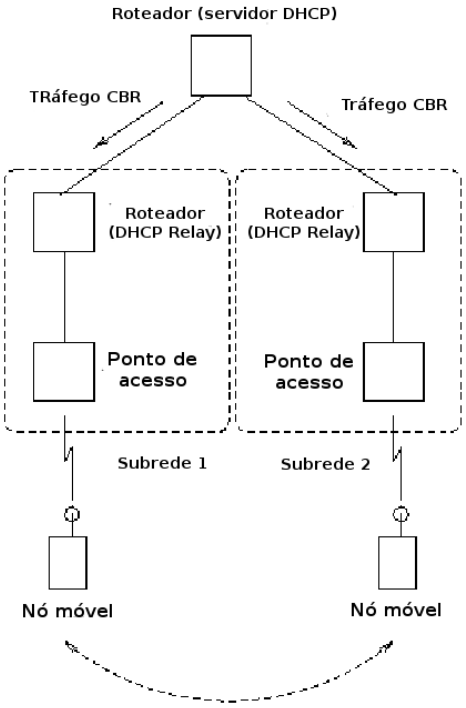

Figura 4.15: Topologia de teste para medição do overhead de camada 3.

duas condições distintas. A primeira delas é utilizando a ferramenta padrão de associação Wi-Fi do Linux e a outra utilizando uma versão modificada desta ferramenta. As máquinas envolvidas são nós de nosso rack com a mesma configuração e especificação e as antenas são duas Linksys WAP54G idênticas.

Utilizando as ferramentas padrão do Linux obtivemos os resultados da tabela abaixo. A partir dos tempos de camada dois medidos na tabela 4.1 no cenário do IPv6 medimos que o tempo de camada dois para o Dell 510 é de 280 milisegundos e para o OptiPlex é 300 milisegundos, o qual deve ser descontado do tempo medido para obtermos somente o *overhead* de camada de rede. Estes tempos são 2,620 segundos para o Dell 510 e 2,6 segundos para o Dell OptiPlex.

Equipamento	Perda de pacotes no *download*	Tempo de *overhead* camada 2 e 3
Dell 510	144,7	2,9 segundos
Dell OptiPlex Desktop	146,5	2,9 segundos

Tabela 4.2: *Overhead* de camada 2 e camada 3 em termos de perda de pacotes no *download*.

Estes tempos são altos para praticamente qualquer aplicação que exige algum requisito de performance e estabilidade de conexão. Este comportamento pode ser explicado pela maneira a qual o utilitário *iwconfig*, o qual é responsável pela associação e reassociação em camada dois, funciona por padrão. Se desejamos associar a interface Wi-Fi “wlan0” com a rede Wi-Fi “ap5” com chave criptográfica “123456789” o comando é:

iwconfig eth1 essid wlan key 123456789

Ao executar este comando no terminal o mesmo retorna imediatamente. Todavia o procedimento de associação na camada dois não necessariamente foi completado. Essa assincronia do utilitário com os eventos da camada de enlace são críticos em um *script* de associação onde a chamada do cliente DHCP sucede este comando de associação ao ponto de acesso. O caso mais comum é o cliente DHCP (no caso o dhcpcd) enviar uma mensagem em broadcast DHCPDISCOVER, a qual não será respondida pois o procedimento de *handover* na camada de enlace não foi concluído. Entendendo que a rede pode estar congestionada, o cliente entra em um estado de espera e só vai tentar retransmitir o DHCPDISCOVER depois de um certo *timeout*. Este procedimento padrão do cliente DHCP gera tempos excessivos para a configuração do endereço de rede.

Outra possível fonte de atraso é a necessidade de terminar o cliente DHCP e reiniciá-lo do zero para completar a transação DHCP na mudança de ponto de acesso[3].

Para contornar esta operação do comando *iwconfig* encontramos uma maneira de simular um comportamento síncrono do mesmo. Após a execução do comando, o *script* que utilizamos para fazer o *handover* começa a executar o comando *arping* em intervalos de 10 em 10 milisegundos. O comando *arping* é similar ao comando *ping*, entretanto ele envia mensagens ARP Requests ao invés de ICMP Echo Requests como no caso do *ping*. A grosso modo, podemos falar que o *arping* é um "ping de camada de enlace". O alvo é o endereço MAC do novo ponto de acesso. Somente a partir do momento que o primeiro *arping* reply é recebido (ou seja, é garantido que a associação em camada dois está concluída) o cliente DHCP é disparado.

Com esta versão alterada do comando *iwconfig* obtivemos os resultados da tabela 4.3. Notadamente os tempos foram melhores com a modificação, pois evitamos o timeout do cliente DHCP quando este encontra a camada de enlace não preparada para transmissão e cai em um tempo de espera.

Equipamento	Perda de pacotes no *download*	Tempo de *overhead* das camada 2 e 3
Dell 510	80	1,6 segundos
Dell OptiPlex Desktop	28	560 milisegundos

Tabela 4.3: *Overhead* de camada 2 e camada 3 em termos de perda de pacotes no *download* usando o comando *iwconfig* alterado.

Overhead da arquitetura MPA

Calculamos o tempo de atraso da arquitetura tanto para redes de transporte IP/IP como também MPLS. A figura 4.16 mostra a topologia usada, a qual é similar à usada nos testes com IPv6. As máquinas consistem em nós do nosso rack, todas com configuração idêntica, assim como os quatro pontos de acesso Linksys WAP54G utilizados.

O *daemon* rsvpd está presente em todas as máquinas, de maneira similar aos testes com IPv6. A máquina DCA03 continuou com o papel de servidor DHCP. Nos roteadores de acesso utilizamos um

[3]Uma mensagem DHCP Renew não é suficiente pois, na mudança do ponto de acesso, o *gateway* padrão previamente configurado - o roteador de acesso anterior - não é alcançável pois não se encontra mais no mesmo enlace. Observe que este procedimento está previsto no funcionamento do DCHP (figura 4.2).

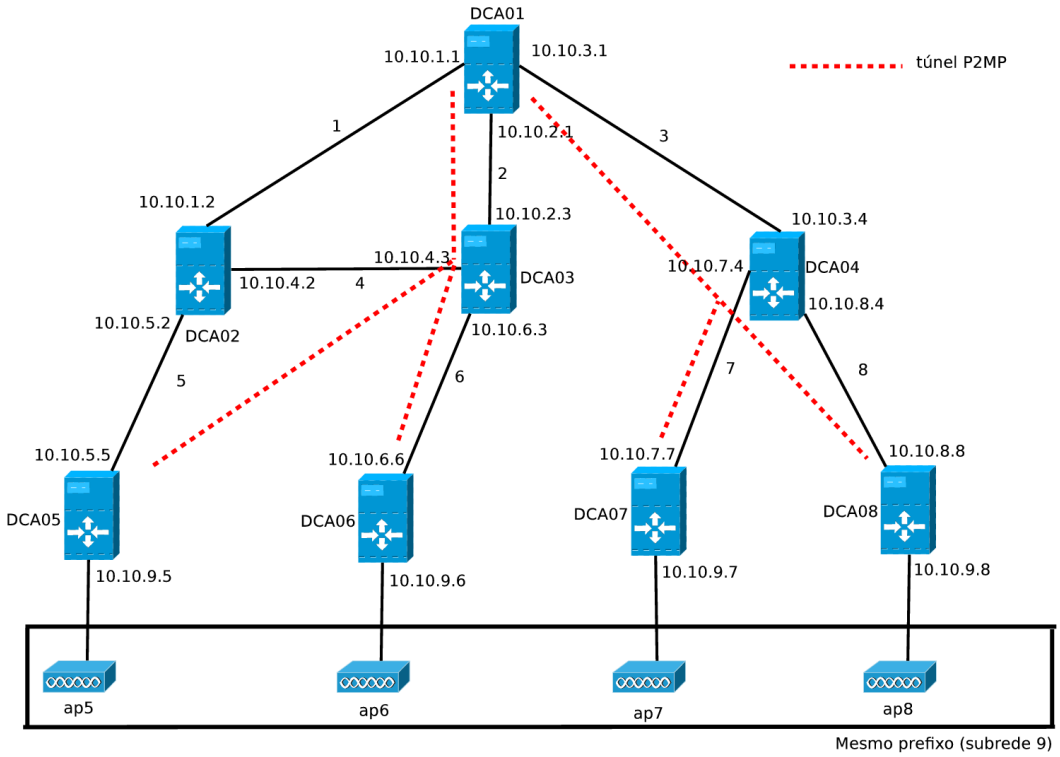

Figura 4.16: Topologia da implementação utilizando IPv4.

