บทที่ 5

การทดสอบอัลกอริทึมการทำการเคลื่อนย้ายเปลี่ยนระบบเครือข่าย

สำหรับในบทนี้จะชี้ให้เห็นถึงปัญหาในการจัดการเหตุการณ์การเคลื่อนย้ายเปลี่ยนระบบเครือข่าย และแนวทางจะการแก้ไขปัญหาดังกล่าว ซึ่งสามารถแบ่งออกเป็น 4 หัวข้อหลักๆ หัวข้อที่ 5.1จะอธิบายถึงปัญหาทั้งหมดที่เกิดขึ้นเมื่อทำการเคลื่อนย้ายเปลี่ยนระบบเครือข่าย หัวข้อที่ 5.2 เป็นการเสนอการแก้ปัญหาโดยใช้โปรโตคอล SIP พร้อมผลการทดสอบ ในขณะที่หัวข้อที่ 5.3 เป็นการเสนอการแก้ปัญหาโดยใช้ Mobile IPv6 พร้อมผลการทดสอบและในหัวข้อสุดท้ายจะเป็นการเปรียบเทียบถึงข้อดี ข้อเสีย ระหว่างการใช้โปรโตคอล SIP และ Mobile IPv6 ในการจัดการเหตุการณ์การเคลื่อนย้ายเปลี่ยนระบบเครือข่าย โดยข้อมูลค่าเวลาหน่วงต่างๆ ได้จากการทดสอบการทำงานและโปรแกรมประยุกต์ซึ่งใช้ในการทดสอบระบบเครือข่าย SIP/MIPv6 ในบทที่ 4

5.1 ปัญหาเมื่อเกิดการเคลื่อนย้ายเปลี่ยนระบบเครือข่าย

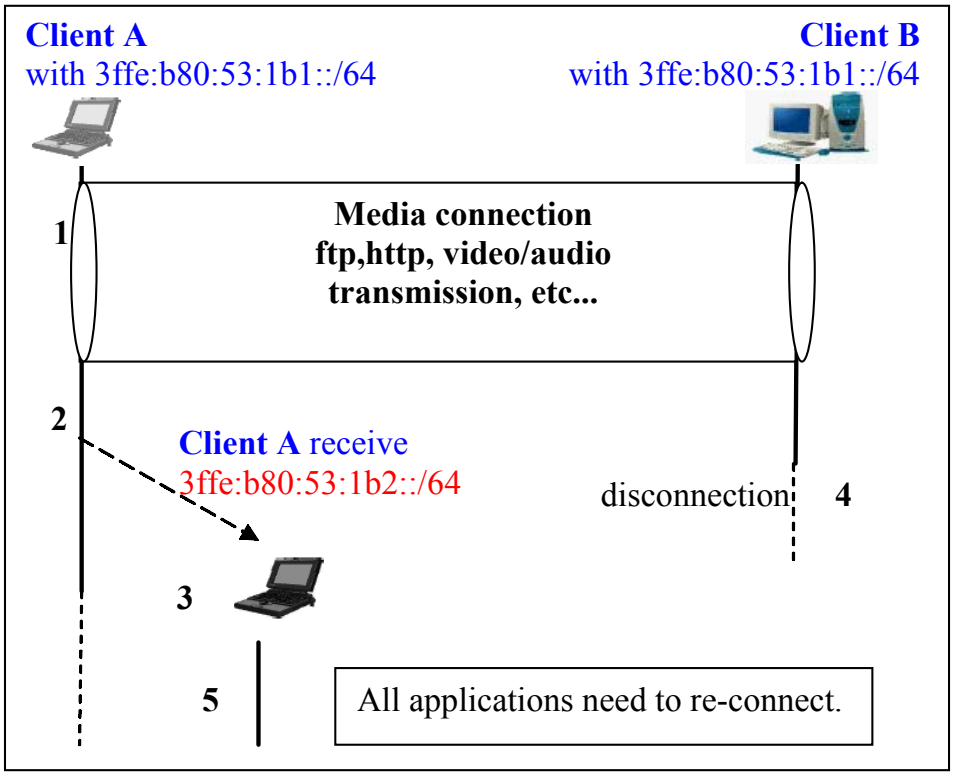
สำหรับหัวข้อนี้จะแสดงปัญหาที่เกิดขึ้นเมื่อ Mobile Node ทำการเคลื่อนย้ายเปลี่ยนระบบเครือข่ายในขณะที่ Mobile Node ทำการเชื่อมต่อหรือมีการส่งและรับข้อมูลกับ Correspond Node ซึ่งแสดงได้ดังรูปที่ 38

รูปที่ 38 มีลำดับขั้นตอนดังนี้

1) Client A ทำการติดต่อไปยัง Client B ด้วยโปรแกรมประยุกต์ต่างๆ รวมถึงมีการใช้งานโปรแกรมประยุกต์ที่มีการเชื่อมต่อกับระบบเครือข่ายไอพี

2) เมื่อ Client A เกิดการเคลื่อนย้ายเปลี่ยนระบบเครือข่าย

3) Client A ได้รับหมายเลข Prefix ใหม่ทำให้หมายเลขไอพีของ Client A มีการเปลี่ยนแปลง ดังนั้นเป็นผลให้โปรแกรมประยุกต์ทั้งหมดที่มีการใช้งานอยู่ไม่สามารถใช้งานต่อไปได้



รูปที่ 38 แสดงปัญหาเมื่อเกิดการเคลื่อนย้ายเปลี่ยนระบบเครือข่าย

4) Client B จะขาดการติดต่อจาก Client A ทันที และเมื่อ Client B ต้องการติดต่อไปยัง Client A ใหม่อีกครั้ง จะไม่สามารถทำได้ เนื่องจาก Client B ไม่ทราบถึงข้อมูลหมายเลขไอพีของ Client A ที่ได้รับใหม่

