Rheinisch-Westfälische Technische Hochschule Aachen

Core Network Mobility
Active MPLS

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der Rheinisch-Westfälischen Technischen Hochschule Aachen zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Diplom-Informatiker Rajendra Persaud
aus Linnich, Kreis Düren

Berichter: Univ.-Prof. Dr. rer. nat. Otto Spaniol
Univ.-Prof. Petri Mähonen, Ph.D.

Tag der mündlichen Prüfung: 15. September 2006

Diese Dissertation ist auf den Internetseiten der Hochschulbibliothek online verfügbar.
Berichte aus der Informatik

Rajendra Persaud

Core Network Mobility
Active MPLS

D 82 (Diss. RWTH Aachen)

Shaker Verlag
Aachen 2006
## Contents

1 Introduction .............................................. 1

2 Core Networks .............................................. 5
   2.1 Architecture ........................................ 5
   2.2 PPP ............................................... 6
   2.3 IPv4 ............................................. 9
      2.3.1 Header ....................................... 9
      2.3.2 Forwarding ................................... 10
      2.3.3 DHCP ......................................... 11
      2.3.4 IPv4 Tunnelling ................................ 14
   2.4 IPv6 ............................................. 15
      2.4.1 Header ....................................... 16
      2.4.2 IPv6 Neighbor Discovery ...................... 17
      2.4.3 Stateless address autoconfiguration .......... 17
      2.4.4 DHCPv6 ...................................... 18
   2.5 MPLS ............................................. 19
      2.5.1 Architecture .................................. 21
      2.5.2 Ingress Function ............................ 21
      2.5.3 LSR Function .................................. 22
      2.5.4 Egress Function .............................. 23
      2.5.5 Label Distribution ............................ 23
         2.5.5.1 Label Distribution Protocols .............. 24
         2.5.5.2 Downstream-On-Demand vs. Unsolicited Downstream .......... 24
         2.5.5.3 Independent vs. Ordered LSP Control ...... 25
   2.5.6 RSVP-TE ...................................... 25
   2.6 QoS ............................................. 29

3 Security ................................................. 33
   3.1 Security Services ................................. 33
      3.1.1 Authentication ............................... 34
      3.1.2 Confidentiality .............................. 35
      3.1.3 Integrity Protection ........................ 35
CONTENTS

3.1.4 Replay Protection .................................. 36
3.1.5 Non-repudiability .................................. 36
3.2 Key Exchange ........................................... 36
3.3 Security Protocols ..................................... 38
  3.3.1 Challenge-Handshake Protocol .................... 38
  3.3.2 One-Time Password ............................. 38
  3.3.3 Transport Layer Security ......................... 39
3.4 Extensible Authentication Protocol .................. 41
3.5 RADIUS ................................................. 43
3.6 RADIUS/EAP ............................................ 45
3.7 Diameter and Diameter/EAP ......................... 47
3.8 IPsec .................................................. 48
  3.8.1 Encapsulating Security Payload ............... 50
  3.8.2 Authentication Header .......................... 51
3.9 PANA .................................................. 52

4 Mobility Management .................................. 55
  4.1 Classification of Handover Types .................. 58
  4.2 Handover Procedure ................................ 60
  4.3 Classification of Mobility Solutions ............. 61
  4.4 IPv4-based mobility ................................ 63
    4.4.1 Mobile IPv4 .................................. 63
    4.4.2 Proxy Mobile IPv4 .......................... 67
    4.4.3 Low Latency Handoffs in Mobile IPv4 .......... 70
      4.4.3.1 Pre-Registration Handover Method .... 71
      4.4.3.2 Post-Registration Handover Method .... 73
      4.4.3.3 Combined Handover Method .......... 75
    4.4.4 HAWAII ......................................... 76
    4.4.5 Cellular IP .................................. 78
    4.4.6 GMM ........................................... 79
    4.4.7 Hierarchical IPv4-based solutions .......... 82
      4.4.7.1 TeleMIP .................................. 82
      4.4.7.2 IDMP ...................................... 84
      4.4.7.3 Micro-Mobile MIP ....................... 85
      4.4.7.4 Combinatorial MIP ...................... 85
      4.4.7.5 Dynamic Hierarchical MIP .......... 87
  4.5 IPv6-based mobility ................................ 88
    4.5.1 Mobile IPv6 .................................. 88
    4.5.2 Hierarchical Mobile IPv6 Mobility Management .... 92
    4.5.3 Fast Handovers for Mobile IPv6 .............. 94
    4.5.4 Combining HMIPv6 and FMIPv6 ............ 97
    4.5.5 Cellular Universal IP ......................... 99
    4.5.6 Edge Mobility Protocol ...................... 100
4.6 MPLS-based mobility ........................................ 103
4.6.1 Solutions only updating the Ingress LER .......... 103
4.6.2 LEMA ............................................. 105
4.7 Summary .................................................. 108

5 Active MPLS .................................................. 109
5.1 User Plane .................................................. 111
  5.1.1 Ingress LER Function ................................. 111
  5.1.2 LSR Function ........................................ 112
  5.1.3 Egress LER Function ................................ 113
5.2 Control Plane ............................................... 114
  5.2.1 LSP Set-Up ........................................... 114
  5.2.2 Set-Up of Primary LSPs .............................. 116
  5.2.3 Set-Up of Handover LSPs ............................. 116
  5.2.4 Handover Indication ................................ 117
  5.2.5 Intra-PoS HO ........................................ 118
  5.2.6 Inter-PoS HO ........................................ 118
  5.2.7 Handover Notification ............................... 119
  5.2.8 Security ............................................. 122
  5.2.9 Example .............................................. 123
5.3 Testbed Implementation ................................... 125
5.4 Simulation .................................................. 130
5.5 Evaluated Parameters ..................................... 131
  5.5.1 Outage Period ....................................... 131
  5.5.2 Loss Period ......................................... 133
  5.5.3 Handover Delay ..................................... 133
  5.5.4 Packet Loss ......................................... 134
  5.5.5 Network Utilization ................................. 134
  5.5.6 Number of User-Specific Table Entries ........... 134
  5.5.7 End-to-End Delay ................................... 135
  5.5.8 Jitter ............................................... 136
5.6 Evaluation .................................................. 136
  5.6.1 Active MPLS vs. Fast Active MPLS .................. 136
  5.6.2 Testbed Implementation .............................. 141
    5.6.2.1 Handover-related parameters .................. 142
    5.6.2.2 Packet-related parameters ..................... 145
  5.6.3 Simulation ........................................... 146
    5.6.3.1 Handover-related parameters .................. 146
    5.6.3.2 Packet-related parameters ..................... 152
  5.6.4 Summary .............................................. 155
## 6 Wireless Technologies

6.1 Wi-Fi ................................................. 157
   6.1.1 Architecture .................................. 158
   6.1.2 Protocol Stacks .................................. 159
   6.1.3 PPPoE .......................................... 160
   6.1.4 Discovery ....................................... 161
   6.1.5 Start-Up ........................................ 162
   6.1.6 Handover ........................................ 164
   6.1.7 Application of Active MPLS to Wi-Fi ............ 164
      6.1.7.1 Start-Up ................................... 164
      6.1.7.2 Handover ................................... 166

6.2 WiMAX .................................................. 167
   6.2.1 Architecture .................................... 168
   6.2.2 Protocol Stacks .................................. 169
   6.2.3 Discovery ....................................... 171
   6.2.4 Application to Active MPLS .................. 172
      6.2.4.1 Start-Up ................................... 172
      6.2.4.2 Handover ................................... 173

6.3 GPRS/UMTS .............................................. 176
   6.3.1 Architecture .................................... 177
   6.3.2 Protocol Stacks .................................. 178
   6.3.3 Discovery ....................................... 180
   6.3.4 Application to Active MPLS .................. 181
      6.3.4.1 Start-Up ................................... 181
      6.3.4.2 Handover ................................... 183

## 7 Inter-technology mobility management

7.1 Mobility Support ................................. 188
   7.1.1 Event Service .................................... 188
   7.1.2 Command Service ................................ 190
   7.1.3 Information Service .............................. 190
   7.1.4 Candidate Access Router Discovery .......... 192

7.2 IEEE 802.21 ........................................ 194

7.3 Inter-Working ........................................ 197
   7.3.1 WLAN-GPRS/UMTS Inter-Working ............. 198
      7.3.1.1 Inter-Working Via BSS Emulator .......... 198
      7.3.1.2 Inter-Working Via SGSN Emulator ....... 199
      7.3.1.3 Inter-Working Via Mobility Gateway .. 201
      7.3.1.4 Inter-Working Via Mobile IPv4 Home Agent .. 202
      7.3.1.5 Discussion .................................. 202

   7.3.2 Other Types of Inter-Working ............... 203

## 8 Conclusions and Outlook

205
<table>
<thead>
<tr>
<th>CONTENTS</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Abbreviations</td>
<td>ix</td>
</tr>
<tr>
<td>Bibliography</td>
<td>209</td>
</tr>
<tr>
<td>Curriculum Vitae</td>
<td>231</td>
</tr>
<tr>
<td></td>
<td>233</td>
</tr>
</tbody>
</table>
List of Figures

2.1 Core network architecture ........................................ 6
2.2 PPP link establishment between two PPP peers .................. 7
2.3 PPP encapsulation .................................................. 9
2.4 IPv4 header .......................................................... 10
2.5 DHCP architecture .................................................. 12
2.6 DHCP message exchange to obtain both an IP address and con-
figuration information .................................................. 12
2.7 DHCP message exchange to obtain only configuration information 14
2.8 IPv4 tunnelling methods ............................................ 15
2.9 IPv6 header ........................................................... 16
2.10 MPLS shim header ................................................... 21
2.11 FEC-to-NHLFE map (FTN) ........................................ 22
2.12 Path set-up with RSVP .............................................. 26
3.1 TLS layers ............................................................ 39
3.2 EAP architecture ..................................................... 42
3.3 RADIUS architecture ................................................ 44
3.4 Successful CHAP authentication using RADIUS ............... 45
3.5 Successful EAP authentication using RADIUS ................. 46
3.6 IPsec common deployment cases .................................... 49
3.7 ESP transport mode .................................................. 50
3.8 ESP tunnel mode ...................................................... 51
3.9 AH transport mode ................................................... 52
3.10 AH tunnel mode ....................................................... 52
3.11 PANA Architecture .................................................. 53
4.1 Peer entities in network access ...................................... 56
4.2 Classification of handover types .................................... 59
4.3 Outage period for reactive and predictive mobility solutions ... 62
4.4 MIPv4 architecture and data path in the FA CoA case ........ 64
4.5 MIPv4 Registration via FA CoA ..................................... 65
4.6 MIPv4 Registration via CCoA ....................................... 65
4.7 MIPv4 control plane protocol stack .............................. 66
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.8</td>
<td>MIPv4 user plane protocol stack in FA CoA case</td>
<td>66</td>
</tr>
<tr>
<td>4.9</td>
<td>MIPv4 user plane protocol stack in CCoA case</td>
<td>66</td>
</tr>
<tr>
<td>4.10</td>
<td>PMIPv4 start-up procedure</td>
<td>68</td>
</tr>
<tr>
<td>4.11</td>
<td>PMIPv4 control plane protocol stack</td>
<td>69</td>
</tr>
<tr>
<td>4.12</td>
<td>PMIPv4 user plane protocol stack</td>
<td>69</td>
</tr>
<tr>
<td>4.13</td>
<td>PMIPv4 handover procedure</td>
<td>70</td>
</tr>
<tr>
<td>4.14</td>
<td>Pre-Registration Handover Method of Low Latency Handoffs in Mobile IPv4 – Link-layer trigger at mobile device</td>
<td>72</td>
</tr>
<tr>
<td>4.15</td>
<td>Pre-Registration Handover Method of Low Latency Handoffs in Mobile IPv4 – Link-layer trigger at current FA</td>
<td>72</td>
</tr>
<tr>
<td>4.16</td>
<td>Pre-Registration Handover Method of Low Latency Handoffs in Mobile IPv4 – Link-layer trigger at new FA</td>
<td>73</td>
</tr>
<tr>
<td>4.17</td>
<td>Pre-registration handover method of Low Latency Handoffs in Mobile IPv4</td>
<td>73</td>
</tr>
<tr>
<td>4.18</td>
<td>Post-Registration Handover Method of Low Latency Handoffs in Mobile IPv4</td>
<td>74</td>
</tr>
<tr>
<td>4.19</td>
<td>Data path of Low Latency Handoffs in Mobile IPv4</td>
<td>75</td>
</tr>
<tr>
<td>4.20</td>
<td>Combined handover method of Low Latency Handoffs in Mobile IPv4</td>
<td>76</td>
</tr>
<tr>
<td>4.21</td>
<td>Entities involved in mobility management in the packet-switched part of cellular networks</td>
<td>80</td>
</tr>
<tr>
<td>4.22</td>
<td>Inter-SGSN Handover</td>
<td>81</td>
</tr>
<tr>
<td>4.23</td>
<td>TeleMIP Architecture</td>
<td>83</td>
</tr>
<tr>
<td>4.24</td>
<td>Combinatorial MIP Architecture</td>
<td>86</td>
</tr>
<tr>
<td>4.25</td>
<td>Dynamic Hierarchical MIP Architecture</td>
<td>87</td>
</tr>
<tr>
<td>4.26</td>
<td>MIPv6 Architecture</td>
<td>90</td>
</tr>
<tr>
<td>4.27</td>
<td>MIPv6 Start-Up procedure</td>
<td>90</td>
</tr>
<tr>
<td>4.28</td>
<td>MIPv6 Communication between CN and MN – Use of Routing Header</td>
<td>91</td>
</tr>
<tr>
<td>4.29</td>
<td>HMIPv6 Architecture</td>
<td>92</td>
</tr>
<tr>
<td>4.30</td>
<td>HMIPv6 Start-Up procedure</td>
<td>93</td>
</tr>
<tr>
<td>4.31</td>
<td>HMIPv6 handover procedure for intra-domain handovers</td>
<td>94</td>
</tr>
<tr>
<td>4.32</td>
<td>Predictive Mode of FMIPv6</td>
<td>96</td>
</tr>
<tr>
<td>4.33</td>
<td>Reactive Mode of FMIPv6</td>
<td>97</td>
</tr>
<tr>
<td>4.34</td>
<td>Handover with CUIP</td>
<td>100</td>
</tr>
<tr>
<td>4.35</td>
<td>Start-up procedure of the Edge Mobility Protocol</td>
<td>101</td>
</tr>
<tr>
<td>4.36</td>
<td>Handover procedure of the Edge Mobility Protocol</td>
<td>102</td>
</tr>
<tr>
<td>4.37</td>
<td>Combination of MPLS and MIPv4 – Start-up procedure</td>
<td>103</td>
</tr>
<tr>
<td>4.38</td>
<td>Combination of MPLS and HMIP – Architecture</td>
<td>104</td>
</tr>
<tr>
<td>4.39</td>
<td>LEMA Architecture and data path</td>
<td>105</td>
</tr>
<tr>
<td>4.40</td>
<td>LEMA start-up procedure</td>
<td>106</td>
</tr>
<tr>
<td>4.41</td>
<td>LEMA handover procedure</td>
<td>107</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

5.1 Protocol stack of the user plane ........................................ 111
5.2 Classification ........................................................................ 112
5.3 The two MPLS shim headers in Active MPLS .......................... 112
5.4 Handover ILM ..................................................................... 113
5.5 Label switching at the LSR function at an A-MPLS-capable node 113
5.6 Protocol stack of the control plane ........................................ 114
5.7 Handover FTN .................................................................... 117
5.8 Example for an A-MPLS domain ........................................... 123
5.9 HO Notification - Behavior of egress LER .............................. 125
5.10 HO Notification - Behavior of LSR ....................................... 126
5.11 HO Notification - Behavior of Ingress ................................. 127
5.12 Message sequence diagram for Ingress-Update ....................... 128
5.13 Implementation set-up ......................................................... 129
5.14 Inter-arrival times for constant bit rate and video traffic ....... 130
5.15 Message sequence diagram for Low Latency Handoffs .......... 131
5.16 Simulation set-up ............................................................... 132
5.17 Message sequence diagram for Active MPLS ....................... 137
5.18 Message sequence diagram for Fast Active MPLS ............... 138
5.19 Mean path length of Active MPLS and Fast Active MPLS during the handover in Fig. 5.13 in function of the parameter $q$ .... 142
5.20 Handover delay for constant bit rate and video traffic ........... 143
5.21 Outage period for constant bit rate and video traffic .......... 144
5.22 Absolute packet loss for constant bit rate and video traffic ... 144
5.23 Packet loss in percent for constant bit rate and video traffic ... 145
5.24 End-to-end delay for Active MPLS, Fast Active MPLS and Ingress-Update evaluated for both constant bit rate and video traffic ... 147
5.25 Jitter for Active MPLS, Fast Active MPLS and Ingress-Update evaluated for both constant bit rate and video traffic ........... 148
5.26 Handover delay for constant bit rate (left diagram) and video traffic (right diagram) .................................................. 149
5.27 Outage period for constant bit rate (left diagram) and video traffic (right diagram) .................................................. 150
5.28 Absolute packet loss for constant bit rate (left diagram) and video traffic (right diagram) .................................................. 151
5.29 Packet loss in percent for constant bit rate (left diagram) and video traffic (right diagram) .................................................. 152
5.30 End-to-end delay for Handover 2 in Active MPLS, Fast Active MPLS and Low Latency Handoffs in Mobile IPv4 evaluated for both constant bit rate and video traffic ....................... 153
5.31 End-to-end delay for Handover 2 in Active MPLS, Fast Active MPLS and Low Latency Handoffs in Mobile IPv4 evaluated for both constant bit rate and video traffic ........... 154
<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td>WiFi architecture</td>
<td>158</td>
</tr>
<tr>
<td>6.2</td>
<td>Wi-Fi protocol stack with IEEE 802.11i</td>
<td>159</td>
</tr>
<tr>
<td>6.3</td>
<td>Wi-Fi protocol stack with PPPoE</td>
<td>160</td>
</tr>
<tr>
<td>6.4</td>
<td>PPPoE discovery stage</td>
<td>161</td>
</tr>
<tr>
<td>6.5</td>
<td>Start-up with IEEE 802.11i</td>
<td>163</td>
</tr>
<tr>
<td>6.6</td>
<td>Start-up procedure</td>
<td>165</td>
</tr>
<tr>
<td>6.7</td>
<td>Handover procedure (mobile-initiated)</td>
<td>166</td>
</tr>
<tr>
<td>6.8</td>
<td>WiMAX architecture</td>
<td>169</td>
</tr>
<tr>
<td>6.9</td>
<td>WiMAX protocol stack for the control plane</td>
<td>170</td>
</tr>
<tr>
<td>6.10</td>
<td>IEEE 802.16e MAC sublayers</td>
<td>170</td>
</tr>
<tr>
<td>6.11</td>
<td>IEEE 802.16e types of MAC PDU</td>
<td>171</td>
</tr>
<tr>
<td>6.12</td>
<td>WiMAX security architecture</td>
<td>171</td>
</tr>
<tr>
<td>6.13</td>
<td>WiMAX protocol stack for the user plane</td>
<td>172</td>
</tr>
<tr>
<td>6.14</td>
<td>WiMAX start-up procedure with Active MPLS</td>
<td>173</td>
</tr>
<tr>
<td>6.15</td>
<td>WiMAX handover procedure with Active MPLS</td>
<td>175</td>
</tr>
<tr>
<td>6.16</td>
<td>GPRS/UMTS packet-switched architecture</td>
<td>178</td>
</tr>
<tr>
<td>6.17</td>
<td>GPRS protocol stack for the control plane</td>
<td>179</td>
</tr>
<tr>
<td>6.18</td>
<td>GPRS protocol stack for the user plane</td>
<td>179</td>
</tr>
<tr>
<td>6.19</td>
<td>UMTS protocol stack for the control plane</td>
<td>180</td>
</tr>
<tr>
<td>6.20</td>
<td>UMTS protocol stack for the user plane</td>
<td>180</td>
</tr>
<tr>
<td>6.21</td>
<td>GPRS/UMTS start-up procedure with Active MPLS</td>
<td>181</td>
</tr>
<tr>
<td>6.22</td>
<td>GPRS/UMTS handover procedure with Active MPLS</td>
<td>184</td>
</tr>
<tr>
<td>6.1</td>
<td>Event Service</td>
<td>189</td>
</tr>
<tr>
<td>7.2</td>
<td>Command Service</td>
<td>190</td>
</tr>
<tr>
<td>7.3</td>
<td>Possibilities for IS location</td>
<td>191</td>
</tr>
<tr>
<td>7.4</td>
<td>IS information retrieval</td>
<td>192</td>
</tr>
<tr>
<td>7.5</td>
<td>CARD architecture and operation</td>
<td>193</td>
</tr>
<tr>
<td>7.6</td>
<td>Location of the IEEE 802.21 MIHF within the mobility management protocol stack</td>
<td>195</td>
</tr>
<tr>
<td>7.7</td>
<td>Reference points of the IEEE 802.21 MIHF communication architecture</td>
<td>195</td>
</tr>
<tr>
<td>7.8</td>
<td>MIH protocol header</td>
<td>197</td>
</tr>
<tr>
<td>7.9</td>
<td>Four inter-working scenarios</td>
<td>198</td>
</tr>
<tr>
<td>7.10</td>
<td>Protocol stacks of the MN for both user and control plane</td>
<td>199</td>
</tr>
<tr>
<td>7.11</td>
<td>WLAN-GPRS integration: Protocol stack for the control plane</td>
<td>200</td>
</tr>
<tr>
<td>7.12</td>
<td>WLAN-GPRS integration: Discovery procedure</td>
<td>201</td>
</tr>
<tr>
<td>7.13</td>
<td>WLAN-GPRS integration: Protocol stack for the user plane</td>
<td>201</td>
</tr>
</tbody>
</table>
List of Tables

4.1 MIPv4 – Summary ................................ 67
4.2 PMIPv4 – Summary ................................ 70
4.3 Low Latency Handoffs in Mobile IPv4 – Summary (Combined hand-
over method) .................................. 76
4.4 HAWAII – Summary ............................. 78
4.5 Cellular IP – Summary ........................... 79
4.6 GMM – Summary .................................. 82
4.7 TeleMIP – Summary ............................... 84
4.8 Dynamic Hierarchical MIP – Summary ............... 88
4.9 HMIPv6 – Summary ............................... 94
4.10 FMIPv6 – Summary ............................... 98
4.11 EMP – Summary .................................. 102
4.12 LEMA – Summary ............................... 108
4.13 Summary ......................................... 108
5.1 Parameter $q$ for different networks ................. 141
7.1 Transport options for MIH payload ................. 196
Chapter 1

Introduction

This dissertation covers mobility in packet-switched wireless networks. It presents a new method for packet redirection during handovers. The new method is based on Multi-Protocol Label Switching (MPLS) and named “Active MPLS”. It is evaluated through an implementation on real hardware and through simulation.

Wireless network access and mobility have become the two driving forces for the development of new wireless technologies and mobile devices. Since circuit-switched networks are more and more superseded by packet-switched networks, the focus of this dissertation is on packet-switched networks. In order to access a packet-switched network in a wireless way, several wireless technologies are available. Two of these are specified by the Institute of Electrical and Electronics Engineers (IEEE). These are IEEE 802.11 and IEEE 802.16. Wireless networks based on IEEE 802.11 are also known as Wireless Local Area Networks (WLANs). Wireless networks based on IEEE 802.16 are also known as Worldwide inter-operability for Microwave Access (WiMAX) networks. Two further wireless technologies are specified by the 3rd Generation Partnership Project (3GPP). These are the General Packet Radio Service (GPRS) and the Universal Mobile Telecommunications System (UMTS). A cellular system based on GPRS is also known as 2.5G system, a cellular system based on UMTS as 3G system.

With respect to the Transport Control Protocol (TCP)/Internet Protocol (IP) reference model, mobility may involve the physical layer, the link layer, the network layer and the application layer. This dissertation covers mobility that involves the physical layer and the link layer, i.e. link-layer mobility, as well as mobility that involves the physical layer, the link layer and the network layer, i.e. network-layer mobility. The focus is on network-layer mobility.

A commonality among the packet-switched wireless networks is the subdivision into an access network part and a core network part. While the access network part generally contains link-layer devices as network nodes, the core network part generally contains network-layer devices.

A mobile device desiring to make use of a service such as web browsing, email transfer, etc. first has to attach to the packet-switched wireless network. The
CHAPTER 1. INTRODUCTION

attachment procedure is twofold. The first step consists of attaching to the first link-layer device in the access network. This link-layer device is also referred to as link-layer Point of Attachment (PoA). The second step, which is also referred to as registration, consists of attaching to the first network-layer device in the core network. This network-layer device is also referred to as network-layer PoA.

Once attached to both the access network and the core network, the mobile device may start communicating with a correspondent node. According to changing link conditions, the mobile device may desire or need to change the link-layer PoA during an ongoing data session. Such a process is referred to as handover.

In general, there is more than one target PoA to handover to. Once the target PoA is selected, the actual handover procedure implemented by a particular mobility mechanism is executed. The goal of any mobility mechanism should be to update the states of both mobile device and network in such a way as if the mobile device had attached to the target PoA without handing over from the previous PoA.

The main task of any mobility mechanism is to update the data path in both access and core network to reflect the new PoA. As the mobile device has been registered with the previous network-layer PoA, the configuration information of the mobile device stored at the previous network-layer PoA, which is also referred to as mobile device context, may be transferred to the new network-layer PoA. This process is referred to as context transfer and is a minor task of a mobility mechanism. The mobile device context may contain authentication credentials and further configuration information. If the mobile device context is not transferred, the mobile device needs to re-establish the mobile device context at the new network-layer PoA. The re-establishment of authentication credentials is also referred to as re-authentication, the re-establishment of further configuration information as re-registration.

When the mobile device changes between link-layer PoAs that both belong to the same access network, the handover is in general resolved by a link-layer mobility mechanism. If, however, the mobile device changes between link-layer PoAs that belong to different access networks, the handover generally needs to be resolved by a network-layer mobility mechanism.

Since link-layer mobility mechanisms are generally incorporated into the corresponding technology, the focus of this dissertation is on network-layer mobility mechanisms. Such network-layer mobility mechanisms may be based on the Internet Protocol (IP) or on Multi-Protocol Label Switching (MPLS). MPLS is a label-switching technology that has been conceived in order to control the paths that datagrams take in a packet-switched network. These paths are called Label Switched Paths (LSPs). With IP, this is only possible through host-specific routing table entries, an approach having bad scalability, or through IP source routing or IP tunnelling, both limited approaches incurring high overhead.

A mobility mechanism is generally deployed in the administrative area of a network provider. In the case of a network-layer mobility mechanism, the ad-
ministrative area is also referred to as domain. Mobility mechanisms are called intra-domain mobility mechanisms when dealing with handovers within a domain and inter-domain mobility mechanisms when dealing with handovers across domains. The only existing inter-domain mobility mechanism is Mobile IP. All other mobility mechanisms have been conceived for intra-domain mobility. The focus of this dissertation is on intra-domain mobility.

All existing IP-based and MPLS-based network-layer mobility mechanisms suffer from several disadvantages. These are long outage periods and packet loss during the handover, high end-to-end delays for those data packets affected by the handover, high overhead due to IP tunnelling, non-scalability due to permanent host-specific routing table entries, etc. The deficiencies of the existing network-layer mobility mechanisms have been the incentive for the development of a novel MPLS-based network-layer mobility mechanism. This mechanism is termed “Active MPLS” and presented in this dissertation.

Active MPLS improves user plane packet transmission during handovers through fast and efficient packet redirection. By equipping user plane packets with an additional user-specific label, an Active MPLS node is enabled to apply a user-specific treatment. Since the user-specific treatment is only desired during a handover, it is triggered through a handover message generated at the beginning of a handover and sent on the reverse path of the user plane packets. When triggered at a particular Active MPLS node, the user plane packets are switched onto a pre-established LSP towards the new PoA.

Active MPLS avoids all the disadvantages of the existing network-layer mobility mechanisms. In particular, in Active MPLS a mobile device does not need to acquire a new IP address during intra-domain handovers, thus eliminating the time-consuming acquisition of a new IP address as well as Duplicate Address Detection. Active MPLS does also not need to establish any paths during an intra-domain handover. Such paths may be host-specific routes, IP tunnels or LSPs. All paths in Active MPLS (i.e. LSPs) can be pre-established. Consequently, the outage period and thus packet loss are reduced with respect to other approaches. The overhead is reduced with respect to IP-based mechanisms applying IP tunnelling since Active MPLS makes use of only two MPLS shim headers of 4 Byte each. Unlike some IP-based approaches, Active MPLS is also scalable with respect to the number of permanent user-specific entries within the domain. Last but not least Active MPLS does not make use of time- and resource consuming re-classifications of IP packets during the handover, thus reducing the consumed resources within the domain as well as the end-to-end delay of user IP packets.

The outline of this dissertation is as follows. Chapter 2 provides an introduction to core networks. The typical core network architecture is presented as well as the most prevailing protocols in core networks. A special focus is set on MPLS since it is the basis for an understanding of Active MPLS.

Since security is an important topic for any wireless network, in particular for
the attachment procedure, security is covered in Chapter 3. After an introduction of the basic security services, the most common approaches for authentication, encryption, etc. are presented.

Mobility management is dealt with in Chapter 4. It is basically subdivided into mobility management mechanisms for IP version 4, IP version 6 and for MPLS. Chapter 4 gives a broad and detailed overview of all mobility mechanisms known to the author, describes them and outlines their benefits and inconveniences. At the end of Chapter 4, the reader is familiar with all different approaches of how the issue of mobility has been solved in current literature.

Active MPLS is introduced in Chapter 5. It is presented with as much detail as necessary to gain a profound understanding of how Active MPLS solves the mobility issue. The main focus is on location update and packet redirection during handovers. A special case of Active MPLS is introduced as Fast Active MPLS. In order to assess the quality of Active MPLS with respect to other mobility mechanisms, Active MPLS has been implemented both on real network processors and in a simulation environment. The main objective of the implementation on real network processors is to show the feasibility of Active MPLS and to compare it with Fast Active MPLS and another MPLS-based approach. The main objective of the implementation in a simulation environment is to compare Active MPLS and Fast Active MPLS with an IP-based approach that cannot easily be implemented on network processors due to code size limitations. A detailed performance evaluation completes the chapter.

Chapter 6 is devoted to the application of Active MPLS in the different types of existing packet-switched networks. It is shown how Active MPLS integrates into WLANs, into WiMAX networks and into cellular systems. In particular, security, addressing and configuration issues are covered for each technology.

Although the main focus of this dissertation is on intra-technology handovers, i.e. on handovers between link-layer PoAs of the same technology, Chapter 7 contains an outlook on inter-technology handovers, i.e. handovers between link-layer PoAs of different technologies. It covers the requirements imposed on the mobile device and the network in order to implement inter-technology handovers and introduces a couple of current approaches (such as IEEE 802.21) of how such handovers may be supported for network-layer mobility mechanisms.

Chapter 8 concludes this dissertation. It summarizes the main results that are in particular obtained from the implementation and the simulation described in Chapter 5 in order to show the quantitative and qualitative benefit of Active MPLS.
Chapter 2
Core Networks

A core network is a transit network with respect to the notions of Open Shortest Path First (OSPF) \[110\], i.e., it is capable of transporting data traffic that is neither locally originated nor locally destined. In wireless environments (cf. Chapter 6), the core network is directly connected to the access networks being responsible for carrying data traffic from and to the wireless end-device. The architecture of a core network is covered briefly in Section 2.1. Although many different protocols such as the Point-to-Point Protocol (PPP), Ethernet, etc., occur as link-layer protocols in packet-switched core networks, only PPP is presented in more detail in Section 2.2. The reason is that PPP will also be exploited for communication between a mobile device and the core network in Chapter 6. On the network layer, the Internet Protocol (IP) version 4 (IPv4) (cf. Section 2.3) or version 6 (IPv6) (cf. Section 2.4) or Multi-Protocol Label Switching (MPLS) (cf. Section 2.5) may be used. These protocols define how data packets are transmitted between two communicating peers on the network layer and are the underlying basis for an understanding of the different mobility management approaches presented in Chapter 4. In particular, an understanding of Chapter 5 is only possible with an understanding of MPLS as described in Section 2.5. Since Quality of Service (QoS) may be an issue in packet-switched core networks, two popular QoS architectures are described in Section 2.6 with respect to both IP-based and MPLS-based core networks. Note, however, that QoS is beyond the focus of this dissertation.

2.1 Architecture

For wireless environments, Fig. 2.1 depicts a typical example of a packet-switched core network architecture. The core network is connected to at least one access network and to at least one external Packet Data Network (PDN). It may also be connected to other external networks such as the Public Switched Telephone Network (PSTN). The core network may be based on IPv4, on IPv6 or on MPLS.
The components of the core network are correspondingly IPv4 nodes, IPv6 nodes or MPLS nodes. A node that is connected to external PDNs or to access networks is called edge node or edge router. If the edge node performs application-layer tasks as well, it is referred to as gateway. Any other node is called core node or core router. Each edge node may be connected to one or more access networks or to one or more external PDNs.

2.2 PPP

The Point-to-Point Protocol (PPP) [143] is a protocol that is often used as link-layer protocol in packet-switched core networks. It mainly provides the mapping of IP or MPLS packets to links based on Time Division Multiplexing (TDM). PPP can also be used for mobile devices to access an Edge Router without having to configure an IP address before. This is further described in Section 6.1.

A PPP connection is established between two network-layer devices that are also called peers (cf. Fig. 2.2). The PPP specification encompasses both control plane and user plane. The control plane is responsible for setting up the PPP connection between the two peers, while the user plane defines how network-layer packets are encapsulated into a PPP frame. Note that PPP control packets are encapsulated in the same way as network-layer packets. PPP imposes two requirements on the underlying layer. The first requirement is that PPP packets have to be transmitted in order since PPP does not incorporate a mechanism that places reordered packets in order again. The second requirement is that the
2.2. PPP

Link Dead

Link indication

Link Establishment Phase

LCP

Authentication Phase

Authentication protocol (e.g. EAP)

Network-Layer Protocol Phase

Network Control Protocol (e.g. IPCP)

PPP peer

PPP peer

Figure 2.2: PPP link establishment between two PPP peers

The establishment of the PPP connection is subdivided into three phases (cf. Fig. 2.2). These are the link establishment phase, the (optional) authentication phase and the network-layer protocol phase. The link establishment phase is executed by the Link Control Protocol (LCP). If an implementation desires that the peer authenticate with some specific authentication protocol (cf. Section 3.3), then it must request the use of that authentication protocol during the link establishment phase. During the network-layer protocol phase, a Network Control Protocol (NCP) is used for establishing and configuring the network-layer protocol. Beyond IP these are e.g. Internet Packet Exchange (IPX) and AppleTalk. Note that more than one NCP may be used.

PPP link establishment necessarily begins and ends in the Link Dead state. When an external event (such as carrier detection) indicates that the physical layer is ready to be used, PPP will proceed to the link establishment phase.

Each peer must first send LCP packets to configure and test the data link, i.e. each peer only establishes one direction of the link. LCP is used to establish the connection through an exchange of Configure-Request and Configure-Ack packets. Through these packets the peers agree on encapsulation format options, handle varying limits on packet sizes, provide for optional authentication of the identity of the peer, etc. This exchange is complete once a Configure-Ack packet has been both sent and received. Reaching the end of the link establishment phase triggers the transition to the authentication phase if an authentication link has to be full-duplex since PPP is based on bidirectional communication.
protocol has been negotiated, otherwise to the network-layer protocol phase.

Each Configure-Request contains zero or more Configuration Options in the
data field that are filled with any desired changes to the link defaults. Note that
all Configuration Options are always negotiated simultaneously. Unless otherwise
specified, all Configuration Options apply in a half-duplex fashion, i.e. in the
receive direction of the Configure-Request with respect to the PPP link.

The Maximum-Receive-Unit (MRU) Configuration Option indicates the max-
imum length for the PPP Information field including padding but not including
the protocol field. It defaults to 1500 octets. The Authentication-Protocol Con-
figuration Option specifies which authentication protocol (cf. also Section 3.1.1)
is to be used. One example is the Extensible Authentication Protocol (EAP)
that is explained in Section 3.4. It is acceptable to use different authentication
protocols in each direction of the PPP link. By default, the Authentication-
Protocol Configuration Option is not negotiated. Other Configuration Options
such as Protocol-Field-Compression or Address-and-Control-Field-Compression
can be found in [143].

After the link establishment phase and the authentication phase have success-
fully terminated, PPP must send NCP packets to choose and configure one or
more network-layer protocols. NCPs are defined in companion documents. The
NCP for IP is called IP Control Protocol (IPCP) and defined in [102]. The link
will remain configured for communications until explicit LCP or NCP packets
close the link down, or until some external event occurs (inactivity timer ex-
piration or administrator intervention). IPCP uses the same packet exchange
mechanism as LCP. The main Configuration Option of IPCP is the IP-Address
Configuration Option. It allows the sender of a Configure-Request to state which
IP address is desired or to request that the peer provide the information by set-
ting the value of the IP-Address Configuration Option to 0.0.0.0, which is the
default value. Once the network-layer protocol phase has terminated successfully,
corresponding network-layer packets can be sent over the PPP link.

The PPP link is closed by an exchange of LCP Terminate packets and not by
an exchange of NCP Terminate packets. The fact that one NCP has sent NCP
Terminate packets is not sufficient for the termination of the PPP link, even if
that NCP was the only established NCP.

A PPP packet consists of a Protocol field and an Information field. The PPP
Protocol field indicates the contents of the PPP Information field that carries
exactly one LCP packet or one authentication protocol packet or one IPCP packet
or one IP packet etc.

PPP encapsulation (cf. Fig. 2.3) refers to adding the protocol field (one or
two octets) and optional padding to the Information field. PPP encapsulation
requires framing to indicate the beginning and end of the encapsulation.

By default, the framing specified in [141] (similar to High-level Data Link Con-
trol (HDLC)) is used. 8 additional octets are necessary for the PPP encapsulation
when used with the default framing. Using both the Protocol-Field-Compression
and the Address-and-Control-Field-Compression Configuration Options, the PPP header can be negotiated to 4 octets. The LCP extensions [142] allow further reduction to 2 octets. However, these LCP extensions have rarely been implemented.

2.3 IPv4

The Internet Protocol version 4 (IPv4) is defined by [126]. Additional information can e.g. be found in [148]. IPv4 is the de-facto standard network layer protocol in the Internet. An IPv4 node serving only as packet source and destination is called host, while an IPv4 node also serving as intermediate node is called router. The packets sent between two hosts are called IP datagrams. One task of a router is to determine an appropriate outgoing interface and next hop for an arriving IP datagram. This task is denoted as forwarding. Another task of a router is to set up the data base with which the appropriate outgoing interface and next hop can be determined. This task is denoted as routing. The data base is denoted as routing table. The IPv4 header is described in more detail in Section 2.3.1. Section 2.3.2 covers the forwarding process. The configuration of IPv4 hosts is dealt with in Section 2.3.3.

2.3.1 Header

The IPv4 header is shown in Fig. 2.4. The Version field is set to 4 to indicate IPv4. The Internet Header Length (IHL) provides the length of the IPv4 header in 32-bit words. The minimum value is 5 which applies when no options are present. The maximum value is 15 limiting the length of the Options field to 40 octets. The Type of Service field is nowadays mainly exploited by the Differentiated Services QoS architecture (cf. Section 2.6). The shaded areas are reserved and set to 0 on transmission and ignored on reception of an IP datagram. The Total length field indicates the total length (header and payload) of the IP datagram in octets. This limits the total length of the IP datagram to 65535 octets. The Identification field contains an identification value for a datagram. All fragments of a datagram are identified by the same identification value. The Don’t Fragment (DF) field indicates that a datagram must never be fragmented. If the DF bit is not set and if a datagram is fragmented, the position of each fragment within the original IP datagram is given by the Fragment offset. In order to provide the receiving
CHAPTER 2. CORE NETWORKS

entity with the information about when the last fragment of an IP datagram is received, all fragments except the last one have the More Fragments (MF) bit set. The Time to Live (TTL) field helps to protect against temporary routing loops. The Protocol field identifies which protocol the payload of the IP datagram belongs to. This can be a standard transport protocol such as Transport Control Protocol (TCP) or User Datagram Protocol (UDP), a network layer protocol such as IPv4 or IPv6 (used when tunnelling IP datagrams) or any other protocol. The Header checksum field protects the IP header against erroneous transmissions. The Header checksum has to be recalculated whenever one of the header fields changes, in particular at each hop decrementing the TTL. The Source address and Destination address identify the IP addresses of the original sender of the IP datagram and its ultimate destination.

Several options have been specified for the Options field. In order to force IP packets to follow a certain path, the Strict Source Routing and the Loose Source Routing options have been specified. Both options specify a sequence of IP routers that the IP datagram has to visit from source to destination. The difference between the two options is that the Loose Source Routing option allows the IP datagram to visit other routers in between. Since the Options field is restricted to 40 octets, at most eight nodes can be specified in the Strict Source Routing or the Loose Source Routing fields so that the practical use has always been restricted. Since the overhead per packet is also very high, these two options have never really been implemented.

2.3.2 Forwarding

Every host and router in the Internet has a unique IP address encoding its network number and host number. Originally, IP addresses were divided into five categories or classes (classful addressing). The class an IP address belonged to was
indicated in its first bits. The length of the network and host number was dependent on the class. Therefore, forwarding was very simple. The class, network and host number of an arriving IP datagram could easily be determined. A routing table look-up in the appropriate class-based routing table then yielded outgoing interface and next hop. However, as classful addressing inherently wastes IP addresses and as the Internet grew and IP addresses became a more and more scarce resource, a new scheme has been designed. This scheme is known as Classless Inter-Domain Routing (CIDR) [65]. The main idea of CIDR is to allocate IP addresses in variable-sized blocks and not with respect to classes. In CIDR, only one routing table is used. However, a routing table entry is extended by a 32-bit subnet mask. The routing table look-up in CIDR works as follows. On receipt of an IP datagram, the Destination address is extracted from the IP header. Unless the Destination address matches one of the routers own IP addresses, a routing table look-up is performed in order to determine outgoing interface and next hop. Conceptually, the routing table is scanned entry by entry. The Destination address is masked by the subnet mask of the entry. If the result matches the network number of the entry, a match is found. Finally, the outgoing interface and the next hop of the longest such match is returned. This is commonly referred to as longest prefix matching. If no next hop is given in the routing table, the router is directly attached to the network containing the ultimate destination of the IP datagram. Eventually, the IP datagram is handed to the link layer and from there to the physical layer for delivery.

2.3.3 DHCP

The Dynamic Host Configuration Protocol (DHCP) [53] provides a framework for passing configuration information to hosts on an IP network. It encompasses delivering an IP address and host-specific configuration parameters that do not necessarily have to be related to IP from a DHCP server to a host. DHCP uses UDP as transport protocol. DHCP messages broadcast by a client prior to obtaining an IP address must have the source IP address set to 0.0.0.0. The format of DHCP messages is based on the format of BOOTP [155] messages. DHCP is also designed to use the same relay agent behavior as specified in the BOOTP protocol specification. The two primary differences to BOOTP are that a DHCP client can be assigned a network address for a finite lease and that DHCP provides the mechanism for a client to acquire all IP configuration parameters that it needs in order to operate. The basic DHCP architecture is shown in Fig. 2.5.

Each DHCP message contains a couple of header fields and further options (specified in [19]). In chapter 5, a new option will be defined. chaddr is used as hardware address for transmission of DHCP reply messages. ciaddr is the IP address of the DHCP client. It is only filled in if the DHCP client is already configured with a valid IP address. yiaddr is the IP address offered by the DHCP
server.  

\textit{giaddr} is the IP address of a possible relay agent. It is used in booting via a relay agent. It is set to 0.0.0.0 by the DHCP client in all \textit{DHCPREQUEST} messages and set by the first DHCP relay agent to the IP address of the interface on which the \textit{DHCPREQUEST} message was received and not modified by further relay agents (several are possibly configured in the relay agent). The \textit{DHCP message type} option must be included in every DHCP message. Message types are \textit{DHCPDISCOVER}, \textit{DHCPOFFER}, \textit{DHCPREQUEST}, \textit{DHCPACK} and \textit{DHCPNAK}.

The DHCP message exchange is shown in Fig. 2.6. The DHCP client broadcasts a \textit{DHCPDISCOVER} message on its local physical subnet. The message may be received and relayed by one or more DHCP relay agents. Each relay agent may itself relay the message to further relay agents or DHCP servers not on the same physical subnet.

Each DHCP server may respond with a \textit{DHCPOFFER} message that includes an available network address in the \textit{giaddr} field and other configuration parameters in the \textit{options} field. When allocating a new address, the server generally checks that the offered IP address is not already in use. Note that the DHCP server does not store any client-related state information at this point.

The DHCP client receives one or more \textit{DHCPOFFER} messages from one or more DHCP servers, chooses (based on the configuration parameters offered in the \textit{DHCPOFFER} messages) one DHCP server from which to request configuration information.
2.3. IPv4

parameters and broadcasts a DHCPREQUEST message where it sets the server identifier option (always equal to the DHCP server’s IP address) to the server identifier of the chosen DHCP server and the requested IP address option to the IP address offered in the yiaddr field of the DHCP server’s DHCPOFFER message. A server receiving a DHCPREQUEST can conclude from the server identifier field whether or not it has been selected.

The selected DHCP server sends a DHCPACK message back to the DHCP client. The DHCPACK message contains the configuration parameters for the client. In order to provide a client with a standard gateway, the router option can be used. The yiaddr field is filled with the selected network address. A DHCP client should, whenever possible, be assigned the same configuration parameters in response to each request (e.g., after DHCP client reboot). After having sent the DHCPACK message, the DHCP server creates a binding for that client. The binding associates the client identifier with the IP address allocated to the client.

The DHCP client should perform a check on the suggested IP address to ensure that the address is not already in use. If Ethernet is the underlying link-layer technology, the check is performed by broadcasting an Address Resolution Protocol (ARP) [123] request for the suggested address while using its own hardware address as the sender’s hardware address and 0.0.0.0 as the sender’s IP address. The DHCP client should broadcast an ARP reply to announce the DHCP client’s new IP address and clear any outdated ARP cache entries in hosts on the DHCP client’s subnet.

The DHCP client may extend its lease with subsequent DHCPREQUEST messages. For reusing a previously allocated network address after a reboot, the DHCP client starts directly with broadcasting a DHCPREQUEST message where it includes its network address in the requested IP address option.

If ciaddr is non-zero, the DHCP server sends DHCPOFFER and DHCPACK messages as unicast messages to the IP address contained in ciaddr. Otherwise, if giaddr is non-zero, the DHCP server sends DHCPOFFER, DHCPACK and DHCPNAK messages as unicast messages to the IP address contained in giaddr, i.e., to the DHCP relay agent closest to the client. Otherwise, the client resides on one of the server’s local subnets so that the DHCP server sends them as unicast messages to the IP address contained in giaddr and the link-layer address contained in chaddr. In practice, the contents of the chaddr field is often used to create an ARP-cache entry.

In case the IP address has already been configured by other means (e.g., through PPP), it suffices for the client to request to be provided with configuration information. The corresponding message exchange is depicted in Fig. 2.7.

The DHCP client broadcasts a DHCPINFORM message. Each DHCP server receiving a DHCPINFORM responds with a DHCPACK message containing the available configuration information.
2.3.4 IPv4 Tunnelling

Tunnelling is a concept used on the network layer. Whenever link- or network-layer information is transported (tunneled) within a network-layer packet, a tunnel is in place between those communicating peers that process the transported (tunneled) link- or network-layer information. Tunnels are made use of in different contexts. They may be used to set up a secure connection between two communicating peers as is the case in a Virtual Private Network (VPN) (cf. Chapter 3) or to set up a virtual connection between two IP nodes as is the case in the framework of some mobility management approaches (cf. Chapter 4), in particular for Mobile IP (cf. Section 4.4.1). Such a virtual connection between two IP nodes may be based on IP (IP tunnel) or on MPLS (MPLS tunnel). An MPLS tunnel is generally denoted as Label Switched Path (LSP) and described in more detail in Section 2.5. Therefore, this section is devoted to IP tunnels.

In order to cope with mobile devices, IP datagrams have to be forced onto a certain path. This can be achieved with IP source routing where the identity of the explicit route is carried with the IP packet. However, the option has rarely been implemented since the overhead imposed on every IP packet has been considered as too high.

Another means to cope with mobile devices is IP tunnelling. IP tunnelling is a means to alter the path IP datagrams take between two communicating peers. Through IP tunnelling, an IP datagram can be sent, i.e. tunneled, to an intermediate node that would otherwise not necessarily be selected by conventional IP routing. The source tunnel end point acts as encapsulator and encapsulates an IP datagram before sending the encapsulated IP datagram along the tunnel. The destination tunnel end point acts as decapsulator and decapsulates an encapsulated IP datagram sent along the tunnel. After decapsulating the IP datagram, the destination tunnel end point forwards the original IP datagram to its ultimate destination. The notion tunnel is used since the original IP datagram is not visible to the IP nodes along the tunnel.