relay agent que possui suporte para editar o *circuit-id* enviado para o servidor [21]. Para este *relay* o valor do campo é igual a interface do MAR de egresso que é usada para encaminhar a mensagem para o servidor. Em todos eles (DCA05, DCA06, DCA07, DCA08) esta interface é a eth1. Renomeamos esta interface em cada um dos MARs de egresso de maneira que seja possível a identificação deste pelo servidor DHCP da seguinte forma:

- Interface eth1 da máquina DCA05 - relay5
- Interface eth1 da máquina DCA06 - relay6
- Interface eth1 da máquina DCA07 - relay7
- Interface eth1 da máquina DCA08 - relay8

Assim a diferenciação é feita a partir do parâmetro *circuit-id* de maneira similar ao cenário IPv6.

Utilizando o *iptables*, instalamos filtros para as mensagens DHCP que seriam responsáveis por gerar *triggers* nos roteadores de acesso, no caso da topologia representada na figura 4.16, as máquinas DCA05, DCA06, DCA07 e DCA08. Os pacotes são enviados para o espaço do usuário, no qual uma aplicação em linguagem C é responsável por retirar destas mensagens o identificador único do nó móvel (seu endereço MAC correspondente) e o endereço atribuido a este nó pelo servidor DHCP. Com estas informações a aplicação envia um *trigger* para o *daemon* RSVP local e devolve o pacote para o *kernel* para este encaminhá-lo normalmente.

Neste cenário de testes o nó móvel se movimentou entre as 4 subredes de acesso, realizando *handover* de um ponto de acesso ao outro. Um tráfego CBR foi gerado a partir do roteador de ingresso, máquina DCA01, usando o MGEN. As medidas foram realizadas com diferentes tipos de tráfego, como nos testes com IPv6. O total de pacotes perdidos em cada mudança de antena ocorre devido ao somatório do tempo de *handover* na camada de enlace, na camada de rede e o *overhead* da arquitetura MPA. Embora tenhamos calculado os dois primeiros, não podemos simplesmente medir o tempo total de *handover* e descontar os tempos de camada dois e três para contabilizarmos o tempo atribuído da arquitetura. Isto porque os procedimentos da MPA começam praticamente em paralelo com os eventos da camada 3, na configuração de endereço de rede do nó. Note que este procedimento é diferente do usado nos testes com IPv6, o qual desconsidera este fato.

Para estimar corretamente qual o peso da arquitetura MPA no tempo de acesso do nó móvel após um *handover* sincronizamos o relógio de todos os 8 nós da rede de testes e marcamos o tempo de processamento de uma mensagem de RESV nos MARs. Este tempo de processamento inclui os procedimentos normais de operação do RSVP-TE mais a atualização da tabela de roteamento móvel. No caso da rede de transporte IPv4, a atualização de rotas é feita pelo módulo IPv4, enquanto que para MPLS esta operação é feita pelo módulo MPLS.

O *overhead* medido a partir deste procedimento foi de 150 milisegundos em média, tanto para IP quanto MPLS. Este tempo é completamente independente dos tempos de *handover* de camada dois e três de qualquer nó móvel, pois é uma característica da rede de transporte. Observando os resultados obtidos na tabela 4.4 podemos notar que estes são similares aos tempos da tabela 4.3, os quais são referentes às medidas feitas sem a MPA. Podemos concluir que a arquitetura não impõe nenhuma latência adicional no processo de *handover*, uma vez que a MPA age em paralelo a partir da primeira interação do cliente DHCP do nó móvel com o servidor DHCP. Não há como evitar o tempo de camada 3 devido à reconfiguração de *gateway* padrão após um *handover*, o qual necessariamente inclui uma interação do cliente DHCP com o servidor. A atuação da MPA ocorre em paralelo com o processo de renovação de endereço do DHCP.

Equipamento	Perda de pacotes no *download*	Tempo de *overhead* total
Dell 510	80	1,6 segundos
Dell OptiPlex Desktop	28	560 milisegundos

Tabela 4.4: *Overhead* total (camada 2, camada 3 e MPA) em termos de perda de pacotes no *download* usando o comando iwconfig alterado.

Medições usando outro tipo de acesso

Os testes acima descritos foram feitos usando-se WEP em todos os pontos de acesso. A utilização de um outro esquema de segurança interfere diretamente no acesso, conforme já discutido no capítulo 3. Fica a seguinte questão: quanto tempo a associação a um ponto de acesso demora com um esquema de segurança diferente. Para o caso WPA deixaremos para o capítulo seguinte, pois é evidente que os tempos são maiores [4]. Todavia e se retirarmos o WEP dos pontos de acesso, deixando-os abertos

[4]Devemos lembrar que WPA geralmente requer servidor de autenticação, o qual muitas vezes pode exigir certificados e verificações de senha.

para qualquer conexão? O provável ganho de performance (uma vez que não há necessidade de transmissão de chave criptográfica) justifica a falta de privacidade e segurança dos dados?

Refizemos o teste de tempo de *handover* na camada 2 utilizando o notebook Acer, com placa Wi-Fi Broadcom 4318, considerando três níveis de segurança:

- Cenário 1: sem qualquer segurança no ponto de acesso.
- Cenário 2: filtragem por endereço MAC no ponto de acesso.
- Cenário 3: chave WEP de 52 bits no ponto de acesso.
- Cenário 4: chave WEP de 104 bits no ponto de acesso.

Este teste foi feito de maneira similar ao teste de camada de enlace feito anteriormente, cuja topologia usada encontra-se na figura 4.4, com pontos de acesso idênticos e nós de rede idênticos. Os resultados estão na tabela 4.5. Como podemos observar não é vantajoso no ponto de vista de performance de *handover* abrir mão de um nível de segurança maior nestes cenários testados.

Cenário 1	Cenário 2	Cenário 3	Cenário 4
19 (380 ms)	19 (380 ms)	19 (380 ms)	19 (380 ms)

Tabela 4.5: Perda de pacotes em *download* para diferentes cenários de segurança utilizando o *notebook* Acer com placa Wi-Fi Broadcom 4318.

No capítulo seguinte discutiremos qual o peso de uma autenticação do tipo WPA no *handover*.

Comentários sobre os testes realizados

A partir dos resultados obtidos concluímos que o tempo da arquitetura MPA para restabelecimento de túneis, reroteamento e redirecionamento de tráfego não varia com a característica do tráfego com destino ao nó móvel e dos tempos individuais de camada de enlace e rede de cada *hardware*.

Outra conclusão interessante é a variação dos resultados obtidos para *hardwares* distintos. A padronização dos drivers de diferentes fabricantes de placas de rede e de pontos de acessos Wi-Fi não está totalmente estabelecida como no caso de redes cabeadas Ethernet. Isto torna mais difícil estimar qual o peso das diferentes soluções de acesso para micro-mobilidade pois, dependendo do caso, uma solução ineficiente passa a ser tolerável uma vez que o tempo de reassociação do driver da placa Wi-Fi com o ponto de acesso é muito mais alto do que o *overhead* da solução adotada.

Capítulo 5

Proposta de acesso utilizando WPA e RADIUS

No capítulo anterior apresentamos os modelos de acesso adotados nos testes da arquitetura MPA com IPv4 e IPv6. Neste capítulo discutiremos melhorias de segurança e de performance que podem ser aplicadas ao acesso a redes Wi-Fi. Inicialmente apresentamos o modelo de autenticação WPA com servidor de autenticação, o mais seguro de todos atualmente, e qual a influência e o peso desta solução na performance de *handover*. A seguir discutiremos como podemos utilizar o protocolo RADIUS para melhorar o desempenho da arquitetura MPA.