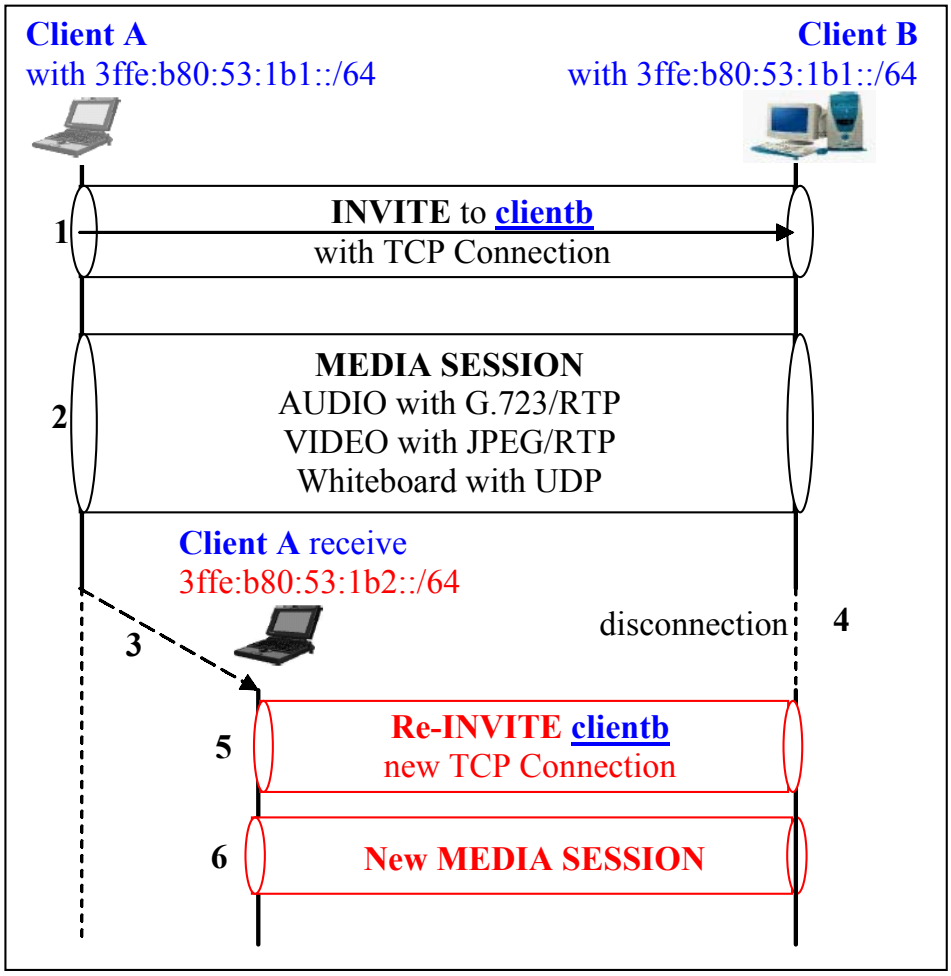
5) Client A จำเป็นต้องทำการเชื่อมต่อใหม่ทั้งหมดในทุกๆ โปรแกรมประยุกต์ที่มีการใช้งานอยู่เดิม

ดังนั้นจะพบว่า เกิดปัญหาขึ้นทันทีเมื่อ Client A ทำการเคลื่อนย้ายเปลี่ยนระบบเครือข่าย นอกจากนี้ในกรณีที่ Client A ทำการเคลื่อนที่ตลอดเวลาซึ่งทำให้ Client A ได้รับค่า Prefix ใหม่และมีการเปลี่ยนแปลงหมายเลขไอพีตลอดเวลา จะเป็นผลให้เกิดการสร้างการเชื่อมต่อใหม่บ่อยครั้ง ซึ่งไม่เป็นที่ต้องการต่อการติดต่อสื่อสารแบบทันเวลา หรือในการส่งข้อมูลภาพและเสียง ดังนั้นการใช้โปรโตคอล SIP และ Mobile IP อาจช่วยแก้ไขปัญหาเหล่านี้ได้ ซึ่งการทดสอบการใช้โปรโตคอล SIP และ Mobile IP ในการแก้ไขปัญหาดังกล่าว จะอธิบายในหัวข้อที่ 5.2 และ 5.3

5.2 ผลการทดสอบการทำ Hand Over ด้วยโปรโตคอล SIP

สรุปขั้นตอนการทดสอบ

สำหรับการทดสอบนี้ เป็นการทดสอบเพื่อตรวจจับค่าเวลาหน่วงในแต่ละขั้นตอน รวมทั้งศึกษาปัญหาและความสามารถเมื่อทำการ Hand Over ด้วยโปรโตคอล SIP โดยการทดสอบเป็นไปดังรูปที่ 39



รูปที่ 39 แสดงการทดสอบการทำโมไบลิตี้ขณะ Hand Over ด้วยโปรโตคอล SIP

จากรูปที่ 39 แสดงลำดับขั้นการทำงานในการทดสอบ โดยใช้โปรแกรม SIP Client ที่ได้ทำการพัฒนาขึ้น ซึ่งโปรแกรม SIP Client นี้มีความสามารถในการให้บริการพหุสื่อได้ 3 ชนิดด้วยกัน ได้แก่ รับ-ส่งข้อมูลเสียงด้วยรูปแบบการเข้ารหัสแบบ G.723 รับ-ส่งข้อมูลภาพด้วยรูปแบบการเข้ารหัสแบบ JPEG และวาดภาพผ่านบริการกระดานสนทนา (Whiteboard) โดยใช้โปรโตคอล RTP สำหรับทั้งการรับและส่งข้อมูลภาพและเสียง และใช้โปรโตคอล UDP ในการส่งข้อมูลของกระดานสนทนา โดย clienta และ clientb ทำการรันโปรแกรม SIP Client ไว้แล้วเริ่มทำการติดต่อเป็นไปตามลำดับดังนี้

1) Client A ทำการโทรไปยัง Client B โดยกำหนดใช้โปรโตคอล TCP สำหรับการส่งสัญญาณควบคุมด้วยโปรโตคอล SIP เพื่อร้องขอเริ่มต้นการติดต่อ (Call Setup Session)
2) เมื่อการร้องขอการติดต่อได้รับการยอมรับโดย Client B แล้ว ทั้ง Client A และ Client B จึงเริ่มต้นสร้าง Media Session สำหรับการรับและส่งข้อมูลพหุสื่อ
3) เมื่อ Client A และ Client B ทำการสนทนาได้ระยะหนึ่ง Client A ซึ่งเป็นเครื่องคอมพิวเตอร์แบบไร้สายทำการเคลื่อนย้ายเข้าสู่ระบบเครือข่ายใหม่ และได้รับค่าหมายเลข prefix ใหม่คือ 3ffe:b80:53:1b2::/64 มีผลให้หมายเลขไอพีเปลี่ยนแปลงไป และไม่สามารถใช้หมายเลขไอพีเดิมได้อีก
4) เมื่อหมายเลขไอพีเดิมของ Client A ไม่สามารถใช้งานได้ เป็นผลให้ Client A ขาดการเชื่อมต่อจาก Client B และ Client B รับรู้การขาดการติดต่อจาก Client A จึงทำการปิดช่องสัญญาณทั้งหมด
5) เมื่อ Client A รับรู้การเปลี่ยนหมายเลขไอพีใหม่แล้ว จึงทำการส่งสัญญาณ re-INVITE ไป Client B เพื่อร้องขอการติดต่อใหม่อีกครั้ง
6) เมื่อ Client B ได้รับสัญญาณ re-INVITE แล้วทำการตรวจสอบพบว่ามาจาก Client A จึงตอบรับ จากนั้นทั้ง Client A และ B จึงเริ่มต้นสร้างช่องสัญญาณและรับ-ส่งข้อมูลพหุสื่อใหม่อีกครั้งหนึ่ง โดย Media Session ใหม่นี้เป็นคนละ Session กับ Media Session ในตอนเริ่มต้น

เมื่อเสร็จสิ้นการทำงานดังกล่าวเป็นผลให้ Client A สามารถกลับมารับ-ส่งข้อมูลพหุสื่อกับ Client B ได้อีกครั้ง เมื่อทำการเคลื่อนย้ายเปลี่ยนระบบเครือข่ายเรียบร้อยแล้ว หรือเมื่อหมายเลขไอพีของ Client A เปลี่ยนแปลงไป