There are several encapsulation methods available (cf. Fig. 2.8). The most common encapsulation method is called IP-in-IP encapsulation [116]. The encapsulator adds an additional IP header to the original IP datagram where the IP source address equals the encapsulator’s IP address and where the IP destination address equals the decapsulator’s IP address. The overhead results to the length
2.4. IPv6

The Internet Protocol version 6 (IPv6) is specified by [50]. The main incentive for the design of IPv6 was the imminent shortage of IPv4 addresses. This goal is achieved by providing 128-bit IPv6 addresses instead of 32-bit IPv4 addresses. IPv6 is an improvement of IPv4 in several respects. The two main improvements are the simplified IPv6 header and better support for options, both features allowing for faster processing at IPv6 routers. The longer IPv6 addresses that come at the expense of higher overhead also bring along two benefits. They
allow for structuring IPv6 addresses in a hierarchical way and enable hosts to autoconfigure their IP addresses by appending their unique link-layer addresses (e.g., a 48-bit Medium Access Control (MAC) address) to a predefined IPv6 address prefix (cf. Section 2.4.2).

2.4.1 Header

The IPv6 header is shown in Fig. 2.9. The Version field is set to 6 to indicate IPv6. The Traffic class field has been defined to distinguish different traffic classes. It is equivalent to the Type of Service field of IPv4 and will mainly be used within the Differentiated Services (DiffServ) QoS architecture (cf. Section 2.6). The Flow label field is still experimental. The objective is to define a micro-flow on the basis of the source IPv6 address, the destination IPv6 address and the Flow label. Note that an IPv4 micro-flow is defined on the basis of the source and destination IPv4 addresses, the source and destination transport-layer ports and the transport-layer protocol. The Payload length indicates the length of the payload. The IPv6 header is simpler than the IPv4 header because of the concept of IPv6 extension headers that may optionally follow the IPv6 main header. Each extension header is identified by a proper protocol identifier. This protocol identifier is placed in the Next header field. The last extension header, or the main IPv6 header if no extension headers are present, carries the protocol identifier of the higher-layer protocol (cf. Section 2.3.1). The Hop limit field has the same function as the TTL in IPv4, namely to protect against temporary routing loops. The Source address and Destination address fields have been chosen as 16-octet fields in order not to have IPv6 run out of addresses as IPv4. So far, six extension headers have been defined. In order to force an IPv6 datagram to be routed on a certain path, the Routing extension header may be used. It is similar to the Loose Source Routing option of IPv4 and exploited by Mobile IPv6 (cf. Section 4.5.1).
2.4. IPV6

2.4.2 IPv6 Neighbor Discovery

IPv6 Neighbor Discovery is defined in [111]. It mainly solves two issues. These are router discovery and address resolution. The messages are based on the Internet Control Message Protocol version 6 (ICMPv6) [43].

Router discovery corresponds to the IPv4 protocol ICMP Router Discovery [49], which is an extension to ICMP [125]. Router discovery defines two ICMPv6 packet types. These are Router Solicitation and Router Advertisement messages. Routers send out Router Advertisement messages either periodically or in response to a Router Solicitation. Router Advertisements contain a number of fields and options.

The most important fields are the M and O flags. When the M flag (Managed address configuration) is set, the receiving host uses DHCPv6 for (subsequent) address autoconfiguration. Otherwise, it uses stateless address autoconfiguration. When the O flag (Other stateful configuration) is set, the receiving host specifies its IP address through stateless address autoconfiguration and uses DHCPv6 for autoconfiguration of other information.

The most important options are the Source link-layer address and the Prefix Information. The Source link-layer address option avoids the need for subsequent address resolution for the router’s IPv6 address. The Prefix Information option specifies the prefixes that are directly reachable on the link reachable via the interface the Router Advertisement is received on. Hence, router discovery not only serves to locate routers on the link but also to acquire other configuration parameters. The advertised prefixes can, in a subsequent step, be used for stateless address autoconfiguration (cf. Section 2.4.3).

Address resolution corresponds to the IPv4 protocol ARP [123]. It defines two ICMPv6 packet types. These are Neighbor Solicitation and Neighbor Advertisement. The Neighbor Advertisement message is either sent as reply to a Neighbor Solicitation or unsolicited in order to propagate new information quickly. The most important field of a Neighbor Advertisement is the Target Address. The Target Address field is the solicited IPv6 address or the IPv6 address for which the link-layer address has changed. The only allowed option is the Target link-layer address. Neighbor Solicitation and Advertisement messages may also be used to detect node reachability and duplicate addresses.

2.4.3 Stateless address autoconfiguration

A host starts autoconfiguration of an interface by generating a link-local address for that interface. A link-local address is an IPv6 address with a scope local to the link reachable via the interface. The link-local address is generated by appending an interface identifier to the link-local IPv6 prefix FE80::. If Ethernet is the underlying link layer, the interface identifier may be chosen as the globally unique MAC address of the interface. If PPP is the underlying link layer, a locally unique
link-layer identifier may be negotiated during the PPP link establishment phase (cf. Section 2.2). For that purpose, [70] introduces a new Interface-Identifier Configuration Option. It is essential that the interface identifier is unique with respect to the IPv6 subnet. Nevertheless, Duplicate Address Detection (DAD) has to be performed for the link-local address before assigning it to the interface. Duplicate Address Detection is done by sending a Neighbor Solicitation message containing the desired link-local address in the Target Address field and by waiting for a Neighbor Advertisement. If, after a configurable number of Neighbor Solicitations have been sent, no Neighbor Advertisement is received, the host may conclude that its autoconfigured link-local address is unique. It then assigns the link-local address to the interface. This step enables IP connectivity with neighboring nodes on the link.

After configuration of the link-local address, the host still needs to configure the IPv6 address of the interface and thus tries to obtain a Router Advertisement by either sending a Router Solicitation or by waiting for the next Router Advertisement sent. If the host does not obtain a Router Advertisement, it proceeds with stateful address autoconfiguration, i.e. with DHCPv6. Otherwise, it evaluates the $M$ and $O$ flags (cf. Section 2.4.2). If the $M$ flag is set, the host obtains both address and non-address configuration via DHCPv6. If the $M$ flag is not set, the host generates the IPv6 address by appending the interface identifier to one of the advertised prefixes. The $O$ flag then specifies whether or not the host uses DHCPv6 to obtain other configuration information.

Before assigning the configured IPv6 address to the interface, it has to be tested for uniqueness. In stateless address autoconfiguration, the uniqueness of the IPv6 address is in general determined by the uniqueness of the interface identifier. If the link-local address has already been determined as unique, no additional Duplicate Address Detection needs to be performed for the configuration of the IPv6 address. If the IPv6 address is obtained manually or via DHCPv6, Duplicate Address Detection should, however, be performed. Once, the IPv6 address has been verified as unique, it can be assigned to the interface.

### 2.4.4 DHCPv6

DHCP for IPv6 (DHCPv6) is specified in [54]. It is also called stateful address autoconfiguration. DHCPv6 uses UDP as transport protocol. Each DHCPv6 message is composed of a fixed format header and a variable format part for options. To allow a DHCPv6 server to be on a different link than the DHCPv6 client, one or more DHCPv6 relay agents may be deployed relaying DHCPv6 message between client and server. At the first relay agent, the DHCPv6 message is encapsulated into another DHCPv6 message containing a Relay Message option. If the client does not need to acquire a new IPv6 address, it suffices to send an Information Request message to a defined multicast address. DHCPv6 servers respond with an Information Reply message containing the desired con-
In order to acquire both one or more IPv6 addresses and configuration information, in general four messages have to be used. The client first sends a Solicit message to a multicast address in order to locate available DHCPv6 servers. Any DHCPv6 server meeting the requirements of the client, sends an Advertise message back. The client then chooses one of the servers and sends a Request message to the chosen server. Finally, the server responds with a Reply message containing the desired IPv6 address or IPv6 addresses and the configuration information.

With respect to DHCP for IPv4, DHCPv6 offers a couple of new possibilities. DHCPv6 allows a client-initiated configuration exchange (as in DHCP for IPv4) and a server-initiated configuration exchange. The server-initiated configuration exchange is useful when links in the DHCP domain have to be renumbered. DHCPv6 clients and servers are identified by a DHCPv6 Unique Identifier. The client DHCPv6 Unique Identifier is transmitted in the Client Identifier option, the server DHCPv6 Unique Identifier in the Server Identifier option.

DHCPv6 introduces the concept of an Identity Association. An Identity Association is a concept through which a DHCPv6 client and a DHCPv6 server can identify and manage a set of IPv6 addresses. An Identity Association consists of an ID (IAID) uniquely chosen by the client and associated configuration information basically consisting of the set of IPv6 addresses. Each interface for which a client requests configuration information must be associated with at least one distinct Identity Association.

2.5 MPLS

In an IP-based domain, IP packets with different IP destination addresses may follow the same path through the domain. These IP packets can be viewed as belonging to the same forwarding class $C$. On the way of such an IP packet through the domain, each IP router along the path the IP packet takes makes an independent forwarding decision based on the IP destination address of the IP packet. The forwarding decision can be viewed as assigning the IP packet to the forwarding class $C$. This observation has been the incentive for the development of Multi-Protocol Label Switching (MPLS) [134].

MPLS is a forwarding paradigm where the classification of an IP packet to a Forwarding Equivalence Class (FEC) is done only once, i.e. when the IP packet first enters the domain. The FEC is encoded as a short value called label and sent along with the IP packet. The resulting packet is called MPLS packet. As all MPLS packets within the same FEC follow the same path, a FEC can be associated with that path. The path is called Label Switched Path (LSP). An LSP is often also referred to as LSP tunnel because the traffic sent through it is not visible to the intermediate nodes along the LSP (cf. Section 2.3.4). This is because an intermediate node only needs to process the MPLS header in order
to forward a data packet. The MPLS forwarding paradigm comes along with a couple of advantages over conventional IP forwarding.

- IP packets can be assigned arbitrarily to a FEC. This allows for traffic flows of arbitrary granularity, i.e., one LSP might be established for a single microflow carrying the traffic of a single application between two end devices while another LSP might be established for a huge traffic flow carrying the traffic between two networks.

- IP packets can easily be directed through the domain according to where they are coming from or where they enter the domain. Hence, the port an IP packet arrives on or the ingress node the IP packet is processed by can be reflected by the choice of LSP. This is not possible with conventional IP forwarding where such information is not stored with the IP packet itself.

- IP packets can be forced to take a particular path through the domain that is different from the hop-by-hop path the IP packets would take in conventional IP forwarding. This may be done as a matter of policy or to support traffic engineering. In conventional IP forwarding, IP packets can be forced to take a particular path through IP source routing, through host-specific routing table entries or through IP tunnelling. While the overhead imposed by IP source routing on every IP packet has been considered as too high, deploying host-specific routing table entries does not scale well with a large number of hosts. IP tunnelling, finally, is restricted with respect to the set of IP packets that may be placed onto the tunnel.

The label sent along with the IP packet can be encoded in different ways. It may be carried in an additional MPLS shim header [133] or in the data link header. If there is no available header field in the data link header, the additional MPLS shim header cannot be avoided. In Asynchronous Transfer Mode (ATM) [104], the Virtual Path Identifier (VPI)/Virtual Channel Identifier (VCI) of the ATM header can be exploited for the label [47]. In Frame Relay [37, 22], the Frame Relay header can be exploited as well [44]. In the following, it is assumed that an MPLS shim header has to be used.

The encoding of the MPLS shim header is specified in [133] and shown in Fig. 2.10. Note that it is allowed to place another MPLS shim header before an MPLS packet, a procedure which is referred to as label stacking. This is useful for nested LSPs. While nested LSPs are beyond the scope of this dissertation, a sort of application of label stacking is made use of in Chapter 5.

The label is a 20-bit value. Apart from sixteen reserved values, an arbitrary value may be chosen. The 3 Exp bits may be used to encode a Differentiated Services Codepoint (cf. Section 2.6). The Stack bit is set to one for the last entry in the label stack. It is set to 0 for any other entry. The MPLS TTL value is a time-to-live value used to protect against forwarding loops.
2.5. MPLS

2.5.1 Architecture

An MPLS domain consists of MPLS-capable nodes. Those MPLS-capable nodes having at least one interface physically connected to a network other than the MPLS domain are called edge MPLS-capable nodes. The other MPLS-capable nodes are called core MPLS-capable nodes.

An LSP may be set up between any pair of MPLS-capable nodes. Note that an LSP is always unidirectional. In order to establish bidirectional communication, two LSPs have to be set up, i.e. one for each direction. With respect to a certain LSP, the first MPLS-capable node on the LSP is called LSP ingress or ingress Label Edge Router (LER), a subsequent MPLS-capable node on the LSP is called Label Switching Router (LSR) and the last MPLS-capable node on the LSP is called LSP egress or egress LER. Note that these roles are associated with a certain LSP, i.e. an MPLS-capable node may have different roles on different LSPs.

The role an MPLS-capable node takes for an arriving packet is determined as follows. If an IP packet arrives that is not destined for the MPLS-capable node itself, the ingress function is invoked. If an MPLS packet arrives, the MPLS-capable node acts as LSR and invokes the LSR function covering the behaviour of both LSR and egress LER.

2.5.2 Ingress Function

The ingress function is invoked on receipt of an IP packet that is not destined for the MPLS-capable node itself. The IP packet is classified in order to determine the FEC. The classification can be based on any fields of the IP or a higher-layer

```
if (IsIP(Packet))
    then begin
        if (ForMe(Packet))
            then Receive(Packet)
        else IngressFunction(Packet);
    end
else if (packet is MPLS)
    then LSRFunction(Packet);
```

Listing 2.1: Behaviour of MPLS-capable node
protocol header. Usually, however, only the IP destination address is taken. The result of the classification is the FEC. If the FEC is not valid, i.e. it does not point to an LSP, the IP packet has to be routed using conventional IP forwarding. Otherwise, the MPLS-capable node acts as ingress LER for that packet with respect to the LSP associated with the FEC and determines the Next-Hop Label Forwarding Entry (NHLFE) from the FEC-to-NHLFE map (FTN). The NHLFE contains the interface towards the next hop, one or more labels and optionally further information that might be necessary to properly forward the packet (cf. Fig. 2.11). In the case that Ethernet is the underlying link layer, such information is e.g. the MAC destination address.

Once the NHLFE is determined, the IP packet is equipped with one or more MPLS shim headers. The fields of each MPLS shim header are set as follows. The label is set to the respective outgoing label retrieved from the NHLFE. The Exp bits are set to 0 or to a Differentiated Services Codepoint (cf. Section 2.6). The Stack bit is set to 1 for the last MPLS shim header and to 0 for all others. The TTL can be processed in two ways. It may be initialized at the ingress LER with the value of the TTL field of the IP header, be decremented at intermediate LSRs and copied back to the IP header at the egress LER. This is referred to as TTL processing for Uniform Model LSPs [15]. Alternatively, the MPLS TTL field can be initialized with an arbitrary value at the ingress LER, be decremented at intermediate LSRs and not be copied back at the egress LER. This is referred to as TTL processing for Short Pipe Model LSPs, which is assumed from now on. Finally, the MPLS packet is sent to the next hop the interface to which has been retrieved from the NHLFE. A simple example for the ingress function is given in Listing 2.2.

2.5.3 LSR Function

The LSR function is invoked on receipt of an MPLS packet. The MPLS-capable node acting as LSR looks up its Incoming Label Map (ILM) in order to retrieve the NHLFE. If the outgoing label contained in the NHLFE is equal to the EXPLICIT_NULL label, the MPLS-capable node acts as egress LER for that packet with respect to the LSP the packet arrived on and invokes the egress function (cf. Section 2.5.4).

Otherwise, the MPLS-capable node continues processing the packet in

<table>
<thead>
<tr>
<th>FEC</th>
<th>One or more labels</th>
<th>Next-Hop</th>
<th>Further information</th>
<th>NHLFE</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

Figure 2.11: FEC-to-NHLFE map (FTN)
FEC := Classification(Packet);

if (FEC is valid) then begin
    NHLFE := FTN−>Get_NHLFE(FEC);
    Packet.Add_MPLS_Shim;
    Packet.MPLS_Shim.Label := NHLFE.Label;
    Packet.MPLS_Shim.Exp := 0;
    Packet.MPLS_Shim.Stack := 1;
    Packet.MPLS_Shim.TTL := 64;
    Send(Packet, NHLFE.Next_Hop);
end
else Forward(Packet);

Listing 2.2: Ingress function (example)

the LSR function. If the label contained in the NHLFE is equal to the
IMPLICIT_NULL label, the MPLS-capable node removes the topmost MPLS
shim header. This reserved label value is used to implement Pen-ultimate Hop
Popping (PHP). The IMPLICIT_NULL label is only allowed at the pen-ultimate
hop. Since the outgoing label would otherwise be popped off the label stack
at the egress LER, there is in general no necessity to carry the label from the
pen-ultimate to the ultimate hop. If the label contained in the NHLFE is a
non-reserved label, the incoming label is replaced by the label contained in the
NHLFE and the MPLS TTL is decremented by one. Finally, the modified packet
is sent to the next hop taken from the NHLFE. A simple example for the LSR
function is given in Listing 2.3.

2.5.4 Egress Function

The egress function is invoked on receipt of an MPLS packet for which the
NHLFE retrieved from the ILM contains the EXPLICIT_NULL label as label.
The MPLS-capable node removes the topmost MPLS shim header and processes
the resulting packet again, requiring another look-up. The first look-up can be
avoided through PHP (cf. Section 2.5.3). If PHP is used, the egress LER receives
either a plain IP packet or an MPLS packet with one MPLS shim header less.

2.5.5 Label Distribution

Since it is the responsibility of each LSR to ensure that it can uniquely interpret
an incoming label, the decision as to which label is to be used between two LSRs is
up to the downstream LSR with respect to the LSP that is to be established. The
CHAPTER 2. CORE NETWORKS

NHLFE := ILM.Get_NHLFE(Packet);

if (NHLFE.Label = EXPLICIT_NULL)
    then Egress_Function(Packet)
else begin
    if (NHLFE.Label = IMPLICIT_NULL)
        then Packet.Remove_MPLS_Shim
    else begin
        Packet.MPLS_Shim.Label := NHLFE.Label;
        Packet.MPLS_Shim.TTL := Packet.MPLS_Shim.TTL - 1;
    end
    Send(Packet,NHLFE.Next_Hop);
end;

Listing 2.3: LSR function (example)

decision as to which label is to be used between two LSRs is also called binding
a label to the FEC with which the LSP, once established, will be associated.

An LSP can be viewed as a sequence of MPLS-capable nodes $R_0, R_1, \ldots, R_n$. It can also be viewed as a sequence of labels $L_1, \ldots, L_n$ where label $L_i$ is used between the MPLS-capable nodes $R_{i-1}$ and $R_i$. The LSP can be set up manually by configuration or with some label distribution protocol.

In the case that the LSP is set up manually, no particular issues with respect to label distribution occur. In the case that a label distribution protocol is used, each LSR on the LSP has to bind a label to the FEC and distribute that binding to the previous LSR on the LSP. This can be done in several ways.

2.5.5.1 Label Distribution Protocols

The Label Distribution Protocol (LDP) [20] has explicitly been developed for label distribution. However, it has neither been designed for resource reservation nor for explicit paths. Although Constraint-based Routing LDP (CR-LDP) [81] has been specified as extension for explicit paths, the IETF has decided [21] to focus on the Resource Reservation Protocol for Traffic Engineering (RSVP-TE) [24] as the basis for constraint-based LSP management. RSVP-TE has been developed as extension to the Resource Reservation Protocol (RSVP) [30] since RSVP has neither been designed for label distribution nor for the establishment of explicit paths. Both deficiencies have been taken into account by RSVP-TE.

2.5.5.2 Downstream-On-Demand vs. Unsolicited Downstream

In downstream-on-demand label distribution, an LSR may only distribute a label binding to another LSR if the latter has explicitly requested for such a label bind-
2.5. MPLS

Any unsolicited distribution of a label binding is not allowed. In unsolicited downstream label distribution, an LSR is allowed to distribute a label binding to another LSR even if the latter has not explicitly requested for it. Two label distribution peers have to agree on which label distribution mode to use.

RSVP-TE uses downstream-on-demand label distribution, while LDP allows for both modes.

2.5.5.3 Independent vs. Ordered LSP Control

In Independent LSP Control, an LSR is allowed to make an independent decision to bind a label to a FEC. In unsolicited downstream label distribution, the LSR can then directly distribute the binding to its label distribution peers, while in downstream-on-demand label distribution, it has to wait for a request. However, on receipt of the request, the LSR can directly distribute the chosen binding. Note that an LSR can only make an independent decision to bind a label to a FEC if it is aware of the FEC. This is e.g. the case if LSPs are set up according to the paths taken by conventional IP forwarding where the LSRs may learn the FECs from the routing table. If explicit LSPs are to be set up, Independent LSP Control may not be used.

For explicit LSPs, Ordered LSP Control has to be used. In Ordered LSP Control, an LSR is only allowed to choose a binding if it is the egress LER or if it already has a binding for that FEC from the next hop with respect to the LSP. If the FEC is chosen by the ingress LER, the ingress LER has to notify the egress LER of the FEC before the egress LER can start binding a label for the FEC. In Ordered LSP Control, the LSP set-up is thus typically composed of one message travelling from ingress LER to egress LER carrying the notification and another message travelling from egress LER to ingress LER carrying the respective label bindings.

RSVP-TE uses Ordered LSP Control, while LDP allows for both modes.

2.5.6 RSVP-TE


RSVP has been designed to establish paths with or without resource reservations. RSVP operates directly on top of IPv4 or IPv6 and can be considered as an Internet control protocol. RSVP is identified with a protocol ID of 46. A path is established as follows. The first node on the path, the source node, makes a decision as to where the path is to be established. Note that RSVP does not allow for the establishment of explicit paths. It generates an RSVP PATH message containing all mandatory and possibly some optional objects and sends that RSVP PATH message according to conventional IP forwarding in direction
of the destination node whose IP address is given in the IP destination address of the IP header (cf. Fig. 2.12).

Note that such an IP packet would not be intercepted by any IP router along the path to the destination node. This is desired for non-RSVP-capable nodes but not for RSVP-capable nodes. Therefore, a Router Alert option [86] has to be placed into the IP header. Routers recognizing the option should examine the IP packet more closely, e.g. by checking the IP protocol field that is equal to RSVP for RSVP messages.

Non-RSVP-capable nodes do not implement RSVP and therefore do not process the RSVP PATH message. They simply forward it in direction of the destination node. The first RSVP-capable node, however, processes the RSVP PATH message and stores it in the Path State Block. Afterwards, it forwards the RSVP PATH message in direction of the destination node. On receipt of the RSVP PATH message, the destination node processes and stores the message and generates an RSVP RESV message that it sends back to the source node. The IP address of the source node is given in the SENDER TEMPLATE object of the RSVP PATH message.

In contrast to the RSVP PATH message, the RSVP RESV message is sent to the previous RSVP-capable node whose IP address has been given in the PHOP object of the corresponding PATH message. Any non-RSVP-capable node forwards the RSVP RESV message in direction of the previous RSVP-capable node using conventional IP routing. On receipt of the RSVP RESV message, the previous RSVP-capable node processes the RSVP RESV message, stores it in the Reservation State Block and sends it again to the previous RSVP-capable node. Eventually, the RSVP RESV message arrives at the source node. The source node processes and stores the RSVP RESV message and thereby completes the path establishment.

Note that the objective of setting up a path with RSVP is to enable a special Quality of Service (QoS) for certain packets at each RSVP-capable node. The specification of the QoS and the definition of the packets are encoded in the FLOWSPEC and SENDER_TSPEC objects. Any IP packets arriving at the source node follow the same path if they are destined to the destination node. If they are destined to some other node, they may follow the same path. By using IP tunnelling they can also be forced to follow the same path. In the case
**2.5. MPLS**

that they follow the same path, they experience the QoS that has been specified during RSVP path set-up at the RSVP-capable nodes.

Since RSVP has not been designed as a label distribution protocol, the necessary extensions are incorporated in RSVP-TE. For a short discussion about the differences between RSVP and RSVP-TE see [25]. As MPLS requires labels to be distributed upstream (cf. Section 2.5.5), it is evident that the allocated labels are to be placed in the RSVP-TE RESV message. As a consequence, RSVP-TE uses downstream-on-demand label distribution and LSP Ordered Control. The request for a label binding is encoded as a new \textit{LABEL REQUEST} object, while the label binding itself, i.e. the label, is distributed in a new \textit{LABEL} object. It suffices to distribute the label only because it is inherently bound to the LSP being set up for a certain FEC. Note that an LSR being involved in setting up an LSP via RSVP-TE is in general not aware of the FEC bound to the LSP and the label the LSR allocates and distributes upstream.

RSVP-TE has not only been extended in order to support the establishment of LSPs but also to support the establishment of explicit LSPs. The establishment of explicit LSPs is achieved via a new \textit{EXPLICIT ROUTE} object.

In the following, the most important RSVP-TE objects are shortly explained.

- **\textit{LABEL REQUEST} object**
  
  In order to establish an LSP with RSVP-TE, the \textit{LABEL REQUEST} object is mandatory. It is only allowed in the RSVP-TE PATH message. The \textit{LABEL REQUEST} object is an indication for an RSVP-TE-capable node to allocate a label for the LSP and to place that label into the corresponding \textit{LABEL} object in the RSVP-TE RESV message. Although RSVP has been designed to cope with non-RSVP-capable nodes, non-RSVP-TE-capable nodes cannot allocate and distribute labels via RSVP-TE. Therefore, an LSP cannot be established with non-RSVP-TE-capable nodes in the path. As a consequence, any RSVP-TE-capable node that is aware of the next hop along the LSP being a non-RSVP-TE-capable node must not forward the RSVP-TE PATH message to that hop and instead send an RSVP-TE PathErr back to the previous RSVP-TE-capable node.

- **\textit{LABEL} object**
  
  In order to establish an LSP with RSVP-TE, the \textit{LABEL} object is mandatory. It is only allowed in the RSVP-TE RESV message. On receipt of an RSVP-TE PATH message containing a \textit{LABEL REQUEST} object, the egress LER behaves according to whether or not PHP is used. If PHP is used, it chooses the \textit{IMPLICIT NULL} label and places the \textit{IMPLICIT NULL} label into the \textit{LABEL} object of a newly created RSVP-TE RESV message that it sends to the previous RSVP-TE-capable node. Note that the egress LER does not create a NHLFE in its ILM since with PHP it does not receive MPLS packets on the LSP. If PHP is not used, the egress
LER chooses an arbitrary label and places that label into the *LABEL* object of a newly created RSVP-TE *RESV* message that it sends to the previous RSVP-TE-capable node. It also creates a *NHLFE* with an outgoing label equal to the *EXPLICIT_NULL* label and stores that *NHLFE* in the ILM at the index of the chosen label.

On receipt of an RSVP-TE *RESV* message containing a *LABEL* object, an LSR chooses a new label and places that label into the *LABEL* object of a newly created RSVP-TE *RESV* message that it sends to the previous RSVP-TE-capable node. It also creates a *NHLFE* with an outgoing label equal to the label received in the RSVP-TE *RESV* message and an outgoing interface equal to the interface it received the RSVP-TE *RESV* message on. It finally stores the *NHLFE* in the ILM at the index of the chosen label.

On receipt of an RSVP-TE *RESV* message, the ingress LER creates a *NHLFE* with an outgoing label equal to the label received in the RSVP-TE *RESV* message and an outgoing interface equal to the interface it received the RSVP-TE *RESV* message on. It stores the *NHLFE* in the FTN and associates the *NHLFE* with the FEC it triggered the LSP establishment for.

- **LSP_TUNNEL_IPv4 SESSION** or **LSP_TUNNEL_IPv6 SESSION** object

  The **LSP_TUNNEL_IPv4 SESSION** object contains the IPv4 address of the ingress LER of the LSP. The **LSP_TUNNEL_IPv6 SESSION** object contains the IPv6 address of the ingress LER of the LSP. Only one of the two objects is allowed in an RSVP-TE *PATH* or *RESV* message. Both objects also contain an LSP ID as identifier for the LSP.

- **LSP_TUNNEL_IPv4 SENDER TEMPLATE** or **LSP_TUNNEL_IPv6 SENDER TEMPLATE** object

  The **LSP_TUNNEL_IPv4 SENDER TEMPLATE** object contains the IPv4 address of the ingress LER of the LSP. The **LSP_TUNNEL_IPv6 SENDER TEMPLATE** object contains the IPv6 address of the ingress LER of the LSP. Only one of the two objects is allowed in an RSVP-TE *PATH* or *RESV* message. In order to group several LSPs (identified by a distinct LSP ID in the **SENDER TEMPLATE** object), a Tunnel ID is present in both the **LSP_TUNNEL_IPv4 SESSION** and **LSP_TUNNEL_IPv6 SESSION** object.

- **EXPLICIT_ROUTE** object

  The **EXPLICIT_ROUTE** object consists of a sequence of subobjects of variable length. Each subobject encodes an abstract node. An abstract node may be a single host, a network or an autonomous system. Each abstract node may be a loose node or a strict node with respect to the previous...
2.6 QoS

A packet-switched network generally has to transport traffic originating from various types of user applications which all have their own QoS requirements. In order to meet these heterogeneous QoS requirements in IP networks, there are generally two approaches. These are the Integrated Services (IntServ) [157] and the Differentiated Services (DiffServ) [29] QoS architectures.

IntServ is a QoS architecture that is based on the establishment of paths and on resource reservation along these paths. A path is set up with RSVP [30] and resources are reserved along that path, i.e. each hop along the path reserves resources for packets sent over the path. Note that a path set up with RSVP is not an LSP as set up with RSVP-TE. With RSVP, bandwidth is the only resource that can be reserved. There are, however, different service categories according to which the reservation is made. The main inconvenient of IntServ is the fact that resources are reserved for every single micro-flow. Hence, each hop along a node. The path chosen between an abstract node and a subsequent strict abstract node may only include nodes that belong to either one of the abstract nodes. The path chosen between an abstract node and a subsequent loose abstract node may also include other nodes. On receipt of an RSVP-TE PATH message containing an EXPLICIT_ROUTE object, the RSVP-TE-capable node has to determine the next hop on the explicit route obeying some rules specified in [24]. As the determination of the next hop is done on the basis of the abstract nodes encoded by the subobjects of the EXPLICIT_ROUTE object, the resulting path may differ from the path that would result without EXPLICIT_ROUTE object.

• TIME_VALUES object

Just as RSVP, RSVP-TE is based on soft states, i.e. PATH messages stored in the Path State Block and RESV messages stored in the Reservation State Block are removed after some specified cleanup timeout interval unless the states are refreshed. Ideally, the refresh timeout interval should be less than the cleanup timeout interval to provide for unpredictable delays and possible loss of the refresh messages. Both RSVP-TE PATH and RESV refresh messages are generated independently at an RSVP-TE-capable node and sent to the next hop along the LSP. Note that a received RSVP-TE PATH or RESV message incurring a state change triggers sending a corresponding refresh message immediately, independently of the refresh timeout interval. In order to avoid global message synchronization, the refresh timer should be randomly set to a value in the range \([0.5R, 1.5R]\) with \(R\) being the refresh timeout interval [63]. Further information on setting time-related values can be found in [30].
path has to store micro-flow-specific state information. The larger the considered packet-switched network, the more increases the number of micro-flows and the more increases the amount of micro-flow-specific state information that has to be stored at each router. IntServ is thus a QoS architecture that does not scale well with the size of the domain.

The non-scalability and the complexity of IntServ have been the main incentives for the design of DiffServ as a different QoS architecture. In DiffServ, resources are reserved for a few traffic classes instead of for micro-flows, resulting in less complexity and much better scalability.

The traffic is classified at the DiffServ edge routers of the domain. The traffic class resulting from classification is encoded as DiffServ Codepoint (DSCP) and stored in the Type of Service field of an IPv4 header or the Traffic class field of an IPv6 header. While travelling through the domain, each hop does not have to re-classify the IP packet in order to determine its traffic class. In contrast to the edge routers, the core routers can thus be kept simple.

The treatment of an IP packet at a core router with respect to its DSCP is called Per-Hop Behavior (PHB). Standardized PHBs are the Expedited Forwarding (EF) PHB [80] and the Assured Forwarding (AF) PHBs [72]. In general, the DSCP translates into a certain queue the IP packet is placed into. The scheduler at the outgoing interface is then responsible for meeting the desired QoS of the corresponding traffic class. As DiffServ operates on the network layer, the scheduled entities are full packets. On narrow-band links, it may be beneficial to apply link-layer fragmentation in order to improve the PHB for certain traffic classes. An analysis of link-layer fragmentation has been performed by [119].

In MPLS networks, both QoS architectures can be used as well. When an LSP is set up using RSVP-TE with bandwidth reservations, inherently IntServ is used. Note that the scalability of using IntServ with MPLS is better than using IntServ alone. The reason is that resources are reserved per LSP (being capable of aggregating many micro-flows) and not per micro-flow. However, the scalability of DiffServ is still better. DiffServ support with MPLS is described in [61]. Furthermore, the complexity of deploying IntServ is very high compared to the one of deploying DiffServ. In general, DiffServ is thus used in MPLS networks.

There are two different possibilities to indicate the PHB depending on how many QoS or traffic classes are required. If at most eight traffic classes are needed, the Exp bits of the MPLS shim header can be exploited. However, since there are only three Exp bits, the 6-bit DSCP has to be mapped to the Exp bits. This mapping is either manually configured at each hop along the LSP or it is signalled during LSP set-up. The main advantage of such E-LSPs is that the traffic class is encoded in the MPLS packets themselves so that MPLS packets of different traffic classes may be sent over the same E-LSP.

If, however, more than eight traffic classes have to be supported, E-LSPs can no longer be used. In that case, so-called L-LSPs have to be used. On L-LSPs,
the traffic class of the packets sent over the L-LSP is either manually configured in each hop along the L-LSP or signalled during L-LSP set-up. Since the MPLS packets do not carry an indication of the traffic class with them, only one traffic class can be supported on one L-LSP. Nevertheless, the Exp bits may be exploited in order to distinguish between different drop precedences that are e.g. used in the AF traffic classes.
Chapter 3

Security

Authentication, confidentiality, integrity protection and protection against replay attacks are security services that require cryptographic mechanisms. Symmetric cryptography refers to all systems where the sending entity uses the same (shared) key as the receiving entity. Asymmetric cryptography refers to all systems where sending and receiving entity are each in possession of a public key that can be disclosed to anybody and a mathematically associated private key that must be kept private. Therefore, asymmetric cryptography is often also called public-key cryptography. The most widespread asymmetric system is the one of Rivest, Adleman and Shamir known as RSA [132]. Other asymmetric systems are the ones of Rabin and ElGamal. A good introduction to cryptography is given by [103]. Security is necessary in any communication system where data, information or communication services are to be protected from eavesdroppers or attackers. In wireless environments where access to data, information or communication services cannot be prevented physically, additional security measures have to be taken. In wireless environments with mobile devices, security becomes even more important since an attacker may mount an attack from any point of attachment to the network. Therefore, security is essential in mobile wireless environments and needs to be treated together with mobility management.

3.1 Security Services

Security services provide a certain level of security for a set of communication peers against possible passive and active attacks. A passive attack is an attack through passively monitoring the data stream to gain knowledge of secret information or to gather knowledge for a later active attack. An active attack attempts to modify the data stream by deleting, adding or modifying one or more of its messages. A replay attack is a special form of an active attack where a message is passively recorded and actively replayed at a later point in time. The objective of an attacker is to cause harm to either entity of the communication
peers without revealing its own identity. It can do so by authenticating on behalf of some other peer, by modifying the origin or the contents of a message sent between two peers, by spoofing the contents of a message, etc. In function of the level of security that is to be achieved, different security services have to be applied. The most important security services are authentication, confidentiality, integrity protection, replay protection and non-repudiability. In mobile wireless environments, some or all of these security services are provided by the different wireless technologies (cf. Chapter 6).

3.1.1 Authentication

Authentication is a security service that applies both to a user, to a device and to a message. When a user communicates with an entity, user authentication may be required if unauthorized use of the entity must be prohibited or if a user-specific treatment is desired by either the user or the entity. In order to prove his identity, the user has to supply the entity with some credentials such as a passport, a Personal Identification Number (PIN), a user name and password, etc.

Device authentication is similar to user authentication. The difference is that not a user but a device has to supply the entity with these credentials. Usually, such credentials are device-specific certificates that are stored in a Read-Only Memory (ROM) on the device.

Message authentication applies to a message sent by a device and is subdivided into message origin authentication and message content authentication. The latter is also called integrity protection.

User authentication can be done via password-based methods or via challenge-response methods. In general, the user does not have to interact personally in the authentication process, e.g. if the user credentials are stored on the user’s device. The entities taking part in the authentication process are called authenticator and supplicant. In a password-based method, the authenticator requires the supplicant to supply a password or some other secret information. A password-based method is vulnerable to replay attacks as long as the password remains the same. Furthermore, the password needs to be transmitted over a secure channel and is inherently exposed to others.

In a challenge-response method, the authenticator sends a random number to the supplicant. If a shared symmetric key is used between supplicant and authenticator, the supplicant may either encrypt the random number or apply a keyed one-way function to the random number and send the result back to the authenticator. By applying the same encryption algorithm or the same keyed one-way function, the authenticator can verify the identity of the supplicant. Replay attacks can be avoided as long as different random numbers are chosen. Therefore, care should be taken when the random number space is limited. A challenge-response method is also vulnerable if the random numbers are chosen
3.1. SECURITY SERVICES

in a predictable way because an attacker may predict the random number for the
subsequent authentication, send the predicted random number to the supplicant
and trick the supplicant into replying with the appropriate result. The attacker
may then re-use the obtained result for authenticating on behalf of the tricked
supplicant.

In a public-key environment with public and private keys, the supplicant may
provide a digital signature of the received random number and send it back along
with the corresponding public key and a chain of digital certificates associating
the public key with the identity of the supplicant. By applying the public key,
the authenticator can verify the identity of the supplicant. Note that sending a
public key along with a chain of digital certificates is sufficient for authentication.
The encryption of a random number proves that the supplicant replies to the
previously sent message. Hence, this authentication method is certainly the most
secure with respect to password-based methods and among the challenge-response
methods. However, it is also the most time-consuming one.

Finally note that authentication is often used to derive keys if these are not
pre-established. Authentication is thus often used as the first method to gain
access to a network before any other security service is applied.

3.1.2 Confidentiality

Confidentiality is a security service applicable to a message. It is generally
achieved through encryption. In a shared-key environment where a shared key
is established between message sender and message receiver, the message can be
encrypted and decrypted with the same shared key. In a public-key environment,
the message is encrypted with the public key of the message receiver. The mes-
sage receiver is then the only entity capable of decrypting the message since it
is in possession of the corresponding private key. Note that public-key opera-
tions are usually time-consuming operations so that public-key encryption of a
message is usually avoided. Instead, the public-key infrastructure is exploited to
derive shared session keys between the message sender and receiver and to use
shared-key encryption for the session.

3.1.3 Integrity Protection

Integrity protection is a security service applicable to a message. A message
is integrity-protected if any hazardous or malicious change of the message is
recognized by the message receiver. Integrity protection is also called message
content authentication.

Integrity protection is generally achieved by calculating a fingerprint or mes-
sage digest of the message and sending the fingerprint or message digest along
with the message. The two most common algorithms are the Message Digest 5
(MD5) [131] algorithm and the Secure Hash Algorithm 1 (SHA-1) [55].
CHAPTER 3. SECURITY

MD5 processes a message of arbitrary length and returns a fingerprint or message digest of 128 bits. SHA-1 processes a message with a length smaller than $2^{64}$ bits and returns a message digest of 160 bits.

It is generally believed that it is computationally infeasible to create a different message having the same message digest. Any change to a message will therefore, with very high probability, result in a different message digest.

The above-mentioned message digest basically prevents erroneous message transmission from passing the receiver unrecognized, but does not guarantee the integrity since an attacker could modify the message content and re-compute the message digest without issues. In order to fully protect the integrity of a message against any attacks, additional measures have to be taken. It must be ensured that the message or the message digest is intrinsically tied to the sender of the message.

This can be achieved through including a shared secret in the calculation of the message digest or through a digital signature. Since digital signatures require time-consuming public-key operations, they are usually applied to the message digest instead of to the message itself. An attacker could still alter the message content and also re-compute the message digest. However, in order to re-compute the message digest, the attacker would have to apply its own private key and provide its own public key along with the associated certificates, thus revealing its own identity.

3.1.4 Replay Protection

Replay protection is a security service applicable to a message and generally achieved by adding some value that is unique per message. If the value is required to be unique over time, a time-based value should be chosen. Otherwise, a sequence number achieving short-term uniqueness can be chosen as well. In order to protect the value from being altered by an attacker, it has to be integrity-protected (cf. Section 3.1.3).

3.1.5 Non-repudiability

Non-repudiability is a security service applicable to a message and designates that the sender of a message cannot later on claim never to have sent that message. Non-repudiability is generally achieved through message origin authentication.

3.2 Key Exchange

In order to apply the security services described in Section 3.1, the communicating peers have to be in possession of some key material. In shared-key environments, they need a shared symmetric key. In public-key environments, the public keys
3.2. KEY EXCHANGE

are publicly available and do not need to be negotiated. Note that public-key operations are much more time-consuming than shared-key operations. Therefore, shared keys are generally also exchanged and made use of in public-key environments. The advantage of public-key environments is that a secure channel is provided through the public keys, i.e. the shared keys can easily be exchanged.

In a shared-key environment, the key exchange may be done in several ways. In general, an out-of-band communication between the communicating peers is not possible and the key exchange has to be done over an insecure channel.

One of the most common protocols for key exchange is the Diffie-Hellman protocol [52]. This original version of the Diffie-Hellman protocol does not provide authentication of the communicating peers. It is therefore vulnerable to man-in-the-middle attacks since the man in the middle may establish two distinct Diffie-Hellman keys with either communicating peer and masquerade as one peer when communicating with the other. In order to protect against such man-in-the-middle attacks, authentication is needed between the two communicating peers. Generally, digital signatures are used to authenticate the key exchange.

In the Internet, the Internet Key Exchange (IKE) protocol [68] is the de-facto standard for authenticated key exchange. IKE operates in two phases. In the first phase, a secure authenticated channel is established between the two communicating peers. The resulting security association defining a set of security parameters is based on the Internet Security Association and Key Management Protocol (ISAKMP) [100]. Such security parameters are an encryption algorithm, a hash algorithm, an authentication method, etc. In the second phase, further Security Associations are established based on the ISAKMP security association negotiated in the first phase. The Security Associations established in the second phase are negotiated on behalf of services such as IP Security (IPsec) (cf. Section 3.8).

For better mobility support, IKEv2 Mobility and Multihoming (MOBIKE) [88] specifies some extensions to IKE. The Security Associations between two communicating peers are bound to the IP addresses of the peers. If, e.g. due to mobility, one of these IP addresses changes, the Security Associations would have to be re-established if only IKE was in use. MOBIKE, however, provides a protocol exchange to update the IP addresses bound to the Security Associations so that the time-consuming re-establishment can be avoided. The protocol exchange is called MOBIKE Address Update.

In WLAN and WiMAX environments (cf. Section 6.1 and Section 6.2), different authentication and key exchange mechanisms between the mobile device and the wireless packet-switched network are possible, whereas cellular systems are generally shared key environments with a pre-shared key that does not need to be negotiated during authentication (cf. Section 6.3). In order to secure data transfer between two nodes within the wireless packet-switched network, the key exchange based on IKE may be used instead of pre-shared keys.
3.3 Security Protocols

In the next subsections, three common security protocols are presented. These are the Challenge-Handshake Protocol (CHAP) (cf. Section 3.3.1), the One-Time Password (OTP) system (cf. Section 3.3.2) and Transport Layer Security (TLS) (cf. Section 3.3.3). CHAP may be used on PPP links as authentication protocol during the PPP authentication phase (cf. Section 2.2). The OTP system is generally deployed in a modified version in the field of e-commerce where transaction numbers are generally used to authorize the execution of commercial transactions. TLS is widely deployed in the Internet in order to establish secure connections between two communicating peers. In mobile wireless environments, TLS is also used in networks based on WiMAX (cf. Section 6.2). In order to deploy it on PPP links, the Extensible Authentication Protocol (EAP) (cf. Section 3.4) needs to be deployed instead of CHAP.

3.3.1 Challenge-Handshake Protocol

The Challenge-Handshake Protocol (CHAP) is a challenge-response authentication protocol based on PPP (cf. Section 2.2) and an improvement of the Password Authentication Protocol (PAP) where a user password (i.e. a shared secret key) is sent as plain text in response to the challenge message. Each CHAP message is encapsulated in the Information field of a PPP frame whose Protocol field indicates CHAP. In order to negotiate CHAP as authentication protocol, the Authentication-Protocol Configuration Option with CHAP as the desired authentication protocol has to be negotiated during the PPP Link Establishment phase. Once the PPP Link Establishment phase has terminated successfully, the authenticator (i.e. the authenticating communication peer) sends a CHAP Challenge message to the supplicant. The CHAP Challenge message contains a random value of arbitrary length. The supplicant responds with a CHAP Response message containing a value calculated as one-way hash over the CHAP Challenge message and a shared secret key. On receipt of the CHAP Response, the authenticator verifies the response value by calculating the same one-way hash. If both hashes match, the authenticator sends a CHAP Success message, otherwise a CHAP Failure message. An extension to CHAP is Microsoft PPP CHAP version 2 (MS-CHAPv2) [164]. The main improvement of MS-CHAPv2 over CHAP is that MS-CHAPv2 provides for mutual authentication.

3.3.2 One-Time Password

A One-Time Password (OTP) [67] system is a challenge-response authentication system and based on [93]. It may be used in environments where users authenticate by providing user ID and password. An OTP system protects against replay attacks when authenticating a user.
By applying a secure hash function multiple times on the concatenation of a secret pass-phrase and a random number, the user generates a sequence of one-time passwords. The last one-time password of that sequence has to be stored in the authentication server. When the user authenticates for the first time, it sends the previous one-time password within the sequence. By applying the secure hash function on this one-time password, the server verifies if the result equals the password stored in its database. If the authentication is successful, the password stored in the database is replaced by the received one. When the user authenticates for the next time, it sends the one-time password within the sequence preceding the one last sent. Hence, the authentication of the user is based on a shared symmetric key, which is the newly initialized password or the one-time password from the last successful authentication.

### 3.3.3 Transport Layer Security

Transport Layer Security (TLS) [51] is a security protocol providing authentication, confidentiality, data integrity and protection against replay attacks between two communicating applications. TLS is based on the Secure Socket Layer (SSL).

TLS consists of two layers (cf. Fig. 3.1). The TLS Record Protocol is on the lower layer. On the higher layer, there are three protocols on the control plane and one protocol on the user plane. On the control plane these are the TLS Handshake Protocol, the TLS Alert Protocol and the TLS Change Cipher Spec Protocol. The protocol on the user plane is the Application Data Protocol.

The TLS Record Protocol lies on top of some reliable transport protocol (e.g. TCP). The transport protocol has to be reliable in order to avoid reordering when fragmentation is deployed. The TLS Record Protocol provides (optional) confidentiality by using symmetric cryptography for data encryption and (optional) integrity by using a keyed Message Authentication Code (MAC). The shared symmetric keys are generated uniquely for each connection and are based on a master secret negotiated by another protocol such as the TLS Handshake Protocol.

The TLS Record Protocol takes the messages from the higher-layer protocols, fragments or coalesces (for messages of the same higher-layer protocol) the data into manageable blocks, optionally compresses the data, applies a MAC (before
encryption), encrypts the record including the MAC and finally transmits the result.

The TLS Handshake Protocol is responsible for (optional) server and (optional) client authentication and negotiation of compression, hash and encryption algorithms and a master secret. The TLS Record Layer uses these security parameters to generate a MAC secret, an encryption key and, for block ciphers, an initialization vector.

The authentication is done by asymmetric, i.e. public-key cryptography. Three authentication modes are supported. These are authentication of both parties, server authentication with an unauthenticated client, and total anonymity. Whenever the server is authenticated, the connection is secure against man-in-the-middle attacks. Only completely anonymous sessions are inherently vulnerable to such attacks.

TLS is based on a connection state. Logically, there are always four states. These are the current and pending read and write states. The TLS Handshake Protocol sets security parameters for the pending states. Once the security parameters are set and the master secret is generated, the current write state is replaced by the pending write state and a Change Cipher Spec message is sent to the peer as instruction to replace the current read state by the pending read state.

Apart from authentication one goal of the TLS Handshake Protocol is to create the above-mentioned master secret known to the communicating peers and not to attackers. The master secret is itself created from a negotiated pre-master secret. By sending a correct Finished message, which is lastly based on that pre-master secret, either peer proves that it knows the correct pre-master secret.

