กลไกสร้างความมั่นคงสำหรับสัญญาณเชื่อมต่อของ SIP

Security Mechanism for Session Initiation Protocol (SIP) Signaling

พรชนก รอดนิกร

Pornchanok Rodnikorn

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
มหาวิทยาลัยสงขลานครินทร์

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science
Prince of Songkla University
2556

ชื่อวิทยานิพนธ์ กลไกสร้างความมั่นคงสำหรับสัญญาณเชื่อมต่อของ SIP
ผู้เขียน นางสาวพรชนก รอดนิกร
สาขาวิชา วิทยาการคอมพิวเตอร์

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

…….....
(ผู้ช่วยศาสตราจารย์ ดร.นิษฐิดา เอลซ์)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

…….....
(ผู้ช่วยศาสตราจารย์ ดร.ลัดดา ปรีชาวีรกุล)

คณะกรรมการสอบ

…..…...……..…ประธานกรรมการ
(ดร.ชัชนันท์ จันแดง)

…….....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.นิษฐิดา เอลซ์)

…….....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.ลัดดา ปรีชาวีรกุล)

…….....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.ศิริรัตน์ วณิชโยบล)

บัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์ อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

.....
(รองศาสตราจารย์ ดร.ธีระพล ศรีชนะ)
คณบดีบัณฑิตวิทยาลัย

ขอรับรองว่า ผลงานวิจัยนี้มาจากการศึกษาวิจัยของนักศึกษาเอง และได้แสดงความขอบคุณบุคคลที่มีส่วนช่วยเหลือแล้ว

ลงชื่อ.....
(ผู้ช่วยศาสตราจารย์ ดร.นิษฐิดา เอลซ์)
อาจารย์ที่ปรึกษาวิทยานิพนธ์

ลงชื่อ.....
(นางสาวพรชนก รอดนิกร)
นักศึกษา

ข้าพเจ้าขอรับรองว่า ผลงานวิจัยนี้ไม่เคยเป็นส่วนหนึ่งในการอนุมัติปริญญาในระดับใดมาก่อน และไม่ได้ถูกใช้ในการยื่นขออนุมัติปริญญาในขณะนี้

ลงชื่อ.....
(นางสาวพรชนก รอดนิกร)
นักศึกษา

ชื่อวิทยานิพนธ์ กลไกสร้างความมั่นคงสำหรับสัญญาณเชื่อมต่อของ SIP
ผู้เขียน นางสาวพรชนก รอดนิกร
สาขาวิชา วิทยาการคอมพิวเตอร์
ปีการศึกษา 2555

บทคัดย่อ

Voice over Internet Protocol (VoIP) ได้รับความนิยมในการนำมาใช้ในเครือข่ายอย่างกว้างขวาง โดยทั่วไป VoIP ใช้แอปพลิเคชันโพรโทคอลหลายโพรโทคอล เช่น H.323, Media Gateway Control Protocol (MGCP) และ Session Initiation Protocol (SIP) โดยที่ SIP ถูกนำมาใช้เป็นโพรโทคอลส่งสัญญาณเชื่อมต่อสำหรับ VoIP มากที่สุด อย่างไรก็ตาม SIP มีช่องโหว่ต่อการถูกโจมตีสัญญาณเชื่อมต่อ (Signaling Attack) โดยเฉพาะจุดอ่อนของกระบวนการพิสูจน์ตัวตนและข้อความที่ไม่ได้เข้ารหัส ซึ่งการโจมตีนี้อาจส่งผลกระทบต่อระบบการคิดค่าบริการของบริการ VoIP วิทยานิพนธ์นี้จึงเสนอ SIP Extension for Signaling Attacks Protection (SIPE-SAP) ซึ่งเป็นกลไกสำหรับป้องกันการโจมตีสัญญาณเชื่อมต่อของ VoIP จากการใช้ SIP โดยเพิ่มเฮดเดอร์ฟิลด์ของ SIP และวิธีการสำหรับป้องกันเฮดเดอร์ฟิลด์บางฟิลด์ที่สำคัญ เช่น Proxy-Authorization, To และ Call-ID การทำงานของกลไกที่นำเสนอช่วยป้องกันการปลอมแปลงและการแก้ไขข้อความ SIP ได้ ผลการทดลองปรากฏว่าระยะเวลาที่ใช้ในการลงทะเบียนและการเชื่อมต่อการโทรเฉลี่ยของ SIPE-SAP เพิ่มขึ้นคิดเป็น 3.3 และ 5.53 เท่าตามลำดับ เมื่อเปรียบเทียบกับการใช้ SIP ร่วมกับ HTTP Digest และใช้เวลาลดลงคิดเป็น 9.28% และ 12.96% ตามลำดับ เมื่อเปรียบเทียบกับการใช้ SIP ร่วมกับ Transport Layer Security (TLS)

Thesis Title Security Mechanism for Session Initiation Protocol (SIP) Signaling

Author Miss Pornchanok Rodnikorn

Major Program Computer Science

Academic Year 2012

ABSTRACT

Voice over Internet Protocol (VoIP) has become popular as the Internet has been widely adopted. Basically, VoIP relies on several other application protocols such as H.323, Media Gateway Control Protocol (MGCP) and Session Initiation Protocol (SIP). Among these protocols, SIP is mostly used for VoIP signaling protocol. However, SIP is known to be vulnerable to signaling attacks due to a weakness of authentication method and unencrypted messages. These attacks may affect the billing system of VoIP service. In this thesis, we proposed SIP Extension for Signaling Attacks Protection (SIPE-SAP), a lightweight mechanism to protect SIP-based VoIP from signaling attacks by introducing an additional header field and a method to protect some important header fields such as Proxy-Authorization, To and Call-ID which are targeted to the attacks. The proposed mechanism protects against SIP messages forgery and modifying SIP messages. The experimental results show that the average registration response time and call setup response time of SIPE-SAP increases about 3.3 and 5.53 times compared to SIP using HTTP Digest respectively. In addition, the SIPE-SAP overhead decreases about 9.28% and 12.96% compared to SIP with Transport Layer Security (TLS) respectively.

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

รายการตาราง

ตาราง **หน้า**

รายการภาพประกอบ

ภาพประกอบ **หน้า**

รายการภาพประกอบ (ต่อ)

ภาพประกอบ **หน้า**

รายการภาพประกอบ (ต่อ)

ภาพประกอบ **หน้า**

รายการภาพประกอบ (ต่อ)

ภาพประกอบ **หน้า**

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของการวิจัย

Voice over Internet Protocol (VoIP) คือเทคโนโลยีสำหรับการสื่อสารเสียงผ่านเครือข่ายที่ใช้อินเทอร์เน็ตโพรโทคอล ที่มีค่าบริการค่อนข้างถูกกว่าเครือข่ายโทรศัพท์พื้นฐาน ส่งผลให้ปัจจุบัน VoIP ได้รับความนิยมมากขึ้น แต่การนำเครือข่ายเสียงมาทำงานบนเครือข่ายอินเทอร์เน็ตโดยไม่มีกลไกการป้องกันความปลอดภัยที่เหมาะสมอาจก่อให้เกิดปัญหาความปลอดภัยได้ เช่น การเข้าถึงระบบ VoIP โดยไม่ได้รับอนุญาตในออสเตรเลีย เมื่อปี 2009 (Tindal, 2009) ทำให้ผู้บุกรุกสามารถโทรออกได้ถึง 11,000 ครั้ง คิดเป็นเงิน 120,000 เหรียญสหรัฐ

การเชื่อมต่อเพื่อสื่อสารเสียง วิดีโอ หรือข้อความในเทคโนโลยี VoIP จะต้องมีการส่งสัญญาณเชื่อมต่อ (Signaling) ก่อน โดยการใช้โพรโทคอลสำหรับการเชื่อมต่อ และโดยทั่วไปนิยมใช้ Session Initiation Protocol (SIP) สำหรับสร้าง แก้ไข และยกเลิกการเชื่อมต่อ SIP ให้บริการในรูปแบบข้อความตัวอักษรที่ไม่มีการเข้ารหัสหรือตรวจสอบการแก้ไขข้อความจากผู้ไม่ประสงค์ดี นอกจากนี้ กระบวนการพิสูจน์ตัวตนของ SIP สามารถใช้กับการส่งข้อความ SIP บางข้อความเท่านั้น ทำให้เกิดปัญหาความมั่นคงปลอดภัยตามมา ผู้บุกรุกอาจมีการแก้ไขหรือปลอมแปลงข้อความ SIP ขึ้นมาใหม่เพื่อใช้โจมตีการทำงานเกี่ยวกับการสร้าง แก้ไข และยกเลิกการเชื่อมต่อ ซึ่งเรียกว่าการโจมตีสัญญาณเชื่อมต่อ (Signaling Attack) (Geneiatakis และคณะ, 2006) ส่งผลให้เกิดการปฏิเสธการให้บริการไปยังผู้ใช้บริการรายอื่น หรือมีการเข้าถึงบริการโดยไม่ได้รับอนุญาตได้ ตัวอย่างการโจมตีไปยังผู้ใช้บริการของผู้ให้บริการรายใหญ่ เช่น Vonage และ AT&T (Zhang, 2010) คือ การดักจับข้อความที่ใช้สร้างการเชื่อมต่อที่มีข้อมูลประจำตัวผู้ใช้ที่แท้จริงอยู่ แล้วแก้ไขหมายเลขไอพีและหมายเลขพอร์ตให้เป็นของผู้บุกรุกก่อนส่งไปยังผู้ให้บริการ ทำให้ผู้บุกรุกสามารถสร้างการเชื่อมต่อได้โดยไม่ต้องเสียค่าบริการ

แนวทางในการแก้ไขปัญหาเหล่านี้อาจใช้วิธีการเข้ารหัสลับข้อความ SIP เช่น ใช้ Transport Layer Security (TLS) แต่การเข้ารหัสข้อความ SIP ทั้งข้อความอาจเสียเวลาในการเข้ารหัสและถอดรหัส และเซิร์ฟเวอร์ต้องทำงานหนักเพื่อดูแลการเชื่อมต่อ TLS จากผู้ใช้งานหลายคนพร้อมๆ กัน นอกจากนี้อาจใช้ Secure/Multipurpose Internet Mail Extensions

(S/MIME) เพื่อเข้ารหัสข้อความหรือรับประกันความถูกต้องสมบูรณ์ของข้อความบางส่วนได้ แต่การใช้ S/MIME เพื่อเข้ารหัสข้อความ SIP ทั้งข้อความ แล้วนำข้อความที่ถูกเข้ารหัสนั้นมาใส่ต่อท้ายเข้าไปในข้อความ SIP เดิมที่ไม่ถูกเข้ารหัส จะทำให้ข้อความทั้งหมดมีขนาดใหญ่มาก (Rosenberg และคณะ, 2002) และเนื่องจากผู้ใช้งานสามารถใช้ใบรับรองดิจิทัลที่ไม่ได้รับมาจากผู้ให้บริการใบรับรอง ส่งผลให้ผู้ที่ต้องการติดต่อสื่อสารด้วยไม่สามารถตรวจสอบความถูกต้องของใบรับรองได้ ผู้บุกรุกอาจอาศัยจุดอ่อนนี้ในการเข้าแทรกกลางการเปลี่ยนแปลงกุญแจระหว่างกันได้

นอกจากนี้ งานวิจัยส่วนใหญ่ที่เกี่ยวข้องกับการแก้ไขปัญหาการโจมตีสัญญาณเชื่อมต่อ สามารถแก้ไขได้บางปัญหาเท่านั้น เช่น การส่งข้อความ BYE เพื่อยกเลิกการเชื่อมต่อของผู้ใช้ที่แท้จริง หรือการส่งข้อความ Re-INVITE เพื่อเปลี่ยนแปลงข้อมูลการเชื่อมต่อ เช่น หมายเลขไอพีที่ใช้เชื่อมต่อ มีเพียงงานวิจัยที่ใช้วิธีการตรวจสอบความถูกต้องสมบูรณ์ของทุกๆ ข้อความ (Geneiatakis และ Lambrinoudakis, 2008) และการประยุกต์ใช้ TLS เพื่อป้องกันการโกงค่าบริการ (Shekokar และ Devane, 2012) เท่านั้นที่ค่อนข้างครอบคลุมการโจมตีหลายรูปแบบ แต่การตรวจสอบความถูกต้องสมบูรณ์ของทุกๆ ข้อความนี้ใช้วิธีการนำรหัสผ่านและข้อความ SIP มาผ่านฟังก์ชันแฮช ซึ่งอาจเกิดการโจมตีด้วยการเดารหัสผ่านได้ และเซิร์ฟเวอร์ต้องรู้รหัสผ่านของผู้ใช้งานทั้ง 2 ฝ่าย วิธีการนี้จึงใช้ได้ดีเมื่อผู้ใช้บริการอยู่ในโดเมนเดียวกันเท่านั้น ส่วนการใช้ TLS ก็มีข้อจำกัดดังที่ได้กล่าวมาแล้ว

จากการพิจารณาการทำงานของ SIP พบว่าข้อความ SIP ที่ส่งผ่านเซิร์ฟเวอร์นอกจากจะถูกเซิร์ฟเวอร์อ่านเพื่อพิจารณาว่าเป็นข้อความประเภทใดและส่งไปหาใครแล้ว เซิร์ฟเวอร์อาจมีการเพิ่มหมายเลขไอพีเข้าไป เพื่อระบุให้มีการส่งข้อความผ่านทางเซิร์ฟเวอร์ ด้วยเหตุนี้ SIP จึงไม่มีกลไกสำหรับตรวจสอบว่าข้อความที่ได้รับมานั้นถูกแก้ไขหรือไม่ ผู้บุกรุกมักอาศัยจุดอ่อนนี้เพื่อแก้ไขข้อความ SIP และข้อมูลรายละเอียดของการเชื่อมต่อ โดยเฉพาะอย่างยิ่งหมายเลขไอพีของผู้สร้างการเชื่อมต่อ ทำให้สามารถเข้าถึงบริการโดยไม่ได้รับอนุญาต จากปัญหาดังกล่าว หากสามารถวิเคราะห์ส่วนประกอบของข้อความ SIP ได้ว่าส่วนใดอาจถูกแก้ไขโดยผู้บุกรุกหรือมีการใช้เพื่อปลอมแปลงข้อความ SIP ขึ้นมาใหม่ ก็จะเป็นแนวทางหนึ่งที่สามารถป้องกันการโจมตีสัญญาณเชื่อมต่อได้

ดังนั้น งานวิจัยนี้ผู้วิจัยได้ทำการวิเคราะห์ส่วนประกอบของข้อความ SIP ที่อาจถูกผู้บุกรุกใช้เพื่อการโจมตี และสร้างกลไกสำหรับป้องกันการโจมตีสัญญาณเชื่อมต่อที่ทำให้ระบบยังคงทำงานได้อย่างมีประสิทธิภาพ ทั้งในแง่ของการโจมตีเพื่อเข้าถึงบริการโดยไม่ได้รับอนุญาต และการปลอมแปลงข้อความ SIP เพื่อโจมตีให้เกิดการปฏิเสธการให้บริการไปยังผู้ใช้บริการรายอื่น

1.2 วัตถุประสงค์ของการวิจัย

1) วิเคราะห์และออกแบบกลไกสร้างความมั่นคงสำหรับสัญญาณเชื่อมต่อของ SIP เพื่อป้องกันการโจมตีในรูปแบบดังต่อไปนี้

- การเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชันโดยผู้ที่ไม่มีสิทธิ์
- การโจมตีการคิดค่าบริการที่อาศัยจุดอ่อนของการส่งสัญญาณเชื่อมต่อด้วย SIP

2) พัฒนาระบบเพื่อป้องกันการโจมตีตามข้อ 1 โดยใช้กลไกที่ได้ออกแบบไว้

1.3 ขอบเขตของการวิจัย

1) วิเคราะห์และออกแบบกลไกสร้างความมั่นคงสำหรับสัญญาณเชื่อมต่อของ SIP ที่สามารถป้องกันการโจมตีแบบ

- Call Establishment Hijacking
- การส่งข้อความ Re-INVITE
- การส่งข้อความ UPDATE
- Broken Handshaking

2) พัฒนาระบบเพื่อป้องกันการโจมตีตามข้อ 1 โดยใช้กลไกที่ได้ออกแบบไว้

3) ทดสอบระบบ

1.4 นิยามศัพท์เฉพาะ

1) วิทยาการเข้ารหัสลับแบบกุญแจสมมาตร คือการใช้กุญแจตัวเดียวกันในการเข้ารหัสและถอดรหัส

2) วิทยาการเข้ารหัสลับแบบกุญแจอสมมาตร คือการใช้กุญแจสาธารณะสำหรับเข้ารหัส และใช้กุญแจส่วนตัวเพื่อถอดรหัส

3) กุญแจเซสชัน คือกุญแจที่กำหนดไว้ใช้ชั่วคราวสำหรับการเชื่อมต่อหนึ่งๆ โดยในงานวิจัยนี้ มีการสร้างกุญแจเซสชันในขั้นตอนการพิสูจน์ตัวตนระหว่างผู้ใช้และเซิร์ฟเวอร์ และในขั้นตอนการเชื่อมต่อระหว่างผู้ใช้

1.5 ขั้นตอนและระยะเวลาการดำเนินการ

1.5.1 ขั้นตอนการดำเนินการ

1) ศึกษางานวิจัยและเอกสารที่เกี่ยวข้อง
2) ศึกษาเทคโนโลยีและเครื่องมือที่สนับสนุนในการพัฒนาระบบ
3) วิเคราะห์และออกแบบสถาปัตยกรรมระบบ
4) พัฒนาระบบ
5) ติดตั้งและทดสอบระบบ
6) เขียนบทความวิจัยและเผยแพร่
7) จัดทำเอกสารวิทยานิพนธ์

1.5.2 ระยะเวลาการดำเนินการ

เดือนมิถุนายน พ.ศ. 2552 – เดือนพฤษภาคม พ.ศ. 2556 การวางแผนการดำเนินการเบื้องต้น แสดงดังตารางที่ 1.1

ตารางที่ 1.1 แผนการดำเนินการ

ขั้นตอน	พ.ศ./เดือน							
	2552	2553		2554		2555		2556
	6-12	1-6	6-12	1-6	6-12	1-6	7-8	1-5
1)	■							
2)		■						
3)			■	■				
4)				■	■			
5)					■	■		
6)				■	■	■		
7)	■	■	■	■	■	■	■	■

1.6 สถานที่ดำเนินการวิจัย

ห้องปฏิบัติการคอมพิวเตอร์ CS 209 ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่

1.7 เครื่องมือที่ใช้ในการดำเนินการ

1.7.1 ฮาร์ดแวร์ (Hardware)

1) เครื่องคอมพิวเตอร์ส่วนบุคคล หน่วยประมวลผล Intel Core 2 Quad 2.40 GHz หน่วยความจำ 2 GB จำนวน 2 เครื่อง

2) เครื่องคอมพิวเตอร์ส่วนบุคคล หน่วยประมวลผล Intel Core 2 Duo 1.86 GHz หน่วยความจำ 2 GB จำนวน 1 เครื่อง

3) เครื่องคอมพิวเตอร์ส่วนบุคคล หน่วยประมวลผล Intel Core 2 Duo 3.00 GHz หน่วยความจำ 2 GB จำนวน 1 เครื่อง

4) เครื่องคอมพิวเตอร์ส่วนบุคคล หน่วยประมวลผล Intel Core 2 Duo 2.40 GHz หน่วยความจำ 1 GB จำนวน 2 เครื่อง

1.7.2 ซอฟต์แวร์ (Software)

1) ระบบปฏิบัติการ FreeBSD 7.4

2) ระบบปฏิบัติการ Ubuntu 10.04

3) โปรแกรม OpenSIPS

4) โปรแกรม Linphone

5) โปรแกรม SIPp

6) โปรแกรม Wireshark

7) โปรแกรม Ettercap

8) ภาษาซี

1.8 ประโยชน์ที่คาดว่าจะได้รับ

1) กระบวนการส่งสัญญาณเชื่อมต่อมีความปลอดภัยมากขึ้น ทำให้การคิดค่าบริการมีความถูกต้องและน่าเชื่อถือ และช่วยลดการโจมตีที่ก่อให้เกิดการปฏิเสธการให้บริการ

2) ช่วยลดการสูญเสียรายได้และทรัพยากรให้กับผู้ให้บริการ VoIP

3) ผู้ใช้งานมีความมั่นใจในการใช้งานระบบมากขึ้น

เนื้อหาในรายงานวิทยานิพนธ์เล่มนี้ประกอบด้วย 5 บท โดยบทที่ 2 กล่าวถึงทฤษฎีเกี่ยวกับเทคโนโลยี VoIP และโพรโทคอล SIP ตลอดจนงานวิจัยที่เกี่ยวข้อง บทที่ 3 กล่าวถึงการวิเคราะห์และออกแบบกลไกสำหรับป้องกัน VoIP จากการใช้ SIP บทที่ 4 กล่าวถึงการพัฒนาระบบ การทดสอบประสิทธิภาพของกลไกที่ได้นำเสนอ รวมถึงการทดสอบกลไกป้องกันการบุกรุก และบทที่ 5 เป็นบทสรุป ปัญหาและข้อเสนอแนะของการทำวิทยานิพนธ์ชุดนี้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 บทนำ

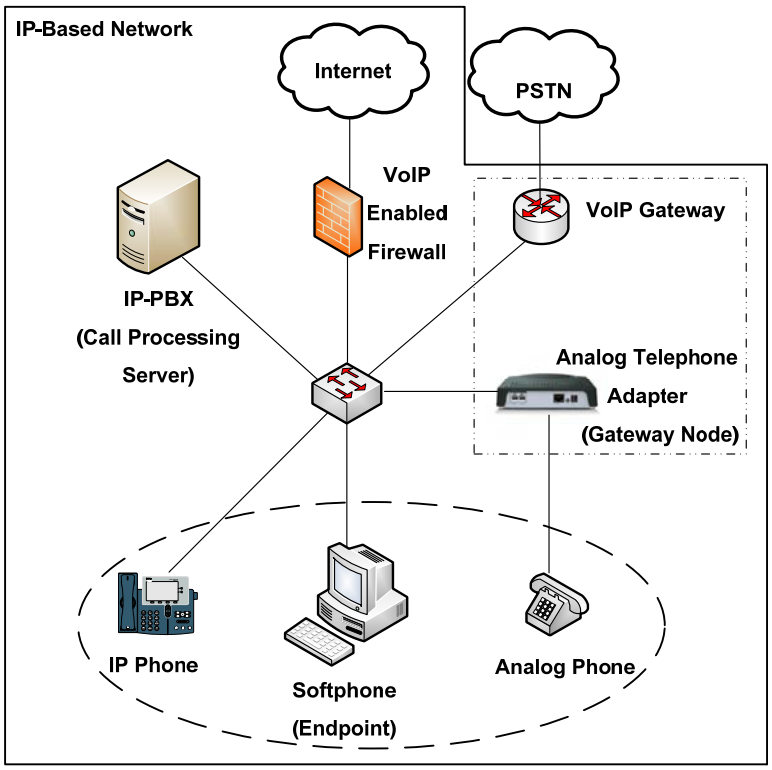
เนื้อหาในบทนี้จะมีการอธิบายเทคโนโลยี Voice over Internet Protocol (VoIP) โดยสรุป อธิบายรายละเอียดของ Session Initiation Protocol (SIP) โดยเริ่มจากส่วนประกอบของ SIP การกำหนดหมายเลข SIP ประเภทของข้อความ SIP และรายละเอียดต่างๆ ของข้อความ กระบวนการในการลงทะเบียนและการสร้างเซสชันโดยใช้ SIP ต่อมากล่าวถึงกลไกการรักษาความมั่นคงปลอดภัยที่สามารถนำมาใช้ป้องกัน VoIP จากการใช้ SIP ได้ รวมถึงวิธีการโจมตีสัญญาณเชื่อมต่อของ SIP (Signaling Attack) รูปแบบต่างๆ สุดท้ายกล่าวถึงงานวิจัยที่เกี่ยวข้องกับการโจมตีสัญญาณเชื่อมต่อของ SIP ดังรายละเอียดต่อไปนี้

2.2 เทคโนโลยี Voice over Internet Protocol (VoIP)

VoIP คือเทคโนโลยีสำหรับการสื่อสารเสียงผ่านเครือข่ายอินเทอร์เน็ต มีการแปลงสัญญาณเสียงซึ่งอยู่ในรูปของสัญญาณแอนะล็อกเป็นสัญญาณดิจิทัล แล้วสร้างเป็นกลุ่มของแพ็กเก็ตส่งไปบนระบบเครือข่ายอินเทอร์เน็ตโดยใช้เส้นทางที่ต่างกันได้ (IP-Based Packet-Switched Networks) ซึ่งมีเร้าเตอร์ (Router) เป็นตัวรับและค้นหาเส้นทางให้กับแพ็กเก็ตเหล่านี้ แทนการใช้สายสัญญาณโทรศัพท์ที่จะต้องมีการจองเส้นทางในการสื่อสาร (Circuit-Switched Networks) ทำให้ใช้งานช่องสัญญาณได้อย่างมีประสิทธิภาพ สามารถรองรับผู้ใช้งานได้มาก การรวมเครือข่ายข้อมูลและเครือข่ายเสียงเป็นเครือข่ายเดียวกัน ช่วยลดต้นทุนในการดูแลและการจัดการ มีความยืดหยุ่น คือสามารถเพิ่ม แก้ไข หรือลบโหนด (เช่น โทรศัพท์ เครื่องเซิร์ฟเวอร์) บนเครือข่ายได้ง่าย นอกจากนี้ VoIP ยังมีคุณลักษณะที่ดีอีกหลายอย่าง เช่น ค่าบริการโทรศัพท์ทางไกลต่ำกว่าเครือข่ายโทรศัพท์พื้นฐาน (Public-Switched Telephone Network: PSTN) การเข้ารหัสช่วยให้ข้อมูลที่ส่งระหว่างกัน ทั้งเสียงและข้อความตัวอักษรเป็นความลับ การรวมข้อมูลการติดต่อสื่อสารเป็นหน่วยเดียวกัน (Unified Messaging) กล่าวคือมีการเก็บรวบรวมข้อความประเภทต่างๆ เข้าไว้ด้วยกัน เช่น ข้อความเสียง อีเมล และแฟกซ์ ทำ

ให้ผู้ใช้สะดวกต่อการเข้าถึงข้อมูล และทำให้ผู้ใช้สามารถรับข้อมูลได้หลายรูปแบบ เช่น การได้รับข้อความเสียงในรูปแบบอีเมล

เครือข่าย VoIP สามารถทำงานพื้นฐานได้เหมือนกับเครือข่ายโทรศัพท์พื้นฐาน คือ อุปกรณ์ต่างๆ ในเครือข่ายสามารถติดต่อสื่อสารกันได้โดยใช้วิธีส่งสัญญาณเชื่อมต่อ (Signaling) การส่งสัญญาณเชื่อมต่อในเครือข่าย VoIP เป็นการแลกเปลี่ยนข้อความระหว่างส่วนประกอบต่างๆ รูปแบบของข้อความจะเป็นไปตามโพรโทคอลที่ใช้ เครือข่าย VoIP มีบริการฐานข้อมูลที่ช่วยหาตำแหน่งที่ตั้งของโทรศัพท์โดยใช้หมายเลขไอพีและหมายเลขพอร์ตเป็นตัวระบุโทรศัพท์ การเชื่อมต่อโทรศัพท์ของเครือข่าย VoIP เป็นการรับส่งเสียงหรือวิดีโอแบบเรียลไทม์ทางช่องทางการสื่อสารที่ได้ตกลงกันไว้ และเมื่อการติดต่อสื่อสารเสร็จสิ้น ช่องทางการสื่อสารนี้จะถูกยกเลิก เครือข่าย VoIP ต้องสามารถแปลงข้อมูลที่อยู่ในรูปสัญญาณแอนะล็อกเป็นสัญญาณดิจิทัลได้ โดยกระบวนการที่ใช้แปลงสัญญาณเรียกว่า Coder-Decoder (CODEC) ซึ่งส่วนประกอบสำคัญของเครือข่าย VoIP มีวิธีการทำงานแตกต่างจากเครือข่ายโทรศัพท์พื้นฐาน โดยโครงสร้างพื้นฐานของเครือข่าย VoIP ประกอบด้วยส่วนประกอบสำคัญหลายส่วนคือ Endpoint, Call Processing Server, Gateway Node และ IP-Based Network ดังแสดงในภาพประกอบที่ 2.1 และรายละเอียดการทำงานของแต่ละส่วนประกอบนี้ดังคำอธิบาย



ภาพประกอบที่ 2.1 ตัวอย่างโครงสร้างเครือข่าย VoIP

- **Endpoint**

Endpoint คือเครื่องมือที่ใช้สำหรับการโทรศัพท์ อาจเป็นโทรศัพท์ที่ออกแบบมาเพื่อใช้กับเครือข่ายอินเทอร์เน็ตโดยเฉพาะหรือที่เรียกว่า IP Phone หรือเป็นซอฟต์แวร์จำลองโทรศัพท์ (Softphone) ที่ติดตั้งบนเครื่องคอมพิวเตอร์ เช่น X-Lite, Linphone เป็นต้น รวมถึงโทรศัพท์บ้านที่ต่อกับตัวแปลงสัญญาณจากแอนะล็อกให้เป็นดิจิทัลเพื่อให้สามารถเชื่อมต่อกับเครือข่ายอินเทอร์เน็ตได้ ซึ่งตัวแปลงนี้เรียกว่า Analog Telephone Adapter (ATA) นอกจากนี้โทรศัพท์เคลื่อนที่บางรุ่นก็สามารถติดตั้งซอฟต์แวร์ที่ใช้สื่อสารเสียงผ่านเครือข่ายได้

เมื่อเชื่อมต่อ Endpoint เข้าสู่ระบบ ต้องมีการลงทะเบียนกับเซิร์ฟเวอร์ที่ใช้จัดการการโทร คือ Call Processing Server เพื่อบันทึกตำแหน่งที่อยู่ซึ่งเป็นหมายเลขไอพีของ Endpoint ก่อน หากต้องการโทรออกหรือยกเลิกการโทรจะใช้วิธีการส่งสัญญาณเชื่อมต่อที่เป็นคำร้องขอ (Request) ไปยังเซิร์ฟเวอร์ เมื่อ Endpoint อีกฝั่งหนึ่งได้รับสัญญาณเชื่อมต่อจากเซิร์ฟเวอร์จะส่งคำตอบกลับ (Response) ผ่านเซิร์ฟเวอร์เช่นเดียวกัน หลังจากเชื่อมต่อการโทรสำเร็จ Endpoint ทั้งสองฝ่ายสามารถส่งเสียงหรือวิดีโอระหว่างกันได้โดยไม่ต้องผ่านเซิร์ฟเวอร์ที่ใช้จัดการการโทรอีก

- **Call Processing Server**

เป็นเซิร์ฟเวอร์ที่ใช้สำหรับจัดการการโทร อาจเรียกว่า Internet Protocol - Private Branch Exchange (IP-PBX) ทำหน้าที่ควบคุมและจัดการบริหารการเชื่อมต่อระหว่าง Endpoint ในเครือข่าย โดยรับสัญญาณเชื่อมต่อจาก Endpoint แล้วจัดหาเส้นทางเพื่อส่งสัญญาณเชื่อมต่อไปยังปลายทาง นอกจากนี้ IP-PBX อาจมีบริการรับฝากข้อความเสียง การประชุมสายโทรศัพท์ และมีระบบบันทึกข้อมูลการโทรเพื่อใช้คำนวณค่าบริการ

ในเครือข่ายโทรศัพท์พื้นฐานมีการต่อคู่สายไปยังตู้ชุมสายโทรศัพท์ แต่สำหรับเครือข่าย VoIP สามารถต่อ Endpoint เข้ากับเครือข่ายอินเทอร์เน็ตได้โดยมีการควบคุมการโทรระหว่างคู่สายโทรศัพท์โดยใช้ซอฟต์แวร์สั่งงานจากเซิร์ฟเวอร์ไปยังเครือข่ายอินเทอร์เน็ต ซอฟต์แวร์ที่นิยมนำมาใช้เป็น IP-PBX คือ Asterisk โพรโทคอลที่นิยมนำมาใช้ในระบบ IP-PBX เพื่อจัดการควบคุมการโทร คือ ชุดโพรโทคอล H.323 (ITU, 1998) และ Session Initiation Protocol (SIP) (Rosenberg และคณะ, 2002) รายละเอียดของโพรโทคอลจะกล่าวถึงในหัวข้อ 2.3

- **Gateway Node**

เมื่อมีการส่งสัญญาณเชื่อมต่อระหว่างเครือข่ายที่ใช้รูปแบบของสัญญาณเชื่อมต่อหรือโพรโทคอลที่แตกต่างกัน เช่น ระหว่างเครือข่ายอินเทอร์เน็ตและเครือข่ายโทรศัพท์พื้นฐาน จะต้องส่งผ่านเกตเวย์ก่อน หน้าที่หลักของเกตเวย์คือการแปลงเสียงซึ่งเป็นสัญญาณแอนะล็อกให้เป็นสัญญาณดิจิทัล แล้วบีบอัดและสร้างเป็นแพ็กเก็ตเพื่อส่งผ่านทางเครือข่าย

อินเทอร์เน็ต หรือรับแพ็กเก็ตจากเครือข่ายอินเทอร์เน็ตเพื่อสร้างสัญญาณแอนะล็อก นอกจากนี้สามารถใช้ในการกำหนดเส้นทาง การรวบรวมข้อมูลทางสถิติและการบริหารจัดการเครือข่ายได้อีกด้วย ตัวอย่างอุปกรณ์ที่ใช้เป็นเกตเวย์โดยเฉพาะคือ ATA

- **IP-Based Network**

เครือข่ายอินเทอร์เน็ตเป็นสื่อกลางการเชื่อมต่อระหว่างส่วนประกอบอื่นๆ เพื่อใช้ในการส่งแพ็กเก็ตที่เป็นสัญญาณเชื่อมต่อและแพ็กเก็ตเสียงระหว่างกัน สื่อกลางที่นำมาใช้ในเครือข่ายอินเทอร์เน็ตมีหลากหลาย เช่น อีเทอร์เน็ต (Ethernet) ไฟเบอร์ (Fiber) และเครือข่ายไร้สาย (Wireless)

จากรายละเอียดการทำงานพื้นฐานของแต่ละส่วนประกอบในเครือข่าย VoIP สามารถสรุปขั้นตอนการทำงานเมื่อมีการใช้บริการ VoIP ได้ดังนี้

1) ใช้ Endpoint ลงทะเบียนกับ IP-PBX เพื่อแจ้งตำแหน่งที่อยู่ที่สามารถติดต่อกลับได้

2) เมื่อต้องการโทรออก Endpoint จะส่งคำร้องขอไปยัง IP-PBX เพื่อให้จัดการส่งคำร้องขอต่อไปยังปลายทาง

3) การเชื่อมต่อถูกสร้างขึ้นเมื่อ Endpoint ที่อยู่ปลายทาง (ฝ่ายรับโทรศัพท์) ตอบกลับคำร้องขอ การติดต่อสื่อสารระหว่าง Endpoint และ IP-PBX นี้เรียกว่าการส่งสัญญาณเชื่อมต่อ

4) เสียงของคู่สนทนาจะถูกแปลงให้อยู่ในรูปสัญญาณดิจิทัล ถูกบีบอัด และอาจมีการเข้ารหัสเพื่อรักษาความลับของข้อมูลแล้วสร้างเป็นแพ็กเก็ตเสียง โดยที่ Endpoint สามารถส่งแพ็กเก็ตเสียงระหว่างกันได้โดยไม่ต้องผ่าน IP-PBX

5) เมื่อต้องการยกเลิกการโทรอาจส่งคำร้องขอไปยัง IP-PBX หรือ Endpoint ปลายทางก็ได้ขึ้นอยู่กับการตั้งค่าระบบ

6) ในกรณีที่ Endpoint ทั้งสองฝ่ายอยู่ในเครือข่ายที่ใช้รูปแบบของสัญญาณเชื่อมต่อหรือโพรโทคอลที่แตกต่างกัน เช่น ระหว่างเครือข่ายอินเทอร์เน็ตและเครือข่ายโทรศัพท์พื้นฐาน IP-PBX จะส่งต่อคำร้องขอไปยังเกตเวย์ก่อน เพื่อแปลงสัญญาณให้อยู่ในรูปแบบที่ Endpoint ปลายทางเข้าใจ

นอกจากการทำงานพื้นฐานเกี่ยวกับการส่งสัญญาณเชื่อมต่อและการรับส่งเสียงผ่านเครือข่ายอินเทอร์เน็ตตามที่ได้กล่าวมาแล้ว การใช้เครือข่ายอินเทอร์เน็ตเพื่อการส่งผ่านเสียงมีผลต่อการรับประกันคุณภาพในการให้บริการ (Quality of Service: QoS) โดยคุณภาพในการให้บริการเกี่ยวข้องกับระยะเวลาที่ใช้ส่งผ่านแพ็กเก็ตในเครือข่ายไปยังปลายทาง ความแตกต่างของระยะเวลาที่ใช้ส่งผ่านแต่ละแพ็กเก็ต และการสูญหายของแพ็กเก็ตซึ่งโดยส่วนใหญ่แล้วเกิดจากการจราจรที่หนาแน่นในเครือข่าย เทคโนโลยี VoIP มีการรับประกันคุณภาพในการให้บริการด้วย ดังนั้น อาจแบ่งโพรโทคอลที่เกี่ยวข้องกับ VoIP ตามลักษณะหน้าที่การทำงานได้

เป็น การส่งสัญญาณเชื่อมต่อ การรับส่งเสียงหรือวิดีโอ และการรับประกันคุณภาพในการให้บริการ หน้าที่ของแต่ละโพรโทคอลแสดงดังคำอธิบายต่อไปนี้

- **โพรโทคอลที่เกี่ยวข้องกับการส่งสัญญาณเชื่อมต่อ**

- H.323 ใช้สร้าง แก้ไข และยกเลิกการเชื่อมต่อ โดยสัญญาณเชื่อมต่อถูกเข้ารหัสให้อยู่ในรูปแบบไบนารี (Binary)

- SIP ใช้สร้าง แก้ไข และยกเลิกการเชื่อมต่อ โดยสัญญาณเชื่อมต่อเป็นข้อความที่ถูกเข้ารหัสให้อยู่ในรูปแบบ American Standard Code for Information Interchange (ASCII)

- Media Gateway Control Protocol (MGCP) ใช้ควบคุมเกตเวย์ที่เชื่อมต่อระหว่างเครือข่ายอินเทอร์เน็ตและเครือข่ายโทรศัพท์พื้นฐาน โดยแปลงข้อมูล (สัญญาณเชื่อมต่อ เสียง และวิดีโอ) จากรูปแบบที่ใช้ในเครือข่ายโทรศัพท์พื้นฐานเป็นแพ็กเก็ตที่ใช้ขนส่งข้อมูลในเครือข่ายอินเทอร์เน็ต

- **โพรโทคอลที่เกี่ยวข้องกับการรับส่งเสียงหรือวิดีโอ**

- Real-Time Transport Protocol (RTP) ใช้รับส่งเสียงหรือวิดีโอแบบเรียลไทม์

- **โพรโทคอลที่เกี่ยวข้องกับการรับประกันคุณภาพในการให้บริการ**

- Real-Time Streaming Protocol (RTSP) ใช้ควบคุมการรับส่งเสียงหรือวิดีโอที่มีการส่งอย่างต่อเนื่องในเครือข่าย เช่น ควบคุมการเล่นวิดีโอทำให้สามารถกดเล่น (Play) หรือหยุดชั่วคราว (Pause) ได้

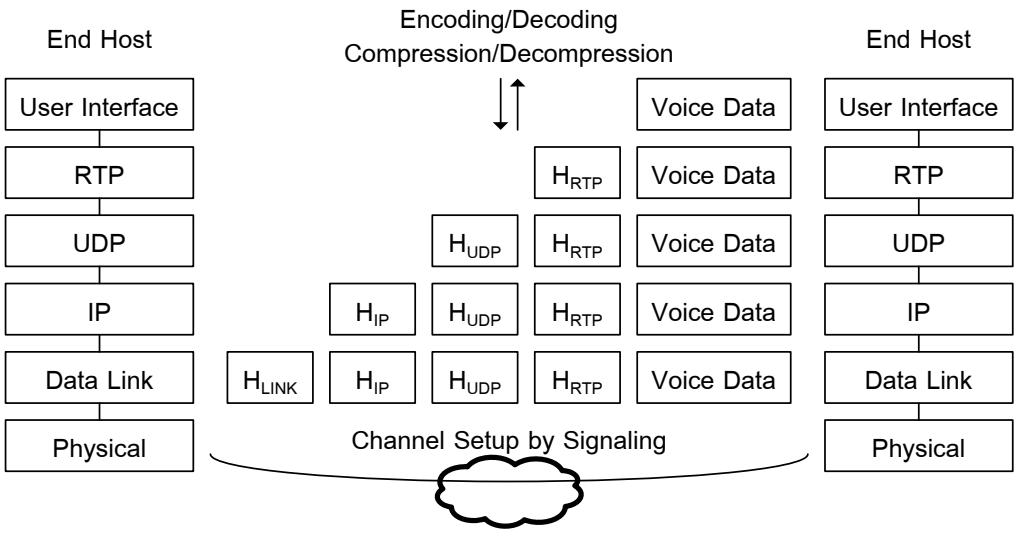
- Resource Reservation Protocol (RSVP) ใช้จองทรัพยากรในเครือข่าย เช่น เส้นทาง (Path) แบนด์วิดท์ (Bandwidth)

- Real-Time Transport Control Protocol (RTCP) ใช้คู่กับ RTP เพื่อควบคุมการส่งเสียงหรือวิดีโอแบบเรียลไทม์ โดยใช้ส่งข้อมูลเกี่ยวกับคุณภาพของการขนส่ง เช่น จำนวนแพ็กเก็ตที่สูญหายโดยเฉลี่ย และให้ข้อมูลเกี่ยวกับคู่สนทนา เช่น ชื่อและอีเมล เป็นต้น