ผลการทดสอบ

จากการทดสอบดังกล่าวสามารถตรวจจับค่าเวลาหน่วงตามขั้นตอนต่างๆดังต่อไปนี้

D_{MD} (Movement Detection Delay) - ค่าเวลาหน่วงในการตรวจสอบหา AP (Access Point) ใหม่ โดยจับเวลานับตั้งแต่เมื่อ Mobile Node (ในที่นี้คือ client A) ขาดการติดต่อจาก AP ตัวปัจจุบันแล้วทำการตรวจหาและเลือก AP ใหม่ได้

D_{RA} (Router Advertisement Delay) – ค่าเวลาหน่วงในการได้รับสัญญาณ Router Advertisement จาก Router ในระบบเครือข่ายนั้น โดยจับเวลานับตั้งแต่เมื่อ Mobile Node เชื่อมต่อกับ AP ใหม่เสร็จสิ้นจนกระทั้งได้รับสัญญาณ Router Advertisement ซึ่งค่าเวลาหน่วงนี้จะขึ้นอยู่กับช่วงเวลาที่ Mobile Node ทำการเชื่อมต่อเข้าสู่ระบบนี้และอัตราการส่งสัญญาณ Router Advertisement ที่มีการกำหนดไว้ที่ตัว Router สำหรับในที่นี้ได้ทำการกำหนดเวลาในการส่งสัญญาณ Router Advertisement ทุกๆ 1 วินาที

$D_{Binding}$ (Binding Update Delay) – ค่าเวลาหน่วงในการทำกระบวนการ Binding Update ซึ่งในที่นี้เริ่มนับเวลาตั้งแต่ client A ทำการส่งสัญญาณ re-INVITE ไปยัง client B จนกระทั้ง client A ได้รับสัญญาณตอบกลับจาก client B เสร็จสิ้น

$D_{Processing}$ (Processing Delay) – ค่าเวลาหน่วงในการสร้างช่องสัญญาณใหม่ โดยเริ่มนับเวลาตั้งแต่เริ่มสร้างช่องสัญญาณในการติดต่อด้วยโปรโตคอล SIP ใหม่ (ในที่นี้คือการสร้างช่องสัญญาณ TCP) จากนั้นจึงสร้างช่องสัญญาณสำหรับส่งข้อมูลพหุสื่อชนิดเสียง, ภาพ และกระดานสนทนา ตามลำดับ สำหรับการสร้างช่องสัญญาณในการส่งข้อมูลและรับข้อมูลพหุสื่อชนิดเสียงและภาพนั้นจะจับเวลาแยกกัน 2 ส่วน คือ ส่วนการส่งข้อมูลพหุสื่อ จะนับเวลาตั้งแต่เริ่มสร้างช่องสัญญาณเพื่อส่งข้อมูลพหุสื่อจนเสร็จ (transmit time) และ ส่วนการรับข้อมูลพหุสื่อ จะนับเวลาตั้งแต่เริ่มสร้างช่องสัญญาณเพื่อรอรับข้อมูลพหุสื่อจนกระทั้งได้รับข้อมูลพหุสื่อ (receive time)

$D_{HandOver}$ (Hand Over Delay) – ค่าเวลาหน่วงในการทำ Hand Over ทั้งหมด โดยนับเวลาตั้งแต่เมื่อ Mobile Node ขาดการติดต่อจาก AP ปัจจุบันจนกระทั้ง Mobile Node เลือก AP ใหม่และสามารถได้รับข้อมูลพหุสื่อได้ใหม่ ซึ่งมีค่าดังนี้

$$D_{Handover} = D_{MD} + D_{RA} + D_{Binding} + D_{Processing}$$

การตรวจจับค่าเวลาหน่วงทั้งหมดเป็นไปดังตารางที่ 7

ตารางที่ 7 ค่าเวลาหน่วงเมื่อทำ Hand Over ด้วยโปรโตคอล SIP

ครั้งที่	D_{MD} (ms.)	D_{RA} (ms.)	$D_{Binding}$ (ms.)	$D_{Processing}$ (ms.)		
				SIP/TCP	AUDIO/RTP	Whiteboard/UDP
1	334	137	140	80	194	110
2	444	1453	103	60	190	109
3	473	1123	95	70	140	109
4	476	121	98	70	182	110
5	351	405	117	71	203	112
6	455	1072	99	61	158	111
7	394	536	109	70	188	124
8	384	827	96	70	162	109
9	429	95	110	90	159	113
10	376	235	96	60	240	118
Total	4116	6004	1063	702	1816	1125
Average	411.6	600.4	106.3	70.2	181.6	112.5

เนื่องจากในส่วนการสร้างช่องสัญญาณสำหรับรับและส่งข้อมูลพหุสื่อใหม่ หรือช่วงเวลาของ Processing Delay นั้นจะเป็นการทำงานแบบพร้อมๆกันหรือ Multi-Threading Processing ดังนั้นค่าเวลาหน่วงที่ได้เป็นไปดังนี้

การสร้างช่องสัญญาณในการรับและส่งข้อมูลพหุสื่อแบบขนาน (Parallel/Multithreading)

parallel processing = max(sip_tcp, audio_rtp, whiteboard_udp)

นั่นคือ ค่า $D_{Processing}$ มีค่าเท่ากับ ค่าเวลามากที่สุดค่าเวลาในการสร้างช่องสัญญาณสำหรับโปรโตคอล SIP และค่าเวลาในการสร้างช่องสัญญาณของข้อมูลเสียง และกระดานสนทนา

โดยสามารถสรุปค่า $D_{HandOver}$ ได้ดังตารางที่ 8

ตารางที่ 8 ค่าเวลาหน่วงสรุปเมื่อทำ Hand Over ด้วยโปรโตคอล SIP

D_{MD} (ms)	D_{RA} (ms)	$D_{Binding}$ (ms)	$D_{Processing}$ (ms)	$D_{HandOver}$ (ms)
411.6	600.4	106.3	181.6	1299.9

จากตารางที่ 8 แสดงให้เห็นว่าค่า $D_{HandOver}$ ในการใช้โปรโตคอล SIP จัดการเหตุการณ์ ซึ่งหมายความว่า Mobile Node ใช้เวลาในการขาดการติดต่อจาก Access Point ตัวเดิมแล้วทำการหา Acess Point ตัวใหม่พร้อมกับทำการสร้างช่องสัญญาณในการรับและส่งข้อมูลอย่างต่อเนื่องด้วยเวลาประมาณ 1299.9 ms แต่อย่างไรก็ตามการจัดการ Hand Over ด้วยโปรโตคอล SIP นั้นสามารถจัดการได้เฉพาะโปรแกรมประยุกต์ที่อยู่ในความรับผิดชอบของโปรโตคอล SIP เท่านั้น ไม่สามารถรองรับหรือควบคุมช่องสัญญาณอื่นๆ ที่มีการเชื่อมต่ออยู่ภายในระบบปฏิบัติการณ์เดียวกันได้ นั่นคือ ในขณะที่เกิดการเปลี่ยนระบบเครือข่ายโปรแกรมประยุกต์อื่นๆ หรือบริการอื่นๆ ได้แก่ เว็บบราว์เซอร์, FTP และโปรแกรมที่ใช้งานในระบบอินเตอร์เน็ต จำพวก Chat เป็นต้น เหล่านี้จะขาดการติดต่อทั้งหมด