5.1 WPA com servidor de autenticação

Como discutido no capítulo 3, o modelo de acesso via servidor de autenticação proporciona hoje o maior nível de segurança para o acesso a uma rede sem fio. Todavia, o custo para a performance de *handover* é relativamente alto. O protocolo utilizado para esta autenticação é o RADIUS (*Remote Authentication Dial In User Service*) [22]. Ele descreve como que um cliente pode negociar credenciais para liberar acesso à camada de enlace.

A figura 5.1 ilustra o funcionamento básico do protocolo RADIUS. Um nó móvel que possua um suplicante[1] compatível com IEEE 802.11i pode se associar a um ponto de acesso com suporte a RADIUS informando suas credenciais que devem ser encaminhadas para o servidor de autenticação (servidor RADIUS), o qual deve autorizar o acesso. Estas credenciais são enviadas para o para o servidor de acesso através de uma mensagem de Access-Request.

WPA é baseado no EAP - *Extensible Authentication Protocol* - o qual descreve somente o formato das mensagens EAP, não definindo como elas devem interagir. Chamamos de método EAP qualquer outro protocolo de segurança que encapsule as mensagens EAP. A escolha do método EAP a ser utilizado interfere no protocolo entre o servidor e o cliente RADIUS e também no tempo de associação. A escolha do método a ser utilizado deve levar em consideração qual propriedade desejamos para nosso acesso. As 5 características principais que os métodos WPA possuem são:

[1] Suplicante é um termo usado pelo padrão IEEE 802.1X que define a entidade que se encontra em uma das pontas de um *link* ponto-a-ponto que deseja ser autenticada pela entidade autenticadora que se localiza na outra ponta do *link*.

Figura 5.1: Funcionamento básico do protocolo RADIUS para acesso Wi-Fi.

- **Troca dinâmica de chaves**: Distribuição de uma chave criptográfica única para cada nó móvel. Sem esse atributo todos os nós móveis devem usar a mesma chave criptográfica, como no caso do WEP.
- **Autenticação mútua**: O ponto de acesso deve autenticar o cliente Wi-Fi. Neste caso, o cliente também autentica o ponto de acesso para certificar-se de que não está se associando com um ponto de acesso clonado.
- **Autenticação baseada em senha**: O nó móvel provê uma senha para ser autenticado na rede de acesso. Esta senha pode ser enviada utilizando-se código *hash* ou em texto simples.
- **Certificado digital**: Utilização de um certificado digital para autenticação do nó móvel. Exige uma infraestrutura que propicie a elaboração de certificados individuais para cada usuário. Isto inclui usar ou criar uma autoridade certificadora para criar os certificados.
- **Túnel criptografado**: Estabelecimento de um túnel criptografado para encaminhamento das mensagens de autenticação e das chaves criptográficas.

Cada método EAP pode ser definido como uma combinação destas características. Alguns dos métodos EAP mais utilizados são listados a seguir:

- **EAP-MD5**: Definido na RFC 3748, este método consiste na troca de senha de autenticação entre o nó móvel e a rede de acesso usando senha codificada com *hash* MD5.
- **EAP-TLS**: Utiliza o mecanismo de autenticação via certificados do TLS (*Transport Layer Security*, esquema de segurança discutido no capítulo 2) dentro do padrão EAP para prover autenticação mútua. Tanto o nó móvel quanto o servidor RADIUS deve ter um certificado SSL

de autenticidade assinado por uma entidade certificadora que ambos confiam. Foi o primeiro método EAP a ser considerado padrão pelo IETF.

- **EAP-TTLS**: Extendendo o EAP-TLS, o EAP-*Tunnelled Transport-Layer Security* ou EAP-TTLS dispensa o uso de um certificado no lado do cliente, sendo necessário somente o servidor RADIUS comprovar sua autenticidade para o nó móvel. Após esta verificação de legitimidade do servidor e da rede de acesso, um túnel é estabelecido para que o nó móvel envie suas credenciais para o servidor para liberação do acesso. Neste cenário é possível a utilização do EAP-MD5 para a autenticação dentro deste túnel, protegendo a troca de mensagens de ataques como *sniffing* e homem do meio[2].

- **EAP-MSCHAP**: *Microsoft Challenge Handshake Accept Protocol*, ou MSCHAP, permite autenticação mútua e verificação via envio de senha através de um algoritmo desenvolvido pela Microsoft. É mais comumente usado em conjunto com outros métodos EAP, como o EAP-TTLS ou o PEAP. Atualmente está em sua segunda revisão, MSCHAPv2.

- **EAP-GTC**: O EAP-GTC (*Generic Token Card*) utiliza o sistema de certificados para autenticar o nó móvel na rede de acesso. Um texto gerado aleatoriamente pelo servidor é enviado para o nó móvel - chamado também de *challenge* - o qual deve ser processado a partir de uma chave que é denominada *token* de segurança. A resposta correta a este *challenge* comprova que o nó móvel validou seu *token*, o qual neste caso tem a mesma função de um certificado SSL convencional.

- **PEAP**: Desenvolvido a partir de um esforço em conjunto da Microsoft, Cisco Systems e RSA Security, é um padrão aberto assim como o seu concorrente direto EAP-TTLS. Seu funcionamento é similar ao EAP-TTLS, com a exceção de que dentro do túnel somente é permitido o uso dos métodos EAP-MSCHAPv2 e EAP-GTC (*Generic Token Card*). A combinação PEAP / EAP-MSCHAPv2 é o segundo maior padrão EAP suportado no mundo todo, perdendo apenas para o EAP-TLS.

Podemos resumir estes métodos EAP a partir das características definidas anteriormente. A tabela 5.1 apresenta um resumo destes métodos e seus respectivos atributos.

Neste trabalho realizamos experimentos utilizando o servidor de autenticação freeradius [23]. O objetivo é usar WPA para a realização de testes de associação para avaliar qual o impacto desta segurança adicional em relação ao WEP na performance de *handover*.

Dos métodos EAP descritos anteriormente, realizamos testes de associação utilizando EAP-MD5, EAP-TLS, EAP-TTLS/MD5 e PEAP/MSCHAPv6. Utilizamos duas antenas Linksys WAP54G e um *notebook* Acer Aspire 3003Lci com placa Wi-Fi Broadcom 4318 em um cenário simples, similar à ilustração da figura 5.1 porém com dois pontos de acesso. Dez medidas foram feitas para cada caso. Um tráfego UDP CBR gerado pelo aplicativo MGEN em direção ao nó móvel é iniciado no momento da associação. O tempo é calculado verificando-se o número de seqüência do primeiro pacote a chegar no nó móvel. O tempo considerado é o intervalo do começo da associação disparada pelo nó móvel até o envio da mensagem RADIUS ACCESS-ACCEPT [3] enviada pelo freeradius e

[2]Mais detalhes sobre os ataques de segurança mais comuns no apêndice A.

[3]Esta é a mensagem enviada pelo servidor de autenticação ao cliente RADIUS, autorizando a liberação do enlace para o nó móvel correspondente. Maiores detalhes sobre o protocolo RADIUS em [22].

Método EAP	Troca dinâmica de chaves	Autenticação mútua	Autenticação baseada em senhas	Certificado digital	Túnel criptografado
EAP-MD5			x		
EAP-TLS	x	x		x	x
EAP-TTLS	x	x	x		x
EAP-MSCHAP		x	x		
EAP-GTC				x	
EAP-PEAP	x	x			x

Tabela 5.1: Descrição simplificada de cinco dos principais métodos EAP.

registrada em seu arquivo de log com a hora de envio. Evitamos o *cache* da antena e do servidor freeradius considerando intervalos maiores do que dois minutos para a realização de dois *handovers* consecutivos.

Como esperado, o tempo de associação é consideravelmente mais alto do que usando WEP. A tabela 5.2 mostra que a menor média conseguida foi de 2,5 segundos, tempo mais alto do que o medido com WEP no capítulo anterior.

Método EAP	Tempo médio de *handover* sem *cache*
EAP-MD5	2,5 segundos
EAP-TLS	8,3 segundos
EAP-TTLS	6,2 segundos
EAP-PEAP	6,5 segundos

Tabela 5.2: Testes de associação feitos com diversos métodos EAP.