โพรโทคอลที่เกี่ยวข้องกับ VoIP เหล่านี้มีการทำงานอยู่ในชั้นแอปพลิเคชัน ซึ่งต้องอาศัยโพรโทคอลอื่นในการรับส่งข้อมูลผ่านเครือข่ายอินเทอร์เน็ต ได้แก่ Transmission Control Protocol (TCP) และ User Datagram Protocol (UDP) โดย TCP มีการรับประกันความน่าเชื่อถือในการขนส่ง จึงอาจใช้รับส่งสัญญาณเชื่อมต่อเพื่อให้มีการรับส่งตามลำดับที่ถูกต้องและไม่สูญหาย ส่วน UDP ใช้สำหรับรับส่งเสียงและวิดีโอ เนื่องจากต้องการความรวดเร็วในการรับส่ง

โพรโทคอลที่เกี่ยวข้องกับการส่งผ่านเสียงของเทคโนโลยี VoIP คือ RTP โดยการประมวลผลข้อมูลเสียงของ VoIP เริ่มต้นจากการแปลงข้อมูลเสียงจากสัญญาณแอนะล็อกให้

เป็นสัญญาณดิจิทัล (Encoding) อาจมีการบีบอัดสัญญาณ แล้วนำมาสร้างเป็นแพ็กเก็ตโดยใช้ RTP ซึ่งจะมีเฮดเดอร์ฟิลด์ที่ใช้สำหรับการประกอบแพ็กเก็ตใหม่ แพ็กเก็ต RTP นี้เป็นข้อมูลที่จะถูกขนส่ง (Payload) บนเครือข่ายอินเทอร์เน็ตโดยใช้ UDP แพ็กเก็ตที่ส่งมายังปลายทางจะถูกประกอบใหม่โดยเรียงตามลำดับที่ถูกต้อง จากนั้นเสียงที่อยู่ในรูปสัญญาณดิจิทัลจึงจะถูกแปลงเป็นสัญญาณแอนะล็อก ขั้นตอนการประมวลผลข้อมูลเสียงของ VoIP แสดงดังภาพประกอบที่ 2.2

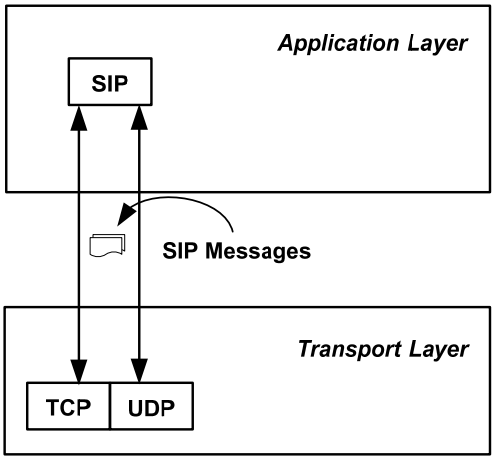


ภาพประกอบที่ 2.2 การประมวลผลข้อมูลเสียงของ VoIP (Butcher, 2007)

ปัจจุบัน โดยส่วนใหญ่ VoIP มีการใช้ H.323 และ SIP เพื่อส่งสัญญาณเชื่อมต่อ ซึ่ง H.323 เป็นชุดโพรโทคอลที่ถูกกำหนดมาตรฐานโดย International Telecommunications Union (ITU) ใช้สำหรับสื่อสารเสียง ข้อมูล และวิดีโอผ่านเครือข่ายอินเทอร์เน็ต สัญญาณเชื่อมต่อของ H.323 ถูกเข้ารหัสให้อยู่ในรูปแบบไบนารี สามารถแปลงสัญญาณเชื่อมต่อระหว่างเครือข่ายโทรศัพท์พื้นฐานและเครือข่าย VoIP ได้ง่ายกว่า SIP โดยที่ SIP ไม่ได้ออกแบบมาเพื่อเชื่อมต่อระหว่างเครือข่ายโดยตรง อย่างไรก็ตาม สัญญาณเชื่อมต่อของ SIP เป็นข้อความที่ถูกเข้ารหัสให้อยู่ในรูปแบบ ASCII จึงมีความยืดหยุ่น เข้าใจง่าย มีความซับซ้อนน้อยกว่าและใช้แบนด์วิดท์น้อยกว่า H.323 (SIP ใช้ข้อความเพื่อขอเชื่อมต่อ 1 ข้อความ ในขณะที่ H.323 ใช้ 8 ข้อความ (Olejniczak, 2009)) อีกทั้งฮาร์ดแวร์ที่ใช้ในการพัฒนาระบบโดยทั่วไปมีราคาถูกกว่าส่งผลให้มีการใช้งาน SIP เพิ่มขึ้นอย่างรวดเร็วเมื่อเทียบกับ H.323 การทำงานของ SIP มีรายละเอียดดังหัวข้อ 2.3

2.3 Session Initiation Protocol (SIP)

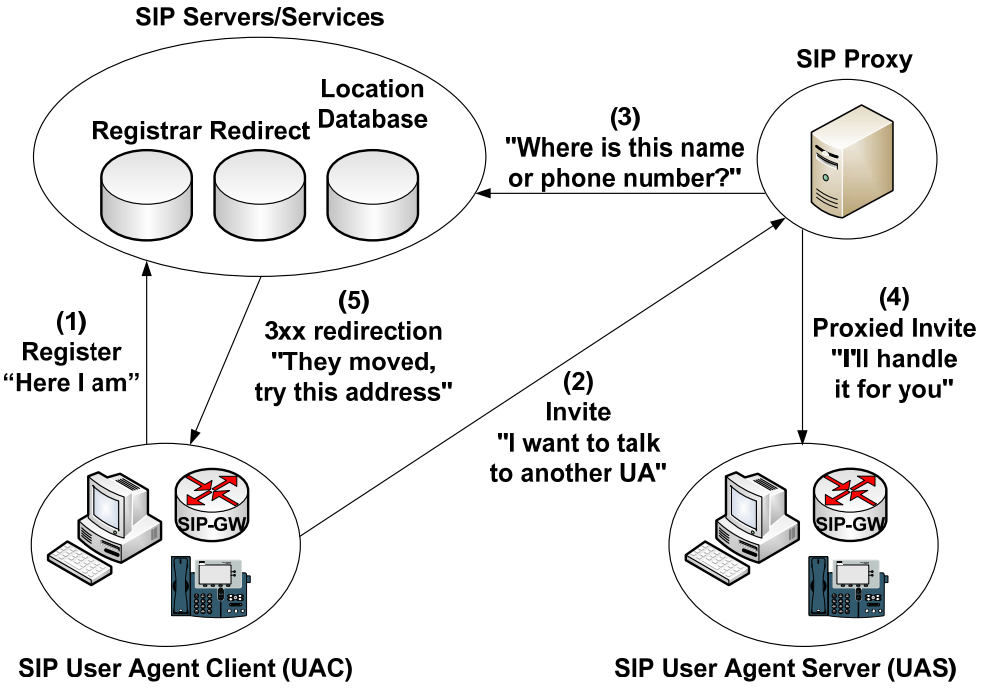
โพรโทคอล SIP ถูกกำหนดขึ้นใน RFC 2543 (Handley และคณะ, 1999) ในปี ค.ศ. 1999 โดย Multiparty Multimedia Session Control (MMUSIC) ซึ่งเป็นกลุ่มปฏิบัติการของ Internet Engineering Task Force (IETF) และในปี ค.ศ. 2002 IETF ได้ตีพิมพ์ SIP RFC ขึ้นมาใหม่คือ RFC 3261 (Rosenberg และคณะ, 2002) โดย SIP เป็นโพรโทคอลส่งสัญญาณเชื่อมต่อ ใช้สำหรับสร้าง แก้ไข และยกเลิกการเชื่อมต่อ เช่น การเชื่อมต่อโทรศัพท์ผ่านอินเทอร์เน็ต การกระจายภาพและเสียง และการประชุมทางไกลแบบแสดงภาพและเสียง การเชื่อมต่อที่เกิดขึ้นในช่วงเวลาหนึ่งเรียกว่าเซสชัน (Session) การรับส่งข้อมูลของ SIP ในรูปแบบข้อความตัวอักษร (Text-Based Protocol) เช่นเดียวกันกับ Hypertext Transfer Protocol (HTTP) หรือ Simple Mail Transfer Protocol (SMTP) SIP มีการทำงานอยู่ในชั้นแอปพลิเคชันและใช้ TCP หรือ UDP ในการรับส่งข้อมูลผ่านเครือข่ายอินเทอร์เน็ต ดังภาพประกอบที่ 2.3



ภาพประกอบที่ 2.3 การรับส่งข้อมูลผ่านเครือข่ายอินเทอร์เน็ตของ SIP

2.3.1 ส่วนประกอบของ SIP (SIP Element)

ส่วนประกอบของ SIP แบ่งออกเป็น User Agent และ SIP Server (Russell, 2008) ดังภาพประกอบที่ 2.4 และมีรายละเอียดดังต่อไปนี้



ภาพประกอบที่ 2.4 ส่วนประกอบของ SIP (Cisco Systems, 2002)

1) User Agent (UA)

คือซอฟต์แวร์ที่ฝังอยู่ภายในอุปกรณ์ สามารถสร้างคำร้องขอ (Request) และส่งคำตอบกลับ (Response) ไปยังการร้องขอได้ UA มี 2 ประเภทซึ่งสามารถอยู่ภายในอุปกรณ์เดียวกันได้แต่มีหน้าที่ต่างกัน ประกอบด้วย

- User Agent Client (UAC) ทำหน้าที่สร้างคำร้องขอ เป็นจุดเริ่มต้นของการสร้างเซสชัน คำร้องขอถูกสร้างขึ้นเมื่อผู้ใช้บริการหมุนโทรศัพท์หรือใช้แอปพลิเคชันอื่นๆ ในการติดต่อสื่อสาร เช่น อีเมล การส่งข้อความโต้ตอบแบบทันที (Instant Messaging) และการส่งข้อความสั้นๆ (Text Messaging) ซึ่ง UAC จะแปลคำสั่งที่ได้รับจากแอปพลิเคชันเพื่อสร้างเป็นข้อความ SIP แล้วส่งไปยังปลายทาง
- User Agent Server (UAS) ทำหน้าที่รับและตอบสนองคำร้องขอจาก UAC เมื่อ UAS ได้รับคำร้องขอจะพิจารณาก่อนว่าอุปกรณ์นั้นๆ สนับสนุนประเภทของคำร้องขอ (Method) ที่ส่งมาดังตัวอย่างในตารางที่ 2.1 หรือไม่ ถ้าไม่สนับสนุนจะส่งคำตอบกลับพร้อมกับพารามิเตอร์ (Parameter) ที่ระบุประเภทของคำร้องขอที่สนับสนุน

2) SIP Server แบ่งเป็น 3 ชนิด คือ Proxy, Redirect และ Registrar การทำงานของแต่ละเซิร์ฟเวอร์มีรายละเอียดดังนี้

- Proxy Server

มีหน้าที่รับคำร้องขอและคำตอบกลับ แปลความหมายของคำร้องขอ หาเส้นทาง และส่งต่อข้อความผ่านเครือข่ายในนามของ UA สามารถแบ่งประเภทของ Proxy Server ได้ดังนี้

- Stateful Proxy มีหน้าที่ดูแลสถานะของทุกๆ การเชื่อมต่อที่ผ่านพร็อกซี เมื่อมีคำร้องขอส่งไปยังพร็อกซีจะมีการเก็บข้อมูลที่ใช้ระบุการโทร (Call Identifier) ของเซสชันนั้นเอาไว้ เช่น ที่อยู่ของผู้โทรและผู้รับ ดังนั้น Stateful Proxy จะรับทราบเมื่อมีการเปลี่ยนสถานะของเซสชัน เช่น การยกเลิกเซสชันหรือมีการเปลี่ยนแปลงตำแหน่งที่อยู่ของผู้โทรและผู้รับ Stateful Proxy จะเพิ่มที่อยู่ของตนเองเข้าไปในเฮดเดอร์ “Via” เพื่อรักษาลำดับของเส้นทางเอาไว้ ทำให้มีการส่งคำตอบกลับผ่านพร็อกซีเดียวกันกับการส่งผ่านคำร้องขอ

- Stateless Proxy มีหน้าที่ส่งต่อคำร้องขอและคำตอบกลับเท่านั้น ไม่มีการเก็บข้อมูลของคำร้องขอและคำตอบกลับก่อนหน้านี้เอาไว้ และไม่มีการเก็บข้อมูลการหาเส้นทางอื่นๆ นอกเหนือจากตารางหาเส้นทางของตนเอง

- Forking Proxy ใช้สำหรับสร้างคำร้องขอจากคำร้องขอเดียวให้เป็นหลายคำร้องขอเพื่อส่งไปยังหลายๆ ปลายทาง สามารถใช้งานได้ในกรณีที่ผู้ใช้ลงทะเบียนตำแหน่งที่อยู่ไว้หลายแห่ง

- Redirect Server

มีหน้าที่ให้ข้อมูลเกี่ยวกับตำแหน่งที่อยู่อื่นๆ ที่สามารถส่งคำร้องขอไปได้ เช่น เมื่อได้รับคำร้องขอ Redirect Server จะส่งตำแหน่งที่อยู่ของ UAS ให้แก่ UAC เพื่อให้ UAC ส่งคำร้องขอไปยัง UAS โดยตรง ทำให้ลดภาระในการประมวลผลบน Proxy Server นอกจากนี้ ยังมีหน้าที่ให้ข้อมูลบริการอื่นๆ ที่อาจจะสามารถตอบสนองต่อการโทรได้ในกรณีที่การโทรไม่สำเร็จ เช่น กำหนดโพรโทคอลที่ใช้ในการขนส่งจาก UDP เป็น TCP

- Registrar Server

ใช้สำหรับการพิสูจน์ตัวตนและเก็บตำแหน่งที่อยู่ปัจจุบันของ UA ที่ลงทะเบียน ถ้าอุปกรณ์มีการเปลี่ยนตำแหน่งที่อยู่ (ทำให้ได้รับไอพีแอดเดรสใหม่) UA นั้นจะส่งคำร้องขอเพื่อลงทะเบียนตำแหน่งที่อยู่ใหม่ไปยัง Registrar Server โดย Registrar Server สามารถดำเนินการได้ 2 แบบคือ

- ยอมรับที่อยู่ใหม่และจัดเก็บลงในฐานข้อมูล ซึ่ง Location Service จะทำหน้าที่ให้ข้อมูลตำแหน่งที่อยู่ของ UA ที่ลงทะเบียนแก่ Redirect Server และ Proxy Server

- ปฏิเสธการลงทะเบียนครั้งแรกและบังคับให้ผู้ลงทะเบียนส่งข้อมูลประจำตัว (Credential) เพื่อยืนยันความถูกต้องว่าผู้ลงทะเบียนเป็นบุคคลที่ตนเองกล่าวอ้างจริงๆ

ขั้นตอนการทำงานระหว่าง UA และเซิร์ฟเวอร์ตามภาพประกอบที่ 2.4 มีรายละเอียดดังนี้

1. UAC ลงทะเบียนกับ Registrar Server เพื่อแจ้งตำแหน่งที่อยู่ที่สามารถติดต่อได้ โดยข้อมูลนี้จะถูกเก็บใน Location Database

2. เมื่อต้องการโทรออก UAC ส่งคำร้องขอเชื่อมต่อการโทรไปยังพร็อกซี

3. พร็อกซีสอบถามหมายเลขไอพีของ UAS ไปยัง Location Service

4. พร็อกซีอาจเพิ่มที่อยู่ของตนเองเข้าไปในคำร้องขอก่อนส่งไปยัง UAS เพื่อให้ UAS ส่งคำตอบกลับผ่านพร็อกซี

5. Redirect Server อาจให้ตำแหน่งที่อยู่อื่นที่สามารถส่งคำร้องขอไปได้ในกรณีที่พร็อกซีไม่สามารถให้บริการได้

จากการทำงานของส่วนประกอบต่างๆ พบว่าตำแหน่งที่อยู่เป็นสิ่งจำเป็นที่ช่วยให้ส่วนประกอบต่างๆ ของ SIP สามารถติดต่อสื่อสารกันได้ การกำหนดที่อยู่ของ SIP มีรายละเอียดดังหัวข้อ 2.3.2

2.3.2 การกำหนดตำแหน่งที่อยู่ของ SIP (SIP Addressing)

ตำแหน่งที่อยู่ของ SIP (SIP Address) ใช้ระบุผู้ใช้หรือทรัพยากรภายในเครือข่าย (Davidson, 2006) โดยปกติจะอ้างถึงแบบ Universal Resource Identifier (URI) คือมีรูปแบบคล้ายอีเมล ตัวอย่างของ SIP URI เช่น

sip:user@domain:port

sip:user@host:port

ฟิลด์ “user” ใช้ระบุชื่อของผู้ใช้โดยอาจใช้ชื่อหรือหมายเลขโทรศัพท์ ฟิลด์ “domain” และ “host” ใช้ระบุชื่อของเครื่องซึ่งอาจเป็นชื่อของเครื่องเซิร์ฟเวอร์ที่ให้บริการหรือชื่อของเครื่องผู้ใช้ ถ้าไม่ได้ระบุพอร์ต SIP URI จะมีค่าโดยปริยาย (Default) ของพอร์ตเป็น 5060 ดังตัวอย่าง

sip:john.doe@company.com:5060

sip:4081234567@proxy1.company.com

RFC 3261 (Rosenberg และคณะ, 2002) ได้กำหนดรูปแบบ SIP URI ที่มีการรักษาความปลอดภัย เรียกว่า SIPS URI ซึ่งมีค่าโดยปริยายของพอร์ตเป็น 5061 และมีรูปแบบคือ

sips:user@domain:port

sips:user@host:port

คำร้องขอและคำตอบกลับของ SIP เป็นข้อความตัวอักษรที่ใช้ SIP URI เพื่อระบุที่อยู่ของผู้ส่งและผู้รับ รูปแบบของข้อความ SIP มีรายละเอียดดังหัวข้อ 2.3.3

2.3.3 ข้อความ SIP (SIP Message)

โครงสร้างของข้อความ SIP ประกอบด้วย บรรทัดเริ่มต้น (Start-Line) ขึ้นอยู่กับชนิดของข้อความที่ส่งไป โดยข้อความ SIP แบ่งออกเป็น 2 แบบคือ คำร้องขอ SIP (SIP Request) และคำตอบกลับ SIP (SIP Response), เฮดเดอร์ฟิลด์ (Header Field), บรรทัดว่าง ใช้ระบุว่าสิ้นสุดส่วนของเฮดเดอร์ฟิลด์แล้ว และเนื้อหาของข้อความ (Message Body) โดยทุกส่วนของข้อความ SIP ต้องลงท้ายด้วย Carriage-Return Line-Feed (CRLF) ดังภาพประกอบที่ 2.5

Start-Line
*Message-Header
CRLF
[Message-Body]

ภาพประกอบที่ 2.5 โครงสร้างของข้อความ SIP โดยทั่วไป (Rosenberg และคณะ, 2002)

1) Start-Line

คำร้องขอจะถูกส่งจากไคลเอนต์ไปยังเซิร์ฟเวอร์ Start-Line ของคำร้องขอเรียกว่า Request-Line ส่วนประกอบของ Request-Line แสดงดังภาพประกอบที่ 2.6 โดย

- Method ใช้ระบุประเภทของการร้องขอ ซึ่ง RFC 3261 (Rosenberg และคณะ, 2002) ได้นิยามไว้ 6 Method ดังตารางที่ 2.1
- Request-URI คือ SIP หรือ SIPS URI ใช้ระบุผู้ใช้หรือบริการที่คำร้องขอนี้จะส่งไปถึง
- SIP-Version คือ เวอร์ชันของ SIP ที่ใช้อยู่

Method	Space	Request-URI	Space	SIP-Version	CRLF

ภาพประกอบที่ 2.6 Request-Line (Rosenberg และคณะ, 2002)

ตารางที่ 2.1 Method ของคำร้องขอ

Method	คำอธิบาย
REGISTER	ใช้เพื่อลงทะเบียนข้อมูลเกี่ยวกับตำแหน่งของผู้ใช้ปัจจุบัน (SIP Address และหมายเลขไอพี) กับ Registrar เซิร์ฟเวอร์
INVITE	ใช้เพื่อเชิญ UA อื่นเข้าร่วมเซสชัน และสามารถใช้ INVITE เพื่อแก้ไขคุณลักษณะของเซสชันได้สร้างไว้แล้ว
ACK	เป็นการยืนยันว่า UAC ได้รับการตอบสนองต่อคำร้องขอ INVITE คือจะส่ง ACK หลังจากได้รับคำตอบกลับ 200 OK โดย ACK จะถูกใช้กับคำร้องขอ INVITE เท่านั้น
CANCEL	ใช้ยกเลิกคำร้องขอที่ดำเนินการอยู่ โดยไม่ส่งผลต่อเซสชันที่สร้างขึ้นแล้ว
BYE	ใช้ร้องขอการยกเลิกเซสชันที่ได้สร้างไว้แล้ว
OPTIONS	ใช้ขอข้อมูลเกี่ยวกับความสามารถ (Capability) ของ UAS

ตัวอย่างของ Request-Line เช่น

- REGISTER sip:sipserver.cs.psu.ac.th:5060 SIP/2.0
- INVITE sip:bob@sipserver.cs.psu.ac.th SIP/2.0
- BYE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0

คำตอบกลับจะถูกส่งจากเซิร์ฟเวอร์ไปยังไคลเอนต์เพื่อระบุสถานะของคำร้องขอที่ไคลเอนต์ได้ส่งไปยังเซิร์ฟเวอร์ โดย UAS หรือพร็อกซีจะสร้างคำตอบกลับเพื่อตอบสนองต่อคำร้องขอที่ UAC ได้สร้างขึ้น Start-Line ของคำตอบกลับเรียกว่า Status-Line ส่วนประกอบของ Status-Line แสดงดังภาพประกอบที่ 2.7 โดย

SIP-Version	Space	Status-Code	Space	Reason-Phrase	CRLF

ภาพประกอบที่ 2.7 Status-Line (Rosenberg และคณะ, 2002)

- SIP-Version คือ เวอร์ชันของ SIP ที่ใช้อยู่
- Reason-Phrase คือคำอธิบายที่เป็นข้อความสั้นๆ ของ Status-Code

- Status-Code เป็นรหัสตัวเลข 3 หลักที่ระบุผลของการร้องขอ โดยที่หลักแรกจะบอกถึงประเภทของคำตอบกลับ ส่วน 2 หลักหลังไม่มีกฎในการจัดประเภท คำตอบกลับแบ่งออกเป็น 6 ประเภทดังนี้

- 1xx: Provisional ใช้สำหรับแสดงว่าได้รับคำร้องและเริ่มประมวลผลคำร้องขอนั้น

- 2xx: Success ใช้สำหรับแสดงว่าได้รับคำร้องขอแล้ว และคำร้องขอนั้นได้รับการยอมรับ

- 3xx: Redirection ใช้สำหรับให้ข้อมูลเกี่ยวกับตำแหน่งที่อยู่อื่นที่สามารถส่งการร้องขอไปได้ หรือบริการอื่นๆ ที่อาจจะสามารถตอบสนองต่อการโทรได้

- 4xx: Client Error ใช้สำหรับแสดงว่าคำร้องขอไม่ถูกหลักไวยากรณ์ หรือเซิร์ฟเวอร์ไม่สามารถดำเนินการตามคำร้องขอนั้นได้

- 5xx: Server Error ใช้สำหรับแสดงว่าเซิร์ฟเวอร์มีความผิดพลาดในการดำเนินการตามคำร้องขอ

- 6xx: Global Failure ใช้สำหรับแสดงว่าคำร้องขอไม่สามารถดำเนินการที่เซิร์ฟเวอร์ใดๆ ได้

ตัวอย่างของ Status-Line เช่น

- SIP/2.0 401 Unauthorized
- SIP/2.0 200 OK
- SIP/2.0 100 Giving a try

2) เฮดเดอร์ฟิลด์ (Header Field)

ก่อนอธิบายรายละเอียดของเฮดเดอร์ ควรทำความรู้จักกับความสัมพันธ์ระหว่าง UA ในการแลกเปลี่ยนข้อความ SIP ระหว่างกัน โดยความสัมพันธ์นี้แบ่งเป็น Dialog และ Transaction โดยที่

- Dialog เป็นความสัมพันธ์แบบ Peer-to-Peer ของ SIP ระหว่าง UA 2 ตัว ที่คงอยู่ในเวลาหนึ่งๆ Dialog ถูกระบุ (Identify) โดย Call-ID, Local Tag และ Remote Tag

- Transaction ในเทคโนโลยี VoIP คือชุดของการแลกเปลี่ยนข้อความที่เป็นอิสระจากกัน ถูกระบุโดย Call-ID, พารามิเตอร์ branch ของ Via, Local Tag, Remote Tag และ Cseq ซึ่ง SIP Transaction สามารถก่อให้เกิดการสร้าง การแก้ไข หรือการยกเลิกเซสชันได้ โดยการสร้างเซสชันจะก่อให้เกิดความสัมพันธ์ที่เรียกว่า Dialog

สำหรับข้อความ SIP แต่ละเฮดเดอร์ฟิลด์จะประกอบด้วยชื่อฟิลด์ ตามด้วยเครื่องหมายมหัพภาคคู่ (:) และค่าของฟิลด์ รูปแบบของเฮดเดอร์ฟิลด์แสดงดังนี้

Field-Name: Field-Value

เฮดเดอร์ฟิลด์โดยทั่วไปประกอบด้วย

- Via

แสดงถึงการขนส่ง (Transport) ที่ใช้ มีการระบุตำแหน่งที่ตั้ง (Location) ที่ต้องการใช้รับคำตอบกลับกลับมา และจะต้องประกอบด้วยพารามิเตอร์ branch ซึ่งใช้ในการระบุ Transaction ที่สร้างโดยคำร้องขอนั้น ตัวอย่างเช่น

Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bK776asdhds

- Max-Forwards

ใช้จำกัดจำนวนของอุปกรณ์ในเครือข่าย (Hop) ที่คำร้องขอสามารถส่งผ่านเพื่อไปถึงยังปลายทาง ค่าเริ่มต้นควรเป็น 70 และเมื่อเดินทางผ่านแต่ละ Hop ค่าจะลดลงทีละ 1 ตัวอย่างเช่น

Max-Forwards: 70

- To

ใช้กำหนดผู้รับคำร้องขอ ประกอบด้วย SIP หรือ SIPS URI อาจมีส่วนแสดงชื่อ (Display-Name) หรือไม่ก็ได้ และอาจมีพารามิเตอร์ tag เพื่อใช้ระบุ Dialog แต่ถ้ายังไม่มีการสร้าง Dialog ก็ไม่ต้องใส่ tag ตัวอย่างเช่น

To: Bob <sip:bob@biloxi.com>

- From

แสดงถึง Identity ของผู้เริ่มต้นสร้างคำร้องขอ ประกอบด้วย SIP หรือ SIPS URI อาจมีส่วนแสดงชื่อ (Display-Name) หรือไม่ก็ได้ และจะต้องประกอบด้วยพารามิเตอร์ tag ซึ่งเป็นข้อความแบบสุ่มที่เลือกโดย UAC เพื่อใช้ระบุ Dialog ตัวอย่างเช่น

From: Alice <sip:alice@atlanta.com>;tag=1928301774

- Call-ID

ทำหน้าที่เป็นตัวระบุชุดของข้อความที่เป็นกลุ่มเดียวกัน แต่ละกลุ่มจะมี Call-ID ที่ไม่ซ้ำกัน คำร้องขอและคำตอบสนองที่ส่งโดย UA ใน Dialog จะต้องมี Call-ID ที่เท่ากัน ตัวอย่างเช่น

Call-ID: a84b4c76e66710@pc33.atlanta.com

- CSeq

ใช้ระบุและจัดลำดับ Transaction ประกอบด้วยหมายเลขลำดับ (Sequence Number) และ Method ค่าของหมายเลขลำดับต้องเป็นตัวเลขที่ไม่มีเครื่องหมาย 32 บิต และต้องน้อยกว่า 2^{31} ตัวอย่างเช่น

CSeq: 314159 INVITE

- Contact

ให้ข้อมูล SIP หรือ SIPS URI ที่สามารถใช้ติดต่อกับ UA สำหรับส่งคำร้องขอถัดไป ตัวอย่างเช่น

Contact: <sip:alice@pc33.atlanta.com>

- Content-Type

แสดงประเภทของเนื้อหาของข้อความ (Message Body) ที่เพิ่มเข้าไปในคำร้องขอหรือคำตอบกลับ ถ้าในข้อความ SIP มีส่วนของ Body จะต้องมีเฮดเดอร์ Content-Type ตัวอย่างเช่น

Content-Type: application/sdp

- Content-Length

ใช้ระบุขนาด Message Body ของข้อความ SIP ในรูปแบบเลขฐานสิบ ตัวอย่างเช่น

Content-Length: 142

- Record-Route

พร็อกซีจะเพิ่ม Record-Route ที่ประกอบด้วยตำแหน่งที่อยู่ของตนเองเข้าไปในคำร้องขอเพื่อบังคับให้คำร้องขออื่นๆ ใน Dialog ถูกส่งผ่านพร็อกซี เพราะเมื่อ UAS ได้รับคำร้องขอนั้น จะใช้ Record-Route เพื่อพิจารณาเส้นทางสำหรับคำตอบกลับ ตัวอย่างเช่น

Record-Route: <sip:server10.biloxi.com;lr>,
<sip:bigbox3.site3.atlanta.com;lr>

- Route

ใช้เพื่อบังคับให้มีการส่งคำร้องขอผ่านเส้นทางตามรายชื่อของพร็อกซีที่ระบุไว้ ตัวอย่างเช่น

Route: <sip:bigbox3.site3.atlanta.com;lr>,
<sip:server10.biloxi.com;lr>

- Retry-After

Retry-After ใช้กับคำตอบกลับ 500 (Server Internal Error) หรือ 503 (Service Unavailable) เพื่อระบุว่าอาจใช้บริการไม่ได้เป็นระยะเวลานานเท่าไร โดยส่งไปยังไคลเอนต์ที่ร้องขอ และใช้กับคำตอบกลับ 404 (Not Found) 413 (Request Entity Too Large) 480 (Temporarily Unavailable) 486 (Busy Here) 600 (Busy) หรือ 603 (Decline) เพื่อแสดงว่าอีกนานเท่าไรที่ฝ่ายที่ถูกเรียก (UAS) คาดว่าจะพร้อมใช้อีกครั้งนับจากเวลาที่ได้รับคำตอบกลับ ค่าของฟิลด์นี้ประกอบด้วยตัวเลขที่เป็นจำนวนเต็มบวกฐานสิบมีหน่วยเป็นวินาที และอาจมีพารามิเตอร์ “duration” เพื่อบอกระยะเวลาที่สามารถเข้าถึงฝ่ายที่ถูกเรียกได้ เริ่มจากเวลาที่สามารถเริ่มเข้าถึงได้ ตัวอย่างเช่น

Retry-After: 18000; duration=3600
Retry-After: 120 (I'm in a meeting)

- Expires

ใช้กำหนดเวลาหมดอายุของคำร้องขอโดยเริ่มนับเวลาหลังจากได้รับคำร้องขอ มีหน่วยเป็นวินาที ตัวอย่างเช่น

Expires: 5

3) Message Body

เนื้อหาของข้อความ SIP อาจเป็นข้อความตัวอักษรที่ใช้สื่อสารกันระหว่างผู้ใช้ หรือเป็นรายละเอียดของเซสชัน เช่น ชนิดของข้อมูลที่ต้องการส่ง (เสียง วิดีโอ หรือข้อความ) และวิธีการแปลงสัญญาณ (Codec) เป็นต้น รายละเอียดของเซสชันจะใช้ Session Description Protocol (SDP) ในการอธิบาย ซึ่งรายละเอียดของ SDP จะกล่าวถึงในหัวข้อ 2.3.4

ตัวอย่างของข้อความ SIP ซึ่งประกอบด้วย Start-Line, Header Field และ Message Body แสดงดังภาพประกอบที่ 2.8 และ 2.9 โดยภาพประกอบที่ 2.8 เป็นคำร้องขอ Start-Line คือ Request-Line คำร้องขอนี้ใช้สำหรับร้องขอการเชื่อมต่อโดยผู้ใช้ชื่อ Alice ที่มี SIP URI คือ sip:alice@company.com ส่งไปยังผู้ใช้ชื่อ Bob ที่มี SIP URI คือ sip:bob@proxy.company.com เซสชันที่จะสร้างขึ้นเป็นเซสชันของการสื่อสารเสียงซึ่ง Alice จะใช้หมายเลขไอพี 172.18.193.102 ในการเชื่อมต่อ ส่วนภาพประกอบที่ 2.9 เป็นคำตอบกลับ Start-Line คือ Status-Line คำตอบกลับนี้ส่งจาก Bob เพื่อตอบกลับคำร้องขอจาก Alice โดยแจ้งว่าตกลงรับคำร้องขอและใช้หมายเลขไอพี 172.18.193.109 ในการเชื่อมต่อ

INVITE sip:bob@proxy.company.com SIP/2.0	**Request-Line**
Via: SIP/2.0/UDP	**SIP Message Headers**
ph1.company.com:5060;branch=z9hG4bK83749.1	
Max-Forwards: 70	
From: Alice <sip:alice@company.com>;tag=1234567	
To: Bob <sip:bob@proxy.company.com>	
Call-ID: 12345601@ph1.company.com	
CSeq: 1 INVITE	
Contact: <sip:alice@ph1.company.com>	
Content-Type: application/sdp	
Content-Length: 142	

Blank Line between SIP Header Fields and Body

v=0	**SDP Body in SIP Message**
o=alice 2890844526 28908445456 IN IP4 172.18.193.102	
s=Session SDP	
c=IN IP4 172.18.193.102	
t=0 0	
m=audio 49170 RTP/AVP 0	

ภาพประกอบที่ 2.8 ตัวอย่างข้อความ SIP Request (Davidson, 2006)

SIP/2.0 200 OK	**Status-Line**
Via: SIP/2.0/UDP	**SIP Message Headers**
ph1.company.com:5060;branch=z9hG4bK83749.1	
From: Alice <sip:alice@company.com>;tag=1234567	
To: Bob <sip:bob@proxy.company.com>;tag=9345678	
Call-ID: 12345601@ph1.company.com	
CSeq: 1 INVITE	
Content-Length: 142	

Blank Line between SIP Header Fields and Body

v=0	**SDP Body in 200 OK Response**
o=bob 3800844316 3760844696 IN IP4 172.18.193.109	
s=Session SDP	
c=IN IP4 172.18.193.109	
t=0 0	
m=audio 48140 RTP/AVP 0	

ภาพประกอบที่ 2.9 ตัวอย่างข้อความ SIP Response (Davidson, 2006)

2.3.4 Session Description Protocol (SDP)

SDP ใช้เพื่อส่งรายละเอียดของข้อมูลที่จำเป็นสำหรับสร้างเซสชันเมื่อมีการส่งข้อมูลมัลติมีเดีย (Multimedia) เช่น เสียง วิดีโอหรือข้อความตัวอักษรผ่านเครือข่ายอินเทอร์เน็ต SDP เป็นโพรโทคอลที่อยู่ในรูปแบบข้อความตัวอักษร โดยมีรูปแบบดังนี้

<Type>=<Value>

<Type> เป็นตัวอักษรตัวเดียวและเป็น Case-Significant (ตัวพิมพ์เล็กและตัวพิมพ์ใหญ่มีความหมายต่างกัน)

<Value> มีลักษณะเป็นข้อความซึ่งรูปแบบขึ้นอยู่กับ <Type>

SDP สามารถแบ่งได้เป็น 3 ส่วนคือ

1) Session-Level Descriptions

ตัวอย่างเฮดเดอร์ที่สำคัญ

● v = Protocol Version

ใช้ระบุเวอร์ชันของ SDP ที่สนับสนุนสำหรับเซสชันนี้ ทำให้ฝ่ายรับ (ทั้ง UAC และ UAS) ทราบว่าจะแปลบรรทัดอื่นๆ ใน SDP ได้อย่างไร ตัวอย่างเช่น

v=0

● o = Owner/Creator and Session Identifier

ใช้ระบุผู้ที่เริ่มต้นสร้างเซสชันและระบุเซสชัน มีรูปแบบดังนี้

o=<Username> <Session ID> <Version> <Network Type> <Address Type> <Address>

- <Username> คือชื่อผู้ใช้ที่เป็นผู้สร้างเซสชัน ต้องไม่ประกอบด้วยช่องว่าง
- <Session ID> เป็นตัวเลขที่ไม่ซ้ำกันเพื่อใช้ระบุเซสชัน
- <Version> เป็นตัวเลขเวอร์ชันของคำอธิบายเซสชัน ใช้เพื่อให้พร็อกซีตรวจสอบได้ว่าคำอธิบายเซสชันใดเป็นเวอร์ชันล่าสุด
- <Network Type> เป็นข้อความสั้นๆ ที่ระบุประเภทของเครือข่าย โดย “IN” ใช้หมายถึง "Internet"
- <Address Type> เป็นข้อความสั้นๆ ที่ระบุประเภทของตำแหน่งที่อยู่ที่จะตามมา เช่น “IP4” และ “IP6”
- <Address> คือที่อยู่ของเครื่องที่สร้างเซสชัน

ตัวอย่างเช่น

o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4

● s = Session Name

ใช้ระบุชื่อของเซสชัน ตัวอย่างเช่น

s=SDP Seminar

● c = Connection Information

ใช้แสดงข้อมูลของการเชื่อมต่อ มีรูปแบบดังนี้

c=<Network Type> <Address Type> <Connection Address>

- < Network Type> เป็นข้อความที่ระบุประเภทของเครือข่าย

- <Address Type> เป็นข้อความที่ระบุประเภทของตำแหน่งที่อยู่

- <Connection Address> คือตำแหน่งที่อยู่ที่ต้องการใช้รับส่งข้อมูล

ตัวอย่างเช่น

c=IN IP4 172.18.193.109

2) Time Descriptions

ตัวอย่างเฮดเดอร์ที่สำคัญ

● t = Time the Session is Active

ใช้ระบุเวลาเริ่มต้นและสิ้นสุดเซสชัน หากไม่ต้องการระบุเวลาที่ใช้เริ่มต้นหรือสิ้นสุดเซสชันให้ตั้งค่าเป็น 0 มีรูปแบบดังนี้

t=<start time> <stop time>

ตัวอย่างเช่น

t=2873397496 2873404696

3) Media Descriptions

ตัวอย่างเฮดเดอร์ที่สำคัญ

● m = Media Name and Transport Address

ใช้แสดงข้อมูลเกี่ยวกับมีเดียของเซสชันนั้นๆ ถ้ามีการสร้างมัลติมีเดียเซสชันอาจมีคำอธิบายมีเดียได้หลายอัน (แต่ละคำอธิบายใช้สำหรับมีเดียแต่ละประเภท) รูปแบบของคำอธิบายมีเดีย มีดังนี้

m=<Media> <Port> <Transport> <Media Formats>

- <Media> ใช้อธิบายชนิดของมีเดีย เช่น Audio, Video, Application, Data และ Control

- <Port> ใช้ระบุพอร์ตที่จะใช้รับเซสชัน ค่าของฟิลด์ Port ขึ้นอยู่กับชนิดของมีเดียที่สนับสนุน และโพรโทคอลขนส่ง (Transport Protocol) ที่ใช้เพื่อสนับสนุนมีเดีย

- <Transport> ระบุโพรโทคอลขนส่งที่ใช้ มี 2 ค่าคือ RTP/AVP หรือ UDP โดย RTP/AVP หมายถึงมีการใช้ RTP ในการขนส่งเสียงหรือวิดีโอโดยใช้ UDP ในการรับส่งผ่านเครือข่ายอินเทอร์เน็ต

- <Media Formats> ใช้ระบุรูปแบบของมีเดียที่กำหนด

ตัวอย่างเช่น

m=audio 49170 RTP/AVP 0

m=video 51372 RTP/AVP 31

m=application 32416 udp wb

2.3.5 กระบวนการการสื่อสารของ SIP

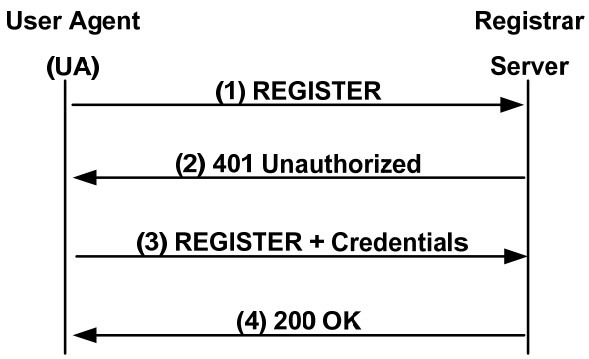
กระบวนการสื่อสารที่สำคัญของ SIP คือการลงทะเบียนและการเชื่อมต่อเซสชัน เมื่อ UA เข้าสู่เครือข่ายอินเทอร์เน็ตจะได้รับหมายเลขไอพี ซึ่ง UA จะต้องมีการลงทะเบียนกับเซิร์ฟเวอร์ที่เพื่อขอเข้าใช้บริการก่อน โดยชื่อของเครื่องเซิร์ฟเวอร์ที่ให้บริการหรือโดเมน (Domain) สังเกตได้จาก SIP URI กระบวนการลงทะเบียนอาจมีการพิสูจน์ตัวตนของผู้ใช้ ซึ่ง SIP ใช้กลไก HTTP Digest (Franks และคณะ, 1999) ในการพิสูจน์ตัวตน กระบวนการลงทะเบียนแสดงดังภาพประกอบที่ 2.10 และมีรายละเอียดดังต่อไปนี้

1. UA ส่งคำร้องขอ (1) REGISTER ซึ่งระบุตำแหน่งที่อยู่ในรูปแบบ SIP URI ไปยังเซิร์ฟเวอร์ที่รับผิดชอบ (รายละเอียดดูภาพประกอบที่ 2.11 บรรทัดที่ 7) และจะต้องลงทะเบียนใหม่เมื่อมีการเปลี่ยนแปลงหมายเลขไอพี หรือคำร้องขอลงทะเบียนหมดอายุ