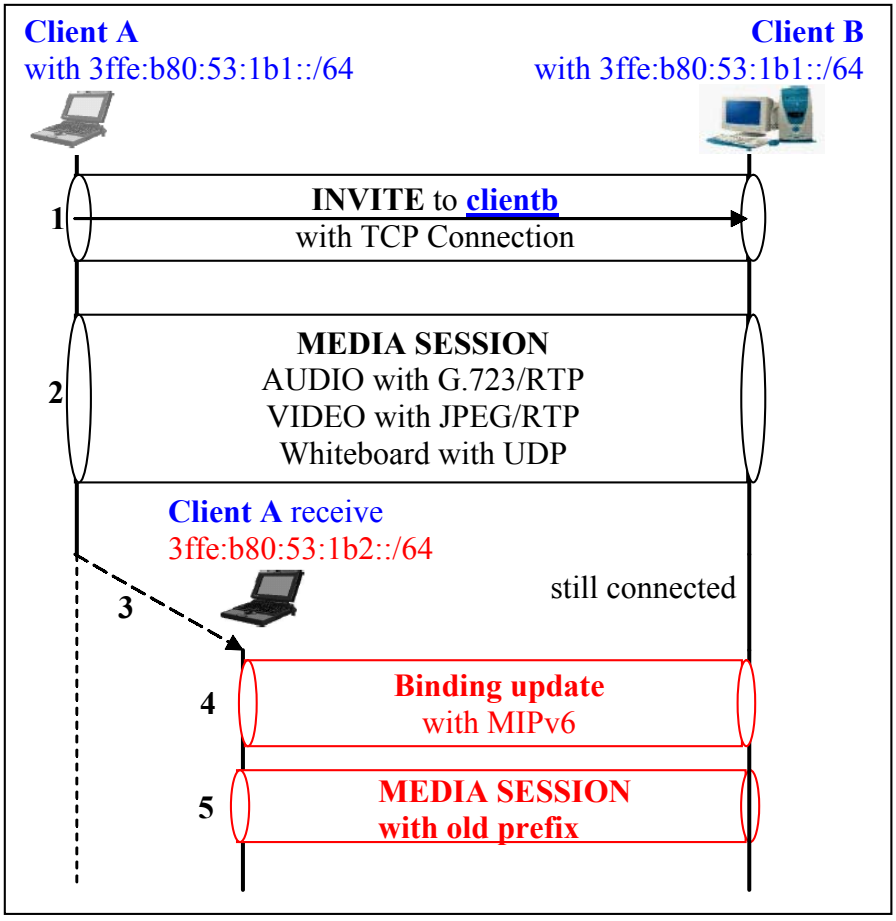
5.3 ผลการทดสอบการทำ Hand Over ด้วย MIPv6

สรุปขั้นตอนการทดสอบ

สำหรับการทดสอบนี้ เป็นการทดสอบเพื่อตรวจจับค่าเวลาหน่วงในแต่ละขั้นตอน รวมทั้งศึกษาปัญหาและความสามารถเมื่อทำการ Hand Over ด้วย MIPv6 โดยการทดสอบเป็นไปดังรูปที่ 40

รูปที่ 40 แสดงการทดสอบการทำ Hand Over โดยใช้ MIPv6 แทนโปรโตตอล SIP ซึ่งได้ทำการทดสอบในลักษณะเดียวกันกับการทดสอบในหัวข้อที่ 5.2 แต่ขณะที่เกิด Hand Over ขึ้นได้ใช้ MIPv6 มาใช้งานเกี่ยวกับการทำ Binding Update แทนโปรโตคอล SIP โดยมีลำดับขั้นตอน เป็นไปดังนี้

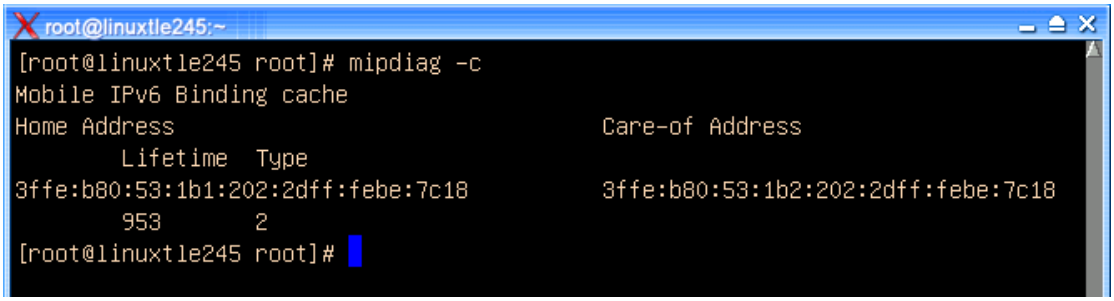
1. Client A ทำการโทรไปยัง Client B โดยกำหนดใช้โปรโตคอล TCP สำหรับการส่งสัญญาณควบคุมด้วยโปรโตคอล SIP เพื่อร้องขอเริ่มต้นการติดต่อ (Call Setup Session)
2. เมื่อการร้องขอการติดต่อได้รับการยอมรับโดย Client B แล้ว ทั้ง Client A และ Client B จึงเริ่มต้นสร้าง Media Session สำหรับการรับและส่งข้อมูลพหุสื่อ



รูปที่ 40 แสดงการทดสอบการทำโมไบลิตี้ขณะ Hand Over ด้วย MIPv6

3. เมื่อ Client A และ Client B ทำการสนทนาได้ระยะหนึ่ง Client A ซึ่งเป็นเครื่องคอมพิวเตอร์แบบไร้สายทำการเคลื่อนย้ายเข้าสู่ระบบเครือข่ายใหม่ และได้รับค่าหมายเลข prefix ใหม่คือ 3ffe:b80:53:1b2::/64 มีผลให้หมายเลขไอพีเปลี่ยนแปลงไป แต่การเปลี่ยนไปดังกล่าว ไม่เป็นผลให้หลุดการติดต่อจาก Client B เนื่องจากการทำงานของ MIPv6 ในระดับชั้น Network ควบคุมการทำงานในขาดการติดต่อนี้อยู่ จึงเป็นผลให้ Client B ยังคงอยู่ในสถานะการติดต่อ
4. เมื่อ Client A รับรู้การเปลี่ยนหมายเลขไอพีใหม่แล้ว จึงทำการส่งสัญญาณ Binding Update ไป Client B เพื่อระบุการเปลี่ยนแปลงหมายเลขไอพีโดยเปลี่ยน prefix เป็น 3ffe:b80:53:1b1::/64 ซึ่งการทำงานในส่วนนี้ทำในระดับ Network ดังนั้นโปรแกรม SIP Client ยังคงใช้หมายเลขไอพีเดิมของ Client A ในการรับ-ส่งข้อมูลต่อไป
5. Client A และ Client B สามารถทำการรับ-ส่งข้อมูลได้อย่างต่อเนื่องโดยยังคงใช้หมายเลขไอพีเดรสเดิมด้วย prefix 3ffe:b80:53:1b1::/64 ในการติดต่อ ดังนั้น Media Session จึงยังคงเป็น Session เดิมตลอดการติดต่อ