Ressaltamos que estas medições foram feitas evitando o *cache* de conexão da antena e do servidor freeradius. Em reassociações consecutivas (com um intervalo menor do que um minuto) o tempo médio fica em torno de 50 milisegundos, similar ao caso com WEP. Sem considerar este *cache*, é fácil observar que não podemos esperar performance de *handover* utilizando qualquer um dos métodos EAP testados. Concluímos que, a partir dos resultados obtidos, não é possível ter mobilidade de alta performance usando-se WPA com qualquer um dos métodos EAP acima testados. Para tentar resolver este problema propomos algumas alternativas.

A primeira é a utilização da nova versão do WPA, WPA2. Esta possui um mecanismo de pré-autenticação para minimizar a latência no momento do *handover*, podendo iniciar o processo de autenticação a um outro ponto de acesso enquanto é mantida a associação com o ponto de acesso atual. Como os pontos de acesso que temos não possuem suporte a WPA2, tomamos como base os resultados da referência [24] que comprovam que a pré-autenticação reduz o tempo de *handover* para cerca de 300 milisegundos.

A partir desta possibilidade, é interessante pensar em um modelo de acesso que usufrua das mensagens de autenticação para gerar eventos de camada de rede. No caso da MPA, elaboramos uma maneira de gerar estes eventos com uma antecedência ainda maior do que utilizando as mensagens

do DHCP.

Uma outra solução que propomos, a qual se aplica também no caso do WPA, é a utilização do protocolo IAPP (*Inter Access Point Protocol*) o qual é descrito na norma IEEE 802.11f. Este protocolo descreve a troca de mensagens entre os pontos de acesso envolvidos no *handover* para que não haja a necessidade de fazer novamente o procedimento de associação se um dos pontos de acesso da mesma rede Wi-Fi já autorizara o nó móvel.

Finalmente, apresentamos também como proposta de solução um modelo de acesso onde todos os pontos de acesso atuam a partir de uma rede roteada com DHT (*Distributed Hash Table*).

5.2 WPA2 com eventos a partir de mensagens RADIUS

O uso do WPA2 não é obrigatório para nenhuma das estratégias a seguir descritas. Contudo, comprovamos anteriormente que o uso do WPA impossibilita uma mobilidade eficiente devido aos altos tempos de *handover*, tanto nas primeiras associações quanto em reassociações.

A vantagem desta abordagem é o fato de que mensagens RADIUS trafegam na rede de transporte entre ponto de acesso e servidor de autenticação. No caso do DHCP era necessário esperar todos os eventos da camada dois se concluírem para enviar uma mensagem de DHCP DISCOVERY do nó móvel em *broadcast* para o ponto de acesso ao qual este se associou.

Fizemos um estudo das mensagens RADIUS trocadas entre o ponto de acesso e o servidor. Para gerar eventos na arquitetura MPA devemos ter os seguintes atributos definidos: endereço de rede do ponto de acesso, endereço de rede do nó móvel e endereço MAC do nó móvel. A primeira mensagem enviada pelo ponto de acesso ao servidor chama-se ACCESS-REQUEST e é a mensagem que contém todas as credenciais enviadas pelo nó móvel para ser autenticado. Nesta mensagem encontramos tudo o que precisávamos, com exceção do endereço de rede do nó móvel.

Na RFC 2865 que define o RADIUS existe um campo chamado Framed-IP-Address o qual indica o endereço IP a ser configurado ou já configurado pelo nó móvel. Entretanto, nesta mesma RFC consta que a mensagem de ACCESS-REQUEST enviada do ponto de acesso para o servidor RADIUS não é obrigada a incluir este campo. De certa forma é compreensível que isto não ocorra, pois a camada de enlace não deve consultar ou fornecer informações a respeito da camada acima. Esta informação é essencial para disparar o *trigger* de camada de rede de maneira eficiente.

Uma solução possível para este problema é admitir que este atributo pode ser eventualmente enviado e considerá-lo presente na mensagem de ACCESS-REQUEST. A outra solução é propor uma mudança no protocolo IEEE 802.11f obrigando a inclusão deste atributo na mensagem. A solução que aplicamos é utilizar a capacidade do servidor RADIUS de configurar os parâmetros devolvidos na mensagem de ACCESS-ACCEPT.

Para cada usuário incluído no banco de dados do freeradius é possível configurar um conjunto de parâmetros da mensagem de ACCESS-ACCEPT que sai do servidor para o ponto de acesso, confirmando a autenticidade do nó móvel. A questão é como prever o endereço de rede do nó móvel antes do cliente DHCP do mesmo iniciar a negociação de endereço com o servidor DHCP.

Na MPA adotamos o endereçamento via DHCP, sobre o qual temos controle total sobre as faixas de endereços servidas nos pontos de acesso. Em um procedimento que batizamos de “configuração conjunta DHCP e RADIUS” podemos obter resultados interessantes. Outra solução que adotamos insere um novo elemento na arquitetura MPA denominado MM (*Mobility Manager*).

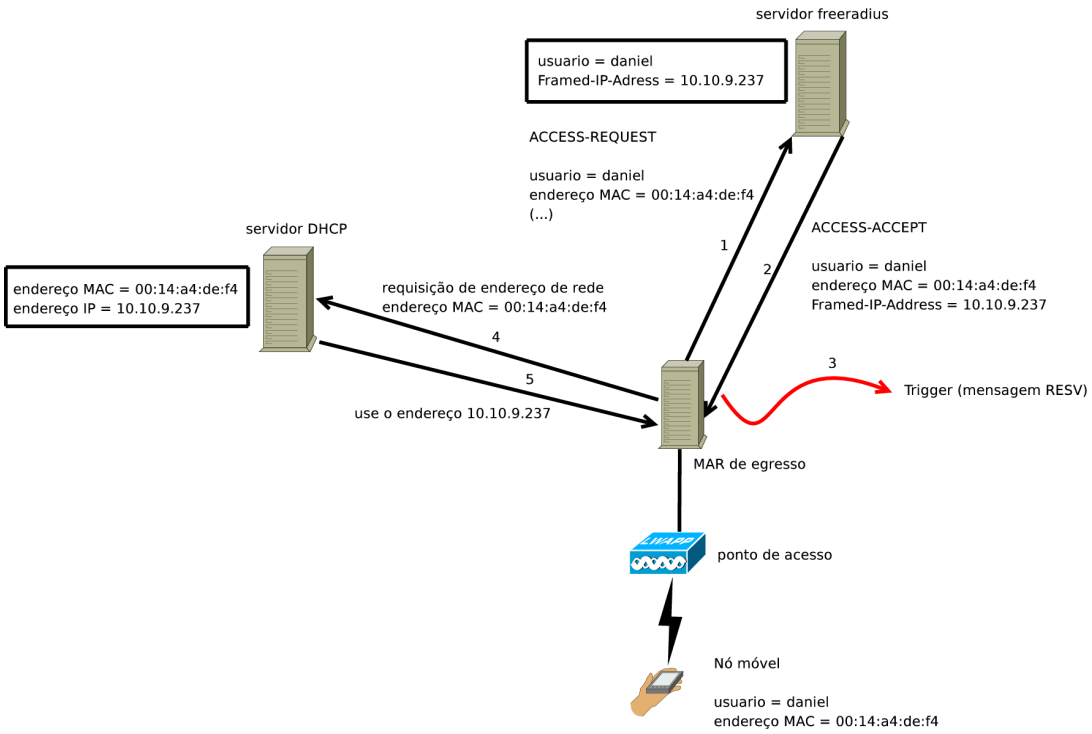

Figura 5.2: Exemplo de um acesso a uma rede usando configuração conjunta DHCP e RADIUS para a geração de trigger na arquitetura MPA.

5.2.1 Configuração conjunta DHCP e RADIUS

Para o servidor DHCP o identificador único do nó móvel é o endereço MAC, sendo possível atribuir endereços IP exclusivos para determinados endereços MAC. Já o freeradius leva em consideração o nome de usuário que chega pela mensagem de ACCESS-REQUEST para especificar atributos exclusivos.