2. เซิร์ฟเวอร์เรียกร้องให้มีการพิสูจน์ตัวตนโดยตอบกลับด้วยข้อความ 401 Unauthorized ที่ประกอบด้วยเฮดเดอร์ “WWW-Authenticate” (รายละเอียดดูภาพประกอบที่ 2.12 บรรทัดที่ 7)

3. UA ส่งคำร้องขอ (3) REGISTER ที่ประกอบด้วยข้อมูลประจำตัวที่ใช้ในการพิสูจน์ตัวตนไปยังเซิร์ฟเวอร์อีกครั้ง โดยข้อมูลประจำตัวนี้จะอยู่ในเฮดเดอร์ “Authorization” (รายละเอียดดูภาพประกอบที่ 2.13 บรรทัดที่ 8)

4. หากการพิสูจน์ตัวตนถูกต้อง เซิร์ฟเวอร์จะบันทึกหมายเลขไอพีของ UA พร้อมด้วย SIP URI จากเฮดเดอร์ “Contact” (รายละเอียดดูภาพประกอบที่ 2.13 บรรทัดที่ 7) ไปยัง Location Database แล้วตอบกลับด้วยข้อความ 200 OK



ภาพประกอบที่ 2.10 กระบวนการลงทะเบียน (Geneiatakis และคณะ, 2006)

```
1. REGISTER sip:sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1689-1-0
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:alice@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1689@10.0.0.3
6. CSeq: 1 REGISTER
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Max-Forwards: 70
9. Expires: 1800
10. User-Agent: SIPp/Linux
11. Content-Length: 0
```

ภาพประกอบที่ 2.11 Registration: (1) REGISTER

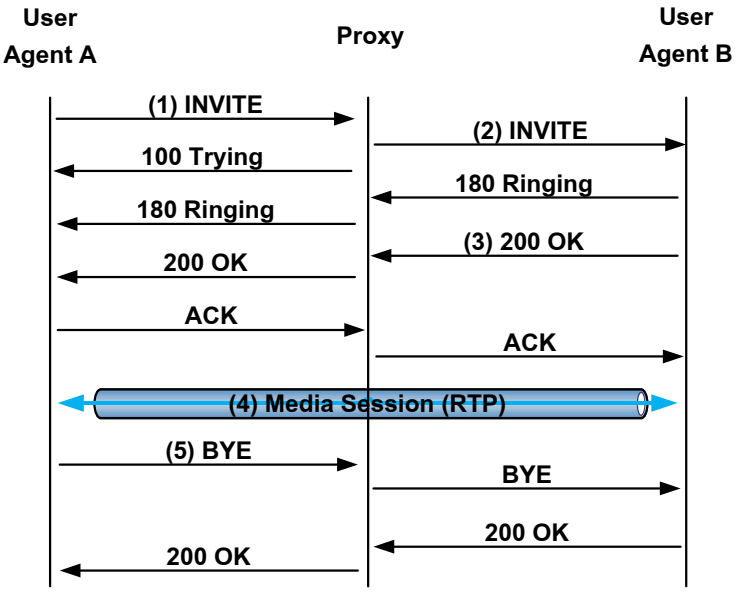
```
1. SIP/2.0 401 Unauthorized
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1689-1-0;
        received=10.0.0.3
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=96c42fb4eaeb131e386501201d85a818.7145
5. Call-ID: 1-1689@10.0.0.3
6. CSeq: 1 REGISTER
7. WWW-Authenticate: Digest realm="sipserver.cs.psu.ac.th",
        nonce="50b629e3fe1bb61807b447f0752a86042f25c5f2"
8. Server: OpenSIPS (1.6.4-2-tls (i386/linux))
9. Content-Length: 0
```

ภาพประกอบที่ 2.12 Registration: (2) 401 Unauthorized

1. REGISTER sip:sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1689-1-2
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:alice@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1689@10.0.0.3
6. CSeq: 2 REGISTER
7. **Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060**
8. **Authorization: Digest**
   username="alice",realm="sipserver.cs.psu.ac.th",uri="sip:10.0.0.2:5060",nonce="50b629e3fe1bb61807b447f0752a86042f25c5f2",response="91464ed926f16fe300e7807df16409eb",algorithm=MD5
9. Max-Forwards: 70
10. Expires: 1800
11. Content-Length: 0

ภาพประกอบที่ 2.13 Registration: (3) REGISTER

การสร้างเซสชันระหว่างผู้ใช้จะมีพร็อกซีเซิร์ฟเวอร์เข้ามาเกี่ยวข้องด้วย เพื่อรับคำร้องขอและคำตอบกลับ แปลความหมายของคำร้องขอ หาเส้นทาง และส่งต่อข้อความผ่านเครือข่ายในนามของ UA อาจมีกระบวนการพิสูจน์ตัวตนได้เช่นเดียวกับการลงทะเบียน กระบวนการในการสร้างเซสชันแสดงดังภาพประกอบที่ 2.14 และมีรายละเอียดดังนี้



ภาพประกอบที่ 2.14 การเชื่อมต่อผู้ใช้ใน SIP (Geneiatakis และคณะ, 2006)

1. UAC ส่ง (1) INVITE ไปยังพร็อกซี พร็อกซีจะแปลงชื่อเครื่องเซิร์ฟเวอร์ใน Request-URI (รายละเอียดดูภาพประกอบที่ 2.15 บรรทัดที่ 1) ให้เป็นชื่อเครื่องหรือหมายเลขไอพีของ UAS (รายละเอียดดูภาพประกอบที่ 2.16 บรรทัดที่ 1) โดยใช้ Location Service หรือ Domain Name System (DNS)

2. ก่อนส่งต่อคำร้องขอไปยังปลายทาง พร็อกซี่อาจเพิ่มเฮดเดอร์ “Via” และ “Record-Route” เพื่อระบุให้ UAS ส่งคำตอบกลับผ่านทางพร็อกซี (รายละเอียดดูภาพประกอบที่ 2.16 บรรทัดที่ 2 และ 3)

3. ถ้า UAS ยอมรับคำร้องขอจะส่งคำตอบกลับผ่านทางพร็อกซี (รายละเอียดดูภาพประกอบที่ 2.17)

4. เมื่อ UAC ได้รับคำตอบกลับก็สามารถสร้างเซสชันระหว่างผู้ใช้ทั้งสองได้โดยไม่ต้องผ่านพร็อกซี เพราะในระหว่างการแลกเปลี่ยนข้อความได้มีการแลกเปลี่ยนที่อยู่และพอร์ต (Port) เพื่อใช้ในการรับส่งมีเดียแล้ว

5. UAC ส่งข้อความ (5) BYE เมื่อต้องการยกเลิกเซสชัน ดังภาพประกอบที่ 2.18

```
1. INVITE sip:bob@sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2356-1-3
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:bob@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-2356@10.0.0.3
6. CSeq: 2 INVITE
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Max-Forwards: 70
9. Content-Type: application/sdp
10. Content-Length: 140

11. v=0
12. o=alice 53655765 2353687637 IN IP4 10.0.0.3
13. s=A conversation
14. c=IN IP4 10.0.0.3
15. t=0 0
16. m=audio 6000 RTP/AVP 0
17. a=rtpmap:0 PCMU/8000
```

ภาพประกอบที่ 2.15 SIP Session Setup: (1) INVITE

```
1. INVITE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Record-Route: <sip:10.0.0.2;lr=on;did=d47.b08c4da7>
3. Via: SIP/2.0/UDP 10.0.0.2;branch=z9hG4bK1876.839b6ec2.0
4. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;received=10.0.0.3;branch=z9hG4bK-
        2356-1-3
5. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
6. To: <sip:bob@sipserver.cs.psu.ac.th:5060>
7. Call-ID: 1-2356@10.0.0.3
8. CSeq: 2 INVITE
9. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
10. Max-Forwards: 69
11. Content-Type: application/sdp
12. Content-Length:   140

(Alice's SDP not shown)
```

ภาพประกอบที่ 2.16 SIP Session Setup: (2) INVITE

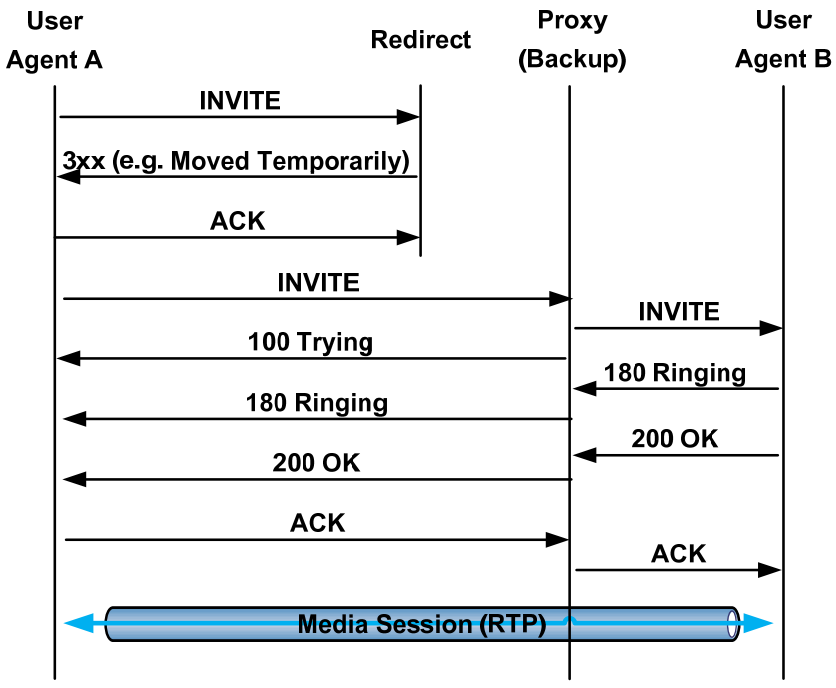
```
1. SIP/2.0 200 OK
2. Via: SIP/2.0/UDP 10.0.0.2;branch=z9hG4bK1876.839b6ec2.0
3. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;received=10.0.0.3;branch=z9hG4bK-
        2356-1-3
4. Record-Route: <sip:10.0.0.2;lr=on;did=d47.b08c4da7>
5. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
6. To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=2
7. Call-ID: 1-2356@10.0.0.3
8. CSeq: 2 INVITE
9. Contact: sip:bob@bobclient.sipserver.cs.psu.ac.th:5060
10. Content-Type: application/sdp
11. Content-Length:   138

12. v=0
13. o=bob 53655765 2353687637 IN IP4 10.0.0.4
14. s=A conversation
15. c=IN IP4 10.0.0.4
16. t=0 0
17. m=audio 6000 RTP/AVP 0
18. a=rtpmap:0 PCMU/8000
```

ภาพประกอบที่ 2.17 SIP Session Setup: (3) 200 OK

```
1. BYE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2356-1-8
3. Route: <sip:10.0.0.2;lr=on;did=d47.b08c4da7>
4. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
5. To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=2
6. Call-ID: 1-2356@10.0.0.3
7. CSeq: 3 BYE
8. Contact: sip:candle@candleclient.sipserver.cs.psu.ac.th:5060
9. Max-Forwards: 70
10. Content-Length: 0
```

ภาพประกอบที่ 2.18 SIP Session Setup: (5) BYE

อย่างไรก็ตาม ในบางสถานการณ์อาจไม่สามารถดำเนินการตามกระบวนการที่กล่าวมาได้ เช่น กรณีที่ไม่สามารถใช้งานพร็อกซีได้ชั่วคราว (อาจเกิดจากการใช้งานมากเกินไปหรือมีการปรับปรุงซอฟต์แวร์) ในกรณีนี้จะมีการเรียกใช้ Redirect Server เพื่อแจ้งให้ UAC ทราบถึงตำแหน่งที่อยู่อื่นๆ ของ URI ที่ถูกร้องขอ ดังนั้นผู้โทรสามารถสร้างคำร้องขอใหม่ไปยังตำแหน่งที่อยู่นั้น กระบวนการเชื่อมต่อโดยใช้ Redirect Server แสดงดังภาพประกอบที่ 2.19



ภาพประกอบที่ 2.19 การเชื่อมต่อโดยใช้ Redirect Server (Geneiatakis และคณะ, 2006)

กระบวนส่งสัญญาณเชื่อมต่อเพื่อสร้าง แก้ไขและยกเลิกเซสชันนี้สามารถถูกโจมตีได้ การโจมตีสัญญาณเชื่อมต่อของ SIP มีรายละเอียดดังหัวข้อ 2.3.6

2.3.6 การโจมตีสัญญาณเชื่อมต่อของ SIP (Signaling Attack)

ผู้บุกรุกสามารถใช้ประโยชน์จากข้อบกพร่องของกลไกการพิสูจน์ตัวตน และการขาดกลไกการรักษาความถูกต้องสมบูรณ์และการรักษาความลับ เพื่อโจมตีสัญญาณเชื่อมต่อโดยการแก้ไขหรือปลอมแปลงข้อความ SIP เพื่อเข้าถึงบริการโดยไม่ได้รับอนุญาตหรือเพื่อแก้ไขสถานะ (State) ของเซสชันที่ได้สร้างขึ้น ซึ่งก่อให้เกิดการปฏิเสธการให้บริการไปยังผู้ใช้ได้ การโจมตีที่สามารถเกิดขึ้นได้ มีดังนี้

1) Registration Hijacking

ผู้บุกรุกอาจโจมตีด้วยวิธีการส่งซ้ำเพื่อลงทะเบียนซ้ำกับเหยื่อ โดยการดักจับข้อความลงทะเบียน แล้วเปลี่ยนแปลงตำแหน่งที่อยู่สำหรับติดต่อกลับในส่วนของเฮดเดอร์ Contact ให้เป็นของตนเอง จากภาพประกอบที่ 2.20 คือบรรทัดที่ 7 แล้วส่งข้อความนั้นเพื่อลงทะเบียน ทำให้การเรียกเข้า เซสชันอื่นๆ หรืออีเมล ถูกส่งไปยังโทรศัพท์ของเหยื่อและของผู้บุกรุกด้วย

```
1. REGISTER sip:sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1689-1-2
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:alice@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1689@10.0.0.3
6. CSeq: 2 REGISTER
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Authorization: Digest
        username="alice",realm="sipserver.cs.psu.ac.th",uri="sip:10.0.0.2:5060",nonce="50b62
        9e3fe1bb61807b447f0752a86042f25c5f2",response="91464ed926f16fe300e7807df16
        409eb",algorithm=MD5
9. Max-Forwards: 70
10. Expires: 1800
11. User-Agent: SIPp/Linux
12. Content-Length: 0
```

ภาพประกอบที่ 2.20 ตัวอย่างข้อความ REGISTER

นอกจากนี้ ผู้บุกรุกอาจยกเลิกการลงทะเบียนของเหยื่อ โดยแก้หมายเลขลำดับของเฮดเดอร์ CSeq (บรรทัดที่ 6) ให้มีค่าเพิ่มขึ้น และแก้ไขค่าเฮดเดอร์ Expires (บรรทัดที่ 10) ของข้อความลงทะเบียนให้เป็น 0 แล้วส่งไปยังเซิร์ฟเวอร์ จากนั้น ส่งข้อความลงทะเบียนที่มีข้อมูลตำแหน่งที่อยู่ในเฮดเดอร์ Contact (บรรทัดที่ 7) เป็นของผู้บุกรุก วิธีการนี้จะทำให้การเรียกเข้า เซสชันอื่นๆ หรืออีเมล ถูกส่งไปยังโทรศัพท์ของผู้บุกรุกแทน

2) Invite Replay Billing Attack

Zhang และคณะ (2007) กล่าวถึงการโจมตีนี้ซึ่งสรุปได้ว่า เป็นการสร้างการเชื่อมต่อโดยไม่ได้รับอนุญาตโดยใช้ประโยชน์จากความผิดพลาดในการพัฒนาฟังก์ชันสำหรับป้องกันการนำข้อมูลกลับมาใช้ใหม่ (Anti-Replay) ในการพิสูจน์ตัวตนของ SIP โดยผู้บุกรุกดักจับข้อความ INVITE ที่มีข้อมูลประจำตัวสำหรับการพิสูจน์ตัวตน ดังภาพประกอบที่ 2.21 บรรทัดที่ 8 แล้วแก้ไขข้อมูลรายละเอียดของเซสชันที่ระบุในส่วนของ SDP เช่น หมายเลขไอพี (บรรทัดที่ 13 และ 16) และหมายเลขพอร์ต (บรรทัดที่ 17) เนื่องจากข้อมูลเหล่านี้ไม่ได้รับการป้องกันโดยการพิสูจน์ตัวตนของ SIP จากนั้น ผู้บุกรุกส่งข้อความ INVITE ที่ถูกแก้ไขแล้วไปยังพร็อกซีเพื่อร้องขอการเชื่อมต่ออีกครั้งหนึ่ง

```
1. INVITE sip:dan@sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1898-1-3
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:dan@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1898@10.0.0.3
6. CSeq: 2 INVITE
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Proxy-Authorization: Digest username="alice",realm="sipserver.cs.psu.ac.th",
        uri="sip:10.0.0.2:5060",nonce="50d4aa981dc9c857c19f0f389d6598186e3fa53d",
        response="f9613ee1833024edaad1c0b3a6794688",algorithm=MD5
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length:  195

12. v=0
13. o=alice 53655765 2353687637 IN IP4 10.0.0.3
14. s=A conversation
15. t=0 0
16. c=IN IP4 10.0.0.3
17. m=audio 6000 RTP/AVP 8
```

ภาพประกอบที่ 2.21 ตัวอย่างข้อความ INVITE สำหรับ Invite Replay Billing Attack

3) Call Establishment Hijacking

การโจมตีแบบ Call Establishment Hijacking (Zhang และคณะ, 2007) ผู้บุกรุกจะดักจับข้อความ INVITE ที่มีข้อมูลประจำตัวสำหรับการพิสูจน์ตัวตน ดังภาพประกอบที่ 2.22 บรรทัดที่ 8 แล้วแก้ไขหมายเลขไอพี (บรรทัดที่ 13 และ 16) และหมายเลขพอร์ต (บรรทัดที่ 17) ในส่วนของ SDP โดยแก้หมายเลขไอพีให้เป็นของตนเอง แล้วส่งไปยังพร็อกซี จากนั้นผู้บุกรุกส่งข้อความ BUSY ไปยังผู้โทร ทำให้ผู้โทรคิดว่าสายไม่ว่าง ทางฝั่งของผู้รับมีผู้บุกรุกอีกคนหนึ่งคอยดักจับข้อความ INVITE นี้ แล้วส่งข้อความ 200 OK ที่ระบุหมายเลขไอพีของตนเองและหมายเลขพอร์ตตอบกลับไป ทำให้ผู้บุกรุกทั้ง 2 ฝ่ายสามารถสื่อสารกันได้โดยที่ผู้ให้บริการจะคิดค่าโทรกับผู้โทร

นอกจากนี้ อาจแก้ไขข้อความ INVITE ตรงส่วน Request-URI (บรรทัดที่ 1) เพื่อระบุผู้รับที่ผู้บุกรุกต้องการสนทนาได้โดยไม่ต้องมีผู้บุกรุกอีกคนหนึ่งทางฝ่ายรับ

```
1. INVITE sip:dan@sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1898-1-3
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:dan@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1898@10.0.0.3
6. CSeq: 2 INVITE
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Proxy-Authorization: Digest
        username="alice",realm="sipserver.cs.psu.ac.th",uri="sip:10.0.0.2:5060",nonce=
        "50d4aa981dc9c857c19f0f389d6598186e3fa53d",response="f9613ee1833024edaa
        d1c0b3a6794688",algorithm=MD5
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length:  195

12. v=0
13. o=alice 53655765 2353687637 IN IP4 10.0.0.3
14. s=A conversation
15. t=0 0
16. c=IN IP4 10.0.0.3
17. m=audio 6000 RTP/AVP 8
18. a=rtpmap:8 PCMA/8000
```

ภาพประกอบที่ 2.22 ตัวอย่างข้อความ INVITE สำหรับ Call Establishment Hijacking

4) Call Termination Hijacking

เป็นการยืดระยะเวลาการโทรระหว่างผู้ใช้บริการ (Zhang และคณะ, 2007) โดยมีการดักจับข้อความ BYE ที่ผู้โทรหรือผู้รับได้ส่งออกมาเมื่อวางสาย แล้วส่งข้อความ 200 OK ตอบกลับไปเพื่อบอกว่าการโทรสิ้นสุดแล้ว ในขณะที่ผู้บุกรุกได้เข้าควบคุมการเชื่อมต่อที่ถูกสร้างขึ้น เนื่องจากก่อนหน้านี้ผู้บุกรุกได้บันทึกหมายเลขลำดับ (Sequence Number) ประทับเวลา (Time Stamp) และ Synchronization Source Identifier ของแพ็กเก็ต RTP ไว้ ผู้บุกรุกแค่สร้างแพ็กเก็ต RTP ปลอมขึ้นมาโดยใช้ หมายเลขลำดับ ประทับเวลา และ Synchronization Source Identifier ที่ถูกต้อง แล้วส่งไปยังเซิร์ฟเวอร์ทำให้คิดว่ายังมีการสนทนาระหว่างผู้โทรและผู้รับอยู่ ดังนั้นจึงมีการคิดค่าบริการของระยะเวลาการโทรที่เพิ่มขึ้นมานี้

5) UPDATE Attack

UPDATE ใช้เพื่อเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชันก่อนที่จะมีการเชื่อมต่อเซสชันเสร็จสมบูรณ์ ดังนั้น ผู้บุกรุกอาจปลอมแปลงข้อความ UPDATE โดยอาศัยข้อมูลจากข้อความ SIP อื่นๆ ได้แก่ Request-URI, Via, Route, From, To, Call-ID และ CSeq ดังภาพประกอบที่ 2.23 บรรทัดที่ 1-7 เพื่อส่งไปเปลี่ยนแปลงข้อมูลรายละเอียดของช่องทางการสื่อสารได้ เช่น หมายเลขไอพี (บรรทัดที่ 13 และ 15) และหมายเลขพอร์ต (บรรทัดที่ 17)

```
1. UPDATE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2105-1-7
3. Route: <sip:10.0.0.2;lr=on;did=cd.40395c44>
4. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
5. To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=1
6. Call-ID: 1-2105@10.0.0.3
7. CSeq: 4 UPDATE
8. Contact: sip:candle@candleclient.sipserver.cs.psu.ac.th:5060
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length:  141

12. v=0
13. o=candle 53655765 2353687637 IN IP4 10.0.0.5
14. s=A conversation
15. c=IN IP4 10.0.0.5
16. t=0 0
17. m=audio 6000 RTP/AVP 0
```

ภาพประกอบที่ 2.23 ตัวอย่างข้อความ UPDATE

6) Re-INVITE Attack

เมื่อเซสชันได้ถูกสร้างขึ้นโดยการส่งคำร้องขอ INVITE แล้ว สามารถส่งคำร้องขอ INVITE อีกครั้ง (Re-INVITE) ดังภาพประกอบที่ 2.24 เพื่อเปลี่ยนแปลงข้อมูลรายละเอียดของช่องทางการสื่อสาร เช่น หมายเลขไอพี (บรรทัดที่ 13 และ 15) และหมายเลขพอร์ต (บรรทัดที่ 17) ดังนั้น ผู้บุกรุกอาจปลอมแปลงข้อความ INVITE โดยอาศัยข้อมูลจากข้อความ SIP อื่นๆ ได้แก่ Request-URI, Via, Route, From, To, Call-ID และ CSeq (บรรทัดที่ 1-7) เพื่อแก้ไขเซสชัน ทำให้เกิดการปฏิเสธการให้บริการ (Denial of Service: DoS) ไปยังผู้ใช้งานที่แท้จริงได้

```
1. INVITE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2356-1-8
3. Route: <sip:10.0.0.2;lr=on;did=d47.b08c4da7>
4. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
5. To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=2
6. Call-ID: 1-2356@10.0.0.3
7. CSeq: 3 INVITE
8. Contact: sip:candle@candleclient.sipserver.cs.psu.ac.th:5060
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length:  141

12. v=0
13. o=candle 53655765 2353687637 IN IP4 10.0.0.5
14. s=A conversation
15. c=IN IP4 10.0.0.5
16. t=0 0
17. m=audio 6000 RTP/AVP 0
18. a=rtpmap:0 PCMU/8000
```

ภาพประกอบที่ 2.24 ตัวอย่างข้อความ INVITE สำหรับ Re-INVITE Attack

7) BYE Attack

คำร้องขอ BYE ใช้สำหรับการยกเลิกเซสชัน ผู้บุกรุกสามารถดักจับข้อมูลในเครือข่ายแล้วนำมาสร้างคำร้องขอ BYE เพื่อยกเลิกเซสชันได้ โดยเฉพาะอย่างยิ่งในกรณีที่ไม่มีกลไกการพิสูจน์ตัวตนเมื่อมีการส่งคำร้องขอ BYE โดยภาพประกอบที่ 2.25 เป็นตัวอย่าง

ข้อความ BYE ที่สร้างขึ้นจากข้อความ ACK โดยการแก้ไขชื่อ Method ใน Request-Line และเฮดเดอร์ CSeq (บรรทัดที่ 1 และ 7) และแก้ไขหมายเลขลำดับในเฮดเดอร์ CSeq

```
1. BYE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1674-1-8
3. Route: <sip:10.0.0.2;lr=on;did=144.fc0de986>
4. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
5. To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=2
6. Call-ID: 1-1674@10.0.0.3
7. CSeq: 3 BYE
8. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
9. Max-Forwards: 70
10. Content-Length: 0
```

ภาพประกอบที่ 2.25 ตัวอย่างข้อความ BYE

8) CANCEL Attack

คำร้องขอ CANCEL ใช้สำหรับยกเลิกคำร้องขอที่ส่งมาจากไคลเอนต์ก่อนหน้านี้ ผู้บุกรุกอาจใช้ประโยชน์ของ CANCEL ในการยกเลิกคำร้องขอ INVITE ที่ส่งมาจากผู้ใช้งานที่แท้จริงได้ โดยภาพประกอบที่ 2.26 เป็นตัวอย่างข้อความ CANCEL ที่ใช้สำหรับยกเลิกคำร้องขอ INVITE มีการใช้ Request-URI, Via, From, To, Call-ID และ CSeq จากคำร้องขอ INVITE (บรรทัดที่ 1-6) โดยแก้ไขชื่อ Method ใน Request-Line และเฮดเดอร์ CSeq (บรรทัดที่ 1 และ 6) ให้เป็น CANCEL

```
1. CANCEL sip:bob@sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1711-1-3
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:bob@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1711@10.0.0.3
6. CSeq: 2 CANCEL
7. Content-Length: 0
```

ภาพประกอบที่ 2.26 ตัวอย่างข้อความ CANCEL

2.3.7 กลไกการรักษาความมั่นคงปลอดภัยสำหรับ SIP

1) HTTP Digest

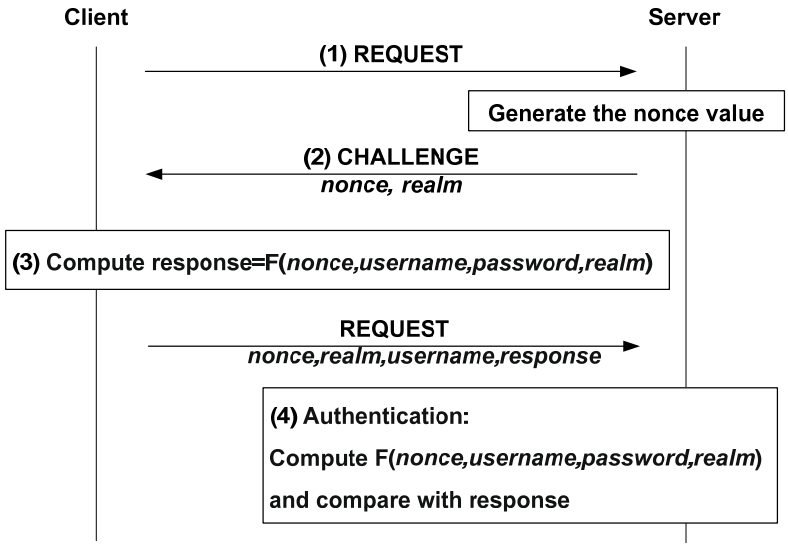
โดยทั่วไป SIP มีการพิสูจน์ตัวตนโดยใช้ HTTP Digest ซึ่งใช้การแลกเปลี่ยนข้อความ Challenge/Request เพื่อตรวจสอบความถูกต้องของ User Agent, Registrar และ Proxy กระบวนการพิสูจน์ตัวตนแสดงดังภาพประกอบที่ 2.27 และมีกระบวนการ ดังนี้

(1) ไคลเอนต์ส่ง REQUEST ไปยังเซิร์ฟเวอร์

(2) เซิร์ฟเวอร์ตอบกลับด้วย CHALLENGE ซึ่งประกอบด้วยค่า nonce และ realm โดยค่า nonce คือสายอักขระ (String) ที่เซิร์ฟเวอร์สร้างขึ้นอย่างไม่ซ้ำกันในแต่ละครั้งที่มีการร้องขอการพิสูจน์ตัวตน อยู่ในรูปแบบเลขฐาน 16 หรือ base64 (ประกอบด้วยตัวอักษรภาษาอังกฤษตัวพิมพ์ใหญ่ ตัวพิมพ์เล็ก ตัวเลข “+” และ “/”) ใช้สำหรับป้องกันการโจมตีด้วยวิธีการนำข้อมูลกลับมาใช้ใหม่ ส่วน realm คือชื่อโดเมนที่ใช้สำหรับการพิสูจน์ตัวตน ข้อความที่ตอบกลับอาจเป็น 401 Unauthorized ซึ่งใช้ในระหว่างการลงทะเบียน หรือ 407 Proxy Authentication Required ที่ใช้ในระหว่างการสร้างการเชื่อมต่อ

(3) ไคลเอนต์คำนวณ response = F(nonce; username; password; realm) โดยค่า username และ password คือค่าที่ใช้ร่วมกันกับเซิร์ฟเวอร์ แล้วส่ง REQUEST พร้อมด้วยค่า response, username, nonce และ realm ไปยังเซิร์ฟเวอร์ ค่า response, username, nonce และ realm เป็นข้อมูลประจำตัวที่ใช้ในการพิสูจน์ตัวตน

(4) เซิร์ฟเวอร์ตรวจสอบค่า nonce ถ้าค่า nonce ถูกต้อง เซิร์ฟเวอร์จะดึงรหัสผ่านโดยใช้ username เพื่อคำนวณค่า response แล้วนำไปเปรียบเทียบกับค่า response ที่ได้รับ ถ้าตรงกันแสดงว่าการพิสูจน์ตัวตนของไคลเอนต์ถูกต้อง



ภาพประกอบที่ 2.27 กระบวนการ HTTP Digest ใน SIP (Wu และคณะ, 2009)

ตัวอย่างข้อความ SIP ที่ใช้ HTTP Digest สำหรับการพิสูจน์ตัวตนในขั้นตอนการลงทะเบียน แสดงดังหัวข้อ 2.3.5 ภาพประกอบที่ 2.11, 2.12 และ 2.13

2) IP Security (IPsec)

IPsec (Tiller, 2000) ประกอบด้วยโพรโทคอล Encapsulating Security Payload (ESP) และ Authentication Header (AH) ที่สามารถใช้รักษาความลับและความถูกต้องสมบูรณ์ของข้อมูล การพิสูจน์ตัวตนของผู้ส่งข้อความ รวมทั้งการป้องกันการนำข้อมูลกลับมาใช้ใหม่ (Anti-Replay) และการวิเคราะห์การจราจรของเครือข่ายได้ และมีการทำงานเป็นอิสระจากโพรโทคอลชั้นเครือข่าย (Network Level Protocol) ในสถาปัตยกรรมอินเทอร์เน็ต

3) Transport Layer Security (TLS)

TLS (Rescorla, 2001) สามารถใช้ในการพิสูจน์ตัวตนซึ่งกันและกันของเอนทิตี (Entity) ต่างๆ ในระบบเครือข่ายได้ โดยการแลกเปลี่ยนใบรับรองซึ่งกันและกันในระหว่างกระบวนการเริ่มต้นการเชื่อมต่อ (Handshaking Procedure)

4) S/MIME

ข้อความ SIP สามารถประกอบด้วยส่วนของ MIME ซึ่งบริการด้านความปลอดภัยที่มีอยู่คือการใช้ Secure MIME (S/MIME) (Ramsdell, 2004) สามารถใช้ S/MIME ในการรักษาความลับและความถูกต้องสมบูรณ์ของข้อมูลตั้งแต่ต้นทางจนถึงปลายทางได้โดยวิธีการสร้างอุโมงค์ให้กับ SIP (Tunneling SIP) การสร้างอุโมงค์ใช้วิธีการห่อหุ้ม (Encapsulate) ข้อความ SIP ทั้งข้อความไว้ใน MIME Body และย่อยข้อความหรือเข้ารหัส MIME Body แล้วส่งข้อความนี้ไปพร้อมกับข้อความ SIP ต้นฉบับหรือข้อความที่ประกอบด้วยเฮดเดอร์ที่ใช้ในการหาเส้นทางจากข้อความ SIP ต้นฉบับ

แม้ว่า SIP จะมีกลไกมาตรฐานที่ช่วยรักษาความปลอดภัยอยู่แล้วแต่ก็มีข้อจำกัดในการใช้งานซึ่งการวิเคราะห์ข้อจำกัดต่างๆ มีการอธิบายรายละเอียดในบทถัดไป ดังนั้นจึงมีงานวิจัยที่ออกแบบกลไกสำหรับป้องกันการโจมตีหลายงานวิจัย ดังแสดงในหัวข้อ 2.4

2.4 งานวิจัยที่เกี่ยวข้อง

เพื่อแก้ปัญหาการโจมตีสัญญาณเชื่อมต่อที่เกิดขึ้นกับโพรโทคอล SIP ในเทคโนโลยี VoIP ได้มีการเสนอระบบตรวจจับการบุกรุกสำหรับระบบ VoIP ที่เรียกว่า SCIDIVE (Wu และคณะ, 2004) ซึ่งประกอบด้วย 2 แนวคิดในการตรวจจับ คือการตรวจจับแบบ Stateful

(Stateful Detection) และการตรวจจับแบบ Cross-Protocol (Cross-Protocol Detection) การตรวจจับแบบ Stateful จะกำหนดสถานะปัจจุบันจากหลายแพ็กเก็ตที่เกี่ยวข้องกันในการเชื่อมต่อและตรวจจับความผิดปกติโดยใช้ Rule Matching Engine ส่วนการตรวจจับแบบ Cross-Protocol จะพิจารณาประเภทของโพรโทคอลที่เกี่ยวข้องกับระบบ VoIP คือโพรโทคอลที่จัดการเกี่ยวกับการโทร เช่น โพรโทคอล SIP หรือ H.323 และโพรโทคอลที่เกี่ยวกับการส่งสัญญาณเสียง เช่น โพรโทคอล RTP ระบบนี้สามารถตรวจสอบการโจมตีจากการส่งข้อความ BYE โดยตรวจสอบว่าหลังจากมีการส่งข้อความ BYE แล้วยังมีการส่งข้อความเสียงมาจากคู่สนทนาหรือไม่ จุดอ่อนของระบบคือผู้บุกรุกสามารถหลีกเลี่ยงกลไกตรวจจับการบุกรุกโดยการโจมตีให้คู่สนทนาถูกปฏิเสธการให้บริการก่อนที่ผู้บุกรุกจะส่งข้อความ BYE ไปยังเป้าหมาย

ต่อมา Cao และ Jennings (2006) เสนอการพิสูจน์ตัวตนของการส่งข้อความตอบกลับและมีการรักษาความถูกต้องสมบูรณ์ของข้อความตอบกลับด้วย กลไกนี้จึงให้บริการความปลอดภัยแก่ผู้โทรเพียงฝ่ายเดียว

Peterson และ Jennings (2006) นำเสนอสถาปัตยกรรมการระบุตัวตนของผู้สร้างข้อความ SIP เพื่อป้องกันการปลอมตัว โดยกำหนดฟิลด์ส่วนหัวของโพรโทคอล SIP ขึ้นมาใหม่ 2 ฟิลด์ คือ Identity ใช้แสดงถึงลายเซ็นที่ใช้สำหรับตรวจสอบความถูกต้องของ Identity และ Identity-Info ใช้แสดงถึงการอ้างอิงถึงหนังสือรับรอง (Certificate) ของผู้ลงนาม ระบบมีการทำงานโดยสรุปคือ เมื่อเซิร์ฟเวอร์พิสูจน์ตัวตนและตรวจสอบความถูกต้องว่าผู้โทรได้รับสิทธิ์ในการยืนยันตัวตน (Identity) ที่แสดงในฟิลด์ส่วนหัว From แล้ว พร็อกซีเซิร์ฟเวอร์จะสร้างแฮชจากฟิลด์ส่วนหัว From และเนื้อหา (Body) ในข้อความ ซึ่งแฮชนี้จะถูกลงนามพร้อมกับหนังสือรับรองสำหรับโดเมนของผู้โทรและถูกใส่เข้าไปในฟิลด์ส่วนหัว Identity ของข้อความ SIP และใส่ URI เพื่อบอกฝ่ายรับว่าจะได้รับหนังสือรับรองของผู้ลงนามได้จากที่ใดเข้าไปในฟิลด์ส่วนหัว Identity-Info เพื่อสื่อสารไปยังเอนทิตี (Entity) อื่นๆ ของ SIP ว่าผู้ส่งได้รับการพิสูจน์ตัวตนแล้วและได้รับสิทธิ์ในการใช้ฟิลด์ส่วนหัว From แต่เป็นการรับประกันการระบุตัวตนของ SIP Requests เท่านั้น ไม่ครอบคลุม SIP Responses

Nassar และคณะ (2007) ได้นำ Honeypot และ SIP Correlation Engine มาใช้เพื่อตรวจจับและการป้องกันการบุกรุก Honeypot สามารถป้องกัน SPIT และ VoIP Phishing (Vishing) รวมถึงการกระทำที่เป็นการสอดแนมอื่นๆ ส่วน SIP Correlation Engine สามารถตรวจจับการปฏิเสธการให้บริการ และการโกงการใช้งาน (Fraudulent Usage) ได้ ซึ่งระบบนี้สามารถตรวจจับการส่งข้อความ BYE ได้ด้วยวิธีเดียวกับ SCIDIVE

ต่อมา Geneiatakis และ Lambrinoudakis (2008) เสนอกลไกการป้องกันการโจมตีด้วยการส่งสัญญาณโดยการใช้เฮดเดอร์ฟิลด์ Integrity-Auth และฟังก์ชันแฮช เพื่อตรวจสอบความถูกต้องสมบูรณ์ของทุกๆ ข้อความ แต่อาจเกิดการโจมตีด้วยการเดารหัสผ่าน

(Off-line Password Guessing) ได้ และพร็อกซีต้องรู้รหัสผ่านของทั้ง 2 ฝ่าย วิธีการนี้จึงใช้ได้ดีเมื่อผู้ใช้อยู่ในโดเมนเดียวกัน

นอกจากนี้ Shekokar และ Devane (2010) เสนอให้ใช้ TLS เพื่อเข้ารหัสข้อมูลทุกอย่างที่ส่งระหว่างไคลเอนต์และเซิร์ฟเวอร์ ซึ่งเสียเวลาในการเข้ารหัสและถอดรหัสทุก ๆ ข้อความ

2.5 สรุป

ในบทนี้ได้นำเสนอทฤษฎีที่เกี่ยวข้องกับงานวิจัย ทั้งเทคโนโลยี VoIP และโพรโตคอล SIP ตลอดจนงานวิจัยที่เกี่ยวข้อง ซึ่งงานวิจัยส่วนใหญ่ยังสามารถป้องกันการโจมตีสัญญาณเชื่อมต่อ SIP ได้ไม่หลากหลาย จึงได้มีการนำเสนอวิทยาวิทยานิพนธ์ชุดนี้ขึ้นเพื่อออกแบบกลไกสร้างความมั่นคงสำหรับสัญญาณเชื่อมต่อของ SIP โดยจะกล่าวถึงการวิเคราะห์และออกแบบกลไกในบทถัดไป

บทที่ 3

การวิเคราะห์และออกแบบระบบ

3.1 บทนำ

จากที่ได้กล่าวมาแล้วในบทที่ 2 ว่าบริการ VoIP สามารถใช้ SIP, H.323 และ MGCP ในการส่งสัญญาณเชื่อมต่อได้ ซึ่ง SIP เป็นโพรโทคอลที่มีการนำมาใช้กันมากในปัจจุบัน งานวิจัยนี้จึงมุ่งเน้นศึกษาเกี่ยวบริการ VoIP ที่มีการใช้ SIP เป็นโพรโทคอลส่งสัญญาณเชื่อมต่อ

บทที่แล้วมีการอธิบายรายละเอียดเกี่ยวกับโพรโทคอล SIP รวมถึงปัญหาและการโจมตีสัญญาณเชื่อมต่อของ SIP ในบทนี้จะมีการวิเคราะห์การโจมตีสัญญาณเชื่อมต่อของ SIP ว่าสามารถเกิดขึ้นได้โดยอาศัยจุดอ่อนใดบ้าง มีการแสดงตัวอย่างการโจมตีพร้อมทั้งวิเคราะห์ผลที่เกิดจากการโจมตีและวิเคราะห์ส่วนต่างๆ ของข้อความ SIP ที่เกี่ยวข้องกับการโจมตี จากนั้นมีการออกแบบกลไกที่ช่วยป้องกันการโจมตีสัญญาณเชื่อมต่อ คือ SIP Extension for Signaling Attacks Protection (SIPE-SAP) โดยการออกแบบจะประกอบด้วยส่วนของกระบวนการทำงานของ SIPE-SAP การนิยามเฮดเดอร์ขึ้นมาใหม่ และตัวอย่างการใช้งาน ดังรายละเอียดต่อไปนี้