จากการขั้นตอนการทำงานดังกล่าว แสดงให้เห็นการทำงานของ MIPv6 ในการทำ Hand Over ซึ่งสามารถดูค่า Binding Update List ได้ที่ Client B โดยใช้คำสั่ง mipdiag –c ดังรูปที่ 41



รูปที่ 41 แสดงข้อมูล Binding Update ของ Client B

อย่างไรก็ตามหาก Client A ทำการเคลื่อนที่ข้ามไปยังระบบเครือข่ายต่อไปและได้รับข้อมูล prefix ใหม่ จึงทำการส่งสัญญาณ Binding Update มายังเครื่อง Client B ข้อมูล Binding Update List นี้จะถูกแก้ไข และเหลือเพียงข้อมูลหมายเลขไอพีด้วยค่า prefix 3ffe:b80:53:1b1::/64 กับค่าข้อมูลหมายเลขไอพีใหม่เท่านั้น นั่นคือ ค่า Care-of Address สามารถเปลี่ยนแปลงไปขึ้นอยู่กับระบบเครือข่ายใหม่

ผลการทดสอบ

จากการทดสอบดังกล่าวสามารถตรวจจับค่าเวลาหน่วงตามขั้นตอนต่างๆดังต่อไปนี้

D_{MD} (Movement Detection Delay) - ค่าเวลาหน่วงในการตรวจสอบหา AP (Access Point) ใหม่ โดยจับเวลานับตั้งแต่เมื่อ Mobile Node (ในที่นี้คือ client A) ขาดการติดต่อจาก AP ตัวปัจจุบันแล้วทำการตรวจหาและเลือก AP ใหม่ได้

D_{RA} (Router Advertisement Delay) – ค่าเวลาหน่วงในการได้รับสัญญาณ Router Advertisement จาก Router ในระบบเครือข่ายนั้น โดยจับเวลานับตั้งแต่เมื่อ Mobile Node เชื่อมต่อกับ AP ใหม่เสร็จสิ้นจนกระทั้งได้รับสัญญาณ Router Advertisement ซึ่งค่าเวลาหน่วงนี้จะขึ้นอยู่กับช่วงเวลาที่ Mobile Node ทำการเชื่อมต่อเข้าสู่ระบบนี้และอัตราการส่งสัญญาณ Router Advertisement ที่มีการกำหนดไว้ที่ตัว Router สำหรับในที่นี้ได้ทำการกำหนดเวลาในการส่งสัญญาณ Router Advertisement ทุกๆ 1 วินาที

$D_{Binding}$ (Binding Update Delay) – ค่าเวลาหน่วงในการทำกระบวนการ Binding Update ซึ่งในที่นี้เริ่มนับเวลาตั้งแต่ client A ทำการส่งสัญญาณ Binding Update ไปยัง client B จนกระทั้ง client Aได้รับสัญญาณตอบกลับจาก client B และปรากฏค่า Binding Update List ที่เครื่อง client B

$D_{HandOver}$ (Hand Over Delay) – ค่าเวลาหน่วงในการทำ Hand Over ทั้งหมด โดยนับเวลาตั้งแต่เมื่อ Mobile Node ขาดการติดต่อจาก AP ปัจจุบันจนกระทั้ง Mobile Node เลือก AP ใหม่และสามารถได้รับข้อมูลพหุสื่อได้ใหม่ ซึ่งมีค่าดังนี้ $D_{Handover} = D_{MD} + D_{RA} + D_{Binding}$

การตรวจจับค่าเวลาหน่วงทั้งหมดเป็นไปดังตารางที่ 9

ตารางที่ 9 ค่าเวลาหน่วงเมื่อทำ Hand Over ด้วย MIPv6

ครั้งที่	D_{MD} (ms)	D_{RA} (ms)	$D_{Binding}$ (ms)
1	304	1281	1000
2	318	384	995
3	477	102	1002
4	460	709	1000
5	479	495	998
6	441	291	992
7	381	462	1000
8	213	1013	998
9	309	212	997
10	382	116	995
total	3764	5065	9977
average	376.4	506.5	997.7

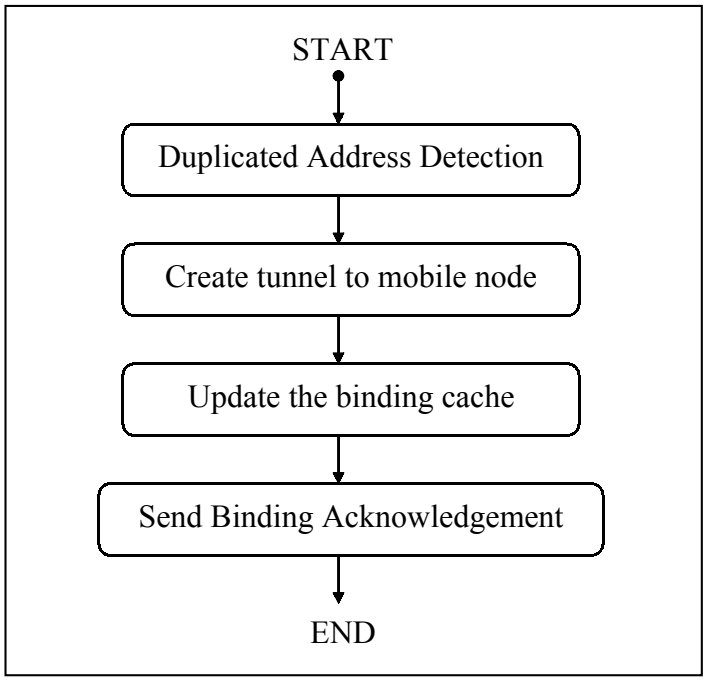
โดยสามารถสรุปค่า $D_{HandOver}$ ได้ดังตารางที่ 10

ตารางที่ 10 ค่าเวลาหน่วงสรุปเมื่อทำ Hand Over ด้วย MIPv6

D_{MD} (ms)	D_{RA} (ms)	$D_{Binding}$ (ms)	$D_{HandOver}$ (ms)
376.4	506.5	997.7	1880.6

สำหรับค่าเวลาหน่วงในการทำ Binding Update พบว่ามีค่าประมาณ 997.7 ms ซึ่งเป็นการจับเวลานับตั้งแต่เครื่อง Mobile Node ทำการส่งสัญญาณ Binding Update ไปยังเครื่องปลายทางจนกระทั่ง