Esta configuração é possível se garantirmos que, para cada nome de usuário configurado no servidor freeradius, haverá uma entrada na configuração do servidor DHCP que configure o endereço IP equivalente ao valor do parâmetro Framed-IP-Address para aquele nome de usuário. Em outras palavras, mapeamos o nome de usuário no freeradius como endereço MAC no servidor DHCP.

A figura 5.2 mostra um exemplo desta configuração. Considere um nó móvel que vai iniciar uma associação ao ponto de acesso, utilizando o nome de usuário “daniel” e o respectivo endereço MAC. Previamente configurado pelo administrador da rede, o servidor freeradius tem um mapeamento para o usuário “daniel” no qual ele deve devolver na autenticação uma mensagem ACCESS-ACCEPT com o parâmetro Framed-IP-Address igual a 10.10.9.237. O servidor DHCP - que na figura estão representados separadamente contudo poderiam estar na mesma máquina física - possui em sua configuração uma entrada que diz que o endereço MAC 00:14:a4:de:f4 deve utilizar o endereço IP 10.10.9.237.

Ao iniciar a associação, a mensagem 1 da figura 5.2 inicia o processo de autenticação carregando nome de usuário, endereço MAC e outro atributos. Após o processo ser bem sucedido, a mensagem 2 confirma a autenticidade do nó móvel e autoriza o acesso, carregando o atributo Framed-IP-Address de valor 10.10.9.237. Um filtro de tráfego instalado no MAR de egresso captura este pacote e envia

um *trigger* (mensagem 3) utilizando-se deste atributo e dos demais que a mensagem carrega. Enquanto o reroteamento e alocação de recursos acontece, a negociação de endereço de rede acontece com o servidor DHCP, representado na figura com as indicações 4 e 5. Observe que não é por acaso que o endereço IP atribuído ao nó móvel é 10.10.9.237, o servidor DHCP foi configurado para tal. Desta forma, o evento de nossa arquitetura que realiza o alocamento de recursos, reroteamento e estabelecimento de túneis para este nó móvel acontece antes do mesmo estabelecer conexão com a camada de enlace.

Observamos que esta configuração é possível com qualquer outra arquitetura de micro-mobilidade que possa utilizar *trigger* a partir de eventos de camada de rede, como é o caso da MPA. Mesmo com esta enorme vantagem de tempo temos uma desvantagem: a associação do nome de usuário do RADIUS com o endereço MAC de uma interface de rede. O usuário "daniel" é vinculado ao endereço MAC 00:14:a4:de:f4. Se este usuário desejar acessar a rede via um outro nó móvel este trigger será inválido, pois o endereço MAC da interface de rede poderá mudar. Além disso, é necessário configurar cada endereço MAC manualmente no servidor DHCP, o que exige um esforço de configuração que não é exigido normalmente. Outra desvantagem é a permanência do mesmo endereço IP por toda a extensão da rede de acesso. Se qualquer um dos outros pontos de acesso servir uma rede Wi-Fi com prefixo diferente o *trigger* também será inválido.

Um fator que ameniza a desvantagem da configuração é o contexto no qual se aplica esta estratégia. Uma rede de acesso que possui um nível de segurança WPA2 com servidor de autenticação exige configuração da base de dados do servidor RADIUS para cada novo usuário. No caso de esquemas de autenticação como o EAP-TLS, deve ser emitido um certificado diferente para cada nó móvel que um mesmo usuário vá utilizar [4], o qual deverá instalar o certificado em cada um deles para ter acesso. Nestas redes com acesso restrito é aceitável que tenhamos configurações de servidor DHCP que fixem um endereço IP exclusivo para determinados endereços MAC. Tendo isto em vista, acreditamos que o fardo da configuração desta estratégia com DHCP e RADIUS é amenizado pelo esforço de configuração exigido normalmente pelo esquema de segurança WPA2.

5.2.2 MPA com *Mobility Manager* (MM)

Com os serviços de segurança do servidor RADIUS, uma nova sinalização trafega pela rede de acesso. Esta, que consiste nas mensagens do protocolo RADIUS, tem como objetivo autorizar o acesso do nó móvel recém chegado a um ponto de acesso. Esta sinalização, conforme dito na subseção anterior, é feita antes da troca de mensagens do protocolo DHCP, uma vez que a configuração de endereço necessita de um *link* de camada de enlace ativo. A proposta de configuração conjunta entre DHCP e RADIUS tem como principal vantagem a notificação de eventos gerando *triggers* com uma antecedência maior do que usando as mensagens do DHCP.

Este cenário inspirou uma outra estratégia de sinalização a ser utilizada na arquitetura MPA. A princípio, toda a configuração dos MARs é feita utilizando as mensagens do RSVP-TE, as quais são enviadas a partir de todos os MARs. A idéia é incluir um novo elemento na arquitetura que seja responsável por receber as notificações de mensagens RADIUS e DHCP e, a partir delas, enviar configurações de re-roteamento de túneis para os MARs correspondentes. Este elemento funciona

[4] Os certificados aceitos pelo servidor RADIUS são gerados utilizando-se o endereço MAC da interface - razão pela qual é necessário gerar um certificado para cada placa Wi-Fi distinta, mesmo que seja o mesmo usuário cadastrado no banco de dados do servidor RADIUS.

como um centralizador da sinalização da rede de acesso, o que, mesmo tendo a desvantagem de ser um único ponto de falha que comprometeria toda a rede de acesso, possui vantagens interessantes como a facilidade de aplicar políticas de acesso e de engenharia de tráfego em uma rede com arquitetura MPA.

Denominamos este novo elemento de MM (*Mobility Manager*), o qual atua de acordo com o significado de seu próprio nome. Trata-se de um gerente de mobilidade, capaz de saber a qualquer momento qual o estado atual da rede como um todo, tendo a capacidade de reconfigurar MARs baseando-se em políticas e regras pré-estabelecidas ou configuradas em tempo real por ações de gerência.

Geração de *triggers*

Este novo elemento dispara as notificações de camada 3 a partir de um processo similar ao descrito na seção 5.2.1, com a vantagem de desvincular as configurações do servidores DHCP e RADIUS, fazendo com que ambos possam operar de maneira independente.

A figura 5.3 mostra como o MM é capaz de disparar *triggers* de camada de rede. Note que representamos o servidor DHCP e o freeradius na mesma máquina física por questão de simplicidade. O nó móvel inicia sua primeira associação na rede de acesso no ponto de acesso AP1. O MAR de egresso MAR1 inicia o procedimento de autenticação que, eventualmente, será autenticado pelo servidor freeradius e terá acesso liberado (mensagem 1). Neste momento, um filtro para estas mensagens do RADIUS notificará o MM de que um nó móvel com o endereço MAC 00:14:a4:de:f4 se associou. Como não havia mapeamento MAC-IP para este nó, nada é feito até o endereço de camada de rede ser atribuído ao nó móvel pelo servidor DHCP (mensagem 2). Com estas informações, o primeiro *trigger* é disparado (mensagem 3). Quando o nó móvel realiza um *handover* para AP2, o mesmo procedimento com o servidor de autenticação é iniciado [5] . Mas, ao confirmar o acesso do nó móvel (mensagem 4), o filtro notifica o MM de que o endereço MAC 00:14:a4:de:f4 se associou a AP2. O MM já possui um mapeamento deste endereço MAC com um endereço IP, disparando o *trigger* antes de qualquer negociação com o servidor DHCP. Este processo opera corretamente mesmo que o nó móvel não efetue negociação de endereço IP após a associação de camada de enlace.

5.3 IAPP - *Inter Access Point Protocol*

IAPP [25] tem como objetivo permitir a transferência segura de estado e de contexto entre pontos de acesso de uma mesma rede Wi-Fi. Este protocolo parte do princípio de que, uma vez associado a um ponto de acesso a partir do procedimento padrão, este nó móvel é confiável para qualquer outro desta mesma rede pois obteve acesso autorizado por um ponto de acesso que pertence a mesma rede Wi-Fi, o qual está submetido às mesmas políticas de acesso. Desta forma não é necessário repetir o processo de associação por completo no momento da reassociação, mas sim transferir o estado e o contexto da associação com o ponto de acesso anterior para o atual.