3.2 ข้อมูลที่เกี่ยวข้องกับการวิเคราะห์ระบบ

เนื่องจากงานวิจัยนี้เป็นการศึกษาเทคโนโลยีสำหรับการสื่อสารเสียงผ่านเครือข่ายอินเทอร์เน็ต หรือ VoIP ซึ่งก่อนที่จะสามารถรับส่งข้อมูลเสียง วิดีโอหรือข้อความระหว่างกันได้ ต้องมีกระบวนการส่งสัญญาณเชื่อมต่อก่อน โพรโทคอลส่งสัญญาณเชื่อมต่อสำหรับสร้าง แก้ไข หรือยกเลิกการเชื่อมต่อที่มีการศึกษาในงานวิจัยนี้คือ SIP ซึ่ง SIP สามารถส่งข้อมูลรายละเอียดที่จำเป็นต่อการสร้างการเชื่อมต่อได้ เช่น ชื่อของเซสชัน หมายเลขไอพีและหมายเลขพอร์ตที่ใช้ในการรับส่งข้อมูล รวมทั้งประเภทของข้อมูลที่ใช้รับส่งกัน โดย SIP อาศัย SDP ในการอธิบายข้อมูลรายละเอียดของเซสชัน สามารถศึกษาการทำงานและรูปแบบของ SIP รวมถึงรูปแบบของ SDP ได้จากบทที่ 2 หัวข้อ 2.3

ในแง่ของบริการ VoIP ความต้องการทางด้านความปลอดภัยของ SIP (SIP Security Requirements) อาจแบ่งได้เป็น 5 หมวดหมู่ ดังนี้

1) การรักษาความลับ (Confidentiality) เป็นการรับประกันว่าจะมีเพียงผู้ที่ได้รับอนุญาตเท่านั้นที่สามารถเข้าถึงข้อความ SIP ได้

2) การรักษาความสมบูรณ์ของข้อมูล (Integrity) ซึ่งข้อมูลในที่นี้หมายถึงข้อความ SIP เนื่องจากเอนทิตี (Entity) ซึ่งเป็นส่วนประกอบต่างๆ ในเครือข่ายของ SIP มีความจำเป็นต้องเข้าถึงข้อความ SIP เพื่อประมวลผลและกำหนดเส้นทางของข้อมูลให้ไปยังปลายทางได้อย่างถูกต้อง ดังนั้นการรักษาความสมบูรณ์ของข้อมูลเป็นการรับประกันว่าข้อความ SIP สามารถถูกเปลี่ยนแปลงแก้ไขได้โดยเอนทิตีของ SIP ที่ได้รับอนุญาตเท่านั้น นอกจากนี้สามารถรับประกันได้ว่าบันทึกการโทรมีความถูกต้องด้วย

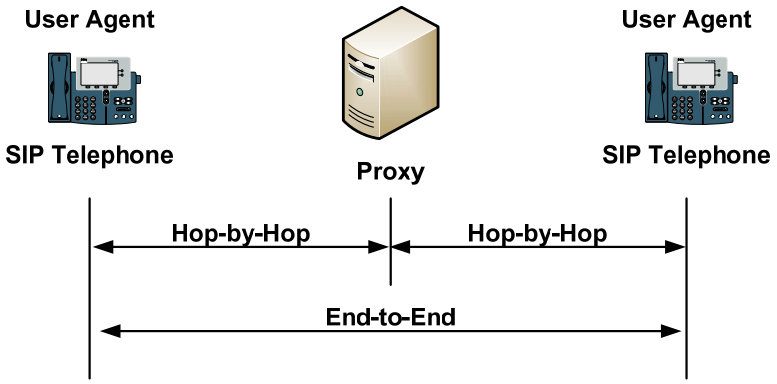
3) การรักษาความพร้อมใช้ (Availability) ผู้ใช้งานสามารถเรียกใช้บริการ VoIP ที่มีการใช้ SIP ในขณะใดก็ตามที่มีการให้บริการ

4) การตรวจสอบความเป็นของแท้ (Authenticity) เป็นการรับประกันว่าข้อความ SIP ที่ได้รับเป็นข้อความ SIP ที่ส่งมาจากเอนทิตีใดเอนทิตีหนึ่งจริงๆ กล่าวคือ มีการรับประกันความถูกต้องแท้จริง (Genuineness) ของเอนทิตีของ SIP ที่สร้างข้อมูลสำหรับส่งสัญญาณเชื่อมต่อ

5) การห้ามปฏิเสธความรับผิดชอบ (Non-Repudiation) ต้องมีหลักฐานยืนยันที่ไม่สามารถโต้แย้งได้ว่าไม่มีการโทรหรือการตอบสนอง (Response) เกิดขึ้น

วิธีการโดยทั่วไปในการโจมตีหรือสร้างปัญหาให้กับการรักษาความพร้อมใช้ คือการใช้ทรัพยากรที่มีอยู่ให้หมดไปโดยการสร้างการร้องขอจำนวนมากไปยังบริการ VoIP แต่ในแง่การโจมตีสัญญาณเชื่อมต่อของ SIP ผู้โจมตีสามารถก่อให้เกิดการปฏิเสธการให้บริการ (Denial of Service: DoS) โดยอาศัยจุดอ่อนของ SIP คือไม่มีกระบวนการพิสูจน์ตัวตนเมื่อได้รับคำร้องขอสำหรับยกเลิกคำร้องขอที่ดำเนินการอยู่ หรือยกเลิกการเชื่อมต่อ และใช้วิธีการแก้ไขข้อมูลรายละเอียดของเซสชันได้โดยไม่ต้องสร้างการร้องขอจำนวนมากๆ

บริการด้านความปลอดภัยของ SIP มีทั้งรูปแบบ Hop-by-Hop คือมีการรักษาความปลอดภัยระหว่างผู้ใช้กับเซิร์ฟเวอร์ หรือเซิร์ฟเวอร์กับเซิร์ฟเวอร์ และรูปแบบ End-to-End ซึ่งเป็นการรักษาความปลอดภัยระหว่างผู้ใช้ตั้งแต่ต้นทางจนถึงปลายทาง ดังภาพประกอบที่ 3.1



ภาพประกอบที่ 3.1 รูปแบบบริการด้านความปลอดภัยใน SIP (Geneiatakis และคณะ, 2006)

3.3 จุดอ่อนที่ทำให้เกิดการโจมตีสัญญาณเชื่อมต่อของ SIP

ในบทที่ 2 ได้กล่าวถึงปัญหาและวิธีการโจมตีสัญญาณเชื่อมต่อของ SIP รวมถึงงานวิจัยที่เกี่ยวข้องกับแก้ปัญหาการโจมตี จากการวิเคราะห์พบว่าปัจจัยสำคัญที่เป็นจุดอ่อนให้เกิดการโจมตีสัญญาณเชื่อมต่อของ SIP มีดังนี้

1) มาตรฐานของข้อความ SIP: SIP เป็นโพรโทคอลที่มีการส่งข้อมูลในรูปแบบข้อความตัวอักษรที่ไม่มีการเข้ารหัสเพื่อการรักษาความลับ (Confidentiality) และไม่มีการรักษาความถูกต้องสมบูรณ์ (Integrity) ให้กับเฮดเดอร์ฟิลด์และ Message Body โดยเฉพาะอย่างยิ่ง Request-Line รวมถึงเฮดเดอร์ Via, From, To, Call-ID, CSeq และ Contact ทำให้ถูกแก้ไขข้อความ หรือปลอมแปลงข้อความส่งสัญญาณเชื่อมต่อได้

นอกจากนี้ การรักษาความลับยังมีข้อจำกัดเนื่องจากเซิร์ฟเวอร์ที่อยู่ระหว่างกลางจะต้องเข้าถึงข้อความ SIP บางส่วนเพื่อให้ดำเนินการได้อย่างถูกต้อง (เช่น เข้าถึง Request-Line เพื่อตรวจสอบว่าเป็นคำร้องขอสร้างเซสชันหรือยกเลิกเซสชัน) และกำหนดเส้นทางของข้อความไปยังปลายทางได้ (เช่น เข้าถึง Request-URI หรือเฮดเดอร์ Record-Route)

2) การพิสูจน์ตัวตนของ SIP: โดยทั่วไป SIP ใช้กระบวนการ HTTP Digest ในการพิสูจน์ตัวตนคือมีการนำชื่อผู้ใช้และรหัสผ่านมาผ่านการย่อยข้อความเพื่อส่งไปพิสูจน์ตัวตนกับเซิร์ฟเวอร์ ผู้บุกรุกสามารถดักจับข้อมูลนี้เพื่อนำมาคำนวณหารหัสผ่านได้และในระบบที่ไม่มีการป้องกันการนำข้อมูลกลับมาใช้ใหม่ อาจเกิดการโจมตีโดยการนำข้อความ SIP กลับมาใช้ใหม่ (Replay Attack) เช่น นำข้อความ INVITE ที่มีข้อมูลประจำตัวสำหรับใช้พิสูจน์ตัวตนของผู้อื่นมาส่งอีกครั้งเพื่อสร้างการเชื่อมต่อ นอกจากนี้ โพรโทคอล SIP ยังมีข้อกำหนดคือ เมื่อมีการส่งข้อความตอบกลับและข้อความร้องขอบางข้อความ เช่น BYE, CANCEL และ ACK

เป็นต้น เซิร์ฟเวอร์จะยอมรับและดำเนินการกับข้อความนั้นทันทีโดยไม่ต้องร้องขอให้มีการพิสูจน์ตัวตน (Challenge) ทำให้ผู้บุกรุกสามารถปลอมแปลงข้อความเหล่านี้ขึ้นมาเพื่อใช้โจมตีได้

การป้องกันการนำข้อมูลกลับมาใช้ใหม่วิธีการหนึ่งคือ การใช้ค่า nonce ซึ่งเป็นสายอักขระ (String) ที่เซิร์ฟเวอร์สร้างขึ้นอย่างไม่ซ้ำกันในแต่ละครั้งที่มีการร้องขอการพิสูจน์ตัวตน อยู่ในรูปแบบเลขฐาน 16 หรือ base64 (ประกอบด้วยตัวอักษรภาษาอังกฤษตัวพิมพ์ใหญ่ ตัวพิมพ์เล็ก ตัวเลข “+” และ “/”) การใช้ค่า nonce เพื่อป้องกันการโจมตีด้วยวิธีการนำข้อมูลกลับมาส่งใหม่ อาจทำได้ 2 วิธีคือ

- การกำหนดระยะเวลาการใช้งานให้กับค่า nonce

เป็นการกำหนดว่าค่า nonce ที่สร้างขึ้นสามารถใช้งานได้นานเท่าไร ถ้าครบกำหนดแล้วจะไม่สามารถใช้งานได้อีก ค่า nonce จะถูกสร้างขึ้นเมื่อมีการร้องขอการพิสูจน์ตัวตน (Challenge) ดังนั้นอาจกล่าวได้อีกนัยหนึ่งว่า การกำหนดระยะเวลาการใช้งานให้กับค่า nonce คือการกำหนดระยะเวลารอคอยการตอบกลับต่อ Challenge แต่วิธีนี้วิธีเดียวยังไม่เพียงพอต่อการป้องกันการโจมตี เพราะในช่วงที่ยังไม่ครบกำหนดระยะเวลา ผู้บุกรุกสามารถดักจับข้อมูลในเครือข่ายแล้วนำข้อมูลที่ใช้พิสูจน์ตัวตนมาใช้ใหม่ในแพ็กเก็ตอื่นๆ เช่น การลงทะเบียนที่มีข้อมูลติดต่อกลับตามที่ผู้บุกรุกต้องการ หรืออาจใช้สร้างการเชื่อมต่อ

- การสร้างดัชนีให้กับค่า nonce

การสร้างดัชนีสามารถรับประกันได้ว่าจะไม่มีการนำข้อมูลที่ใช้พิสูจน์ตัวตนกลับมาใช้ใหม่เนื่องจากดัชนีที่สร้างขึ้นจะไม่ซ้ำกัน (Unique) ตลอดอายุการใช้งานของ nonce ดังนั้น หากผู้บุกรุกต้องการนำข้อมูลที่ใช้พิสูจน์ตัวตนของผู้อื่นมาใช้ อาจต้องอาศัยวิธีการ Man-in-the-Middle (MITM) เพื่อสกัดกั้น (Intercept) แพ็กเก็ตของเหยื่อ แล้วนำข้อมูลนั้นมาใช้เอง

อย่างไรก็ตาม ในสถาปัตยกรรมที่มีหลายๆ เซิร์ฟเวอร์ใช้ชื่อ Domain Name System (DNS) เดียวกันจะไม่สามารถใช้วิธีการสร้างดัชนีเพื่อป้องกันการโจมตีด้วยวิธีการส่งข้อมูลซ้ำได้

ผู้บุกรุกอาศัยจุดอ่อนเหล่านี้เพื่อโจมตีการทำงานของ SIP ส่งผลให้สามารถเข้าใช้บริการโดยไม่ได้รับอนุญาตหรือทำให้ผู้ใช้บริการอื่นๆ ไม่สามารถใช้บริการได้ ตัวอย่างของการโจมตีและส่วนประกอบของข้อความ SIP ที่เกี่ยวข้องรวมถึงผลที่เกิดจากการโจมตีมีรายละเอียดดังหัวข้อ 3.4

3.4 ตัวอย่างการโจมตีและส่วนประกอบของข้อความ SIP ที่เกี่ยวข้อง

การนำ SIP มาใช้สำหรับส่งสัญญาณเชื่อมต่อในบริการ VoIP หากต้องการให้ระบบสามารถคิดระยะเวลาการโทรได้ เซิร์ฟเวอร์จะต้องมีการทำงานของแบบ Stateful เพื่อติดตามสถานะของเซสชัน (การเชื่อมต่อเพื่อสื่อสารเสียง วิดีโอหรือข้อความตัวอักษร) ว่ามีการสร้างและสิ้นสุดเซสชันเมื่อไร ส่วนการพิสูจน์ตัวตนของผู้ใช้ SIP ใช้กลไก HTTP Digest เพราะง่าย สะดวก รวดเร็วกว่ากลไกอื่น ไคลเอนต์ที่รองรับ SIP ทุกประเภทจึงมีการพัฒนาโปรแกรมที่รองรับการพิสูจน์ตัวตนด้วยวิธีนี้

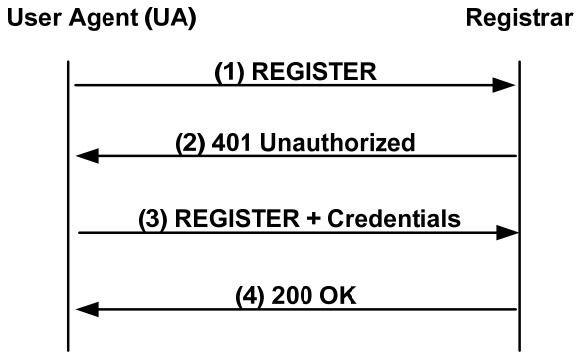
ดังนั้น ตัวอย่างการโจมตีจะอยู่บนพื้นฐานของเซิร์ฟเวอร์ที่มีการทำงานแบบ Stateful และมีการพิสูจน์ตัวตนโดยใช้ HTTP Digest เป็นหลัก โดยสามารถศึกษาข้อมูลส่วนประกอบของข้อความ SIP เกี่ยวกับรูปแบบของเฮดเดอร์ฟิลด์ได้ในบทที่ 2 หัวข้อ 2.3.3 รายละเอียดของ Message Body ที่อยู่ในรูปแบบ SDP ในหัวข้อ 2.3.4 และกลไก HTTP Digest ในหัวข้อ 2.3.7 ตัวอย่างการโจมตีมีรายละเอียดดังนี้

1) Registration Hijacking

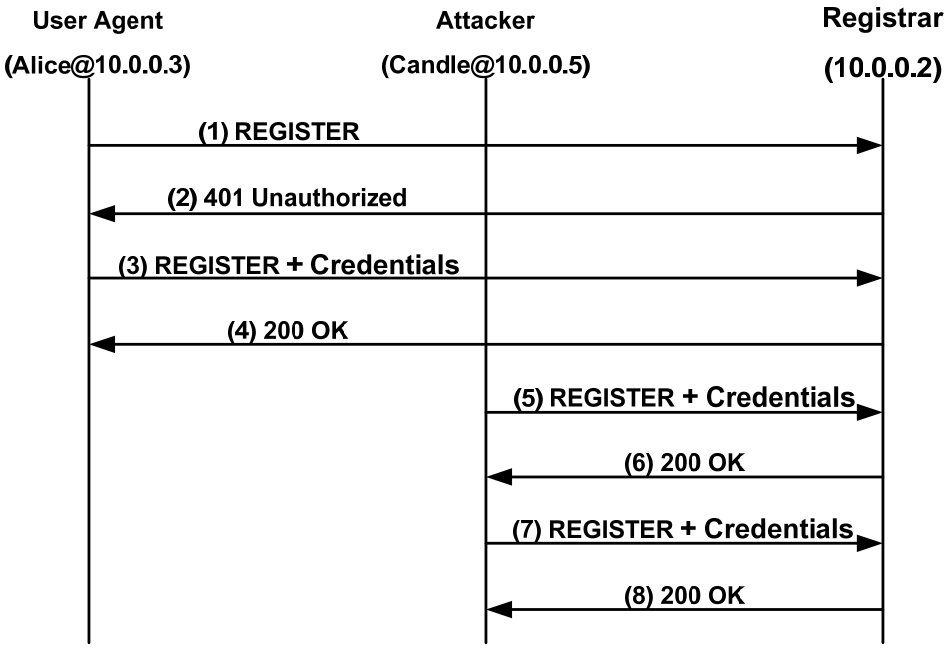
โดยปกติในขั้นตอนการลงทะเบียน UA จะส่งคำร้องขอซึ่งเป็นข้อความ REGISTER ไปยังเซิร์ฟเวอร์ เซิร์ฟเวอร์เรียกร้องให้มีการพิสูจน์ตัวตนโดยการตอบกลับด้วยข้อความ 401 Unauthorized จากนั้น UA ส่งข้อความ REGISTER พร้อมด้วยข้อมูลประจำตัวไปยังเซิร์ฟเวอร์อีกครั้ง เซิร์ฟเวอร์ใช้ข้อมูลประจำตัวนี้ในตรวจสอบตัวตน หากการพิสูจน์ตัวตนถูกต้องจะตอบกลับด้วยข้อความ 200 OK ดังภาพประกอบที่ 3.2

การโจมตีแบบ Registration Hijacking สามารถเกิดขึ้นหลังจากที่มีการลงทะเบียนเรียบร้อยแล้วดังภาพประกอบที่ 3.3 ผู้บุกรุกจะใช้วิธีการดักจับข้อความ (3) REGISTER (รายละเอียดดูภาพประกอบที่ 3.4) แล้วอาศัยจุดอ่อนของ SIP คือข้อความ SIP ไม่มีการเข้ารหัสเพื่อรักษาความลับและไม่มีการรักษาความถูกต้องสมบูรณ์ให้กับข้อความ ทำให้สามารถแก้ไขเฮดเดอร์ CSeq และ Expires (บรรทัดที่ 6 และ 10 ตามลำดับ) เพื่อใช้ยกเลิกการลงทะเบียน ได้เป็นข้อความ (5) REGISTER (รายละเอียดดูภาพประกอบที่ 3.5) ซึ่งในระบบที่ไม่มีการป้องกันการนำข้อมูลกลับมาใช้ใหม่คือ ไม่มีการกำหนดระยะเวลาการใช้งานและการสร้างดัชนีให้กับค่า nonce จะทำให้ผู้บุกรุกยกเลิกการลงทะเบียนสำเร็จได้ จากนั้นผู้บุกรุกจะลงทะเบียนใหม่โดยแก้ไขเฮดเดอร์ CSeq และ Contact (บรรทัดที่ 6 และ 7 ตามลำดับ) มีการเปลี่ยนที่อยู่สำหรับติดต่อกลับให้เป็นตำแหน่งที่อยู่ของผู้บุกรุก ได้ดังข้อความ (7) REGISTER (รายละเอียดดูภาพประกอบที่ 3.6) ผลที่ตามมาคือสายเรียกเข้าที่จะไปยังผู้ใช้ทั้งหมดถูกส่งไป

ยังผู้บุกรุกแทน และเมื่อตรวจสอบข้อมูลของผู้ลงทะเบียนกับเซิร์ฟเวอร์ ข้อมูล Contact จะเปลี่ยนไปเป็นตำแหน่งที่อยู่ของผู้บุกรุกดังภาพประกอบที่ 3.7



ภาพประกอบที่ 3.2 กระบวนการลงทะเบียน (Geneiatakis และคณะ, 2006)



ภาพประกอบที่ 3.3 ตัวอย่าง Registration Hijacking

```
1. REGISTER sip:sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1689-1-2
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:alice@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1689@10.0.0.3
6. CSeq: 2 REGISTER
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Authorization: Digest
        username="alice",realm="sipserver.cs.psu.ac.th",uri="sip:10.0.0.2:5060",nonce="50b62
        9e3fe1bb61807b447f0752a86042f25c5f2",response="91464ed926f16fe300e7807df16
        409eb",algorithm=MD5
9. Max-Forwards: 70
10. Expires: 1800
11. User-Agent: SIPp/Linux
12. Content-Length: 0
```

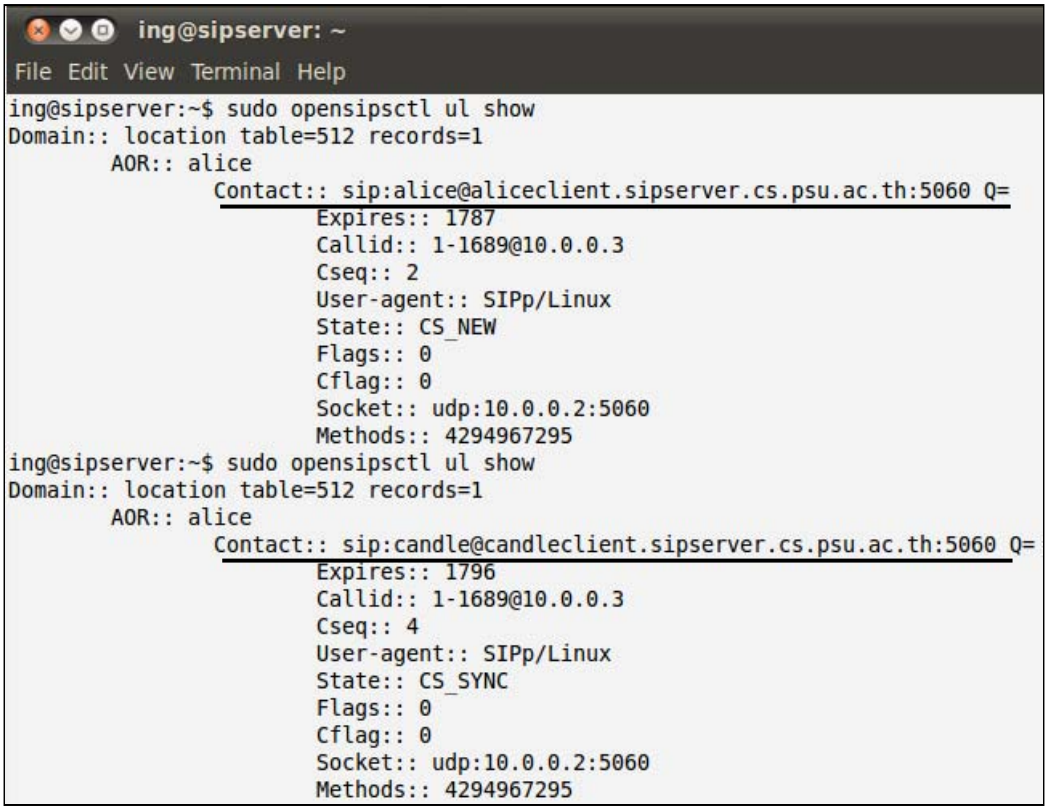
ภาพประกอบที่ 3.4 (3) REGISTER Message

```
1. REGISTER sip:sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1689-1-2
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:alice@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1689@10.0.0.3
6. CSeq: 3 REGISTER
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Authorization: Digest
        username="alice",realm="sipserver.cs.psu.ac.th",uri="sip:10.0.0.2:5060",nonce="50b62
        9e3fe1bb61807b447f0752a86042f25c5f2",response="91464ed926f16fe300e7807df16
        409eb",algorithm=MD5
9. Max-Forwards: 70
10. Expires: 0
11. User-Agent: SIPp/Linux
12. Content-Length: 0
```

ภาพประกอบที่ 3.5 Registration Hijacking: (5) REGISTER
(ยกเลิกการลงทะเบียน จากภาพประกอบที่ 3.4)

1. REGISTER sip:sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1689-1-2
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:alice@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1689@10.0.0.3
6. **CSeq: 4 REGISTER**
7. **Contact: sip:candle@candleclient.sipserver.cs.psu.ac.th:5060**
8. Authorization: Digest username="alice",realm="sipserver.cs.psu.ac.th",uri="sip:10.0.0.2:5060", nonce="50b629e3fe1bb61807b447f0752a86042f25c5f2",response="91464ed926f16fe300e7807df16409eb",algorithm=MD5
9. Max-Forwards: 70
10. Expires: 1800
11. User-Agent: SIPp/Linux
12. Content-Length: 0

ภาพประกอบที่ 3.6 Registration Hijacking: (7) REGISTER
(ลงทะเบียนใหม่ภายใต้ชื่อ candle)

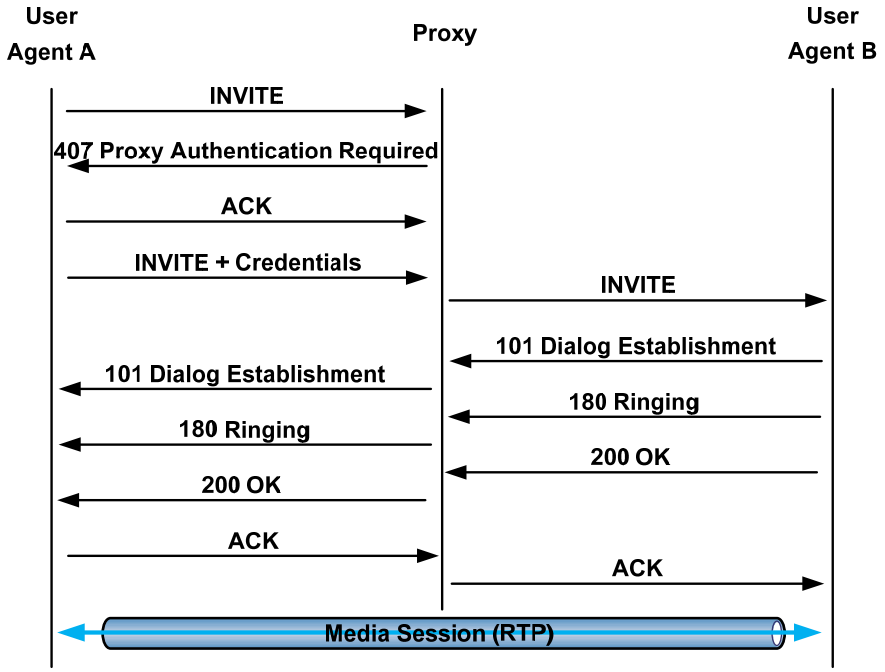


ภาพประกอบที่ 3.7 การตรวจสอบรายชื่อผู้ใช้ที่ลงทะเบียนเข้าสู่ระบบ

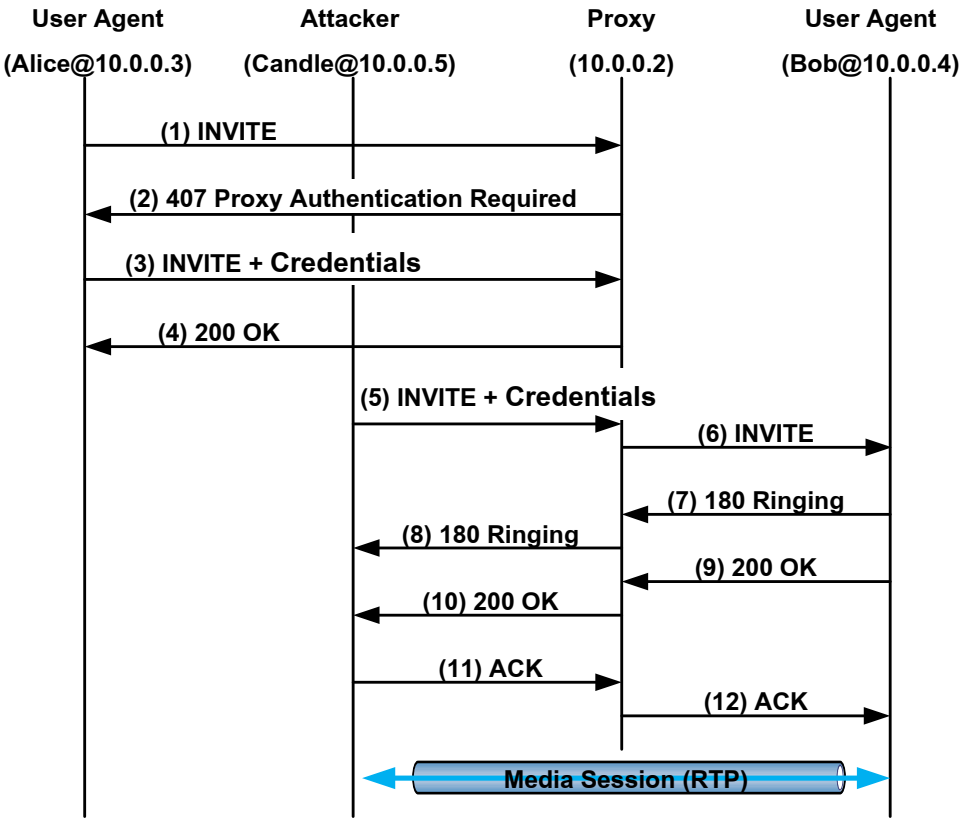
2) Invite Replay Billing Attack

โดยปกติในขั้นตอนการเชื่อมต่อ UAC จะส่งคำร้องขอซึ่งเป็นข้อความ INVITE ไปยังเซิร์ฟเวอร์ เซิร์ฟเวอร์เรียกร้องให้มีการพิสูจน์ตัวตนโดยการตอบกลับด้วยข้อความ 407 Proxy Authentication Required จากนั้น UAC ส่งข้อความ INVITE พร้อมด้วยข้อมูลประจำตัวไปยังเซิร์ฟเวอร์อีกครั้ง เซิร์ฟเวอร์ใช้ข้อมูลประจำตัวนี้ในตรวจสอบตัวตน หากการพิสูจน์ตัวตนถูกต้องจะส่งต่อ INVITE ไปยัง UAS หาก UAS ยอมรับการเชื่อมต่อจะตอบกลับด้วยข้อความ 200 OK ดังภาพประกอบที่ 3.8

การโจมตีแบบ Invite Replay Billing Attack สามารถเกิดขึ้นหลังจาก UAC ส่งข้อความ INVITE พร้อมด้วยข้อมูลประจำตัวยังเซิร์ฟเวอร์แล้วดังภาพประกอบที่ 3.9 ผู้บุกรุกอาศัยจุดอ่อนของ SIP คือข้อความ SIP ไม่มีการเข้ารหัสเพื่อรักษาความลับและไม่มีการรักษาความถูกต้องสมบูรณ์ให้กับข้อความ และอาศัยจุดอ่อนของระบบที่ไม่มีการป้องกันการนำข้อมูลกลับมาใช้ใหม่เช่นเดียวกับการโจมตีแบบ Registration Hijacking โดยเริ่มจากการดักจับข้อความ (3) INVITE (รายละเอียดดูภาพประกอบที่ 3.10) แล้วแก้ไข Request-URI (บรรทัดที่ 1) เพื่อระบุที่อยู่ของผู้รับปลายทางตามที่ต้องการ เพราะเซิร์ฟเวอร์จะใช้ Request-URI ในการกำหนดผู้รับปลายทาง (การแก้ไขเฉพาะเฮดเดอร์ To ไม่สามารถเปลี่ยนแปลงผู้รับปลายทางได้) รวมทั้งแก้ไข CSeq (บรรทัดที่ 6) และหมายเลขไอพีในส่วนข้อมูลรายละเอียดของเซสชันให้เป็นของผู้บุกรุก (บรรทัดที่ 13 และ 16) ข้อความใหม่ที่ได้คือ (5) INVITE (รายละเอียดดูภาพประกอบที่ 3.11) จากนั้นส่งข้อความนี้ไปยังเซิร์ฟเวอร์ เมื่อเซิร์ฟเวอร์พิสูจน์ตัวตนแล้วว่าถูกต้องจะส่งต่อข้อความไปยังปลายทางตามที่ระบุ ผลที่ตามมาคือผู้บุกรุกสามารถเชื่อมต่อได้โดยไม่ต้องเสียค่าบริการ เพราะเมื่อตรวจสอบบันทึกข้อมูลการโทรในฐานข้อมูลของเซิร์ฟเวอร์ชื่อผู้โทรจะไม่ใช่ผู้บุกรุกแต่เป็นชื่อ UAC ทำให้ UAC ต้องรับผิดชอบต่อค่าบริการที่เกิดขึ้น ดังภาพประกอบที่ 3.12



ภาพประกอบที่ 3.8 การเชื่อมต่อผู้ใช้ในเหตุการณ์ปกติ



ภาพประกอบที่ 3.9 ตัวอย่าง Invite Replay Billing Attack

```
1. INVITE sip:dan@sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1898-1-3
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:dan@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1898@10.0.0.3
6. CSeq: 2 INVITE
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Proxy-Authorization: Digest
        username="alice",realm="sipserver.cs.psu.ac.th",uri="sip:10.0.0.2:5060",nonce="50d4a
        a981dc9c857c19f0f389d6598186e3fa53d",response="f9613ee1833024edaad1c0b3a6
        794688",algorithm=MD5
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length:   195

12. v=0
13. o=alice 53655765 2353687637 IN IP4 10.0.0.3
14. s=A conversation
15. t=0 0
16. c=IN IP4 10.0.0.3
17. m=audio 6000 RTP/AVP 8
18. a=rtpmap:8 PCMA/8000
19. a=rtpmap:101 telephone-event/8000
20. a=fmtp:101 0-11,16
```

ภาพประกอบที่ 3.10 (3) INVITE Message

```
1. INVITE sip:bob@sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1898-1-3
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:dan@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1898@10.0.0.3
6. CSeq: 3 INVITE
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Proxy-Authorization: Digest
        username="alice",realm="sipserver.cs.psu.ac.th",uri="sip:10.0.0.2:5060",nonce="50d4a
        a981dc9c857c19f0f389d6598186e3fa53d",response="f9613ee1833024edaad1c0b3a6
        794688",algorithm=MD5
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length:   195

12. v=0
13. o=alice 53655765 2353687637 IN IP4 10.0.0.5
14. s=A conversation
15. t=0 0
16. c=IN IP4 10.0.0.5
17. m=audio 6000 RTP/AVP 8
18. a=rtpmap:8 PCMA/8000
19. a=rtpmap:101 telephone-event/8000
20. a=fmtp:101 0-11,16
```

ภาพประกอบที่ 3.11 Invite Replay Billing Attack: (5) INVITE

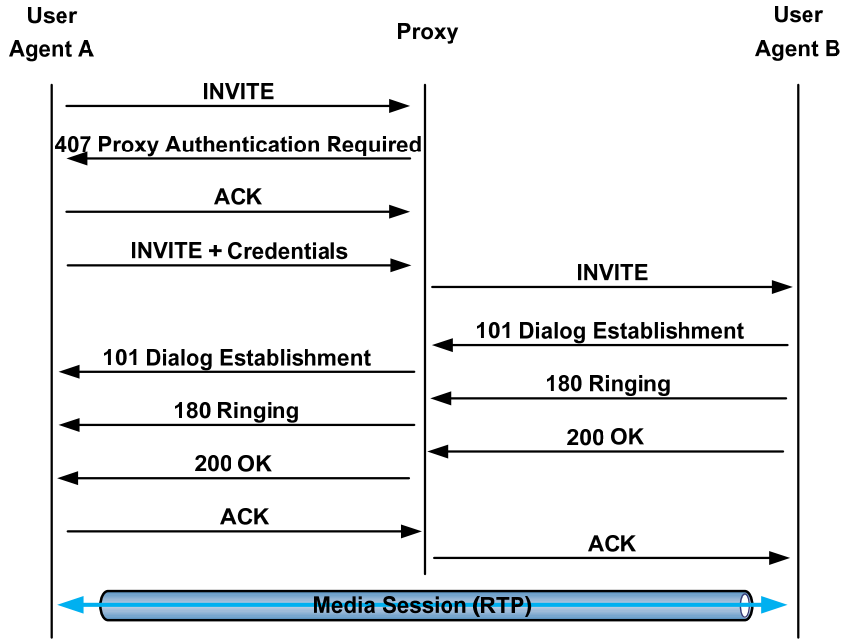
CDR ID	Sip Call ID	Call Start Time	Caller	Callee	Duration
1333728	1-1898@10.0.0.3	2012-12-22 01:28:25	sip:alice@sipserver.cs.psu.ac.th:5060	sip:dan@sipserver.cs.psu.ac.th:5060	2
1333727	1-1898@10.0.0.3	2012-12-22 01:27:44	sip:alice@sipserver.cs.psu.ac.th:5060	sip:dan@sipserver.cs.psu.ac.th:5060	86

ภาพประกอบที่ 3.12 บันทึกข้อมูลการโทร

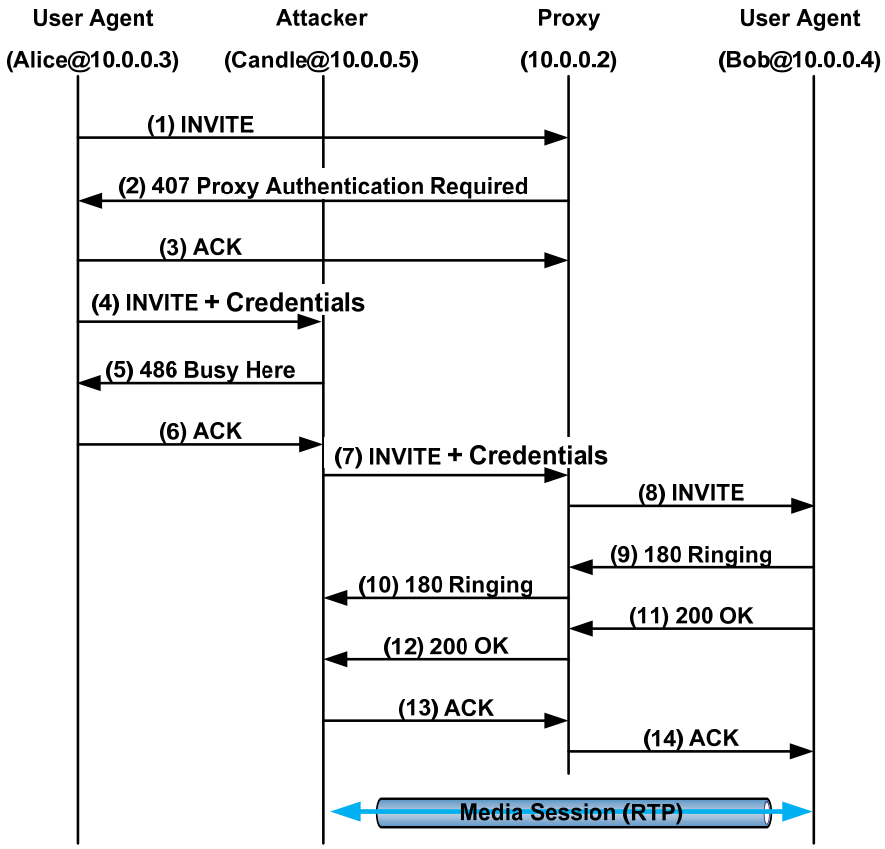
3) Call Establishment Hijacking

ตามที่ได้กล่าวมาแล้วคือ โดยปกติในขั้นตอนการเชื่อมต่อ เมื่อ UAC ส่งคำร้องขอ INVITE ไปยังเซิร์ฟเวอร์ แล้วเซิร์ฟเวอร์เรียกร้องให้มีการพิสูจน์ตัวตน UAC จะส่งข้อความ INVITE พร้อมด้วยข้อมูลประจำตัวไปยังเซิร์ฟเวอร์อีกครั้งเพื่อพิสูจน์ตัวตน ดังภาพประกอบที่ 3.13

ผู้บุกรุกสามารถโจมตีแบบ Call Establishment Hijacking โดยแทรกกลางการสื่อสารระหว่างเหยื่อและเซิร์ฟเวอร์ ทำให้ผู้บุกรุกสามารถสกัดกั้น (Intercept) แพ็กเก็ตที่เหยื่อส่งไปยังเซิร์ฟเวอร์และปลอมแปลงแพ็กเก็ตเพื่อส่งไปยังเหยื่อได้ดังภาพประกอบที่ 3.14 ผู้บุกรุกอาจใช้วิธีการโจมตีแบบ ARP Poisoning คือการจับคู่หมายเลข MAC (Media Access Control Address) ของตนเองกับหมายเลข IP ของ UAC เพื่อให้ข้อความที่ UAC ส่งไปยังเซิร์ฟเวอร์ทั้งหมดส่งมายังผู้บุกรุกแทน ทำให้ผู้บุกรุกสามารถเปลี่ยนแปลงแก้ไขข้อความเหล่านั้นก่อนส่งต่อไปยังเซิร์ฟเวอร์หรือละทิ้งข้อความเพื่อไม่ให้ส่งไปยังเซิร์ฟเวอร์ก็ได้ ซึ่งจากตัวอย่างการโจมตีแบบ Call Establishment Hijacking นี้ ผู้บุกรุกจะสกัดกั้นข้อความที่มีข้อมูลประจำตัวสำหรับใช้พิสูจน์ตัวตนคือ (4) INVITE ของ UAC เพื่อไม่ให้ส่งไปยังเซิร์ฟเวอร์แล้วส่งข้อความ (5) 486 Busy Here (รายละเอียดดูภาพประกอบที่ 3.15) เพื่อบอก UAC ว่า UAS ไม่ว่าง ทำให้ UAC ยกเลิกการเชื่อมต่อ การปลอมแปลงข้อความนี้สามารถทำได้โดยการคัดลอกเฮดเดอร์ Via, From, To, Call-ID และ CSeq จากข้อความ (4) INVITE จากนั้นอาศัยจุดอ่อนของ SIP คือข้อความ SIP ไม่มีการเข้ารหัสเพื่อรักษาความลับและไม่มีการรักษาความถูกต้องสมบูรณ์ให้กับข้อความ ทำให้สามารถแก้ไขข้อความ (4) INVITE ให้เป็นข้อความ (7) INVITE (รายละเอียดดูภาพประกอบที่ 3.16) ได้โดยมีการแก้ไข Request-URI (บรรทัดที่ 1) ให้เป็นปลายทางที่ต้องการ รวมทั้งแก้หมายเลขไอพีในส่วนข้อมูลรายละเอียดของเซสชันให้เป็นหมายเลขไอพีของผู้บุกรุก (บรรทัดที่ 13 และ 16) แล้วส่งไปยังเซิร์ฟเวอร์ ผลที่ตามมามีลักษณะเดียวกันกับการโจมตีแบบ Call Establishment Hijacking คือผู้บุกรุกสามารถเชื่อมต่อได้โดยไม่ต้องเสียค่าบริการ แต่ UAC ต้องรับผิดชอบต่อค่าบริการที่เกิดขึ้นแทน และยังทำให้ UAC ไม่ได้รับบริการจากเซิร์ฟเวอร์ตามที่ต้องการด้วย