ได้รับสัญญาณ Binding Acknowledgement ตอบกลับมา โดยสามารถแยกแยะกระบวนการทำงาน ณ เครื่องคู่การติดต่อได้ดังนี้



รูปที่ 42 แสดงกระบวนการทำงาน ณ เครื่องคู่การติดต่อ

จากรูปที่ 42 แสดงขั้นตอนการทำงาน ณ เครื่องคู่การติดต่อ ซึ่งจะเริ่มทำงานในขั้นตอนดังกล่าวทันทีที่ได้รับสัญญาณ Binding Update จากเครื่อง Mobile Node โดยสามารถอธิบายแต่ละขั้นตอนได้ดังนี้

1. Duplicated Address Detection เป็นกระบวนการทำงานที่ทำการตรวจสอบว่าค่า prefix หรือหมายเลขไอพีใหม่ที่เครื่อง Mobile Node ได้รับนั้นมีค่าซ้ำหรือมีการใช้งานอยู่ภายในระบบเครือข่ายนี้หรือไม่

2. ทำการสร้างหน่วยการทำงานเกี่ยวกับการทำ Tunneling เพื่อที่จะทำการ Encapsulate ทุกๆแพ็กเกตไปยังหมายเลขไอพีใหม่ของเครื่อง Mobile Node

3. ทำการเก็บค่าหมายเลขไอพีใหม่ไว้ใน Binding Cache

4. ส่งสัญญาณ Binding Acknowledgement ตอบกลับไปยังเครื่อง Mobile Node

ซึ่งจากการตรวจจับเวลาพบว่า ช่วงการทำงานที่ทำให้เกิดค่าเวลาหน่วงมากที่สุดคือ การทำ Duplicated Address Detection ซึ่งจะใช้เวลาประมาณ 900 ms ในขณะที่ขั้นตอนอื่นๆใช้เวลาเฉลี่ยไม่เกิน 100 ms และการทำงานของ MIPv6 จะทำให้ทุกๆแพ็กเกตมีการทำ Encapsulate ซึ่งจะทำให้เกิดค่าเวลาหน่วงประมาณ 18.2 ms

อย่างก็ตามการทำงานของ MIPv6 สามารถจัดการเหตุการณ์ Hand Over โดยรักษาการเชื่อมต่อของช่องสัญญาณของทุกๆโปรแกรมประยุกต์ที่มีการทำงานภายในระบบปฏิบัติการณ์เดียวกันได้

5.4 การวิเคราะห์ความสามารถในการทำ Hand Over ด้วยโปรโตคอล SIP และ MIPv6

จากการทดสอบการทำ Hand Over ด้วยโปรโตคอล SIP และ MIPv6 สามารถแยกวิเคราะห์ตามประเด็นที่น่าสนใจ ได้ 7 ประเด็นด้วยกัน ดังนี้

ค่าเวลาหน่วงเมื่อเกิด Hand Over (Hand Over Delay)

ค่าเวลาหน่วงสรุปในการทดสอบการทำ Hand Over ด้วยโปรโตคอล SIP และ MIPv6

ตารางที่ 11 แสดงค่าเวลาหน่วงสรุปในการทำ Hand Over ด้วยโปรโตคอล SIP และ MIPv6

ประเภท	เวลา (ms.)				
	D_{MD}	D_{RA}	$D_{Binding}$	$D_{Processing}$	$D_{HandOver}$ (x = 0, y = 0)
SIP	x	y	106.3	181.6	287.9
MIPv6	x	y	997.7	-	997.7

โดยในที่นี้กำหนดให้ค่า x และ y มีค่าเท่ากับ 0 เนื่องจากต้องการให้เห็นถึงความแตกต่างระหว่างการใช้โปรโตคอล SIP และ MIPv6 ในการทำ Hand Over อีกทั้งค่า D_{MD} และ D_{RA} ไม่เกี่ยวข้องกับอัลกอริทึมการทำ Hand Over แต่เป็นผลมาจากความสามารถของอุปกรณ์และความสามารถของ Router ภายในระบบเครือข่าย

และสามารถสรุปค่าเวลาหน่วงเมื่อเกิด Hand Over หรือ $D_{Handover}$ ได้ เป็นไปดังสมการต่อไปนี้

Hand Over using SIP

$$D_{Handover} = D_{MD} + D_{RA} + D_{Binding} + D_{Processing}$$

$D_{Handover}$ คือ ค่าเวลาหน่วงที่เกิดขึ้นตั้งแต่เกิด Hand Over จนกระทั้งสามารถส่งและรับข้อมูลมีเดียชุดถัดไปได้ถูกต้อง

D_{MD} คือ ค่าเวลาหน่วงในการตรวจสอบการเปลี่ยน subnet (Movement Detection Delay)

D_{RA} คือ ค่าเวลาหน่วงในการได้รับสัญญาณ Router Advertisement ใหม่

$D_{Binding}$ คือ ค่าเวลาหน่วงในการส่งสัญญาณ Binding Update เพื่อ update ค่าหมายเลข IP Address

$D_{Processing}$ คือ ค่าเวลาหน่วงในการสร้างช่องสัญญาณในการส่งและรับข้อมูลใหม่

รูปที่ 43 แสดงสมการเพื่อหาค่า Hand Over Delay ของการใช้ SIP ในการทำ Hand Over

Hand Over using MIPv6

$$D_{Handover} = D_{MD} + D_{RA} + D_{Binding}$$

$D_{Handover}$ คือ ค่าเวลาหน่วงที่เกิดขึ้นตั้งแต่เกิด Hand Over จนกระทั้งสามารถส่งข้อมูลชุดถัดไปได้ถูกต้อง

D_{MD} คือ ค่าเวลาหน่วงในการตรวจสอบการเปลี่ยน subnet (Movement Detection Delay)

D_{RA} คือ ค่าเวลาหน่วงในการได้รับสัญญาณ Router Advertisement ใหม่

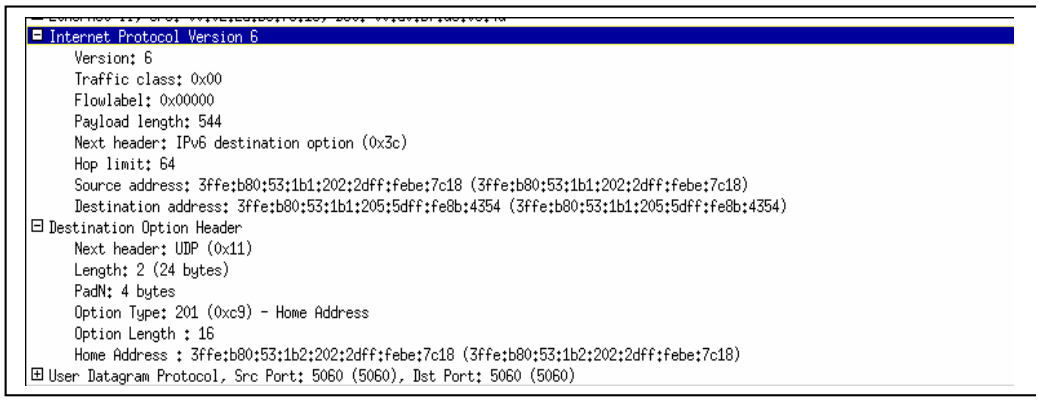
$D_{Binding}$ คือ ค่าเวลาหน่วงในการส่งสัญญาณ Binding Update เพื่อ update ค่าหมายเลข IP Address