The key exchange can be done via Diffie-Hellman or via public-key exchange, i.e. with RSA. With RSA, key exchange and server authentication are combined. The server public key may be contained in the server’s certificate (which could be sent by anybody). After verifying the certificate, the client encrypts a random pre-master secret with the server’s public key. By successfully decoding the pre-master secret and producing a correct Finished message, the server demonstrates that it knows the private key corresponding to the server’s certificate and thus authenticates itself.

From the pre-master secret, both communicating peers derive a master secret. The client authenticates itself by sending a message containing a signed (with the client’s private key) value derived from the master secret and all preceding handshake messages binding that message to the server (as one of the preceding handshake messages contains the server certificate) and to the current handshake process (as one of the preceding handshake messages contains a random value chosen by the server).

On the user plane, packets are received by TCP and handed to the application bound to the destination port (e.g. 443 for https) carried by the TCP segment.
3.4 Extensible Authentication Protocol

The existence of a multitude of different authentication methods has been the incentive for the design of the Extensible Authentication Protocol (EAP) [10]. EAP is an authentication framework that supports multiple authentication methods each identified by a specific type. Typically, it runs directly over data link layers such as PPP (cf. Section 2.2) or IEEE-802-based link layers. The necessary extensions to IEEE 802.3 are contained in [76], the ones to IEEE 802.11 in [78].

EAP is reliable in the sense that it provides support for duplicate elimination and retransmission. Since EAP is a lock-step protocol only supporting a single packet in flight, duplication can be detected with the Identifier field contained in each EAP packet.

Note that EAP does not prevent packet reordering though. Packet reordering should be avoided by the lower layer. The reason is that the EAP packet identifier need not to be monotonically increasing as EAP has originally been designed for PPP.

Fragmentation is not supported and may be done in individual EAP methods such as EAP-TLS or in the lower layer. If the EAP packet size exceeds the EAP Maximum Transmission Unit (MTU) of the link, the EAP method will encounter difficulties. The EAP peer typically obtains the EAP MTU from the lower layer, while the backend authentication server obtains it from the authenticator (e.g. via the Framed-MTU attribute in RADIUS (cf. Section 3.5)).

EAP is based on a two- or three-party model (cf. Fig. 3.2). The authenticator is the end of the link initiating the EAP authentication. The peer is the end of the link that responds to the authenticator. The EAP server is the entity that terminates the EAP authentication method with the peer. In the case where no backend authentication server is used, the EAP server is part of the authenticator. In the case where the authenticator operates in pass-through mode, the EAP server is located in the backend authentication server. The backend authentication server is also called Authentication, Authorization and Accounting (AAA) server. The communication between the EAP authenticator and the AAA server is done via AAA protocols with EAP support such as RADIUS (cf. Section 3.5) and Diameter (cf. Section 3.7). AAA protocols such as RADIUS and Diameter only support an EAP authenticator operating in pass-through mode.

An EAP-based authentication is a challenge-response authentication consisting of a sequence of EAP Request and Response messages. The EAP Request messages are sent from the EAP server to the peer, while the EAP Response messages are sent from the peer to the EAP server. If the EAP server is located in a backend authentication server, the EAP Request and Response messages
are relayed by the authenticator. The first EAP Request message, however, may be sent directly by the authenticator. The number of necessary message exchanges depends on the authentication method in use. Based on the message type, incoming EAP packets are handed to the corresponding EAP method. If the authentication terminates successfully, the EAP server sends an EAP Success message, otherwise, an EAP Failure message. Note that the authentication occurs between peer and EAP server. If the EAP server does not reside on the same machine as the authenticator, the identity of the authenticator, not taking part in the authentication, cannot be verified.

It is necessary for the lower layer to provide authentication, data integrity, and replay protection in order to provide assurance that the entities transmitting data are the same ones that successfully completed EAP authentication. These per-packet services have to be bound to the keys derived during EAP authentication. In the case that the EAP server is located in a backend authentication server, these keys have to be transmitted from the backend authentication server to the authenticator. The key transfer can be achieved with RADIUS. In the following, some of the existing authentication methods are described briefly.

EAP-OTP is not specified in a separate document. It is based on [67] (cf. Section 3.3.2) and consists of an OTP challenge and its response message, respectively encapsulated in EAP Request and Response messages.

A Generic Token Card (GTC) method can be used as EAP-GTC. It is not specified in a separate document. The GTC challenge contains a displayable
3.5. RADIUS

Remote Access Dial In User Service (RADIUS) is a protocol for carrying authentication, authorization and configuration information. It runs over UDP with exactly one RADIUS packet being encapsulated in the UDP data field (destination port 1812) and is deployed between a RADIUS client and a RADIUS server. The client is called Network Access Server (NAS). The entity that is being authenticated, authorized and accounted for is a user either directly or remotely.
(e.g. via PPP) connected to the NAS. The NAS provides one or more services to the dial-in user (e.g. PPP). In larger networks, in particular in roaming scenarios, one or more proxy RADIUS servers are commonly used. A RADIUS server has a database of users containing user name and password as well as configuration information detailing the type of service to deliver to the user.

Transactions between NAS and RADIUS server or between two RADIUS servers are authenticated through the use of a shared secret, which is never sent over the network. There is a shared secret for each communicating peer.

RADIUS is a hop-by-hop protocol, i.e. a RADIUS message travels from NAS to zero or more RADIUS proxies to the RADIUS server. At each hop the source IP address, the source UDP port and the destination IP address are changed when forwarding the message. Using the source IP address of the RADIUS packet, a RADIUS server can decide which of its shared secrets to use.

Initially, the NAS sends an Access-Request to the RADIUS server (cf. Fig. 3.4). The Access-Request basically contains a Request Authenticator and a number of attributes such as User-Name, User-Password, NAS-Identifier or NAS-IP-Address, the NAS-Port the user is accessing etc. User authentication is done using the User-Password or CHAP-Password attributes. Any user password is sent encrypted between the NAS and the RADIUS server (using MD5). If the user password is provided to the NAS by a user dialog or through PPP PAP (cf. Section 3.3.1) authentication, it is sent encrypted in the User-Password attribute. If the user password is provided to the NAS through PPP CHAP authentication, at which it is already encrypted, it is sent in the CHAP-Password attribute. If CHAP is used for authentication, the CHAP challenge is provided in the Request Authenticator or the CHAP-Challenge attribute. Note that the Access-Request has originally not been integrity-protected. Integrity protection can be added through the Message-Authenticator attribute [11].

On receipt of the Access-Request, the RADIUS server validates the sending NAS, consults its database of users to find the user and verifies some requirements (password match, NAS match, port match, etc.) to allow access for the user. If any requirement is not met, the RADIUS server sends an Access-Reject, otherwise an Access-Challenge or an Access-Accept. Each message is integrity-protected
3.6. RADIUS/EAP

In environments where the EAP authenticator operates in pass-through mode, the EAP messages have to be relayed by the EAP authenticator to a backend Authentication Server (AS). RADIUS (cf. Section 3.5) is an appropriate protocol for the transport of EAP messages. Nevertheless, some extensions cannot be avoided. These are specified in [11].

Note that if RADIUS and EAP are deployed together, the EAP authenticator and the NAS designate the same node. The same holds for the RADIUS server and the EAP server. The main achievement of RADIUS/EAP is the encapsulation of EAP messages into RADIUS messages and the transport of these messages between NAS and RADIUS server.

RADIUS/EAP defines two new RADIUS attributes. These are the EAP-Message and the Message-Authenticator attributes. Any EAP packets sent between NAS and RADIUS server are encapsulated into the EAP-Message at-
The Message-Authenticator attribute serves for integrity protection of the RADIUS message and is mandatory for any RADIUS message containing an EAP-Message attribute. Although Access-Challenge, Access-Accept and Access-Reject messages contain the Response Authenticator field, the Message-Authenticator attribute is mandatory for these messages, too, since the Response Authenticator field does not cover the EAP-Message attribute.

The EAP peer and the NAS begin the EAP conversation by negotiating the use of EAP (cf. Fig. 3.5). Once EAP has been negotiated, the NAS should send an initial EAP-Request to the EAP peer. If the NAS sends an EAP-Request/Identity, the peer responds with an EAP-Response/Identity. If the peer is local, the NAS may proceed with local authentication. Otherwise, it encapsulates the EAP-Response/Identity within an EAP-Message attribute, places it into a RADIUS Access-Request message and sends the RADIUS message to the RADIUS server.

Note that since the NAS only encapsulates the EAP-Response in its initial Access-Request, the initial EAP-Request is not available to the RADIUS server. Relevant information in the EAP-Request thus has to be reflected in the initial EAP-Response.

The RADIUS server must respond with an Access-Challenge message containing one or more EAP-Message attributes. An EAP-Message attribute encapsulates a single (and not multiple) EAP messages which the NAS decapsulates and passes on to the authenticating peer. Note that it is possible for the RADIUS server to encapsulate an EAP message that is larger than the MTU on the link between the NAS and the peer. Therefore, the NAS may provide the RADIUS server with this information by including the Framed-MTU attribute in an Access-Request message.
3.7 DIAMETER AND DIAMETER/EAP

The RADIUS server has to be able to distinguish between EAP packets with the same value of the Identifier field that are sent as part of distinct sessions, however originating on the same NAS. Sessions can be distinguished with respect to the NAS by looking at the NAS-Identifier, NAS-IP-Address, etc. and with respect to session identification attributes such as User-Name, NAS-Port (in its sense of a physical connection on the NAS, not in the sense of a TCP or UDP port number), etc.

Note that mutual authentication occurs between the peer and the RADIUS server, not between the peer and the EAP authenticator. This means that it is not possible for the peer to validate the identity of the NAS that it is speaking to, using EAP alone.

On receipt of the Access-Challenge, the NAS decapsulates the EAP Request message and sends it to the EAP peer. The EAP peer passes the EAP Request message to the EAP method indicated in the Type field of the EAP message. The EAP method creates the appropriate response that is sent to the NAS in terms of an EAP Response message. The NAS encapsulates the EAP Response message into a RADIUS Access-Request message that it sends to the RADIUS server.

The EAP and RADIUS message exchange continues until the RADIUS server decides to accept or to reject access for the user. In Fig. 3.5, the RADIUS server accepts access and sends a RADIUS Access-Accept message to the NAS. The NAS finally decapsulates the EAP Success message and sends it to the user.

Note that the EAP Master Session Key (MSK) negotiated between the peer and the RADIUS Server will need to be transmitted to the EAP authenticator, i.e. to the NAS. Therefore a mechanism needs to be provided to transmit the MSK from the RADIUS Server to the NAS. For this purpose, the MS-MPPE-Send-Key and the MS-MPPE-Recv-Key attributes [163] may be exploited.

Since EAP supports any authentication method and since RADIUS supports the transport of EAP messages, RADIUS and EAP are ideally deployed together. While they are not used in cellular systems, they are suited for WLANs (cf. Section 6.1) and networks based on WiMAX (cf. Section 6.2). The current WiMAX draft e.g. specifies the use of RADIUS (as one option) and EAP.

3.7 Diameter and Diameter/EAP

AAA protocols such as RADIUS have initially been designed to support user access remotely via PPP or locally via direct interaction between user and NAS. With the growth of the Internet and the emergence of new access technologies, the requirements of a NAS have increased considerably [12]. This has motivated the design of Diameter as a new AAA protocol supporting on the one hand RADIUS-based systems and meeting on the other hand the increased requirements of AAA protocols.

Diameter is specified by a set of documents. The base protocol is specified...
in [31]. NAS-related issues are specified in [33]. The interworking with EAP is specified in [57].

With respect to RADIUS, Diameter is a complete re-design and as such offers a couple of new functionalities. The most important new functionalities are briefly described in the following.

RADIUS does not provide a consistent security support. Data integrity and origin authentication is only required for RADIUS messages sent from the RADIUS server to the NAS. Although [11] extends data integrity and origin authentication with the Message-Authenticator attribute to RADIUS messages sent from the NAS to the RADIUS server, this extension is only required when transporting EAP messages. Furthermore, RADIUS does not provide for per-message confidentiality. A remedy would be the deployment of IPsec (cf. Section 3.8), which is defined to be used with RADIUS in [14], but its use is only optional. As Diameter is a complete re-design, it thus requires use of IPsec and optionally supports TLS as well.

Since RADIUS runs over UDP and since it does not provide support for re-transmissions, reliability is not guaranteed. Therefore, Diameter requires transport over TCP or the Stream Control Transmission Protocol (SCTP) [113].

Apart from these essential differences, Diameter has been designed to include a couple of minor but useful new functionalities such as an explicit support for agents (proxies, redirects and relays), capability negotiation between NAS and Diameter server or the automatic discovery of clients and servers. Finally, the issue of key transport between RADIUS server and NAS has been taken into account with a new EAP-Master-Session-Key attribute in Diameter.

Since the Diameter protocol operation is very similar to the RADIUS protocol operation described in Section 3.5 and Section 3.6, the reader is referred to [31, 33, 57] for a detailed description of the Diameter protocol operation. In this dissertation, the focus will be on RADIUS.

### 3.8 IPsec

IP Security (IPsec) is specified in [87]. It addresses security at the IP layer. IPsec can be used to protect the traffic between two communicating IP hosts. It does not necessarily run end-to-end between the two communicating hosts. If security gateways are implemented on intermediate nodes, IPsec may run between one of the hosts and one of the security gateways (cf. Fig. 3.6 cases 1 and 2) or between the two security gateways (cf. Fig. 3.6 case 3).

A notion fundamental to IPsec is a security association. A security association is a logical connection between two peers. The connection is unidirectional. If a bidirectional connection is desired, two security associations, one for each direction, have to be established. A security association defines a set of security parameters on either peer that are applied to all packets travelling over that
3.8. IPSEC

In IPsec, a security association may be set up using IKE (cf. Section 3.2) and is implemented either through the Encapsulating Security Payload (ESP) protocol (cf. Section 3.8.1) or through the Authentication Header (AH) protocol (cf. Section 3.8.2). If both ESP and AH are to be applied to the traffic between the security association end points, two security associations have to be established in each direction, one for ESP and one for AH.

At the source node of a security association, the security association is thus uniquely identified by a triplet consisting of a Security Parameter Index (SPI), the destination IP address of the destination node of the security association and a protocol identifier. The protocol identifier designates either ESP or AH. The triplet is used to retrieve the security parameters stored in the security association database.

A security association may be established in two modes. These are transport mode and tunnel mode. Both ESP and AH support both modes. In transport mode, protection is primarily provided for layers higher than the network layer, whereas in tunnel mode, protection is provided for the network layer by securing the original IP packet and prepending it with a new IP header.

Tunnel mode has to be used whenever either end of the security association is a security gateway. The main reason for this restriction goes back to fragmentation and reassembly and is elaborated on in the following.

In the case that the destination end point of the security association is a security gateway, the following issue may occur. If the receiving destination host is reachable via more than one security gateway, fragments issued by the sending source host may arrive at different security gateways. However, the security services applied to the fragments are bound to the security association. Fragments going to another security gateway have to be discarded. So when decryption etc. is being done by an interposed security gateway rather than by the end host, the security gateway must be the ostensible destination of the IP packets to ensure that it gets all the fragments.

In the case that the source end point of the security association is a security gateway, the security gateway may receive a 1500-Byte fragment from the source
host. When it adds an ESP or an AH header to it, the resulting packet may then again be too big and must be fragmented again. The problem is that the size of the total packet changes and that it is not possible to calculate fragment offsets in an unambiguous manner without keeping enough state to calculate new offsets for all the other fragments.

Both ESP and AH provide a certain set of security services (cf. Section 3.1). Since these security services are based on shared secrets, i.e. on cryptographic keys, a mechanism for key establishment is required. IPsec supports both manual and automated key establishment. Automated key establishment is supposed to be done via IKE.

IPsec is generally used to secure a connection between two stationary nodes. With MOBIKE (cf. Section 3.2), it may also be used between a stationary and a mobile host or even between two mobile hosts. However, since MOBIKE is still in draft status, these cases are not considered further in this dissertation. IPsec will be made use of in Chapter 5 and in Chapter 6 to secure connections between two core network nodes.

### 3.8.1 Encapsulating Security Payload

The Encapsulating Security Payload (ESP) provides confidentiality, data origin authentication, integrity and protects against replay attacks. Data origin authentication and integrity are joint security services and are referred to as authentication. Authentication and confidentiality are optional security services although one of them is mandatory.

ESP may be employed in either transport mode or tunnel mode. In transport mode, the ESP header is placed between the original IP header and the payload (cf. Fig. 3.7). Note that the IP header in IPv6 may contain several extension headers. The ESP header is added as a new extension header and may be placed anywhere within the sequence of extension headers. As only those extension headers placed after the ESP header will be encrypted, care has to be taken as
3.8. IPSEC

Figure 3.8: ESP tunnel mode

to which extension headers are placed before and which are placed after the ESP header. Note that payload and ESP trailer are encrypted, while ESP header, payload and ESP trailer are authenticated.

In tunnel mode, the ESP header is placed before the original IP packet (cf. Fig. 3.8). As the complete original IP packet is encrypted along with the ESP trailer, a new IP header has to be prepended to the ESP packet. Authentication further extends to the ESP header.

The ESP header contains the Security Parameter Index (SPI) and a sequence number. The sequence number serves protection against replay attacks and is not allowed to cycle if such protection is negotiated. If it was to cycle, a new security association would have to be established instead. The ESP trailer contains optional padding that might be required for certain encryption algorithms and a next header field indicating the transport protocol in IPv4 and IPv6 or the next extension header in IPv6.

ESP provides encryption with a symmetric encryption algorithm e.g. negotiated by IKE. Authentication is provided by a keyed MAC such as MD5 or SHA-1 (cf. Section 3.1.3). If both encryption and authentication are negotiated, encryption is performed before authentication. This way, replayed or bogus packets can be detected rapidly without having to decrypt the packet first. Also, decryption and verification of the authentication data can be done in parallel at the receiver.

3.8.2 Authentication Header

The IP Authentication Header (AH) provides authentication (joint security service of data origin authentication and integrity) and protection against replay attacks. It does not provide confidentiality as ESP. The advantage of AH over ESP is that the AH authentication data covers more of the IP packet than the ESP authentication data. While ESP does not protect any IP header fields in transport mode, AH protects at least those IP header fields that are not subject to change when travelling from hop to hop.

As ESP, AH may be employed in transport mode or in tunnel mode. In
transport mode, the AH is placed between the original IP header and the payload (cf. Fig. 3.9). In IPv6 it may be placed anywhere within the sequence of extension headers. No particular care has to be taken as to the placement.

In tunnel mode, the AH is placed before the original IP header, thereby protecting the complete original IP packet. Since the change of any field within the original IP header would lead to its refusal at the destination end point of the security association, a new IP header is placed before the AH.

The AH essentially contains the Security Parameter Index (SPI) of the security association, a sequence number optionally protecting against replay attacks, the calculated authentication data and the next header pointing to the transport protocol in IPv4 and IPv6 or the next extension header in IPv6. The authentication algorithm is negotiated e.g. in IKE and is generally provided by a keyed MAC such as MD5 or SHA-1 (cf. Section 3.1.3).

3.9 PANA

Wired and wireless network access generally requires some form of authentication. Some link-layer technologies provide authentication support, while others do not. Some networks require authentication with the Network Access Provider (NAP),
others with the Internet Service Provider (ISP), and yet others with both. Since authentication is usually required before gaining network access, the link-layer technology would have to be capable of dealing with these heterogeneous scenarios, which is rarely the case. Therefore, the Protocol for Carrying Authentication for Network Access (PANA) has been designed as a carrier for security protocols such as EAP (cf. Section 3.4). PANA is a network-layer access authentication protocol that can be used over any link layer supporting IP. Some requirements with respect to PANA are specified in [159].

The PANA architecture [83] consists of a PANA Client (PaC), a PANA Authentication Agent (PAA), an Enforcement Point (EP) and a backend authentication server (cf. Fig. 3.11).

The PaC is the client requesting for network access. The PAA is located in the access network and the network-side end point of the PANA protocol. It communicates with a backend authentication server using a AAA protocol such as RADIUS (cf. Section 3.5) or Diameter (cf. Section 3.7). The EP selectively allows and discards incoming IP packets. Unless the PaC has successfully authenticated to the backend authentication server, the EP only allows IP packets from PANA and other configuration protocols such as DHCP (cf. Section 2.3.3) or DHCPv6 (cf. Section 2.4.4). Only after successful authentication, the EP allows general IP packets as well.

The PANA protocol operation [64] is as follows. Through an initial discovery phase, the PaC learns the IP address of the PAA and the basic network capability information. Note that the PaC must configure an IP address before starting
PANA protocol operation. The IP address is called pre-PANA address and may be obtained via DHCP or DHCPv6 or configured as a link-local address using IPv4 or the automatic configuration of a link-local address in IPv6. In the subsequent authentication phase, an EAP message exchange as described in Section 3.4 is performed. The PAA acts as pass-through authenticator so that the actual authentication is performed between PaC and backend authentication server.

If the PaC authenticates successfully to the backend authentication server, the backend authentication server transfers the cryptographic keying material derived within the EAP method to the PAA. The PAA then transfers that keying material to the EP using a separate protocol (e.g., the Simple Network Management Protocol version 3 (SNMPv3) [69, 105]). After transfer of the keying material to the EP, secure communication between PaC and EP is enabled. Hence, the EP changes its filtering rules for that PaC to allow for general IP packets as well.

Although PANA is a promising approach allowing any authentication over any link layer when carrying EAP above IP, it is not yet standardized. Since EAP can be used with PPP and since PPP can be deployed over several link layers, EAP can be used over several link layers with PPP. In spite of the drawbacks of PPP incurring additional message exchanges during connection set-up and additional packet processing, EAP should preferably be used with PPP so long as PANA is not fully standardized.
Chapter 4

Mobility Management

Wireless packet-switched networks may support nomadic or mobile user behavior. A nomadic user remains attached via the same Point of Attachment (PoA) and the same Point of Service (PoS) during an active data session. Note that a data session may carry any type of traffic, i.e., files, emails, audio or video data, etc. Before changing the PoA or the PoS, the nomadic user terminates any active data session. While the nomadic user is on the move, it remains attached with the network. However, the network does not need to be notified of every change of PoA or PoS. It suffices to notify the network once in a while of the PoA or PoS the nomadic user is currently attached with. The necessary signalling operations are termed as location management.

A mobile user may change the PoA during an active data session. Note that changing the PoA requires manual user intervention in wired networks and that it is generally done automatically in wireless networks. If the network maintains the active data session, the change of PoA is denoted as Handover (HO). During a handover, the network has to be notified of the new PoA. Furthermore, care has to be taken as to those data packets that are already on their way towards the previous PoA. Those data packets have to be redirected towards the new PoA. The necessary signalling operations for both location update and packet redirection are termed as mobility management. Mobility management in general has been addressed by a couple of research papers such as [124, 16, 17, 35, 18].

A host desiring to gain access to a network, i.e., to send and receive IP packets over the network, needs to be authorized to do so. Whether or not a host is authorized to gain access to the network can be determined with the help of authentication. Since authentication reveals the identity of the host, an associated database may be consulted indicating whether to grant or deny access for the host. The identity of the host may be given as a link-layer address, an IP address, a Network Access Identifier (NAI) [32], etc., and depends on the type of network.

Network access involves the physical layer (L1), the link layer (L2) and the network layer (L3). The peer entities terminating the respective layer at the
CHAPTER 4. MOBILITY MANAGEMENT

Figure 4.1: Peer entities in network access

In wired networks, the L1 PoA is the socket the Network Interface Card (NIC) of the host is connected to. The L2 PoA and the L3 PoA are generally the same and represented by the interface of the first IP hop in IP-based networks. Note that although a switch is a L2 device, it cannot be considered as L2 PoA because it does not terminate the link-layer protocol at the other end.

In wireless networks, the L1 PoA and the L2 PoA are generally the same and represented by the entity at the other end of the wireless medium. The L3 PoA is represented by the interface of the first IP hop in IP-based networks. Since the notion “L1 PoA” is rarely used, the L2 PoA is usually referred to as PoA while the L3 PoA is usually referred to as Point of Service (PoS).

As long as a host is unknown to the network, IP packets arriving from or destined to the host need to be discarded. IP packets arriving from the host are either discarded at the PoA or at the PoS. IP packets destined to the host are discarded at some entity within the network. The choice of entity is network- and implementation-dependent. If access is granted, those nodes that may otherwise discard IP packets arriving from or destined to the host need to be notified of the host’s presence, a procedure which is generally referred to as registration.

In the following, it is assumed that a mobile host, i.e. e.g. a user’s mobile phone, is receiving IP packets from one or more distant correspondent nodes and that the mobile host changes both the PoA and the PoS so that core network mobility management is involved.

An IP packet is sent through the Internet via routing. In order to be independent of the data path, it carries source IP address and destination IP address in the IP header. Since routing is based on the network part of the destination IP address, an IP packet is always routed to the network indicated in the network part of the destination IP address. If a mobile host changes the network it has been attached to (i.e. the PoS), all IP packets with the same destination IP address continue to be routed to the previous network and do not arrive at the mobile host – unless other means are taken.

One such means are host-specific routing table entries. The routing tables of all routers along the data path from the correspondent node to the mobile host could be updated with a host-specific routing table entry reflecting the direction towards the new network of the mobile host. Note that the mobile host can keep
its IP address when host-specific routing table entries are deployed. However, with millions of mobile hosts, millions of host-specific routing table entries occur in many routers so that this approach is not scalable with respect to the number of routing table entries.

Another such means is to simply update the destination IP address of the mobile host to an IP address within the new network. The new IP address could be communicated to the correspondent nodes of the mobile host. Although an update of the destination IP address would have all IP packets reach the mobile host at its new location, all transport-layer connections would break [28]. The reason is that the application associated with the transport-layer port can in general not decide if IP packets received with a different source or destination IP address belong to the same transport-layer session. Therefore, IP packets with a different source or destination IP address are in general simply discarded so that the corresponding connections break. Even if these IP packets could be associated with the same transport-layer session, long delays would arise through the acquisition of a new IP address and the registration of that IP address at the correspondent nodes.

IP tunnelling, finally, is a means that avoids both host-specific routing table entries and that maintains existing connections by maintaining the IP address of the mobile host. However, the IP address can only be maintained through the acquisition of a second IP address used as the tunnel end point. Although IP tunnelling is suitable for mobility management, it has a couple of disadvantages such as high overhead and long delays through the acquisition of a new IP address and the registration of that IP address. Such delays are sufficient to affect real-time applications and throughput-sensitive applications.

In order to avoid IP tunnelling, MPLS-based solutions have been conceived. The MPLS-based solutions conceived so far either replace IP tunnelling by LSPs or perform time-consuming re-classifications of IP packets. The issue of long delays through the acquisition of a new IP address and the registration of that IP address, however, persists.

Active MPLS, eventually, avoids these disadvantages and the aforementioned ones. While moving within an Active MPLS domain, a mobile host may keep its IP address so that no delay arises through the acquisition of a new IP address. As a consequence, transport-layer connections do not break. Active MPLS is also a scalable mobility solution. The overhead is very low since Active MPLS is not based on IP tunnelling but on LSPs. The outage period is reduced to a minimum since packets are redirected as early as possible. Finally, no time- and resource-consuming re-classifications of IP packets are necessary.
4.1 Classification of Handover Types

A handover is the process of changing the PoA while maintaining an active data session.

Basically, there are two reasons to perform a handover. These are mobility and policy. A mobility-driven handover is executed when mobile host movement requires the handover in order not to lose network connectivity. Such handovers generally occur when the mobile host leaves the radio coverage of the current PoA while entering the radio coverage of another PoA. A policy-driven handover is executed when the handover to another PoA is more attractive than remaining attached to the current PoA. Such handovers may occur when the other PoA offers higher data rates, when the associated network offers better services, when the radio link conditions change, when new applications with other radio link requirements are launched, when the connection via the other PoA is associated with less cost, etc. Note that such handovers may occur both for mobile and for stationary hosts.

In case both PoAs implement the same physical-layer technology, the handover is an intra-technology handover. In case the PoAs implement different physical-layer technologies, the handover is an inter-technology handover. Intra-technology handovers are also called horizontal handovers, while inter-technology handovers are also called vertical handovers. Note that in order to perform an inter-technology handover, the mobile host has to be equipped with more than one NIC.

A L1 handover may be defined as a change of L1 information associated with the current attachment between mobile host and network. Such L1 information may be a frequency, a time slot or a L1 PoA such as the socket of a hub. An example is a laptop that is connected to a hub via cable and where the socket to which the laptop is connected is changed manually.

A L2 handover may be defined as a change of L2 information associated with the current attachment between mobile host and network. Such L2 information may be a L2 address or a L2 PoA such as an Access Point (AP) in WLANs. An example is a laptop that is associated with a WLAN AP and that changes to another WLAN AP in the same subnet.

A L3 handover may be defined as a change of L3 information associated with the current attachment between mobile host and network. Such L3 information may be a L3 address, a routing table entry or a PoS such as an Access Router in WLANs. An example is a laptop that is associated with a WLAN AP in one subnet and that changes to a WLAN AP in another subnet.

According to which network the PoAs belong to with respect to IP subnetting, the handover may entail a subnet or even a domain change. If the subnet does not change due to the handover, the handover is an intra-subnet handover. If the subnet changes, the handover is an inter-subnet handover. If the domain remains the same, the handover is an intra-domain handover. If the domain changes, the
handover is an inter-domain handover.

Note that an intra-subnet handover may be a L1, a L2 or even a L3 handover. An intra-subnet L3 handover is e.g. given when a mobile device is attached to a WLAN Access Router with two links to the WLAN and where the Access Router changes the default link due to failure of the previous default link. An inter-subnet handover, however, is always a L3 handovers. Note further that an intra-technology handover may be an intra-subnet or an inter-subnet handover, whereas an inter-technology handover is in general an inter-subnet handover.

Handovers can thus be classified as depicted in Fig. 4.2.

The set of mechanisms implementing a certain type of handover is referred to as mobility solution. Many technologies come with their own mobility solution for intra-technology intra-subnet handovers (cf. Chapter 6). In order to support inter-technology handovers or intra-technology inter-subnet handovers, higher-layer mobility solutions are needed. Such higher-layer mobility solutions may be based on IPv4, on IPv6 or on MPLS and are described in Section 4.4, in Section 4.5 and in Section 4.6, respectively.

The implementation of a mobility solution makes the corresponding handover hard, soft, smooth or seamless. A hard handover is a handover where the mobile host may receive data packets from at most one PoA during the handover. A soft handover is a handover where the mobile host may receive data packets by more than one PoA simultaneously. Currently, soft handovers are only supported by UMTS through the Code Division Multiple Access (CDMA) wireless access method where two radio signals may be transmitted and received over the same frequency at the same time using orthogonal scrambling codes.
Apart from intra-UMTS handovers, intra-technology handovers are hard handovers. On the network side, deploying soft handovers is rather complex and expensive. In UMTS, this is e.g. only done for circuit-switched and not for packet-switched services. On the mobile host side, other reasons speak against soft handovers. If only one Network Interface Card (NIC) is deployed, soft handovers need to be supported by the corresponding technology (currently only UMTS). If two NICs (of the same technology) are deployed in the mobile host, soft handovers are theoretically conceivable. However, in practice, two NICs of the same technology are not deployed in the mobile host. The reasons are that two NICs are either too expensive or that the energy consumption at the mobile host is too high or that the interference between the radio signals is too high during the handover.

For inter-technology handovers, two NICs are required so that the reason that two NICs are too expensive does not apply. However, the other reasons persist. Therefore, the focus is on hard handovers in this dissertation.

A smooth handover is a handover where packet loss is minimized. A seamless handover is a handover that is transparent to the application. In particular, the handover may be seamless for one application and not for another.

4.2 Handover Procedure

The handover procedure starts as soon as a particular event triggering the handover occurs. The event is also referred to as handover trigger. Movement detection may be such a handover trigger. The handover trigger may occur in the mobile device or in the network. If it occurs in the mobile device, the handover is a mobile-initiated handover. If it occurs in the network, the handover is a network-initiated handover.

Possibly there are more than one PoAs to handover to. The decision as to which PoA to handover to may be up to the mobile device or up to the network. If it is up to the mobile device, the handover is a mobile-controlled handover. If it is up to the network, the handover is a network-controlled handover.

If the handover decision of a mobile-controlled handover is based on information received from the network, e.g. about the utilization of neighboring L2 PoAs or about the corresponding PoSs, the handover is mobile-controlled and network-assisted. If the handover decision in a network-controlled mobility solution is based on information received from the mobile device, e.g. about the signal quality of neighboring L2 PoAs, the handover is network-controlled and mobile-assisted. Handovers in cellular systems are typically network-controlled and mobile-assisted.

Once the decision as to which PoA to handover to is taken, the actual handover procedure implemented by a particular mobility solution is executed. In order to keep the consumed resources to a minimum, the goal of any mobility solution
should be to update the states of both mobile device and network in such a way as if the mobile device had attached to the target PoA without handing over from the previous PoA.

Updating the state of the network generally involves updating the location of the mobile device at one or more location update points and transferring the mobile device context from the previous PoAs to the target PoAs. The mobile device context contains e.g. the mobile device IP address, authentication credentials, an encryption key, etc. If the mobile device context is not transferred, it may be re-established. The re-establishment of configuration information is also referred to as re-registration, while the re-establishment of authentication information is also referred to as re-authentication. Note that context transfer of authentication information is also referred to as fast re-authentication.

The handover procedure can be subdivided into the L1 handover procedure, the L2 handover procedure and the L3 handover procedure. The L2 handover procedure involves the exchange of messages between the mobile device and the previous and target L2 PoA. The time period between the first L2 signalling message and the last L2 signalling message is defined here as the L2 handover delay.

The L3 handover procedure involves the exchange of messages between the mobile device and the previous and target PoS. The time period between the first L3 signalling message and the last L3 signalling message is defined here as the L3 handover delay.

Note that the mobile device may not receive or send any network-layer packets during the L1 handover, the L2 handover and some mobility-solution-dependent part of the L3 handover. This time period is defined here as outage period. A long outage period affects in particular real-time applications and throughput-sensitive applications.

4.3 Classification of Mobility Solutions

Mobility solutions may be classified according to which handovers they address. Hence, there are intra-technology and inter-technology mobility solutions as well as intra-subnet, inter-subnet, intra-domain and inter-domain mobility solutions (cf. Section 4.1). Since the focus of this document is on core network mobility, i.e. on L3 handovers, the focus is on inter-subnet mobility solutions.

Such inter-subnet mobility solutions may be classified according to the protocol they are based on. Hence, there are IPv4-based, IPv6-based and MPLS-based mobility solutions. Inter-subnet mobility solutions that are based on multiple such protocols are called hybrid solutions.

Those inter-subnet mobility solutions addressing intra-domain handovers are also called micro-mobility or local mobility solutions. Those addressing inter-domain handovers are also called macro-mobility or global mobility solutions.
CHAPTER 4. MOBILITY MANAGEMENT

Some mobility solutions trigger the L3 handover after the L2 handover. These are called reactive solutions. Other mobility solutions trigger the L3 handover before the L2 handover. These are called predictive solutions. Note that predictive solutions may be in need of some mechanism to provide network-layer information of the new PoS via the current PoS. This is further described in Chapter 7.

Fig. 4.3 shows that the outage period of predictive mobility solutions is inherently shorter than the outage period of reactive mobility solutions. Reactive mobility solutions perform the L3 handover after the L2 handover. The outage period starts at the beginning of the L1 handover and covers the L2 handover as well as a mobility-solution-dependent part of the L3 handover, but at most the complete L3 handover.

Predictive mobility solutions perform some part of the L3 Handover (HO) before the L1 HO. This part is denoted as L3 HO (a) in Fig. 4.3, while the remaining part of the L3 HO is denoted as L3 HO (b). The outage period starts at the beginning of the L1 handover and covers the L2 handover as well as some mobility-solution-dependent part of L3 HO (b), but at most the complete L3 HO (b). Since L3 HO (b) is shorter than the complete L3 handover, the outage period is inherently shorter as well. Note that if L3 HO (a) is executed far before the actual L1 handover, the outage period may extend to before the L1 handover. Therefore, care must be taken as to when to start L3 HO (a).

Some mobility solutions introduce an additional hierarchy level yielding one global (inter-domain) location update point and another local (intra-domain) location update point. The intention here is to keep an intra-domain handover transparent to the global location update point by keeping the location update local to the intra-domain location update point. Updating the location at the intra-domain location update point also results in a lower location update signalling delay (during intra-domain handovers). Furthermore, no messages need to be exchanged with the global location update point during an intra-domain
location update.

In order to shorten the outage period during the location update at the location update point, some mobility solutions make use of a tunnel between the previous and the new PoS. Some mobility solutions further provide buffering either at the previous PoS, the new PoS or at both PoSs in order to eliminate packet loss almost completely. Note that although packet loss may almost completely be eliminated, packet throughput is still affected.

Since a handover is a user-specific action, additional user-specific entries cannot be avoided as part of the mobility solution. While all mobility solutions keep a user-specific entry at the location update point, some keep a user-specific entry also at the PoS, whereas others keep user-specific entries at all network nodes. Some mobility solutions keep user-specific entries at nodes other than the location update point only during the handover, while other mobility solutions keep user-specific entries as long as the user is registered in the network. Those mobility solutions keeping user-specific entries at all network nodes as long as the user is registered in the network suffer from scalability issues in larger networks.

Some mobility solutions require the use of more than one routable, i.e., public, IP address for the mobile device, while others allow the use of private IP addresses. Yet others do not need any further IP addresses at all. A high consumption of IP addresses is in particular an issue in IPv4-based environments.

Finally, some mobility solutions such as [136] deploy IP- or MPLS-based bi-casting or multi-casting during handover in order to decrease the outage period further. However, this comes at the expense of increased network utilization within the network and duplicated packets at the mobile device. Duplicated packets may in particular be an issue for UDP-based applications.

4.4 IPv4-based mobility

4.4.1 Mobile IPv4

Mobile IPv4 (MIPv4) [118, 145] is based on the concept of IP tunnelling to support mobility for a Mobile Node (MN). A MN is either permanently subscribed to a home network or subscribes to a network (thereby becoming its home network) on start-up. Through the subscription procedure the MN acquires an IP address which is referred to as its Home Address (HoA).

The two main entities specified in MIPv4 are the Home Agent (HA) located in the home network and the Foreign Agent (FA) located in a foreign network (cf. Fig. 4.4). Both HA and FAs periodically broadcast Agent Advertisements.

After obtaining link-layer connectivity with a PoA, the MN has to obtain network-layer connectivity. With MIPv4, the MN may either wait for an Agent Advertisement or solicit one by sending an Agent Solicitation (cf. Fig. 4.5). Agent Advertisements are extensions of the ICMP Router Advertisements for MIPv4.
Figure 4.4: MIPv4 architecture and data path in the FA CoA case

An *Agent Advertisement* contains a FA Care-of Address (CoA) and optionally a prefix-length extension. With the help of this prefix-length extension, the MN can determine whether it is attaching to its home network or to a foreign network. When the MN does not receive an *Agent Advertisement* for some time, it may perform DHCP to request an IP address from a DHCP server (cf. Fig. 4.6). Such an IP address is called *Collocated CoA (CCoA)*. In both cases, the MN registers the obtained CoA with its HA. In general, the MN uses its HoA as identifier. However, it may be desirable to allocate the HoA on demand and not permanently. It is thus also possible for the MN to identify itself with a Network Access Identifier (NAI) and request a HoA be assigned by the HA [32].

In the case that the MN obtains the CoA from a FA (cf. Fig. 4.5), it registers the FA CoA with its HA by sending a MIPv4 Registration Request (RegReq) to the FA. The FA relays the RegReq to the HA. In the case that the MN obtains the CoA via DHCP (cf. Fig. 4.6), it registers the CCoA with its HA by sending a MIPv4 RegReq directly to its HA. In the case that the MN identifies itself with a NAI and requests assignment of a HoA, the HA allocates and assigns a HoA to the MN. In either case, the HA stores the CoA together with the HoA and
4.4. IPV4-BASED MOBILITY

The HA notifies the link the MN’s HoA belongs to of the new link-layer address to be associated with the MN’s HoA in order to have further IP packets destined to the MN’s HoA be directed to itself. When Ethernet is the underlying link-layer, a gratuitous ARP with the MN’s HoA and the HA’s MAC address may be sent as notification. Finally, the HA responds by sending a Registration Reply (RegReply) back to the source IP address of the RegReq, i.e. either to the FA or directly to the MN. If the HA has assigned a HoA, it includes the assigned HoA together with the NAI the MN placed into the RegReq into the RegReply.

The control plane protocol stack is depicted in Fig. 4.7. MIPv4 uses UDP and IP as underlying transport-layer and network-layer protocols.

When a Correspondent Node (CN) has an IP packet to send to a MN, it sends the IP packet to the fixed HoA of the MN where the packet is intercepted by the HA. The HA tunnels the IP packet to the current location of the MN, i.e. to the
registered CoA. In case of a FA CoA (cf. Fig. 4.4), the tunnel leads to the FA. In case of a CCoA it leads directly to the MN. If the tunnel leads to the FA, the FA detunnels and finally delivers the packet to the MN, whereas the MN detunnels the packet itself if the tunnel leads directly to the MN. The user plane protocol stack is depicted in Fig. 4.8 for the FA CoA case and in Fig. 4.9 for the CCoA case. For the tunnel between HA and FA three different tunnelling protocols may be used. These are IP-in-IP encapsulation, Generic Routing Encapsulation (GRE) and Minimal Routing Encapsulation (MRE). They are described in Section 2.3.4.

A MN may be identified at a FA either by the combination of HA address and HoA, which is necessary for different MNs having the same private IP address as HoA. Otherwise, a MN may add a NAI [32]. In the latter case, the MN may use dynamic assignment of the HA [91] in order to assign an optimal HA while allowing any suitable method for HA selection.
4.4. IPV4-BASED MOBILITY

<table>
<thead>
<tr>
<th>Predictive/Reactive</th>
<th>Reactive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intra-/Inter-technology</td>
<td>Both</td>
</tr>
<tr>
<td>Micro-/Macro-mobility</td>
<td>Both</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>No</td>
</tr>
<tr>
<td>Data path between PoSs</td>
<td>No</td>
</tr>
<tr>
<td>Permanent user-specific entries</td>
<td>HA, FA</td>
</tr>
<tr>
<td>Temporary user-specific entries</td>
<td>None</td>
</tr>
<tr>
<td>IP addresses</td>
<td>Permanent HoA, changing CoA</td>
</tr>
<tr>
<td>Main objective</td>
<td>Enable mobility</td>
</tr>
<tr>
<td>Main disadvantage</td>
<td>Long outage period</td>
</tr>
</tbody>
</table>

Table 4.1: MIPv4 – Summary

Note that the same procedure is executed both for start-up and during handover. In particular, all IP packets arriving at the previous FA are lost after link-layer handover. Furthermore, the procedure can only be started after link-layer handover. Since the link-layer handover remains in general unnoticed by the network layer, either the MN has to wait for the next Agent Advertisement or it has to periodically send Agent Solicitations. While the first increases the outage period due to handover, the second has a negative impact on bandwidth.

Up to now, Mobile IP can be considered as the only existing macro-mobility solution for IP-based applications (cf. also [97]). The greatest disadvantage of Mobile IP are the long signalling delays and the high additional overhead introduced through IP tunnelling. The impact of these disadvantages is even worse in environments with frequent handovers where a strong service degradation is to be expected. In order to better support handovers within such environments, several micro-mobility solutions have been conceived. The main characteristics are summarized in Table 4.1.

4.4.2 Proxy Mobile IPv4

Proxy MIPv4 (PMIPv4) has been specified within the WiMAX forum (cf. Section 6.2). The main incentives for the specification of PMIPv4 are the following. PMIPv4 does not impose a MIPv4 client to be implemented on the mobile device. It is also transparent to the applications running on the mobile device; it is e.g. possible to run MIPv4 or any Virtual Private Network (VPN) client to a corporate network on top of PMIPv4. This is not easily possible with MIPv4. Probably the major incentive for the use of PMIPv4 is the fact that it is completely network-controlled.

The PMIPv4 start-up procedure is depicted in Fig. 4.10. The involved entities are the Mobile Subscriber Station (MSS), the Access Services Network Gateway (ASN-GW) also acting as Foreign Agent (FA), the DHCP server and the Home
Agent (HA). When the DHCPDISCOVER from the MSS arrives at the ASN GW, there are three options to provide the fixed Home Address (HoA)\(^1\) for the MSS.

If the IP address has been provided during AAA access authentication, it can directly be assigned by the ASN-GW. If the address of the DHCP server has been provided during AAA access authentication, the ASN-GW may also relay the DHCPDISCOVER to the DHCP server. If, finally, the address of the HA has been provided during AAA access authentication, the ASN-GW may send a MIPv4 RegReq to the HA and request assignment of an IP address. In the last case, the NAI is used to identify the MSS so that the assignment of a HoA can be deferred to after sending the MIPv4 RegReq. In Fig. 4.10, the ASN-GW acts as conventional DHCP relay in order to retrieve the HoA for the MSS.

Once the DHCP message exchange is complete, the ASN-GW acting as PMIPv4 client registers the MSS at the HA. It uses one of its own IP addresses as FA CoA, the retrieved HoA for the MSS and the NAI of the MSS as identifier. Only if the MIPv4 registration completes successfully, the ASN-GW accomplishes the start-up procedure with a DHCPACK. Otherwise, it sends a DHCPNAK. The control plane protocol stack is depicted in Fig. 4.11.

Note that the resulting data path for PMIPv4 is the same as the data path for MIPv4 when the mobile node acquires a FA CoA (cf. Fig. 4.4). The user plane protocol stack (cf. Fig. 4.12) also looks very alike. The only difference is that MPLS is another option for the tunnel between the HA and the PMIPv4 client in the ASN-GW. Note also that PMIPv4 introduces a split in the path of signalling and user data messages. While signalling messages flow between HA and PMIPv4 client, user data messages flow between HA and MSS.

\(^1\)The HoA is called Point of Attachment (PoA) address within the WiMAX specifications.
The WiMAX handover procedure (cf. Fig. 4.13) is both network-initiated and network-controlled. When a corresponding handover trigger occurs, the old ASN-GW (oASN-GW) sends a MIPv4 RegReq message to the new ASN-GW (nASN-GW). The new ASN-GW updates the MIPv4 RegReq message with its own FA CoA and forwards it to the HA.

Since the location update at the HA may take considerable time, a tunnel is established between oASN-GW and nASN-GW. The tunnel is set up via the R4 data path establishment procedure. Once the R4 data path is established, the oASN-GW sends an R6 HO Indication message to the old BS (oBS), while the nASN-GW establishes an R6 data path with the new BS (nBS). The oBS notifies the MSS of the necessary handover with an R1 HO Indication message. After an R1 HO Indication Ack and a subsequent R6 HO Indication Ack message, the MSS only needs to associate with the nBS.

Eventually, the HA updates the MIPv4 binding for the MSS so that further data packets are sent to the new ASN-GW. On receipt of the MIPv4 RegReply message, the new ASN-GW relays the MIPv4 RegReply message to the old ASN-GW, thereby accomplishing the handover.

Note that the MSS keeps the HoA it has acquired during start-up during a handover. Note also that the MSS is not aware of a CoA. For inter-technology handovers PMIPv4 cannot be deployed since inter-technology handovers require the change of NIC at the MSS which PMIPv4 cannot influence. That might, however, become possible through IEEE 802.21 (cf. Section 7.2).

Although PMIPv4 has some advantages, an IP tunnel has to be set up between
oASN-GW and nASN-GW during the handover. Furthermore, the overhead is high due to the use of IP tunnelling. The main characteristics are summarized in Table 4.2.

4.4.3 Low Latency Handoffs in Mobile IPv4

MIPv4 has been designed without assuming any interaction between link and network layers. Therefore, MIPv4 can be deployed over any link layer. However, when moving to a new subnet, a mobile device is only aware of the handover when it receives an Agent Advertisement from a Foreign Agent of the new subnet. Only then can the mobile device initiate the network-layer handover. Since these Agent Advertisements are sent periodically, the time until the mobile device, i.e. the MIPv4 implementation on the mobile device, is aware of the handover may be considerable. Through periodic PrRtSol messages soliciting an Agent Advertisement, the mobile device may decrease that time. However, that comes at the expense of a considerable increase of the consumed bandwidth. Of course,
The FA of the previous subnet could provide the mobile device while being attached to that FA with all kinds of information, commands etc., but without any interaction with the link layer, neither that FA nor the mobile device MIPv4 implementation are aware of the characteristics of the current link.

The introduction of an interaction between link and network layers has thus been one of the main objectives of Low Latency Handoffs\(^2\) in Mobile IPv4 [98]. The interaction is achieved via link-layer triggers, i.e. via event notifications from the link layer to the network layer. Link-layer triggers serve to initiate the network-layer handover. The main benefit of these link-layer triggers is that the network-layer handover can be triggered before the link-layer handover through which the outage period is reduced (cf. Fig. 4.3). They also allow for network-initiated handovers which are not possible with MIPv4. Finally, periodic Agent Advertisements or PrRtsol messages are not necessary with link-layer triggers.