ภาพประกอบที่ 3.13 การเชื่อมต่อผู้ใช้ในเหตุการณ์ปกติ



ภาพประกอบที่ 3.14 ตัวอย่าง Call Establishment Hijacking

```
SIP/2.0 486 Busy Here
Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1897-1-3
From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
To: <sip:dan@sipserver.cs.psu.ac.th:5060>
Call-ID: 1-1897@10.0.0.3
CSeq: 2 INVITE
Content-Length: 0
```

ภาพประกอบที่ 3.15 Call Establishment Hijacking: (5) 486 Busy Here

```
1. INVITE sip:bob@sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1897-1-3
3. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
4. To: <sip:dan@sipserver.cs.psu.ac.th:5060>
5. Call-ID: 1-1897@10.0.0.3
6. CSeq: 2 INVITE
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Proxy-Authorization: Digest
        username="alice",realm="sipserver.cs.psu.ac.th",uri="sip:10.0.0.2:5060",nonce="50b8d
        ba8ad61017b8ae9b802f59b203b4536df9e",response="16b2f4450ef02dcd0f6ebf15d07
        822a9",algorithm=MD5
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length:   195

12. v=0
13. o=alice 53655765 2353687637 IN IP4 10.0.0.5
14. s=A conversation
15. t=0 0
16. c=IN IP4 10.0.0.5
17. m=audio 6000 RTP/AVP 8
18. a=rtpmap:8 PCMA/8000
19. a=rtpmap:101 telephone-event/8000
20. a=fmtp:101 0-11,16
```

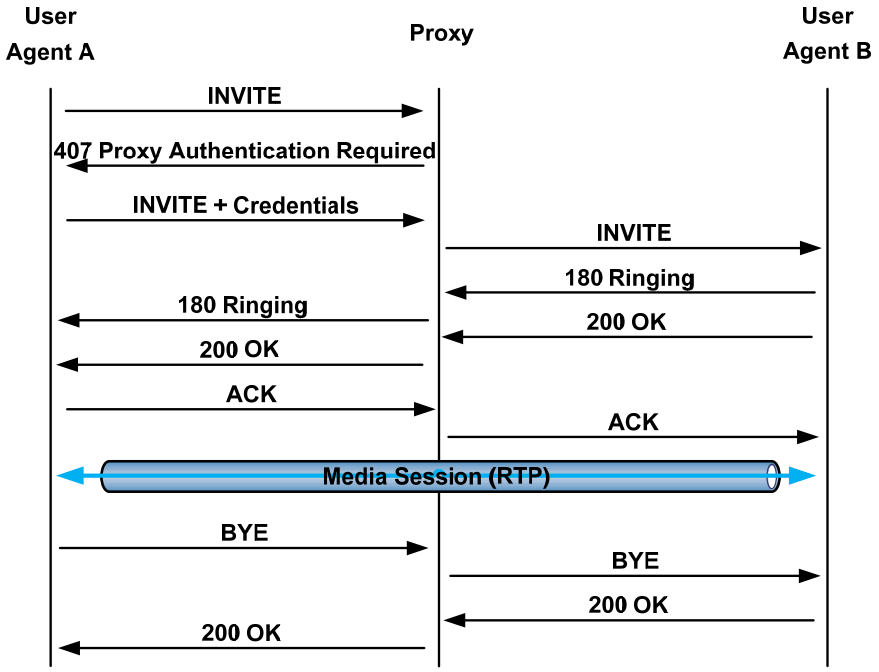
ภาพประกอบที่ 3.16 Call Establishment Hijacking: (7) INVITE

4) Call Termination Hijacking

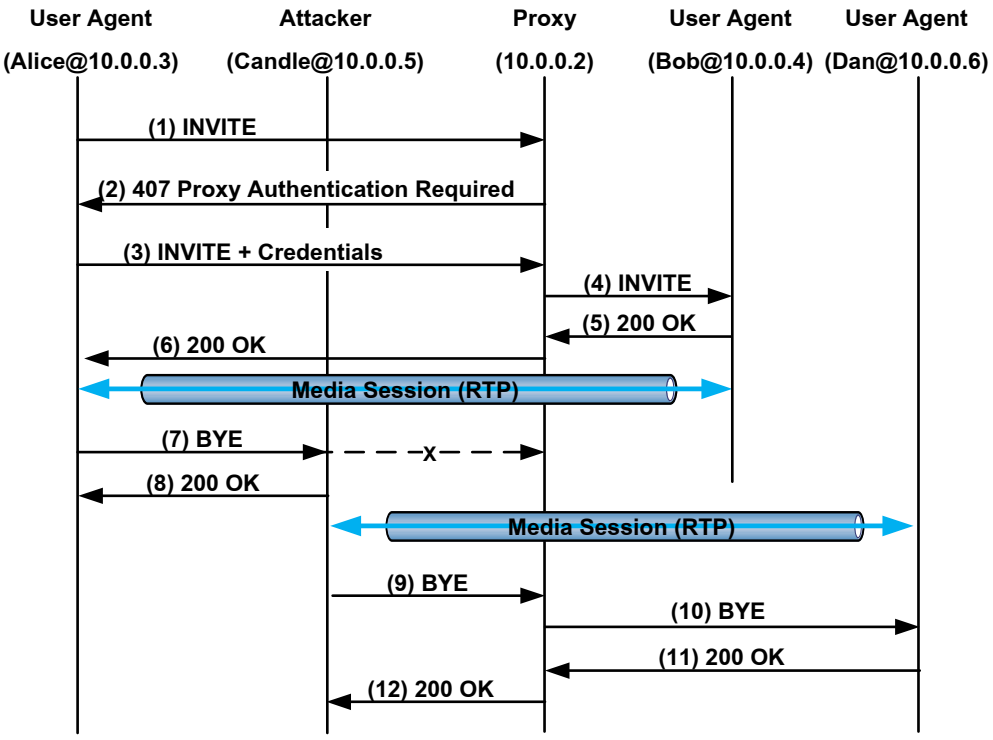
โดยปกติ UAC สามารถยกเลิกการเชื่อมต่อที่เกิดขึ้นได้โดยส่งคำร้องขอที่เป็นข้อความ BYE ไปยัง UAS ดังภาพประกอบที่ 3.17

การโจมตีแบบ Call Termination Hijacking ผู้บุกรุกอาจใช้วิธีการโจมตี ARP Poisoning เช่นเดียวกับการโจมตีแบบ Call Establishment Hijacking เพื่อขัดขวางไม่ให้ UAC สามารถส่งข้อความ BYE ไปยังเซิร์ฟเวอร์ได้และสร้างข้อความ 200 OK ส่งกลับไปเพื่อให้ UAC คิดว่าสามารถยกเลิกเซสชันได้สำเร็จแล้ว ดังภาพประกอบที่ 3.18 การสร้างข้อความ 200 OK นี้ ผู้บุกรุกอาศัยจุดอ่อนของ SIP ที่ไม่มีการเข้ารหัสเพื่อรักษาความลับให้กับข้อความ ทำให้สามารถคัดลอกเฮดเดอร์ Via, From, To, Call-ID และ CSeq จากข้อความ BYE มาสร้างข้อความ 200 OK ได้ และจากจุดอ่อนที่ไม่มีการพิสูจน์ตัวตนเมื่อส่งข้อความ BYE ทำให้ผู้บุกรุกสามารถส่งข้อความ BYE ไปยังเซิร์ฟเวอร์เพื่อขอยกเลิกการเชื่อมต่อได้ นอกจากนี้ ผู้บุกรุกมีการปลอมแพ็กเก็ต RTP โดยอาศัยข้อมูล หมายเลขลำดับ (Sequence Number) ประทับเวลา (Time Stamp) และ Synchronization Source Identifier ของแพ็กเก็ต RTP ที่ดักจับได้ เนื่องจากแพ็กเก็ตเหล่านี้ไม่ถูกเข้ารหัส ผลที่ตามมาคือผู้บุกรุกสามารถเชื่อมต่อได้โดยไม่ต้องเสียค่าบริการ

การป้องกัน Call Termination Hijacking อาจใช้วิธีการเข้ารหัสแพ็กเก็ต RTP หรืออาจมีการตรวจสอบหมายเลขไอพีต้นทางและหมายเลขไอพีปลายทางของแพ็กเก็ต RTP ที่เกตเวย์ว่าตรงกับหมายเลขไอพีที่ระบุในขั้นตอนของการสร้างการเชื่อมต่อหรือไม่



ภาพประกอบที่ 3.17 การเชื่อมต่อผู้ใช้ในเหตุการณ์ปกติ



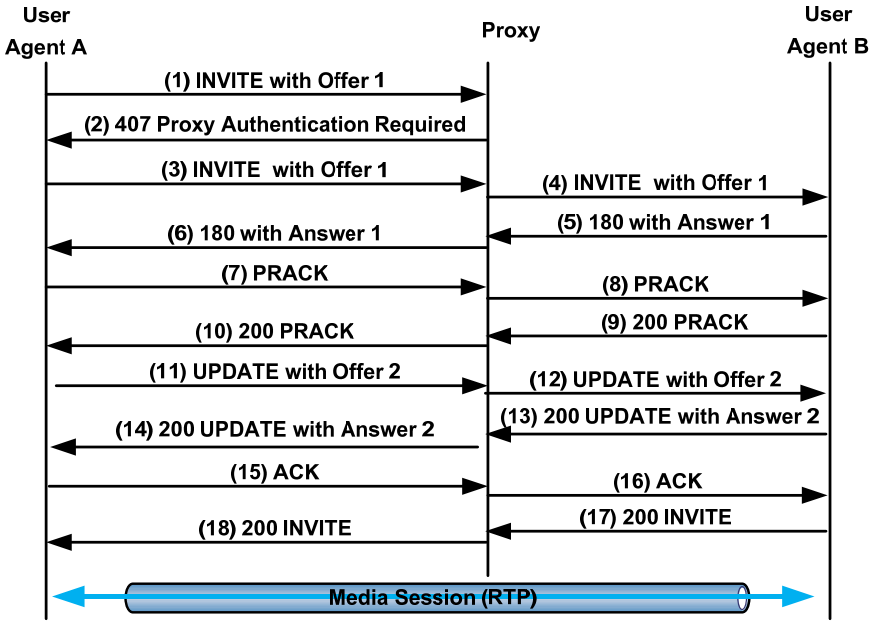
ภาพประกอบที่ 3.18 ตัวอย่าง Call Termination Hijacking

5) UPDATE Attack

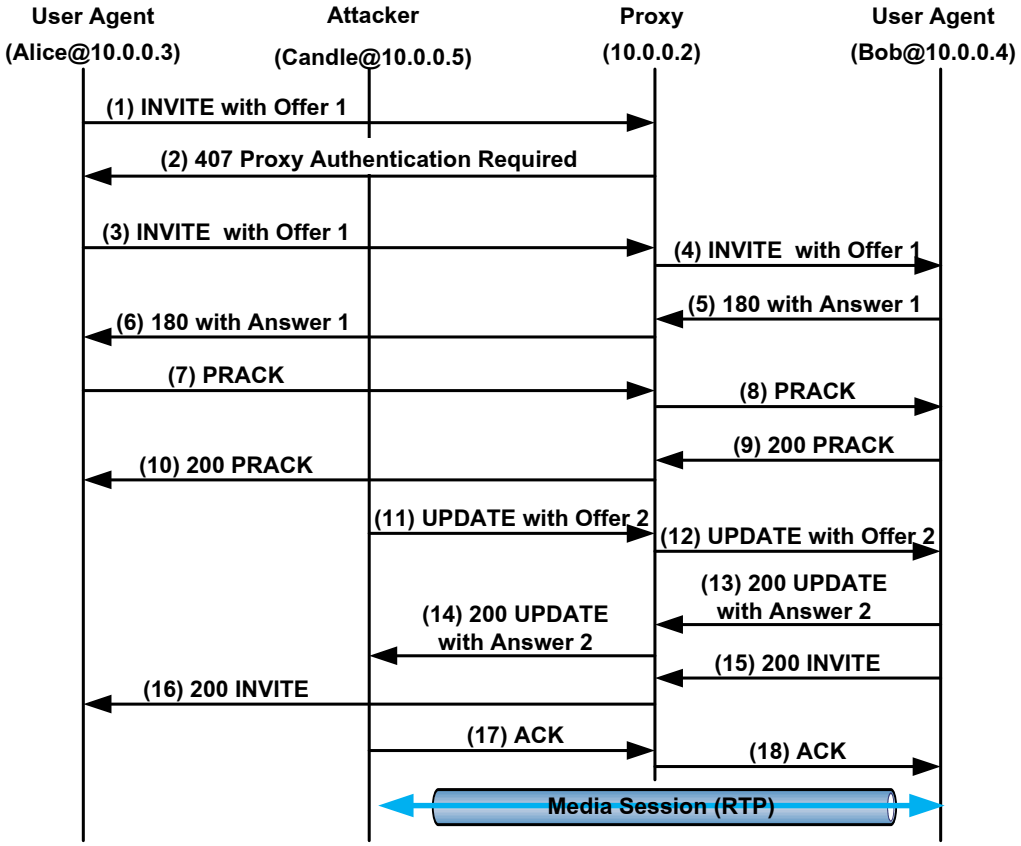
คำร้องขอ UPDATE ถูกนิยามไว้ใน RFC 3311 (Rosenberg, 2002) ใช้สำหรับเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชันก่อนที่การเชื่อมต่อจะเสร็จสมบูรณ์ ดังนั้นข้อความ UPDATE สามารถประกอบด้วยข้อมูลรายละเอียดของเซสชันที่อยู่ในรูปแบบ SDP เช่นเดียวกันกับข้อความ INVITE

การเปลี่ยนแปลงรายละเอียดของเซสชันมีกระบวนการตามภาพประกอบที่ 3.19 ซึ่ง Offer และ Answer คือข้อมูลรายละเอียดของเซสชันที่อยู่ในรูปแบบ SDP และ Provisional Response ACKnowledgement (PRACK) ตามที่ได้นิยามไว้ใน RFC 3262 (Rosenberg และ Schulzrinne, 2002) คือคำร้องขอที่ใช้สำหรับยืนยันว่า UAC ได้รับคำตอบกลับแล้วคล้ายกับ ACK แต่ต่างกันตรงที่ PRACK ใช้หลังจากได้รับคำตอบกลับที่ใช้บอกความก้าวหน้าของการประมวลผลคำร้องขอ (คำตอบกลับประเภท 1XX) เท่านั้น การเปลี่ยนแปลงรายละเอียดของเซสชันเริ่มจาก UAC ส่งคำร้องขอที่เป็นข้อความ INVITE ไปยัง UAS จากนั้น UAS ตอบกลับด้วยข้อความ 180 (Ringing) เพื่อบอกว่ากำลังแจ้งเตือนผู้รับว่ามีการร้องขอการเชื่อมต่อเข้ามา UAC จะส่งข้อความ PRACK ยืนยันว่าได้รับคำตอบกลับ 180 (Ringing) แล้ว ในระหว่างนี้ถ้า UAC ต้องการเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชัน เช่น หมายเลขไอพี พอร์ต ประเภทของข้อมูลที่ใช้รับส่งกัน ก็จะส่งคำร้องขอ UPDATE พร้อมด้วยข้อมูลรายละเอียดของเซสชันใหม่ที่ต้องการไปยัง UAS จากนั้น UAS จะตอบกลับด้วยข้อความ 200 UPDATE เพื่อบอกว่าได้ปรับปรุงข้อมูลรายละเอียดของเซสชันแล้วพร้อมทั้งให้ข้อมูลรายละเอียดของเซสชันฝั่ง UAS ด้วย

การโจมตีโดยใช้คำร้องขอ UPDATE อาจเกิดขึ้นได้ตามตัวอย่างภาพประกอบที่ 3.20 ผู้บุกรุกจะดักจับข้อความ (7) PRACK (รายละเอียดดูภาพประกอบที่ 3.21) แล้วอาศัยจุดอ่อนที่ข้อความ SIP ไม่กระบวนการรักษาความลับ ทำให้ผู้บุกรุกสามารถใช้ข้อมูล Request-URI, Via, Route, From, To, Call-ID และ CSeq (บรรทัดที่ 1-6 และ 8 ตามลำดับ) จากข้อความนี้ในการสร้างข้อความ (11) UPDATE (รายละเอียดดูภาพประกอบที่ 3.22) นอกจากนี้มีการใช้หมายเลขไอพีของผู้บุกรุกในข้อมูลรายละเอียดของเซสชัน เนื่องจากทั้งเซิร์ฟเวอร์และ UAS ไม่มีกระบวนการพิสูจน์ตัวตนเมื่อได้รับคำร้องขอ UPDATE ทำให้เมื่อ UAS ได้รับข้อความ UPDATE นี้ จะมีการปรับปรุงข้อมูลรายละเอียดของเซสชัน ส่งผลให้ข้อมูลเสียง วิดีโอ หรือข้อความตัวอักษรถูกส่งไปยังผู้บุกรุกแทนที่จะส่งไปยัง UAC



ภาพประกอบที่ 3.19 การเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชันโดยใช้คำร้องขอ UPDATE



ภาพประกอบที่ 3.20 ตัวอย่าง UPDATE Attack

1. PRACK **sip:bob@bobclient.sipserver.cs.psu.ac.th:5060** SIP/2.0
2. **Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2105-1-7**
3. **Route: <sip:10.0.0.2;lr=on;did=cd.40395c44>**
4. **From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1**
5. **To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=1**
6. **Call-ID: 1-2105@10.0.0.3**
7. RAck: 1 2 INVITE
8. **CSeq: 3 PRACK**
9. Max-Forwards: 70
10. Content-Length: 0

ภาพประกอบที่ 3.21 UPDATE Attack: (7) PRACK

1. UPDATE **sip:bob@bobclient.sipserver.cs.psu.ac.th:5060** SIP/2.0
2. **Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2105-1-7**
3. **Route: <sip:10.0.0.2;lr=on;did=cd.40395c44>**
4. **From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1**
5. **To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=1**
6. **Call-ID: 1-2105@10.0.0.3**
7. **CSeq: 4 UPDATE**
8. Contact: sip:candle@candleclient.sipserver.cs.psu.ac.th:5060
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length: 141

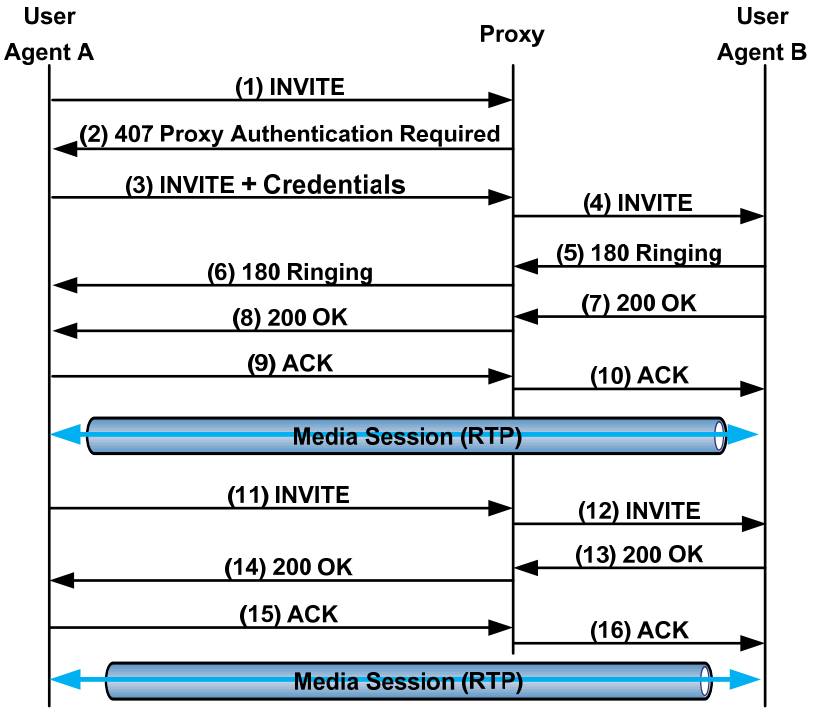
12. v=0
13. o=candle 53655765 2353687637 IN IP4 **10.0.0.5**
14. s=A conversation
15. c=IN IP4 **10.0.0.5**
16. t=0 0
17. m=audio 6000 RTP/AVP 0
18. a=rtpmap:0 PCMU/8000

ภาพประกอบที่ 3.22 UPDATE Attack: (11) UPDATE

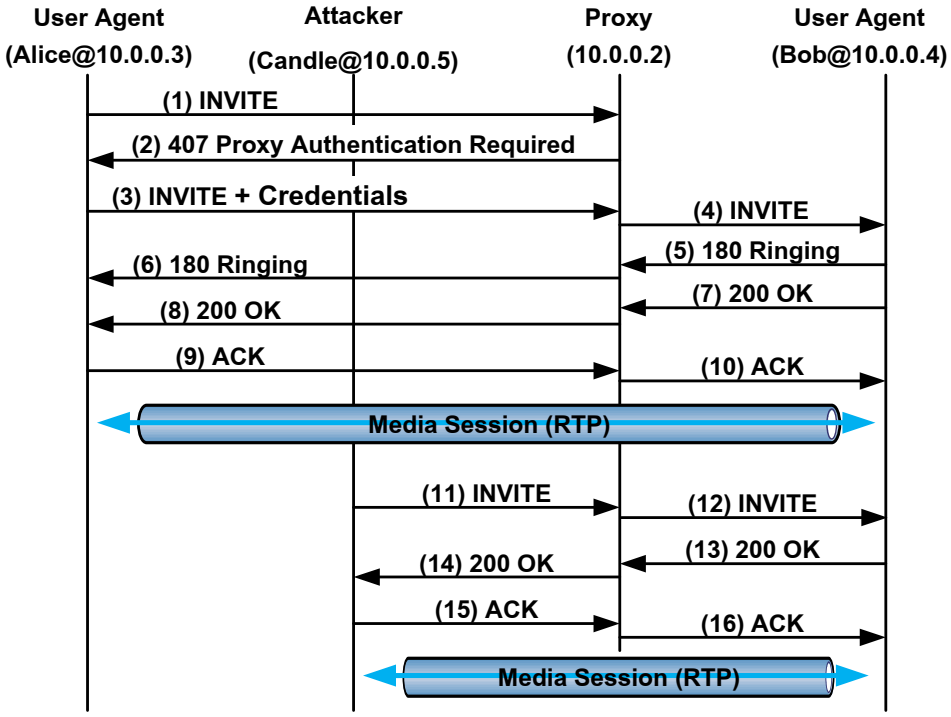
6) Re-INVITE Attack

คำร้องขอ Re-INVITE คือคำร้องขอ INVITE ที่มีการส่งอีกครั้งหลังจากการเชื่อมต่อสำเร็จแล้วเพื่อเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชัน ดังภาพประกอบที่ 3.23 คำร้องขอ Re-INVITE ได้แก่ข้อความ (11) INVITE

เมื่อสร้างการเชื่อมต่อเรียบร้อยแล้ว การโจมตีแบบ Re-INVITE Attack เกิดขึ้นได้ดังภาพประกอบที่ 3.24 ผู้บุกรุกอาศัยจุดอ่อนที่ข้อความ SIP ไม่กระบวนการรักษาความลับ ผู้บุกรุกจึงสามารถเข้าถึงข้อมูล Request-URI, Via, Route, From, To, Call-ID และ CSeq ในข้อความ (9) ACK (รายละเอียดดูภาพประกอบที่ 3.25) เพื่อนำมาสร้างข้อความ (11) INVITE ได้ (รายละเอียดดูภาพประกอบที่ 3.26) สังเกตเฮดเดอร์ To เมื่อมีการเชื่อมต่อแล้วจะต้องมีข้อมูล tag (บรรทัดที่ 5) และทั้งแก้ไขข้อมูลรายละเอียดของเซสชัน เช่น หมายเลขไอพี (บรรทัดที่ 13 และ 15) เพื่อส่งไปยัง UAS โดยอาศัยจุดอ่อนที่เซิร์ฟเวอร์และ UAS ไม่มีกระบวนการพิสูจน์ตัวตนเมื่อได้รับคำร้องขอ Re-INVITE ส่งผลให้ผู้บุกรุกสามารถแก้ไขข้อมูลรายละเอียดของเซสชันได้สำเร็จ ข้อมูลเสียง วิดีโอ หรือข้อความตัวอักษรจะถูกส่งไปยังผู้บุกรุกแทนที่จะส่งไปยัง UAC



ภาพประกอบที่ 3.23 การเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชันโดยใช้คำร้องขอ INVITE



ภาพประกอบที่ **3.24** Re-INVITE Attack

1. ACK **sip:bob@bobclient.sipserver.cs.psu.ac.th:5060** SIP/2.0
2. **Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2356-1-8**
3. **Route: <sip:10.0.0.2;lr=on;did=d47.b08c4da7>**
4. **From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1**
5. **To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=2**
6. **Call-ID: 1-2356@10.0.0.3**
7. **CSeq: 2 ACK**
8. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
9. Max-Forwards: 70
10. Content-Length: 0

ภาพประกอบที่ **3.25** Re-INVITE Attack: (9) ACK

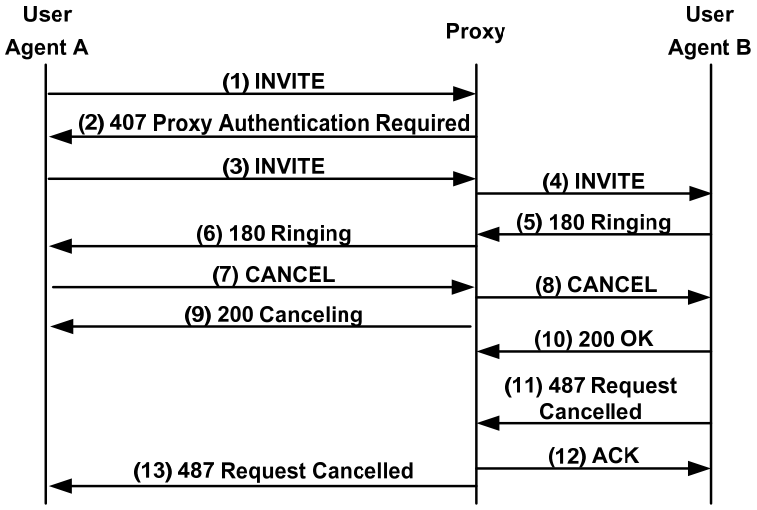
```
1. INVITE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2356-1-8
3. Route: <sip:10.0.0.2;lr=on;did=d47.b08c4da7>
4. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
5. To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=2
6. Call-ID: 1-2356@10.0.0.3
7. CSeq: 3 INVITE
8. Contact: sip:candle@candleclient.sipserver.cs.psu.ac.th:5060
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length:   141

12. v=0
13. o=candle 53655765 2353687637 IN IP4 10.0.0.5
14. s=A conversation
15. c=IN IP4 10.0.0.5
16. t=0 0
17. m=audio 6000 RTP/AVP 0
18. a=rtpmap:0 PCMU/8000
```

ภาพประกอบที่ 3.26 Re-INVITE Attack: (11) INVITE

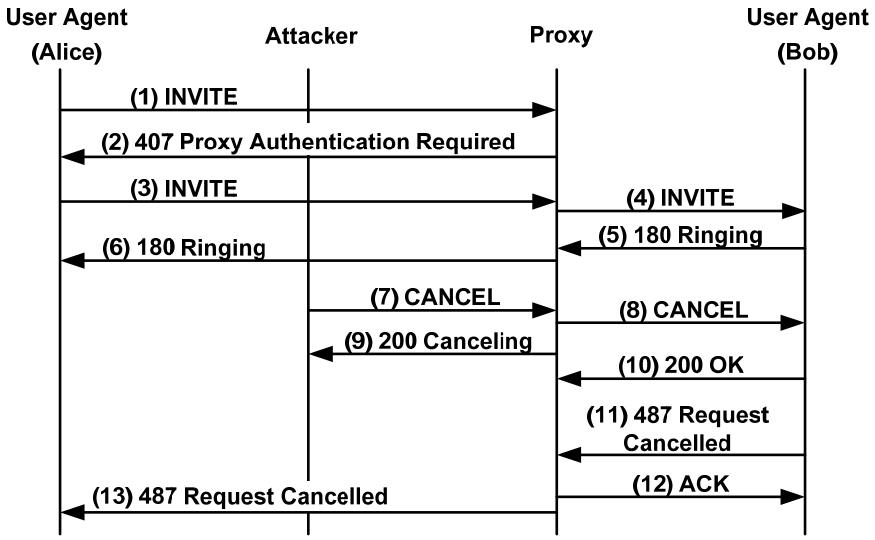
7) CANCEL Attack

คำร้องขอ CANCEL ใช้สำหรับยกเลิกคำร้องขอที่กำลังดำเนินการอยู่ เช่น การยกเลิกคำร้องขอ INVITE จะต้องดำเนินการก่อนที่จะมีการเชื่อมต่อเสร็จสิ้น (ก่อน UAS ส่งข้อความ 200 OK) ดังภาพประกอบที่ 3.27 ซึ่งพร็อกซีที่มีการทำงานแบบ Stateful จะไม่ส่งต่อคำร้องขอ CANCEL แต่จะสร้างคำร้องขอขึ้นมาใหม่เพื่อส่งไปยัง UAS

ตัวอย่างการโจมตีโดยใช้คำร้องขอ CANCEL เพื่อขอยกเลิกการเชื่อมต่อแสดงดังภาพประกอบที่ 3.28 ผู้บุกรุกอาศัยจุดอ่อนที่ข้อความ SIP ไม่กระบวนการรักษาความลับและไม่มีกระบวนการพิสูจน์ตัวตนเมื่อได้รับคำร้องขอ CANCEL โดยใช้ Request-URI, Via, From, To, Call-ID และ CSeq ที่สอดคล้องคำร้องขอ (3) INVITE (ภาพประกอบที่ 3.29) เพื่อปลอมแปลงคำร้องขอ (7) CANCEL ส่งผลให้คำร้องร้อง INVITE ของ UAC ถูกยกเลิก



ภาพประกอบที่ 3.27 ตัวอย่างการขอยกเลิกการเชื่อมต่อโดยใช้ CANCEL



ภาพประกอบที่ 3.28 ตัวอย่าง CANCEL Attack

1. INVITE **sip:bob@sipserver.cs.psu.ac.th:5060** SIP/2.0
2. **Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1711-1-3**
3. **From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1**
4. **To: <sip:bob@sipserver.cs.psu.ac.th:5060>**
5. **Call-ID: 1-1711@10.0.0.3**
6. **CSeq: 2 INVITE**
7. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
8. Proxy-Authorization: Digest username="alice", … ,algorithm=MD5
9. Max-Forwards: 70
10. Content-Type: application/sdp
11. Content-Length: 140

ภาพประกอบที่ 3.29 CANCEL Attack: (3) INVITE

จากที่ได้กล่าวมาแล้วว่าพร็อกซีไม่ส่งต่อคำร้องขอ CANCEL แต่จะสร้างคำร้องขอขึ้นมาใหม่เพื่อส่งไปยัง UAS ดังนั้นคำร้องขอ (7) CANCEL และ (8) CANCEL (รายละเอียดดูภาพประกอบที่ 3.30 และ 3.31 ตามลำดับ) มีส่วนของ Request-URI และ Via แตกต่างกัน

1. CANCEL **sip:bob@sipserver.cs.psu.ac.th:5060** SIP/2.0
2. **Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1711-1-3**
3. **From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1**
4. **To: <sip:bob@sipserver.cs.psu.ac.th:5060>**
5. **Call-ID: 1-1711@10.0.0.3**
6. **CSeq: 2 CANCEL**
7. Content-Length: 0

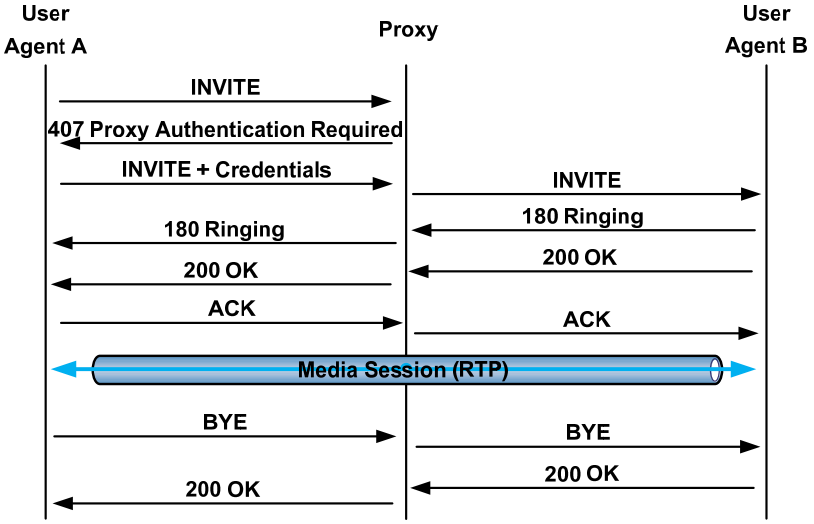
ภาพประกอบที่ 3.30 CANCEL Attack: (7) CANCEL

CANCEL **sip:bob@bobclient.sipserver.cs.psu.ac.th:5060** SIP/2.0
Via: SIP/2.0/UDP **10.0.0.2;branch=z9hG4bKa37d.b94f7756.0**
From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
To: <sip:bob@sipserver.cs.psu.ac.th:5060>
Call-ID: 1-1711@10.0.0.3
CSeq: 2 CANCEL
Max-Forwards: 70
User-Agent: OpenSIPS (1.6.4-2-tls (i386/linux))
Content-Length: 0

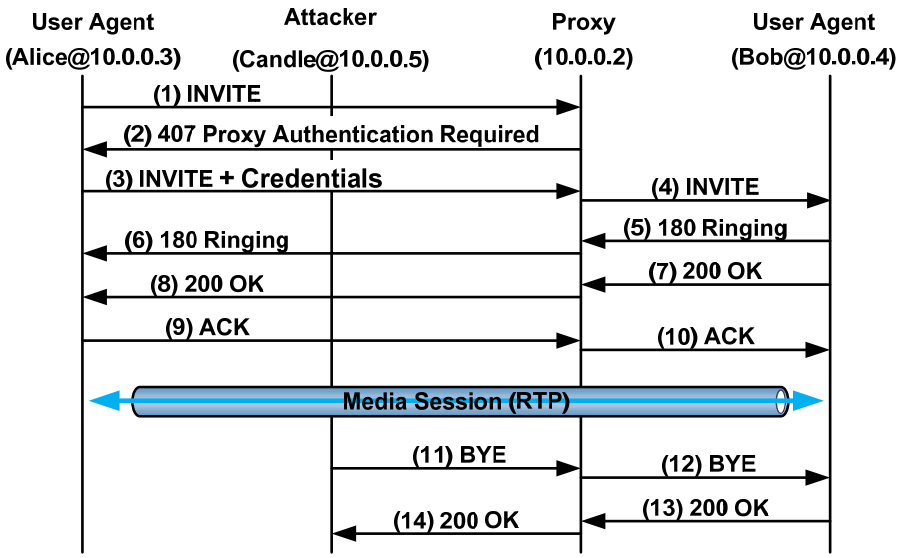
ภาพประกอบที่ 3.31 CANCEL Attack: (8) CANCEL

8) BYE Attack

คำร้องขอ BYE ใช้ยกเลิกการเชื่อมต่อที่สร้างขึ้นมาแล้ว ดังภาพประกอบที่ 3.32 ผู้บุกรุกอาศัยจุดอ่อนที่ข้อความ SIP ไม่กระบวนการรักษาความลับและไม่มีกระบวนการพิสูจน์ตัวตนเมื่อได้รับคำร้องขอ BYE เพื่อการโจมตีดังภาพประกอบที่ 3.33 โดยใช้ Request-URI, Via, Route, From, To, Call-ID และ CSeq จากการดักจับข้อความ (9) ACK (รายละเอียดดูภาพประกอบที่ 3.34) แล้วสร้างคำร้องขอ (11) BYE โดยการเปลี่ยน Method จาก ACK เป็น BYE และเพิ่มหมายเลขลำดับให้กับเฮดเดอร์ CSeq ดังภาพประกอบที่ 3.35 เมื่อ UAS ได้รับคำร้องขอ BYE ที่มาจากผู้บุกรุกนี้ก็จะหยุดการเชื่อมต่อกับ UAC ส่งผลให้ UAC ซึ่งยังมีการเชื่อมต่ออยู่นั้นไม่ได้รับข้อมูล ทั้งเสียง วิดีโอ หรือข้อความใดๆ จาก UAS



ภาพประกอบที่ 3.32 การเชื่อมต่อผู้ใช้ในเหตุการณ์ปกติ



ภาพประกอบที่ 3.33 ตัวอย่าง BYE Attack

1. **ACK** sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1674-1-8
3. Route: <sip:10.0.0.2;lr=on;did=144.fc0de986>
4. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
5. To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=2
6. Call-ID: 1-1674@10.0.0.3
7. CSeq: **2 ACK**
8. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
9. Max-Forwards: 70
10. Content-Length: 0

ภาพประกอบที่ 3.34 BYE Attack: (9) ACK

1. **BYE** sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
2. Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1674-1-8
3. Route: <sip:10.0.0.2;lr=on;did=144.fc0de986>
4. From: <sip:alice@sipserver.cs.psu.ac.th:5060>;tag=1
5. To: <sip:bob@sipserver.cs.psu.ac.th:5060>;tag=2
6. Call-ID: 1-1674@10.0.0.3
7. CSeq: **3 BYE**
8. Contact: sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060
9. Max-Forwards: 70
10. Content-Length: 0

ภาพประกอบที่ 3.35 BYE Attack: (11) BYE

จากตัวอย่างพบว่า การโจมตีแบบ Call Termination Hijacking มีการปลอมแปลงแพ็กเก็ต RTP ด้วย ซึ่งงานวิจัยนี้เน้นการแก้ปัญหาที่ SIP ดังนั้นจึงไม่นำการโจมตีวิธีนี้มาพิจารณาเพื่อออกแบบระบบ

การโจมตีในขั้นตอนการพิสูจน์ตัวตนระหว่าง UAC และเซิร์ฟเวอร์ที่เกิดขึ้นได้ คือ Registration Hijacking, Invite Replay Billing Attack และ Call Establishment Hijacking วิธีการสำคัญที่ใช้โจมตีคือ การแก้ไขข้อความ SIP ที่มีข้อมูลประจำตัวของผู้ใช้อยู่ ข้อมูลประจำตัวนี้อยู่ในเฮดเดอร์ Authorization และ Proxy-Authorization ส่วนการเชื่อมต่อระหว่าง UAC และ UAS สามารถถูกโจมตีแบบ UPDATE Attack, Re-INVITE Attack, CANCEL Attack และ BYE Attack โดยใช้วิธีการปลอมแปลงข้อความขึ้นมาใหม่จากข้อมูลที่ดักจับได้ แต่

ต้องแก้ไขข้อมูลบางส่วนเพื่อให้โจมตีได้สำเร็จ ซึ่งส่วนของข้อความ SIP ที่อาจถูกแก้ไขเพื่อการโจมตีสัญญาณเชื่อมต่อแสดงดังตารางที่ 3.1

ตารางที่ 3.1 ส่วนของข้อความ SIP ที่อาจถูกแก้ไขเพื่อการโจมตีสัญญาณเชื่อมต่อ

การโจมตี (Attack Mechanism)	**Request-URI**	**CSeq**	**Contact**	**Expires**	**SDP Body**
1. Registration Hijacking		✓	✓	✓	
2. Invite Replay Billing Attack	✓	✓			✓
3. Call Establishment Hijacking	✓				✓
4. UPDATE Attack		✓			✓
5. Re-INVITE Attack		✓			✓
6. CANCEL Attack		✓			
7. BYE Attack		✓			

ดังนั้น สามารถแบ่งกลุ่มส่วนของข้อความ SIP ที่เสี่ยงต่อการใช้โจมตีสัญญาณเชื่อมต่อได้เป็น 3 กลุ่ม ดังนี้

1) เฮดเดอร์ที่จำเป็นสำหรับการปลอมแปลงข้อความ SIP ขึ้นมาใหม่ ได้แก่ พารามิเตอร์ branch ของ Via, From, To, Call-ID และ CSeq

2) เฮดเดอร์ที่ประกอบด้วยข้อมูลประจำตัว ได้แก่ Authorization และ Proxy-Authorization

3) เฮดเดอร์ที่อาจถูกแก้ไข ได้แก่ CSeq และ Contact ส่วนของ Request-URI และ SDP Body ของ SIP