A figura 5.4 descreve de maneira simplificada o funcionamento do IAPP. Temos dois pontos de acesso (AP1 e AP2) que compõem uma mesma rede de acesso Wi-Fi e possuem uma relação de confiança entre eles. Ambos usam um esquema de segurança WPA (ou WPA2, neste contexto não

[5] Mesmo com a pré-autenticação o servidor RADIUS deverá ser informado da reassociação do nó móvel

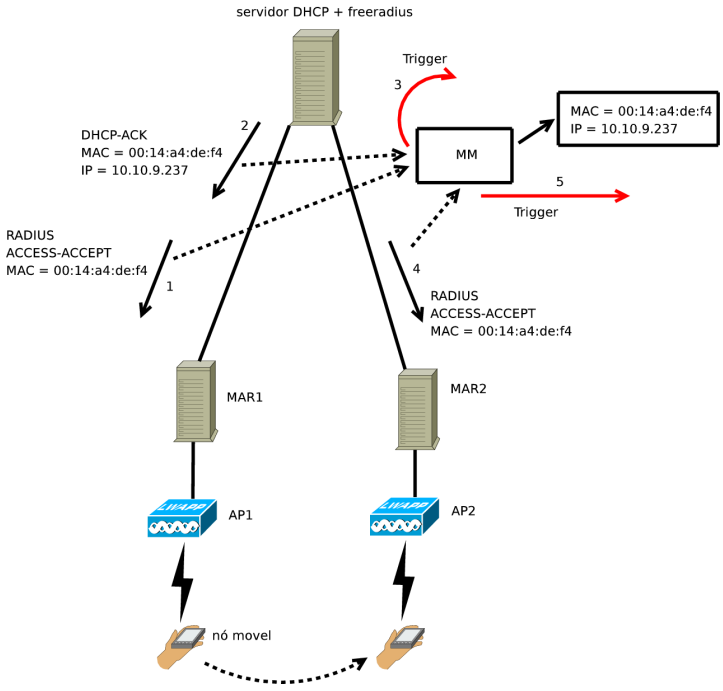

Figura 5.3: Exemplo de um acesso a uma rede usando a MPA com o Mobility Manager para a geração de triggers.

faz diferença) com um servidor de autenticação RADIUS compartilhado. Um nó móvel inicia sua primeira associação com a rede de acesso a partir do ponto de acesso AP1 na posição A.

O procedimento contempla as mensagens da figura 5.4. Ao iniciar a primeira associação (m1) o ponto de acesso não possui nenhum estado prévio deste nó móvel, sendo necessário fazer todo o procedimento de autenticação junto ao servidor RADIUS. O ponto de acesso encaminha a requisição para o servidor de autenticação (m2) o qual, depois de diversas mensagens intermediárias entre o ponto de acesso e ele, autoriza o acesso do nó móvel (m3 e m4). Após permanecer um intervalo de tempo conectado, o nó móvel inicia um movimento em direção a posição B, forçando um *handover* para o ponto de acesso AP2. O pedido de reassociação (m5) é encaminhado para AP2 informando que o nó móvel estava anteriormente associado a AP1. Ao invés de refazer a autenticação no servidor RADIUS, AP2 pede a AP1 (m6) que envie o estado e o contexto do nó móvel. Ao receber estas informações (m7) AP2 libera o acesso ao enlace para o nó móvel (m8). É evidente que o processo de *handover* é mais eficiente do que enviar uma nova requisição para o servidor RADIUS, objetivo principal do IAPP. Note que esta descrição geral do funcionamento omite diversas mensagens do protocolo [25].

A norma IEEE 802.11f, a qual descreve o protocolo IAPP, não é implementada pela maioria dos dispositivos Wi-Fi atuais. A referência [26] descreve um trabalho onde máquinas rodando OpenBSD 3.1 com uma implementação preliminar do IAPP foram usadas como ponto de acesso. Diversos testes de *handovers* utilizando-se IAPP com tempos de reassociação de camada de enlace na ordem de 23 milisegundos, valor abaixo dos 50 milisegundos recomendado para aplicações como voz sobre IP.

Entretanto, se não existe uma interação no servidor de autenticação a cada *handover*, não é possível enviar eventos na camada de rede anunciando a nova associação do nó móvel capturando-se as

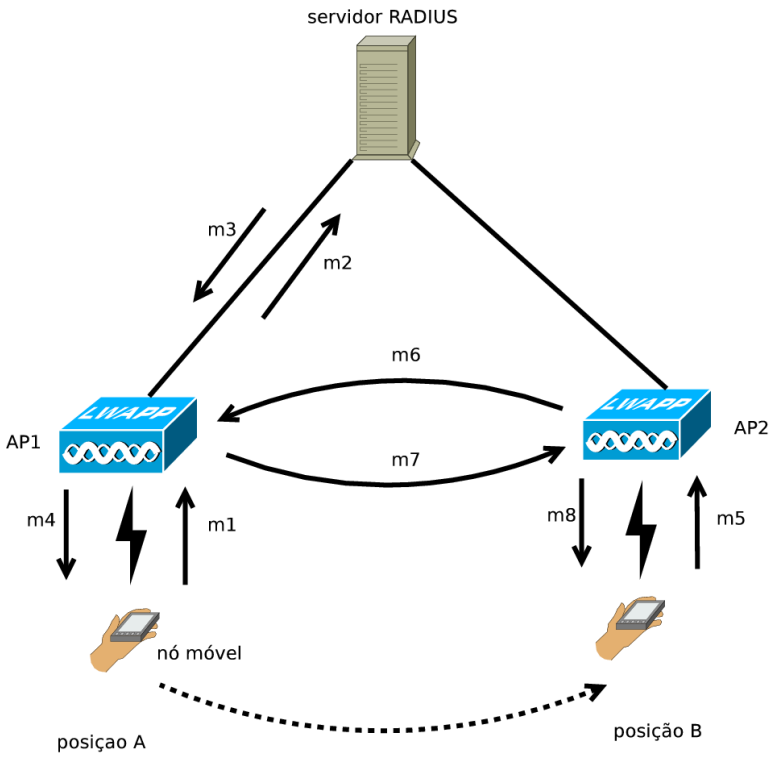

Figura 5.4: Descrição resumida do funcionamento do IAPP.

mensagens do RADIUS. Neste caso, este *trigger* pode ser gerado a partir da captura das mensagens de renovação de endereço do DHCP, como acontece na MPA.

A partir destas considerações e dos bons resultados da eficiência do IAPP obtidos na referência [26], acreditamos que o uso da norma IEEE 802.11f, tanto nos pontos de acesso quanto nos nós móveis, é uma solução para as altas latências de autenticação do RADIUS, em especial usando-se WPA, o qual não possui os recursos de pré-autenticação disponíveis no WPA2.

5.3.1 Geração de *triggers* usando IAPP

A comunicação entre os pontos de acesso é útil na notificação de eventos de camada três para a geração de *triggers*. Utilizando filtros de tráfego na comunicação entre os pontos de acesso, é possível interceptar mensagens do IAPP referentes a requisição de credenciais de segurança.

Na figura 5.4, vimos que a mensagem m6 é enviada pelo ponto de acesso AP2 requisitando informações sobre as credenciais do nó móvel para AP1, último ponto de acesso do nó móvel antes da nova associação a AP2. Um filtro de tráfego colocado entre estes dois pontos de acesso pode disparar um *trigger* indicando que o nó móvel acabou de chegar em AP2, uma vez que se AP2 está pedindo informações sobre nó móvel para AP1, então o nó móvel acabou de chegar em AP2.

Deve-se observar que esta solução opera apenas com a utilização do *Mobility Manager* pois a mensagem IAPP de solicitação de contexto não contém o endereço IP do nó móvel, mas apenas o seu endereço MAC.

Outra alternativa à utilização de filtros de tráfego é a utilização de *triggers* nativos do ponto do

acesso. Em outras palavras, o ponto de acesso seria responsável por disparar um *trigger* assim que certas mensagens do IAPP fossem recebidas ou enviadas. Embora exija um esforço para construir pontos de acesso especializados, ferramentas como o OpenWrt [29][6] possibilitam a customização do sistema operacional dos pontos de acesso, permitindo que novas funcionalidades sejam incorporadas.