A second objective has been the possibility of establishing a bi-directional tunnel between previous and new FA. The main benefit of this bi-directional tunnel is that it further reduces the outage period during the long signalling delay from the new FA to the HA.

The two objectives result in two different registration methods that may also be combined. These are the pre-registration and the post-registration handover methods. Note that each of the two registration methods covers only one of the two objectives – and not the other. In order to implement both objectives, the registration methods have to be combined. Post-registration means that the MIPv4 registration is postponed.

A performance analysis of the Low Latency Handoffs schemes in comparison to MIPv4 has been performed by [60]. However, only the outage period has been taken into account. In Section 5.4, a much more exhaustive performance analysis is performed that takes many mobility solutions into account and that evaluates them with respect to many parameters.

Note that Low Latency Handoffs in Mobile IPv4 is based on Mobile IPv4, i.e. in particular the involved components and the start-up procedure are the same. The protocol stack on the user plane is also the same. The protocol stack on the control plane only differs in that MIPv4 in Fig. 4.7 needs to be replaced by Low Latency Handoffs in Mobile IPv4.

### 4.4.3.1 Pre-Registration Handover Method

The pre-registration handover method addresses the update of the data path at the Home Agent before the actual link-layer handover and may be mobile-initiated or network-initiated. In the case that it is mobile-initiated (cf. Fig. 4.14), the link-layer trigger occurs at the mobile device. Such a trigger could be the link-layer interface reporting that the signal strength to another point of attachment

\(^2\)Throughout the whole document, the notion handover is preferred over the notion handoff. It is maintained here since it is part of a proper name.
has been better for a configurable period of time and that it is probable that the attachment to the current point of attachment is lost soon. The link-layer trigger needs to contain an identifier of the new point of attachment, the new subnet or the new FA.

The mobile device then places the identifier into a Generalized Link Layer and IPv4 Address extension and sends the extension in a \textit{PrRtSol} message to the current FA. The current FA resolves the identifier to the IPv4 address of the new FA. If the current FA does not yet have a cached \textit{Agent Advertisement} of the new FA, it requests one by sending a \textit{PrRtSol} to the new FA. When it receives the corresponding \textit{PrRtAdv} from the new FA, it caches this advertisement and sends it to the mobile device.

In the case that the link-layer trigger is network-initiated, it may occur at the current FA or at the new FA. If it occurs at the current FA (cf. Fig. 4.15), it contains an identifier of the mobile device and an identifier of the new FA. On receipt of the link-layer trigger, the current FA sends an unsolicited \textit{PrRtAdv} message to the mobile device on behalf of the new FA unless it still has to request the \textit{PrRtAdv} from the new FA.

If the link-layer trigger occurs at the new FA (cf. Fig. 4.16), it contains an identifier of the mobile device and an identifier of the current FA. The Low Latency Handoffs for Mobile IPv4 specifies that the new FA shall tunnel an \textit{Agent Advertisement} via the current FA to the mobile device. Tunnelling the \textit{Agent Advertisement} is necessary since the communication between two FAs is secured via an IPsec tunnel (cf. Section 3.8). In either case, the mobile device receives the \textit{PrRtAdv} of the new FA via the current FA.
4.4. IPV4-BASED MOBILITY

4.4.1 Pre-Registration Handover Method

It then registers with its HA by exchanging MIPv4 RegReq and RegReply messages (cf. Fig. 4.17). Note that the MIPv4 RegReq issued by the mobile device before the L2 handover is addressed to the new FA and must therefore be routed through the old FA. As long as the mobile device does not announce its presence at the new FA (e.g. because it is still involved in link-layer handover), the new FA buffers the MIPv4 RegReply and any further message destined to the mobile device.

The mobile device may announce its presence by sending an Agent Solicitation to the new FA on receipt of a link-layer trigger indicating that the link to the new point of attachment is up. The Agent Solicitation may itself be implemented as a corresponding link-layer trigger at the new FA, thus triggering the transmission of a buffered RegReply message.

4.4.3.2 Post-Registration Handover Method

The post-registration handover method addresses the update of the data path at the current FA in that it establishes a Bidirectional Edge Tunnel (BET) between current and new FA. Since the update of the data path at the HA may take very
long, independently of whether or not the pre-registration handover method is used, the data path can be updated at the current FA and the data packets be tunnelled to the new FA while the data path update at the HA is in progress. The name post-registration handover method is puzzling because this method is triggered before connectivity to the current FA is lost.

The trigger for the post-registration handover method may occur at the current FA or at the new FA. For simplicity, Fig. 4.18 only shows the case where the trigger occurs at the current FA. The current FA then sends a HRqst message to the new FA. The new FA replies with a HReply message. Both messages are carried over IP/UDP and contain the L2 address of the mobile device, its IPv4 HoA and the address of its HA.

When the current FA receives a link-down trigger, it starts tunnelling packets to the previously established BET (cf. Fig. 4.19). The new FA receiving these packets buffers them until it receives a link-up trigger. Only then it starts delivering the packets to the mobile device. Since it may not yet be the default router of the mobile device, it has to be considered that the mobile device must send upstream data packets to the new FA link-layer address instead of to the previous FA link-layer address. In WLAN environments this can be achieved by having the new FA send a proxy ARP containing the IPv4 address of the previous FA and its own link-layer address.

On receipt of the link-up trigger, the mobile device may initiate MIPv4 registration with its HA. Note that during MIPv4 registration, user data packets are sent inefficiently from the HA to the previous FA and from there to the new FA. The negative impact on network utilization is in particular high because of the high overhead due to IP tunnelling. Furthermore, the outage period is still relatively long since the BET between previous and new FA has to be established at the beginning of the handover.
4.4.3.3 Combined Handover Method

The combined handover method (cf. Fig. 4.20) starts with the pre-registration handover method. If the link-layer trigger initiating the pre-registration handover method occurs at the mobile device or at the old FA, the new FA starts a timer on receipt of the MIPv4 RegReq message. If the link-layer trigger initiating the pre-registration handover method occurs at the new FA, the new FA starts the timer on sending the PrRtAdv message via the old FA to the mobile device.

If the timer expires before the new FA has received the MIPv4 RegReply message, it starts the post-registration handover method and establishes the BET with the old FA. Once the mobile device registers with the new FA via an Agent Solicitation message, the data packets arriving over the BET may directly be delivered to the mobile device. Note that the corresponding Agent Advertisement is only sent when both the Agent Solicitation and the HReply message have been received. The main characteristics of the combined handover method are summarized in Table 4.3.
CHAPTER 4. MOBILITY MANAGEMENT

Figure 4.20: Combined handover method of Low Latency Handoffs in Mobile IPv4

<table>
<thead>
<tr>
<th>Predictive/Reactive</th>
<th>Predictive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intra-/Inter-technology</td>
<td>Both</td>
</tr>
<tr>
<td>Micro-/Macro-mobility</td>
<td>Micro-mobility</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>No</td>
</tr>
<tr>
<td>Data path between PoSs</td>
<td>Yes</td>
</tr>
<tr>
<td>Permanent user-specific entries</td>
<td>HA, FA</td>
</tr>
<tr>
<td>Temporary user-specific entries</td>
<td>Old FA</td>
</tr>
<tr>
<td>IP addresses</td>
<td>Changing FA CoA</td>
</tr>
<tr>
<td>Main objective</td>
<td>Reduce outage period</td>
</tr>
<tr>
<td>Main disadvantage</td>
<td>Long outage period</td>
</tr>
</tbody>
</table>

Table 4.3: Low Latency Handoffs in Mobile IPv4 – Summary (Combined handover method)

4.4.4 HAWAII

Handoff-Aware Wireless Access Internet Infrastructure (HAWAII) [129] extends MIPv4 with optimizations in routing and forwarding in order to more efficiently support intra-domain mobility. When a MN first enters a network, it acquires a CCoA, i.e. no FAs are present in the HAWAII domain, and registers the CCoA with its HA. It then sends a Path Set-Up Power-Up message to the domain root router whose address might be obtained via DHCP. However, this is not mentioned in the paper. The Path Set-Up Power-Up message has the effect of establishing a host-specific routing table entry in each router along the path to the domain root router. Other routers are not involved. A communication
between two mobile nodes is thus directed over the domain root router since every router has a default routing table entry pointing to the domain root router. Note that multiple gateways, i.e. domain root routers, are not explicitly supported by HAWAII.

As in MIPv4, data packets from correspondent nodes are sent to the fixed HoA of the MN where they are intercepted by the HA and tunnelled to the registered CCoA, i.e. directly to the mobile node. Within the HAWAII domain, the data packets are forwarded according to the host-specific routing table entries. The protocol stack on the user plane is the same as in the CCoA case of MIPv4 (cf. Fig. 4.9). The protocol stack on the control plane only differs in that MIPv4 in Fig. 4.7 needs to be replaced by HAWAII (being an extension to MIPv4).

If the MN moves within the HAWAII domain, the data path from the domain root router to the MN has to be updated. Such an update is done by Path Set-Up Update messages and addressed by path set-up schemes. The path set-up schemes are divided into forwarding and non-forwarding schemes. Both schemes consist of sending a Path Set-Up Update message to the old PoS via the new PoS.

In the forwarding schemes, the Path Set-Up Update message is only evaluated on arrival at the old PoS. The old PoS then sends the Path Set-Up Update message towards the new PoS. One reason might be that no changes shall be made to the HAWAII domain before the old PoS (possibly containing authentication information for the MN) has confirmed the authenticity of the Path Set-Up Update message. Each router on the path from old PoS to new PoS updates or creates a host-specific routing table entry. Since the update of the data path starts at the old PoS where the data packets are forwarded towards the new PoS, these schemes are called forwarding schemes.

In the non-forwarding schemes, the Path Set-Up Update message directly leads to the update or the creation of host-specific routing table entries along the path from the new PoS to the old PoS. The result is that data packets are redirected by the cross-over router and thus before the Path Set-Up Update message arrives at the old PoS. The cross-over router is located at the intersection of the path from the domain root router to the old PoS and the path from the domain root router to the new PoS. Note that no packets are forwarded by the old PoS since the last packet arriving at the old PoS arrives before the Path Set-Up Update message. Therefore, these schemes are referred to as non-forwarding schemes.

In order to avoid that most or even all of the routers in the HAWAII domain contain a host-specific routing table entry for a particular MN after several handovers, the host-specific routing table entries are associated with a timer. The timer is reset by Path Refresh messages that the MN has to send periodically to its current PoS. If the timer expires, the associated host-specific routing table entry is deleted.

As HAWAII is based on permanent host-specific routing table entries in all
CHAPTER 4. MOBILITY MANAGEMENT

<table>
<thead>
<tr>
<th>Handover method</th>
<th>Forwarding</th>
<th>Non-forwarding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predictive/Reactive</td>
<td>Reactive</td>
<td>Reactive</td>
</tr>
<tr>
<td>Intra-/Inter-technology</td>
<td>Both</td>
<td>Both</td>
</tr>
<tr>
<td>Micro-/Macro-mobility</td>
<td>Micro-mobility</td>
<td>Micro-mobility</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Data path between PoSs</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Permanent user-specific entries</td>
<td>from GW to PoS</td>
<td>from GW to PoS</td>
</tr>
<tr>
<td>Temporary user-specific entries</td>
<td>from old to new PoS</td>
<td>from old to new PoS</td>
</tr>
<tr>
<td>IP addresses</td>
<td>Permanent CCoA</td>
<td>Permanent CCoA</td>
</tr>
<tr>
<td>Main objective</td>
<td>Avoid changing IP addresses</td>
<td></td>
</tr>
<tr>
<td>Main disadvantage</td>
<td>Scalability issues</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.4: HAWAII - Summary

nodes along the data path from the domain root router to the current PoS, it cannot be considered as scalable approach for core networks. Furthermore, the association of timers with host-specific routing table entries imposes high overhead onto the control plane. Finally, the periodic Path Refresh messages lead to an unnecessary consumption of resources on the link between the mobile device and the current L2 PoA which is in particular an issue for wireless links. The main characteristics of both forwarding schemes are summarized in Table 4.4.

4.4.5 Cellular IP

Cellular IP [150] is another micro-mobility approach that is based on host-specific entries in routing tables. The idea is to install a routing cache in selected routers and to monitor passing packets sent by the MN which are used to create bindings between the MN and the incoming port number in the routing cache. Whenever a data packet destined for the MN arrives at a router with a routing cache, the packet is sent to the port stored in the routing cache, i.e. on the reverse data path. In order to minimize control traffic, the entries in the routing caches are not cleared explicitly after a handover. Instead, they are associated with timers and cleared automatically when the timer expires. In order to maintain the bindings in the routing caches, a MN has to send so-called route-update packets periodically when it does not have data packets to send.

The start-up procedure is not explicitly described in [150]. However, it can be assumed that the MN acquires an IP address (e.g. through DHCP) and starts sending data packets or route-update packets when it does not have data packets to send in order to establish the reverse data path for downstream data packets. When the MN makes use of MIPv4, the protocol stack on the user plane is the same as that depicted in Fig. 4.9 and the protocol stack on the control plane is the same as that depicted in Fig. 4.7. Note that the forwarding plane of each IP
4.4. IPV4-BASED MOBILITY

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Predictive/Reactive</td>
<td>Reactive</td>
</tr>
<tr>
<td>Intra-/Inter-technology</td>
<td>Intra-technology</td>
</tr>
<tr>
<td>Micro-/Macro-mobility</td>
<td>Micro-mobility</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>No</td>
</tr>
<tr>
<td>Data path between PoSs</td>
<td>No</td>
</tr>
<tr>
<td>Permanent user-specific entries</td>
<td>from GW to PoS</td>
</tr>
<tr>
<td>Temporary user-specific entries</td>
<td>from GW to old PoS</td>
</tr>
<tr>
<td>IP addresses</td>
<td>Permanent intra-domain IP address</td>
</tr>
<tr>
<td>Main objective</td>
<td>Avoid changing IP addresses</td>
</tr>
<tr>
<td>Main disadvantage</td>
<td>Scalability issues</td>
</tr>
</tbody>
</table>

Table 4.5: Cellular IP – Summary

router needs to be adapted in order to implement the idea of Cellular IP.

During a handover a MN moves from one PoS to another so that for a short period of time the data packets issued by the MN lead to the creation of two reverse paths, one from the ingress domain router to the old PoS and one from the ingress domain router to the new PoS. At the intersection of the two paths, the packets are then unnecessarily duplicated until the path from the ingress domain router to the old PoS is released due to expiration of the timers associated with the corresponding routing table entries. Like HAWAII (cf. Section 4.4.4), Cellular IP is based on host-specific routing table entries along the data path from the domain ingress router to the current PoS and cannot be considered as scalable approach for core networks. Furthermore, when changing the PoS, all packets sent to the previous PoS are lost until the first data packet sent by the MN updates the first router located both on the previous and on the new data paths.

The main characteristics of Cellular IP are summarized in Table 4.5.

4.4.6 GMM

GPRS Mobility Management (GMM) is a protocol that has been designed for cellular networks. It is specified in [1]. Mobility management in the packet-switched part of cellular networks involves four different entities. These are the Mobile Station (MS), the Serving GPRS Support Node (SGSN), the Gateway GPRS Support Node (GGSN) and the Home Location Register (HLR) (cf. Fig. 4.21).

The start-up procedure is explained in detail in Section 6.3. The MS has to register its current location with the core network, i.e. with the current SGSN and the HLR, a procedure which is referred to as GPRS attach. In order to send and receive data packets, a Packet Data Protocol (PDP) context has to be established at the MS, the current SGSN and the GGSN. The PDP context is established whenever a data session is set up and may be removed after termination of the data session.
An IP packet arriving at a GGSN from a certain Packet Data Network (PDN) attached to the GGSN is tunnelled via the GPRS Tunnelling Protocol (GTP) to the appropriate SGSN where it is detunnelled and delivered via the appropriate access network to the MS. Since GTP messages are transported over UDP/IP, the resulting protocol stack in the core network becomes very complex on the user plane. Both the protocol stack on the user plane and the protocol stack on the control plane are presented in Section 6.3.2.

Mobility management in the packet-switched part of cellular networks is based on GMM between the MS and the SGSN, on GTP between SGSN and GGSN and on the Mobile Application Part (MAP) protocol between SGSN and HLR. Note that there is a strong analogy between mobility management in the packet-switched part of cellular networks and Mobile IP (cf. Section 4.4.1), i.e. between a MS and a MN, between an SGSN and a FA and between a GGSN and a HA. Note also that the IP address the MS acquires during PDP context establishment does not change upon handover.

When the MS moves to a new SGSN, it performs the link-layer handover before the network-layer handover, i.e. the mobility solution in the packet-switched part of cellular networks is reactive. In principle, the PDP context at the GGSN
4.4. IPV4-BASED MOBILITY

<table>
<thead>
<tr>
<th>MS</th>
<th>New SGSN</th>
<th>Previous SGSN</th>
<th>GGSN</th>
<th>HLR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RAU Request</td>
<td>SGSN Context Request</td>
<td>GGSN</td>
<td>HLR</td>
</tr>
<tr>
<td></td>
<td>GGSN</td>
<td>SGSN Context Response</td>
<td>HLR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SGSN Context Ack</td>
<td>IP Packets</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Update PDP Context Request</td>
<td>Update PDP Context Response</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Update Location</td>
<td>Cancel Location</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cancel Location Ack</td>
<td>Insert Subscriber Data</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Insert Subscriber Data Ack</td>
<td>Update Location Ack</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RAU Accept</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.22: Inter-SGSN Handover

has to be updated, a new PDP context at the new SGSN has to be established and the PDP context at the previous SGSN has to be deleted. The location at the HLR has to be updated as well. This is done by GMM, GTP and MAP and shown in Fig. 4.22. The handover procedure is also called Routing Area Update (RAU).

The procedure begins with the MS issuing a RAU Request to the new SGSN. The new SGSN then obtains the MS PDP context and further information from the previous SGSN via the SGSN Context Request, SGSN Context Response and SGSN Context Ack messages. At this point in time, packets currently stored and packets still arriving at the previous SGSN are forwarded over the tunnel also established via the context exchange messages to the new SGSN. Once the new SGSN has obtained all information for the MS, it updates the PDP context at the GGSN via Update PDP Context Request and Update PDP Context Response messages. The new SGSN also triggers the update of the MS location in the HLR. This is done via the Update Location and Update Location Ack messages. On receipt of the Update Location message, the HLR first cancels the
location information in the previous SGSN via the Cancel Location and Cancel Location Ack messages before it triggers the new SGSN to insert corresponding subscription information into a local database via the Insert Subscriber Data and Insert Subscriber Data Ack messages. Finally, the new SGSN confirms the RAU Request with a RAU Accept in order to complete the handover.

GMM suffers in particular from a long outage period since it is a reactive mobility solution and since the tunnel between previous and new SGSN has to be established during the handover. The main characteristics of GMM are summarized in Table 4.6.

<table>
<thead>
<tr>
<th>Predictive/Reactive</th>
<th>Reactive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intra-/Inter-technology</td>
<td>Intra-technology</td>
</tr>
<tr>
<td>Micro-/Macro-mobility</td>
<td>Micro-mobility</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>No</td>
</tr>
<tr>
<td>Data path between PoSs</td>
<td>Yes</td>
</tr>
<tr>
<td>Permanent user-specific entries</td>
<td>GGSN, SGSN</td>
</tr>
<tr>
<td>Temporary user-specific entries</td>
<td>old SGSN</td>
</tr>
<tr>
<td>IP addresses</td>
<td>Permanent intra-domain IP address</td>
</tr>
<tr>
<td>Main objective</td>
<td>Avoid changing IP addresses</td>
</tr>
<tr>
<td>Main disadvantage</td>
<td>Long outage period</td>
</tr>
</tbody>
</table>

Table 4.6: GMM – Summary

4.4.7 Hierarchical IPv4-based solutions

4.4.7.1 TeleMIP

Telecommunication Enhanced Mobile IP (TeleMIP) [46] is an extension to MIPv4. It introduces an intermediate hierarchy level with the Mobility Agent (MA). The MA is responsible for mobility management within the TeleMIP domain. The TeleMIP architecture is shown in Fig. 4.23.

When a MN with a TeleMIP implementation first enters a TeleMIP domain, it acts as described in Section 4.4.1 in order to acquire either a FA CoA or a CCoA. Along with this local CoA, it acquires another CoA from the FA in the FA CoA case or from the DHCP server in the CCoA case. This second CoA is an IP address of the MA. The MN then registers the local CoA with the MA and the second CoA with the HA in its home network. For both registrations the MN makes use of conventional MIPv4 signalling messages.

Whenever the MN moves to a new subnet within the TeleMIP domain, the FA changes while the MA remains the same. The MN then registers the newly received FA CoA with its MA without updating the registration with its HA. Only when the MN moves to a new subnet of another TeleMIP domain, the MA changes so that the registration with its HA has to be updated as well.
The benefits of this hierarchical approach apply for local movements, i.e. which remain within the same TeleMIP domain. The signalling delay of the MIPv4 RegReq is lower with TeleMIP than with MIPv4, thus reducing the outage period. Since buffering is neither performed in MIPv4 nor in TeleMIP, packet loss is reduced in the same way as the outage period. Since the registration messages remain local to the domain for local movements, the number of messages travelling beyond the domain are reduced considerably. Finally, as every micro-mobility approach, TeleMIP allows the use of private IP addresses within the domain. Note that these benefits come at the expense of possible routing inefficiencies since external IP packets are always routed to a particular MA no matter where within the TeleMIP domain the MN is currently located. Furthermore, TeleMIP still suffers from long outage periods since it is a reactive mobility solution and since the MN needs to acquire a new IP address at the new PoS. The main characteristics of TeleMIP are summarized in Table 4.7.
4.4.7.2 IDMP

The Intra-Domain Mobility Management Protocol (IDMP) [45] has been developed as extension to TeleMIP (cf. Section 4.4.7.1). In contrast to TeleMIP, which requires MIPv4 as mobility solution for inter-domain mobility, IDMP is designed as standalone solution and does not assume the use of MIPv4 for inter-domain mobility.

When a MN first enters an IDMP domain, it registers with a Subnet Agent. A Subnet Agent is a FA enhanced by a DHCP server, i.e. it may operate in both FA CoA mode and in CCoA mode. The CoA the MN acquires is called Local CoA (LCoA). Along with the LCoA the Subnet Agent provides the MN with the address of a Mobility Agent (MA). The MA has the same functionality as the MA introduced in TeleMIP (cf. Section 4.4.7.1). The MN then registers the acquired LCoA with this MA. During the registration with the MA, the MN acquires a Global CoA (GCoA). The MN then registers the acquired GCoA with its Correspondent Nodes (CNs), its HA or other entities. Note that IDMP does not explicitly require MIPv4 to be in place. However, when many CNs are involved in the communication with the MN, a registration with a MIPv4 HA might be much more efficient than a dedicated registration with each CN.

Packets sent by a CN are sent to the GCoA where they are intercepted by the MA and tunnelled to the current LCoA. If the LCoA is a FA CoA, the packets are intercepted by the where they are detunneled and delivered to the MN. If the LCoA is a CCoA, the packets are directly sent to the MN where they are detunneled and consumed.

When the MN moves to another Subnet Agent, it acquires a new LCoA. As the Subnet Agent learns that the MN has already been assigned a MA, it does not change this assignment. The MN then updates its registration with the MA in order to provide it with the new LCoA. However, since the MA does not change, it does not assign the MN a new GCoA. Hence, the MN does not have to register a new GCoA with its CNs or the HA as long as it moves within the same IDMP...
IDMP claims to enhance TeleMIP by offering a smoother handover. A link-layer trigger at the MN leads to the transmission of a MovementImminent message to the current MA. On receipt of this message, the MA starts multicasting all further packets to the current Subnet Agent and all Subnet Agents in its direct vicinity. IDMP requires each of these Subnet Agents to buffer all arriving packets in a user-specific buffer. As soon as the MN registers with one of these Subnet Agents, the Subnet Agent delivers all buffered packets to the MN. It remains open if and how the other Subnet Agents are informed of this registration and when they remove the packets from their buffers. Furthermore, the fact that the MN does not provide the MA with the address of the Subnet Agent it is moving to is presented as an advantage instead of a deficiency.

In order to reduce the number of location update messages in idle state, IDMP also introduces the concept of paging. Paging is not addressed by TeleMIP.

4.4.7.3 Micro-Mobile MIP

Micro-Mobile IP [48] has been designed as extension to Mobile IPv4 Regional Registration. Note, however, that the Internet-Draft specifying Mobile IPv4 Regional Registration as equivalent to Hierarchical Mobile IPv6 (cf. Section 4.5.2) has expired.

The objective has been to extend Mobile IPv4 Regional Registration with a proactive handover mechanism and with paging. The proactive handover mechanism is based on link-layer triggers and works as follows. When the MN receives a link-layer trigger indicating that a handover is imminent, it requests an Agent Advertisement from the target FA via the current FA and registers the CoA contained therein with its Gateway Foreign Agent (name of the TeleMIP MA in Mobile IPv4 Regional Registration). Along with this registration, the MN requests the Gateway Foreign Agent to buffer further packets in order to reduce packet loss and in order to avoid duplicated packets through bicasting or multicasting. The Gateway Foreign Agent is supposed to buffer the packets until it receives a notification from the MN of the arrival at the target FA.

4.4.7.4 Combinatorial MIP

Another approach introducing a second hierarchy level has been published as Combinatorial Mobile IP [41]. The main focus is, however, not on mobility management, but on location management. The objective is to reduce the number of signalling messages when the mobile device is in idle mode. This is achieved through paging, i.e. a concept taken from cellular networks. The architecture is as depicted in Fig. 4.24.

The entities involved are a Controlling Foreign Agent (CFA) and a Paging Foreign Agent (PFA). The PFA corresponds to a MIPv4 FA extended by a paging
function. The CFA is a newly added entity constituting the second hierarchy level apart from the HA. The CFA, too, is extended by a paging function. Paging requires the distinction of two modes. These are the active and the idle mode. A MN is in active mode as long as it sends or receives data packets. After a certain period of inactivity that is controlled by a timer, the mode switches to idle. The mode is stored both in the CFA, the PFA and in the MN itself.

On receipt of a data packet at the CFA, the CFA checks the mode of the MN. If the MN is in active mode, the stored PFA is up to date so that the data packet can be tunnelled to that PFA. Otherwise, the stored PFA might be obsolete so that the CFA has to page all PFAs within the Combinatorial MIP domain. The PFA to which the MN is currently attached pages the MN over the air interface. The result is signalled to the CFA. If the paging has been successful, the CFA stores the corresponding PFA as PFA for the MN and forwards all buffered data packets and those newly arriving for the MN to that PFA. Whenever the MN moves to a new PoA, i.e. to a new PFA, it sends a MIPv4 RegReq to the new PFA. In active mode, the PFA relays the MIPv4 RegReq to the CFA. In idle mode, the PFA does not relay the MIPv4 RegReq to the CFA so that the number of signalling messages within the Combinatorial MIP domain is reduced.
4.4.7.5 Dynamic Hierarchical MIP

The hierarchical approach pursued by [97] is different from the hierarchical approaches presented in Sections 4.4.7.1, 4.4.7.3 and 4.4.7.4 in that it does not introduce a further hierarchical level common to all users, i.e. in principle at the border of the mobility domain. Instead, it introduces a hierarchy specific to each user by registering a newly acquired CoA at the previous FA. Therefore, it is called Dynamic Hierarchical MIP. The main objective is to reduce the signalling traffic during handover not only beyond the mobility domain but also within the mobility domain and to distribute the processing overhead at the HA or the border router of the mobility domain among all FAs.

The start-up procedure is analogous to that of MIPv4. Whenever the MN moves to a new FA, it acquires a new CoA, i.e. either a FA CoA or a CCoA. While this CoA is registered with the HA in MIPv4 or with a border router of the mobility domain in the mobility solutions presented in Sections 4.4.7.1, 4.4.7.3 and 4.4.7.4, it is registered with the previous FA in Dynamic Hierarchical MIP (Fig. 4.25).

Since the registration with the HA is not updated, all further packets arrive at the previous FA where they are detunneled and retunneled to the new FA. The new FA finally delivers the packets to the MN. Since the arising chain of FAs prolongs the data path further and further, the number of FAs can be bounded...
Predictive/Reactive & Reactive \\
Intra-/Inter-technology & Both \\
Micro-/Macro-mobility & Micro-mobility \\
Hierarchical & No \\
Data path between PoSs & Yes \\
Permanent user-specific entries & HA, chain of FAs \\
Temporary user-specific entries & None \\
IP addresses & Changing CoA \\
Main objective & Reduce signalling traffic \\
Main disadvantage & Inefficient data transmission \\

Table 4.8: Dynamic Hierarchical MIP – Summary

above. As soon as the boundary is exceeded, the MN has to register the newly acquired CoA with its HA as in standard MIPv4. The main characteristics of Dynamic Hierarchical MIP are summarized in Table 4.8. A similar though more analytical approach is described in [26].

4.5 IPv6-based mobility

An extensive performance comparison of Mobile IPv6 (MIPv6), Hierarchical MIPv6 (HMIPv6) and Fast Handovers for MIPv6 (FMIPv6) has been performed by [128]. Different parameters such as handoff latency, packet loss and signalling load are evaluated with respect to the number of MNs. A scarce performance analysis of HMIPv6 and FMIPv6 focussing on WLAN environments has been performed by [108].

4.5.1 Mobile IPv6

Mobile IPv6 (MIPv6) is specified in [84]. There are many similarities to MIPv4. However, MIPv6 is integrated into IPv6 and provides many other improvements. The main difference with respect to MIPv4 are listed in the following.

- MIPv6 operates without Foreign Agents. A MN acquires a new IPv6 address either through stateful (DHCPv6) or stateless address autoconfiguration (cf. Section 2.4.4 and Section 2.4.3). In both cases, the address is a collocated address. The need for FA CoAs is not so prevailing in IPv6 since there is no scarcity of IPv6 addresses as in IPv4.

- When a MN is attached to a foreign network, packets sent to its fixed Home Address (HoA) are intercepted by the HA and tunnelled to its current location specified by its CoA. When the CoA has been registered with the CN itself, the CN can tunnel the packets directly to the MN without the
detour over the HA. In MIPv4, the original IPv4 packet is encapsulated into another IPv4 packet. Since IPv6 is based on extension headers, MIPv6 makes instead use of an IPv6 routing header containing the ultimate destination address (i.e., the MN HoA), thereby reducing the relative amount of overhead. Note that the absolute amount of overhead is significantly higher with respect to MIPv4 due to the length of IPv6 addresses.

Note also that encapsulation cannot be avoided when the HA receives the packets from the CN. In order to prevent intermediate routers on the path from HA to MN from discarding the packets due to ingress filtering rules, the HA would have to place its own IP address in the source IP header. Since the replaced source address (i.e., the CN address) cannot be placed into the routing header that needs to contain the MN HoA, the receiving MN would not know which CN the packet has originally come from. Therefore, encapsulation is mandatory when the HA receives the packets from the CN.

- Although MIPv4 has been designed to operate over any particular link-link technology, it has been based on ARP that is primarily used in IEEE-802-based environments. MIPv6 uses IPv6 Neighbor Discovery instead of ARP and thus increases its robustness.

The entities involved in MIPv6 are the Home Agent (HA), one or more Correspondent Nodes (CNs), the Mobile Node (MN) and a default router within each foreign network. The MN has a fixed Home Address (HoA) that is always reachable by the CNs whether or not the MN is in its home network. The MIPv6 architecture is depicted in Fig. 4.26. Note that the user and control plane protocol stacks equal those of MIPv4 when replacing MIPv4 by MIPv6 and IPv4 by IPv6.

The start-up procedure (cf. Fig. 4.27) is analogous to MIPv4. When a MN first enters a foreign network, it performs Router Discovery in order to obtain the IPv6 address of a default router and prefix information of the current subnet. It then performs either stateless or stateful address autoconfiguration in order to obtain a CoA within the foreign network. Finally, it registers the newly obtained CoA with its HA.

When the MN moves to another foreign network, it has to detect the movement. Several methods are available. When the MN is currently sending packets, it may use the Neighbor Unreachability Detection which is part of IPv6 Neighbor Discovery. Otherwise, it may wait for periodic Router Advertisements or send itself Router Solicitations periodically. If link-layer triggers are available, these should be used as indication. Once the MN has detected the movement, it performs Router Discovery and obtains a new CoA as described for the start-up procedure. Finally, it registers the newly obtained CoA with its HA. Optionally, it may also register the CoA with some or all of its CNs, resulting in two modes.
CHAPTER 4. MOBILITY MANAGEMENT

If the MN does not register its CoA with a CN, the mode is called Bidirectional Tunnelling. Otherwise, the mode is called Route Optimization.

Bidirectional Tunnelling does not require the CN to support MIPv6. The CN sends all IPv6 packets to the fixed HoA of the MN where they are intercepted by the HA and tunneled to the MN at its current location (cf. Fig. 4.26). The MN, on the other hand, sends all IPv6 packets to the HA via a reverse tunnel where they are received in order to be forwarded to the CN.

Route Optimization requires the CN to support MIPv6 and to store a binding for the MN. In this mode, packets from the CN to the MN and from the MN to the CN are sent directly without taking the detour to the HA. Route Optimization generally leads to shorter paths between MN and CN.
4.5. IPv6-BASED MOBILITY

In order to send packets on the direct way from CN to MN (cf. Fig. 4.28), IPv6 provides the IPv6 routing header. The ultimate IPv6 destination address is placed in the IPv6 routing header, while an intermediate destination address is placed in the IPv6 destination address field within the main IPv6 header.

An IPv6 packet constructed in this way is routed towards the node whose address has been placed in the IPv6 destination address field of the main IPv6 header. On receipt of the packet, the node examines the IPv6 routing header and replaces the IPv6 destination address by the IPv6 address contained in the IPv6 routing header. The main use of this functionality is for mobile nodes. When the MN is located in a foreign network, it is not directly reachable via its fixed HoA. By placing its CoA into the IPv6 destination address field of the main IPv6 header and the fixed HoA into the IPv6 routing header, an IPv6 packet reaches the MN at its current location. On receipt of the IPv6 packet, the MN evaluates the IPv6 routing header, finds one of its own IPv6 addresses and delivers the IPv6 packet internally.

Note that despite of its differences MIPv6 is very similar to MIPv4 and suffers basically from the same disadvantages such as high overhead, long outage periods, high packet loss and long signalling delays. The summary from Table 4.1 thus also applies here.
4.5.2 Hierarchical Mobile IPv6 Mobility Management

Hierarchical Mobile IPv6 (HMIPv6) [144] is an extension to Mobile IPv6 that aims at reducing the number of signalling messages during intra-domain handovers and at shortening the corresponding signalling delay. It issues from [36] also containing a short performance analysis. HMIPv6 introduces a new network element called Mobility Anchor Point (MAP) that can be considered as a local HA. The HMIPv6 architecture is depicted in Fig. 4.29.

When a MN enters a foreign HMIPv6 domain for the first time, it has to configure two CoAs, one On-link CoA (LCoA) that equals the MIPv6 CoA (cf. Section 4.5.1) and a Regional CoA (RCoA). The RCoA is an IPv6 address from the MAP subnet and is configured via stateless address autoconfiguration (cf. Section 2.4.3). HMIPv6 extends the Agent Advertisements that are used by MIPv6 by a new MAP option containing one of the routable IPv6 addresses of the MAP. The MN then extracts the first 64 bits and adds a 64-bit interface identifier (e.g. based on its MAC address) in order to obtain the RCoA. Hence, different MNs configure different RCoAs.

After configuring both the LCoA and the RCoA, the MN first registers the
4.5. IPV6-BASED MOBILITY

LCoA with the MAP (cf. Fig. 4.30). It uses the RCoA as identifier. On receipt of the Binding Update (BU), the MAP performs Duplicate Address Detection (cf. Section 2.4.3) before storing a binding between RCoA and LCoA. After registering the LCoA with the MAP, the MN registers the RCoA with the HA (cf. Fig. 4.30). The MN uses its Home Address (HoA) as identifier. The HA stores a binding between HoA and RCoA. Optionally, the MN may also register the RCoA with any CNs it might be communicating with.

Whenever a packet is sent to the MN’s HoA (cf. Fig. 4.29), the HA intercepts the packet and tunnels it to the RCoA for which it has stored a binding with the HoA. Since the MAP advertises reachability under the RCoA, it intercepts the packet addressed to RCoA, detunnels it and re-tunnels it to the LCoA for which it has stored a binding with the RCoA. On receipt of the encapsulated packet, the MN detunnels it and delivers the resulting packet internally to the HoA.

When moving to a new subnet, the MN configures a new LCoA and a new RCoA. If the RCoA changes, it has to perform the same registration procedure as depicted in Fig. 4.30. If, however, the RCoA does not change, which is the case as long as the MN moves across subnets within the same HMIPv6 domain, a local registration with the MAP is sufficient (cf. Fig. 4.31). Note that neither the HA nor any CNs are involved in the local registration. Movements within the HMIPv6 domain remain transparent for the HA and CNs.

Note that although the signalling delay for the location update during intra-domain handovers is reduced with respect to the signalling delay for the location update during inter-domain handovers, the outage period remains high. Furthermore, all packets arriving at the previous PoS during the outage period are lost. The main characteristics of HMIPv6 are summarized in Table 4.9.
4.5.3 Fast Handovers for Mobile IPv6

Fast Handovers for Mobile IPv6 (FMIPv6) [90, 89] mitigates the issue of long outage periods by moving the configuration of the new IP address from after the corresponding link-layer handover to before the link-layer handover and through the use of a bi-directional tunnel between the access routers concerned during the handover. FMIPv6 also minimizes the packet loss during the reduced outage period through buffering both at the old and the new PoS.

FMIPv6 is an extension to MIPv6. Therefore, the architecture and the start-up procedure are the same as in MIPv6 (cf. Section 4.5.1 as well as Fig. 4.26 and Fig. 4.27).

The extension refers to the operation during handover. FMIPv6 introduces a predictive mode and a reactive mode. It also introduces new options within the RtSolPr and PrRtAdv messages. Before a handover, a MN typically receives one or more link-layer broadcast frames specifying an identifier (e.g. the MAC address) of the corresponding PoA. The MN may then place this identifier within the New Access Point Link-Layer Address option into the RtSolPr message. On receipt of that message, the current PoS responds with a PrRtAdv. If the current
4.5. IPV6-BASED MOBILITY

PoS has subnet-specific information about the PoA for which these are requested, it places them within the corresponding options into the PrRtAdv message. The means by which PoSs exchange subnet-specific information for the PoAs within their administrative area is not specified in FMIPv6. The exchange of the extended RtSolPr and PrRtAdv messages allows the MN to configure the new CoA even before attaching to the new PoA. The time to configure the new CoA can thus completely be shifted from after to before the link-layer handover.

FMIPv6 is based on link-layer triggers. Link-layer triggers are recommended to be used to trigger messages on the current or on the new link. The RtSolPr is e.g., a message that can be sent on receipt of an appropriate link-layer trigger.

The Fast Binding Update (FBU) message is sent to initiate the handover and may itself be triggered by a corresponding link-layer trigger. The FBU message may be sent on the link of the current PoS or on the link of the new PoS.

If the FBU is sent on the link of the new PoS, the handover mode is reactive. If it is sent on the link of the current PoS, the MN may or may not receive the FBack message on the link of the current PoS. If the MN receives the FBack message on the link of the current PoS, the handover mode is predictive. If the MN does not receive the FBack message on the link of the current PoS, it cannot know whether or not the PAR has successfully processed the FBU message and thus needs to send the FBU message on the link of the new PoS. In that case, the handover mode is reactive, too.

Note that the FBU message is destined to the current PoS in both modes and that it contains the new CoA that the MN has configured previously.

The predictive mode is shown in Fig. 4.32. On receipt of the FBU message, the Previous Access Router (PAR) analyzes the source IPv6 address. Since it equals the Previous CoA (PCoA), the PAR determines that the FBU message has been sent on the PAR’s link and starts the predictive mode. It then establishes a Bidirectional Edge Tunnel (BET) with the NAR by exchanging Handover Initiate (HI) and Handover Acknowledgement (HAck) messages. The HI message contains among other information the PCoA and the NCoA of the MN. The NAR determines whether or not NCoA is valid, e.g., by performing Duplicate Address Detection. If NCoA is not valid, it assigns a valid NCoA and places it into the HAck message. Let the CoA eventually contained in the HAck message be denoted by NCoA’.

On receipt of the HAck message, the PAR establishes a binding between PCoA and NCoA’. It then sends a Fast Binding Acknowledgement (FBack) to the MN over both the PAR’s link and via the NAR over the NAR’s link. It includes NCoA’ in the FBack message. Further IPv6 packets arriving at the PAR are tunnelled to the NAR where they are buffered until arrival of the MN. As soon as the MN establishes link-layer connectivity with the nPoA, it sends a Fast Neighbor Advertisement (FNA) message to the NAR. On receipt of the FNA message, the NAR begins delivering all buffered IPv6 packets to the MN.

The reactive mode is shown in Fig. 4.33. Since the MN cannot know whether
or not the configured NCoA is valid, it should allow the NAR to process the FBU message destined to the PAR. Therefore, the FBU message should be encapsulated into a FNA message destined to the NAR. If the NAR then determines that the NCoA contained in the FBU is not valid, it replies with a Router Advertisement denying the use of the NCoA and optionally proposing a valid one. Otherwise, the NAR sends the inner FBU message to the PAR.

On receipt of the FBU message from the NAR, the PAR analyzes the source IPv6 address. Since it does not equal the PCoA of the MN, the PAR determines that the FBU message has been sent on the NAR’s link and starts the reactive mode. Optionally, it may start buffering packets for the MN. It then establishes the BET as described for the predictive mode. Note, however, that the NCoA has already been verified on receipt of the FNA message encapsulating the FBU message. After establishment of the BET, the PAR creates the binding between PCoA and NCoA and sends a FBack back to the MN via the NAR. Note that it is not necessary to send the FBack message over the PAR’s link since the PAR is performing the reactive mode. All buffered packets and those arriving for the MN are then tunnelled to the NAR where they can directly be delivered to the NCoA of the MN.

Note that in the case that the MN has sent the FBU message on the link of the PAR and not received the FBack message on that link, it needs to send the FBU message encapsulated in a FNA message on the link of the NAR. However, if the NAR determines that the tunnel between PAR and NAR is already in place, the inner FBU and the HI and HAck messages do not need to be exchanged. Instead, the NAR may directly respond with the FBack message.
After performing the Fast Binding Update of FMIPv6, the MN still has to update the registration at its HA and optionally at its Correspondent Nodes by exchanging MIPv6 Binding Update and Binding Acknowledgement messages. Since the Fast Binding Update procedure provides for packet forwarding from PAR to NAR, the MIPv6 Binding Update procedure is not time-critical anymore.

The main benefit is achieved by the predictive handover mode. However, the exchange of HI and HAck messages cannot be avoided since the NAR has to verify the NCoA. The verification itself (e.g., if DAD is performed) is also a time-consuming operation. Real-time packet-switched services are thus likely to be affected by FMIPv6 handovers. The application of FMIPv6 to WiFi is described in [101]. A performance analysis with respect to handover latency has been done by [109]. The application of FMIPv6 to networks based on IEEE 802.16 (cf. Section 6.2) is described in [82] and to cellular networks in [162]. The main characteristics of FMIPv6 are summarized in Table 4.10.

4.5.4 Combining HMIPv6 and FMIPv6

HMIPv6 and FMIPv6 can be combined in two ways. The first way is to deploy FMIPv6 with HMIPv6 instead of deploying FMIPv6 with MIPv6. The benefit with respect to FMIPv6 is that the time period for the location update and thus the time period during which data is transmitted inefficiently is reduced.
CHAPTER 4. MOBILITY MANAGEMENT

<table>
<thead>
<tr>
<th>Predictive/Reactive</th>
<th>Both</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intra-/Inter-technology</td>
<td>Both</td>
</tr>
<tr>
<td>Micro-/Macro-mobility</td>
<td>Micro-mobility</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>No</td>
</tr>
<tr>
<td>Data path between PoSs</td>
<td>Yes</td>
</tr>
<tr>
<td>Permanent user-specific entries</td>
<td>HA</td>
</tr>
<tr>
<td>Temporary user-specific entries</td>
<td>PAR</td>
</tr>
<tr>
<td>IP addresses</td>
<td>Changing CCoA</td>
</tr>
<tr>
<td>Main objective</td>
<td>Reduce outage period</td>
</tr>
<tr>
<td>Main disadvantage</td>
<td>Long outage period</td>
</tr>
</tbody>
</table>

Table 4.10: FMIPv6 – Summary

for intra-domain handovers. The benefit with respect to HMIPv6 is that all the advantages of FMIPv6 apply.

The second way is briefly described in [144]. There, FMIPv6 is executed between MN, MAP and NAR instead of between MN, PAR and NAR so that the BET is set up between MAP and NAR instead of between PAR and NAR. The difference to HMIPv6 is that the intra-domain location update message (here: FBU) is sent to the MAP before and not after the actual L2 handover. The difference to FMIPv6 is that the intra-domain location update is done at the MAP instead of at the PAR.

The benefit with respect to FMIPv6 is that data packets are always sent on the direct path between MAP and NAR, i.e. no inefficient packet transmission occurs during a handover. However, the outage period is longer than in plain FMIPv6 since the FBU message has to continue travelling from PAR to MAP and since the tunnel is set up between MAP and NAR instead of between PAR and NAR. The distance between MAP and NAR in general being much longer than the distance between PAR and NAR. Furthermore, while the FBU message travels from PAR to MAP, downstream data packets travelling from MAP to PAR may not be received by the MN anymore since the MN may already be involved in L2 handover and since the PAR is not responsible for buffering. This issue does not occur in FMIPv6 since the FBU message only travels from MN to PAR that is here responsible for buffering.

The benefit with respect to HMIPv6 is that the outage period is reduced by the L2 handover delay and the time period the FBU message is sent to the MAP before the L2 handover. In theory, the outage period could thus even be eliminated if the handover was really executed. But in general the handover cannot be predicted with reasonably high probability as much in advance. Therefore, the outage period can in general only be reduced by the L2 handover delay.

Overall, the main disadvantage is the long outage period that is only a little less than the outage period of HMIPv6.
4.5.5 Cellular Universal IP

The acquisition of a new IPv6 CoA when entering a new subnet is a time-consuming operation. Cellular Universal IP (CUIP) [92] is a mobility solution that avoids the acquisition of new CoAs both for intra-domain subnet changes and for inter-domain subnet changes. While the assumption of having a unique IPv6 address for a MN within a domain is a realistic one, the assumption of having a globally unique IPv6 address for a MN seems to be hardly achievable.

CUIP is based on host-specific routing table entries. These are at least necessary on all routers along the path from the MN's home network to its current location (cf. Fig. 4.34). If it is desired to have Correspondent Nodes send their packets directly to the MN (without the detour via the MN's home network), host-specific routing table entries are needed on the way from these Correspondent Nodes to the current location of the MN, too. When the MN changes its PoS (cf. Fig. 4.34), the path from its home network to its current location changes. In that case, all routers along the new path beginning with the new PoS up to the Cross-Over Router (COR) have to be updated with a new host-specific routing table entry. The COR is located at the nearest intersection (with respect to the MN) of the previous path and the new path. Nothing is said as to whether or not host-specific routing table entries are removed if they are no longer used. In order to keep processing overhead low, a new data structure called mobility routing table is designed.

Handover initiation is based on link-layer triggers. During handover the MN is assumed to always have data packets to send because the MN notifies the network of the location update through a new extension header added to data packets. The extension header is called CUIP Notification header. On receipt of a packet with this extension header, the receiving node updates its mobility routing table with a host-specific routing table entry for the MN. At some point in time, the IPv6 data packets arrive at the COR. The COR identifies itself as the COR if it already has an existing host-specific entry for the MN. It then removes the CUIP Notification header from the IPv6 data packets and forwards them in direction of their final destination. Finally, it sends an acknowledgement back to the MN so that the MN can stop adding the CUIP Notification header to outgoing data packets.

CUIP is an approach very similar to Cellular IP (cf. Section 4.4.5). The main difference is that Cellular IP has been designed for IPv4 while CUIP is designed for IPv6. Cellular IP does not specify the use of link-layer triggers for handover initiation since the host-specific routing table entries are updated on receipt of every data packet and not only on receipt of those containing a certain option or extension header (in IPv6). In contrast to CUIP, Cellular IP specifies the use of timers that are associated with the host-specific routing table entries so that entries that are no longer used time out and can be removed.
4.5.6 Edge Mobility Protocol

The Edge Mobility Protocol (EMP) [156] is a micro-mobility solution for use within an Edge Mobility Domain (EMD). The main components in the EMD are the Advanced Access Router (AAR) and the Edge Mobility Anchor Point (EMAP). EMP protocol messages are transported by the Stream Control Transmission Protocol (SCTP) [113] and are only sent between the AAR and the EMAP. EMP is based on IPv6 Neighbor Discovery (cf. Section 2.4.2) as trigger for the start-up and handover procedures. Although a predictive mode was conceivable, only a reactive mode is defined in [156].