ในการแบ่งกลุ่มนี้ไม่ได้รวมเฮดเดอร์ Expires เข้าไปด้วยเนื่องจากเฮดเดอร์ Expires ใช้เพื่อยกเลิกการลงทะเบียนสำหรับการโจมตีแบบ Registration Hijacking เพียงอย่างเดียว ซึ่งการโจมตีวิธีนี้จำเป็นต้องแก้ไขเฮดเดอร์ CSeq ด้วย การตรวจสอบการโจมตีจึงอาจใช้เฮดเดอร์ CSeq เพียงอย่างเดียวก็ได้ โดยส่วนของข้อความ SIP ที่ได้แบ่งกลุ่มไว้นี้จะนำมาใช้ออกแบบกลไกสร้างความมั่นคงให้กับสัญญาณเชื่อมต่อของ SIP ต่อไป โดยผลที่เกิดจากการโจมตีสัญญาณเชื่อมต่อของ SIP สามารถสรุปได้ดังหัวข้อถัดไป

3.5 สรุปการโจมตีและผลที่เกิดจากการโจมตีสัญญาณเชื่อมต่อของ SIP

จากจุดอ่อนของ SIP ตามที่ได้กล่าวมาแล้วข้างต้น ส่งผลให้ผู้บุกรุกสามารถส่งข้อความ SIP ต่างๆ เพื่อขอเชื่อมต่อ ยกเลิกหรือสิ้นสุดการโทร เปลี่ยนเส้นทางการโทร และเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชัน เช่น วิธีการแปลงสัญญาณ (Codec) หมายเลขไอพี และหมายเลขพอร์ตของการส่งข้อมูลด้วย RTP ซึ่งวิธีการโจมตีที่มักถูกกล่าวถึงในงานวิจัยแสดงดังตารางที่ 3.2 ทั้งนี้เพราะเป็นการโจมตีการทำงานหลักๆ ของ SIP คือ REGISTER, INVITE, CANCEL และ BYE นอกจากนี้ SIP ยังมีส่วนขยาย (Extension) อื่นๆ ที่ช่วยเพิ่มความสามารถในการทำงานของ SIP เช่น INFO Method (Donovan, 2000) ใช้สำหรับส่งข้อมูลที่เกี่ยวข้องกับเซสชันหลังจากที่สร้างเซสชันขึ้นมาแล้ว ได้แก่ ข้อมูลรูปภาพและยอดเงินคงเหลือในบัญชี และ REFER Method (Sparks, 2003) ใช้สำหรับระบุว่าผู้รับคำร้องขอนี้ควรติดต่อบุคคลอื่นโดยใช้ข้อมูลการติดต่อที่ให้ไว้ในคำร้องขอ ส่วน UPDATE Method (Rosenberg, 2002) ก็เป็นส่วนขยายของ SIP เช่นเดียวกัน แต่มีลักษณะการทำงานคล้าย Re-INVITE จึงรวม UPDATE ไว้ในตารางด้วย และเมื่อวิเคราะห์การโจมตีต่างๆ สามารถสรุปได้ว่าการโจมตีแบบ Registration Hijacking, Invite Replay Billing Attack และ Call Establishment Hijacking เกิดขึ้นได้เนื่องจากไม่มีการรักษาความถูกต้องสมบูรณ์และการรักษาความลับให้กับข้อความ SIP ผู้บุกรุกจึงสามารถดักจับและศึกษาข้อมูล เพื่อเปลี่ยนแปลงข้อมูลได้ โดยเฉพาะอย่างยิ่งถ้าเซิร์ฟเวอร์มีข้อผิดพลาดในการพัฒนาฟังก์ชันสำหรับป้องกันการนำข้อมูลกลับมาส่งใหม่ (ไม่มีการป้องกันการนำค่า nonce กลับมาใช้ใหม่) จะทำให้เกิดการโจมตีแบบ Replay Attack นอกจากการเปลี่ยนแปลงข้อมูลแล้ว ผู้โจมตีสามารถปลอมแปลงข้อความ SIP ขึ้นมาใหม่โดยอาศัยข้อมูลที่ดักจับไว้เพื่อใช้โจมตี เช่น UPDATE Attack, Re-INVITE Attack, CANCEL Attack และ BYE Attack นอกจากนี้ ยังสามารถนำค่า response จากเฮดเดอร์ Authorization และ Proxy-Authorization มาคำนวณหารหัสผ่านได้อีกด้วย การเข้ารหัสสามารถแก้ปัญหาเหล่านี้ได้ แต่การนำข้อความ SIP มาเข้ารหัสมีข้อจำกัดคือ เซิร์ฟเวอร์มีความจำเป็นต้องเข้าถึงบางเฮดเดอร์ ดังนั้นทุกเซิร์ฟเวอร์ที่อยู่ระหว่างทางของการสื่อสารจะต้องมีทั้งการถอดรหัสและเข้ารหัสใหม่ ซึ่งนอกจากเสียเวลาแล้วยังไม่สามารถรับประกันได้ว่าทุกเอนทิตีจะมีกลไกการเข้ารหัสอยู่ ทำให้ผู้โจมตีมีโอกาสดักจับข้อมูลได้ ผลกระทบที่สำคัญคือ ก่อให้เกิดการเข้าถึงบริการโดยไม่ได้รับอนุญาต (Unauthorized Access) และการปฏิเสธการให้บริการ (DoS) การเข้าถึงบริการโทรออกโดยไม่ได้รับอนุญาตย่อมส่งผลต่อผู้ใช้งานที่แท้จริงเพราะต้องรับผิดชอบต่อค่าโทรที่เกิดขึ้น

ตารางที่ 3.2 การโจมตีสัญญาณเชื่อมต่อและผลกระทบ

การโจมตี (Attack Mechanism)	จุดอ่อน				ผลกระทบ				ผลที่ตามมา	
	C	I	A	Anti-Replay	C	I	Au	Av	DoS	UnA
1. Registration Hijacking	✓	✓	-	✓	✓	✓	✓	✓	✓	✓
2. Invite Replay Billing Attack	✓	✓	-	✓	✓	✓	✓	-	-	✓
3. Call Establishment Hijacking	✓	✓	-	-	✓	✓	✓	✓	✓	✓
4. Call Termination Hijacking	✓	-	✓	-	✓	✓	-	-	-	✓
5. UPDATE Attack	✓	-	✓	-	✓	-	✓	✓	✓	✓
6. Re-INVITE Attack	✓	-	✓	-	✓	-	✓	✓	✓	✓
7. CANCEL Attack	✓	-	✓	-	✓	-	✓	✓	✓	-
8. BYE Attack	✓	-	✓	-	✓	-	✓	✓	✓	-

หมายเหตุ

- C = Confidentiality
- I = Integrity
- A = Authentication
- Av = Availability
- Au = Authenticity
- UnA = Unauthorized Access
- DoS = Denial of Service

3.6 การวิเคราะห์กลไกด้านความปลอดภัยของ SIP

กลไกด้านความปลอดภัยแต่ละประเภทมีความสามารถในการให้บริการด้านความปลอดภัยในรูปแบบที่แตกต่างกัน ใน RFC 3261 (Rosenberg และคณะ, 2002) มีการแนะนำกลไกด้านความปลอดภัยบนอินเทอร์เน็ตที่รู้จักกันโดยทั่วไปที่สามารถนำมาใช้กับ SIP ได้ ข้อมูลโดยสรุปของแต่ละกลไกสามารถศึกษาได้จากบทที่ 2 และบริการด้านความปลอดภัยที่สนับสนุนโดยกลไกเหล่านี้แสดงดังตารางที่ 3.3 โดยมีรายละเอียดของบริการด้านความปลอดภัยและข้อจำกัดบางประการของแต่ละกลไกดังคำอธิบายข้างล่างนี้

ตารางที่ 3.3 บริการด้านความปลอดภัยที่สนับสนุนโดยกลไกด้านความปลอดภัยของ SIP

Security Service	SIP's Security Mechanism			
	HTTP Digest	**IPSec**	**TLS**	**S/MIME**
Integrity	-	Hop-by-Hop	Hop-by-Hop	End-to-End (Partial)
Availability	-	Hop-by-Hop (Partial)	Hop-by-Hop (Partial)	End-to-End (Partial)
Authenticity	One-Way	Hop-by-Hop	Hop-by-Hop	End-to-End
Confidentiality	-	Hop-by-Hop	Hop-by-Hop	End-to-End (Partial)
Non-Repudiation	-	Hop-by-Hop	-	End-to-End

1) HTTP Digest

วิธีการนี้เป็นการพิสูจน์ตัวตนของผู้ส่งข้อความเพียงฝ่ายเดียว ไม่มีกระบวนการพิสูจน์ตัวตนของเซิร์ฟเวอร์และการพิสูจน์ตัวตนแบบ End-to-End นอกจากนี้ HTTP Digest ไม่มีกระบวนการรักษาความลับและความถูกต้องสมบูรณ์ของข้อมูล ทำให้ถูกดักจับข้อมูลประจำตัว (Credential) เพื่อเดารหัสผ่านได้

2) IP Security (IPsec)

IPsec สามารถใช้รักษาความลับและความถูกต้องสมบูรณ์ของข้อมูล การพิสูจน์ตัวตนของผู้ส่งข้อความ รวมทั้งการป้องกันการนำข้อมูลกลับมาใช้ใหม่ (Anti-Replay) และการวิเคราะห์การจราจรของเครือข่ายได้ แต่สามารถรักษาความปลอดภัยได้เพียงแบบ Hop-by-Hop เท่านั้น และจะต้องสร้างความไว้วางใจ (Trust) (เช่น Pre-Shared Keys) ขึ้นระหว่างฝ่ายต่างๆ ก่อนที่จะสื่อสารกัน IPsec ถูกพัฒนาที่ระดับระบบปฏิบัติการ (Geneiatakis และคณะ, 2006) ซึ่งมีความซับซ้อนมาก ยากต่อการพัฒนา ทำให้ไคลเอนต์ของ SIP ส่วนใหญ่ยังไม่ได้มีการพัฒนาโพรโทคอลนี้

3) Transport Layer Security (TLS)

TLS ใช้รักษาความลับ ความถูกต้องสมบูรณ์ของข้อมูล และใช้พิสูจน์ตัวตนระหว่างกันได้เช่นเดียวกับ IPSec โดยไม่ต้องมีการกำหนดกุญแจที่ใช่ร่วมกันเอาไว้ก่อน และไม่มีกระบวนการห้ามปฏิเสธความรับผิดชอบ (Non-Repudiation) TLS ไม่สามารถใช้ร่วมกันกับ UDP ได้ เนื่องจากต้องมีการจองเส้นทางการสื่อสารกันก่อน (Connection-Oriented Protocols)

การเปิดการเชื่อมต่อ TCP หลายการเชื่อมต่อพร้อมๆ กัน อาจทำให้พร็อกซีเซิร์ฟเวอร์ทำงานหนักเกินไป และเนื่องจากไม่สามารถรับประกันได้ว่าทุกจุดจากต้นทางจนถึงปลายทางจะมีการใช้งาน TLS ทำให้ไม่สามารถสร้างความปลอดภัยตั้งแต่ต้นทางจนถึงปลายทาง (End-to-End) ได้

4) S/MIME

S/MIME สนับสนุนการรักษาความลับและความถูกต้องสมบูรณ์ของข้อมูลแบบ End-to-End แต่จากข้อจำกัดในการทำงานของ SIP เซิร์ฟเวอร์จะต้องเข้าถึงเฮดเดอร์เพื่ออ่าน แก้ไขหรือใช้กำหนดเส้นทางของข้อความ SIP ไปยังปลายทางที่กำหนดไว้ จึงไม่สามารถใช้ S/MIME เพื่อรักษาความลับและความถูกต้องสมบูรณ์ของข้อความ SIP ทั้งข้อความได้ และการใช้ S/MIME จะทำให้ข้อความ SIP มีขนาดใหญ่มาก รวมทั้งต้องใช้เวลาในการดำเนินการเกี่ยวกับวิทยาการเข้ารหัสลับ (Cryptography) ค่อนข้างมาก

ในบทที่ 2 ได้กล่าวถึงงานวิจัยที่มีการออกแบบกลไกความปลอดภัยสำหรับป้องกันการโจมตีสัญญาณเชื่อมต่อ โดยแต่ละกลไกสามารถสนับสนุนบริการด้านความปลอดภัยได้ดังตารางที่ 3.4 และสามารถป้องกันการโจมตีสัญญาณเชื่อมต่อต่างๆ ได้ดังตารางที่ 3.5 และเพื่อความสะดวกต่อการสรุปเป็นตาราง จึงใช้ตัวเลขแทนการอ้างอิงถึงบทความ ดังนี้

[1] = SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments (Wu และคณะ, 2004)

วิธีการนี้ช่วยป้องกันการปฏิเสธการให้บริการที่อาศัยจุดอ่อนของ SIP คือไม่มีกระบวนการพิสูจน์ตัวตนเมื่อได้รับคำร้องขอในการยกเลิกการเชื่อมต่อ และไม่มีการตรวจสอบการแก้ไขข้อมูลรายละเอียดของเซสชัน โดยสามารถตรวจจับการโจมตีด้วยการส่งข้อความ BYE และ Re-INVITE ได้

[2] = Providing Response Identity and Authentication in IP Telephony (Cao และ Jennings, 2006)

มีการรักษาความถูกต้องสมบูรณ์ของข้อความตอบกลับและการพิสูจน์ตัวตนของข้อความต่างๆ ที่มีการส่งหลังจากส่งข้อความ INVITE ทำให้สามารถป้องกันการโจมตีด้วยการส่งข้อความ BYE, CANCEL, UPDATE และ Re-INVITE ได้

[3] = Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP) (Peterson และ Jennings, 2006)

กลไกนี้มีการแฮช Message Body ของข้อความแล้วเข้ารหัสด้วยกุญแจส่วนตัวของผู้ส่งข้อความ จึงช่วยป้องกันความปลอดภัยให้กับข้อความที่มีส่วนของ Message Body ได้แก่ INVITE, Re-INVITE และ UPDATE

[4] = Holistic VoIP Intrusion Detection and Prevention System (Nassar และคณะ, 2007)

เป็นกลไกที่สามารถตรวจจับการโจมตีเมื่อได้รับข้อความ BYE และ CANCEL จึงช่วยป้องกันการปฏิเสธการให้บริการได้อีกด้วย

[5] = A Lightweight Protection Mechanism against Signaling Attacks in a SIP-Based VoIP Environment (Geneiatakis และ Lambrinoudakis, 2008)

กลไกนี้ใช้วิธีการตรวจสอบความถูกต้องสมบูรณ์ของทุกๆ ข้อความ จึงสามารถป้องกันการโจมตีสัญญาณเชื่อมต่อได้ทุกประเภทที่กล่าวมาในตารางที่ 3.5 ยกเว้นการโจมตีแบบ Call Termination Hijacking

[6] = A Novel Approach to Avoid Billing Attack on VOIP System. World Academy of Science, Engineering and Technology (Shekokar และ Devane, 2010)

กลไกนี้มีการใช้ TLS เพื่อเข้ารหัสข้อความ SIP ทำให้สามารถป้องกันการโจมตีสัญญาณเชื่อมต่อได้ทุกประเภทที่กล่าวมาในตารางที่ 3.5 ยกเว้นการโจมตีแบบ Call Termination Hijacking

ตารางที่ 3.4 บริการด้านความปลอดภัยที่สนับสนุนโดยกลไกป้องกันการโจมตีสัญญาณเชื่อมต่อ

Security Service	**งานวิจัย/RFC**					
	[1]	**[2]**	**[3]**	**[4]**	**[5]**	**[6]**
Integrity	-	Partial	-	-	Hop-by-Hop	Hop-by-Hop
Availability	Partial	-	Partial	Partial	Hop-by-Hop (Partial)	Hop-by-Hop (Partial)
Authenticity	-	Partial	Partial	-	Hop-by-Hop	Hop-by-Hop
Confidentiality	-	-	-	-	-	Hop-by-Hop
Non-Repudiation	-	Partial	Partial	-	-	-

ตารางที่ 3.5 การป้องกันการโจมตีที่สนับสนุนโดยกลไกป้องกันการโจมตีสัญญาณเชื่อมต่อ

การโจมตี (Attack Mechanism)	งานวิจัย/RFC					
	[1]	[2]	[3]	[4]	[5]	[6]
1. Registration Hijacking	-	-	-	-	✓	✓
2. Invite Replay Billing Attack	-	-	✓	-	✓	✓
3. Call Establishment Hijacking	-	-	✓	-	✓	✓
4. Call Termination Hijacking	-	-	-	-	-	-
5. UPDATE Attack	-	✓	✓	-	✓	✓
6. Re-INVITE Attack	✓	✓	✓	-	✓	✓
7. CANCEL Attack	-	✓	-	✓	✓	✓
8. BYE Attack	✓	✓	-	✓	✓	✓

ถึงแม้ว่าการเพิ่มเฮดเดอร์ Integrity-Auth จะสามารถป้องกันการโจมตีสัญญาณเชื่อมต่อได้เกือบทั้งหมด แต่กลไกนี้อาจเกิดการโจมตีด้วยการคำนวณหารหัสผ่าน (Off-Line Password Guessing) ได้ และพร็อกซีต้องรู้รหัสผ่านของทั้ง 2 ฝ่ายเพื่อให้กลไกสามารถทำงานได้ วิธีการนี้จึงใช้ได้เมื่อผู้ใช้อยู่ในโดเมนเดียวกันเท่านั้น ส่วน TLS จะต้องมีการเข้ารหัสข้อความที่ส่งทั้งข้อความ ซึ่งต้องใช้เวลาในการเข้ารหัสและถอดรหัส

ในวิทยานิพนธ์เล่มนี้ ผู้วิจัยจึงได้มุ่งเน้นการป้องกันการโจมตีสัญญาณเชื่อมต่อที่ทำให้ระบบยังคงทำงานได้อย่างมีประสิทธิภาพ และช่วยป้องกันไม่ให้มีการโจมตีรหัสผ่านได้

3.7 SIP Extension for Signaling Attacks Protection (SIPE-SAP)

จากกลไกต่างๆ ที่สามารถนำมาใช้ป้องกันการโจมตีสัญญาณเชื่อมต่อตามที่ได้กล่าวถึงในหัวข้อที่แล้ว พบว่าการเพิ่มเฮดเดอร์ Integrity-Auth และการใช้ TLS สามารถป้องกันการโจมตีสัญญาณเชื่อมต่อได้เกือบทั้งหมด ยกเว้น Call Termination Hijacking แต่กลไกที่มีการเพิ่มเฮดเดอร์ Integrity-Auth นี้ไม่มีการรักษาความลับให้กับรหัสผ่านคือ รหัสผ่านถูกนำมาผ่านการย่อย (Digest) พร้อมด้วยเนื้อหาของข้อความ SIP ซึ่งผู้บุกรุกสามารถนำข้อความที่ผ่านการย่อยนี้มาใช้คำนวณเพื่อหารหัสผ่านได้ และพร็อกซีจำเป็นต้องรู้รหัสผ่านของผู้ใช้ทั้ง 2 ฝ่ายเพื่อให้กลไกสามารถทำงานได้ วิธีการนี้จึงใช้ได้เมื่อผู้ใช้อยู่ในโดเมนเดียวกันคือใช้บริการ

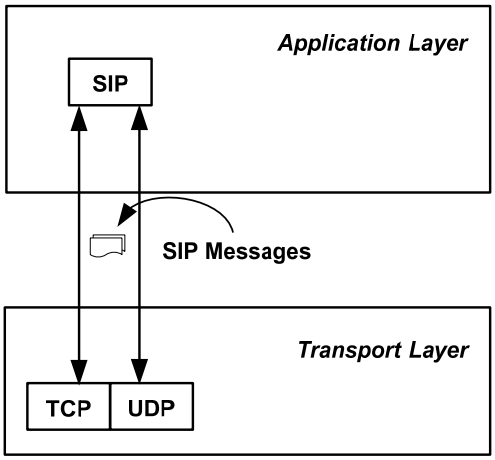
เซิร์ฟเวอร์เครื่องเดียวกันเท่านั้น ส่วนการใช้ TLS จะต้องมีการเข้ารหัสข้อความที่ส่งทั้งข้อความ ซึ่งต้องใช้เวลาในการเข้ารหัสและถอดรหัส สำหรับวิทยานิพนธ์เล่มนี้ ผู้วิจัยจึงมีวัตถุประสงค์เพื่อออกแบบกลไกที่มีความสามารถดังต่อไปนี้

1) สามารถป้องกันการโจมตีสัญญาณเชื่อมต่อด้วยวิธีต่อไปนี้
   - Registration Hijacking
   - Invite Replay Billing Attack
   - Call Establishment Hijacking
   - UPDATE Attack
   - Re-INVITE Attack
   - CANCEL Attack
   - BYE Attack
2) สามารถประยุกต์ใช้กลไกเพื่อป้องกัน Call Termination Hijacking
3) มีการรักษาความลับให้กับรหัสผ่าน
4) รองรับการใช้งานต่างโดเมน
5) สามารถทำงานได้อย่างมีประสิทธิภาพเมื่อเทียบกับ TLS

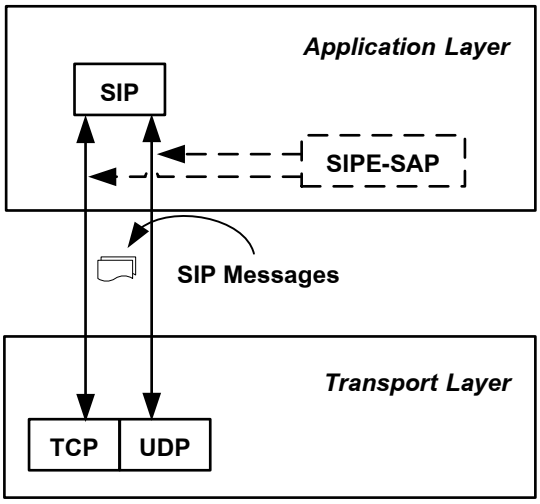
เพื่อรองรับการทำงานดังกล่าว ผู้วิจัยจึงเสนอกลไก SIPE-SAP ซึ่งเป็นส่วนขยายที่เพิ่มเข้าไปในข้อความ SIP โดยทั่วไปเมื่อมีการสร้างข้อความ SIP จากชั้นแอปพลิเคชันแล้วสามารถเรียกใช้ TCP หรือ UDP ในชั้นทรานสปอร์ตเพื่อขนส่งข้อความผ่านเครือข่ายไปยังผู้รับ ดังภาพประกอบที่ 3.36 กลไก SIPE-SAP จะทำงานอยู่ในชั้นแอปพลิเคชัน โดยเพิ่มเติมความสามารถในการทำงานให้กับ SIP เพื่อเพิ่มความปลอดภัยให้กับข้อความ SIP สามารถเรียกใช้กลไก SIPE-SAP ก่อนจะส่งข้อความไปยังชั้นทรานสปอร์ต หรือเรียกใช้กลไก SIPE-SAP เมื่อได้รับข้อความจากชั้นทรานสปอร์ตได้ดังภาพประกอบที่ 3.37

SIPE-SAP ใช้กระบวนการเข้ารหัสและการย่อยข้อความเพื่อป้องกันการโจมตีสัญญาณเชื่อมต่อรูปแบบต่างๆ โดยเข้ารหัสค่า response ที่อยู่ในเฮดเดอร์ Authorization และ Proxy-Authorization ของข้อความ REGISTER และ INVITE เพราะจากการวิเคราะห์การโจมตีพบว่าผู้บุกรุกสามารถเดารหัสผ่านโดยใช้วิธีการคำนวณและเปรียบเทียบค่า response ได้ ส่วนการย่อยข้อความจะใช้กับส่วนต่างๆ ของข้อความ SIP ที่อาจถูกแก้ไขเพื่อให้การโจมตีประสบความสำเร็จตามที่ได้วิเคราะห์ไว้ในตารางที่ 3.1 โดยถ้าเป็นข้อความ REGISTER และ INVITE ส่วนต่างๆ ของข้อความ SIP ที่จะถูกนำมาย่อย ได้แก่ Request-URI, Cseq, Contact และ SDP Body แต่ถ้าเป็นข้อความ UPDATE, Re-INVITE, CANCEL และ BYE ส่วนต่างๆ ของข้อความ SIP ที่จะถูกนำมาย่อย ได้แก่ Cseq, และ SDP Body รวมถึงเฮดเดอร์อื่นๆ ที่จำเป็นสำหรับการปลอมแปลงข้อความ SIP ขึ้นมาใหม่ ได้แก่ พารามิเตอร์ branch ของเฮดเดอร์ Via, From, To

และ Call-ID เพราะเฮดเดอร์เหล่านี้ใช้ระบุกลุ่มของข้อความในแต่ละ Dialog และ Transaction ได้ การนำเฮดเดอร์เหล่านี้มาย่อยข้อความจะช่วยป้องกันการนำข้อมูลที่ผ่านการย่อยนี้ไปใช้กับ Dialog หรือ Transaction อื่นๆ โดยกระบวนการทำงานของ SIPE-SAP มีรายละเอียดดังคำอธิบายในหัวข้อ 3.7.1



ภาพประกอบที่ 3.36 การรับส่งข้อมูลผ่านเครือข่ายอินเทอร์เน็ตของ SIP



ภาพประกอบที่ 3.37 การรับส่งข้อมูลผ่านเครือข่ายของ SIP โดยมีกลไก SIPE-SAP

3.7.1 กระบวนการทำงานของ SIPE-SAP

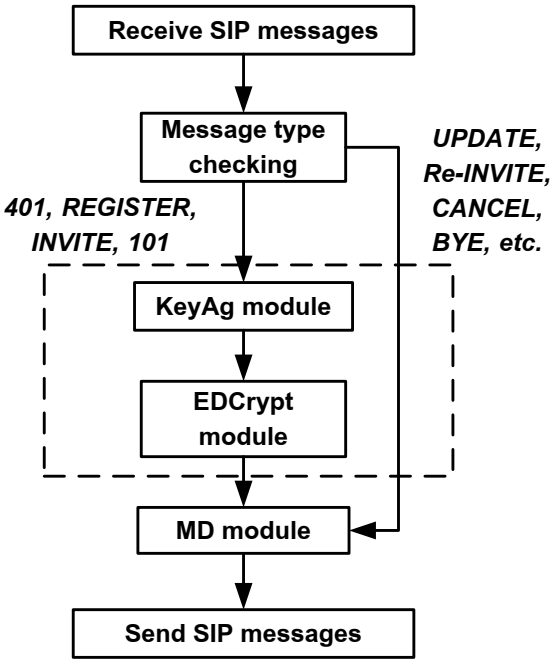
SIPE-SAP สามารถแบ่งกระบวนการทำงานออกเป็น 3 โมดูลหลักๆ (ตามภาพประกอบ 3.38) ที่ทำงานเกี่ยวข้องสัมพันธ์กันคือ KeyAg, EDCrypt และ MD แต่ละโมดูลมีวัตถุประสงค์หลักคือ

1) KeyAg สนับสนุนการทำงานของ MD และ EDCrypt โดยรองรับการใช้งานต่างโดเมนกัน และสามารถนำไปประยุกต์ใช้เพื่อป้องกัน Call Termination Hijacking ได้

2) MD ช่วยป้องกันการโจมตีสัญญาณเชื่อมต่อด้วยวิธีต่อไปนี้

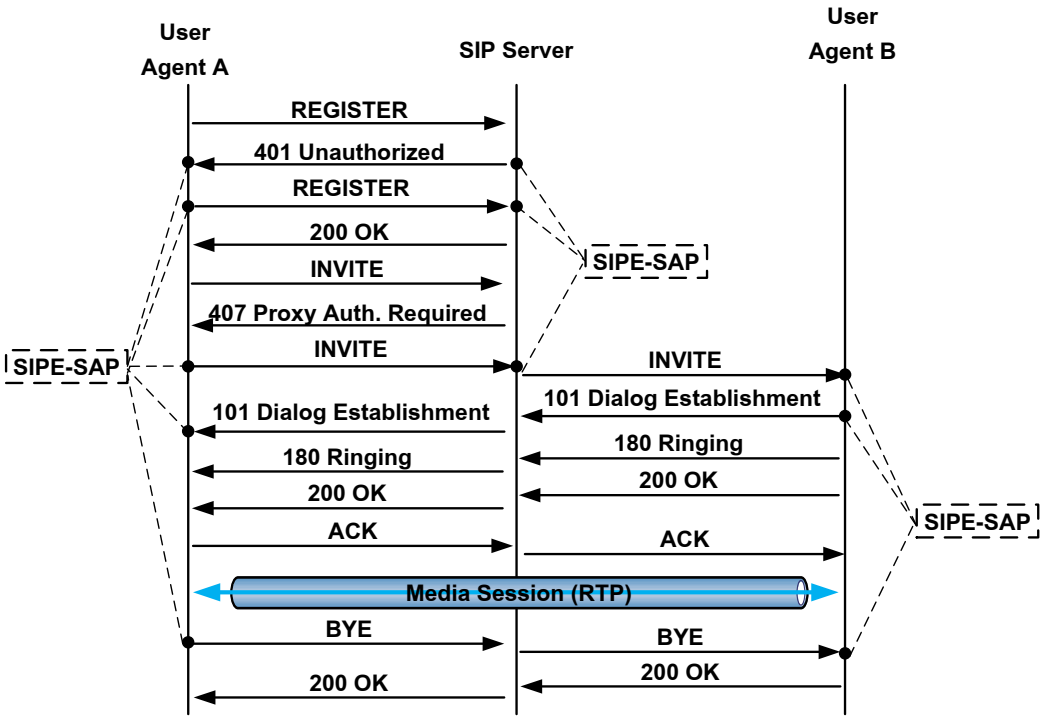
- Registration Hijacking
- Invite Replay Billing Attack
- Call Establishment Hijacking
- UPDATE Attack
- Re-INVITE Attack
- CANCEL Attack
- BYE Attack

3) EDCrypt ช่วยรักษาความลับให้กับรหัสผ่าน และสร้างลายเซ็นดิจิทัล



ภาพประกอบที่ 3.38 กระบวนการทำงานของ SIPE-SAP

กระบวนการทำงานของ SIPE-SAP เมื่อได้รับข้อความ SIP คือ จะตรวจสอบข้อความที่ได้รับก่อนว่าเป็นข้อความประเภทใด ซึ่งข้อความ SIP ที่เกี่ยวข้องกับการทำงานของ SIPE-SAP ในกระบวนการลงทะเบียนและการเชื่อมต่อผู้ใช้ ประกอบด้วย 401 Unauthorized, REGISTER, INVITE, 101 Dialog Establishment และ BYE ดังภาพประกอบที่ 3.39 นอกจากนี้ยังสามารถใช้งาน SIPE-SAP กับข้อความ UPDATE, Re-INVITE และ CANCEL ได้อีกด้วย การดำเนินการกับข้อความ SIP โดยสรุปมีดังนี้



ภาพประกอบที่ 3.39 ข้อความ SIP ที่เกี่ยวข้องกับการทำงานของ SIPE-SAP ในกระบวนการลงทะเบียนและการเชื่อมต่อผู้ใช้

- 401 Unauthorized: ฝั่งเซิร์ฟเวอร์จะสร้างลายเซ็นดิจิทัลพร้อมทั้งให้ข้อมูลใบรับรอง ส่วน UAC จะตรวจสอบใบรับรองและลายเซ็นดิจิทัลว่าถูกต้องหรือไม่ เพื่อพิสูจน์ตัวตนของเซิร์ฟเวอร์และยืนยันว่าใบรับรองนั้นเป็นของเซิร์ฟเวอร์จริงๆ

- REGISTER: UAC จะสุ่มกุญแจเซสชันแล้วใช้เข้ารหัสค่า response เพื่อช่วยรักษาความลับให้กับรหัสผ่าน และนำข้อมูล Request-URI, Cseq และ Contact รวมทั้งกุญแจเซสชันมาย่อยข้อความเพื่อใช้ป้องกันการโจมตีแบบ Registration Hijacking มีการเข้ารหัสกุญแจเซสชันด้วยกุญแจสาธารณะของเซิร์ฟเวอร์ แล้วส่งไปยังเซิร์ฟเวอร์ เมื่อเซิร์ฟเวอร์ได้รับข้อความ REGISTER จะตรวจสอบได้ว่าข้อมูลที่ใช้ย่อยข้อความเหล่านี้ถูกแก้ไขหรือไม่ ถ้าถูกแก้ไขแสดงว่ามีการโจมตีแบบ Registration Hijacking ดังนั้น เซิร์ฟเวอร์จะไม่ดำเนินการกับคำ

ร้องขอ REGISTER นี้ แต่ถ้าข้อมูลไม่ถูกแก้ไข เซิร์ฟเวอร์จะใช้กุญแจเซสชันถอดรหัสค่า response แล้วดำเนินการพิสูจน์ตัวตนของ UAC ต่อไป

- INVITE: UAC สุ่มกุญแจเซสชันแล้วใช้เข้ารหัสค่า response เพื่อช่วยรักษาความลับให้กับรหัสผ่าน และนำ Request-URI, Cseq, Contact และ SDP Body รวมทั้งกุญแจเซสชันมาย่อยข้อความเพื่อใช้ป้องกันการโจมตีแบบ Invite Replay Billing Attack และ Call Establishment Hijacking มีการเข้ารหัสกุญแจเซสชันด้วยกุญแจสาธารณะของเซิร์ฟเวอร์ และสร้างลายเซ็นดิจิทัลพร้อมทั้งให้ข้อมูลใบรับรอง แล้วส่งไปยังเซิร์ฟเวอร์ เมื่อเซิร์ฟเวอร์ได้รับข้อความ INVITE จะตรวจสอบได้ว่าข้อมูลที่ใช้ย่อยข้อความเหล่านี้ถูกแก้ไขหรือไม่ ถ้าถูกแก้ไขแสดงว่ามีการโจมตีเกิดขึ้น ดังนั้น เซิร์ฟเวอร์จะไม่ดำเนินการกับคำร้องขอ INVITE นี้ แต่ถ้าข้อมูลไม่ถูกแก้ไข เซิร์ฟเวอร์จะใช้กุญแจเซสชันถอดรหัสค่า response แล้วดำเนินการพิสูจน์ตัวตนของ UAC ต่อไป หากการพิสูจน์ตัวตนถูกต้อง เซิร์ฟเวอร์จะลบกุญแจเซสชันที่ถูกเข้ารหัสค่า response ที่ถูกเข้ารหัส และข้อความย่อย ออกจากข้อความ SIP แล้วส่งต่อไปยัง UAS ซึ่ง UAS จะตรวจสอบใบรับรองและลายเซ็นดิจิทัลว่าถูกต้องหรือไม่ เพื่อพิสูจน์ตัวตนของ UAC และยืนยันว่าใบรับรองนั้นเป็นของ UAC จริงๆ

- 101 Dialog Establishment: UAS สุ่มกุญแจเซสชันแล้วเข้ารหัสด้วยกุญแจสาธารณะของ UAC จากนั้นสร้างลายเซ็นดิจิทัลพร้อมทั้งให้ข้อมูลใบรับรอง เมื่อ UAC ได้รับข้อความ 101 Dialog Establishment จะตรวจสอบใบรับรองและลายเซ็นดิจิทัลว่าถูกต้องหรือไม่ เพื่อพิสูจน์ตัวตนของ UAS และยืนยันว่าใบรับรองนั้นเป็นของ UAS จริงๆ แล้วใช้กุญแจส่วนตัวเพื่อถอดรหัสกุญแจเซสชัน

- BYE, UPDATE, Re-INVITE และ CANCEL: UAC จะนำข้อมูล พารามิเตอร์ branch ของเฮดเดอร์ Via, From, To, Call-ID, Cseq และ SDP Body รวมทั้งกุญแจเซสชันมาย่อยข้อความเพื่อใช้ป้องกันการโจมตีแบบ BYE Attack, UPDATE Attack, Re-INVITE Attack และ CANCEL Attack เมื่อ UAS ได้รับข้อความเหล่านี้จะมีการย่อยข้อความด้วยเช่นเดียวกัน แล้วข้อความย่อยที่ได้มาเปรียบเทียบกับข้อความย่อยจาก UAC ถ้าได้ค่าไม่ตรงกัน แสดงว่าอาจมีการปลอมแปลงข้อความ SIP หรือข้อความ SIP นี้ส่งมาจากผู้อื่นที่ไม่ใช่คู่สนทนา เพื่อใช้ในการโจมตี ดังนั้น UAS จะไม่ดำเนินตามคำร้องขอที่ส่งมานี้

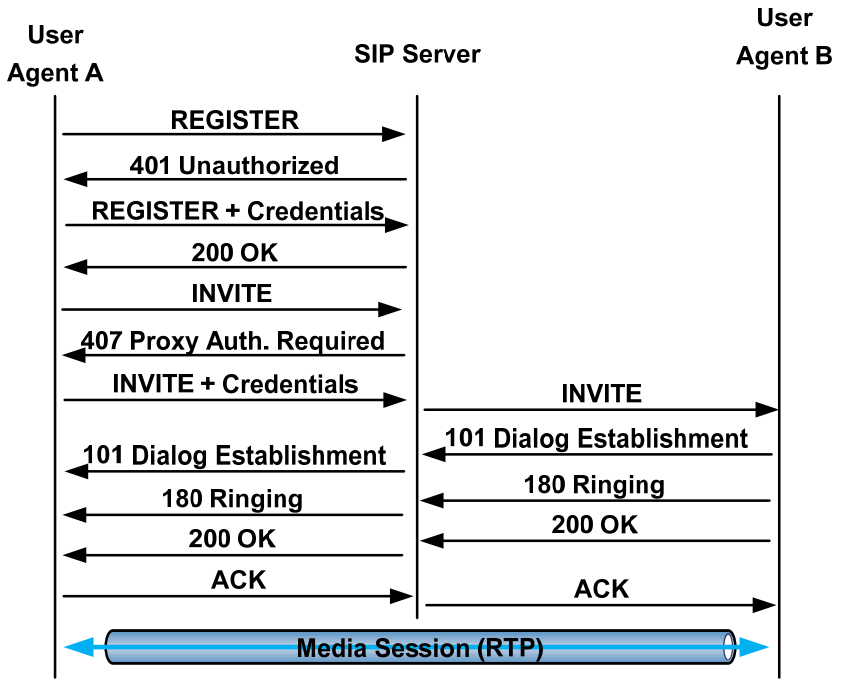
กระบวนการทำงานเหล่านี้สามารถแบ่งออกเป็น 3 โมดูล คือ KeyAg, EDCrypt และ MD ตามที่ได้กล่าวมาแล้วข้างต้น โดยรายละเอียดของแต่โมดูลมีดังนี้

1) โมดูล KeyAg

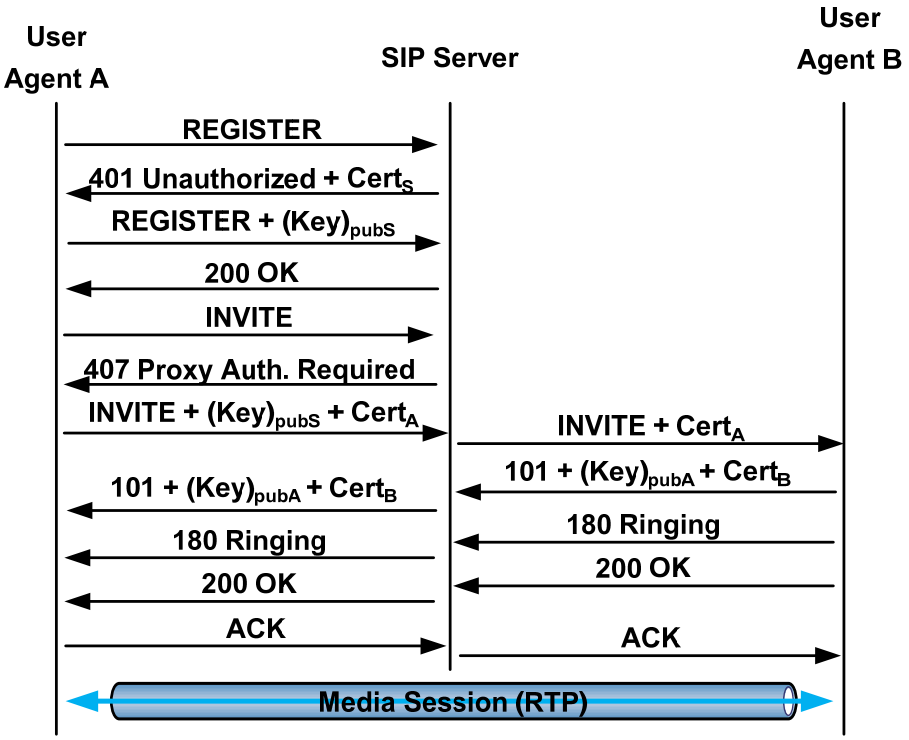
ใช้สำหรับแลกเปลี่ยนกุญแจระหว่าง UAC, UAS และ SIP Server โดยการแลกเปลี่ยนกุญแจนี้มีการนำใบรับรอง (Certificate) มาใช้กับ SIP ใบรับรองอยู่ในรูปแบบ X.509 (Housley และคณะ, 1999) ใช้เพื่อกระจายกุญแจสาธารณะที่ใช้สำหรับเข้ารหัสกุญแจเซสชัน ซึ่งส่วนประกอบของใบรับรองจะกล่าวถึงในบทถัดไป โดยในที่นี้จะให้ SIP Server เป็นผู้ออกใบรับรอง