5.4 Rede de acesso usando DHT - *Distributed Hash Table*

Uma DHT consiste em uma tabela de *hash* descentralizada. Os registros não são armazenados em uma única tabela, mas sim distribuídos em várias outras tabelas. A partir deste conceito simples, estudos de algoritmos de como agrupar as chaves em cada tabela e como otimizar a busca por um registro tornaram DHT uma solução viável para diversos problemas, dentre eles o de roteamento em redes P2P (*peer to peer*) de compartilhamento de dados, a qual é descentralizada por natureza.

Em uma rede *overlay* que usa DHT o roteamento e armazenamento de dados não é centralizado. Todos os nós possuem uma parte da tabela de roteamento e uma parte do armazenamento de forma que a rede, como um todo, possua todos os dados. Cada nó que ingressa na rede recebe um identificador único gerado por uma função de *hash*. A vizinhança desta rede é determinada a partir de como cada algoritmo posiciona cada endereço *hash* na topologia desta rede *overlay*.

Como podemos usar DHT em uma solução de micro-mobilidade? Alguns algoritmos de roteamento usando redes DHT, como o Uinta [27], levam em consideração a topologia da rede física na composição da rede overlay. Podemos imaginar então um esquema de acesso onde todos os pontos de acesso fazem parte de uma rede P2P com DHT onde todos que são vizinhos na topologia física o serão também na topologia *overlay*. Nesta rede P2P, os dados a serem armazenados são as informações de autenticação dos nós móveis na rede de acesso Wi-Fi. Com algoritmos adequados acreditamos ser possível implementar uma estratégia de acesso onde todos os pontos de acesso vizinhos a um determinado ponto de acesso terão em cache as informações dos nós atualmente associados ao mesmo.

A grande vantagem é o fato de que um nó móvel tende sempre a realizar *handover* no ponto de acesso mais próximo ao qual já está associado. Quando isto ocorrer, o ponto de acesso já possuirá em cache as informações de autenticação deste nó móvel, reduzindo drasticamente o tempo de associação em camada de enlace. Nos casos mais raros onde o nó móvel fará *handover* em algum outro ponto de acesso que não é vizinho do anterior, uma busca pela informação na rede DHT trará a informação de autenticação para o novo ponto de acesso em um tempo ainda sim menor do que uma transação com o servidor RADIUS. Na figura 5.5 temos um exemplo de como funcionaria este esquema de cache. O nó móvel está inicialmente associado ao ponto de acesso AP2. AP1 e AP3, vizinhos de AP2, guardaram em cache a informação de autenticação do nó móvel. O nó móvel faz *handover* para AP3 onde é prontamente autenticado devido ao cache já existente. AP2 e AP4 devem agora armazenar em seus caches a informação de autenticação, enquanto que a entrada do cache de AP1 expira depois de um certo tempo. Depois o nó móvel muda para AP4, o cache da autenticação fica em AP3 e AP5 e a entrada de AP2 e assim sucessivamente.

Fizemos testes com uma implementação de DHT denominada Chimera [28] em nosso rack para comprovar a eficácia da troca de mensagens do roteamento DHT. A conclusão que chegamos é que o

[6]OpenWrt é uma distribuição Linux feita exclusivamente para dispositivos embarcados, podendo ser instalada em pontos de acesso com modificações e funcionalidades adicionais ao comportamento normal do mesmo.

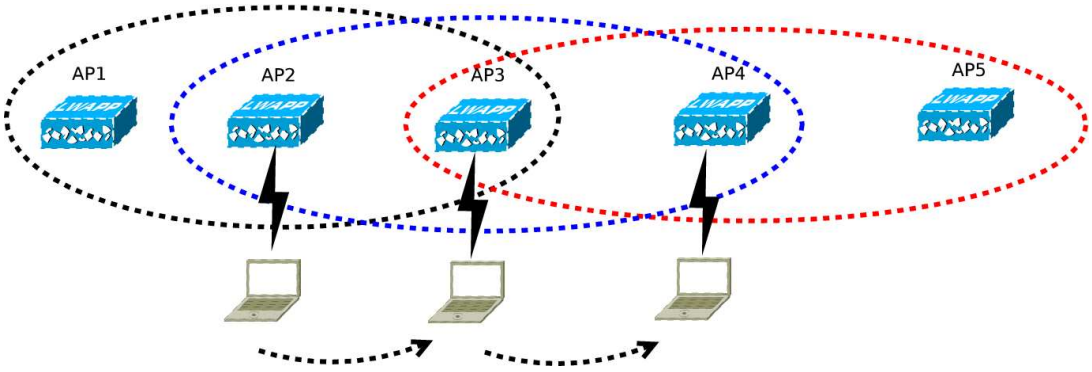

Figura 5.5: Rede de acesso que utiliza DHT para cacheamento de informações de autenticação.

roteamento é eficiente, com tempos médios de 0,05 ms para envio de mensagens para nós distantes na rede *overlay*, o que reforça a viabilidade da solução proposta acima.

Para viabilizar esta solução, é necessário implementar o *software* com este mecanismo de DHT nos pontos de acesso. Para tal, soluções como o OpenWrt [29] podem ser investigadas para substituir o sistema operacional de fábrica dos pontos de acesso por um que seja customizado com as aplicações e funcionalidades desejadas.

5.4.1 *Trigger* em uma rede de acesso com DHT

Tanto o IAPP como esta solução têm em comum a transferência de credenciais de segurança entre os pontos de acesso, de maneira a evitar qualquer interação adicional desnecessária com o servidor de autenticação. Dado estas igualdades, a geração de *triggers* para uma rede com DHT é similar a descrita na seção 5.3.1 para o cenário com IAPP. As diferenças estão nas mensagens que são consideradas para a geração do *trigger*. Ao invés de utilizar mensagens do IAPP, seriam utilizadas mensagens trocadas pelo algoritmo da DHT em uso. A solução de embarcar a geração do *trigger* nos pontos de acesso, conforme discutido na seção 5.3.1, também é válida para este cenário com DHT.

Capítulo 6

Conclusões e trabalhos futuros

Nesta dissertação de mestrado investigamos o impacto dos protocolos de acesso a rede Wi-Fi nas arquiteturas que suportam micro-mobilidade no nível de rede. Diversas arquiteturas de micro e macro-mobilidade foram estudadas e a arquitetura de micro-mobilidade MPA foi desenvolvida com a participação deste estudo.

Um desafio que foi vencido neste trabalho foi a não utilização de protocolos proprietários e de softwares adicionais no nó móvel. Por exemplo, o protocolo DHCP utilizado na implementação da arquitetura MPA é parte da grande maioria dos sistemas operacionais. Isso possibilita que as análises descritas neste estudo possam ser aplicadas utilizando-se qualquer *hardware* que possui uma interface Wi-Fi sem a necessidade de nenhum programa adicional. Esta transparência do processo de *handover* para o usuário, enquanto a inteligência do acesso concentra-se na rede de acesso, é um grande mérito desta arquitetura.

Outra vantagem da arquitetura MPA é a não necessidade de *hardware* especializado. Outras propostas dependem de roteadores e pontos de acesso especiais. Nossa solução de acesso e mobilidade é simples o suficiente para ser implantada com qualquer equipamento de rede disponível hoje.

Estas vantagens implicam na desvantagem de oferecer mobilidade com protocolos que não foram desenvolvidos para tal. Em especial, o DHCP não foi desenvolvido com o intuito de ser eficiente para situações de *handover* Wi-Fi. Ele foi desenvolvido para prover endereçamento de nós. Um atraso de segundos para a configuração de endereço IP via DHCP é tolerável para nós estacionários que não efetuam *handover*.

De maneira similar, o protocolo RADIUS não apresenta eficiência neste cenário. Trocas de certificados para comprovar autenticidade, além de exigir configuração manual no nó móvel, atrasam uma associação a um ponto de acesso de maneira a inviabilizar qualquer tentativa de *handover* eficiente. WPA2 possui recursos que contornam esta característica de maneira promissora, contudo, não são todos os equipamentos disponíveis hoje que são compatíveis com WPA2.