รูปที่ 44 สมการเพื่อหาค่า Hand Over Delay ของการใช้ MIPv6 ในการทำ Hand Over

จากตารางที่ 11 พบว่าการใช้ MIPv6 ทำให้ค่าเวลาหน่วงโดยรวมในการทำ Hand Over มีค่าต่างจากการใช้โปรโตคอล SIP อยู่ประมาณ 709.8 ms เนื่องจาก MIPv6 ช่วยให้การเชื่อมต่อระหว่างการติดต่อไม่ขาดการติดต่อและการเปลี่ยนแปลงของหมายเลขไอพีถูกซ่อนจากการทำงานในระดับ Application ทำให้ไม่เกิด $D_{Processing}$ ขึ้น ดังนั้นแม้ว่ากระบวนการทำงานของ MIPv6 จะให้ค่าเวลาหน่วงมากกว่าการใช้โปรโตคอล SIP แต่ MIPv6 สามารถรักษาการเชื่อมต่อของทุกๆช่องสัญญาณได้ในขณะเกิด Hand Over ขึ้น ในบทที่ 6 จะมีการเสนออัลกอริทึมในการใช้ทั้ง 2 โปรโตคอลทำงานร่วมกันเพื่อให้เกิดค่าหน่วงเวลาที่เหมาะสมและให้ประโยชน์ต่อการจัดการเหตุการณ์ Hand Over มากที่สุด

ผลกระทบต่อการส่งข้อมูลพหุสื่อ

การส่งข้อมูลพหุสื่อจะใช้โปรโตคอล RTP ในการส่งข้อมูล ในกรณีที่ใช้โปรโตคอล SIP ในการทำ Hand Over นั้น เมื่อถึงขั้นตอนการส่งข้อมูลพหุสื่อ การทำงานของโปรโตคอลจะไม่มีผลใดๆ ต่อการส่งข้อมูลพหุสื่อ แต่ในกรณีการใช้ MIPv6 ในการทำ Hand Over นั้นจะพบว่าข้อมูลทุกแพ็กเกตจะถูกทำ Tunneling ด้วยการใช้ Destination Option Header เพื่อบอกค่าหมายเลขไอพีที่ถูกต้องเพิ่มเติมต่อจาก IPv6 Header ทำให้เกิด Over Head บนแพ็กเกตของไอพีรุ่นที่ 6 ดังรูปที่ 45



รูปที่ 45 แสดง Destination Option Header ใน IPv6 Packet จับข้อมูลด้วยโปรแกรม Ethereal

ดังนั้นจึงได้ทำการทดสอบหาค่า RTT (Round Trip Time) โดยทำการตรวจจับค่า RTT เมื่อทำการส่งข้อมูลพหุสื่อระหว่างเครื่อง client a และ client b เพื่อศึกษาผลกระทบต่อการส่งข้อมูลเมื่อใช้ MIPv6 ได้ผลดังนี้

ตารางที่ 12 แสดงค่า RTT ในการส่งข้อมูลพหุสื่อระหว่างเมื่อไม่ใช้ MIPv6 กับเมื่อใช้ MIPv6

□□□ □□□	เวลา (ms.)	
	RTT	RTT-MIPv6
1	120	132
2	117	139
3	113	124
4	116	133
5	101	128
6	104	125
7	116	138
8	114	136
9	119	122
10	102	127
average	112.2	130.4

จากตารางที่ 12 แสดงให้เห็นว่าการส่งข้อมูลเมื่อใช้ MIPv6 ค่า RTT จะมีค่าสูงกว่าปกติประมาณ 18.2 ms นั่นหมายความว่าข้อมูลเสียงและภาพ เมื่อทำการส่งข้อมูลขณะใช้ MIPv6 จะได้รับช้ากว่าปกติ

ผลกระทบต่อช่องสัญญาณแบบ Connection Oriented

การใช้ช่องสัญญาณแบบ Connection Oriented เช่น การใช้โปรโตคอล TCP ในการติดต่อ โดยปกติหากเกิดการ Hand Over ขึ้น หรือหมายเลขไอพีที่ใช้งานอยู่เปลี่ยนแปลงไป จะเป็นผลให้ขาดการเชื่อมต่อทันที แต่หากใช้ MIPv6 จะสามารถทำให้ไม่ขาดการเชื่อมต่อได้ ด้วยอัลกอริทึมการทำงานในระดับ Network

ผลกระทบต่อการให้บริการโมไบลิตี้ประเภทอื่นๆ

บริการโมไบลิตี้ประเภทอื่นๆ ได้แก่ การทำ Transfer Call, Forward Call และ Call Pickup เมื่อเกิด Hand Over ขึ้น หากใช้โปรโตคอล SIP ในการทำ Hand Over จะพบว่า Media Session และค่าหมายเลขไอพีจะต้องถูก Update ใหม่ทั้งหมด นอกจากนี้ในการทำการติดต่อแบบ Multiple Connection เช่น บริการถ่ายทอดสด, บริการการประชุม เป็นต้น หากไม่ใช่ MIPv6 เมื่อ Mobile Node ทำการเคลื่อนที่เปลี่ยนระบบเครือข่ายจะต้องทำการส่งข้อมูลหมายเลขไอพีเพื่อ Update ข้อมูลไปยังทุกๆโหนดที่ทำการติดต่อทั้งหมด

ในขณะที่เมื่อใช้ MIPv6 ในส่วน Media Session ยังคงเป็น Session เดิมเสมอ และในกรณีติดต่อแบบ Multiple Connection โหนดอื่นๆที่กำลังติดต่ออยู่จะไม่ทราบว่าเกิดอะไรขึ้นกับ Mobile Node อีกทั้งสามารถรับส่งข้อมูลได้ต่อเนื่อง

ผลกระทบต่อความเป็นมาตรฐานของโปรโตคอล SIP

หากใช้โปรแกรม SIP Client ที่พัฒนาขึ้นซึ่งมีส่วนการจัดการเมื่อเกิด Hand Over ขึ้นใช้งานติดต่อกับโปรแกรม SIP Client ขององค์กรอื่นๆ เช่น Hotsip (www.hotsip.com), SIP User Agent (www.ubiguity.net) เป็นต้น ซึ่งในขณะนี้ยังไม่มีการพัฒนารองรับการทำ Hand Over ดังกล่าว ดังนั้นโปรแกรม SIP Client ที่พัฒนาขึ้นจึงไม่สามารถใช้ความสามารถในการทำ Hand Over ได้เมื่อใช้งานร่วมกับโปรแกรม SIP Client ขององค์กรอื่นๆ