โดยปกติเมื่อมีการลงทะเบียน UAC จะส่งคำร้องขอ REGISTER ไปยัง SIP Server แล้ว SIP Server เรียกร้องให้มีการพิสูจน์ตัวตน UAC จะส่งคำร้องขอ REGISTER ที่ประกอบด้วยข้อมูลประจำตัวไปอีกครั้งเพื่อให้ SIP Server ดำเนินการพิสูจน์ตัวตน การเชื่อมต่อผู้ใช้ก็มีกระบวนเช่นเดียวกันนี้ดังภาพประกอบที่ 3.40 แต่การใช้กลไก SIPE-SAP เมื่อมีการลงทะเบียน SIP Server จะส่งใบรับรองมาให้ UAC ซึ่ง UAC ต้องตรวจสอบใบรับรองก่อน จากนั้นสุ่มกุญแจเซสชัน (กุญแจลับ) เพื่อใช้เข้ารหัสข้อมูลพิสูจน์ตัวตนในโมดูล EDCrypt พร้อมทั้งเข้ารหัสกุญแจเซสชันด้วยกุญแจสาธารณะของ SIP Server แล้วส่งไปยัง SIP Server และเมื่อ UAC ต้องการสร้างเซสชัน สามารถใช้กุญแจสาธารณะจากใบรับรองที่ได้รับมาในขั้นตอนการลงทะเบียนเพื่อเข้ารหัสกุญแจสุ่มได้เลย

ส่วนการแลกเปลี่ยนกุญแจระหว่าง UAC และ UAS กระทำเมื่อ UAC ส่งข้อความ INVITE โดยส่งใบรับรองไปด้วย เมื่อ UAS ได้รับจะตรวจสอบใบรับรองก่อน จากนั้นสุ่มกุญแจเซสชันพร้อมทั้งเข้ารหัสด้วยกุญแจสาธารณะของ UAC แล้วส่งไปยัง UAC พร้อมด้วยใบรับรองของ UAS ดังภาพประกอบที่ 3.41



ภาพประกอบที่ 3.40 กระบวนการลงทะเบียนและการเชื่อมต่อผู้ใช้ตามปกติ



ภาพประกอบที่ 3.41 กระบวนการแลกเปลี่ยนกุญแจระหว่างพร็อกซีและไคลเอนต์

2) โมดูล EDCrypt

ใช้สำหรับเข้ารหัสและถอดรหัสค่า response ของเฮดเดอร์ Authorization และ Proxy-Authorization ในกระบวนการพิสูจน์ตัวตนที่อาจถูกโจมตีโดยการเดารหัสผ่าน เช่น Digest Authentication หรือเข้ารหัสเฮดเดอร์อื่นๆ ตามที่ต้องการ โดยค่า response มีขนาด 32 ไบต์ การเข้ารหัสและถอดรหัสใช้วิทยาการเข้ารหัสลับแบบกุญแจสมมาตร ได้แก่ Data Encryption Standard (DES) ซึ่งกุญแจที่ใช้มีขนาด 56 บิต หรือ Advanced Encryption Standard (AES) กุญแจที่ใช้มีขนาด 128 192 และ 256 บิต ทั้งนี้เพราะมีความรวดเร็วในการเข้าและถอดรหัสมากกว่าการใช้วิทยาการเข้ารหัสลับแบบกุญแจอสมมาตรและข้อความไซเฟอร์ (Cipher) ที่ได้โดยทั่วไปจะมีขนาดเท่ากับหรือน้อยกว่าต้นฉบับ

การสร้างลายเซ็นดิจิทัล จะมีการนำพารามิเตอร์ branch ของ Via, From, To, Call-ID และ Cseq มาผ่านการย่อยข้อความก่อนแล้วใช้กุญแจส่วนตัวของวิทยาการเข้ารหัสลับแบบกุญแจอสมมาตรเพื่อเข้ารหัส การเข้ารหัสนี้ใช้อัลกอริทึม Rivest-Shamir-Adleman (RSA)

3) โมดูล MD

ใช้สำหรับย่อย (Digest) เฮดเดอร์สำคัญที่เสี่ยงต่อการใช้โจมตี ข้อมูลเซสชันและกุญแจเซสชัน เพื่อใช้ตรวจสอบความถูกต้องสมบูรณ์ของข้อมูลและตรวจสอบว่าข้อความเหล่านี้สร้างโดยคู่สนทนาหรือไม่ โดยข้อมูลที่จะนำมาย่อยประกอบด้วย ส่วนของข้อความ SIP ที่อาจถูกแก้ไขโดยผู้โจมตีและเฮดเดอร์ที่ใช้ระบุเซสชันตามที่ได้วิเคราะห์ไว้ แต่เฮดเดอร์ Via สามารถถูกเพิ่ม ลบ หรือแก้ไขโดยเซิร์ฟเวอร์ได้ ดังนั้นจึงเลือกใช้เฉพาะพารามิเตอร์ branch ของเฮดเดอร์ Via ที่อยู่บนสุดของข้อความเท่านั้น ข้อมูลที่นำมาย่อยแบ่งเป็น 2 กลุ่ม กลุ่มแรกใช้กับข้อความ REGISTER และ INVITE กลุ่มที่สองใช้กับข้อความอื่นๆ ดังนี้

- Request-URI, Cseq, Contact, SDP Body และ Session Key
- พารามิเตอร์ branch ของ Via, From, To, Call-ID, Cseq, Contact, SDP Body และ Session Key

โมดูล MD มีการนำข้อมูลเหล่านี้มาต่อท้ายกันแล้วใช้อัลกอริทึมที่ใช้ในการย่อยข้อความ คือ Secure Hash Algorithm - 1 (SHA-1) ซึ่งผลลัพธ์จากขั้นตอนวิธีย่อยข้อความจะมีขนาด 160 บิต หรือ Message-Digest Algorithm 5 (MD5) ผลลัพธ์มีขนาด 128 บิต โดยที่ MD5 สามารถทำงานได้เร็ว ผลลัพธ์หรือข้อความย่อยที่ได้ไม่เกิดการชนกัน

ระบบบางระบบอาจไม่ได้เลือกใช้กระบวนการพิสูจน์ตัวตนแบบ Digest Authentication และบางกระบวนการพิสูจน์ตัวตนก็มีการแลกเปลี่ยนกุญแจอยู่แล้ว เช่น การพิสูจน์ตัวตนของ Liao และ Wang (2010) มีการแลกเปลี่ยนกุญแจอยู่แล้วและไม่มีการส่ง

รหัสผ่านระหว่าง UAC และเซิร์ฟเวอร์ เป็นต้น เพื่อความยืดหยุ่นในการใช้งาน ระบบเหล่านี้อาจเลือกใช้แต่โมดูล MD เพื่อเพิ่มความปลอดภัยในการใช้งานโพรโทคอล SIP ได้

และเมื่อพิจารณาการโจมตีพบว่าการโจมตีแบบ Registration Hijacking, Invite Replay Billing Attack และ Call Establishment Hijacking จะเกิดขึ้นในช่วงที่ผู้ใช้มีการพิสูจน์ตัวตนไปยังเซิร์ฟเวอร์ การโจมตีเหล่านี้จึงให้เซิร์ฟเวอร์เป็นผู้ตรวจสอบ ส่วนการโจมตีแบบ UPDATE Attack, Re-INVITE Attack, CANCEL Attack และ BYE Attack เมื่อมีการส่งข้อความ SIP ไปยังเซิร์ฟเวอร์จะไม่มีการพิสูจน์ตัวตน และผู้บุกรุกสามารถส่งข้อความ SIP ไปยังผู้ใช้เพื่อการโจมตีเหล่านี้ได้โดยตรง ไม่จำเป็นต้องส่งผ่านเซิร์ฟเวอร์ ดังนั้นจึงควรให้ฝั่งผู้ใช้เป็นผู้ตรวจสอบการโจมตีดังสรุปตารางในตารางที่ 3.6

ตารางที่ 3.6 การโจมตีที่มีการตรวจสอบโดย SIP Server และ User Agent

Attack Mechanism	SIP Server	User Agent
1. Registration Hijacking	✓	-
2. Invite Replay Billing Attack	✓	-
3. Call Establishment Hijacking	✓	-
4. UPDATE Attack	-	✓
5. Re-INVITE Attack	-	✓
6. CANCEL Attack	-	✓
7. BYE Attack	-	✓

นอกจากนี้ ผู้วิจัยได้ออกแบบเฮดเดอร์ใหม่เพื่อรองรับการทำงานของกลไก SIPE-SAP โดยจะกล่าวถึงรายละเอียดในหัวข้อถัดไป

3.7.2 การออกแบบเฮดเดอร์

เนื่องจาก RFC 3261 (Rosenberg และคณะ, 2002) อนุญาตให้ผู้พัฒนาสามารถกำหนดเฮดเดอร์ฟิลด์หรือพารามิเตอร์ใหม่ให้กับข้อความ SIP ได้ ผู้วิจัยจึงออกแบบเฮดเดอร์ Sig-Sec เพื่อรองรับการทำงานของกลไก SIPE-SAP เฮดเดอร์ Sig-Sec สามารถนำมาใช้ได้ทั้งกับข้อความร้องขอ (Request) และข้อความตอบกลับ (Response) และอาจประกอบด้วยข้อความที่ถูกย่อยเพียงอย่างเดียว หรือมีทั้งข้อความที่ถูกย่อยและข้อความที่ถูกเข้ารหัสก็ได้ วากยสัมพันธ์ของเฮดเดอร์มีการแปล (Encoding) โดยใช้ไวยากรณ์ของ Augmented Backus-Naur Form (ABNF) ซึ่งแสดงดังภาพประกอบที่ 3.42

```
Sig-Sec = “Sig-Sec” HCOLON 1*mess_type
mess_type = encrypted/skey/digest/signature
encrypted = “encrypted” EQUAL 1*alphanum COMMA ed_algo
ed_algo = “ed_algo” EQUAL “DES/AES” COMMA
skey = “skey” EQUAL 1*alphanum COMMA
digest = “digest” EQUAL LDQUOT 32LHEX RDQUOT COMMA md_algo
md_algo = “md_algo” EQUAL “MD5/SHA1”
signature = “signature” EQUAL 1*alphanum
```

ภาพประกอบที่ 3.42 วากยสัมพันธ์ของเฮดเดอร์ Sig-Sec

วากยสัมพันธ์ของเฮดเดอร์ Sig-Sec สามารถอธิบายได้ดังนี้

- Sig-Sec มีการนิยามชื่อของเฮดเดอร์คือ “Sig-Sec” ซึ่งชื่อของเฮดเดอร์จะตามด้วยประเภทของข้อความ

- mess_type ประเภทของข้อความมี 2 ประเภทคือ ข้อความที่ถูกเข้ารหัสและข้อความที่ถูกย่อย

- encrypted ข้อความที่ถูกเข้ารหัสระบุด้วยข้อความ “encrypted” เมื่อเข้ารหัสแล้ว ข้อความจะถูกแปลให้อยู่ในรูปของตัวอักษรภาษาอังกฤษทั้งตัวใหญ่และตัวเล็ก และตัวเลข 0 – 9 แล้วตามด้วยอัลกอริทึมที่ใช้ในการเข้ารหัส

- ed_algo อัลกอริทึมที่ใช้ในการเข้ารหัสระบุด้วยข้อความ “ed_algo” อาจเป็น DES หรือ AES

- skey คือกุญแจเซสชันที่ถูกเข้ารหัสด้วยกุญแจสาธารณะ

- digest ข้อความที่ถูกย่อยระบุด้วยข้อความ “digest” ข้อความอยู่ในรูปของตัวอักษรภาษาอังกฤษทั้งตัวใหญ่ ตัวเล็กและตัวเลข

- md_algo อัลกอริทึมที่ใช้ในการย่อยระบุด้วยข้อความ “md_algo” อาจเป็น MD5 หรือ SHA1

- signature คือลายเซ็นของผู้ส่งข้อความ ได้จากการย่อยข้อความแล้วเข้ารหัสด้วยกุญแจส่วนตัวของผู้ส่ง ลายเซ็นถูกระบุด้วยข้อความ “signature” ข้อความอยู่ในรูปของตัวอักษรภาษาอังกฤษทั้งตัวใหญ่ ตัวเล็กและตัวเลข

เฮดเดอร์ Sig-Sec สามารถนำมาใช้งานกับข้อความที่เป็นคำร้องขอ ได้แก่ BYE, CANCEL, INVITE (และ Re-INVITE), REGISTER และ UPDATE โดยพร็อกซีสามารถแก้ไข หรือลบเฮดเดอร์นี้ได้ หรืออาจนำไปปรับใช้กับคำร้องขอ ACK และ OPTION ได้ และ

นำมาใช้งานกับข้อความที่เป็นคำตอบกลับคือ 101 Dialog Establishment หรือใช้กับ 401 Unauthorized โดยพร็อกซีสามารถเพิ่มเฮดเดอร์นี้ได้ ดังสรุปในตารางที่ 3.7 ซึ่งเป็นตารางที่เขียนตามรูปแบบของตารางสรุปเฮดเดอร์ของ SIP ใน RFC 3261 (Rosenberg และคณะ, 2002) โดยมีรายละเอียดคือ

- "where" เป็นคอลัมน์ที่อธิบายว่าคำร้องขอหรือคำตอบกลับใดบ้างที่สามารถนำเฮดเดอร์ฟิลด์นี้ไปใช้งานได้ โดย R หมายถึงเฮดเดอร์ฟิลด์สามารถปรากฏในคำร้องขอ
- "proxy" เป็นคอลัมน์ที่อธิบายว่าพร็อกซีสามารถดำเนินการอะไรกับเฮดเดอร์ฟิลด์นั้นได้บ้าง การดำเนินการประกอบด้วย
  - a: พร็อกซีสามารถเพิ่มหรือเติมข้อความต่อท้ายเฮดเดอร์ฟิลด์
  - m: พร็อกซีสามารถแก้ไขค่าของเฮดเดอร์ฟิลด์
  - d: พร็อกซีสามารถลบค่าของเฮดเดอร์ฟิลด์
  - r: พร็อกซีต้องสามารถอ่านค่าของเฮดเดอร์ฟิลด์
- 7 คอลัมน์ที่เหลือใช้แสดงความจำเป็นของการมีเฮดเดอร์ฟิลด์ใน Method
  - m: จำเป็นต้องมีเฮดเดอร์ฟิลด์
  - o: มีหรือไม่มีเฮดเดอร์ฟิลด์ก็ได้
  - -: ไม่สามารถนำเฮดเดอร์ฟิลด์มาปรับใช้ได้

ตารางที่ 3.7 สรุปการนำเฮดเดอร์ Sig-Sec มาใช้งานกับข้อความ SIP

Header Field	where	proxy	ACK	BYE	CAN	INV	OPT	REG	UPD
Sig-Sec	R	md	o	o	o	o	o	o	o
Sig-Sec	101		-	-	-	o	-	-	-
Sig-Sec	401	a	-	-	-	-	-	o	-

การใส่เฮดเดอร์ Sig-Sec ในข้อความ SIP แต่ละข้อความจะมีส่วนประกอบของเฮดเดอร์ต่างกันเพื่อป้องกันการโจมตีแต่ละแบบ ได้แก่ encrypted, skey, digest และ signature รวมถึงใบรับรองด้วย ซึ่งส่วนประกอบของเฮดเดอร์ Sig-Sec และใบรับรองที่สามารถเพิ่มเข้าไปในข้อความ SIP แต่ละข้อความแสดงดังตารางที่ 3.8

ตารางที่ 3.8 ส่วนประกอบของเฮดเดอร์ Sig-Sec และใบรับรองที่ใช้ในข้อความ SIP

SIP Message	encrypted	skey	digest	signature	ใบรับรอง
Register	✓	✓	✓	-	-
401 Unauthorized	-	-	-	✓	✓
INVITE	✓	✓	✓	✓	✓
101 Dialog Establishment	-	✓	-	✓	✓
UPDATE	-	-	✓	-	-
Re-INVITE	-	-	✓	-	-
CANCEL	-	-	✓	-	-
BYE	-	-	✓	-	-

3.8 สรุป

สำหรับบทนี้มีการวิเคราะห์การโจมตีสัญญาณเชื่อมต่อ และส่วนต่างๆ ของข้อความ SIP ที่เสี่ยงต่อการนำมาใช้โจมตี รวมทั้งมีการวิเคราะห์ผลที่เกิดจากการโจมตี จากนั้นมีการออกแบบกลไกที่ช่วยป้องกันการโจมตีสัญญาณเชื่อมต่อ คือ SIP Extension for Signaling Attacks Protection (SIPE-SAP) ซึ่งมีการนิยามเฮดเดอร์ใหม่คือ Sig-Sec เพื่อรองรับการทำงานของ SIPE-SAP ในบทต่อไปจะเป็นการพัฒนาระบบตามที่ได้ออกแบบไว้

บทที่ 4

การพัฒนาและการทดสอบระบบ

4.1 บทนำ

ในบทนี้จะกล่าวถึงการพัฒนาระบบตามที่ได้ออกแบบในบทที่ 3 โดยเริ่มจากการแนะนำเครื่องมือที่ใช้ และผลการพัฒนาระบบ จากนั้นมีการแสดงข้อมูลของการทดสอบระบบ โดยประกอบด้วยการทดสอบประสิทธิภาพและการทดสอบการโจมตีสัญญาณเชื่อมต่อ ซึ่งในบทที่ 3 ได้แสดงตัวอย่างการโจมตีสัญญาณเชื่อมต่อของ SIP ที่ไม่ได้ใช้กลไก SIPE-SAP ในบทนี้จะแสดงตัวอย่างการโจมตีสัญญาณเชื่อมต่อของ SIP ที่มีการใช้กลไก SIPE-SAP โดยมีรายละเอียดต่างๆ ดังนี้

4.2 การพัฒนาระบบ

การพัฒนาระบบสำหรับวิทยานิพนธ์นี้ได้นำเอากระบวนการทำงานของระบบที่ได้ออกแบบไว้ในบทที่ 3 มาดำเนินการพัฒนาระบบ โดยในหัวข้อนี้จะกล่าวถึงเครื่องมือที่ใช้การพัฒนาระบบ วิธีการพัฒนาระบบ และมีการยกตัวผลการพัฒนาระบบในส่วนของการลงทะเบียน โดยมีรายละเอียดดังต่อไปนี้

4.2.1 เครื่องมือที่ใช้ในการพัฒนาระบบ

การพัฒนาระบบจะใช้ระบบปฏิบัติการ Ubuntu เวอร์ชัน 10.04 และภาษาซีเป็นหลัก โดยระบบที่พัฒนาขึ้นนี้มีการทำงานเฉพาะในเครือข่ายอินเทอร์เน็ต ไม่ได้มีการเชื่อมต่อไปยังเครือข่ายโทรศัพท์พื้นฐาน จึงใช้ OpenSIPS เป็น SIP Server ซึ่ง OpenSIPS ใช้ภาษาซีในการพัฒนาโปรแกรม การเลือกเครื่องมือที่ใช้เป็น User Agent จึงเลือกซอฟต์แวร์รหัสเปิดที่พัฒนาด้วยภาษาซีคือ Linphone ข้อมูลโดยสรุปมีดังนี้

1) OpenSIPS

OpenSIPS เป็นโปรแกรมประยุกต์ที่สามารถใช้เป็น SIP Registrar และ Proxy Server ในโครงสร้างพื้นฐาน VoIP ได้ โดยที่ OpenSIPS มีต้นกำเนิดมาจาก SIP Express

Router (SER) (Goncalves, 2008) ซึ่งเป็น SIP Server ที่เป็นซอฟต์แวร์รหัสเปิด เริ่มพัฒนาโดยสถาบันวิจัย FhG Fokus ในเบอร์ลิน ประเทศเยอรมัน

คุณลักษณะที่สำคัญของ OpenSIPS คือ มีความรวดเร็ว ยืดหยุ่น สามารถทำงานในสภาพแวดล้อมที่มีความสามารถในการประมวลผลต่ำได้

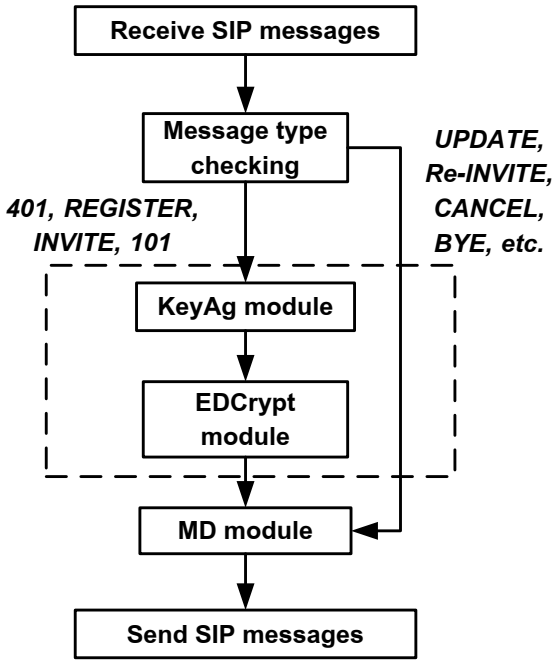
2) Linphone

คือโทรศัพท์อินเทอร์เน็ตหรือโทรศัพท์ VoIP ซึ่งสามารถใช้เพื่อติดต่อสื่อสารด้วยเสียง วีดีโอ และข้อความ กับบุคคลอื่นๆ ผ่านทางอินเทอร์เน็ต โดยใช้โพรโทคอล SIP

Linphone เป็นซอฟต์แวร์รหัสเปิด สามารถใช้งานผ่านเครื่องคอมพิวเตอร์ที่ใช้ระบบปฏิบัติการ Linux, Windows และ MacOSX หรือใช้งานผ่านโทรศัพท์เคลื่อนที่ Android, iPhone และ Blackberry

4.2.2 การพัฒนาระบบ

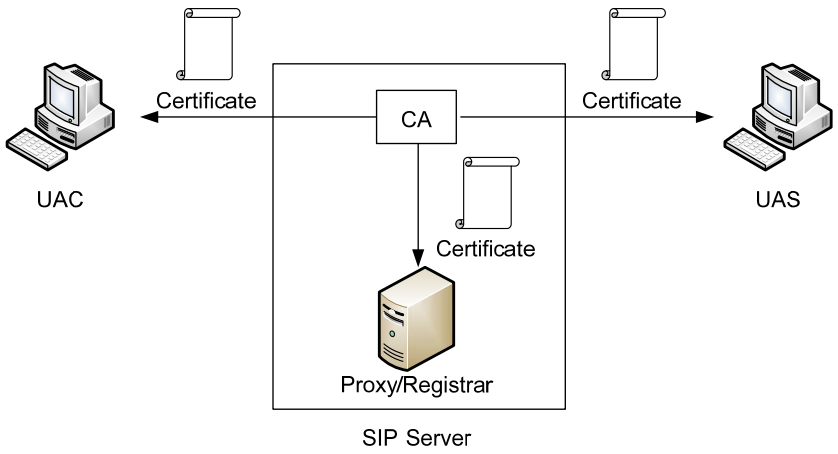
งานวิจัยนี้ได้พัฒนาเพิ่มเติมการทำงานของ SIP ให้สามารถป้องกันการโจมตีสัญญาณเชื่อมต่อได้ โดยกลไกที่พัฒนาขึ้นมาเรียกว่า SIPE-SAP ประกอบด้วยโมดูลหลัก 3 โมดูล คือ KeyAg, EDCrypt และ MD ดังภาพประกอบที่ 4.1 ถ้าข้อความ SIP ที่ต้องดำเนินการเป็นข้อความ INVITE และ REGISTER ที่มีข้อมูลประจำตัวของผู้ใช้ หรือข้อความ 101 Dialog Establishment และ 401 Unauthorized สามารถเรียกใช้โมดูล KeyAg เพื่อแลกเปลี่ยนกุญแจกันก่อน อาจมีการเรียกใช้โมดูล EDCrypt เพื่อเข้ารหัสกุญแจเซสชันที่ต้องการแลกเปลี่ยนหรือเข้ารหัสข้อมูลประจำตัวของผู้ใช้ แต่ถ้าเป็นข้อความอื่นที่ไม่ใช่ INVITE, REGISTER, 101 Dialog Establishment และ 401 Unauthorized สามารถเรียกใช้โมดูล MD ได้เลย เพราะได้มีการแลกเปลี่ยนกุญแจไว้ก่อนแล้ว การพัฒนาแต่ละโมดูลมีรายละเอียดดังนี้



ภาพประกอบที่ 4.1 กระบวนการทำงานของ SIPE-SAP

1) โมดูล KeyAg

โมดูล KeyAg ในส่วนของการแลกเปลี่ยนกุญแจ มีการนำใบรับรองมาใช้กับ SIP โดยใบรับรองใช้สำหรับกระจายกุญแจสาธารณะที่ใช้สำหรับเข้ารหัสกุญแจเซสชัน ซึ่ง OpenSIPS มีฟังก์ชันสำหรับใช้สร้าง Root Certificate อยู่แล้ว การพัฒนาระบบนี้จึงให้ OpenSIPS เป็นผู้ออกใบรับรอง (Certificate Authority: CA) โดยทำหน้าที่สร้างใบรับรองแจกจ่ายให้กับ UAC, UAS, Proxy และ Registrar ดังภาพประกอบที่ 4.2



ภาพประกอบที่ 4.2 การออกใบรับรองดิจิทัลของ SIPE-SAP

ใบรับรองที่สร้างขึ้นอยู่ในรูปแบบ X.509 (Housley และคณะ, 1999) กุญแจสาธารณะที่ได้จากใบรับรองนี้เป็นกุญแจของขั้นตอนวิธีการเข้ารหัสแบบ RSA รายละเอียดของใบรับรองมีดังนี้

- Version: เป็นข้อมูลเกี่ยวกับเวอร์ชันของมาตรฐานใบรับรอง โดยเวอร์ชันล่าสุดของมาตรฐาน X.509 ก็คือเวอร์ชัน 3

- Serial Number: เป็นเลขจำนวนเต็มที่ไม่ซ้ำกันใน CA ที่ออกใบรับรอง

- Signature Algorithm: ใช้ระบุอัลกอริทึมที่ใช้ในการในการสร้างลายเซ็น เช่น sha1WithRSAEncryption หมายถึงการสร้างลายเซ็นใช้ Hashing Algorithm ที่ชื่อ SHA1 และ Encryption Algorithm ที่ชื่อ RSA

- Issuer: เป็นชื่อของ CA ที่ทำการสร้างและเซ็นใบรับรอง

- Validity: เป็นการระบุระยะเวลาว่าใบรับรองสามารถใช้ได้ตั้งแต่เมื่อไร ถึงเมื่อไร

- Subject: เป็นชื่อของเจ้าของใบรับรองดิจิทัลนี้ และต้องเป็นเจ้าของกุญแจสาธารณะที่ระบุอยู่ในใบรับรองนี้ด้วย

- Subject Public Key Info: เป็นฟิลด์ที่เก็บกุญแจสาธารณะ โดยระบุถึงอัลกอริทึมที่ใช้กับกุญแจนี้ และพารามิเตอร์อื่นๆ เช่น ขนาดของกุญแจ

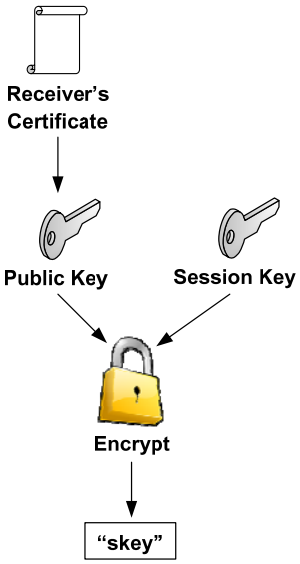
- Extension: เป็นส่วนที่มีเฉพาะในเวอร์ชัน 3 เท่านั้นเพื่อแก้ไขข้อจำกัดที่มีในเวอร์ชัน 1 และ 2 ข้อมูลที่มีอยู่ใน Extension เช่น ข้อมูลเกี่ยวกับนโยบายของใบรับรอง และข้อจำกัดเกี่ยวกับการใช้กุญแจ

- Signature: ประกอบด้วยอัลกอริทึมที่ใช้สร้างลายเซ็นอิเล็กทรอนิกส์ของ CA และส่วนของลายเซ็นอิเล็กทรอนิกส์ ซึ่งสร้างโดยการแฮชข้อมูลทั้งหมดที่ได้กล่าวมาแล้ว จากนั้นเข้ารหัสลับด้วยกุญแจส่วนตัวของ CA

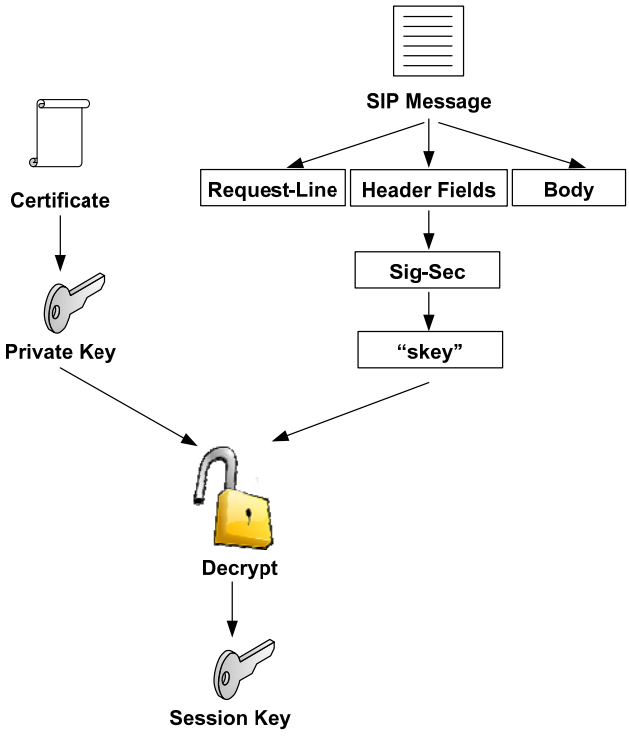
ขั้นตอนการสร้างใบรับรองจะต้องมีการระบุชื่อผู้ถือใบรับรอง ดังนั้น เพื่อให้สอดคล้องกับการใช้งาน SIP ถ้าผู้ถือใบรับรองเป็นเซิร์ฟเวอร์ ชื่อผู้ถือใบรับรองจะเป็นโดเมนเนมของของเซิร์ฟเวอร์ ส่วน UA ใช้ SIP URI เป็นชื่อผู้ถือใบรับรอง เช่น ถ้าผู้ใช้ชื่อ alice มี SIP URI คือ alice@sipserver.cs.psu.ac.th ชื่อของผู้ถือใบรับรองที่ปรากฏในฟิลด์สำหรับเก็บข้อมูลส่วนตัวของผู้ถือใบรับรอง (ฟิลด์ Subject) คือ alice@sipserver.cs.psu.ac.th

เนื่องจากข้อความ SIP ประกอบด้วย Multipurpose Internet Mail Extensions (MIME) (Rosenberg และคณะ, 2002) การส่งใบรับรองไปกับข้อความ SIP จึงอาศัย MIME Body แต่จะต้องมีการเพิ่มเฮดเดอร์ Content-Type ด้วยเพื่อแจ้งให้ปลายทาง (UAC และ UAS) ทราบถึงชนิดของข้อมูลที่ส่งมา

เมื่อต้องการแลกเปลี่ยนกุญแจเซสชัน ฝ่ายผู้ส่งจะนำกุญแจสาธารณะที่ได้จากใบรับรองของฝ่ายผู้รับมาเข้ารหัสกุญแจเซสชันที่สุ่มขึ้น ได้ข้อมูล “skey” ซึ่งจะถูกนำมาสร้างเป็นเฮดเดอร์ Sig-Sec ในข้อความ SIP ดังภาพประกอบที่ 4.3 ส่วนฝ่ายผู้รับจะนำข้อความ SIP มาสกัดจนได้ข้อมูล “skey” แล้วนำกุญแจส่วนตัวมาถอดรหัส ได้กุญแจเซสชัน ดังภาพประกอบที่ 4.4



ภาพประกอบที่ 4.3 การแลกเปลี่ยนกุญแจฝ่ายผู้ส่ง

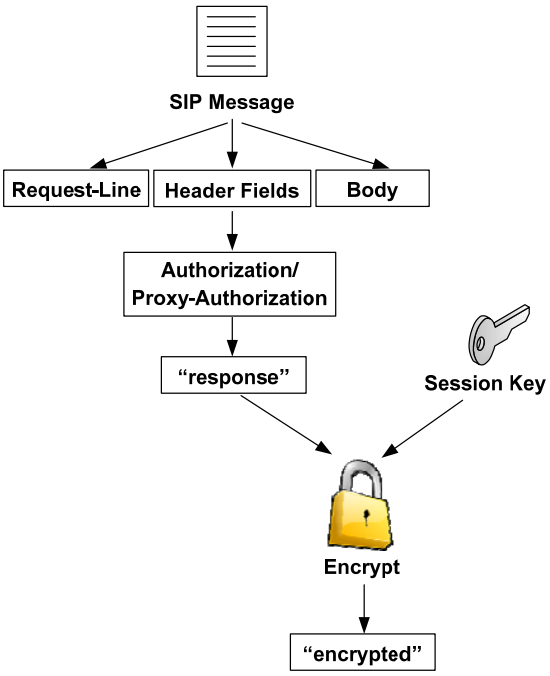


ภาพประกอบที่ 4.4 การแลกเปลี่ยนกุญแจฝ่ายผู้รับ

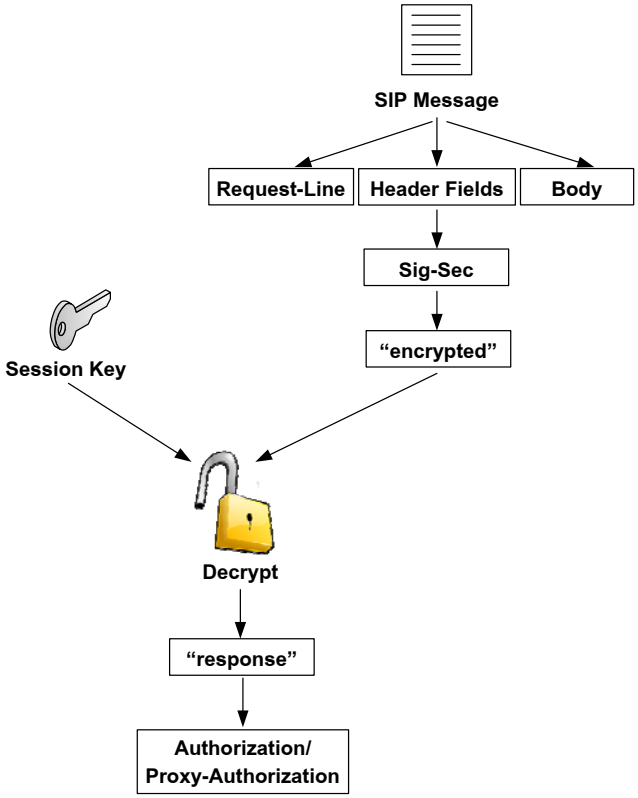
2) โมดูล EDCrypt

การพัฒนาระบบส่วนของโมดูล EDCrypt ผู้วิจัยได้ทดลองพัฒนาโดยใช้ขั้นตอนวิธี AES โดยใช้กุญแจขนาด 128 บิต ข้อมูลที่เข้ารหัสลับจะถูกแปลง (Encode) ให้อยู่ในรูปแบบ Base64 เพื่อให้เหมาะสมต่อการใช้งานกับโพรโทคอล SIP ซึ่ง Base64 ประกอบด้วยตัวอักษรภาษาอังกฤษตัวพิมพ์ใหญ่ ตัวพิมพ์เล็ก ตัวเลข “+” และ “/”

กระบวนการเข้ารหัสฝ่ายผู้ส่ง เริ่มจากนำข้อความ SIP มาสกัดเฮดเดอร์ Authorization หรือ Proxy-Authorization จนได้ข้อมูล “response” แล้วนำกุญแจเซสชันมาเข้ารหัส ได้ข้อมูล “encrypted” ซึ่งจะถูกนำมาสร้างเป็นเฮดเดอร์ Sig-Sec ในข้อความ SIP ดังภาพประกอบที่ 4.5 ทางฝ่ายผู้รับจะนำข้อความ SIP มาสกัดจนได้ข้อมูล “encrypted” แล้วนำกุญแจเซสชันมาถอดรหัส ได้ข้อมูล “response” โดยข้อมูล “response” นี้ถูกนำมาเพิ่มเข้าไปในเฮดเดอร์ Authorization หรือ Proxy-Authorization เพื่อนำไปใช้ในกระบวนการพิสูจน์ตัวตนแบบ HTTP Digest ดังภาพประกอบที่ 4.6

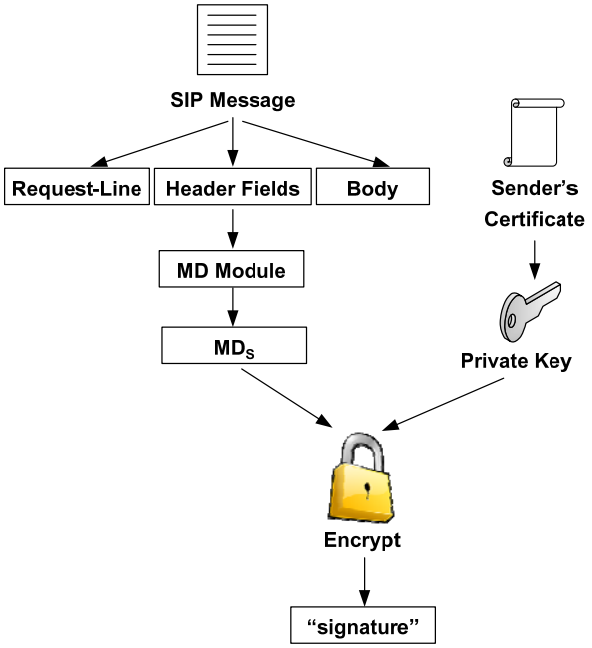


ภาพประกอบที่ 4.5 การเข้ารหัสฝ่ายผู้ส่ง

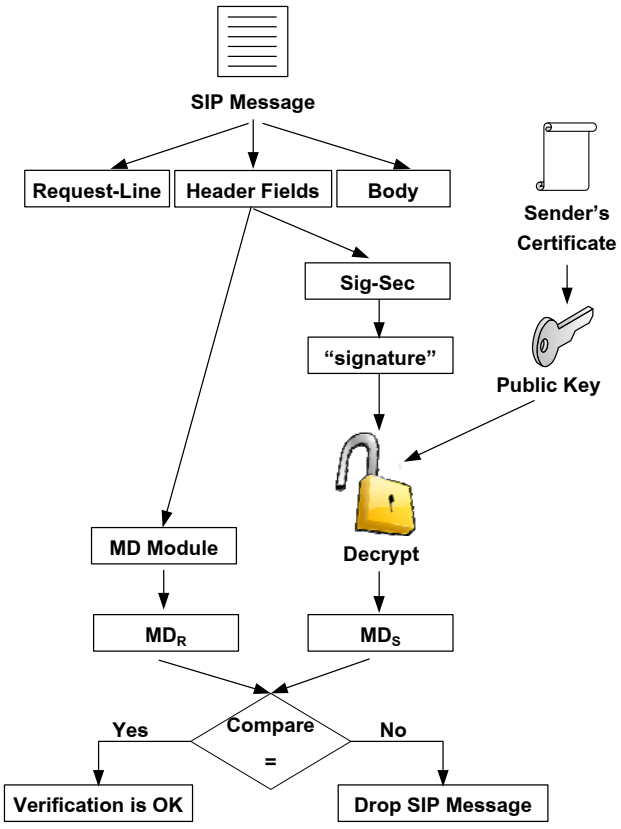


ภาพประกอบที่ 4.6 การถอดรหัสฝ่ายผู้รับ

นอกจากนี้ เมื่อต้องการสร้างลายเซ็นดิจิทัล ฝ่ายผู้ส่งจะนำเฮดเดอร์ของข้อความ SIP ได้แก่ Via (พารามิเตอร์ branch), From, To, Call-ID และ Cseq มาผ่านการย่อยข้อความโดยใช้โมดูล MD ได้ MD_S แล้วนำกุญแจส่วนตัวของผู้ส่งมาเข้ารหัส ได้ข้อมูล "signature" ซึ่งเป็นลายเซ็นดิจิทัล โดยข้อมูล "signature" ถูกนำมาใช้สร้างเฮดเดอร์ Sig-Sec ในข้อความ SIP ดังภาพประกอบที่ 4.7 สำหรับการตรวจสอบลายเซ็นดิจิทัลฝ่ายผู้รับ จะมีการนำเฮดเดอร์ของข้อความ SIP มาผ่านการย่อยข้อความเช่นเดียวกัน ได้ MD_R แล้วสกัดเฮดเดอร์ Sig-Sec จนได้ข้อมูล "signature" จากนั้น นำกุญแจสาธารณะของผู้ส่งมาถอดรหัสข้อมูล "signature" ได้ MD_S ฝ่ายผู้รับนำ MD_R และ MD_S มาเปรียบเทียบกัน ถ้ามีค่าเท่ากันจะนำข้อความ SIP มาดำเนินการตามโพรโทคอล SIP แต่ถ้าได้ค่าไม่เท่ากันให้ละทิ้งข้อความ SIP นั้นไป ดังภาพประกอบที่ 4.8



ภาพประกอบที่ 4.7 การสร้างลายเซ็นดิจิทัลฝ่ายผู้ส่ง



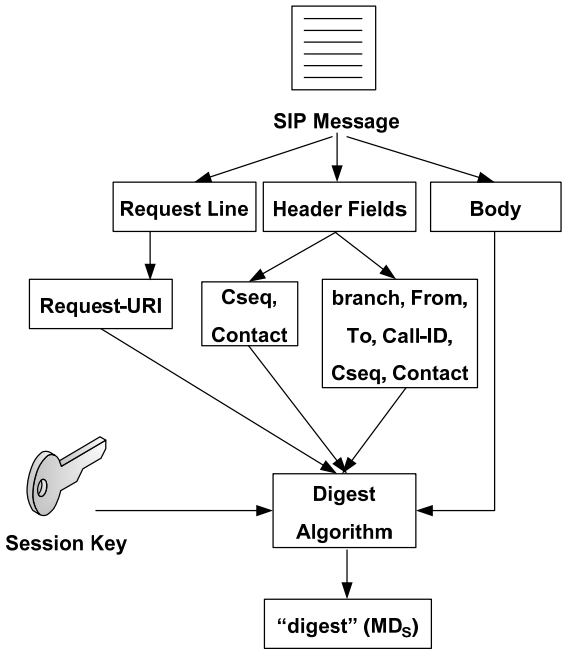
ภาพประกอบที่ 4.8 การตรวจสอบลายเซ็นดิจิทัลฝ่ายผู้รับ