A utilização do protocolo IAPP prevê a segurança de acesso do servidor RADIUS junto com alta eficiência de *handover*, desde que todos os dispositivos de rede tenham suporte. Gerando-se os *triggers* a partir das mensagens do DHCP ou mesmo das mensagens IAPP, temos um modelo de acesso seguro, eficiente e com notificações de eventos na camada de rede.

As contribuições deste trabalho foram:

- Uma análise prática sobre as diversas formas de acessar uma rede Wi-Fi, considerando diferentes padrões de segurança e utilizando equipamento não especializado e protocolos bem

conhecidos.

- Técnicas para a obtenção de *triggers* de camada de rede através de eventos de DHCP e de eventos de camada dois a partir da operação do RADIUS. Aplicamos estas técnicas na arquitetura MPA. Entretanto, estas podem ser aplicadas em outras soluções de micro-mobilidade.

Observamos contudo que os frutos deste trabalho estão vinculados não só à tecnologia disponível hoje mas também à disponibilidade de equipamento com estes recursos, como por exemplo pontos de acesso compatíveis com o WPA2 e com a norma IEEE 802.11f.

6.1 Trabalhos futuros

Este estudo trouxe novas questões a serem discutidas. A primeira é a viabilidade do DHCP, tanto o DHCPv4 quanto o DHCPv6, para arquiteturas de micro-mobilidade de alta performance. A especificação do IEEE 802.11r, a qual será publicada como padrão oficial em abril de 2008, promete diminuir drasticamente o tempo de *handover* do IEEE 802.11b/g. Um estudo de como endereçar nós compatíveis com IEEE 802.11r, talvez culminando em uma alteração no DHCP convencional transformando-o em "DHCP Mobile" é uma das questões a serem analisadas.

Outro possível trabalho futuro é explorar o protocolo de segurança RADIUS para disparar eventos de camada de rede, podendo chegar ao ponto de dispensar o uso de um protocolo como o DHCP. Mesmo que sendo indesejável, a configuração manual do endereço de rede no nó móvel pode ter implicações interessantes. Como observado no capítulo 5, é necessário garantir que o endereço enviado pelo servidor RADIUS no campo Framed-IP-Address seja coerente com o endereço de rede que o nó móvel usará. Se este endereço foi pré-configurado manualmente antes mesmo do começo da associação ou configurado depois via DHCP é completamente irrelevante, desde que a notificação seja enviada com as informações corretas e que o nó móvel tenha conectividade na camada de rede. Para evitar a configuração manual, é possível imaginar um software ou ferramenta que seja capaz de se conectar em pontos de acesso com WPA e WPA2 e configurar o endereço IP a partir do valor do campo Framed-IP-Address e outros parâmetros (ex. *gateway* padrão) que podem ser carregados na mensagem de ACCESS-ACCEPT enviada pelo servidor RADIUS quando o acesso foi liberado.

Mais uma questão que pode ser discutida é se a utilização de pontos de acesso com mais recursos pode simplificar e melhorar o acesso. Existem atualmente pontos de acesso que possuem *relay agents* DHCP e servidor RADIUS embutidos, além de outros recursos. Com isso é possível usar os roteadores da rede de acesso para desempenhar outros tipos de tarefa, sempre com o objetivo de tornar o acesso Wi-Fi mais eficiente. Para pontos de acesso menos sofisticados pode-se usar o OpenWrt [29], uma distribuição Linux para sistemas embarcados que pode ser instalado nos pontos de acesso mais antigos para a instalação de um servidor DHCP ou um servidor de autenticação, por exemplo.

Finalmente, a proposta de utilização de DHTs para acelerar o processo de autenticação está sendo investigada no escopo de uma outra dissertação de mestrado.

Referências Bibliográficas

[1] Thomas, Alfred, *A story of wireless telegraphy* New York, D. Appleton and Co., 1904.

[2] Soliman, Hesham, *Mobile IPv6 - Mobility in a Wireless Internet* (Addison-Wesley, 2004).

[3] RFC 3775 - Mobility Support in IPv6 *http://www.ietf.org/rfc/rfc3775.txt*

[4] H. Soliman, C. Castelluccia, K. Malki e L. Bellier, *Hierarchical Mobile IPv6 mobility management (HMIPv6)* (Internet Draft,IETF, Dezembro de 2004).

[5] *Network-based Localized Mobility Management (netlmm) Charter - http://www.ietf.org/html.charters/netlmm-charter.html* acessado em 12/03/2008.

[6] A. T. Campbell, J. Gomez, S. Kim, A. G. Valko et al, *Design, implementation and evaluation of Cellular IP, in IEEE Personal Communications*, 2002

[7] R. ramjee, T. La Porta, S. Thuel, K. Varadhan and S.Y. Wang, *HAWAII: A domain-based Approach for Supporting Mobility in Wide-are Wireless Network*, Proc. IEEE Intl. Conf. Network Protocols, 1999.

[8] *O'Reilly Network – Wireless LAN Security: A Short History - http://www.oreillynet.com/pub/a/wireless/2002/04/19/security.html* acessado em 12/03/2008.

[9] RFC 2205 - Resource Reservation Protocol *http://www.ietf.org/rfc/rfc2205.txt* acessado em 12/03/2008.

[10] RFC 3209 - RSVP-TE: Extensions to RSVP for LSP Tunnels *http://www.ietf.org/rfc/rfc3209.txt* acessado em 12/03/2008.

[11] *Wireless Security - How WEP works - http://palisade.plynt.com/issues/2006Dec/wep-encryption* acessado em 12/03/2008.

[12] Scott R. Fluhrer, Itsik Mantin, Adi Shamir, *Weaknesses in the Key Scheduling Algorithm of RC4* (Selected Areas in Cryptography, 2001 - p. 1-24)

[13] *WPA Security Enhancements - http://www.wi-fiplanet.com/tutorials/article.php/2148721* acessado em 12/03/2008.

[14] *RFC 3748 - Extensible Authentication Protocol (EAP) - www.ietf.org/rfc/rfc3748.txt* acessado em 12/03/2008.

[15] Lehembre, Guillaume, *Wi-Fi Security - WEP, WPA and WPA2* (www.hakin9.org, 2005)

[16] Dibbler - a portable DHCPv6 *http://klub.com.pl/dhcpv6* acessado em 12/03/2008.

[17] Linux IPv6 Router Advertisement Daemon (radvd) *http://www.litech.org/radvd* acessado em 12/03/2008.

[18] dhcprelay - *DHCP Relay for 'DHCPv4 Configuration of IPSec Tunnel Mode'*

[19] Mishra, Arunesh et al. *An Empirical Analisys of the IEEE 802.11 MAC Layer Handoff Process*

[20] MGEN - The Multi-Generator Toolset *http://pf.itd.nrl.navy.mil/mgen/* acessado em 12/03/2008.

[21] DHCP-over-IPsec support for Linux FreeS/WAN *http://www.strongsec.com/freeswan/dhcprelay/index.htm* acessado em 12/03/2008.

[22] RFC 2865 - Remote Authentication Dial In User Service (RADIUS) *http://www.ietf.org/rfc/rfc2865.txt* acessado em 12/03/2008.

[23] The freeRADIUS Project *http://www.freeradius.org* acessado em 12/03/2008.

[24] Martinovic et al. *Measurement and Analysis of Handover Latencies in IEEE 802.11i Secured Networks*

[25] 802.11F - *IEEE Trial-Use Recommended Practice for Multi-Vendor Access Point Interoperability via an Inter-Access Point Protocol Across Distribution Systems Supporting IEEE 802.11 Operation*

[26] Mishra, Arunesh et al. *Context Caching using Neighbor Graphs for Fast Handoffs in a Wireless Network*

[27] Jin, Hai et al. *UINTA: A P2P Routing Algorithm Based On User's Interest And The Network Topology*, Distributed Computing - IWDC 2005, p. 238-249, Springer Berlin/Heideberg, 2005

[28] Chimera, a Structured Peer-to-Peer Overlay *http://current.cs.ucsb.edu/projects/chimera* acessado em 12/03/2008.

[29] OpenWrt - A Linux Distribution for Embedded Devices *http://openwrt.org* acessado em 12/03/2008.