The start-up procedure is depicted in Fig. 4.35. After link-layer association, the MN broadcasts a Router Solicitation message containing an ID for the MN (MNID). The MNID may be a unique MAC address, a MIP HoA, a NAI, etc. Since the AAR is not aware of whether or not the MN is already registered in the EMD, it sends an EMP Query message including the MNID to the EMAP configured in the AAR. It also sends a Router Advertisement message back to
4.5. IPV6-BASED MOBILITY

The MN.

On receipt of the EMP Query message, the EMAP searches its database of registered MNs within the EMD for an entry of the MN. When the MN is starting up, the EMAP does not have an entry and replies with an EMP Reply message only containing the MNID. Otherwise, the EMAP would include all CoAs of the MN into the EMP Reply message as well.

On receipt of the EMP Reply message, the AAR does not need to take any action since the MN still has to configure/acquire a CoA. Once the MN has configured/acquired a CoA and performs Duplicate Address Detection (DAD) by sending a corresponding Neighbor Solicitation message, the AAR sends an EMP Update message containing the MNID and the CoA to the EMAP. The EMAP checks whether or not the CoA is already registered and places a corresponding status code into the EMP Reply message that it sends to the AAR. It also registers an entry for the MN. On receipt of an EMP Reply message stating that the MN has successfully been registered at the EMAP, the AAR creates a host-specific routing table entry and registers an EMD entry for the MN. In order to complete the start-up procedure, the MN sends a Binding Update with the newly acquired CoA to its HA. The Binding Ack message sent in response accomplishes the start-up procedure.

IP packets arriving at the EMAP are tunnelled to the current AAR. EMP may negotiate any tunneling method such as IPv6 tunnelling, GRE or MPLS. IPv4 tunneling or MRE may not be used since the EMD is an IPv6 domain.

In case of a handover from an old AAR (oAAR) to a new AAR (nAAR), the MN first sends a Router Solicitation message to the nAAR (cf. Fig. 4.36). The nAAR replies with a Router Advertisement message. As in the start-up procedure, the nAAR is not aware of whether or not the MN is already registered in the EMD and thus sends an EMP Query message containing the MNID to the
EMAP. The EMAP searches its database of registered MNs and finds an entry for the MN. It therefore sends an EMP Reply message containing the MNID and the registered CoA back to the nAAR. On receipt of the EMP Reply message containing the CoA, the nAAR adds a host-specific entry into its routing table. It also creates an EMD entry for the MN.

The EMAP also sends an EMP Update message containing the Delete option to the oAAR. On receipt of this Update message, the oAAR removes the host-specific routing table entry and the EMD entry of the MN.

Finally, the MN sends a Binding Update message to the HA that responds with a corresponding Binding Ack message. The Binding Ack message accomplishes the handover procedure.

Although EMP shows that intra-domain mobility can be solved by simply updating the location at the domain ingress node (here: the EMAP), it does not address the long outage period or packet loss during handover. The main characteristics of EMP are summarized in Table 4.11.

Table 4.11: EMP – Summary

<table>
<thead>
<tr>
<th>Predictive/Reactive</th>
<th>Reactive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intra-/Inter-technology</td>
<td>Both</td>
</tr>
<tr>
<td>Micro-/Macro-mobility</td>
<td>Micro-mobility</td>
</tr>
<tr>
<td>Hierarchical</td>
<td>Yes</td>
</tr>
<tr>
<td>Data path between PoSs</td>
<td>No</td>
</tr>
<tr>
<td>Permanent user-specific entries</td>
<td>EMAP, AAR</td>
</tr>
<tr>
<td>Temporary user-specific entries</td>
<td>None</td>
</tr>
<tr>
<td>IP addresses</td>
<td>Changing CCoA</td>
</tr>
<tr>
<td>Main objective</td>
<td>None</td>
</tr>
<tr>
<td>Main disadvantage</td>
<td>Long outage period</td>
</tr>
</tbody>
</table>

Figure 4.36: Handover procedure of the Edge Mobility Protocol
4.6 MPLS-based mobility

4.6.1 Solutions only updating the Ingress LER

Many MPLS-based mobility management solutions such as [130, 158, 94, 151, 152] propose the combination of MPLS and MIPv4 or MPLS and HMIPv6. While MIPv4 and HMIPv6 are exploited to provide the signalling necessary for mobility management, MPLS is exploited instead of IP encapsulation for data delivery on the user plane. Since the cited mobility solutions are based on the signalling of MIPv4 and HMIPv6, they are all reactive solutions. The main objective is to reduce the overhead of IP encapsulation. Some papers also claim that MPLS label switching is much faster than IP routing. However, [121] has performed an extensive performance analysis showing that this does not hold in general.

In [130], MPLS is combined with MIPv4. The start-up procedure is depicted in Fig. 4.37. Note that the same procedure is executed during network-layer handover. Apart from reducing the overhead of IP encapsulation, the proposed solution does not intend to reduce the signalling delay, the packet loss or the outage period during handover. As soon as the Home Agent is notified of the new location of the MN, it simply updates the FTN to place packets destined to the MN’s HoA on the LSP to the new location of the MN.

Therefore, [158] and [94] propose an enhancement by introducing a new component called Foreign Domain Agent in [158] and LER/GW in [94], both being similar to the Mobility Anchor Point (MAP) in Hierarchical Mobile IPv6 (cf. Section 4.5.2). The objective is to reduce the signalling delay for the location update during an intra-domain handover.

The architecture is depicted in Fig. 4.38. When first entering the domain, the start-up procedure basically works as in [130]. However, when moving between subnets within the same domain, the shorter signalling delay takes effect since the location update remains local to the domain. A further advantage is that an
LSP does not have to extend from the HA over possibly one or more domains to the foreign domain the MN is currently located at. Instead, there is one LSP from the HA to the Foreign Domain Agent or the LER/GW and another one from the Foreign Domain Agent or the LER/GW to the FA.

While [130], [158] and [94] propose to combine MPLS and MIPv4, [151] proposes to combine MPLS and HMIPv6. There, too, no single LSP exists between the HA and the default router of the MN. [152] extends that approach to inter-domain and thus macro-mobility. As the total path between HA and MN or Correspondent Node and MN is assumed to be composed of several single LSPs, the main focus is on avoiding the time-consuming layer change when forwarding data packets at intermediate LERs or MAPs.

All mentioned solutions suffer from similarly long signalling delays as the IP-based handover solutions elucidated in Section 4.4 and in Section 4.5 because packets are not rerouted before the ingress LER is notified of the handover. Since the packets are not buffered at the previous PoS, packet loss is the main disadvantage of MPLS-based mobility solutions only updating the ingress LER.
4.6. MPLS-BASED MOBILITY

4.6.2 LEMA

A different approach introduces the concept of a Label Edge Mobility Agent (LEMA). A LEMA is an LER enhanced by a function for mobility management. The LEMA concept is first described in [40] and put in a larger context in [39]. A very similar approach is described in [138] and is thus not further described in this paper. The objective of deploying MPLS is the same as described in Section 4.6.1. The objective of deploying LEMAs is to further reduce the long signalling delay between FA and HA and between FA and CNs.

The architecture is depicted in Fig. 4.39. The LEMA domain contains a set of LEMAs that may be distributed arbitrarily within the domain. An algorithm to compute such a set is provided by [137] that is closely related to [138]. The only restriction is that all access routers are LEMAs. These LEMAs are called access LEMAs. The LEMAs form an overlay network whose edges are pre-established LSPs. Each LSP can be used for more than one user.

Each access LEMA advertises a certain topology of LEMAs with itself as root. The MN receiving the advertisement chooses a chain of LEMAs to register with.
It then sends a registration message containing an identifier for itself and the chosen chain of LEMAs to the access LEMA. The access LEMA removes itself from the chain of LEMAs and relays the registration message to the next LEMA within the chain. Each of the subsequent LEMAs receiving the registration message associates the identifier for the MN with the FEC of the LSP leading towards the LEMA it has received the registration message from. It then removes itself from the chain of LEMAs and stores the remaining chain of LEMAs together with the identifier for the MN. Finally, it relays the registration message to the next LEMA within the chain. The last LEMA within the chain registers one of its own IP addresses as the MN CoA with the MN’s HA and the MN’s CNs. Therefore, the address of the HA and the addresses of all CNs have to be included in the registration message, too. An example start-up procedure is shown in Fig. 4.40 for a chain of two LEMAs including the access LEMA.

When the MN moves into the area of another access LEMA, it detects movement through the different LEMA topology advertised by the new access LEMA. The MN then computes a new registration chain, possibly by taking the previous registration chain into account, and sends a registration message specifying the identifier for itself, the registration chain, its previous LEMA, the address of its HA and the addresses of all its CNs to the new access LEMA. On receipt of the registration message, the new access LEMA sends a redirection message to the previous access LEMA. The purpose of that redirection message is to release the MN context. On receipt of a redirection confirmation, the new access LEMA sends a redirection message to the next LEMA within the chain. If the receiving LEMA already has an entry for the MN, it sends a registration release message to the stored downstream LEMA to remove the MN entry there. If the remainder of the chain is different from the remainder of the chain stored for the MN, it forwards the redirection message further upstream. Otherwise, the redirection procedure terminates. Fig. 4.41 shows the handover procedure at an example.

Note that a packet sent from HA to MN arrives at the topmost LEMA from
where it is possibly placed on several LSPs one after the other requiring a time-consuming re-classification into a FEC at each LEMA. In one extreme, only two LEMA levels exist, the access LEMAs constituting one level and the gateway LEMA the other level. That approach resembles the reactive mode of FMIPv6 with the difference that the MN is not involved in a separate MIPv6 registration with its HA and CNs. This task is taken by the gateway LEMA. In the other extreme, all nodes are LEMAs. In that case, every LSP is only two hops long, and the time-consuming re-classification of IP packets has to be done at every hop. The advantage of this extreme is that packets are rerouted before the location is updated at the gateway of the MPLS domain. In that respect, the LEMA approach reduces the outage period for an intra-domain handover. However, the time-consuming re-classification is necessary for all packets arriving at a LEMA so that the delay of an IP packet from the gateway to the MN is considerably increased. In practice, an operator would certainly deploy a set of LEMA nodes that lies between these two extremes.

The main disadvantage of the LEMA approach is that a time- and resource-consuming re-classification of all packets arriving at a LEMA is necessary. The negative impact on user plane processing is considerable since classification is a much more time-consuming operation than IP routing or MPLS label switching (cf. [121]). The benefit of a reduced outage period during intra-domain handovers is thus eliminated with additional LEMAs. The main characteristics of the LEMA approach are summarized in Table 4.12.
CHAPTER 4. MOBILITY MANAGEMENT

Predictive/Reactive | Reactive
---|---
Intra-/Inter-technology | Both
Micro-/Macro-mobility | Micro-mobility
Hierarchical | No
Data path between PoSs | No
Permanent user-specific entries | LEMA chain
Temporary user-specific entries | old LEMA chain
IP addresses | Permanent CoA
Main objective | Reduce outage period
Main disadvantage | Long outage period

Table 4.12: LEMA – Summary

<table>
<thead>
<tr>
<th>Mobility solution</th>
<th>Main disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIPv4 / Proxy MIPv4</td>
<td>Long outage period</td>
</tr>
<tr>
<td>Low Latency Handoffs in Mobile IPv4</td>
<td>Long outage period</td>
</tr>
<tr>
<td>HAWAII / Cellular IP</td>
<td>Scalability issues</td>
</tr>
<tr>
<td>GMM / TeleMIP</td>
<td>Long outage period</td>
</tr>
<tr>
<td>Dynamic Hierarchical MIP</td>
<td>Inefficient data transmission</td>
</tr>
<tr>
<td>MIPv6 / HMIPv6 / FMIPv6</td>
<td>Long outage period</td>
</tr>
<tr>
<td>Combined HMIPv6 and FMIPv6</td>
<td>Long outage period</td>
</tr>
<tr>
<td>Cellular Universal IP</td>
<td>Scalability issues</td>
</tr>
<tr>
<td>Edge Mobility Protocol</td>
<td>Long outage period</td>
</tr>
<tr>
<td>Solutions only updating the ingress LER</td>
<td>Packet loss</td>
</tr>
<tr>
<td>LEMA</td>
<td>Re-classifications</td>
</tr>
</tbody>
</table>

Table 4.13: Summary

4.7 Summary

The main disadvantages of the most important mobility solutions presented in Sections 4.4, 4.5 and 4.6 are given in Table 4.13. These disadvantages have been the main incentive for the development of Active MPLS that is introduced and evaluated in Chapter 5.
Chapter 5

Active MPLS

Active MPLS [120] is a novel MPLS-based micro-mobility solution. The incentive for the development of Active MPLS has been the shortcoming of the existing MPLS-based and IP-based micro-mobility solutions in achieving seamless handovers for real-time packet-switched services. The main disadvantages of the existing MPLS-based and IP-based micro-mobility solutions have already been stated in Sections 4.4, 4.5 and 4.6. Active MPLS avoids all these disadvantages.

Active MPLS outperforms all existing IP-based micro-mobility solutions for the following reasons. A mobile device does not need to acquire or configure a new IP address during intra-domain handovers, thus eliminating the time-consuming acquisition of a new IP address as well as Duplicate Address Detection. The overhead is reduced with respect to IP-based solutions applying IP tunnelling since Active MPLS makes use of only two MPLS shim headers of 4 Byte each. Unlike some IP-based approaches, Active MPLS is scalable with respect to the number of permanent user-specific entries within the domain. Active MPLS does also not need to establish any host-specific paths or tunnels during an intra-domain handover. All paths in Active MPLS (i.e. LSPs) can be pre-established. Consequently, the outage period and thus packet loss is reduced with respect to many other approaches.

Active MPLS outperforms all existing MPLS-based micro-mobility solutions for the following reasons. Active MPLS does not make use of time- and resource consuming re-classifications of IP packets during the handover, thus reducing the overhead within the domain as well as the end-to-end delay of user IP packets. It also does not need to establish any LSPs during the handover as all LSPs may be pre-established. Since Active MPLS is a pure MPLS-based approach, a mobile device does not need to acquire a new IP address during intra-domain handovers, thus eliminating the time-consuming acquisition of a new IP address as well as Duplicate Address Detection.

The main ideas of Active MPLS are the use of a label stack of two labels, the use of a Handover ILM (HO ILM) and a Handover FTN (HO FTN) as well as the use of primary LSPs and handover LSPs. Note that the HO FTN is not used in
the context of re-classification. It is named after the FTN at a standard MPLS ingress LER because its structure is the same. The HO ILM does not necessarily constitute a separate database. It may instead be implemented as part of the ILM. It is here used as a separate database for clarity.

In this dissertation, primary LSPs denote LSPs as defined by [134]. As described in Section 2.5, the first node on a primary LSP is denoted as ingress LER, the intermediate nodes are denoted as LSRs and the last node is denoted as egress LER. Handover LSPs extend from an LSR or the egress LER of a primary LSP, over zero or more intermediate MPLS-capable nodes to the egress LER of another primary LSP. Note that the first and last nodes on both primary and handover LSPs need to support Active MPLS. All other nodes on handover and primary LSPs may be conventional LSRs. Note also that the first node on a handover LSP stores the handover LSP entry in the HO FTN and not in an FTN as a standard MPLS ingress LER would do it. Finally note that Active MPLS exploits the capability of MPLS to establish LSPs a priori.

The two labels are used for different purposes. One label is used as identification of the traffic aggregate placed onto a primary LSP. The use of this label allows for pre-establishment of all primary LSPs since a single primary LSP may be set up for aggregates of arbitrary granularity – independently of which users the traffic transported over that primary LSP issues from. The second label is used as identification of a particular user. Although an MPLS packet always carries both labels, the user label only comes into play during a handover from one PoS to another. As a handover is a user-specific operation, a user-specific treatment cannot be avoided.

Before the handover from the previous PoS to the new PoS, the traffic of the considered user is sent along the primary LSP towards the previous PoS as part of a traffic aggregate. During the handover it is successively and temporarily placed on handover LSPs leading towards the new PoS and thus separated from the aggregate. This is made possible through the user label that identifies the traffic of the user performing the handover. After the handover, the traffic of the considered user is placed onto the primary LSP towards the new PoS and thus re-inserted into a traffic aggregate, i.e. into the traffic aggregate sent along the new primary LSP.

Note that the LSRs along the previous primary LSP may be configured to redirect the user traffic during intra-domain handovers or not to redirect the user traffic during intra-domain handovers. This leads to basically two variants. Active MPLS denotes the approach where all LSRs along the previous primary LSP are configured to redirect the user traffic during intra-domain handovers, while Fast Active MPLS denotes the approach where none of the LSRs is configured this way. In particular, Fast Active MPLS can be viewed as a special case of Active MPLS. Of course, other approaches between these two extremes are conceivable and may be beneficial according to the type of network where Active MPLS is deployed (cf. discussion in Section 5.6.1).
The user plane of Active MPLS is described in Section 5.1 while the control plane is described in Section 5.2. Since Fast Active MPLS can be viewed as a special case of Active MPLS, it does not have to be explained separately. Nevertheless, a detailed analysis of the difference between Active MPLS and Fast Active MPLS is done in Section 5.6.1. On receipt of a packet, the packet is either handed to the user plane or to the control plane. An MPLS packet is always handed to the user plane. An IP packet that does not contain control plane information is also handed to the user plane. Otherwise, e.g. in the case of an RSVP-TE packet, it is handed to the control plane.

Although Active MPLS avoids all the above-mentioned disadvantages, it requires certain MPLS-capable nodes to support Active MPLS and thus a modification of standard software. The necessary changes are, however, of only minor nature. A further drawback of Active MPLS with respect to other MPLS-based approaches is that two labels are used instead of one. The increase in byte overhead can, however, be considered as small, in particular in core networks where the link bandwidths are rather high.

In the following, Active MPLS is sometimes abbreviated as A-MPLS.

## 5.1 User Plane

The user plane functions describe the behavior of A-MPLS-capable nodes on receipt of user plane IP and MPLS packets. In principle, an A-MPLS-capable node hands an incoming IP packet to its ingress LER function and an incoming MPLS packet to its LSR function. The protocol stack of the user plane looks as depicted in Fig. 5.1.

### 5.1.1 Ingress LER Function

On receipt of a user plane IP packet, the ingress LER function of the A-MPLS-capable node performs the classification described in Section 2.5. In standard MPLS, the classification result is the FEC. In A-MPLS, the classification result is both the FEC and the User ID (UID) of the mobile device (cf. Fig. 5.2). A

<table>
<thead>
<tr>
<th>LSP ingress</th>
<th>Intermediate MPLS node</th>
<th>LSP egress</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP</td>
<td>MPLS</td>
<td>IP</td>
</tr>
<tr>
<td>L2</td>
<td>L2</td>
<td>L2</td>
</tr>
<tr>
<td>L1</td>
<td>L1</td>
<td>L1</td>
</tr>
</tbody>
</table>
112

CHAPTER 5.ACTIVE MPLS

IP packet
Classification
FEC
UID

Figure 5.2: Classification

<table>
<thead>
<tr>
<th>Outer MPLS shim header</th>
<th>Inner MPLS shim header</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outgoing Label</td>
<td>According to DSCP</td>
</tr>
<tr>
<td>Label</td>
<td>Exp</td>
</tr>
</tbody>
</table>

Figure 5.3: The two MPLS shim headers in Active MPLS

UID is already deployed in cellular systems and in WiMAX environments. In WiFi environments, however, a UID is not separately assigned. There, the MAC address of the mobile device could be exploited for the UID. The mobile device could e.g. provide its MAC address during authentication (cf. Section 6.1).

The FEC is used to retrieve the NHLFE from the FTN (cf. Section 2.5). The NHLFE contains the outgoing label, possibly a DiffServ Codepoint (DSCP) and the outgoing interface to the next hop. In standard MPLS, the original IP packet is prepended with an MPLS shim header containing the outgoing label retrieved from the NHLFE, the Exp bits set according to the DiffServ Codepoint and operator policy and the stack bit set to 1 (cf. Section 2.5). In A-MPLS, the original IP packet is prepended with two MPLS shim headers. The outer MPLS shim header is constructed in the same way as in standard MPLS. Since it is the outer MPLS shim header, the Stack bit is set to 0. The inner MPLS shim header contains the UID as label. The Exp bits are set to 0, the stack bit is set to 1 (cf. Fig. 5.3).

The MPLS packet is finally handed to the link layer associated with the outgoing interface retrieved from the NHLFE. From there, it is sent to the next hop also retrieved from the NHLFE.

5.1.2 LSR Function

On receipt of an MPLS packet, the LSR function of the A-MPLS-capable node extracts the UID from the inner MPLS shim header unless it is configured to behave as a standard MPLS-capable node. This behavior is different from standard MPLS and constitutes the modification required by Active MPLS. The UID is looked for in the HO ILM (cf. Fig. 5.4). If the UID is found there, the mobile device identified by the UID is currently performing a handover. The NHLFE is retrieved from the HO ILM where it is stored together with the UID. If configured as a standard MPLS-capable node, the LSR function does not evaluate the inner MPLS shim header and proceeds as if the UID was not found in the HO ILM.
5.1. USER PLANE

If the UID is not found in the HO ILM, the mobile device is not currently performing a handover. In that case, the incoming label is extracted from the outer MPLS shim header and taken as index into the ILM. If it is not found there, the MPLS packet is silently discarded. If it is found there, the NHLFE stored together with the incoming label is retrieved.

In both cases, the NHLFE yields the outgoing label, possibly a DiffServ Codepoint and the outgoing interface to the next hop. If the outgoing label equals the IMPLICIT_NULL or the EXPLICIT_NULL label [133], the MPLS packet is handed to the egress LER function. Otherwise, the outgoing MPLS packet is constructed as follows (cf. Fig. 5.5). The outer MPLS shim header is modified by substituting the incoming label for the outgoing label retrieved from the NHLFE. The Exp bits remain in general unchanged. However, local policy may require substituting the incoming Exp bits for the outgoing Exp bits retrieved from the NHLFE. The inner MPLS shim header containing the UID as label is not changed.

The resulting MPLS packet is finally handed to the link layer associated with the outgoing interface retrieved from the NHLFE. From there, it is sent to the next hop also retrieved from the NHLFE.

5.1.3 Egress LER Function

On receipt of an MPLS packet for which the LSR function has already determined an outgoing label equal to the IMPLICIT_NULL or the EXPLICIT_NULL label, the egress LER function removes both MPLS shim headers from the IP packet.
The IMPLICIT_NULL label is an indication that the outgoing interface and the next hop have to be retrieved from the NHLFE, whereas the EXPLICIT_NULL label is an indication that both the outgoing interface and the next hop have to be retrieved by other means, e.g. by a routing table lookup. Once both outgoing interface and next hop are determined, the plain IP packet can be forwarded.

Note that for a proper operation of Active MPLS, at least the LSP ingress and the LSP egress have to be A-MPLS-capable nodes. Note that this is the special case denoted as Fast Active MPLS. The LSRs may either be A-MPLS-capable nodes or standard MPLS-capable nodes. If an LSR is an A-MPLS-capable node, it may still be configured to behave as a standard MPLS-capable node.

5.2 Control Plane

LSPs can be set up manually or by using a signalling protocol such as LDP or RSVP-TE. As Traffic Engineering is one of the advantages of MPLS and as explicit LSPs cannot be set up with LDP, RSVP-TE should be deployed unless the core network is small enough to allow for manual configuration.

RSVP-TE (cf. Section 2.5.6) is an extension to RSVP. Explicit LSPs are set up with the help of the Explicit Route Object (ERO).

In the core network, it is proposed to use RSVP-TE to set up primary and handover LSPs (cf. Section 5.2.2 and Section 5.2.3) and to transport HO Notification and HO Notification Ack messages (cf. Section 5.2.7).

As RSVP-TE is run over the User Datagram Protocol (UDP) and IP, the protocol stack of the control plane looks as depicted in Fig. 5.6.

5.2.1 LSP Set-Up

In order to support mobility with Active MPLS, LSPs are mandatory for downstream data traffic. For upstream data traffic, LSPs may be used, but are not mandatory since the domain egress remains the same during a data session. If
5.2. CONTROL PLANE

LSPs are used both for downstream and for upstream data traffic, they may use the same or different paths. Although the same paths are generally used, different paths may occur in domains where Traffic Engineering is deployed.

Unless the egress LER is the final destination of an IP packet, the egress LER has to forward the IP packet in direction of its final destination. In general, an LSP carries IP packets with different IP destination addresses. If the egress LER has more than one outgoing interface, some of the IP packets taken off the LSP may have to be forwarded over one outgoing interface while others may have to be forwarded over another outgoing interface. The outgoing interface is thus not necessarily the same for all IP packets carried by the LSP so that it cannot be stored in the ILM of the egress LER.

In those cases, however, where the outgoing interface is the same for all IP packets taken off the LSP, the outgoing interface may be stored in the ILM of the egress LER. If more than one outgoing interface exists at the egress LER, a separate LSP can be set up for each outgoing interface. However, if the ingress LER is not aware of which outgoing interface the egress LER is choosing for a certain IP packet, it cannot put the IP packet onto the right LSP. Only if the ingress LER is aware of this information, it is reasonable to set up distinct LSPs for each outgoing interface at the egress LER.

According to how an LSP is set up, two modes can be distinguished. If an LSP is set up in such a way that the outgoing interface is not stored in the ILM at the egress LER, the mode is called per-node-set-up. As one of the objectives of A-MPLS is to avoid any change of IP address of the mobile device (even across subnets), host-specific routing table entries would be needed at the egress LER for such an LSP. Otherwise, i.e. if the outgoing interface is stored in the ILM at the egress LER, the mode is called per-interface-set-up. For such an LSP no host-specific routing table entries are necessary. Note that the set-up mode is associated with an LSP. Two different LSPs can be set up with different modes. Standard MPLS [134] uses the per-node-set-up mode.

An LSP can be set up in per-interface-set-up mode by providing a new object in an RSVP-TE PATH message. The object may be called INTERFACE object and may contain an identifier for the interface at the egress LER.

Since the set-up mode is a characteristic of an LSP, LSPs set up in per-node-set-up mode and LSPs set up in per-interface set-up mode can coexist in an A-MPLS domain. However, there are some cases where such a coexistence cannot be recommended. The following discussion serves to shed some light on coexistence issues.

In general, only an ingress LER of an LSP is aware of which user plane IP packets are sent along the LSP. An egress LER that terminates at least one LSP set up in per-node-set-up mode, be it a primary LSP or a HO LSP, therefore has to keep host-specific routing table entries for all mobile devices registered with it. An egress LER that terminates only LSPs set up in per-interface-set-up mode does not need any host-specific routing table entries. Therefore, it is strongly
recommended that all LSPs terminating at a certain MPLS-capable node be set up in the same mode.

In order to avoid host-specific routing table entries at an egress LER, all LSPs terminating at that egress LER have to be set up in per-interface-set-up mode. Note that this is only possible if it can be guaranteed that the receiving host is definitively located in the subnet associated with the interface for which the corresponding LSP has been set up.

The advantages of the per-interface-set-up mode are that the egress LER only performs one look-up (as in PHP) and that no routing table look-up and thus no host-specific routing table entries are needed. The disadvantages are that RSVP-TE needs to be extended and that different LSPs need to be set up for different outgoing interfaces at the egress LER. However, the number of outgoing interfaces at an edge node in an A-MPLS domain is assumed to be small so that the number of LSPs does not increase considerably. Also, the extension to RSVP-TE is of only minor nature.

Note that Active MPLS works with both per-interface-set-up mode and with per-node-set-up mode.

5.2.2 Set-Up of Primary LSPs

In an A-MPLS domain, primary LSPs are set up between each pair of edge MPLS-capable nodes.

In per-node-set-up mode, an LSP is set up from one edge MPLS-capable node, the ingress LER, to another edge MPLS-capable node, the egress LER. The FEC retrieved through classification may be chosen in such a way that it reflects the ID or IP address of the egress LER.

In per-interface-set-up mode, a primary LSP is set up from one edge MPLS-capable node, the ingress LER, to another edge MPLS-capable node, the egress LER, and associated with a certain outgoing interface at the egress LER. The outgoing interface leads to an access network (cf. Section 2.1). Note that the incoming interface at the ingress LER has no relevance. The FEC retrieved through classification needs to uniquely identify the LSP. If it can be guaranteed that each access network is attached to only one edge MPLS-capable node, the FEC may be chosen as the Network ID of the access network. Otherwise, the FEC may be chosen as a combination of an identifier for the edge MPLS-capable node and the Network ID of the access network.

5.2.3 Set-Up of Handover LSPs

In an A-MPLS domain, one or more handover LSPs may be set up for a primary LSP. The ingress node of a particular handover LSP may be any non-ingress node along the primary LSP. The egress node of that handover LSP is one of those
edge MPLS-capable nodes to which a mobile device may perform a handover to when registered with the egress node of the considered primary LSP.

The set-up of HO LSPs is done in the same way as the set-up of primary LSPs. It is based on RSVP-TE and can be done in either per-interface-set-up mode or in per-node-set-up mode.

Note that the only difference to the set-up of primary LSPs is that the ingress of the HO LSP does not create an entry in its FTN but in its HO FTN (cf. Fig. 5.7). All other nodes behave as for primary LSPs.

The first column of the HO FTN contains the FEC that is chosen according to the set-up mode of the LSP. Note that it will not be necessary to perform a time- and resource-consuming re-classification of IP packets in order to find out the FEC and the UID.

### 5.2.4 Handover Indication

Whenever a mobile device moves from one access network to another, a L3 handover becomes necessary. The L3 handover is either mobile-initiated or network-initiated. Note that Active MPLS works with both handover types. If the handover is network-initiated, the previous PoS may directly start the Active MPLS handover procedure and notify the mobile device of the handover. If the handover is mobile-initiated, the network needs to be notified of the handover. The mobile device thus needs to send a **HO Indication** message containing a HO sequence number or a time stamp in order to distinguish different **HO Indications** from one another, the Network ID (NID) of the access network the mobile device has previously been attached with (old NID) and the NID of the access network the mobile device is going to be attached with (new NID).

The **HO Indication** can be sent before the L2 handover or after the L2 handover. If it is sent before the L2 handover, it triggers the predictive mode of A-MPLS. The PoS receiving the **HO Indication** knows that it is the previous PoS because the mobile device includes the old NID in the **HO Indication**. The previous PoS sends a **HO Indication Ack** back to the mobile device and starts the Active MPLS handover procedure. Note that it may happen that the mobile device does not receive the **HO Indication Ack**, e.g. when it has already left the range of the previous access network. Since in this case the mobile device cannot be sure to have triggered the predictive mode of Active MPLS, it needs to trigger...
CHAPTER 5. ACTIVE MPLS

the reactive mode of Active MPLS as well by sending a HO Indication message to the new PoS after the L2 handover. Note that the use of the HO sequence number contained in the HO Indication message prevents the Active MPLS handover procedure to be executed twice.

If the HO Indication is sent after the L2 handover to the new PoS, the mobile device triggers the reactive mode of Active MPLS. The PoS receiving the HO Indication knows that it is the new PoS because the mobile device includes the new NID in the HO Indication. The new PoS cannot start the Active MPLS handover procedure because it does not know over which LSPs the mobile node has been receiving packets at the previous PoS. Therefore, the new PoS has to send a network-initiated HO Indication to the previous PoS. The IP address of the previous PoS can be determined from the old NID. On receipt of the network-initiated HO Indication, the previous PoS first responds with a HO Indication Ack in order to acknowledge receipt of the HO Indication. It then starts the Active MPLS handover procedure.

5.2.5 Intra-PoS HO

In the case of an intra-PoS HO, the outgoing interface changes at the PoS, while the PoS itself remains the same. The data the mobile device receives at the previous access network may originate from several LSPs. For the following explanations, let \( I_{\text{old}} \) be the old interface at the PoS, i.e. the interface via which the mobile device is reached before the intra-PoS HO, and \( I_{\text{new}} \) be the new interface at the PoS, i.e. the interface via which the mobile device will be reached after the intra-PoS HO.

An LSP set up in per-node-set-up mode does not change due to an intra-PoS HO. If at least one LSP, be it a primary or a handover LSP, set up in per-node-set-up mode terminates at the PoS, the PoS has a host-specific routing table entry that needs to be updated during the intra-PoS HO.

An LSP set up in per-interface-set-up mode changes due to an intra-PoS HO. Packets received via such an LSP need to be redirected either onto an LSP set up in per-node-set-up mode or onto an LSP set up in per-interface-set-up mode that is set up for \( I_{\text{new}} \). The HO Notification message has to be sent along the reverse path of each LSP set up in per-interface-set-up mode that is set up for \( I_{\text{old}} \). If the egress LER is aware of via which of these LSPs the mobile device is receiving packets, it may send the HO Notification message along the reverse path of only those LSPs.

5.2.6 Inter-PoS HO

In the case of an inter-PoS HO, the egress LER being notified of the HO will no longer receive any packets for the mobile device. If it has a host-specific routing
table entry, it can remove this entry. If one or more LSPs set up in per-interface-setup mode terminate at the egress LER, a *HO Notification* message has to be sent along the reverse path of each of these LERs. Such a *HO Notification* message also has to be sent along each LSP set up in per-node-setup mode and terminating at the egress LER.

### 5.2.7 Handover Notification

The handover notification procedure is described for the set $S_{LSP}$ of LSPs which have to be updated. Two modes exist, per-node-notification and per-LSP-notification.

In per-LSP-notification mode, the egress LER generates a *HO Notification* message for each LSP within $S_{LSP}$ and sends the *HO Notification* message to the previous node of the LSP. In order to achieve the desired behavior, an RSVP-TE *RESV* message with an additional HO_NOTIFICATION object may be used. If the HO_NOTIFICATION object is not recognized by an LSR, the RSVP-TE *RESV* message is simply relayed to the previous node along the LSP without taking any specific action. Each LSR receiving such a *RESV* message processes the message and forwards it along the reverse path of the LSP as it would forward any other *RESV* message, i.e., it takes the Reservation State Block into account. This mode allows for simple processing of *RESV* messages at LSRs, but cannot avoid that possibly several *HO Notification* messages are sent between two nodes.

In per-node-notification mode, the egress LER generates only one *HO Notification* message per node, i.e., for each LSP within $S_{LSP}$ it generates the set of previous nodes and sends the *HO Notification* message to each node within that set. Each LSR receiving the *HO Notification* message proceeds in a similar way. In this mode, an LSR keeps a list of $(M, N, E)$ triplets where $M$ is the node it receives a *PATH* message from, where $N$ is the node it forwards the *PATH* message to and where $E$ is the egress LER the *PATH* message is destined to. On receiving a *HO Notification* message from some node $N$ that has been issued by the egress LER $E$, the LSR can then find out which nodes $M$ to forward that *HO Notification* message to. Note that a separate message (beyond *PATH* and *RESV* message) is required because a single such message possibly refers to more than one LSP.

Although the number of *HO Notification* messages is certainly lower in per-node-notification mode than in per-LSP-notification mode, the implementation effort at the LSR is higher. Since per-LSP-notification is certainly easier to implement, it is recommended to use per-LSP-notification mode.

The *HO Notification* message contains the UID of the mobile device, the old NID, the new NID and the sequence number or timestamp sent in the *HO Indication* message.

Each LSR receiving the RSVP-TE *HO Notification* message does the following (cf. Listing 5.1) unless it is configured as standard MPLS-capable node. If the *HO
Notification message belongs to a new handover that is not currently in progress, it needs to update both its ILM and possibly also its FTN entry. It thus retrieves the NHLFE from the HO FTN. The FEC used as index has to be determined from the new NID. The NHLFE designates the HO LSP leading from this LSR towards the edge node in charge of the network the mobile device is moving to and would be used for all mobile devices moving to that network. The LSR then places the NHLFE into the HO ILM and associates it with the UID given in the HO Notification message. If the entry at the UID already exists, a previous handover is still in progress. The ILM then needs to be updated since the current HO Notification message is more recent. If the entry at the UID does not yet exist, a new one is created. The consequence is that all further arriving MPLS packets that would otherwise be sent along the primary LSP towards the previous edge MPLS-capable node are now sent along the HO LSP towards the new edge MPLS-capable node. Note that no time- and resource-consuming reclassification of IP packets is necessary. If the LSR also operates as ingress LER, then the classification database needs to be updated as well. The HO Notification message is then sent to the previous hop along the reverse path of the primary LSP it has been received on. The number of pending HO Notification Ack messages is thus initialized with one. Note that the next A-MPLS-capable node updating the ILM does not necessarily equal the next MPLS-capable node relaying the HO Notification message. Since the next A-MPLS-capable node updating the ILM has to send a HO Notification Ack message back, an AMPLS_PHOP object has to be added to the HO Notification message. The AMPLS_PHOP object contains the IP address of the A-MPLS-capable node having updated the ILM and relaying the HO Notification message.

If the HO Notification message is part of the current handover, the HO Notification message is simply sent to the previous hop of the primary LSP it has been received on and the number of pending HO Notification Ack messages is increased by one.

In each case, the LSR sends a HO Notification Ack back to the A-MPLS-capable node having last updated the ILM. The address of the A-MPLS-capable node having last updated the ILM is taken from the AMPLS_PHOP object included into the HO Notification message. The receipt of a HO Notification Ack message is shown in pseudo-code in Listing 5.2. If the HO Notification Ack message is part of the current handover, the number of pending HO Notification Ack messages is decremented by one. On receipt of the last HO Notification Ack message, the LSR can remove the entry in the HO ILM.

When a new HO Notification that is not part of the current handover finally arrives at an ingress LER (cf. Listing 5.3), the ingress LER substitutes the FEC in the classification database by the FEC determined from the new NID. If an ILM entry exists, the ILM entry is also updated. Otherwise, the ingress LER determines whether or not it is also acting as LSR on some LSP to the new PoS. If so, it creates an ILM entry at the UID in order to have arriving MPLS packets
redirected although possibly no HO Notification message has yet arrived on such an LSP.

The procedure by which an ingress LER determines whether or not it is also acting as LSR on some LSP to the new PoS is given in Listing 5.4. The ingress LER goes through each entry in its Path State Block where it stores the issued and relayed PATH messages. If the PATH message is destined to the PoS the HO Notification message has been received from, the ingress LER is part of an LSP leading to that PoS. If, moreover, the original sender of the PATH message is not equal to the ingress LER, the ingress LER acts as LSR on that LSP.
122

CHAPTER 5. ACTIVE MPLS

Listing 5.3: Receipt of HO_Notification at ingress LER

if (HO_Notification_is_New) {
    FEC := Determine_FEC(nNID);
    NHLFE := HO_FTN --> Get_NHLFE(FEC);
    Classification --> Update(UID,FEC);
    if ILM --> NHLFE_Exists(UID) {
        ILM --> NHLFE_Update(UID,NHLFE)
    } else {
        if (am_LSR_for_Egress) {
            // Counter := 0
            ILM --> Create_NHLFE(UID,NHLFE);
        }
    }
}

Listing 5.4: Function returning whether or not the node is an LSR for the egress

int am_LSR_for_Egress {
    for each entry e in Path State Block {
        if (e.Session_Object.Ipv4_Tunnel_End_Point_Address ==
            HO_Notification.Original_Sender &
            e.Sender_Template_Object.Ipv4_Tunnel_Sender_Address != MySelf) {
            return 1; // true
        }
    }
    return 0; // false
}

5.2.8 Security

If the network operator wishes to secure the control plane communication between two MPLS-capable nodes, IPSec can be deployed (cf. Section 3.8). Securing the HO Indication sent by the mobile device over the air interface is subject to the technology used in the access network and described in Chapter 6.
5.2. **CONTROL PLANE**

5.2.9 **Example**

The following example is based on the architecture shown in Fig. 5.8. Note that all nodes are A-MPLS-capable nodes and configured to react on receipt of HO Notification messages.

There are two primary LSPs. Primary LSP 1 goes from R1 over R2 to R3. Primary LSP 2 goes from R1 over R4 to R5. Furthermore, two HO LSPs are set up. HO LSP 1 goes from R2 to R5. HO LSP 2 goes from R3 to R5. IP packets that arrive at R1 and that are destined to the mobile device with UID 42 are first classified. The result of the classification is FEC R3 and UID 42. By looking up FEC R3 in the FTN, R1 retrieves the NHLFE associated with FEC R3, places the outgoing label 5 contained in the NHLFE as well as the UID 42 before the IP packet and sends the resulting Active MPLS packet in direction of the next hop R2 contained in the NHLFE.

On receipt of the Active MPLS packet, R2 first looks up the UID in the HO ILM. As this look-up remains successful, R2 then looks up the incoming label 5 in the ILM, retrieves the NHLFE, switches the incoming label 5 for the outgoing label 7 and forwards the resulting Active MPLS packet in direction of R3.

R3 also has to perform two look-ups in order to find out how to proceed with...
the Active MPLS packet. Since the first look-up for the UID remains successless, R3 then looks up the incoming label 7 in the ILM and retrieves the NHLFE indicating to pop both the UID and the incoming label from the Active MPLS packet. If the LSP is set up in per-interface-set-up mode, R3 sends the resulting IP packet in direction of the next hop stored in the NHLFE. If the LSP is set up in per-node-set-up mode, R3 performs an additional routing-table look-up in order to determine the next hop and the outgoing interface.

Note that the packets of every user registered with R3 are processed in the same way, i.e. sent along primary LSP 1.

In the following, it is assumed that the mobile device with UID 42 changes the PoS from R3 to R5. It therefore issues a \textit{HO Indication} message (cf. Fig. 5.9) towards R3 before the actual L2 handover. The \textit{HO Indication} message contains the UID, a sequence number or time stamp, the old NID and the new NID. Note that only the UID and the new NID are shown in Fig. 5.9. Note also that the new NID is chosen as the name of the new PoS (i.e. as R5).

On receipt of the \textit{HO Indication} message, R3 looks up the identifier for the PoS the mobile device is moving to in its HO FTN, retrieves the NHLFE and places the NHLFE into the HO ILM at the UID (cf. Fig. 5.9). Further Active MPLS packets that arrive at R3 and that are destined for the mobile device with UID 42 are thus not forwarded towards the previous access network anymore but sent along HO LSP 2 from R3 to R5. At R5, these Active MPLS packets are finally forwarded towards the new access network the mobile device is moving to.

After having updated its HO ILM, R3 sends a \textit{HO Indication Ack} message back to the mobile device and creates a new RSVP-TE \textit{RESV} message, adds a \textit{HO NOTIFICATION} object to it and sends the message along the reverse path of primary LSP 1 to R2.

R2 performs the same operations as R3 such that further Active MPLS packets arriving at R2 carrying UID 42 are not sent to R3 anymore but along HO LSP 1 towards R5 (cf. Fig. 5.10). After having updated its HO ILM, R2 sends a \textit{HO Ack} message back to R3 indicating that it now takes care of all further arriving Active MPLS packets carrying UID 42. On receipt of the \textit{HO Ack} message, R3 can therefore remove the newly created entry in the HO ILM for the mobile device since the \textit{HO Ack} message is the only pending \textit{HO Ack} message. Eventually, R2 forwards the \textit{HO Notification} message to R1.

On receipt of the \textit{HO Notification} message, R1 as the ingress LER updates the classification data structure to take the FEC for the access network the mobile device is moving into account (cf. Fig. 5.11). Further IP packets are thus classified into FEC R5 and thus sent along primary LSP 2 towards R5. R1 finally sends a \textit{HO Notification Ack} message back to R2.
5.3 Testbed Implementation

The main objective of implementing Active MPLS on real hardware has been to show its feasibility. Since eight work stations each equipped with an Intel IXP1200 network processor have been available through an Intel grant, the implementation has been done on the IXP1200 network processor.

The objective of the implementation has also been to compare Active MPLS with Fast Active MPLS and with a third MPLS-based micro-mobility solution. In the third approach, only the ingress LER is updated which is why this approach is referred to as Ingress-Update. While the HO Notification message travels upstream to the ingress LER, the IP packets continue to be sent to the previous egress LER. These IP packets are considered as lost since with sending the HO Indication message the mobile host is assumed to start performing the L2 handover. Fig. 5.12 shows the message sequence diagram of Ingress-Update. Note that the approaches existing in current literature (cf. Section 4.6.1) propose a reactive mode of Ingress-Update. However, in order not to unnecessarily penalize Ingress-Update with respect to Active MPLS and Fast Active MPLS, a predictive mode is proposed here.
The architecture used to obtain implementation results is shown in Fig. 5.13. The nodes Dell0 through Dell7 are equipped with the IXP1200 and part of the actual domain. Dell8 and Dell9 perform necessary address translations described below.

There are two primary LSPs. Primary LSP 1 goes from Dell0 over Dell1 and Dell3 to Dell6, while primary LSP 2 goes from Dell0 over Dell2 and Dell5 to Dell7. Furthermore, there are three handover LSPs. HO LSP 1 goes from Dell6 over Dell4 and Dell5 to Dell7, HO LSP 2 goes from Dell3 over Dell4 and Dell5 to Dell7, and HO LSP 3 goes from Dell1 over Dell2 and Dell5 to Dell7. Note that Active MPLS makes use of all HO LSPs, whereas Fast Active MPLS only makes use of HO LSP 1 and Ingress-Update of no handover LSP at all.

IP packets are generated at and received by the laptop in order to avoid clock synchronization issues. Therefore, the laptop is connected to the ingress LER as well as to both egress LERs via a switch. Since it is not possible to transmit IP packets to a destination IP address bound to one of the laptop’s own interfaces, the IP packets have to be destined to an IP address that is not bound to one of the laptop’s own interfaces. In order to have the laptop still receive these packets, the destination IP address has to be changed at some node after having
5.3. TESTBED IMPLEMENTATION

Figure 5.11: HO Notification - Behavior of Ingress

been transmitted by the laptop. The nodes performing this address translation are Dell8 and Dell9.

An IP packet generated by the laptop is sent to the IP address 192.168.12.1. A routing table look-up yields the outgoing interface and the source IP address, i.e. 192.168.12.2. When handed to the link-layer the MAC address D is used as source address and the MAC address A as destination address. Neither Dell8 nor Dell9 accept packets with that MAC destination address. Only the ingress LER accepts the corresponding packets, classifies them and places them on primary LSP 1 towards Dell6. Dell6 takes off the labels and sends the resulting IP packet towards Dell8. Since the destination MAC address is not changed while the IP packet is running through the Active MPLS domain, Dell8 accepts the packet. It then changes both the source and the destination IP addresses and forwards the IP packet towards the laptop. Because of the change of the destination IP address, the laptop accepts the incoming IP packet. The MAC destination address is set correctly through a corresponding permanent ARP cache entry at Dell8. Note that downstream packets that are received by the switch are prevented from traversing the Active MPLS domain again. Even if the switch were to broadcast these downstream packets and if they thus arrived at the ingress
CHAPTER 5. ACTIVE MPLS

LER, the ingress LER would (in this set-up) drop them since their destination IP address would not equal the destination IP address of the packets generated by the laptop (192.168.12.1).

In order to perform a handover, the laptop generates a corresponding DHCPINFORM message with a HO object. Once the DHCPINFORM message is received by the egress LER Dell8, the respective mobility solution is triggered. In order to keep the implementation simple, Dell8 simply relays the DHCPINFORM message upstream along the reverse path of primary LSP 1.

At each hop processing the DHCPINFORM message, the message is delayed for about 50 ms before updating the ILM or the FTN. After the update the message is again delayed for about 50 ms before being forwarded to the next hop. The reason is that the implementation is restricted to the very basic functionalities so that the DHCPINFORM can be processed in less than 1 ms. Since processing times of control plane messages in this order of magnitude are not realistic for equipment deployed in the area, the DHCPINFORM message is delayed accordingly. Note that in Active MPLS, the DHCPINFORM message is processed at every hop along the reverse path of the previous primary LSP, whereas it is only processed at the LSP egress and the LSP ingress in Fast Active MPLS and Ingress-Update.

When the DHCPINFORM message arrives at the ingress LER, the ingress LER sends an acknowledgement towards the laptop in order to signal completion of the handover.
In order to evaluate the implementation, the handover has to be performed repeatedly. A run consists of the laptop performing a handover from Dell6 to Dell7 and back to Dell6. Note that only the handovers from Dell6 to Dell7 are evaluated. After each run, the whole network configuration is reset in order to keep the runs independent from one another. 2000 runs are performed.

The evaluation is done for two different traffic generation scenarios. In the first scenario, the Linux tool rude/crude is used for packet generation and reception. Packets are generated at a constant rate of 500 packets/s with 200 Byte each. This scenario is denoted as Constant Bit Rate (CBR) traffic scenario. In the second scenario, a video stream of a real video is transmitted through the domain. The average data rate is 362.85 kbit/s. This scenario is denoted as video traffic scenario. The inter-arrival times of both scenarios are depicted in Fig. 5.14. The left diagram shows the inter-arrival times of the data packets in the CBR traffic scenario, the right diagram the inter-arrival times of the data packets in the video traffic scenario.
traffic scenario. It can be seen that the relative frequency of the inter-arrival time is nearly 1 for an inter-arrival time of about 2 ms in the CBR traffic scenario, whereas it is distributed more or less equally in the range from 0 to 60 ms in the video traffic scenario. Note that all evaluation results are given in Section 5.6.