แต่หากใช้ MIPv6 จะทำให้ยังคงสามารถติดต่อสื่อสารกับโปรแกรมที่เป็นไปตามมาตรฐาน SIP อื่นๆได้ เนื่องจากมีการทำงานอยู่ในระดับ Network เท่านั้น

ผลกระทบเมื่อเกิดการ Hand Over ไปยังเครือข่ายถัดไป

ในกรณีที่ใช้โปรโตคอล SIP ในการทำ Hand Over ทุกๆครั้งที่เกิดการ Hand Over ขึ้นจะทำการ Update ค่าหมายเลขไอพีไปยังทุกๆโหนดที่ทำการติดต่อหรือเกี่ยวข้อง ทำให้เกิดการสร้าง Media Session ขึ้นใหม่ทุกครั้ง และทุกโหนดรับรู้การเปลี่ยนแปลงของหมายเลขไอพีของ Mobile Node

ในกรณีที่ใช้ MIPv6 การเปลี่ยนแปลงการเชื่อมต่อหรือหมายเลขไอพีจะมีการทำงานในระดับ Network โดยทุกๆโหนดที่ทำการเชื่อมต่อไม่สามารถรับรู้ได้ว่าเกิดอะไรขึ้นกับ Mobile Node ค่าหมายเลขไอพีและการเชื่อมต่อยังคงเป็น Session เดิม

แนวโน้มในการติดต่อระหว่างระบบเครือข่ายไอพีรุ่นที่ 4 กับไอพีรุ่นที่ 6

ไม่ว่าจะใช้อัลกอริทึมในการทำ Hand Over แบบใด การเชื่อมต่อระหว่างโหนดบนระบบเครือข่ายที่ใช้ไอพีรุ่นที่ 4 กับโหนดบนระบบเครือข่ายไอพีรุ่นที่ 6 จำเป็นต้องใช้อัลกอริทึมการทำงานอื่นช่วย โดยเรียกว่า การทำ IPv4-IPv6 Transition ซึ่งเป็นอัลกอริทึมในการนำข้อมูลไอพีรุ่น 4 มาใช้ภายในการเชื่อมต่อแบบไอพีรุ่น 6 หรือการนำข้อมูลไอพีรุ่น 6 มาใช้ภายในการเชื่อมต่อแบบไอพีรุ่น 4 ได้เป็นต้น

5.5 สรุป

สรุปข้อเปรียบเทียบระหว่างการใช้ MIPv6 กับการใช้โปรโตคอล SIP ในการทำ Mobility กรณีเกิด Hand Over ได้ดังตารางต่อไปนี้

ตารางที่ 13 เปรียบเทียบการทำ Hand Over ด้วยโปรโตคอล SIP และ MIPv6

MIPv6	SIP
- เมื่อเกิด Hand Over ไม่ส่งผลกระทบต่อการทำงานในระดับชั้นเหนือจากระดับชั้น Network	- เมื่อเกิด Hand Over จะส่งผลกระทบต่อการทำงานในระดับชั้นเหนือระดับชั้น Network
- ไม่ต้องสร้างช่องสัญญาณสำหรับการส่งข้อมูลพหุสื่อและสัญญาณ SIP ใหม่	- จำเป็นต้องสร้างช่องสัญญาณสำหรับการส่งข้อมูลพหุสื่อและสัญญาณ SIP ใหม่ ได้แก่ re-binding IP Address และ re-create socket

MIPv6	SIP
- มี Over Head เพิ่มขึ้นในทุกๆข้อมูลพหุสื่อทำให้เกิดค่าเวลาหน่วงเพิ่มมากขึ้น	- ไม่มี Over Head เพิ่มขึ้น
- ทุกโหนดในระบบต้องสนับสนุนการทำงานของ Mobile IP	- โหนด SIP Client จำเป็นต้องมีอัลกอริทึมการทำ Mobility เมื่อเกิด Hand Over
- โหนดในระบบ SIP/MIPv6 จำเป็นต้องรองรับการทำงานบนไอพีรุ่นที่ 6	- โหนดในระบบไม่จำเป็นต้องรองรับการทำงานบนไอพีรุ่นที่ 6 เนื่องจากสามารถทำงานได้ทั้งไอพีรุ่นที่ 6และไอพีรุ่นที่4
- สามารถใช้งานกับโหนดบนระบบเครือข่ายไอพีรุ่นที่ 4 ได้ แต่เรียกร้องการทำ IPv4-IPv6 Transition	- สามารถใช้งานกับโหนดบนระบบเครือข่ายไอพีรุ่นที่ 4 หรือไอพีรุ่นที่ 6 ได้ แต่เรียกร้องการทำ IPv4-IPv6 Transition

ดังนั้นทั้งโปรโตคอล SIP และ MIPv6 ต่างสามารถทำโมไบลิตี้ได้ด้วยลำดับสัญญาณและความสามารถในการทำงานต่างกัน โดยโปรโตคอล SIP มีขอบเขตความสามารถจำกัดในการทำโมไบลิตี้ได้เฉพาะโปรแกรมประยุกต์ซึ่งโปรโตคอล SIP รับผิดชอบเท่านั้น ในขณะที่ MIPv6 สามารถรับผิดชอบการรักษาการเชื่อมต่อของโปรแกรมประยุกต์ทั้งหมดที่มีการทำงานอยู่บนระบบปฏิบัติการเดียวกัน แต่เรียกร้องขั้นตอนการทำงานในระดับชั้น Network เพิ่มเติมในทุกๆเครื่องในระบบเครือข่ายที่มีการเชื่อมต่อในขณะนั้น และเมื่อพิจารณาในประเด็นของค่าเวลาหน่วงที่เกิดขึ้นจากทั้ง 2 โปรโตคอล พบว่า การใช้ MIPv6 ทำให้เกิดค่าเวลาหน่วงมากกว่าการใช้โปรโตคอล SIP แม้ว่าค่าเวลาหน่วงดังกล่าวไม่ได้เกิดจากการทำงานของ MIPv6 โดยตรง แต่เป็นผลมาจากขั้นตอนการทำ Duplicated Address Detection ของโปรโตคอลไอพีรุ่นที่ 6 ซึ่งเกิดขึ้นที่ Router ซึ่งทำหน้าที่เป็น Home Agent ในระบบเครือข่ายนั้น จึงเป็นผลให้การใช้โปรโตคอล SIP เกิดความรวดเร็วกว่าการใช้ MIPv6 ซึ่งสำคัญต่อโปรแกรมประยุกต์ในระบบไอพีเทเลโฟนนี่ แต่อย่างไรก็ตามความสามารถของ MIPv6 ในการรับผิดชอบการรักษาการเชื่อมต่อของโปรแกรมประยุกต์ทั่วไปยังคงมีความสำคัญ ด้วยความสามารถของทั้ง 2 โปรโตคอลจึงจำเป็นต้องมีการประยุกต์และใช้งานร่วมกัน ซึ่งจะอธิบายอัลกอริทึมการใช้งานทั้ง 2 โปรโตคอลร่วมกันในหัวข้อถัดไป