5.4 Simulation

The objective of the simulation is to analyze larger architectures and to compare Active MPLS and Fast Active MPLS with an IP-based micro-mobility solution that may not easily be implemented on real hardware. The simulation is performed with ns-2 [112]. The IP-based micro-mobility solution is the combined method of Low Latency Handoffs in Mobile IPv4 (cf. Section 4.4.3). Note that the implementation incorporated into ns-2 did not correspond to the specification in [98] so that it has had to be adapted to comply with that specification. The message sequence diagram of the combined method of Low Latency Handoffs in Mobile IPv4 is shown in Fig. 5.15.

The architecture used in the simulation is depicted in Fig. 5.16. Note that the MIPv4 Home Agent, which is necessary for Low Latency Handoffs in Mobile IPv4, is located in R0 and thus separated from the ingress LER R1. By assigning different propagation delays to the link connecting R0 and R1, different distances of the HA to the network domain can be simulated.

The evaluation is done for two different traffic generation scenarios. In the first scenario, packets are generated at a rate of 0.15 Mbit/s with 100 Byte each. This scenario is denoted as CBR traffic scenario. In a more advanced scenario, a video stream is transmitted through the domain. The packet arrival process is modelled as follows. The inter-arrival times are exponentially distributed with an arrival rate of 200 packets/s. The packet size is left constant at 100 Byte. This scenario is denoted as video traffic scenario. All evaluation results are given in Section 5.6. In each traffic generation scenario, 1000 runs are performed.
5.5 Evaluated Parameters

There are many parameters that may be evaluated in the implementation and the simulation. These parameters may be classified into three groups. The first group contains parameters related to the handover itself. These are the outage period, the handover delay, the loss period and packet loss. The second group contains the parameters related to the network. These are e.g. network utilization and the number of user-specific routing table or ILM entries within the domain. The third and last group contains the parameters related to the data packets. These are e.g. end-to-end delay and jitter. All parameters are explained in more detail in the next subsections.

5.5.1 Outage Period

The outage period is a handover-related parameter and also referred to as disruption time, handover latency [60] or handover time [48]. It can be defined as the time between reception of the last data packet via the previous PoA and...
reception of the first data packet via the new PoA [48]. The difficulty with this definition is that the outage period depends on the characteristics of the traffic flow.

The outage period is precisely given as the time period during which the mobile device may not receive or send any data packets. This time period starts when the mobile device starts the L1 handover and ends when the mobile device has registered with the new PoS and is enabled to receive the first data packet, independently of whether or not a data packet is really sent. The difficulty of this precise definition is that the start point in time can hardly be measured since it is generally not known when the mobile device actually starts performing the L1 handover.

The start point in time is therefore chosen as the beginning of the L2 handover. In the practical implementation, this point in time can be properly determined by communicating with the device driver. Note that the outage period is in particular relevant for inter-active and real-time applications where a long outage period has a negative impact on the perceived QoS.
5.5. **EVALUATED PARAMETERS**

5.5.2 **Loss Period**

During the outage period the mobile device may not send or receive any data packets. However, downstream data packets are not necessarily lost. Data packets are only lost if the mobility solution does not provide a means to buffer the data packets during the outage period. The period during which data packets are definitively lost is here denoted as loss period. Note that the loss period is always less than or equal to the outage period. Note also that the loss period primarily determines the packet loss during the handover. Finally note that the loss period is relevant for any type of application that the end user is running during the handover.

5.5.3 **Handover Delay**

The handover delay is a handover-related parameter and defined in this document as the time period between the point in time \( t_{\text{first}} \) when the first signalling message that can be associated with the handover is sent and the point in time \( t_{\text{last}} \) when the last signalling message that can be associated with the handover is received. The handover delay is mainly needed to restrict the set of packets evaluated for packet-related parameters to those packets affected by the handover. Note that the handover delay usually differs from the outage period, e.g. when L3 or L2 signalling messages are sent before the L2 handover (as is the case for predictive mobility solutions) or when L3 messages are sent after the L2 handover although the mobile device is already enabled to send and receive data packets via the new PoS (as is the case for FMIPv6, Active MPLS, etc.). The handover delay is given as the difference \( t_{\text{last}} - t_{\text{first}} \). The interval \([t_{\text{first}}, t_{\text{last}}]\) is denoted as handover interval.

For the packet-related parameters it is necessary to restrict the set of all packets sent to those that are affected by the handover. These packets can be filtered out of the set of all packets as follows. Each packet \( P_i \) enters the micro-mobility domain at a point in time \( t_{\text{in}} \). This point in time is when the packet is received by the domain ingress node it arrives at. If it is not lost, the packet is received by the mobile device at a point in time \( t_{\text{recv}} \). Let the interval \([t_{\text{recv}}, t_{\text{in}}]\) be denoted as packet-in-transit interval.

If for a particular packet \( P_i \) the packet-in-transit interval \([t_{\text{recv}}, t_{\text{in}}]\) intersects with the handover interval \([t_{\text{first}}, t_{\text{last}}]\), the packet is considered as being affected by the handover. These packets are taken into account for the packet-related parameters.

Note that the handover delay primarily determines both the packet-related and the network-related parameters and is thus indirectly relevant both for the network operator and the end user.
CHAPTER 5. ACTIVE MPLS

5.5.4 Packet Loss

Packet loss is a handover-related parameter. It is calculated as the number of packets lost during the handover. If the set of packets affected by the handover is given by \( S_{\text{affected}} \), the packet loss is determined as follows.

\[
\text{Loss} := |\{ P \in S_{\text{affected}} \mid P \text{ is lost} \}| \quad (5.1)
\]

Packets can be lost for different reasons. Packets can be lost on the air interface, in the access network or in the core network. In the access or core network, packets may be lost on a link (e.g. due to link failures) or in an MPLS-capable node (e.g. due to node failures, buffer overflows, link unavailability, etc.). For the evaluation of the micro-mobility solutions, packet loss is assumed to not occur due to link or node failures. Furthermore, the buffers and traffic flows are dimensioned in such a way that packet loss due to buffer overflows should be an event with negligible probability, i.e. it is not considered here that a handover for one traffic flow may affect other traffic flows which are not involved in a handover. On the contrary, packet loss is evaluated for those cases where an edge MPLS-capable node receives a packet for a mobile device and where it has to drop the packet. Such a case occurs when the decision about where to forward the packet cannot be taken and where the mobility solution does not incorporate a mechanism to buffer the packet for later forwarding. Such a case also occurs when the outgoing interface can be determined and where the mobile device has already left the range of the associated PoS.

5.5.5 Network Utilization

Network utilization is a network-related parameter. It designates the utilization of the operator’s network. If the network is given as a graph \( G = (V, E) \) with \( V \) being the set of nodes and \( E \) the set of unidirectional links between two nodes, the network utilization \( \rho_{\text{network}} \) can be calculated as follows.

\[
\rho_{\text{network}} := \frac{\sum_{e \in E} C_{\text{used}}(e)}{\sum_{e \in E} C_{\text{total}}(e)} \quad (5.2)
\]

\( C_{\text{used}}(e) \) designates the capacity currently used on the link \( e \), whereas \( C_{\text{total}}(e) \) designates the total capacity of the link \( e \). Note that \( \rho_{\text{network}} \) does not indicate the utilization on individual links.

5.5.6 Number of User-Specific Table Entries

The number of user-specific table entries is a network-related parameter and a measure for the memory size needed at a core node and lastly an important measure for scalability. Since core nodes are supposed to be simple and fast devices,
small memories and thus small tables are important. Small tables generally also allow for fast table look-ups and facilitate table configuration and management. Note that the number of permanent user-specific entries has to be distinguished from the number of temporary user-specific entries.

Let the network be given as a graph $G = (V, E)$ with $V$ being the set of nodes and $E$ the set of unidirectional links between two nodes. Let further the number of nodes be given as $N = |V|$, the number of ingress nodes as $N_{in}$, the number of egress nodes as $N_{out}$ and the number of nodes acting both as ingress and egress as $N_{both}$. Let finally the number of users be given as $N_{user}$.

In some micro-mobility solutions, user-specific routing table entries are deployed. These are necessary for every user on the paths from every ingress node to the egress node currently serving that user. The number of paths thus equals $N_{user} \cdot N_{in}$. The number of host-specific routing table entries equals the sum of the path lengths of each of the $N_{user} \cdot N_{in}$ paths and depends on the topology and the choice of paths.

In other micro-mobility solutions, user-specific table entries are deployed. Since an LSP from an ingress node to an egress node may carry the traffic of all users which are currently served by the egress node, the number of LSPs may be as less as one LSP between every pair of ingress and egress nodes. Note that $N_{in} \cdot N_{out}$ does not describe the desired number correctly since an LSP from a node that acts both as ingress and as egress to that same node is counted as well. Therefore, the number of these LSPs has to be subtracted. The resulting number of LSPs amounts to $N_{in} \cdot N_{out} - N_{both}$.

Beyond these primary LSPs, Active MPLS also deploys handover LSPs. Handover LSPs are at most established from every non-ingress node to every egress node. Note that $(N - N_{in}) \cdot N_{out}$ does not describe the desired number correctly since an LSP from a node that acts both as non-ingress and as egress to that same node is counted as well. Therefore, the number of these LSPs has to be subtracted. The resulting number of LSPs amounts to $(N - N_{in}) \cdot N_{out} - (N_{out} - N_{both})$.

Note that the actual number of handover LSPs is generally lower since handover LSPs are only needed from those non-ingress nodes that are part of some primary LSP. Note also that the formula is valid only if all LSPs are set up in per-node-set-up mode. If LSPs are set up in per-interface-set-up mode, the number of LSPs increases with the number of interfaces at ingress and egress LERs.

Eventually, the number of user-specific table entries is bounded above by the sum of the path lengths of each of the LSPs and depends on the topology and the choice of primary and handover LSPs.

5.5.7 End-to-End Delay

End-to-end delay is a packet-related parameter. It is calculated as the difference of the point in time $t'_{rcvd}$ when a packet $P_i$ is received at the new egress LER and the point in time $t'_{sent}$ when the same packet is sent by the ingress LER.
CHAPTER 5. ACTIVE MPLS

By restricting the end-to-end delay to the domain, the different micro-mobility solutions can be properly distinguished.

\[ \text{e2e-delay}(P_i) := t_{i, \text{rcvd}} - t_{i, \text{sent}} \]  \hspace{1cm} (5.3)

5.5.8 Jitter

There is more than one definition for jitter. Some references calculate it as the difference of the maximum and minimum (end-to-end) delays with respect to the (end-to-end) delays of all packets within a packet flow so that the jitter becomes a packet-flow related parameter. This definition is well suited for buffer dimensioning since it describes a worst-case scenario. In this dissertation, jitter is defined as a per-packet value and given as the difference of the end-to-end delay of a packet \( P_i \) (with sequence number \( i \)) and the one of packet \( P_{i-1} \) (with sequence number \( i - 1 \)).

\[ \text{jitter}(P_i) := \text{e2e-delay}(P_i) - \text{e2e-delay}(P_{i-1}) \]  \hspace{1cm} (5.4)

This definition implies that the two packets have to be from the same traffic flow with at least the same source and the same destination. Note that the jitter may be negative in which case the end-to-end delay of packet \( P_i \) has been less than the end-to-end delay of packet \( P_{i-1} \).

5.6 Evaluation

A comparison of some micro-mobility protocols has been performed by [34]. However, the comparison is restricted to Cellular IP, HAWAII and Hierarchical Mobile IPv4. Furthermore, the comparison is only based on throughput and packet loss and does not take any other parameters into account. The evaluation presented in this document is much more extensive.

5.6.1 Active MPLS vs. Fast Active MPLS

The difference of Fast Active MPLS with respect to Active MPLS is that the LSRs between LSP ingress and LSP egress are either standard MPLS-capable nodes or A-MPLS-capable nodes that are configured to behave as standard MPLS-capable nodes, i.e., only the LSP ingress and the LSP egress act as A-MPLS-capable nodes. Fast Active MPLS benefits from the fact that the LSP egress may send the \( \text{HO Notification} \) message directly to the LSP ingress so that the intermediate LSRs may forward the \( \text{HO Notification} \) message on the user plane and do not need to process it. Forwarding the \( \text{HO Notification} \) message on the user plane is much faster than relaying it on the control plane, which is why Fast Active MPLS has been assigned its name.
The different operation of Active MPLS and Fast Active MPLS can best be viewed with the help of message sequence diagrams. Fig. 5.17 shows the message sequence diagram for Active MPLS, while Fig. 5.18 shows the message sequence diagram for Fast Active MPLS. Note that both message sequence diagrams are based on the architecture shown in Fig. 5.13.

After sending the DHCPINFORM message to the previous egress LER, the Mobile Node (MN) is assumed to perform the L2 HO. The length of the L2 HO is configured to 20 ms which is considered a realistic assumption. After receipt of the DHCPINFORM message, the previous egress LER is assumed to take 50 ms to process the message on the control plane. During that time, it continues sending data packets over the air interface. The last such data packet completes the loss period since further data packets may be redirected to and buffered at
the new egress LER. After another 50 ms during which the outgoing message or messages are prepared, the previous egress LER sends a context transfer message towards the new egress LER. The context transfer message may contain authentication and configuration information for the MN. In the meantime, the MN has completed the L2 HO and registers with the new egress LER by sending a Registration message. Note that the new egress LER can only reply with a Registration Ack message after receipt of both the context transfer message and the Registration message. The receipt of the Registration Ack message completes the outage period. The message exchange described so far is the same for Active MPLS and Fast Active MPLS since it does not affect the location update within the network domain. Active MPLS and Fast Active MPLS differ from one another in the way the HO Request and HO Response messages are sent. Accordingly, the last HO Response message (after having been processed) accomplishes the handover delay that is much lower in Fast Active MPLS than in Active MPLS.

The packet-related parameters end-to-end delay and jitter are actually only evaluated for those packets affected by the handover. However, when comparing Active MPLS with Fast Active MPLS, it becomes evident that during the relatively long handover delay of Active MPLS packets are successively redirected over different LSPs, whereas during the relatively short handover delay of Fast Active MPLS all packets are sent from the LSP ingress to the previous LSP egress and from there to the new LSP egress. In Fast Active MPLS, all packets affected
by the handover thus travel via an inefficient data path. However, these are only very few packets since the Ho Notification message arrives at the LSP ingress very quickly. In order to take this into consideration when comparing Active MPLS with Fast Active MPLS, the longer handover delay has to be taken into account when determining the packets for the evaluation of the packet-related parameters. The question that arises is under what circumstances Active MPLS and under what circumstances Fast Active MPLS should be deployed.

Therefore, the path lengths of the data paths used during a handover are derived analytically below. The comparison of Active MPLS with Fast Active MPLS is, however, based on the mean path lengths of the data paths used during a handover. Note that other measures are possible.

Let the previous primary LSP be denoted as \( v_1, \ldots, v_n \). The new primary LSP extends from \( v_1 \) to the new egress LER, while handover LSPs may be deployed from \( v_i \) (\( 2 \leq i \leq n \)) to the new egress LER. Let further \( t_{air} \) denote the serialization delay over the air interface and \( t_{forw} \) the serialization delay within the network domain. \( t_{proc} \) denotes the processing delay after receipt of a control plane message and before taking any action, while \( t_{proc2} \) denotes the processing delay after taking that action and before serializing the altered or a new message.

The handover delay can be obtained in a straightforward manner from Fig. 5.17 and Fig. 5.18.

\[
A_{MPLS}^{HD} = t_{air} + t_{proc1} + n \cdot (t_{proc2} + t_{prop} + t_{proc1}) \quad (5.5)
\]

\[
Fast^{HD} = t_{air} + t_{proc1} + 2 \cdot (t_{proc2} + t_{prop} + t_{proc1})
+ 2 \cdot (n-2) \cdot (t_{forw} + t_{prop}) \quad (5.6)
\]

Active MPLS performs location updates at the egress LER, every intermediate LSR and the ingress LER. The point in time of each location update is calculated as shown below. It is obtained in a straightforward manner from Fig. 5.17.

\[
A_{MPLS}^{LU}[1] = t_{air} + t_{proc1} \quad (5.7)
\]

\[
A_{MPLS}^{LU}[i] = A_{MPLS}^{LU}[i - 1] + t_{proc2} + t_{prop} + t_{proc1} \quad (i = 2, \ldots, n)
= t_{air} + t_{proc1} + (i - 1) \cdot (t_{forw} + t_{prop} + t_{proc1}) \quad (5.8)
\]

Fast Active MPLS only performs location updates at the egress LER and the ingress LER. The points in time of these two location updates are obtained analogously.

\[
Fast^{LU}[1] = A_{MPLS}^{LU}[1] \quad (5.9)
\]

\[
Fast^{LU}[2] = Fast^{LU}[1] + t_{proc2} + t_{prop} + (n-2) \cdot (t_{forw} + t_{prop}) + t_{proc1} \quad (5.10)
\]
CHAPTER 5. ACTIVE MPLS

The mean path length during the handover is calculated independently from the packets that are actually affected by the handover by taking into account both the lengths of the paths that are made use of and the duration during which these paths are actually used. $d_{LU}^{\text{A-MPLS}}[i]$ is the duration during which location update $i-1$ is active. Location update $i-1$ is active from the point in time $t_{LU}^{\text{A-MPLS}}[i-1]$ where location update $i-1$ is completed up to the point in time $t_{LU}^{\text{A-MPLS}}[i]$ where location update $i$ is completed. $d_{LU}^{\text{Fast}}$ is the duration during which the first location update in Fast Active MPLS is active.

$$d_{LU}^{\text{A-MPLS}}[i] = t_{\text{proc}2} + t_{\text{prop}} + t_{\text{proc}1} \quad (i = 2, \ldots, n)$$
$$= t_{\text{redirection}} \quad (5.11)$$

$$d_{LU}^{\text{Fast}} = t_{\text{proc}2} + t_{\text{prop}} + (n-2) \cdot (t_{\text{forw}} + t_{\text{prop}}) + t_{\text{proc}1}$$
$$= t_{\text{redirection}} + (n-2) \cdot t_{\text{delay}} \quad (5.12)$$

The handover delay of Active MPLS is much longer than that of Fast Active MPLS. In order to compare both approaches, the longer handover delay has to be taken into account. Since the behavior of both approaches is the same until the first location update, the time until the first location update is not taken into account. The resulting duration is here denoted as $t_{\text{total}}$.

$$t_{\text{total}} = (n-1) \cdot (t_{\text{proc}2} + t_{\text{prop}} + t_{\text{proc}1})$$
$$= (n-1) \cdot t_{\text{redirection}} \quad (5.13)$$

The mean path length of Active MPLS and Fast Active MPLS is computed as follows. $l_i$ denotes the length of the data path from the ingress LER to the egress LER after location update at node $v_i$ on the previous primary LSP. In particular, $l_1$ denotes the length of the new primary LSP.

$$\bar{l}_{\text{path}}^{\text{A-MPLS}} = \frac{1}{t_{\text{total}}} \cdot \sum_{i=2}^{n} l_i \cdot t_{\text{redirection}}$$
$$= \frac{1}{n-1} \cdot \sum_{i=2}^{n} l_i \quad (5.14)$$

$$\bar{l}_{\text{path}}^{\text{Fast}} = \frac{1}{t_{\text{total}}} \left( d_{LU}^{\text{Fast}} \cdot l_n + (t_{\text{total}} - d_{LU}^{\text{Fast}}) \cdot l_1 \right)$$
$$= \frac{d_{LU}^{\text{Fast}}}{t_{\text{total}}} \cdot l_n + \left( 1 - \frac{d_{LU}^{\text{Fast}}}{t_{\text{total}}} \right) \cdot l_1$$
$$= \frac{d_{LU}^{\text{Fast}}}{t_{\text{total}}} \cdot (l_n - l_1) + l_1$$
$$= \frac{t_{\text{redirection}} + (n-2) \cdot t_{\text{delay}}}{(n-1) \cdot t_{\text{redirection}}} \cdot (l_n - l_1) + l_1$$
### 5.6. EVALUATION

<table>
<thead>
<tr>
<th>Network</th>
<th>Access</th>
<th>Core</th>
<th>Satellite-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_{\text{proc1}}$</td>
<td>50 ms</td>
<td>50 ms</td>
<td>50 ms</td>
</tr>
<tr>
<td>$t_{\text{proc2}}$</td>
<td>50 ms</td>
<td>50 ms</td>
<td>50 ms</td>
</tr>
<tr>
<td>Length of links</td>
<td>20 km</td>
<td>50 km</td>
<td>50,000 km</td>
</tr>
<tr>
<td>$t_{\text{prop}}$</td>
<td>0.1 ms</td>
<td>0.25 ms</td>
<td>250 ms</td>
</tr>
<tr>
<td>Packet size</td>
<td>1000 Byte</td>
<td>1000 Byte</td>
<td>1000 Byte</td>
</tr>
<tr>
<td>Data rate</td>
<td>2 Mbit/s</td>
<td>100 Mbit/s</td>
<td>100 Mbit/s</td>
</tr>
<tr>
<td>$t_{\text{forw}}$</td>
<td>4 ms</td>
<td>0.08 ms</td>
<td>0.08 ms</td>
</tr>
<tr>
<td>$q$</td>
<td>24</td>
<td>304</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 5.1: Parameter $q$ for different networks

\[
q = \left( \frac{1}{n-1} + \frac{n-2}{n-1} \cdot \frac{1}{q} \right) \cdot (l_n - l_1) + l_1 \tag{5.15}
\]

The quotient $q$ is a quotient that is characteristic for a network and its nodes. It is determined as follows.

\[
q = \frac{t_{\text{redirection}}}{t_{\text{delay}}} = \frac{t_{\text{proc2}} + t_{\text{prop}} + t_{\text{proc1}}}{t_{\text{forw}} + t_{\text{prop}}} \tag{5.16}
\]

Table 5.1 shows the quotient $q$ for different types of networks and scenarios. Fig. 5.19 shows the mean path length for Active MPLS and Fast Active MPLS in function of the quotient $q$ for the network architecture depicted in Fig. 5.13. With computation of $q$ for a certain network, it can be determined whether it is beneficial to deploy Active MPLS or Fast Active MPLS.

In a common access or core network, the quotient $q$ is rather large, i.e. it is more beneficial to deploy Fast Active MPLS than to deploy Active MPLS. However, in a purely satellite-based network where the propagation delays are very long, the quotient $q$ equals about 1 indicating that Active MPLS is to be preferred (cf. Fig. 5.19). Note that purely satellite-based networks do not yet exist today, but may possibly be deployed in the future.

#### 5.6.2 Testbed Implementation

The implementation provides only handover-related parameters and packet-related parameters. Network-related parameters are hard to obtain in environments based on network processors. The results are provided for the three micro-mobility solutions that are evaluated. These are Active MPLS, Fast Active MPLS and Ingress-Update. End-to-end delay and jitter are provided in terms of histograms since the characteristics of the micro-mobility solutions come out better than in the respective cumulative distribution functions.
5.6.2.1 Handover-related parameters

The estimated mean handover delay for Active MPLS, Fast Active MPLS and Ingress-Update is shown in Fig. 5.20. The left diagram depicts the estimated mean handover delay in the CBR traffic scenario, the right diagram the estimated mean handover delay in the video traffic scenario. In both scenarios, the estimated mean handover delay of Active MPLS is much higher than that of Fast Active MPLS and Ingress-Update. The reason becomes evident when looking at the message sequence diagrams in Fig. 5.17, Fig. 5.18 and Fig. 5.12. Note that the handover delay is independent of the actual data traffic sent on the user plane so that the values are very much alike in both scenarios. In the CBR traffic scenario, the estimated mean handover delay equals about 537 ms for Active MPLS and about 298 ms for both Fast Active MPLS and Ingress-Update. The 0.95 confidence interval amounts to about ±4 ms for Active MPLS and to about ±2 ms for both Fast Active MPLS and Ingress-Update. In the video traffic scenario, the estimated mean handover delay equals about 540 ms for Active MPLS, about 307 ms for Fast Active MPLS and about 326 ms for Ingress-Update. The 0.95 confidence interval amounts to about ±2 ms for both Active MPLS and Fast Active MPLS and to about ±3 ms for Ingress-Update.

The estimated mean outage period for Active MPLS, Fast Active MPLS and Ingress-Update is shown in Fig. 5.21. The left diagram depicts the estimated mean outage period in the CBR traffic scenario, the right diagram the estimated
mean outage period in the video traffic scenario. In both scenarios, the estimated mean outage period of the three approaches is very much alike. Note that the end point in time of the outage period of Active MPLS and Fast Active MPLS can be obtained via signalling messages (cf. the message sequence diagrams of Fig. 5.17 and Fig. 5.18), whereas in Ingress-Update it needs to be obtained via the first data packet arriving at the mobile device (cf. the message sequence diagram of Fig. 5.12). Since the data rate in the video traffic scenario is lower than in the CBR traffic scenario, the estimated mean outage period is accordingly higher in the video traffic scenario although the actual outage periods should be equal. In the CBR traffic scenario, the estimated mean outage period equals about 183 ms for Active MPLS, about 164 ms for Fast Active MPLS and about 177 ms for Ingress-Update. The 0.95 confidence interval amounts to about ±5 ms for Active MPLS, to about ±3 ms for Fast Active MPLS and to about 2 ms for Ingress-Update. In the video traffic scenario, the estimated mean outage period equals about 170 ms for Active MPLS, about 173 ms for Fast Active MPLS and about 223 ms for Ingress-Update. The 0.95 confidence interval amounts to about ±2 ms for both Active MPLS and Fast Active MPLS and to about ±4 ms for Ingress-Update.

The estimated mean number of lost packets is shown in Fig. 5.22 for Active MPLS, Fast Active MPLS and Ingress-Update. The left diagram depicts the estimated mean number of lost packets in the CBR traffic scenario, the right diagram the estimated mean number of lost packets in the video traffic scenario. In both scenarios, the estimated mean number of lost packets is considerably higher for Ingress-Update than for Active MPLS and Fast Active MPLS. The reason becomes evident when looking at the data loss period in the message sequence diagrams of Fig. 5.17, Fig. 5.18 and Fig. 5.12. In Ingress-Update data packets continue to be sent to the previous egress until the ingress is updated and instructed to redirect the data packets towards the new egress, whereas in Active MPLS and Fast Active MPLS the data packets are already redirected at
the previous egress. The estimated mean number of lost packets is different in the two scenarios since it is dependent on the actual data rates used. In the CBR traffic scenario, the estimated mean number of lost packets equals about 45 packets for Active MPLS, about 42 packets for Fast Active MPLS and about 98 packets for Ingress-Update. The 0.95 confidence interval amounts to about ±2 packets for both Active MPLS, Fast Active MPLS and Ingress-Update. In the video traffic scenario, the estimated mean number of lost packets equals about 2 packets for both Active MPLS and Fast Active MPLS and about 7 packets for Ingress-Update. The 0.95 confidence interval amounts to about ±0.1 packets for both Active MPLS and Fast Active MPLS and to about ±0.3 packets for Ingress-Update.

In order to mitigate the influence of the data rate on packet loss, the estimated mean percentage of lost packets is shown in the left diagram of Fig. 5.23 for the CBR traffic scenario and in the right diagram for the video traffic scenario. The estimated mean percentage of lost packets is depicted for Active MPLS, Fast Active MPLS and Ingress-Update. Note that the percentage of lost packets is computed by dividing the number of lost packets by the number of all packets af-
In order to make the percentages of the three approaches comparable to one another, the number of all packets affected by the handover should be the same. Therefore, the handover delay of Active MPLS (the longest handover delay) is taken into account in order to obtain the number of all packets affected by the handover. In both scenarios, the estimated mean percentage of lost packets is considerably higher for Ingress-Update than for Active MPLS and Fast Active MPLS. In the CBR traffic scenario, the estimated mean percentage of lost packets equals about 15% for Active MPLS, about 14% for Fast Active MPLS and about 34% for Ingress-Update. The 0.95 confidence intervals amount to about ±1% for both Active MPLS, Fast Active MPLS and Ingress-Update. In the video traffic scenario, the estimated mean percentage of lost packets equals about 13% for both Active MPLS and Fast Active MPLS and about 38% for Ingress-Update. The 0.95 confidence intervals amount to about ±0.5% for Active MPLS and to about ±1% for both Fast Active MPLS and Ingress-Update.

5.6.2.2 Packet-related parameters

The upper left diagram of Fig. 5.24 shows the end-to-end delay for Active MPLS in the CBR traffic scenario, the upper right diagram the end-to-end delay in the video traffic scenario. Four peaks can be seen. Each peak results from a certain set of packets. The first peak results from those packets that are sent along the new primary LSP towards Dell7 (cf. Fig. 5.13). The path length is three. The second peak results from those packets that are redirected over handover LSP 3 at Dell1. The path length is four. The third peak results from those packets that are redirected over handover LSP 2 at Dell3. The path length is five. The last peak results from those packets that are redirected at the previous egress LER Dell6. The path length is six. The confidence intervals are depicted in the diagrams. Note that the absolute values for the end-to-end delay are higher in the video traffic scenario because the packet size of the video packets is higher.

The middle left diagram of Fig. 5.24 shows the end-to-end delay for Fast
Active MPLS in the CBR traffic scenario, the middle right diagram the end-
to-end delay in the video traffic scenario. Only two peaks can be seen. The
first peak results from those packets that are sent along the new primary LSP
towards Dell7. The second peak results from those packets that are redirected
at the previous egress LER Dell6. These two peaks clearly show the benefit of
Fast Active MPLS with respect to Active MPLS in the two scenarios since the
shorter handover delay leads to a larger set of packets sent via the new primary
LSP. The confidence intervals are depicted in the diagrams.

The lower left diagram of Fig. 5.24 shows the end-to-end delay for Ingress-
Update in the CBR traffic scenario, the lower right diagram the end-to-end delay
in the video traffic scenario. Only one peak can be seen. The peak results from
those packets that are sent along the new primary LSP towards Dell7. All other
packets are lost since they continue to be sent to the previous egress LER. The
confidence intervals are depicted in the diagrams.

The upper left diagram of Fig. 5.25 shows the jitter for Active MPLS in
the CBR traffic scenario, the upper right diagram the jitter in the video traffic
scenario. It can be seen that the jitter is primarily zero for all packets. The
reason is that the jitter is only higher for those packets that are the first to be
redirected. All further packets following the first packet along a handover LSP
or the new primary LSP should basically have the same end-to-end delay as the
packet before. The confidence intervals are depicted in the diagrams.

The middle left diagram of Fig. 5.25 shows the jitter for Fast Active MPLS in
the CBR traffic scenario, the middle right diagram the jitter in the video traffic
scenario. It can be seen that the jitter is primarily zero for all packets. The
reason is the same as for Active MPLS. The confidence intervals are depicted in
the diagrams. The lower left diagram of Fig. 5.25 shows the jitter for Ingress-
Update in the CBR traffic scenario, the lower right diagram the jitter in the video
traffic scenario. It can be seen that the jitter is primarily zero for all packets. The
reason is the same as for Active MPLS and Fast Active MPLS. The confidence
intervals are depicted in the diagrams.

5.6.3 Simulation

In the same way as the implementation, the simulation yields handover-related
and packet-related parameters. The results are provided for Active MPLS, Fast
Active MPLS and the combined handover method of Low Latency Handoffs in
Mobile IPv4. The results have been evaluated for the five handovers depicted in
the architecture shown in Fig. 5.16.

5.6.3.1 Handover-related parameters

The estimated mean handover delay for Active MPLS, Fast Active MPLS and
Low Latency Handoffs is shown in Fig. 5.26. The left diagram depicts the esti-
5.6. Evaluation

![Graphs showing end-to-end delay for Active MPLS, Fast Active MPLS, and Ingress-Update for both CBR and video traffic.](image)

Figure 5.24: End-to-end delay for Active MPLS, Fast Active MPLS and Ingress-Update evaluated for both constant bit rate and video traffic

The estimated mean handover delay in the CBR traffic scenario, the right diagram, shows the estimated mean handover delay in the video traffic scenario. Note that the handover delay is independent of the actual data traffic sent on the user plane so that the values are very much alike in both scenarios. In the CBR traffic scenario, the estimated mean handover delay for Active MPLS equals about 552 ms for handovers 1, 2 and 5 and about 452 ms for handovers 3 and 4. For Fast Active MPLS, it equals about 251 ms for each handover, while it amounts to about 353 ms for Low Latency Handoffs. The 0.95 confidence intervals amount
Figure 5.25: Jitter for Active MPLS, Fast Active MPLS and Ingress-Update evaluated for both constant bit rate and video traffic to about ±0.03 ms for both Active MPLS and Fast Active MPLS and to about ±0.04 ms for Low Latency Handoffs. In the video traffic scenario, the estimated mean handover delays have about the same values as in the CBR traffic scenario. The 0.95 confidence intervals amount to about ±0.02 ms for both Active MPLS and Fast Active MPLS and to about ±0.17 ms for Low Latency Handoffs.

Note that the estimated mean handover delay for Active MPLS is much higher than that for Fast Active MPLS and Low Latency Handoffs. The reason becomes evident when looking at the message sequence diagrams in Fig. 5.17, Fig. 5.18.
5.6. EVALUATION

Figure 5.26: Handover delay for constant bit rate (left diagram) and video traffic (right diagram)

and Fig. 5.15. Note also that the estimated mean handover delay of Active MPLS is about 100 ms higher for handovers 1, 2 and 5 with respect to handovers 3 and 4. The reason is that in the simulation all A-MPLS-capable nodes are configured as potential ingress LERs so that all A-MPLS-capable nodes have to be notified of the corresponding handover. In handovers 1, 2 and 5, the most distant A-MPLS-capable node is 4 hops away from the previous PoS, in handovers 3 and 4, the most distant A-MPLS-capable node is, however, only 3 hops away from the previous PoS. Since the processing delay of the HO Notification message is assumed to equal about 100 ms (cf. Section 5.3), the above-mentioned difference of 100 ms in the estimated mean handover delays becomes evident.

The estimated mean outage period for Active MPLS, Fast Active MPLS and Low Latency Handoffs is shown in Fig. 5.27. The left diagram depicts the estimated mean outage period in the CBR traffic scenario, the right diagram the estimated mean outage period in the video traffic scenario. In the CBR traffic scenario, the estimated mean outage period equals about 102 ms for Active MPLS and Fast Active MPLS during handovers 1, 3 and 5 and about 153 ms during handovers 2 and 4. For Low Latency Handoffs it amounts to about 103 ms for handovers 1, 3 and 5 and to about 252 ms for handovers 2 and 4. The 0.95 confidence intervals amount to about ±0.05 ms for Active MPLS and Fast Active MPLS and to about ±0.07 ms for Low Latency Handoffs. In the video traffic scenario, the estimated mean outage periods have about the same values as in the CBR traffic scenario. The 0.95 confidence intervals amount to about ±0.04 ms for both Active MPLS and Fast Active MPLS and to about ±0.68 ms for Low Latency Handoffs.

When comparing the two scenarios, it can be seen that the estimated mean outage periods of the three approaches are very much alike for all handovers. The reason is that the outage period is independent of the actual data traffic sent on the user plane and thus of the two traffic scenarios. When looking at one
Figure 5.27: Outage period for constant bit rate (left diagram) and video traffic (right diagram)

scenario separately, it can be seen that the estimated mean outage periods of the three approaches are about equal for handovers 1, 3 and 5 and much higher in Low Latency Handoffs in Mobile IPv4 for handovers 2 and 4. The reason is that handovers 1, 3 and 5 are intra-PoS handovers, whereas handovers 2 and 4 are inter-PoS handovers where Low Latency Handoffs in Mobile IPv4 requires to set up a bidirectional tunnel between previous and new PoS on demand.

The estimated mean number of lost packets is shown in Fig. 5.28 for Active MPLS, Fast Active MPLS and Low Latency Handoffs. The left diagram depicts the estimated mean number of lost packets in the CBR traffic scenario, the right diagram the estimated mean number of lost packets in the video traffic scenario. When looking at one scenario separately, the estimated mean number of lost packets is about equal across the three mobility management approaches during the three intra-PoS handovers and considerably higher for Low Latency Handoffs than for Active MPLS and Fast Active MPLS during the two inter-PoS handovers. The reason becomes evident when looking at the data loss periods in the message sequence diagrams of Fig. 5.17, Fig. 5.18 and Fig. 5.15. In Low Latency Handoffs, the data packets cannot be redirected before the previous PoS has received a message from the new PoS requesting to set up the corresponding IP tunnel, whereas in Active MPLS and Fast Active MPLS the data packets can be redirected on receipt of the initial DHCPINFORM message (cf. Fig. 5.17) at the previous PoS. The estimated mean number of lost packets is different in the two scenarios since it is dependent on the actual data rates used. In the CBR traffic scenario, the estimated mean number of lost packets equals about 10 packets for Active MPLS and Fast Active MPLS during all handovers and for Low Latency Handoffs during the three intra-PoS handovers and about 29 packets for Low Latency Handoffs during the two inter-PoS handovers. The 0.95 confidence intervals amount to about ±0.03 packets for both Active MPLS, Fast Active MPLS and Low Latency Handoffs. In the video traffic scenario, the estimated mean number
5.6. EVALUATION

Figure 5.28: Absolute packet loss for constant bit rate (left diagram) and video traffic (right diagram)

of lost packets equals about 2 packets for both Active MPLS and Fast Active MPLS across all handovers and for Low Latency Handoffs during the three intra-PoS handovers and about 6 packets during the two inter-PoS handovers. The 0.95 confidence intervals amount to about ±0.26 packets for both Active MPLS and Fast Active MPLS and to about ±0.58 packets for Low Latency Handoffs.

The estimated mean percentage of lost packets is shown in the left diagram of Fig. 5.29 for the CBR traffic scenario and in the right diagram for the video traffic scenario. The estimated mean percentage of lost packets is depicted for Active MPLS, Fast Active MPLS and Low Latency Handoffs. Note that the percentage of lost packets is computed by dividing the number of lost packets by the number of all packets affected by the handover. In order to make the percentages of the three approaches comparable to one another, the number of all packets affected by the handover should be the same. Therefore, the handover delay of Active MPLS (the longest handover delay) is taken into account in order to obtain the number of all packets affected by the handover.

In both scenarios, the estimated mean percentage of lost packets is considerably higher for Low Latency Handoffs during the two inter-PoS handovers than for Active MPLS and Fast Active MPLS. In the CBR traffic scenario, the estimated mean percentages of lost packets equal about 10 % for Active MPLS and Fast Active MPLS across all handovers. For Low Latency Handoffs, they equal about 10 % during the three intra-PoS handovers and about 27 % during the second and about 33 % during the fourth handover. The difference results from the differently long handover delays (cf. Fig. 5.26). The 0.95 confidence intervals amount to about ±0.03 % for both Active MPLS, Fast Active MPLS and Low Latency Handoffs. In the video traffic scenario, the estimated mean percentages of lost packets lie between 10 % and 14 % for both Active MPLS and Fast Active MPLS during all handovers. For Low Latency Handoffs, they lie in the same range during the three intra-PoS handovers, but equal about 30 % during the
second and 35 % during the fourth handover. The reason is the same as in the CBR traffic scenario. The 0.95 confidence intervals amount to about \(\pm 1.9 \%\) for both Active MPLS and Fast Active MPLS and to about \(\pm 3.5 \%\) for Low Latency Handoffs.

### 5.6.3.2 Packet-related parameters

The end-to-end delay for Active MPLS, Fast Active MPLS and Low Latency Handoffs in Mobile IPv4 is shown in Fig. 5.30 for the second handover and in Fig. 5.31 for the fourth handover. The second handover is performed from R11 to R12 (cf. Fig. 5.16), the fourth handover from R13 to R14. Note that the jitter is not evaluated since the results of the testbed implementation (cf. Section 5.6.2) already show that the jitter is primarily 0 for all packets affected by a handover.

The upper left diagram of Fig. 5.30 shows the end-to-end delay for Active MPLS in the CBR traffic scenario, the upper right diagram the end-to-end delay in the video traffic scenario. Basically three peaks can be seen. Each peak results from a certain set of packets. The first peak results from those packets that are sent along the new primary LSP over R2 and R5 to R8 and those packets that are redirected at R2 (over R5) to R8. The path length is three in both cases. The second peak results from those packets that are redirected at R4 over R5 to R8. The path length is four. The last peak results from those packets that are redirected at the previous egress LER R7. The path length is six. The confidence intervals are depicted in the diagrams.

The middle left diagram of Fig. 5.30 shows the end-to-end delay for Fast Active MPLS in the CBR traffic scenario, the middle right diagram the end-to-end delay in the video traffic scenario. Only two peaks can be seen. The first peak results from those packets that are sent along the new primary LSP towards R8. The second peak results from those packets that are redirected at the previous egress LER R7. These two peaks clearly show the benefit of Fast Active MPLS with
5.6. EVALUATION

Figure 5.30: End-to-end delay for Handover 2 in Active MPLS, Fast Active MPLS and Low Latency Handoffs in Mobile IPv4 evaluated for both constant bit rate and video traffic

respect to Active MPLS in the two scenarios since the shorter handover delay leads to a larger set of packets sent via the new primary LSP. The confidence intervals are depicted in the diagrams.

The lower left diagram of Fig. 5.30 shows the end-to-end delay for Low Latency Handoffs in the CBR traffic scenario, the lower right diagram the end-to-end delay in the video traffic scenario. Here, too, two peaks can be seen. The first peak results from those packets that are tunneled from the Home Agent along the new
Figure 5.31: End-to-end delay for Handover 2 in Active MPLS, Fast Active MPLS and Low Latency Handoffs in Mobile IPv4 evaluated for both constant bit rate and video traffic.

path towards R8. The second peak results from those packets that are redirected at the previous egress LER R7 over the established IP tunnel towards R8. Note that the absolute end-to-end delay values are higher for Low Latency Handoffs than for Active MPLS and Fast Active MPLS since the byte overhead and thus the serialization delay is higher. The confidence intervals are depicted in the diagrams.

The diagrams of Fig. 5.31 showing the end-to-end delay for the fourth han-
dover are very similar to those of Fig. 5.30. The only difference can be seen in the diagrams showing the end-to-end delay for Active MPLS. Although three peaks can be seen as well, the first peak only results from those packets sent along the new primary LSP towards R9 (cf. Fig. 5.16). The path length is three. The second peak results from those packets that are either redirected at R2 over R3 and R6 to R9 or at R5 over R6 to R9. In both cases, the path length is 4. The last peak results from those packets that are redirected at R8. The path length is six. Overall, the benefit of Fast Active MPLS with respect to Active MPLS and Low Latency Handoffs is clearly visible for both inter-PoS handovers.

5.6.4 Summary

The testbed implementation has shown (cf. Section 5.6.2) that Active MPLS and Fast Active MPLS can both be implemented on real hardware. In comparison to Ingress-Update the main result is that the packet loss occurring with Ingress-Update can be reduced by about 60% during inter-PoS handovers when deploying Active MPLS or Fast Active MPLS.

The simulation has been performed in order to evaluate Active MPLS and Fast Active MPLS in a larger network architecture and to compare it with an IP-based approach that could not be implemented on the IXP1200 network processors due to code size limitations. The evaluation has shown (cf. Section 5.6.3) that Active MPLS and Fast Active MPLS outperform Low Latency Handoffs in Mobile IPv4 in the outage period (reduction by about 40%) and the packet loss during the inter-PoS handovers (reduction by about 60%). The mean end-to-end delay is also lower. Note that these results are a lower bound on the gain of Active MPLS and Fast Active MPLS since the acquisition of a new IP address as well as Duplicate Address Detection which cannot be avoided when acquiring a new IP address have not been simulated. The actual gain of Active MPLS and Fast Active MPLS with respect to outage period and packet loss can thus be expected to be much higher.

Since Fast Active MPLS is a special case of Active MPLS, these two approaches have been compared analytically (cf. Section 5.6.1). The main result of the analytical comparison is a formula with which it can be determined whether it is more beneficial to deploy Active MPLS or Fast Active MPLS. The formula is based on the fact that the difference between Active MPLS and Fast Active MPLS primarily translates into the lengths of the paths over which the user plane data packets are sent during a handover. Therefore, the mean path length has been chosen as criterion although other criteria are conceivable. In current core networks with high-capacity links and devices that process user plane data packets at much higher speeds than control plane signalling messages, the formula yields that it is more beneficial to deploy Fast Active MPLS. In purely satellite-based networks that do not yet exist today but that are possible future networks, it is, however, more beneficial to deploy Active MPLS.
Chapter 6

Wireless Technologies

Although many different wireless technologies operating on different frequency bands and providing different ranges and data rates exist, the focus of this chapter is on Wireless LANs (cf. Section 6.1), on WiMAX (cf. Section 6.2) and on GPRS and UMTS (cf. Section 6.3). These wireless technologies have in common that the link between the wireless device and the network is wireless while the links within and between the access and core networks are generally wired. The focus of this chapter is also on mobile devices and not on stationary devices. Wireless networks providing services to mobile devices are sometimes also referred to as mobile wireless networks, whereas those providing services only to stationary devices are sometimes also referred to as fixed wireless networks. Each of the three technologies is presented through a description of its architecture, the way a mobile device attaches to the network and the way a handover is performed with Active MPLS. The intention is to show the applicability of Active MPLS to a variety of wireless technologies.

6.1 Wi-Fi

A simple short-range broadband wireless access technology has been developed by the IEEE under the name IEEE 802.11. The original standard specifies data rates of 1 Mbit/s and 2 Mbit/s and an operation at 2.4 GHz. In the same frequency band, IEEE 802.11b has been developed for data rates of 5.5 Mbit/s and 11 Mbit/s, and IEEE 802.11g for data rates of up to 54 Mbit/s. IEEE 802.11a also allows for data rates of up to 54 Mbit/s, but operates at a frequency of 5 GHz. The term Wireless Fidelity (Wi-Fi) denotes the set of all products based on one of the IEEE 802.11 standards. A good introduction to IEEE 802.11 is given by [66, 160].
6.1.1 Architecture

The IEEE 802.11 architecture consists of three entities. These are the Mobile Node (MN), the Access Point (AP) and the Access Router (AR). The AP is the Point of Attachment (PoA) for the MN, the AR is the Point of Service (PoS) for the MN. An AP belongs to one and only one network. In general, the network is based on the Internet Protocol (IP) in which case it is referred to as subnet. Note that the link between MN and AP is wireless while the link between AP and AR is generally wired.

An AR may belong to more than one subnet. In that case, the AR has an IP address for each subnet it belongs to. Each subnet may contain more than one AP and more than one AR. Fig. 6.1 shows the different cases in an example. AR1 belongs to the first subnet, AR3 to the second, and AR2 to both. AP1 belongs to the first subnet while AP2 and AP3 belong to the second.

In wireless environments, security is even more important than in wired environments where many attacks are prevented simply because an attacker cannot physically access the network. This is, however, different in wireless environments. Therefore, security services are incorporated into the IEEE 802.11 MAC layer.

However, EAP (cf. Section 3.4) is not supported by the original IEEE 802.11 standards which gave rise to the development of the amendment IEEE 802.11i [78]. Further information on deploying EAP over IEEE 802.11 can be found in [147]. In IEEE 802.11i, the AP operates as EAP authenticator in pass-through mode relaying authentication messages between the MN and a backend Authentication Server (AS).

Consequently, the AP implementation has to include the AAA protocol for
communication with the AS which increases the development cost for such a low-cost device. Since the backend AS is in general located in the core network, a further issue arises. In the case that the access network is owned by a different operator than the core network, the access network operator will probably not tolerate an AS in the core network. In order to avoid these issues, authentication can directly be performed with the AR as part of the core network. This is in general necessary anyway when moving between access networks. An authentication with the core network can be achieved via the AR that may take the role of an EAP authenticator. In that case, the protocol between MN and AR should be PPP allowing for the use of EAP as authentication protocol. IEEE 802.11i cannot be used since it is restricted to the link between MN and AP. The above deliberations lead to two different scenarios. In the first scenario, the AP takes the role of the EAP authenticator so that IEEE 802.11i is used between MN and AP. In the second scenario, the AR takes the role of the EAP authenticator so that IEEE 802.11 is used between MN and AP and PPPoE between MN and AR. These two scenarios are further described in Section 6.1.2.

6.1.2 Protocol Stacks

If the IEEE 802.11i amendment is used in which the AP takes the role of the EAP authenticator, the protocol stack looks as depicted in Fig. 6.2. Note that no distinct protocol stacks are shown for the user and the control planes since this protocol stack is used by any higher-layer protocol whether it belongs to the user plane or to the control plane.

On the air interface between MN and AP, the IEEE 802.11 physical and MAC layer with the IEEE 802.11i amendment are deployed. Since an IEEE 802.11 MAC layer frame does not include a field to identify the network-layer protocol, a separate Logical Link Control (LLC) [75] protocol header is necessary along with a SubNetwork-Dependent Application Protocol (SNAP) [127] protocol header. In this case, the network-layer protocol is identified as IP.

The AP operates as a bridge between two link-layer technologies. On the air interface, it implements IEEE 802.11, while it implements a different link-layer technology on the wired interface. This may e.g. be IEEE 802.3, Ethernet or some other link-layer technology. Therefore, the respective protocol entries are
given as L2.

If the IEEE 802.11i amendment is not used, the MN uses PPP authentication with the AR. The network-layer protocol identified by LLC/SNAP is PPPoE in that case. PPPoE is explained in more detail in Section 6.1.3. Fig. 6.3 shows the corresponding protocol stack. Here, too, the AP operates as simple bridge between two link-layer technologies.

6.1.3 PPPoE

The motivation to connect a MN via PPP (cf. Section 2.2) instead of via Ethernet to an AR is that PPP provides session control via connection establishment and termination which is necessary for any time-based tariffs. It also provides user authentication with username and password which is necessary for accounting. For tariffs based on volume only, session control would not be necessary provided that the user could be identified by means other than username and password. However, that is in general much more complicated than simply using PPP.

Since PPP is based on a peer-to-peer relationship and since Ethernet or IEEE 802.11 is based on a shared medium, a protocol is needed to establish a peer-to-peer relationship over a shared medium. This is done by PPP over Ethernet (PPPoE).

Although PPPoE is only standardized for Ethernet, it can be deployed on a MN which is connected by an IEEE 802.11 Network Interface Card (NIC) to an AP since the IEEE 802.11 NIC driver generally provides an Ethernet interface towards the operating system.

PPPoE is specified in [99]. The objective of PPPoE is to establish a peer-to-peer relationship over Ethernet. The peer-to-peer relationship is established through the PPPoE discovery stage. Both peers identify their respective MAC addresses and set up a common session identifier to uniquely identify the subsequent PPP session. A PPP session is defined by the MAC addresses of the two peers and the common session identifier. Note that the peer of the MN is called Access Concentrator (AC) in PPPoE.

If PPPoE is to be used over a certain link, it is generally configured for the corresponding NIC in the MN. As soon as the link is activated, the MN starts...
with the discovery stage (cf. Fig. 6.4) and broadcasts a PPPoE Active Discovery Initiation (PADI) message requesting a particular service. Each AR capable of offering the requested service may reply with a PPPoE Active Discovery Offer (PADO) message offering the requested and any number of other services. The PADO is sent unicast to the MN having sent the PADI. Based on the received offers, the MN selects one of the ARs for further communication and sends a unicast PPPoE Active Discovery Request (PADR) to the selected AR. The selected AR finally replies with a unicast PPPoE Active Discovery Session-Confirmation (PADS) message as confirmation. The PADS contains a session identifier chosen by the AR to uniquely identify the just established session between the MN and the AR. To later terminate a session, a unicast PPPoE Active Discovery Terminate (PADT) message containing the chosen session identifier may be sent anytime by the MN or the AR.

After the PPPoE discovery stage, the MN starts the PPPoE session stage. All packets sent in the PPPoE session stage are sent as unicast packets between the two peers. The PPPoE session stage simply consists of both peers performing the standard PPP message exchange (cf. Section 2.2). Note that no particular PPP encapsulation such as HDLC-like framing is needed with PPPoE. Note also that the MRU negotiated by PPP must not be negotiated to a size larger than 1492 octets since the Ethernet payload is bounded by 1500 octets and since the PPPoE header and the PPP Protocol field are 6 and 2 octets long, respectively.

6.1.4 Discovery

Network discovery is called scanning in IEEE 802.11. Each AP periodically transmits beacon frames containing the values for a couple of subnet-specific parameters. Such parameters are the Service Set ID (SSID), which is an identifier of the subnet the AP belongs to, the beacon interval, a timestamp, etc. The default value for the beacon interval is generally set to 100 ms. Since the beacon frames are transmitted by the AP on one of several parallel channels, a MN has to scan each channel for beacon frames. There are two modes for performing a scan.
CHAPTER 6. WIRELESS TECHNOLOGIES

These are passive scan and active scan.

In passive scan, the MN switches to the corresponding channel, starts a timer and waits for a beacon frame. If the timer expires before a beacon frame is received, the MN concludes that the channel is not in use.

In active scan, the MN switches to the corresponding channel, starts a probe delay timer and waits for a beacon or another frame. The probe delay timer prevents the MN from waiting indefinitely. If the probe delay timer expires and if the channel has continuously been idle, the MN concludes that the channel is not in use and moves to the next channel. If a beacon frame is received, the scan has been successful and the MN switches to the next channel. If a frame other than a beacon is received before the probe delay timer expires, the MN concludes that the channel is in use. It tries to gain access to the medium using the common access procedure and sends a Probe Request frame. It then starts two other timers and waits for a Probe Response frame. The first timer is initialized with MinChannelTime, the second timer with MaxChannelTime. If the MinChannelTime timer expires and if the channel has continuously been idle, the MN concludes that it has mistakenly chosen this channel and moves to the next channel. If, however, the channel has been busy, the MN continues waiting. If the MaxChannelTime timer expires and if the MN has not received a Probe Response frame, it concludes that the channel is not in use and moves to the next channel. Otherwise, the scan has been successful and the MN switches to the next channel. Note that the choice of MinChannelTime and MaxChannelTime has considerable impact on the total scanning time.

The MN decodes each beacon or Probe Response frame it receives and places these information into a scan report. Note that the Probe Response frame basically contains the same information as a beacon frame. The AP the MN is to associate with is either chosen by user intervention or automatically with respect to a certain policy.

6.1.5 Start-Up

After the MN has found the subnet to join, it has to authenticate and associate with the AP it has received the beacon frame from (cf. Fig. 6.5). If no authentication with the access network is desired, Open System Authentication may be performed. Open System Authentication is also performed when IEEE 802.11i is used. The association serves to register the MN with the AP so that frames arriving for the MN can be bridged appropriately onto the wireless interface.

If IEEE 802.11i is used, the EAP authentication is performed after successful IEEE 802.11 authentication and association. Note that IEEE 802.11i is based on Port-Based Network Access Control also referred to as IEEE 802.1X [76], offering EAP transport with the EAP over LAN (EAPOL) protocol. To initiate the EAP message exchange, the AP sends an EAPOL/EAP frame in general requesting for the identity of the MN. The MN responds with an EAPOL/EAP frame that
is relayed by the AP to the AS. RADIUS-EAP (cf. Section 3.6) may be deployed for this purpose. Some guidelines to use RADIUS together with IEEE 802.1X can be found in [42].

Then the EAP authentication takes place during which both the MN and the AS derive a common Pairwise Master Key (PMK). If the EAP authentication terminates successfully, the AS transfers the PMK to the AP. Even then, the MN is not yet authorized to access the access network. Before authorizing the MN, a Pairwise Transient Key (PTK) has to be derived between MN and AP that is based on two random values ANonce and SNonce. Only if the PTK has successfully be installed at the AP and the MN, the MN is authorized to access the access network, i.e. the AP unblocks the corresponding port.

In order to provide the MN with all necessary configuration information such as IP address, IP address of default router, etc., an appropriate signaling protocol such as DHCP (cf. Fig. 2.3.3) may be used. The registration of the MN’s current location is up to the deployed L3 mobility solution and thus not further described here.
6.1.6 Handover

In general, the handover is both mobile-initiated and mobile-controlled, i.e. the 
MN initiates the handover and determines which AP to handover to. The decision 
as to which AP to handover to is based on the scan report (cf. Section 6.1.4) that 
is in general continuously updated while the MN is associated with the current 
AP.

When the MN moves from one AP to another, it has to disassociate from the 
old AP before re-associating with the new AP. This is done automatically by the 
NIC. If both APs are located in the same subnet, the Inter-Access Point Protocol 
(IAPP) [77] serves to update the location of the MN within the access network. 
This is mainly done by broadcasting a L2 Update Frame within the subnet as 
soon as the MN associates or re-associates with an AP.

The handover process has been analyzed by [106]. The paper proposes ba-
sically two approaches to reduce the time for a link-layer handover based on 
IEEE 802.11. One approach tries to reduce the probe delay by better estimat-
ing the MinChannelTime and MaxChannelTime parameters. This approach has 
been further developed by [153]. Another approach (cf. e.g. [140]) tries to reduce 
the number of active scans during a handover. This is achieved with the help of 
a neighbor graph revealing which APs a MN may move to and which channels 
these APs transmit their beacon frames on. Equipped with this information, 
the MN only has to probe the corresponding channels for beacon frames. The 
neighbor graph may be retrieved after associating with an AP from the AP itself 
or from a centralized server. It may also be combined with the mobility graph 
representing the movement pattern of the MN over time. In [74], whose title is a 
little misleading since the focus is not on Mobile IP but on improving the link-
layer handover latency, the handover is still mobile-initiated, but the decision as 
to which AP to handover to is incumbent to the network. Therefore, a movement 
tracking system is developed supporting this decision.

Note that when moving between APs belonging to different subnets, a L3 
mobility solution is necessary since the scope of the IAPP does not go beyond a 
broadcast domain, i.e. not beyond a single subnet. In function of the L3 mobility 
solution used, the MN has to configure a new IP address for the new subnet or 
not. Since no L3 mobility solution is specified to be used with Wi-Fi, any L3 
mobility solution may be deployed without necessitating any specific changes to 
the Wi-Fi architecture.

6.1.7 Application of Active MPLS to Wi-Fi

6.1.7.1 Start-Up

After the MN has found the network to join, it has to authenticate and associate 
with the AP it has received the beacon frame from (cf. Fig. 6.6). If no authenti-
cation with the access network is desired, Open System Authentication may be
If authentication with the core network is desired, PPP may be deployed over IEEE 802.11 using PPPoE and LLC/SNAP encapsulation as shown in Fig. 6.3. In order to be capable of negotiating an arbitrary authentication protocol, EAP is negotiated during the PPP link establishment phase.

In general, the AR is deployed in pass-through mode so that EAP authentication involves a back-end authentication server. If RADIUS (cf. Section 3.5) is used, RADIUS/EAP (cf. Section 3.6) has to be used as well. The AR is provided with an IP address and the subnet mask for the MN. In the subsequent PPP network-layer protocol phase based on IPCP, the IP address can thus directly be returned by the AR.

Finally, DHCP may be performed to obtain further configuration information such as the IP address of a default router and the subnet mask. The subnet mask has been received from the RADIUS server by the AR during the PPP authentication phase and can at this point be returned to the MN. The IP address of a default router may be one of the IP addresses of the AR itself.

Once the MN is configured, it has to be registered in the core network. As the core network is based on Active MPLS, RSVP-TE is proposed to be exploited for registration. An additional REGISTRATION object should be introduced for that purpose. The trigger for registration is proposed to be the DHCPINFORM message. The MN has to be registered with any potential ingress LER. For a discussion about the different modes, confer to Section 5.2.7.

Note that Fig. 6.6 is based on IPv4. In the case of IPv6, IPv6CP [70] has to be used instead of IPCP and DHCP has to be replaced by either stateless address
autoconfiguration (cf. Section 2.4.3) or by stateful address autoconfiguration, i.e. by DHCPv6 (cf. Section 2.4.4).

6.1.7.2 Handover

The predictive mode of Active MPLS is based on link-layer triggers, which has been elaborated by [122]. On receipt of a corresponding Link-Going-Down trigger, the MN sends a network-layer handover indication message to the old AR in the mobile-initiated case (cf. Fig. 6.7). In the network-initiated case, the Link-Going-Down trigger occurs at the old AR that may then force the MN to handover, e.g. through an appropriate IEEE 802.21 command (cf. Section 7.2). The network-layer handover indication message is not sent in the network-initiated case.

The network-layer handover indication is encoded as a DHCPINFORM message extended by a HO option. The DHCPINFORM message is acknowledged by a DHCPACK. On receipt of the Link-Going-Down trigger or the DHCPINFORM message, the old AR triggers the Active MPLS handover procedure as described in Chapter 5. It also contacts the new AR for transfer of the mobility...
6.2. WiMAX

One main characteristic of IEEE 802.11 is the relatively short range of the APs. Therefore, IEEE 802.11 has only been exploited for WLANs. In a Metropolitan Area Network (MAN), wireless transmission is addressed by IEEE 802.16. IEEE 802.16 is specified by the IEEE Working Group 802.16 on Broadband Wireless Access (BWA). The main incentive for the development of IEEE 802.16 has been the high cost of wired technologies for covering a larger geographical area with cables such as in Digital Subscriber Line (DSL) environments.

The first IEEE 802.16 standard has been accomplished in 2002 and was designed for fixed wireless networks in the 10-66 GHz frequency range necessitating Line-Of-Sight (LoS) propagation. The corresponding air interface is called WirelessMAN-SC. The IEEE 802.16a amendment has extended the frequency
range with the 2-11 GHz frequency range. The incentive for this amendment has been the need for Non-Line-Of-Sight (NLOS) propagation to reach individual users as well. IEEE 802.16a includes three different air interface specifications. These are called WirelessMAN-SC2, WirelessMAN-OFDM and WirelessMAN-OFDMA. The last amendment called IEEE 802.16e specifies all necessary changes for mobile wireless networks.

Since the IEEE 802.16 standards only cover the physical and MAC layers of the air interface, an industry-led forum has been founded in 2001. The forum is called Worldwide inter-operability for Microwave Access (WiMAX) forum and aims at specifying a standard based on IEEE 802.16 that ensures compatibility and inter-operability among operators of networks based on IEEE 802.16 and equipment and component suppliers of IEEE-802.16-based equipment.

WiMAX-certified systems are expected to deliver data rates of up to 40 Mbit/s per user for fixed and portable access applications. Mobile network deployments are expected to provide up to 15 Mbit/s per user within a typical cell radius deployment of up to three kilometers.

6.2.1 Architecture

A WiMAX network can be subdivided into an Access Services Network (ASN) and a Core Services Network (CSN). A CSN is in general operated by a Network Service Provider (NSP), while an ASN is in general operated by a Network Access Provider (NAP). A NSP may be operating one or more CSNs. A NAP may be operating one or more ASNs. A certain operator may also be NSP and NAP at the same time. A particular ASN may be connected to one or more CSNs. The case where it is connected to more than one CSN is referred to as ASN sharing since more than one CSNs and possibly more than one NSPs share the same ASN.

Fig. 6.8 shows one CSN and two ASNs that are both connected to the CSN. An ASN is managed by an ASN Gateway (ASN-GW). The ASN-GW is in charge of one or more Base Stations (BSs). The BS implements the IEEE 802.16e air interface that is also implemented on the NIC of the Mobile Subscriber Station (MSS). The ASN-GW is responsible for intra-ASN mobility, for inter-ASN mobility, for billing and other tasks. Fig. 6.8 also shows the main reference points between the different network entities. A reference point defines the end-point in the corresponding protocols, but not the data path the protocol messages take.

Reference point R1 defines the air interface between MSS and BS. Reference point R6 defines the link between BS and ASN-GW. In the case that BS and ASN-GW are implemented on the same device, R6 may be implemented in terms of an Application Programming Interface (API), otherwise in terms of one or more protocols. The reference point R3 defines the link between ASN and CSN. In order to support mobility, two horizontal reference points are specified. Reference point R4 in general defines inter-ASN mobility, while reference point R8 in general defines intra-ASN mobility.
6.2. WIMAX

6.2.2 Protocol Stacks

The MSS uses DHCP for host configuration. If Client MIP (CMIP) is used as CSN mobility solution, it uses MIP as protocol for support. If Proxy MIPv4 (PMIPv4) is used as CSN mobility solution, it uses DHCP as protocol for mobility support. Both DHCP and MIP run over IP and UDP. The physical and MAC layers are specified by IEEE 802.16e.

The BS terminates the IEEE 802.16e physical and MAC layers. In order to keep the address space for user IP addresses independent from the transportation within the access network, some protocol is needed to transport user IP packets between the ASN-GW and the BS. This protocol is not specified in the WiMAX draft. Different protocols are conceivable such as IP, MPLS or even RADIUS.

Fig. 6.9 shows the WiMAX protocol stack on the control plane. Since the WiMAX architecture is based on IEEE 802.16e, the IEEE 802.16e MAC layer is explained in more detail in the following. A good overview of IEEE 802.16 is given by [56]. More information on the physical layer can be found in [73].

The IEEE 802.16e MAC layer is subdivided into three sublayers (cf. Fig. 6.10). The highest sublayer is the Service-Specific Convergence Sublayer (SSCS). The intermediate sublayer is the MAC Common Part Sublayer. The lowest sublayer is the privacy sublayer.

The SSCS is responsible for mapping network-layer services to the
IEEE 802.16 MAC layer. There are two general SSCSs. The ATM SSCS is used for mapping ATM services to the IEEE 802.16 MAC layer, while the packet SSCS is used for mapping packet services such as Ethernet, IPv4, IPv6, etc. Since the IEEE 802.16e MAC is connection-oriented, the main task of the SSCSs is to map the network-layer services to the appropriate MAC connection.

A MAC connection is identified by a Connection ID (CID) and associated with a set of QoS parameters etc. Although the MSS has a standard 48-bit MAC address, the CIDs of the connections are used as identification. When entering an IEEE 802.16e network, the MSS is assigned three management connections. Since a MAC connection is unidirectional, these management connections are assigned in each direction, i.e., three management connections for the uplink and three management connections for the downlink.

The three management connections are the basic connection, the primary management connection and the secondary management connection. DHCP messages are e.g., assigned to the secondary management connection.

For user data, additional transport connections are allocated to a MSS. Some CIDs common to all MSSs are reserved for different purposes, e.g., for contention-based initial access or for broadcast downlink messages.

There are three types of MAC Packet Data Unit (PDU) (cf. Fig. 6.11). The bandwidth request MAC PDU contains a Bandwidth Request Header and no payload. The other types of MAC PDU contain a Generic MAC Header (GMH), a variable-length payload and an optional Cyclic Redundancy Check (CRC). Such MAC PDUs may either carry a MAC management message or SSCS data issuing from the corresponding network-layer. The connection the MAC PDU belongs to is encoded in the GMH with the 16-bit CID.
The privacy sublayer serves for authentication of the MSS to the BS as well as for the establishment of a secure connection between the two entities. Authentication is based on X.509 certificates that are sent along with the MSS’s public key to the BS. The secure connection is established via the Privacy Key Management (PKM) protocol defined by IEEE 802.16. After IEEE 802.16 has been found to offer an insufficient level of security [85], IEEE 802.16e specifies PKMv2 as enhancement of PKM. PKMv2 allows for EAP-based authentication using PKMv2 as the underlying message transport protocol. Through the use of EAP (cf. Section 3.4) any authentication method can be deployed. The WiMAX security architecture is shown in Fig. 6.12.

With respect to the EAP architecture described in [10] the MSS is the supplicant, the ASN-GW is the authenticator operating in pass-through mode and the AAA server is the backend authentication server. Note that the BS only acts as authentication relay and does not actively participate in the EAP communication. However, the privacy sublayer of the BS has to be provided with the Pairwise Master Key (PMK) derived during the EAP authentication procedure in order to ensure a secure connection between MSS and BS.

The protocol stack on the user plane is very similar to that of the control plane and given for completeness (cf. Fig. 6.13).

### 6.2.3 Discovery

Network discovery in WiMAX environments can be subdivided into NAP discovery and NSP discovery.
A MSS detects available NAPs by scanning and decoding the Downlink Map (DL-MAP) of a downlink broadcast frame sent on the basic connection. The DL-MAP contains a Base Station ID parameter with a 24-bit Operator ID which is considered as NAP identifier.

NSP discovery is done in a subsequent step. Static NSP discovery is done through a list of NAP-NSP mappings stored in the MSS with a list of NSP identifiers associated with each NAP. Dynamic NSP discovery is done through a list of NSP identifiers broadcast by the BS in the System Information Identity message. Note that none of the lists of NSP identifiers has to be complete.

If more than one NSP is discovered, the MSS selects the most preferred NSP based on policy, profiles or user intervention. After selection of an NSP, the MSS attaches to an ASN associated with the selected NSP.

### 6.2.4 Application to Active MPLS

#### 6.2.4.1 Start-Up

The WiMAX start-up procedure (cf. Fig. 6.14) begins with the MSS scanning for an appropriate BS (cf. Section 6.2.3) and associating with that BS. Since the air interface of WiMAX equipment is a WirelessMAN air interface, the association is based on IEEE 802.16e. As described in Section 6.2.2, the IEEE 802.16e MAC layer is composed of three sublayers one of which is the privacy sublayer dealing with MSS authentication. The MSS authentication is based on the Privacy Key Management version 2 (PKMv2) protocol that makes use of EAP (cf. Section 3.4) allowing for any EAP method. The most suitable EAP method is EAP-TLS since it offers the greatest possible set of security services.

Since the BS only acts as an authentication relay, some authentication relay protocol has to be used in order to relay EAP messages to the ASN-GW. This protocol is not yet specified in the current WiMAX draft. The ASN-GW finally makes use of an AAA protocol to access the backend AAA server. RADIUS (cf. Section 3.5) or Diameter (cf. Section 3.7) are suitable for this purpose. In Fig. 6.14, RADIUS is used. At the end of the RADIUS-EAP message exchange, the RADIUS server delivers an IP address for the MSS and a Pairwise Master
Key (PMK) along with other information to the ASN-GW. The PMK is relayed to the BS via the authentication relay protocol and installed for use by EAP and PKMv2.

In order to transport subsequent messages issued to and from the MSS, an R6 tunnel needs to be established between BS and ASN-GW. Note that the procedure for R6 tunnel establishment depends on the protocol used. Note also that the tunnel does not need to be a MSS-specific tunnel. In contrast to MSS-specific tunnels, such tunnels may be pre-established.

At this point, the MSS still needs to configure the network-layer, i.e. acquire an IP address and other network-layer configuration information. The protocol proposed by the WiMAX forum is DHCP (cf. Section 2.3.3). In principle, there are two possible scenarios. Either the ASN-GW acts as DHCP relay and relays all DHCP messages to and from a DHCP server not depicted in Fig. 6.14 or the ASN-GW acts as a DHCP proxy and assigns the IP address received for the MSS during the authentication exchange with the RADIUS server.

Once the DHCP message exchange is complete, the ASN-GW needs to register the MSS in the CSN unless the CSN is configured in such a way that it forwards IP packets destined to the MSS to the ASN-GW the MSS has associated with. This would be possible by appropriate IP address assignment and an IP-based CSN. In order to deploy Active MPLS, however, the CSN is based on MPLS. In that case, the CSN edge routers need to be informed of the current location of the MSS in order to place incoming IP packets on the proper LSPs. The most suitable protocol for this task is RSVP-TE (cf. Section 2.5.6). The necessary extensions are described in Chapter 5.

6.2.4.2 Handover

The WiMAX handover procedure is triggered whenever the MSS needs to change the BS it is associated with. The type of handover is determined by the administrative area the BS belongs to. Therefore, the handover may be an intra-ASN handover, an inter-ASN handover, an intra-CSN handover, an inter-CSN han-
dover, an intra-NAP handover, an inter-NAP handover, an intra-NSP handover, an inter-NSP handover, an intra-ASN-GW handover or an inter-ASN-GW handover. In function of the respective administrative areas in a real-life scenario, some of these notions may describe the same type of handover.

Note that in the WiMAX terminology, handovers are often referred to by the reference points involved. The term R3 handover is a commonly used term there. However, the fact that the R3 reference point is involved rather denotes the way the handover is implemented than the type of handover itself. Therefore, such notions will not be used within this document. The handover that is described here is an intra-CSN and inter-ASN handover. Such handovers are restricted to a single domain and may thus be solved with micro-mobility solutions. Furthermore, they cannot be solved by a link-layer solution since they extend beyond a single ASN.

The WiMAX handover procedure for this type of handover where Active MPLS is used as micro-mobility solution is depicted in Fig. 6.15.

The movement to another BS may be mobile-initiated or network-initiated. In Fig. 6.15, the mobile-initiated case is depicted. The MSS sends an IEEE 802.16e MOB_MSSH0_REQ message and includes a list of candidate BSs to which it may handover to. The old BS (oBS) simply relays this request over the R6 tunnel established with the old ASN-GW (oASN-GW). From the R6 HO Request message, the oASN-GW determines the new ASN-GW (nASN-GW) responsible for the new BS (nBS) the MSS is moving to. It sends an R4 HO Request message to the nASN-GW and includes a MSS ID so that the nASN-GW is aware of which MSS is possibly moving to one of the BSs within its ASN.

On receipt of the R4 HO Request message, the nASN-GW pre-notifies the BSs contained in the list of candidate BSs of the imminent HO. This is done to determine which of the candidate BSs have enough resources available to accept another MSS. Each BS responds with an R6 HO Pre-Notification Ack message.

The nASN-GW collects the responses of all BSs it sent the R6 HO Pre-Notification message to and creates a list of recommended BSs that is based on the contents of the response messages. The nASN-GW then sends an R4 HO Response message back to the oASN-GW.

On receipt of the R4 HO Response message, the oASN-GW may select one of the BSs within the list of recommended BSs (control on the network side) or leave the list unmodified (control on the MSS side). It then sends an R6 HO Response message back to the oBS where it is relayed as IEEE 802.16e MOB_BSHO_RSP message to the MSS.

In Fig. 6.15, the control lies on the MSS side. Therefore, the MSS selects one of the recommended BSs where to handover to. This BS is the target BS. In the predictive mode of Active MPLS, the MSS now sends a DHCPINFORM message towards the oASN-GW. The DHCPINFORM message contains the HO object introduced in Section 6.1.7.1. On receipt of the DHCPINFORM message, the oASN-GW starts the Active MPLS procedure described in Chapter 5. Note
that the MSS does not have to wait on the DHCPACK sent by the oASN-GW. If it receives the DHCPACK, it knows that the predictive mode of Active MPLS has successfully been triggered. Otherwise, it needs to initiate the reactive mode of Active MPLS after having re-associated with the target BS.

The MSS then initiates the link-layer handover by sending an IEEE 802.16e MOB_HO_IND message specifying the target BS to the oBS. The oBS relays the message to the oASN-GW that sends an R4 HO Confirm message to the nASN-GW. If possible, the R4 HO Confirm message is extended with the MSS contexts. Otherwise, separate context transfer messages would be required which could be done with RADIUS.
On receipt of the \textit{R4 HO Confirm} message, the nASN-GW confirms the HO with the target BS by sending an \textit{R6 HO Confirm} message. This way the target BS is made aware of the arriving MSS from the network side.

In the meantime, the MSS may be re-associating with the target BS using PKMv2 and EAP as already explained in the start-up procedure (cf. Section 6.2.4.1). If possible, the fast re-authentication procedure of [154] can be applied. As in the start-up procedure, the nASN-GW establishes an R6 data path with the target BS before any messages to and from the MSS can be sent between the two entities.

The MSS now sends a \textit{DHCPINFORM} message to the nASN-GW. If it does not know whether or not the predictive mode of Active MPLS has successfully been triggered, it includes a HO object into the \textit{DHCPINFORM} message to initiate the reactive mode of Active MPLS. The nASN-GW replies with a \textit{DHCPACK} message. The reactive mode of Active MPLS is explained in detail in Chapter 5. The \textit{DHCPACK} message accomplishes the WiMAX handover procedure.

### 6.3 GPRS/UMTS

The Global System for Mobile communications (GSM) has been developed in the 1980s as a digital evolution of the analog cellular systems of the first generation. It operates on different frequency bands in different parts of the world. The most common frequency band is the 900 MHz band. Note that there are different bands for uplink and downlink traffic. In the 900-MHz band, the uplink band covers the 890-915 MHz band, while the downlink covers the 935-960 MHz band. In the meantime, these bands have been extended to 880-915 MHz for the uplink and to 925-960 MHz for the downlink. GSM systems are also referred to as systems of the 2nd Generation (2G).

Since GSM has originally been designed for circuit-switched voice, but not for packet-switched data, General Packet Radio Service (GPRS) [3] has been designed as an evolution of GSM. It makes use of the same frequency bands. GPRS systems are also referred to as 2.5G systems. A good overview of GPRS is given by [139, 27].

The Universal Mobile Telecommunications System (UMTS) standard has been finalized in 1999. It operates in the 1900-2200 MHz frequency band and allows for data rates of up to 384 kbit/s per user. With High Speed Uplink Packet Access (HSUPA) and High Speed Downlink Packet Access (HSDPA) much higher data rates can be achieved. In GPRS, the data rate depends on the coding scheme and the number of channels assigned to a user. The coding scheme with the smallest level of protection is CS-4 and offers a data rate of 21.4 kbit/s. Since at most eight channels may be assigned to a user, the maximum data rate amounts to 171.2 kbit/s. However, channels in use for GPRS cannot simultaneously be assigned to circuit-switched voice users so that generally only one channel is...
6.3. GPRS/UMTS

assigned to a user. Furthermore, the CS-2 coding scheme with 13.4 kbit/s is generally preferred so that the above-mentioned maximum data rate is hardly achieved in practice.

6.3.1 Architecture

A GSM network can be subdivided into an access network and a core network. The access network that is also referred to as Base Station Subsystem (BSS) consists of two different entities. These are the Base Transceiver Station (BTS) and the Base Station Controller (BSC). The core network consists of many different entities such as the Mobile services Switching Center (MSC), the Visitor Location Register (VLR), the Home Location Register (HLR) and the Authentication Center (AuC). If a GSM network is enhanced by GPRS, new entities are added to the core network. These are the Serving GPRS Support Node (SGSN) and the Gateway GPRS Support Node (GGSN). In a UMTS network, finally, the core network components are the same as in a GSM/GPRS network. In particular, a common core network is possible. The access network components, however, are replaced by new ones. The new components are the Node-B (being the equivalent of the BTS) and the Radio Network Controller (RNC) (being the equivalent of the BSC). The UMTS access network is also referred to as Radio Access Network (RAN). In general, a RAN contains many RNCs. An RNC together with the Node-Bs under its control is also referred to as Radio Network Subsystem (RNS).

Since the architectures of GPRS and UMTS are closely related to one another, Fig. 6.16 shows both architectures together in a simplified view. Both BTS and Node-B constitute the other end of the radio link with the Mobile Station (MS). While a BTS uses a combination of Frequency Division Duplex (FDD) and Time Division Duplex (TDD), a Node-B uses Wideband Code Division Multiple Access (W-CDMA) as radio transmission technology.

Several BTSs are controlled by a BSC, while several Node-Bs are controlled by a RNC. In general, the entities are organized in a strictly hierarchical way with two levels. However, other topologies are conceivable and remain for further study. Since BTSs and Node-Bs can be viewed as physical relays, a BSC and a RNC have the task of controlling the allocation and release of radio resources, the management of a handover between BTSs or Node-Bs under their control, etc.

The SGSNs are responsible for location management, paging and mobility management with respect to the packet-switched part of the core network. In general, each access network under the control of an SGSN constitutes a different Routing Area (RA) which is identified by a proper Routing Area Identifier (RAI) broadcast by the BTSs or Node-Bs belonging to that RA.

A GGSN constitutes the border of the core network to an external Packet Data Network (PDN) such as the Internet. A GGSN is not aware of the current
location of the MS. If the MS is registered at the GGSN, the GGSN is only aware of the SGSN currently in charge of the MS. It forwards any incoming IP packets destined to the MS to the corresponding SGSN.

6.3.2 Protocol Stacks

The protocol stacks of GPRS and UMTS differ sufficiently to be presented separately. The GPRS protocol stack for the control plane is depicted in Fig. 6.17. The GPRS Mobility Management (GMM) protocol has already been described in Section 4.4.6. The MS uses GMM to update its current location at the SGSN currently in charge of the MS. Session Management (SM) is used for establishing the Packet Data Protocol (PDP) context. SM is terminated by the current SGSN that actually establishes the PDP context at the corresponding GGSN via the GPRS Tunnelling Protocol (GTP) for the control plane. GMM and SM identify the MS by a Packet Temporary Mobile Subscriber Identity (P-TMSI) assigned by the SGSN on start-up.

The Logical Link Control (LLC) [7] protocol maintains a logical connection between the MS and the SGSN. The logical connection is provided by the concept of a Service Access Point Identifier (SAPI). Higher-layer protocols such as GMM/SM or others are identified by a particular SAPI. Hence, different higher-
layer protocols can be multiplexed over the same LLC connection. The MS is identified by a Temporary Logical Link Identifier (TLLI) which is derived from the P-TMSI so that the LLC layer at the SGSN can assign arriving LLC packets to the correct MS.

The Radio Link Control (RLC) protocol provides a reliable connection between MS and BSS. On the sender side, it segments LLC packets into numbered RLC data blocks. On the receiver side, RLC reassembles these data blocks. The MAC layer finally is in charge of cell selection, management of radio resources, etc.

On the user plane, the protocol stack is almost the same. The only differences are that instead of GMM/SM the SubNetwork-Dependent Convergence Protocol (SNDCP) [8] is used to multiplex IP packets issuing from different applications over the same LLC connection. The application is identified by a Network Service Access Point Identifier (NSAPI). From the BSS the packets are transported over the BSS GPRS Protocol and the Network Service layer to the SGSN where they are received by the SNDCP layer. The SGSN then tunnels the IP packets to the GGSN via the GTP protocol for the user plane.

The UMTS protocol stacks are shown in Fig. 6.19 and in Fig. 6.20. On the control plane, several changes apply with respect to GPRS. Apart from the different air interface, LLC has been replaced by the Radio Resource Control (RRC) protocol. An RRC connection is established between MS and RNC for any circuit-switched or packet-switched data that is to be transmitted between the two entities. The data transport between RNC and SGSN is done via the Radio Access Network Application Part (RANAP) and the Signalling Connection
Control Part (SCCP) protocols. RANAP manages e.g. the GTP connections that are used on the Iu interface between RNC and SGSN. In contrast to GPRS where integrity protection is not offered as security service and where encryption is applied in the LLC layer, both integrity protection and encryption are provided by the RLC or the MAC layer in function of the RLC mode [5].

On the user plane, the main differences are that GTP has been extended to the RNC and that SNDCP has been replaced by the Packet Data Convergence Protocol (PDCP). The main task of PDCP is to map higher-layer protocols such as IP, PPP, etc. onto the lower-layer RLC or MAC protocols.

### 6.3.3 Discovery

Each BTS or Node-B broadcasts general signalling information on a particular physical channel. On this broadcast channel, the information of several logical channels are transmitted. After switching to each possible carrier frequency of the broadcast channel, the MS synchronizes with the transmitting BTS or Node-B and then decodes the information broadcast on the broadcast channel. According to one or more selection criteria, the MS may then automatically choose the best BTS or Node-B to camp on. Optionally, user intervention may be required in the selection process.
6.3.4 Application to Active MPLS

The GPRS or UMTS start-up and handover procedures are complex procedures involving many protocols and network entities and specified in many different specifications. A complete description is thus beyond the scope of this document. Nevertheless, both procedures are roughly sketched in the following two subsections. Adaptations necessary through the application of Active MPLS are marked as bold in the respective figures. More information on the standard start-up procedure can be found in [115]. The following description of the start-up and handover procedures are based on GPRS. The procedures for UMTS are analogous.

6.3.4.1 Start-Up

Before a MS may transmit or receive any packet-switched data, it has to GPRS attach with the SGSN and establish a PDP context with the GGSN. Both procedures are depicted in Fig. 6.21.

The GPRS attach procedure is performed by GPRS Mobility Management (GMM) (cf. Section 4.4.6). The first message a MS sends is a GMM Attach Request message containing an identifier of the MS and the Routing Area Identifier (RAI) as identifier of the current routing area. If the MS has already been assigned a Packet Temporary Mobile Subscriber Identity (P-TMSI), it sends the...
CHAPTER 6. WIRELESS TECHNOLOGIES

P-TMSI, otherwise its International Mobile Subscriber Identity (IMSI) as MS identifier. In both cases, the SGSN is aware of the MS’s IMSI.

If the MS is not yet authenticated, it has to be authenticated. Authentication is done with the help of the AuC associated with the HLR. The protocol running between SGSN and HLR/AuC is the Mobile Application Part (MAP) protocol [2]. The SGSN sends a MAP Send Authentication Information message including the MS’s IMSI. Authentication is primarily based on a shared key \( K_i \) between the MS and the AuC. On receipt of the MAP Send Authentication Information message, the AuC chooses a random number \( RAND \) and computes an expected response \( XRES \) with the \( A_3 \) algorithm and a cipher key with the \( A_8 \) algorithm. Both algorithms are, in their current implementation, not publicly available.

\[
XRES = A_3(RAND, K_i) \quad (6.1)
\]

\[
K_c = A_8(RAND, K_i) \quad (6.2)
\]

The AuC returns the random number \( RAND \) and the derived expected response \( XRES \) and the cipher key \( K_c \) back to the SGSN. The SGSN relays the random number \( RAND \) to the MS and requests the MS to compute a signed response \( SRES \) and the cipher key \( K_c \) and to send the signed response \( SRES \) back to the SGSN. This is done via GMM Authentication and Ciphering messages. On receipt of the signed response \( SRES \), the SGSN compares \( SRES \) to the expected response \( XRES \) received from the AuC. If both values are equal, the MS is authenticated. Otherwise, it is not authenticated.

If authentication is successful, the current location of the MS can be updated in the HLR. The update is initiated by a MAP Update GPRS Location message containing the MS’s IMSI as identifier. In order to provide the SGSN with local subscriber information, the HLR sends a MAP Insert Subscriber Data message to the SGSN. After acknowledging this message, the HLR acknowledges the location update by sending a MAP Update GPRS Location Ack message to the SGSN.

After updating the MS location at the HLR, the attach procedure can be accomplished by sending a GMM Attach Accept message to the MS. In general, the SGSN includes a new P-TMSI and a new P-TMSI signature based on the derived cipher key \( K_c \) into this message. If a new P-TMSI is included, the MS has to acknowledge receipt of the new P-TMSI by sending a GMM Attach Complete message back to the SGSN.

PDP context establishment is done via the Session Management (SM) protocol [1]. The MS sends a SM Activate PDP Context Request message to the SGSN containing the Access Point Name (APN) at the GGSN and two identifiers as to how the application-specific traffic is to be mapped onto lower-layer connections. These are the NSAPI for mapping the application-specific traffic to the SNDCP layer and the LLC SAPI for mapping the SNDCP layer to a particular LLC connection. At the SGSN, the MS is identified by the TLLI contained in the LLC header.
The SGSN then sends a GPRS Tunnelling Protocol (GTP) \[4\] Create PDP Context Request message to the GGSN associated with the APN. On receipt of this message, the GGSN allocates an IP address according to the APN. It may choose a private IP address or use DHCP to acquire a public IP address from the operator network or use DHCP to acquire a private or public IP address from a corporate network.

When Active MPLS or Fast Active MPLS is used, the core network needs to be based on MPLS. An implementation of Active MPLS is required in the SGSNs and the GGSNs and optional in the other core network nodes. If it is only implemented in the SGSNs and the GGSNs, the resulting approach is Fast Active MPLS (cf. Chapter 5 and in particular Section 5.6.1). Note that all primary and handover LSPs necessary for the operation of Active MPLS or Fast Active MPLS have to be pre-established. In order to keep the necessary changes small, the GPRS/UMTS start-up procedure as well as the involved signalling protocols may still be used. On receipt of the GTP Create PDP Context Request message, the GGSN only has to update the classification data base for incoming IP packets such that IP packets destined to the IP address just allocated are placed on the pre-established LSP towards the SGSN it has received the GTP Create PDP Context Request message from.

Afterwards, the GGSN responds with a GTP Create PDP Context Response message containing the allocated IP address. The SGSN accomplishes the PDP context establishment procedure by sending a SM Activate PDP Context Accept message to the MS.

Finally, in order to transmit and receive data reliably, the MS and the SGSN have to establish an LLC connection. A description of the LLC connection establishment is, however, beyond the scope of this document.

**6.3.4.2 Handover**

When the MS camps on a cell, it receives from the BTS the list of neighbor cells. Each neighbor cell is represented by a BTS and identified by an identifier for the BTS and the frequency of the BTS’s broadcast channel. The MS continuously monitors the signal strength on these broadcast channels. The decision as to whether the current cell has to be changed, a process which is referred to as cell reselection, may be incumbent to the MS or to the network. More information on cell reselection can be found in \[149\].

If cell reselection is incumbent to the network, the MS continuously sends measurement reports to the network. A handover would be a network-initiated mobile-assisted one in this case. If cell reselection is incumbent to the MS, a handover would be mobile-initiated.

If cell reselection comes along with a change of the Routing Area (RA), higher-layer mobility management procedures are required. The protocol used is GMM (cf. Section 4.4.6). Since the MS sends the first message, the predictive mode is
executed when this message is sent before cell reselection, and the reactive mode is executed when this message is sent after cell reselection.

The message sent is a GMM RAU Request (cf. Fig. 6.22) containing the old P-TMSI, the old P-TMSI signature and the old RAI. If this message is sent to the new SGSN as is the case in the reactive mode, the new SGSN has to determine the old SGSN based on the information contained in the GMM RAU Request message. It then sends a GTP SGSN Context Request message to the old SGSN.

If the GMM RAU Request is sent to the old SGSN as is the case in the predictive mode, the GTP SGSN Context Request message is dispensable. In both modes, the procedure continues at the old SGSN sending a GTP SGSN Context Response message including the MS Mobility Management and all its PDP contexts to the new SGSN. The new SGSN acknowledges receipt of this message with a GTP SGSN Context Ack message.

The update of the PDP context at the GGSN can only be triggered by the old SGSN since the old SGSN has established a GTP tunnel with the GGSN. Therefore, the update of the PDP context can only be triggered as soon as the
old SGSN is aware of the handover. In GPRS as specified by the 3GPP, this is done after context transfer to the new SGSN. For optimization, the update of the PDP context, may, however, be brought forward to after receipt of the GTP SGSN Context Request message.

In any case, Active MPLS as described in Chapter 5 may be deployed to update the PDP context, thus replacing the corresponding GTP messages.

In the meantime, the new SGSN updates the location of the MS at the HLR by sending a MAP Update GPRS Location message. Since the HLR has stored a different SGSN in its database, it sends a MAP Cancel Location message to the old SGSN. As soon as this message is acknowledged, the HLR inserts the subscriber data into the new SGSN. After acknowledgement of the corresponding MAP Insert Subscriber Data message, the HLR acknowledges the location update by sending a MAP Update GPRS Location Ack message to the new SGSN.

On receipt of this message, the new SGSN accomplishes the RAU procedure by sending a GMM RAU Accept message to the MS. It may and in general does assign a new P-TMSI so that the MS is required to acknowledge receipt of the new P-TMSI with a GMM RAU Complete message.

Eventually, a previously existing LLC connection with the old SGSN has to be re-established with the new SGSN.
CHAPTER 6. WIRELESS TECHNOLOGIES
Chapter 7

Inter-technology mobility management

In the previous chapters, mobility management has been covered for intratechnology handovers. This chapter covers inter-technology handovers, i.e. handovers between Points of Attachment (PoAs) of different link-layer technologies.

Since inter-technology handovers occur between PoAs belonging to access networks of different technologies and since access networks of different technologies generally belong to different IP subnets, an inter-technology handover generally requires the mobile device to acquire a new IP address.

In principle, there are two basic requirements for the mobile device to be capable of inter-technology handovers, one hardware requirement and one software requirement. With respect to the hardware, the mobile device is required to contain at least two NICs of different technologies, e.g. a GPRS NIC and a WLAN NIC. With respect to the software, the mobile device is required to incorporate some Mobility Management Entity (MME) in charge of changing the NIC in case of an inter-technology handover. Since changing the NIC is not necessary in case of an intra-technology handover, a MME is not required for intra-technology handovers.

Although intra-technology handovers are the prevailing type of handover, there are also use cases for inter-technology handovers. However, note that an incentive is needed for either the user or the network operator to transfer the current connection via the current access network and technology to another connection via another access network and technology. On the user side, possible incentives may be that the connection via the other access network and technology is cheaper or offers better QoS. It may also be attractive for a user to attach with a low-range high-bandwidth technology while being stationary and to seamlessly change to a wide-range technology when starting to move. On the operator side, possible incentives may be that the current access network is congested or that network resources have become too limited or that it is cheaper to transfer the user data via another access network and technology.
Since the MME needs to decide which NIC to change to, it needs to be provided with relevant link-layer indications and information. Such link-layer indications and information may, of course, also be exploited for intra-technology handovers, in particular for handovers in WLAN environments where in general no such link-layer indications and information are available to the network-layer mobility management protocol such as Mobile IP or Active MPLS.

7.1 Mobility Support

Mobility support may be provided in terms of handover services. Handover services aim at improving the handover process by making it more seamless and smoother and by making it transparent to the user. Therefore, handover services need to report events and commands between the link layer and the network layer. These events and commands may be reported locally or remotely. Remote events and commands require a link-layer or network-layer transport mechanism. Handover services also need to exchange information between the mobile device and the network.

In 3GPP systems, such handover services are implemented by some 3GPP-specific signalling protocols that may not be used for intra-technology handovers of other link-layer technologies or for inter-technology handovers. In networks based on WiMAX, again only WiMAX-specific handover services exist. In networks based on WiFi, no handover services are available. With the IEEE 802.11r standard, the IEEE is currently developing an extension to support fast intra-IEEE 802.11 handovers. In order to support inter-technology handovers, the IEEE 802.21 standard (cf. Section 7.2) specifies the necessary handover services. Since the handover services affect both the link layer and the network layer, the IETF is dealing with handover services as well. The corresponding draft documents are [146, 9, 58].

So far, three handover services have been defined. These are the Event Service (cf. Section 7.1.1), the Command Service (cf. Section 7.1.2) and the Information Service (IS) (cf. Section 7.1.3).

7.1.1 Event Service

The Event Service [146] provides link indications from the link layer to a higher layer about changes in the link state. A link indication denotes any kind of information provided by the link layer to some layer above the link layer and regarding the state of the link. Examples are Link Down, Link Up etc. Architectural implications of link indications on the different layers in the protocol stack are given by [9].

It is important to clearly define when link indications are sent and how they may be interpreted. Link indications should not be sent due to transient changes
7.1. MOBILITY SUPPORT

Figure 7.1: Event Service

in link conditions, but rather due to a change in link state. The link state may change in response to a corresponding link-layer message exchange or as a result of permanent frame loss. A Link Up indication may e.g. be sent after completion of the IEEE 802.11 authentication and association, while a Link Down indication may be sent after transmission or receipt of an IEEE 802.11 Deauthentication frame.

While in wired environments, high frame loss can only be experienced after a Link Down indication, this assumption does not necessarily hold in wireless environments. So high frame loss cannot easily be used as Link Down indication in wireless environments and should be used with care.

Note that in general only the network-layer is a reasonable recipient of link indications since a link-layer Link Up indication does not necessarily mean that the network layer is configured and ready for use. However, there are scenarios in which the transport layer or the application layer may benefit from link indications as well.

Link indications are in general required by mobility management solutions offering a predictive mode such as FMIPv6 (cf. Section 4.5.3) or Active MPLS (cf. Chapter 5). Link indications are generally also required for inter-technology handovers such as for GPRS-UMTS handovers. Note, however, that for GPRS-UMTS handovers the link indications are specified in the corresponding 3GPP standards so that no link indications specified by the IETF may be used.

Link indications may be informative, in which case they are also referred to as hints, or mandatory, in which case they are also referred to as triggers. Link indications may be issued locally or sent as remote events (cf. Fig. 7.1) to a peer [146].

Remote events have to be transported to the entity for which they are destined. Link indications may be transported via the link layer, in which case the link-layer control or management plane may be exploited, or via the network layer, in which case the link layer data plane may be exploited.

Remote events are exchanged between the MME on one node and the MME on another node. A special transport mechanism is needed. If the two nodes are on the same link, the remote events may be transposed via the corresponding link-layer protocol that generally needs to be appropriately adapted. Otherwise, the remote events may be transported over IP. Note that the protocol stack (cf.
7.1.2 Command Service

The Command Service incorporates local commands and remote commands [146]. Local commands are sent within a node, whereas remote commands are sent between nodes. In both cases, the command sent instructs the receiving entity to execute a particular function. Such a function may be to disconnect a certain link layer connection, to establish a new link layer connection, etc. Fig. 7.2 shows the different types of commands in the context of IEEE 802.21.

Remote commands are exchanged between the MME on one node and the MME on another node. A special transport mechanism is needed. If the two nodes are on the same link, the remote events may be transported via the corresponding link-layer protocol that generally needs to be appropriately adapted. Otherwise, the remote events may be transported over IP. Note that the protocol stack (cf. Section 7.2 and Fig. 7.6) is the same for both transport options.

Local commands are in general issued from a higher layer to a lower layer, i.e., from the MME to the corresponding link layer. Note that a remote command may trigger a subsequent local command or another remote command that may itself again trigger another local or remote command, etc.

7.1.3 Information Service

The Information Service (IS) [58] provides a network or a mobile device with information that may assist the network or the mobile device during a handover. Such information may be information about neighboring Points of Attachment (PoAs) or about neighboring networks. Information about neighboring PoAs may contain identifiers, names, physical parameters, resource utilization, etc. Information about neighboring networks may contain identifiers, names, the subnet mask, etc.
7.1. MOBILITY SUPPORT

In the network, the IS is generally provided by one or more IS servers. An IS server may be located in the PoA, in the access network or in the core network (cf. Fig. 7.3).

In Fig. 7.3 (a), the IS is deployed in the PoA. Since a PoA may be aware of information concerning the access network it is located in, this is a likely scenario. If the link-layer protocol may be adapted, it is possible to incorporate the IS data into the link-layer protocol messages. Otherwise, the IS can simply be transported over IP. In this case, IS server discovery is needed.

In Fig. 7.3 (b), the IS is deployed in the access network, e.g. in the first-hop IP router. By using a centralized server in the access network, the administrative overhead of configuring each PoA is mitigated. In this scenario, the IS may be transported over the corresponding link-layer protocol or over IP. However, the transport over the corresponding link-layer protocol is only possible if the IS is deployed in the first-hop IP router. Otherwise, the IS has to be transported over IP. IS server discovery is needed. Deploying the IS in the access network is already addressed by the Candidate Access Router Discovery (CARD) [95] (cf. Section 7.1.4) and by FMIPv6 (cf. Section 4.5.3).

In Fig. 7.3 (c), the IS is deployed in the core network. This scenario proves most appropriate in environments where the access networks can hardly be adapted or not be adapted at all. Furthermore, the administrative configuration overhead is further reduced compared to scenarios (a) and (b). The IS needs to be transported over IP. IS server discovery is needed.

Note that it is conceivable to deploy a chain of IS servers. When a mobile device discovers and contacts a certain IS server for a particular piece of information and if that IS server is not aware of the requested piece of information, it may contact another IS server and request itself the desired piece of information.

The process of obtaining information from an IS server is shown in Fig. 7.4.
Initially, the IS server needs to be discovered. There are basically three possibilities for IS server discovery. One possibility is that the IS server itself participates in a corresponding discovery protocol. Another possibility is that another entity or set of entities within the access or core network participates in a corresponding discovery protocol. Finally, a directory service may be used. In each case, the mobile device is provided with an address of the IS server. In general, this address is an IP address. However, other types of addresses may be used as well.

Once provided with an address of the IS server, the mobile device contacts the IS server in order to check the IS server’s reachability. The contact message may also be the first message of the subsequent security association message exchange.

In order to ensure the validity of any data sent by the IS server, IS server authentication is required. In some environments, mobile device authentication is required as well in order to prevent resource exhaustion attacks. While message authentication is mandatory, message encryption is optional. The establishment of security associations and the different security services are covered in Chapter 3.

Only when an appropriate security association is established between the mobile device and the IS server, Information Request and Information Response messages may be exchanged.

### 7.1.4 Candidate Access Router Discovery

The Candidate Access Router Discovery (CARD) [95] is an IS protocol aiming at providing the mobile device with information about the networks of neighboring PoAs. When a handover becomes imminent, a mobile device presumably is aware of one or more PoAs it may handover to. In order to decide which of these PoAs to handover to, the mobile device might be in need of more information about the networks the corresponding PoAs belong to. CARD has been designed for this purpose.
When a mobile device knows the link-layer identifier of neighboring PoAs, it may exchange CARD messages with its current Access Router (AR) in order to retrieve more information about the networks these PoAs belong to. CARD provides the mobile device with the IP address of the Candidate Access Router (CAR) a PoA is connected to as well as with other information such as the CAR’s capabilities.

The mobile device may send a MN-AR CARD Request message to its current AR at any time (cf. Fig. 7.5). The mobile device may include zero, one or more link-layer identifiers (L2 IDs) into the MN-AR CARD Request message. By default, the current AR performs reverse address translation and resolves each L2 ID into the IP address of the associated CAR. If the mobile device desires to be provided with the CAR’s capabilities as well, it needs to set a special flag in the MN-AR CARD Request message.

On receipt of the MN-AR CARD Request message, the current AR performs reverse address translation for the L2 IDs contained in the MN-AR CARD Request message. If the message contains no L2 ID, the current AR simply retrieves all pairs of L2 ID and IP address from its local CARD table. If the capabilities flag is set, the current AR also retrieves the corresponding capabilities. Finally, it sends a MN-AR CARD Reply message with the requested information back to the mobile device.

As just mentioned, an AR maintains a local CARD table (cf. Fig. 7.5). Each entry in the local CARD table contains the L2 ID of a PoA, the IP address of the associated CAR and the capabilities of that CAR. Each entry is moreover
associated with a timer. It is the AR’s responsibility to refresh the entry before
the timer expires. The protocol messages used are the AR-AR CARD Request
and the AR-AR CARD Reply messages. Note that the local CARD table contains
more information than the conventional ARP table. Furthermore, the CARD
table may contain information about CARs not part of the current subnet.

In order to keep the number of messages low, the current AR may place its
own IP address and capabilities into the AR-AR CARD Request message. So
the receiving CAR may update its own entry for the current AR.

ARs also need to share an IPsec Security Association offering mutual au-
thentication in order to ensure that an AR does not accept information from an
unauthenticated source.

7.2 IEEE 802.21

The purpose of the IEEE 802.21 draft standard [79] is to support inter-technology
handovers. In particular, IEEE 802.21 aims at providing transparent service con-
tinuity during inter-technology handovers, i.e. the handover shall not be percep-
tible by higher layers within the protocol stack or the user himself. Note that
the ultimate decision as to whether or not a handover is to be executed or which
PoA to handover to is incumbent to a higher-layer mobility management protocol
such as Mobile IP.

IEEE 802.21 introduces a new function to the protocol stack of the rele-
vant network entities. The function is denoted as Media-Independent Handover
Function (MIHF). The MIHF interacts with media-specific link-layers via media-
specific Service Access Points (SAPs) and with media-independent higher layers
via a media-independent SAP (MIH SAP). The MIHF provides an Event Service
(cf. also Section 7.1.1), a Command Service (cf. also Section 7.1.2) and an Infor-
mation Service (IS) (cf. also Section 7.1.3). Note that the same terminology is
used within IEEE 802.21 and within the IETF drafts referred to in Section 7.1.
The reason is that during the specification of IEEE 802.21, the corresponding
drafts have been submitted to the IETF in order to disseminate the idea of
IEEE 802.21. The location of the MIHF within the mobility management pro-
tocol stack is shown in Fig. 7.6. Note that the MIHF is not located on the user
plane but only on the control plane.

In order to interact with the MIHF through Event Service, Command Service
and IS, the media-specific link-layers have to be amended by new SAPs. These
new SAPs may be specified as future amendments of the respective link-layer
standard specifications.

The MIHF is located in the mobile device and those network elements pro-
viding handover support. Each of the three services Event Service, Command
Service and IS may be provided and used locally, i.e. within an entity, or re-
motely, i.e. between two entities. The reference points for the communication
7.2. IEEE 802.21

Figure 7.6: Location of the IEEE 802.21 MIHF within the mobility management protocol stack.

Figure 7.7: Reference points of the IEEE 802.21 MIHF communication architecture between two entities are depicted in Fig. 7.7.

On the network side, the MIHF may be located in a PoA or in a non-PoA. Note that a PoA resides on the same link as the MN so that a non-PoA may be a PoS, a gateway, etc. The PoA may be the serving PoA or a candidate PoA. When the MIHF of a network element communicates directly with the MIHF of the mobile device, it is denoted as MIH PoS.

The MIHF of the mobile device may communicate with the MIHF of the serving PoA (reference point R1), the MIHF of a candidate PoA (reference point R2) or a non-PoA (reference point R3). The reference point between two entities acting as MIH PoS is referred to as R5. The reference point between a MIH PoS and a MIH non-PoS is denoted as R4.

In order to exchange MIH information between two MIH entities, L2 or L3
CHAPTER 7. INTER-TECHNOLOGY MOBILITY MANAGEMENT

<table>
<thead>
<tr>
<th>Media type</th>
<th>L2 transport</th>
<th>L3 transport</th>
<th>Preference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethernet</td>
<td>Data frames</td>
<td>IP-based</td>
<td>L2</td>
</tr>
<tr>
<td>IEEE 802.11</td>
<td>Data and management frames</td>
<td>IP-based</td>
<td>L2</td>
</tr>
<tr>
<td>IEEE 802.16</td>
<td>Data and management frames</td>
<td>IP-based</td>
<td>L2</td>
</tr>
<tr>
<td>3GPP/3GPP2</td>
<td>Requires protocol stack changes</td>
<td>IP-based</td>
<td>L3</td>
</tr>
</tbody>
</table>

Table 7.1: Transport options for MIH payload

transport may be used. If L2 transport is used, MIH payload is encapsulated into L2 frames. IEEE 802.3 allows MIH payload to be placed into data frames, whereas IEEE 802.11 allows MIH payload to be placed both into data and management frames. However, in order to place MIH payload into data frames, the mobile device has to be associated with the AP. Therefore, it is recommended to encapsulate MIH payload into IEEE 802.11 management frames.

If L3 transport is used, the MIH payload may be encapsulated into network-layer datagrams. A new L3 protocol type has to be defined in this case. The MIH may also be encapsulated into UDP or TCP segments. In that case, a new MIH port has to be defined. Table 7.1 summarizes the different possibilities for transport of MIH payload.

The MIH protocol header is shown in Fig. 7.8. The Protocol Version field is specified as 1 in the current IEEE 802.21 standards draft. The MIH Service ID specifies the service (Event Service, Command Service or IS) this protocol message refers to. The MIH Opcode field specifies the operation to be performed. Three values are possible. These are Request, Response and Indication. The Transaction ID field is used to match requests and responses. The Message Length denotes the total message length in Byte.

The Fr flag indicates whether or not the current message is fragmented. Fragment No contains the fragment number of the fragment within the message. Since a MIH message is transmitted via L2 or via L3 and between entities of varying link-layer technologies, the MIHF identifiers vary, too. The MIHF Identifier Length indicates the length of the MIHF identifiers while the Source MIHF Identifier and the Destination MIHF Identifier contain the respective MIHF identifiers themselves. The identifier may be a L2 hardware address, an IP address or some identifier generated during MIHF discovery. The MIH Message Identifier designates the message itself. Various messages such as MIH Capability Discover, MIH Event Register, MIH Link Up, MIH Link Down, MIH Link Going Down, MIH Handover Initiate, etc. have already been defined. Finally, MIH Message Data contains the variable-length MIH payload.
7.3. INTER-WORKING

Since inter-technology mobility management deals with handovers between Points of Attachment (PoAs) of different technologies, the wireless systems these PoAs belong to have to be inter-connected. The inter-connection is also referred to as inter-working. Inter-technology mobility management and its benefits both for the user and for the network operator (cf. the introductory part of Chapter 7) are thus one incentive for network convergence and inter-working.

Another incentive for inter-working is the following. Currently, all packet-switched networks are operated separately by either the same or different operators. The infrastructure of each network is kept separate from those of the other networks, i.e. similar or equal components occur in each network. All packet-switched services are only made available to the network they are offered by and not to other networks. Furthermore, customer data bases are kept separate. Even authentication and billing are done separately for each network. The common objective may thus be to eventually converge to one common network based on IP. While WLANs and WiMAX networks as well as the packet-switched core network part of cellular systems are each based on IP, it will be a large step to converge these networks to one with a common IP-based core network where packet-switched services may be exchanged seamlessly across the different networks. Such a network is sometimes referred to as 4G network.

In general, there are different inter-working scenarios. Since WLAN-GPRS/UMTS inter-working is the type of inter-working that has been analyzed the most, it is presented in Section 7.3.1. Other types of inter-working are only briefly dealt with in Section 7.3.2.
7.3.1 WLAN-GPRS/UMTS Inter-Working

Many research papers focus on the inter-working of GPRS and WLAN. In [114], basically four inter-working scenarios (cf. Fig. 7.9) are presented. These four inter-working scenarios, the used mobility solutions as well as the corresponding architectures and protocol stacks are briefly described in the following four sub-sections. Note that the protocol stacks on the user plane are the same across all four inter-working scenarios (cf. Fig. 7.10 (c)). In downlink direction, the IP software figures out with a routing table look-up which interface and thus which access technology to use in order to send out upper layer data packets.

7.3.1.1 Inter-Working Via BSS Emulator

In the first inter-working scenario, the WLAN is connected to the GPRS network through an entity emulating a BSS. The entity is called BSS emulator and behaves towards the SGSN as if it was a BSC and towards the WLAN as if it was a conventional IP router. This inter-working scenario has attracted strong attention under the name Unlicensed Mobile Access (UMA) and has meanwhile been handed to the 3GPP where it is being specified in [6]. Note that mobile devices supporting UMA are already commercially available.
7.3. INTER-WORKING

199

**Figure 7.10: Protocol stacks of the MN for both user and control plane**

User IP packets arriving at the GGSN are tunnelled to the SGSN the MN is currently attached with. If the MN is associated with a GPRS BS, the SGSN delivers the user IP packets to the corresponding BSC from where they are sent via the BS to the MN. If the MN is associated with a WLAN AP, the SGSN delivers the user IP packets to the corresponding BSS emulator. Through the BSS emulator, the implementation of the SGSN does not have to be adapted. The BSS emulator performs the necessary conversion and delivers the packet via the WLAN to the MN.

Mobility management is done via GMM and GTP, i.e. the MN updates its current location via a corresponding GMM RAU message. Therefore, the GMM layer in the MN needs to know which access technology the MN is currently using. If the MN is currently associated with a GPRS BS, GMM needs to pass any GMM message to the LLC layer (cf. Fig. 7.10 (a)). If the MN is currently associated with a WLAN AP, GMM needs to pass any GMM message to UDP. Changes are thus required in the GMM implementation of the MN.

### 7.3.1.2 Inter-Working Via SGSN Emulator

In the second inter-working scenario, the WLAN is connected to the GPRS network through an entity emulating an SGSN. The entity is called SGSN emulator and behaves towards the GGSN as if it was an SGSN and towards the WLAN as if it was a conventional IP router.

User IP packets arriving at the GGSN are tunnelled to the SGSN the MN is currently attached with. If the MN is associated with a GPRS BS, the SGSN delivers the user IP packets to the corresponding BSC from where they are sent via the BS to the MN. If the MN is associated with a WLAN AP, the SGSN emulator performs the necessary conversion and delivers the packet via the WLAN to the MN.

As in the first inter-working scenario, mobility management is done via GMM.
and GTP, i.e. the MN updates its current location via a corresponding GMM RAU message. Therefore, the GMM layer in the MN needs to know which access technology the MN is currently using. If the MN is currently associated with a GPRS BS, GMM needs to pass any GMM message to the LLC layer (cf. Fig. 7.10 (a)). If the MN is currently associated with a WLAN AP, GMM needs to pass any GMM message to UDP. Changes are thus required in the GMM implementation of the MN.

The second inter-working scenario has been studied in more detail by [135] where it is referred to as tight coupling architecture. The WLAN is considered as distinct Routing Area. The SGSN emulator is denoted as GPRS Interworking Function (GIF). Note that tight coupling generally denotes the fact that one network is dominant over another in an inter-working scenario. This entails that packets destined to the mobile device always arrive at the dominant network from where they are further delivered to the mobile device whether it is currently located in the dominant or the other network. Note that both the first and the second inter-working scenario are tight coupling scenarios. Note also that Active MPLS and Fast Active MPLS may be used for mobility management in a tight coupling scenario.

The protocol stack for the control plane is depicted in Fig. 7.11. The main component is the WLAN Adaptation Function (WAF) that is deployed both in the MN and in the GIF. When a MN discovers an AP with the configured SSID, it associates with the AP. The IEEE 802.11 MAC layer signals activation of the WLAN NIC to the WAF. The WAF then initiates a RAI/GIF discovery procedure (cf. Fig. 7.12) by broadcasting a RAI/GIF Discovery Request message including the IMSI. On receipt of the message, the GIF stores a binding between IMSI and MAC address and responds with a RAI/GIF Discovery Response message containing the RAI associated with the WLAN. On receipt of the response, the WAF stores the RAI and the MAC address of the GIF. The WAF also informs LLC of the WLAN activation and GMM of the RAI change. GMM finally initiates the RAU procedure by sending a RAU Request message to the SGSN.

User packets originated at the MN are passed to SNDCP and from there to LLC (cf. Fig. 7.13). The LLC protocol is aware of whether or not WLAN is currently in use and passes upstream user packets to the corresponding link-
7.3. INTER-WORKING

Figure 7.12: WLAN-GPRS integration: Discovery procedure

Figure 7.13: WLAN-GPRS integration: Protocol stack for the user plane

layer. At the GIF, the user packets are received by the IEEE 802.3 MAC layer and forwarded to the connected SGSN via the BSS GPRS Protocol and Network Service layers. Downstream user packets arriving at the GIF are received by the BSS GPRS Protocol and Network Service layers and passed to the IEEE 802.3 MAC layer in direction of the WLAN AP and the MN. At the MN, they are passed to LLC.

The second inter-working scenario has also been proposed by [107]. The focus is, however, more on the GPRS core network for which the LEMA mobility solution (cf. Section 4.6.2) is proposed. Mobility management is described for both WLAN to UMTS handover and for UMTS to WLAN handover. Unfortunately, the description of both handover procedures is not conform to the 3GPP specifications, which is not pointed out.

7.3.1.3 Inter-Working Via Mobility Gateway

In the third inter-working scenario, a mobility gateway is deployed in the operator backbone network. The mobility gateway is connected to both the WLAN and the GGSN of the GPRS network. The protocol stacks both on the user plane and on the control plane are not specified. The idea of this inter-working scenario is to present an alternative to Mobile IPv4. Any user IP packets are sent to a fixed home address of the MN where they are intercepted by the mobility gateway that is aware of the current location of the MN. Note that the mobility gateway is similar to the HA of Mobile IPv4. On receipt of an IP packet for a MN, it looks up the current location of the MN and finally delivers the IP packet.
CHAPTER 7. INTER-TECHNOLOGY MOBILITY MANAGEMENT

7.3.1.4 Inter-Working Via Mobile IPv4 Home Agent

In the fourth inter-working scenario, Mobile IPv4 is used. The HA may be located in any network. If it is not located in the WLAN, a FA needs to be located there. As described in Section 4.4.1, user IP packets are routed to the HA. The HA intercepts these user IP packets and tunnels them to the FA CoA of the MN, i.e. to the FA. The FA finally decapsulates the user IP packets and delivers them to the MN. Fig. 7.10 (b) shows the protocol stack at the MN on the control plane. MIPv4 signalling messages are sent over UDP and IP so that the decision as to which access technology to use lies at the IP layer.

The fourth inter-working scenario is also referred to as loose coupling by [135]. Note that loose coupling generally denotes two networks having a common operator backbone but that are otherwise kept separate from one another. This entails that packets destined to the mobile device always arrive at some point outside both networks (the mobility gateway in the third or the HA in the last inter-working scenario) from where they are further delivered to the mobile device at its current location. Note that both the third and the last inter-working scenario are loose coupling scenarios. Note also that the necessity for having a point outside both networks prevents the use of an intra-domain mobility solution. Note finally that in particular the last inter-working scenario has been analyzed by numerous research papers.

In [161], mean throughput and handover delay are compared between vertical and horizontal handovers. The comparison is based on a simulation with the Matlab simulation framework. In [62], a real-life experiment with Linux hosts, a Dynamics Mobile IPv4 implementation and a public GPRS access network has been performed. The Dynamics Mobile IPv4 implementation has been extended for vertical handover support. Measurements have been performed for both a GPRS to WLAN handover and for a WLAN to GPRS handover. The only evaluated parameters are, however, sequence numbers and packet loss.

7.3.1.5 Discussion

With a technology such as GPRS that is ubiquitously available, the decision as to when to perform a handover from GPRS to WLAN is not time-critical since the GPRS signal is continuously available. A GPRS-to-WLAN handover is thus in general a policy-based handover. Note that the outage period caused by the handover still has an impact on the active data sessions and should be kept as low as possible. The decision as to when to perform a handover from WLAN to GPRS, on the other hand, may be time-critical, in particular when the WLAN signal may suddenly be lost. For this type of handover, an advanced handover decision algorithm is needed in the MN.

Both loose and tight coupling architectures have their advantages and disadvantages. Loose coupling architectures generally require no changes at the
involved networks since data packets are transported between the inter-working entity located outside the involved networks and the mobile device transparently over the involved networks. On the other hand, a loosely coupled architecture does not provide efficient handover support. Since the mobile device moves between two separately administered networks, re-registration and re-authentication are kept separate and can in general not be facilitated or accelerated.

Tight coupling architectures may exploit the facilities provided by the dominant network in order to facilitate and accelerate handovers between the networks. Since such handovers remain local to the dominant network, handover performance can be improved. However, tight coupling requires changes to the dominant network in order to properly integrate the different networks.

For the end user, tight coupling is more beneficial than loose coupling because handover performance is better and because the end user does not need to be registered with more than one network operator. Furthermore, extending a network with wide-area coverage such as a cellular system with several or even many other networks is more straightforward than establishing more than one network with wide-area coverage and operating them in parallel. Since UMA also shows that there is a certain tendency to tight coupling, tight coupling architectures will probably be the prevailing type of future network architectures.

### 7.3.2 Other Types of Inter-Working

Other types of inter-working are WiMAX-WLAN and WiMAX-GPRS/UMTS inter-working that are being specified in the current WiMAX draft. The mobility solution proposed for vertical handovers is Client Mobile IP (CMIP). Proxy MIP (PMIP) (cf. Section 6.2.2) cannot be used because there is no mobility client in the MN that could deal with updating routing table entries, etc. For vertical handovers, some entity is needed in the MN. CMIP provides such an entity and is therefore chosen as mobility solution for inter-technology mobility management within the WiMAX draft.

In the case of WiMAX-GPRS/UMTS inter-working, there are basically three different scenarios. If the CMIP HA is deployed in the WiMAX home network, the inter-working architecture can be viewed as tightly coupled with the WiMAX network as the dominant network. If the CMIP HA is deployed in the core network of the cellular system, the inter-working architecture can be viewed as tightly coupled with the cellular system as the dominant network. If, finally, the CMIP HA is neither deployed in the WiMAX home network nor in the core network of the cellular system, the inter-working architecture can be viewed as loosely coupled.

GPRS-UMTS inter-system handovers, eventually, are specified in [3]. An additional overview is provided by [96]. GPRS-UMTS inter-working architectures are generally tightly coupled architectures since the core network may be used as
common core network for the two different RANs. Since UMTS networks are an evolution of GSM/GPRS networks, the UMTS network can be considered as the dominant one.
Chapter 8

Conclusions and Outlook

This dissertation has introduced a novel mechanism based on Multi-Protocol Label Switching (MPLS) for mobility management in core networks. The mechanism has been named Active MPLS. A special case of Active MPLS has been named Fast Active MPLS. Depending on the particular core network characteristics, it may be more beneficial to deploy Active MPLS or more beneficial to deploy Fast Active MPLS. Active MPLS and Fast Active MPLS have been practically evaluated and compared to other approaches through testbed implementations and simulations.

The wireless packet-switched networks existing today (i.e. WLANs, WiMAX environments, 2.5G and 3G systems) generally offer a number of different packet-based services. For the data transfer required by these packet-based services, there is a strong tendency away from connection- and cell-oriented data transfer to packet-oriented data transfer. The main reasons are lower cost and lower complexity. Therefore, the focus of this dissertation has been on IP and MPLS.

If user mobility is supported, a mechanism needs to be in place solving the issue of redirecting packets to the new location of the mobile device. The necessary signalling operations for location update and packet redirection are termed as mobility management and have been presented in detail in Chapter 4.

Due to the architecture that wireless packet-switched networks generally exhibit nowadays and that generally encompasses one or more access networks and a core network, mobility management mechanisms are needed for both the access networks and the core network. Since the mobility management mechanisms for the access networks are generally incorporated into the corresponding access network technologies, this dissertation has focussed on core network mobility.

Packet redirection can be achieved via different mechanisms. One mechanism is to establish host-specific paths that are updated during the handover. The main disadvantage of such a mechanism is that it does not scale well in environments with many users. Another mechanism is to make use of tunnelling and to update the location at the tunnel source point during the handover. The main disadvantages of such a mechanism are the need for a new IP address and Duplic-
cate Address Detection, the overhead imposed through tunnelling and the packet loss occurring at the previous tunnel end point as long as the location is not yet updated at the tunnel source point. Finally, there are mechanisms that make use of tunnel sequences and that only update the location at those tunnel source points that really need to be updated during the handover. The main disadvantage of such a mechanism is that it does not scale well in environments with many users and that it has a negative impact on processing delay since packets have to be detunnelled and re-tunnelled at every node residing at the intersection of two subsequent tunnels within the tunnel sequence. Chapter 4 has presented a multitude of different IP- and MPLS-based mobility management approaches and pointed out their disadvantages. These disadvantages have been the incentive for the development of Active MPLS as a novel mobility mechanism.

Active MPLS that is presented in detail in Chapter 5 avoids the above-mentioned disadvantages and additionally improves the packet transmission of data packets during handovers through a well-adapted tunnel concept. The tunnels that are made use of are conventional LSPs. The description of Active MPLS covers both the processing of data packets on the user plane as well as of all necessary signalling messages on the control plane. Data packets are equipped with an additional user-specific label enabling an Active MPLS node to apply a user-specific treatment. Since a user-specific treatment is only needed during a handover, it is triggered through a handover message sent on the reverse path of the data packets. Each Active MPLS node receiving the handover message may be configured to act on it or not. In particular, in Active MPLS, every node along the reverse path of the data packets is configured to act on an arriving handover message, whereas in Fast Active MPLS, only the previous tunnel end point and the tunnel source point are configured to act on it. When the user-specific treatment is triggered at a particular Active MPLS node, the data packets are switched onto a pre-established tunnel (i.e. LSP) towards the new point of attachment. This makes packet redirection fast and efficient.

In order to assess Active MPLS with respect to other MPLS-based approaches, an implementation on Intel IXP1200 network processors has been performed. Active MPLS has been compared with Fast Active MPLS and Ingress-Update, an MPLS-based solution that only updates the location at the tunnel source point during a handover. It has been shown that the packet loss occurring with Ingress-Update can be reduced by roughly 60 % with Active MPLS and Fast Active MPLS although the outage periods are basically equal.

Since Active MPLS and Fast Active MPLS only differ in which nodes act on the handover message sent along the reverse path of the data packets, these two approaches have been compared analytically. The main achievement of this comparison is a formula which can be applied to any type of network and via which it can be assessed whether it is more beneficial to deploy Active MPLS or Fast Active MPLS. In core networks where the propagation delay between two nodes is rather low, it has been found out that it is more beneficial to deploy Fast
Active MPLS. In potential future satellite-based networks, where the propagation delay between two nodes is rather high, it is, however, more beneficial to deploy Active MPLS.

Since it would have been difficult to implement an IP-based approach on the network processors as well, an ns-2 simulation has been performed comparing Active MPLS with Fast Active MPLS and one of the most promising IP-based approaches, i.e. with Low Latency Handoffs in Mobile IPv4. The simulation results show that the outage period occurring with Low Latency Handoffs in Mobile IPv4 can be reduced by about 40% during inter-subnet handovers when using Active MPLS or Fast Active MPLS. In the same way, the absolute packet loss occurring during inter-subnet handovers can be reduced by about 60%. The mean end-to-end delay of the data packets affected by an inter-subnet handover is also lower in Active MPLS and Fast Active MPLS. The reason is that the LSPs used in Active MPLS and Fast Active MPLS may all be pre-established, whereas the IP tunnels in Low Latency Handoffs in Mobile IPv4 have to be set up during the handover in order to avoid scalability issues.

In Chapter 6, Active MPLS has been applied to networks based on IEEE 802.11, i.e. to WLANs, to networks based on IEEE 802.16e, i.e. to WiMAX environments, and to 2.5G/3G cellular systems. In particular, it has been shown how a user equipment attaches to the system and how the system performs a handover. Security and addressing issues are covered as well. This chapter proves that Active MPLS may indeed be applied to different wireless technologies. The required extensions to the user equipment and the system can generally be kept very low so that Active MPLS and Fast Active MPLS, respectively, can strongly be recommended as mobility management approach for current and future core networks.

Chapter 7 finally deals with the issue of inter-technology mobility management. Although intra-technology handovers are the prevailing type of handover nowadays, there are also use cases for inter-technology handovers. Inter-technology handovers may be beneficial when the new technology offers cheaper network access or network access with higher data rates or better Quality of Service. Moving out of the range of a short-range high data-rate technology such as IEEE 802.11 and performing a handover to a wide-range lower data-rate technology such as GPRS/UMTS while maintaining an ongoing data session may also be an incentive. Finally, congestion of the previous network might be a reason for the network operator to transfer an ongoing packet-based data session to a network of a different technology. Whether or not Active MPLS and Fast Active MPLS may be deployed for inter-technology mobility management depends on the type of inter-working used for the two network systems. If they have a common core network, which is also referred to as tight coupling, Active MPLS or Fast Active MPLS may be deployed within that core network. In this case, the same advantages over other MPLS-based or IP-based mechanisms (cf. Chapter 4) apply. In a loosely coupled inter-working scenario, where both network
systems have their own core networks, Active MPLS and Fast Active MPLS cannot be deployed. Note that a mechanism in the mobile device needs to support inter-technology handovers. Currently, the IEEE is working on the IEEE 802.21 standard dealing with this issue. Nevertheless, the network-layer mobility management approach would have to closely work together with IEEE 802.21 in order to make inter-technology handovers happen. This could be an interesting field for further study.

In future wireless packet-switched networks, the access networks may also be based on packet-based data transfer such as IP. This may already be the case for WiMAX networks as well as for the next-generation cellular systems which are often referred to as 3.9G. In order to deploy Active MPLS and Fast Active MPLS in these networks, further study is needed as to the feasibility and the inter-operation between Active MPLS and the protocols used in the respective access networks. Note that Active MPLS and Fast Active MPLS may in principle be deployed in any IP- or MPLS-based network.

Finally, an interesting field of further research is the field of link and router failures and all mechanisms resolving such link and router failures. In MPLS-based networks, MPLS-based backup mechanisms are already available. It would be interesting to combine such backup methods with Active MPLS or Fast Active MPLS or to conceive a completely new backup method that properly integrates with Active MPLS and Fast Active MPLS. However, these questions are complex enough for another dissertation.
## Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>3GPP</td>
<td>3rd Generation Partnership Project</td>
</tr>
<tr>
<td>AAA</td>
<td>Authentication, Authorization and Accounting</td>
</tr>
<tr>
<td>AAR</td>
<td>Advanced Access Router</td>
</tr>
<tr>
<td>AC</td>
<td>Access Concentrator</td>
</tr>
<tr>
<td>AF</td>
<td>Assured Forwarding</td>
</tr>
<tr>
<td>AH</td>
<td>Authentication Header</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>APN</td>
<td>Access Point Name</td>
</tr>
<tr>
<td>ARP</td>
<td>Address Resolution Protocol</td>
</tr>
<tr>
<td>AS</td>
<td>Authentication Server</td>
</tr>
<tr>
<td>ASN</td>
<td>Access Services Network</td>
</tr>
<tr>
<td>ASN-GW</td>
<td>ASN Gateway</td>
</tr>
<tr>
<td>ATM</td>
<td>Asynchronous Transfer Mode</td>
</tr>
<tr>
<td>AuC</td>
<td>Authentication Center</td>
</tr>
<tr>
<td>BET</td>
<td>Bidirectional Edge Tunnel</td>
</tr>
<tr>
<td>BSC</td>
<td>Base Station Controller</td>
</tr>
<tr>
<td>BSS</td>
<td>Base Station Subsystem</td>
</tr>
<tr>
<td>BTS</td>
<td>Base Transceiver Station</td>
</tr>
<tr>
<td>BU</td>
<td>Binding Update</td>
</tr>
<tr>
<td>BWA</td>
<td>Broadband Wireless Access</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>CAR</td>
<td>Candidate Access Router</td>
</tr>
<tr>
<td>CARD</td>
<td>Candidate Access Router Discovery</td>
</tr>
<tr>
<td>CBR</td>
<td>Constant Bit Rate</td>
</tr>
<tr>
<td>CCoA</td>
<td>Collocated CoA</td>
</tr>
<tr>
<td>CDMA</td>
<td>Code Division Multiple Access</td>
</tr>
<tr>
<td>CFA</td>
<td>Controlling Foreign Agent</td>
</tr>
<tr>
<td>CHAP</td>
<td>Challenge-Handshake Protocol</td>
</tr>
<tr>
<td>CID</td>
<td>Connection ID</td>
</tr>
<tr>
<td>CIDR</td>
<td>Classless Inter-Domain Routing</td>
</tr>
<tr>
<td>CMIP</td>
<td>Client MIP</td>
</tr>
<tr>
<td>CN</td>
<td>Correspondent Node</td>
</tr>
<tr>
<td>CoA</td>
<td>Care-of Address</td>
</tr>
<tr>
<td>COR</td>
<td>Cross-Over Router</td>
</tr>
<tr>
<td>CR-LDP</td>
<td>Constraint-based Routing LDP</td>
</tr>
<tr>
<td>CRC</td>
<td>Cyclic Redundancy Check</td>
</tr>
<tr>
<td>CSN</td>
<td>Core Services Network</td>
</tr>
<tr>
<td>CUIP</td>
<td>Cellular Universal IP</td>
</tr>
<tr>
<td>DAD</td>
<td>Duplicate Address Detection</td>
</tr>
<tr>
<td>DF</td>
<td>Don’t Fragment</td>
</tr>
<tr>
<td>DHCP</td>
<td>Dynamic Host Configuration Protocol</td>
</tr>
<tr>
<td>DHCPv6</td>
<td>DHCP for IPv6</td>
</tr>
<tr>
<td>DiffServ</td>
<td>Differentiated Services</td>
</tr>
<tr>
<td>DL-MAP</td>
<td>Downlink Map</td>
</tr>
<tr>
<td>DSCP</td>
<td>DiffServ Codepoint</td>
</tr>
<tr>
<td>DSL</td>
<td>Digital Subscriber Line</td>
</tr>
<tr>
<td>EAP</td>
<td>Extensible Authentication Protocol</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>EAP-FAST</td>
<td>EAP-based Flexible Authentication via Secure Tunnelling</td>
</tr>
<tr>
<td>EAPOL</td>
<td>EAP over LAN</td>
</tr>
<tr>
<td>EF</td>
<td>Expedited Forwarding</td>
</tr>
<tr>
<td>EMAP</td>
<td>Edge Mobility Anchor Point</td>
</tr>
<tr>
<td>EMD</td>
<td>Edge Mobility Domain</td>
</tr>
<tr>
<td>EMP</td>
<td>Edge Mobility Protocol</td>
</tr>
<tr>
<td>EP</td>
<td>Enforcement Point</td>
</tr>
<tr>
<td>ESP</td>
<td>Encapsulating Security Payload</td>
</tr>
<tr>
<td>FA</td>
<td>Foreign Agent</td>
</tr>
<tr>
<td>FBack</td>
<td>Fast Binding Acknowledgement</td>
</tr>
<tr>
<td>FBU</td>
<td>Fast Binding Update</td>
</tr>
<tr>
<td>FDD</td>
<td>Frequency Division Duplex</td>
</tr>
<tr>
<td>FEC</td>
<td>Forwarding Equivalence Class</td>
</tr>
<tr>
<td>FMIPv6</td>
<td>Fast Handovers for Mobile IPv6</td>
</tr>
<tr>
<td>FNA</td>
<td>Fast Neighbor Advertisement</td>
</tr>
<tr>
<td>FTN</td>
<td>FEC-to-NHLFE map</td>
</tr>
<tr>
<td>GCoA</td>
<td>Global CoA</td>
</tr>
<tr>
<td>GGSN</td>
<td>Gateway GPRS Support Node</td>
</tr>
<tr>
<td>GIF</td>
<td>GPRS Interworking Function</td>
</tr>
<tr>
<td>GMH</td>
<td>Generic MAC Header</td>
</tr>
<tr>
<td>GMM</td>
<td>GPRS Mobility Management</td>
</tr>
<tr>
<td>GPRS</td>
<td>General Packet Radio Service</td>
</tr>
<tr>
<td>GRE</td>
<td>Generic Routing Encapsulation</td>
</tr>
<tr>
<td>GSM</td>
<td>Global System for Mobile communications</td>
</tr>
<tr>
<td>GTC</td>
<td>Generic Token Card</td>
</tr>
<tr>
<td>GTP</td>
<td>GPRS Tunnelling Protocol</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>HA</td>
<td>Home Agent</td>
</tr>
<tr>
<td>HAck</td>
<td>Handover Acknowledgement</td>
</tr>
<tr>
<td>HAWAII</td>
<td>Handoff-Aware Wireless Access Internet Infrastructure</td>
</tr>
<tr>
<td>HDLC</td>
<td>High-level Data Link Control</td>
</tr>
<tr>
<td>HI</td>
<td>Handover Initiate</td>
</tr>
<tr>
<td>HLR</td>
<td>Home Location Register</td>
</tr>
<tr>
<td>HMIPv6</td>
<td>Hierarchical Mobile IPv6</td>
</tr>
<tr>
<td>HO</td>
<td>Handover</td>
</tr>
<tr>
<td>HoA</td>
<td>Home Address</td>
</tr>
<tr>
<td>HSDPA</td>
<td>High Speed Downlink Packet Access</td>
</tr>
<tr>
<td>HSUPA</td>
<td>High Speed Uplink Packet Access</td>
</tr>
<tr>
<td>IAPP</td>
<td>Inter-Access Point Protocol</td>
</tr>
<tr>
<td>ICMPv6</td>
<td>Internet Control Message Protocol version 6</td>
</tr>
<tr>
<td>IDMP</td>
<td>Intra-Domain Mobility Management Protocol</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>IHL</td>
<td>Internet Header Length</td>
</tr>
<tr>
<td>IKE</td>
<td>Internet Key Exchange</td>
</tr>
<tr>
<td>ILM</td>
<td>Incoming Label Map</td>
</tr>
<tr>
<td>IntServ</td>
<td>Integrated Services</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>IPCP</td>
<td>IP Control Protocol</td>
</tr>
<tr>
<td>IPsec</td>
<td>IP Security</td>
</tr>
<tr>
<td>IPv4</td>
<td>Internet Protocol version 4</td>
</tr>
<tr>
<td>IPv6</td>
<td>Internet Protocol version 6</td>
</tr>
<tr>
<td>IPX</td>
<td>Internet Packet Exchange</td>
</tr>
<tr>
<td>IS</td>
<td>Information Service</td>
</tr>
<tr>
<td>ABBREVIATIONS</td>
<td>Definition</td>
</tr>
<tr>
<td>---------------</td>
<td>------------</td>
</tr>
<tr>
<td>ISAKMP</td>
<td>Internet Security Association and Key Management Protocol</td>
</tr>
<tr>
<td>ISP</td>
<td>Internet Service Provider</td>
</tr>
<tr>
<td>LCoA</td>
<td>Local CoA</td>
</tr>
<tr>
<td>LCP</td>
<td>Link Control Protocol</td>
</tr>
<tr>
<td>LDP</td>
<td>Label Distribution Protocol</td>
</tr>
<tr>
<td>LEMA</td>
<td>Label Edge Mobility Agent</td>
</tr>
<tr>
<td>LER</td>
<td>Label Edge Router</td>
</tr>
<tr>
<td>LLC</td>
<td>Logical Link Control</td>
</tr>
<tr>
<td>LoS</td>
<td>Line-Of-Sight</td>
</tr>
<tr>
<td>LSP</td>
<td>Label Switched Path</td>
</tr>
<tr>
<td>LSR</td>
<td>Label Switching Router</td>
</tr>
<tr>
<td>MA</td>
<td>Mobility Agent</td>
</tr>
<tr>
<td>MAC</td>
<td>Medium Access Control</td>
</tr>
<tr>
<td>MAC</td>
<td>Message Authentication Code</td>
</tr>
<tr>
<td>MAN</td>
<td>Metropolitan Area Network</td>
</tr>
<tr>
<td>MAP</td>
<td>Mobile Application Part</td>
</tr>
<tr>
<td>MAP</td>
<td>Mobility Anchor Point</td>
</tr>
<tr>
<td>MD5</td>
<td>Message Digest 5</td>
</tr>
<tr>
<td>MF</td>
<td>More Fragments</td>
</tr>
<tr>
<td>MIPv4</td>
<td>Mobile IPv4</td>
</tr>
<tr>
<td>MIPv6</td>
<td>Mobile IPv6</td>
</tr>
<tr>
<td>MME</td>
<td>Mobility Management Entity</td>
</tr>
<tr>
<td>MN</td>
<td>Mobile Node</td>
</tr>
<tr>
<td>MOBIKE</td>
<td>IKEv2 Mobility and Multihoming</td>
</tr>
<tr>
<td>MPLS</td>
<td>Multi-Protocol Label Switching</td>
</tr>
<tr>
<td>MRE</td>
<td>Minimal Routing Encapsulation</td>
</tr>
</tbody>
</table>
MRU  Maximum-Receive-Unit
MS   Mobile Station
MS-CHAPv2 Microsoft PPP CHAP version 2
MSC  Mobile services Switching Center
MSK  Master Session Key
MSS  Mobile Subscriber Station
MTU  Maximum Transmission Unit
NAP  Network Access Provider
NAS  Network Access Server
NCP  Network Control Protocol
NHLFE Next-Hop Label Forwarding Entry
NIC  Network Interface Card
NID  Network ID
NLOS Non-Line-Of-Sight
NSAPI Network Service Access Point Identifier
NSP  Network Service Provider
OSPF Open Shortest Path First
OTP  One-Time Password
P-TMSI Packet Temporary Mobile Subscriber Identity
PAA  PANA Authentication Agent
PaC  PANA Client
PADI  PPPoE Active Discovery Initiation
PADO  PPPoE Active Discovery Offer
PADR  PPPoE Active Discovery Request
PADS  PPPoE Active Discovery Session-Confirmation
PADT  PPPoE Active Discovery Terminate
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PANA</td>
<td>Protocol for Carrying Authentication for Network Access</td>
</tr>
<tr>
<td>PAP</td>
<td>Password Authentication Protocol</td>
</tr>
<tr>
<td>PAR</td>
<td>Previous Access Router</td>
</tr>
<tr>
<td>PCoA</td>
<td>Previous CoA</td>
</tr>
<tr>
<td>PDCP</td>
<td>Packet Data Convergence Protocol</td>
</tr>
<tr>
<td>PDN</td>
<td>Packet Data Network</td>
</tr>
<tr>
<td>PDP</td>
<td>Packet Data Protocol</td>
</tr>
<tr>
<td>PDU</td>
<td>Packet Data Unit</td>
</tr>
<tr>
<td>PFA</td>
<td>Paging Foreign Agent</td>
</tr>
<tr>
<td>PHB</td>
<td>Per-Hop Behavior</td>
</tr>
<tr>
<td>PHP</td>
<td>Penultimate Hop Popping</td>
</tr>
<tr>
<td>PIN</td>
<td>Personal Identification Number</td>
</tr>
<tr>
<td>PKM</td>
<td>Privacy Key Management</td>
</tr>
<tr>
<td>PMIPv4</td>
<td>Proxy MIPv4</td>
</tr>
<tr>
<td>PMK</td>
<td>Pairwise Master Key</td>
</tr>
<tr>
<td>PoA</td>
<td>Point of Attachment</td>
</tr>
<tr>
<td>PoS</td>
<td>Point of Service</td>
</tr>
<tr>
<td>PPP</td>
<td>Point-to-Point Protocol</td>
</tr>
<tr>
<td>PPPoE</td>
<td>PPP over Ethernet</td>
</tr>
<tr>
<td>PSTN</td>
<td>Public Switched Telephone Network</td>
</tr>
<tr>
<td>PTK</td>
<td>Pairwise Transient Key</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>RA</td>
<td>Routing Area</td>
</tr>
<tr>
<td>RADIUS</td>
<td>Remote Access Dial In User Service</td>
</tr>
<tr>
<td>RAI</td>
<td>Routing Area Identifier</td>
</tr>
<tr>
<td>RAN</td>
<td>Radio Access Network</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>RANAP</td>
<td>Radio Access Network Application Part</td>
</tr>
<tr>
<td>RAU</td>
<td>Routing Area Update</td>
</tr>
<tr>
<td>RegReply</td>
<td>Registration Reply</td>
</tr>
<tr>
<td>RegReq</td>
<td>Registration Request</td>
</tr>
<tr>
<td>RLC</td>
<td>Radio Link Control</td>
</tr>
<tr>
<td>RNC</td>
<td>Radio Network Controller</td>
</tr>
<tr>
<td>RNS</td>
<td>Radio Network Subsystem</td>
</tr>
<tr>
<td>ROM</td>
<td>Read-Only Memory</td>
</tr>
<tr>
<td>RRC</td>
<td>Radio Resource Control</td>
</tr>
<tr>
<td>RSVP</td>
<td>Resource Reservation Protocol</td>
</tr>
<tr>
<td>RSVP-TE</td>
<td>Resource Reservation Protocol for Traffic Engineering</td>
</tr>
<tr>
<td>SAPI</td>
<td>Service Access Point Identifier</td>
</tr>
<tr>
<td>SCCP</td>
<td>Signalling Connection Control Part</td>
</tr>
<tr>
<td>SCTP</td>
<td>Stream Control Transmission Protocol</td>
</tr>
<tr>
<td>SGSN</td>
<td>Serving GPRS Support Node</td>
</tr>
<tr>
<td>SHA-1</td>
<td>Secure Hash Algorithm 1</td>
</tr>
<tr>
<td>SIM</td>
<td>Subscriber Identity Module</td>
</tr>
<tr>
<td>SM</td>
<td>Session Management</td>
</tr>
<tr>
<td>SNAP</td>
<td>SubNetwork-Dependent Application Protocol</td>
</tr>
<tr>
<td>SNDCP</td>
<td>SubNetwork-Dependent Convergence Protocol</td>
</tr>
<tr>
<td>SNMPv3</td>
<td>Simple Network Management Protocol version 3</td>
</tr>
<tr>
<td>SPI</td>
<td>Security Parameter Index</td>
</tr>
<tr>
<td>SSCS</td>
<td>Service-Specific Convergence Sublayer</td>
</tr>
<tr>
<td>SSID</td>
<td>Service Set ID</td>
</tr>
<tr>
<td>SSL</td>
<td>Secure Socket Layer</td>
</tr>
<tr>
<td>TCP</td>
<td>Transport Control Protocol</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>TDD</td>
<td>Time Division Duplex</td>
</tr>
<tr>
<td>TDM</td>
<td>Time Division Multiplexing</td>
</tr>
<tr>
<td>TeleMIP</td>
<td>Telecommunication Enhanced Mobile IP</td>
</tr>
<tr>
<td>TLLI</td>
<td>Temporary Logical Link Identifier</td>
</tr>
<tr>
<td>TLS</td>
<td>Transport Layer Security</td>
</tr>
<tr>
<td>TTL</td>
<td>Time to Live</td>
</tr>
<tr>
<td>UDP</td>
<td>User Datagram Protocol</td>
</tr>
<tr>
<td>UID</td>
<td>User ID</td>
</tr>
<tr>
<td>UMA</td>
<td>Unlicensed Mobile Access</td>
</tr>
<tr>
<td>UMTS</td>
<td>Universal Mobile Telecommunications System</td>
</tr>
<tr>
<td>VCI</td>
<td>Virtual Channel Identifier</td>
</tr>
<tr>
<td>VLR</td>
<td>Visitor Location Register</td>
</tr>
<tr>
<td>VPI</td>
<td>Virtual Path Identifier</td>
</tr>
<tr>
<td>VPN</td>
<td>Virtual Private Network</td>
</tr>
<tr>
<td>W-CDMA</td>
<td>Wideband Code Division Multiple Access</td>
</tr>
<tr>
<td>WAF</td>
<td>WLAN Adaptation Function</td>
</tr>
<tr>
<td>Wi-Fi</td>
<td>Wireless Fidelity</td>
</tr>
<tr>
<td>WiMAX</td>
<td>Worldwide inter-operability for Microwave Access</td>
</tr>
</tbody>
</table>
ABBREVIATIONS
Bibliography


IEEE. 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. IEEE 802.11F, July 2003.

IEEE. Part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications – amendment 6: Medium access control (mac) security enhancements (802.11i). IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements, July 2004.


Curriculum Vitae

Name Rajendra Persaud
Date of birth March 31, 1976
Birth-place Linnich

Education
11/2002 - 10/2006 Scientific employee at the chair of computer science 4
RWTH Aachen University, Aachen, Germany
Cooperation with Nokia Networks, Düsseldorf, Germany
U.S. patent application

10/1996 - 09/2002 Studies in computer science
RWTH Aachen University, Aachen, Germany
Acquisition of the “Diplom” (German graduation certificate)

10/1999 - 09/2000 Studies in computer science (DAAD scholarship)
Université Pierre et Marie Curie, Paris, France
Acquisition of the “Maîtrise” (French graduation certificate)

06/1995 - 06/1996 Civilian service
Old people’s home “Haus Berg”, Brachelen, Germany

08/1986 - 06/1995 Secondary education
Cusanus-Gymnasium, Erkelenz, Germany
Acquisition of the “Abitur” (German secondary school graduation certificate)