2) โมดูล MD

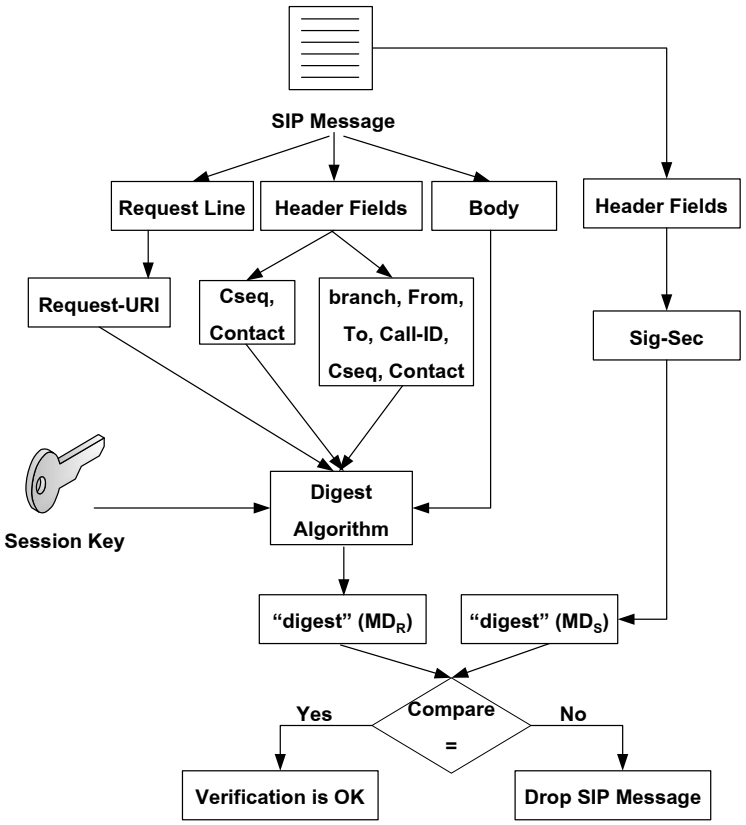
การพัฒนาระบบส่วนของโมดูล MD ใช้ขั้นตอนวิธี MD5 เมื่อย่อยข้อความ (Digest) แล้วจะได้ข้อมูลขนาด 128 บิต ข้อความ SIP แต่ละประเภทประกอบด้วยข้อมูลที่นำมาย่อยต่างกันคือ

- ข้อความ REGISTER และ INVITE ข้อมูลที่จะถูกนำมาย่อยประกอบด้วย
  - Request-URI, Cseq, Contact, SDP Body และ Session Key
- ข้อความอื่นๆ ข้อมูลที่จะถูกนำมาย่อยประกอบด้วย
  - พารามิเตอร์ branch ของ Via, From, To, Call-ID, Cseq, Contact, SDP Body และ Session Key

การย่อยข้อความฝ่ายผู้ส่ง มีการสกัดข้อมูลที่เกี่ยวข้องจากข้อความ SIP ตามประเภทของข้อความ แล้วนำมาย่อยข้อความพร้อมกับกุญแจเซสชัน ได้ข้อมูล “digest” ซึ่งถูกนำมาใช้สร้างเฮดเดอร์ Sig-Sec ในข้อความ SIP ดังภาพประกอบที่ 4.9 ฝ่ายผู้รับดำเนินการย่อยข้อความเช่นเดียวกัน ได้ MD_R แล้วสกัดเฮดเดอร์ Sig-Sec เพื่อนำข้อมูล “digest” (MD_S) มาเปรียบเทียบกับ MD_R หากได้ค่าเท่ากันจะนำข้อความ SIP มาดำเนินการตามโพรโทคอล SIP แต่ถ้าได้ค่าไม่เท่ากันให้ละทิ้งข้อความ SIP นั้นไป ดังภาพประกอบที่ 4.10



ภาพประกอบที่ 4.9 การย่อยข้อความฝ่ายผู้ส่ง



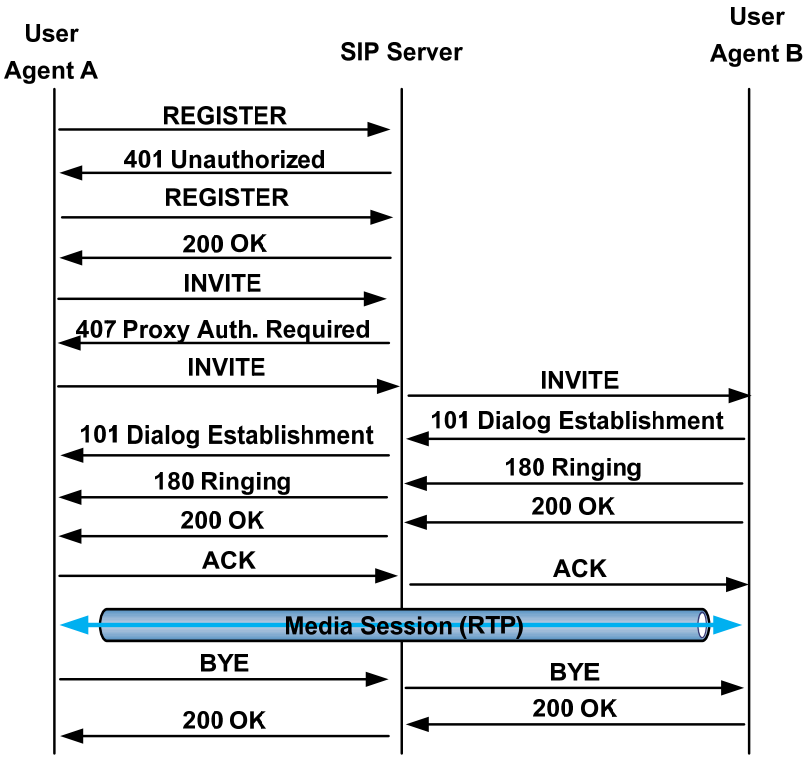
ภาพประกอบที่ 4.10 การย่อยข้อความฝ่ายผู้รับ

จากการทำงานตามปกติของ SIP คือกระบวนการลงทะเบียนและการเชื่อมต่อผู้ใช้ดังภาพประกอบที่ 4.11 หากมีการนำ SIPE-SAP มาใช้งานจะเพิ่มกระบวนการดำเนินการกับข้อความ SIP โดยแบ่งการทำงานออกเป็น 3 ส่วน คือ UAC, UAS และ SIP Server ดังภาพประกอบที่ 4.12 โดย

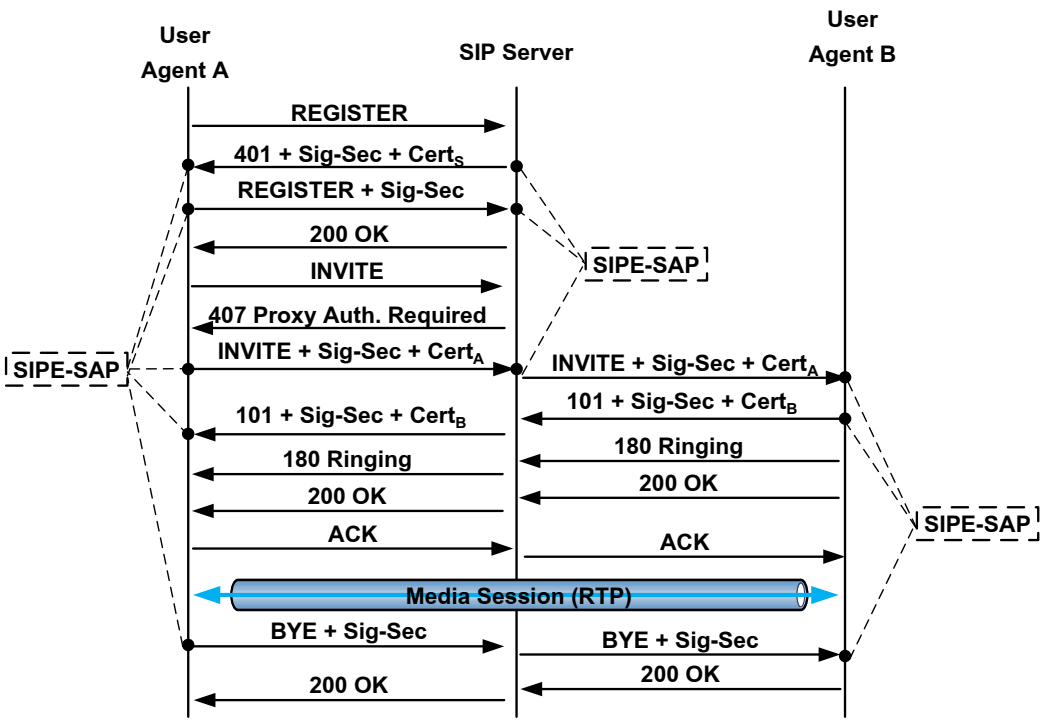
- UAC มีการทำงานเมื่อได้รับข้อความ 401 Unauthorized และ 101 Dialog Establishment และก่อนส่งข้อความ REGISTER, INVITE, BYE รวมถึง CANCEL, UPDATE และ Re-INVITE

- UAS มีการทำงานเมื่อได้รับข้อความ INVITE, BYE, CANCEL, UPDATE และ Re-INVITE และก่อนส่งข้อความ 101 Dialog Establishment

- SIP Server มีการทำงานเมื่อได้รับข้อความ REGISTER และ INVITE และก่อนส่งข้อความ 401 Unauthorized



ภาพประกอบที่ 4.11 กระบวนการลงทะเบียนและการเชื่อมต่อผู้ใช้



ภาพประกอบที่ 4.12 กระบวนการลงทะเบียนและการเชื่อมต่อผู้ใช้ที่มีการใช้งาน SIPE-SAP

โดยข้อความ SIP เหล่านี้จะประกอบด้วยเฮดเดอร์ Sig-Sec ซึ่งได้มีการนิยามเพิ่มเติมเข้าไปเพื่อรองรับกระบวนการทำงานของ SIPE-SAP แต่ละข้อความจะมีส่วนประกอบของเฮดเดอร์ Sig-Sec แตกต่างกันออกไป ได้แก่ ข้อมูลประจำตัวที่ถูกเข้ารหัสด้วยกุญแจเซสชัน (encrypted) กุญแจเซสชันที่ถูกเข้ารหัส (skey) ส่วนประกอบต่างๆ ของข้อความ SIP ที่ผ่านกระบวนการย่อยข้อมูล (digest) และลายเซนดิจิทัล (signature) นอกจากนี้บางข้อความอาจประกอบด้วยใบรับรองด้วย ดังตารางที่ 4.1

ตารางที่ 4.1 ส่วนประกอบของเฮดเดอร์ Sig-Sec และใบรับรองที่ใช้ในข้อความ SIP

SIP Message	encrypted	skey	digest	signature	ใบรับรอง
Register	✓	✓	✓	-	-
401 Unauthorized	-	-	-	✓	✓
INVITE	✓	✓	✓	✓	✓
101 Dialog Establishment	-	✓	-	✓	✓
UPDATE	-	-	✓	-	-
Re-INVITE	-	-	✓	-	-
CANCEL	-	-	✓	-	-
BYE	-	-	✓	-	-

การพัฒนากลไกการทำงานของ SIPE-SAP เพื่อตรวจสอบการโจมตีในส่วนของ SIP Server, UAC และ UAS มีขั้นตอนวิธีตามภาพประกอบที่ 4.13, 4.14 และ 4.15 ตามลำดับ

ทั้งนี้ การพัฒนาระบบมีการเรียกใช้ไลบรารี OpenSSL ซึ่งเป็นเครื่องมือที่ใช้พัฒนาโพรโทคอล TLS ในกระบวนการทำงานเกี่ยวกับวิทยาการเข้ารหัสลับ การพัฒนาระบบในส่วนของ UA ซึ่งใช้ Linphone จะมีการพัฒนาเพิ่มเติมเข้าไปในไลบรารี GNU oSIP เพราะ Linphone ใช้ oSIP เพื่อทำงานตามโพรโทคอล SIP (RFC 3261)

Algorithm sipesap_auth(msg)

1 **if** (msg is 401) **then**

1.1 Provide server's certificate and signature

2 **else if** (msg is REGISTER or INVITE) **then**

2.1 Decrypt a session key

2.2 Extract Request-URI, Cseq, Contact and SDP Body

2.3 Digest the data in step 2.2 including with the session key (MD_{SP})

2.4 Extract a digest value from a Sig-Sec (MD_{SO})

2.5 **if** (MD_{SP} != MD_{SO}) **then**

2.5.1 Decline a call

2.5.2 Go to step 3

2.6 Decrypt an encrypted data in the Sig-Sec (response)

2.7 Insert a response value to an Authorization header or a Proxy-Authenticate header

2.8 **if** (msg is INVITE) **then**

2.8.1 Remove "encrypted", "skey" and "digest" from the Sig-Sec

3 End

ภาพประกอบที่ 4.13 SIP Server, ขั้นตอนวิธี sipesap_auth

Algorithm sipesap_retry_with_auth(msg)

1 **if** (msg is 101 or 401) **then**

1.1 Verify a certificate and a signature

1.2 **if** (msg is 101) **then**

1.2.1 Decrypt a session key

2 **else if** (msg is REGISTER or INVITE) **then**

2.1 Extract and remove a response value from Authorization or Proxy-Authenticate header

2.2 Generate a session key

2.3 Encrypt the response value with the session key (encrypted)

2.4 Extract Via's branch, From, To, Call-ID, Cseq, Contact and SDP Body

2.5 Digest the data in step 2.4 including with the session key (digest)

2.6 Encrypt the session key (skey)

2.7 Create a Sig-Sec header using "encrypted", "skey" and "digest"

2.8 **if** (msg is INVITE) **then**

2.8.1 Provide UAC's certificate and signature

3 **else if** (msg is CANCEL, BYE, Re-INVITE or UPDATE) **then**

3.1 Extract Via's branch, From, To, Call-ID, Cseq, Contact and SDP Body

3.2 Digest the data in step 3.1 including with the session key (digest)

3.3 Create a Sig-Sec header using "digest"

4 End

ภาพประกอบที่ 4.14 UAC, ขั้นตอนวิธี sipesap_retry_with_auth

Algorithm sipesap_process_digest(msg)

1 **if** (msg is 101) **then**

1.1 Provide UAS's certificate and signature

1.2 Generate a session key

1.3 Encrypt the session key (skey)

1.4 Create a Sig-Sec header using "skey"

2 **else if** (msg is INVITE) **then**

2.1 Verify a certificate and a signature

3 **else if** (msg is CANCEL, BYE, Re-INVITE or UPDATE) **then**

3.1 Extract Via's branch, From, To, Call-ID, Cseq, Contact and SDP Body

3.2 Digest the data in step 3.1 including with the session key (MD_{CP})

3.3 Extract a digest value from a Sig-Sec (MD_{CO})

3.4 **if** (MD_{CP} != MD_{CO}) **then**

3.4.1 Decline a call

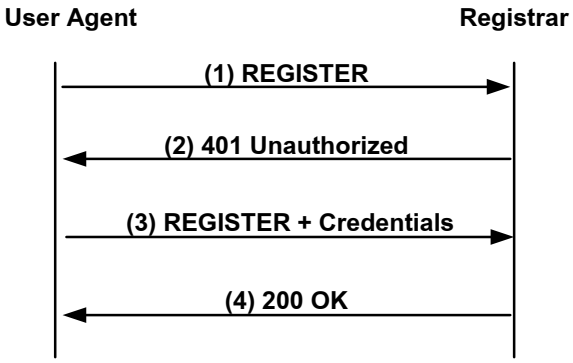
4 End

ภาพประกอบที่ 4.15 UAS, ขั้นตอนวิธี sipesap_process_digest

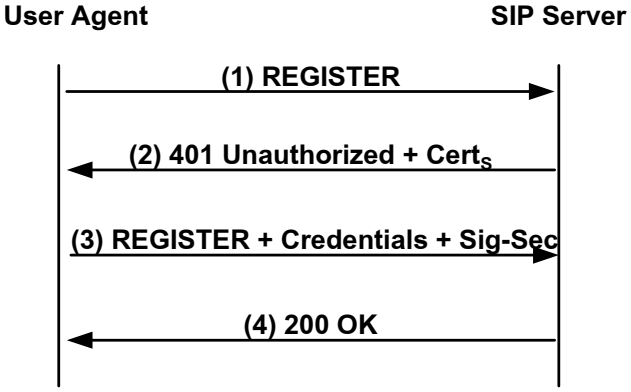
4.2.3 ตัวอย่างผลการพัฒนาระบบ

โดยปกติในขั้นตอนการลงทะเบียน UA จะส่งคำร้องขอซึ่งเป็นข้อความ REGISTER ไปยังเซิร์ฟเวอร์ เซิร์ฟเวอร์เรียกร้องให้มีการพิสูจน์ตัวตนโดยการตอบกลับด้วยข้อความ 401 Unauthorized จากนั้น UA ส่งข้อความ REGISTER พร้อมด้วยข้อมูลประจำตัวไปยังเซิร์ฟเวอร์อีกครั้ง เพื่อให้เซิร์ฟเวอร์ดำเนินการพิสูจน์ตัวตน ดังภาพประกอบที่ 4.16

กระบวนการลงทะเบียนที่มีการใช้กลไก SIPE-SAP แสดงดังภาพประกอบที่ 4.17 จากตัวอย่างเมื่อเซิร์ฟเวอร์ร้องขอให้มีการลงทะเบียนจะส่งใบรับรองมาด้วย ตัวอย่างข้อความ (2) 401 Unauthorized + Cert แสดงดังภาพประกอบที่ 4.18 UA ต้องส่งข้อความลงทะเบียนไปใหม่พร้อมด้วยข้อมูลประจำตัวของผู้ใช้ โดยกลไก SIPE-SAP จะนำค่า response เข้ารหัสและมีการย่อยข้อความเพื่อสร้างเฮดเดอร์ Sig-Sec ข้อความ (3) REGISTER ที่ได้มีลักษณะดังภาพประกอบที่ 4.19 บรรทัดที่ 9 คือเฮดเดอร์ Sig-Sec ที่สร้างขึ้น เมื่อสังเกตบรรทัดที่ 8 จะพบว่าค่า response ถูกลบออกจากเฮดเดอร์ Authorization เพื่อป้องกันการนำข้อมูลไปใช้ และภาพประกอบที่ 4.20 คือโปรแกรม Linphone มีการใช้ SIPE-SAP ในขั้นตอนการลงทะเบียน



ภาพประกอบที่ 4.16 กระบวนการลงทะเบียน (Geneiatakis และคณะ, 2006)



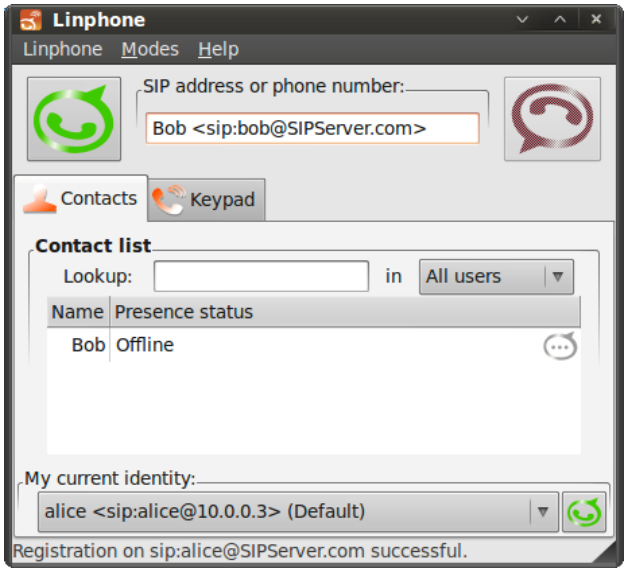
ภาพประกอบที่ 4.17 ตัวอย่างกระบวนการลงทะเบียนที่มีการใช้กลไก SIPE-SAP

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 10.0.0.3:5060;rport=5060;branch=z9hG4bK854375514
From: alice <sip:alice@SIPServer.com>;tag=1791746489
To: alice <sip:alice@SIPServer.com>;tag=96c42fb4eaeb131e386501201d85a818.651b
Call-ID: 1285097432
CSeq: 1 REGISTER
Content-Type: text/plain
WWW-Authenticate: Digest realm="SIPServer.com",
nonce="506d8e9a000000064e496c0df7c8c12b4b868934b5cb8420"
Server: OpenSIPS (1.6.4-2-tls (i386/linux))
Content-Length: 1070

-----BEGIN CERTIFICATE-----
MIIC6TCCAdGgAwIBAgIBATANBgkqhkiG9w0BAQUFADBpMRIwEAYDVQQDFAlZb3Vy
X05BTUUxEzARBgNVBAgUCllvdXJfU1RBVEUxCzAJBgNVBAYTAkNPMRkwFwYJKoZI
hvcNAQkBFgpZT1VSX0VNQUlMMRYwFAYDVQQKFA1ZT1VSX09SR19OQU1FMB4XDTEy
MTAwNDEzMDc0NFoXDTEzMTAwNDEzMDc0NFowgb8xCzAJBgNVBAYTAlhZMRMwEQYD
VQQIEwpTb21lIFN0YXRlMSMwIQYDVQQKExpNeSBMYXJnZSBPcmdhbml6YXRpb24g
TmFtZTEpMCcGA1UECxMgTXkgU3VidW5pdCBvZiBMYXJnZSBPcmdhbml6YXRpb24x
HzAdBgNVBAMTFnNvbWVuYW1lLnNvbWV3aGVyZS5jb20xKjAoBgkqhkiG9w0BCQEW
G3Jvb3RAc29tZW5hbWUuc29tZXdoZXJlLmNvbTBcMA0GCSqGSIb3DQEBAQUAA0sA
MEgCQQDGHHX2+hd2APSwVHEaPw/bVDGhYaCccxcsA3HmCX2Mn3lnoxwGziyFY6Fv
xqHoyJH1J+VZnSPUUfYNscaAlLR7AgMBAAGjDTALMAkGA1UdEwQCMAAwDQYJKoZI
hvcNAQEFBQADggEBAJR90ZHUYIo3tn2RMsqjTkjtqOjRo7CjLAb8+mwV8ID6qvZt
mbfv/mwVhdLaoYIanqczLPhPV605+bOwOaU1H02CPy2fLIEj9ch4/jX9gD9QgbY6
aWtR8TspP/Hm9uTf3RFlcUDZcrHUrnAPdNn6sEmrg9umWgHJ6G0YpZLxoDPmoWp+
g/M7VdK+EtBlTlSJLWX4Fm6jCeIarIMbidv4eACj0PQOAy/HLQodbbW0VV9Dhnb4
y6cg761e5B9MSt5nnMJlTiMfr3UnzjlvvruBdgio0jAjO2+qS/aiclDNMgVrkKmk
s5eNL/1WJo0/bRrWm1sFQUvmNGIFamT0DDV1KiA=
-----END CERTIFICATE-----
```

ภาพประกอบที่ 4.18 ตัวอย่างข้อความ Unauthorized

1. REGISTER sip:alice@SIPServer.com SIP/2.0
2. Via: SIP/2.0/UDP 10.0.0.3:5060;rport;branch=z9hG4bK1055313841
3. From: alice <sip:alice@SIPServer.com>;tag=1791746489
4. To: alice <sip:alice@SIPServer.com>
5. Call-ID: 1285097432
6. CSeq: 2 REGISTER
7. Contact: <sip:alice@10.0.0.3:5060;line=24452eb21cfdafc>
8. Authorization: Digest username="alice", realm="SIPServer.com",
   nonce="506d8e9a000000064e496c0df7c8c12b4b868934b5cb8420",
   uri="sip:alice@SIPServer.com", algorithm=MD5
9. **Sig-Sec: encrypted=gsKJITWwyFIU6OH3J3I6cweWmMcbW0OEL8LCOA1YaCMNKpv8dj**
   uurXi9/gZOg1Ro, rkey=XFf3ToY1nQ4tislibTJz+LVSsbpXZGTk9cu3P4fz36pztq
   eK8sBbGhfuPKnLUuxw3tWyUleKkecfUMXbUKiVUw==, ed_algo=AES,
   digest=6496a65c6d9ec969571734eb59b2ecc3, md_algo=MD5
10. Max-Forwards: 70
11. User-Agent: Linphone/3.3.0 (eXosip2/3.3.0)
12. Expires: 3600
13. Content-Length: 0

ภาพประกอบที่ 4.19 ตัวอย่างข้อความ (3) REGISTER



ภาพประกอบที่ 4.20 โปรแกรม Linphone มีการใช้ SIPE-SAP ในขั้นตอนการลงทะเบียน

4.3 การทดสอบระบบ

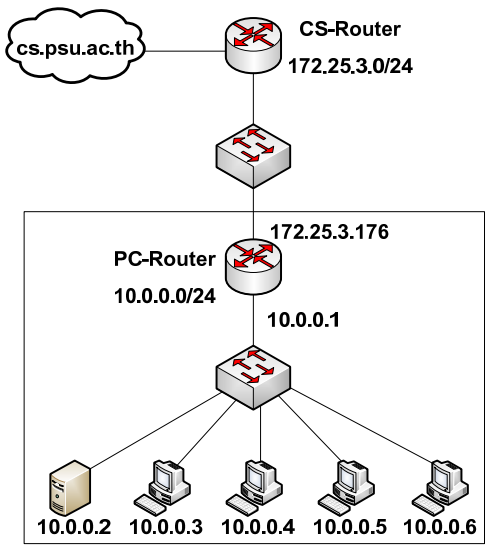
ในหัวข้อนี้จะกล่าวถึงการทดสอบระบบ โดยแบ่งเป็นการทดสอบประสิทธิภาพของระบบ และการทดสอบการโจมตีต่อกลไกที่ได้พัฒนาขึ้น รายละเอียดของการทดสอบระบบมีดังนี้

4.3.1 สภาพแวดล้อมในการทดสอบระบบ

การทดสอบระบบดำเนินการภายใต้ระบบปิด และใบรับรองออกโดยผู้ให้บริการใบรับรองที่ไม่ถูกโจมตีโดยผู้บุกรุก มีการใช้เครื่องคอมพิวเตอร์ในการทดสอบจำนวน 6 เครื่อง รายละเอียดของเครื่องคอมพิวเตอร์ที่ใช้ แสดงดังตารางที่ 4.2 และเครือข่ายที่ใช้ในการทดสอบระบบแสดงดังภาพประกอบที่ 4.21

ตารางที่ 4.2 คุณลักษณะและระบบปฏิบัติการของเครื่องคอมพิวเตอร์ที่ใช้

เครื่องคอมพิวเตอร์	คุณลักษณะ	ระบบปฏิบัติการ
PC-Router	CPU Intel Core 2 Quad 2.40 GHz, RAM 2 GB	FreeBSD
10.0.0.2	CPU Intel Core 2 Quad 2.40 GHz, RAM 2 GB	LINUX
10.0.0.3	CPU Intel Core 2 Duo 3.00 GHz, RAM 2 GB	LINUX
10.0.0.4	CPU Intel Core 2 Duo 1.86 GHz, RAM 2 GB	LINUX
10.0.0.5	CPU Intel Core 2 Duo 2.40 GHz, RAM 1 GB	LINUX
10.0.0.6	CPU Intel Core 2 Duo 2.40 GHz, RAM 1 GB	LINUX



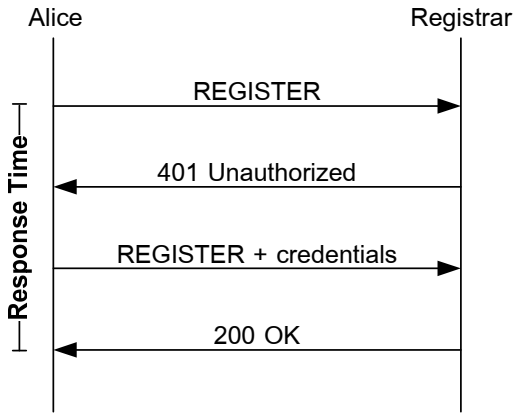
ภาพประกอบที่ 4.21 สภาพแวดล้อมในการทดสอบระบบ

4.3.2 การทดสอบประสิทธิภาพ

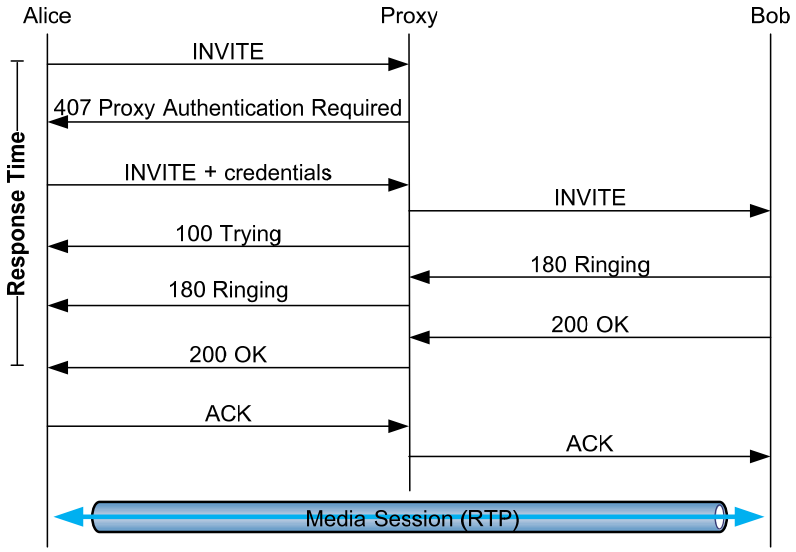
เครื่องมือที่ใช้ในการทดสอบประสิทธิภาพคือ SIPp (Gayraud และคณะ, 2012) ซึ่งเป็นเครื่องมือที่ใช้สำหรับทดสอบการทำงานของ SIP Server โดยทางนักพัฒนาของ OpenSIPS ก็ใช้เครื่องมือนี้ในการวัดประสิทธิภาพของเซิร์ฟเวอร์

SIPp สามารถสร้างคำร้องขอจำนวนมากพร้อมทั้งบันทึกระยะเวลาเวลาที่ใช้ในการรับส่งข้อความตามที่ต้องการได้ โดย SIPp ใช้ XML ในการกำหนดรูปแบบของข้อความ SIP และกำหนดลำดับการรับส่งข้อความระหว่าง UAC และ UAS ส่วนการทำงานอื่นๆ ของ SIPp มีการพัฒนาโดยใช้ภาษา C ร่วมกับ C++

การทดสอบประสิทธิภาพแบ่งเป็น 2 ส่วนคือ การลงทะเบียน และการเชื่อมต่อการโทร และใช้ระยะเวลาการตอบสนอง (Response Times) เพื่อวัดประสิทธิภาพของระบบ โดยระยะเวลาการตอบสนองสำหรับขั้นตอนการลงทะเบียนถูกบันทึกตั้งแต่การเริ่มส่งข้อความ REGISTER จนกระทั่งได้รับข้อความ OK ตอบกลับมา ดังภาพประกอบที่ 4.22 และระยะเวลาการตอบสนองสำหรับขั้นตอนการเชื่อมต่อการโทรถูกบันทึกตั้งแต่การเริ่มส่งข้อความ INVITE จนกระทั่งได้รับข้อความ OK ตอบกลับมาดังภาพประกอบที่ 4.23



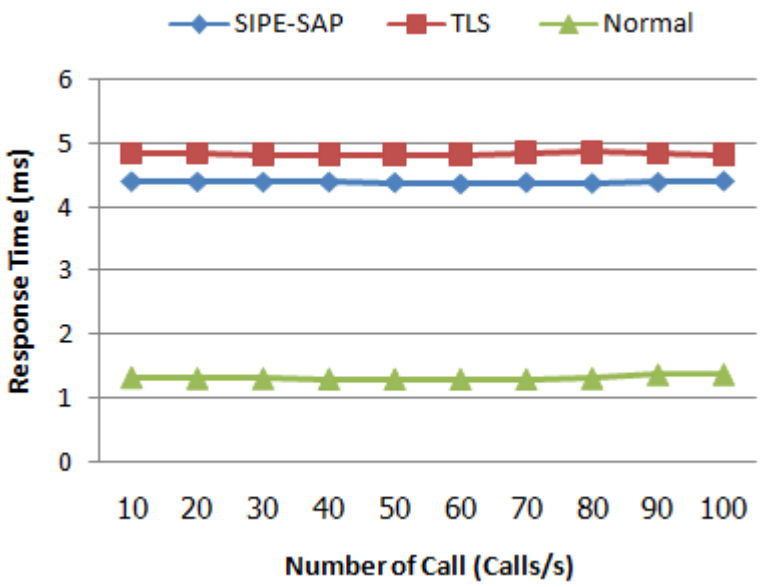
ภาพประกอบที่ 4.22 การวัดระยะเวลาการตอบสนองสำหรับขั้นตอนการลงทะเบียน



ภาพประกอบที่ 4.23 การวัดระยะเวลาการตอบสนองสำหรับขั้นตอนการเชื่อมต่อการโทร

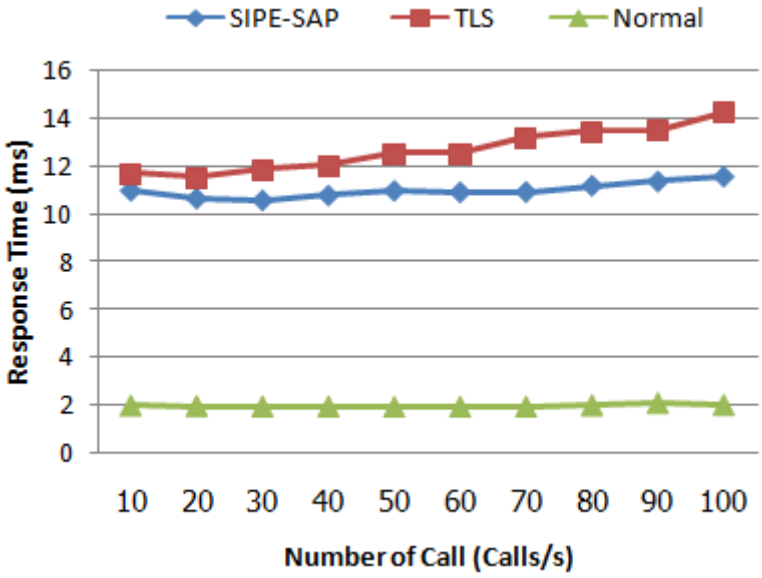
การทดสอบประสิทธิภาพจะใช้จำนวนการโทรที่แตกต่างกันในการส่งแต่ละครั้ง โดยเริ่มจากครั้งละ 10 Call จนถึง 100 Call มีการทดสอบทั้งหมดจำนวน 5 ครั้ง แล้วนำมาหาค่าเฉลี่ยเพื่อสร้างกราฟเปรียบเทียบระยะเวลาการตอบสนองเฉลี่ย เมื่อมีการพิสูจน์ตัวตนโดยใช้ HTTP Digest ตามการทำงานปกติของ SIP การพิสูจน์ตัวตนโดยใช้ HTTP Digest ร่วมกับ SIPE-SAP และการใช้งาน TLS เพื่อเข้ารหัสข้อความ SIP

ระยะเวลาการตอบสนองเฉลี่ยในขั้นตอนการลงทะเบียนของ SIPE-SAP เปรียบเทียบกับการทำงานของตามปกติของ SIP ที่มีการพิสูจน์ตัวตนโดยใช้ HTTP Digest และการใช้งาน TLS แสดงดังภาพประกอบที่ 4.24 โดยระยะเวลาการตอบสนองเฉลี่ยจากการใช้ SIPE-SAP ร่วมกับ HTTP Digest คือ 4.39 มิลลิวินาที ระยะเวลาการตอบสนองเฉลี่ยจากการใช้ SIP ที่มีการพิสูจน์ตัวตนแบบ HTTP Digest คือ 1.33 มิลลิวินาที และระยะเวลาการตอบสนองเฉลี่ยจากการใช้ TLS คือ 4.84 มิลลิวินาที สามารถสรุปได้ว่า ระยะเวลาการตอบสนองเฉลี่ยของการใช้ SIPE-SAP เพิ่มขึ้นประมาณ 3.06 มิลลิวินาที หรือคิดเป็น 3.3 เท่า เมื่อเปรียบเทียบกับโพรโทคอลปกติ และลดลงประมาณ 0.45 มิลลิวินาที หรือคิดเป็น 9.28% เมื่อเปรียบเทียบกับ TLS



ภาพประกอบที่ 4.24 ระยะเวลาการตอบสนองเฉลี่ยในการลงทะเบียนด้วย SIPE-SAP

การเปรียบเทียบระยะเวลาการตอบสนองเฉลี่ยสำหรับขั้นตอนการเชื่อมต่อการโทร แสดงดังภาพประกอบที่ 4.25 โดยระยะเวลาการตอบสนองเฉลี่ยจากการใช้ SIPE-SAP ร่วมกับ HTTP Digest คือ 11.01 มิลลิวินาที ระยะเวลาการตอบสนองเฉลี่ยจากการใช้ SIP ที่มีการพิสูจน์ตัวตนแบบ HTTP Digest คือ 1.99 มิลลิวินาที และระยะเวลาการตอบสนองเฉลี่ยจากการใช้ TLS คือ 12.65 มิลลิวินาที สามารถสรุปได้ว่า ระยะเวลาการตอบสนองเฉลี่ยของการใช้ SIPE-SAP เพิ่มขึ้นประมาณ 9.02 มิลลิวินาที หรือคิดเป็น 5.53 เท่า เมื่อเปรียบเทียบกับโพรโทคอลปกติ และลดลงประมาณ 1.64 มิลลิวินาที หรือคิดเป็น 12.96% เมื่อเปรียบเทียบกับ TLS



ภาพประกอบที่ 4.25 ระยะเวลาการตอบสนองเฉลี่ยในการเชื่อมต่อการโทรด้วย SIPE-SAP

จากกราฟแสดงระยะเวลาการตอบสนองเฉลี่ยในการเชื่อมต่อการโทรด้วย SIPE-SAP เมื่อมีจำนวนการโทร 20 และ 30 Call ใช้เวลาการตอบสนองเฉลี่ยน้อยกว่าจำนวน 10 Call ส่งผลให้กราฟตกลงมาเล็กน้อย ซึ่งเป็นผลมาจากการกำหนดค่าจำนวน Process ของเซิร์ฟเวอร์ โดยจากการทดลองนี้มีการกำหนดจำนวน Process เป็น 32 เพื่อรองรับการทำงานแบบ Stateful และกระบวนการพิสูจน์ตัวตนของระบบ ดังนั้น หากกำหนดจำนวน Process ที่เหมาะสมก็จะทำให้ระยะเวลาการตอบสนองเฉลี่ยน้อยลง

แม้ว่าการใช้ SIPE-SAP ร่วมกับ HTTP Digest ใช้เวลาในการทำงานมากกว่าการใช้ SIP ที่มีการพิสูจน์ตัวตนแบบ HTTP Digest แต่ก็มีความปลอดภัยมากกว่า เพราะในกลไก SIPE-SAP จะมีกระบวนการเข้ารหัสและการย่อยข้อความรวมอยู่ ทำให้ผู้บุกรุกโจมตีได้ยากขึ้น

การเปรียบเทียบพบว่า SIPE-SAP มีระยะเวลาการตอบสนองเฉลี่ยดีกว่า TLS เนื่องจาก TLS มีการเข้ารหัสข้อความ SIP ที่ส่งทั้งข้อความ ซึ่งบางเฮดเดอร์ไม่จำเป็นต้องถูกเข้ารหัส เช่น Via ในขณะที่ SIPE-SAP มีการเข้ารหัสเฉพาะค่า response ที่อยู่ในเฮดเดอร์ Authorization หรือ Proxy-Authorization เท่านั้น นอกจากนี้ ก่อนเริ่มต้นการเชื่อมต่อด้วย SIP การใช้งาน TLS ต้องแลกเปลี่ยนกุญแจที่ใช้ร่วมกันและการตรวจสอบใบรับรองก่อน แต่ SIPE-SAP มีการแลกเปลี่ยนกุญแจที่ใช้ร่วมกันและการตรวจสอบใบรับรอง ในขณะที่มีการเชื่อมต่อด้วย SIP ส่งผลให้ TLS ใช้เวลาในการเชื่อมต่อมากกว่า อย่างไรก็ดี TLS มีความปลอดภัยมากกว่า SIPE-SAP เนื่องจากมีการรักษาความลับให้กับข้อความ SIP ที่ส่งทั้งข้อความ ในขณะที่ SIPE-SAP รักษาความลับเฉพาะค่า response เท่านั้น

นอกจากนี้ SIPE-SAP ช่วยป้องกันรหัสผ่านของผู้ใช้ได้เช่นเดียวกับ TLS และใช้เพียงแค่การย่อยข้อความสำหรับตรวจสอบว่ามีการปลอมแปลงหรือแก้ไขข้อความเพื่อโจมตีสัญญาณเชื่อมต่อหรือไม่

4.3.3 การทดสอบการโจมตี

การโจมตีผู้วิจัยใช้ Ettercap (Ornaghi และคณะ, 2012) ซึ่งเป็นชุดเครื่องมือสำหรับการโจมตีแบบแบบ Man-in-the-Middle (MITM) ที่ทำงานบน LINUX สามารถใช้เพื่อวิเคราะห์โพรโทคอลของเครือข่าย สามารถดักจับข้อมูลที่ส่งผ่านเครือข่าย รวมถึงรหัสผ่านและยังสามารถดำเนินกับข้อมูลที่ดักจับได้ด้วย เช่น การสกัดกั้นแพ็กเก็ต (Intercept)

Ettercap จะทำงานในรูปแบบ “Promiscuous Mode” สามารถจับคู่หมายเลข MAC ของตนเองกับหมายเลข IP ของเป้าหมาย (ARP Poisoning) ได้ นอกจากนี้ยังสนับสนุนการใช้งานปลั๊กอิน ทำให้สามารถขยายความสามารถของโปรแกรมด้วยการเพิ่มปลั๊กอินใหม่ๆ

การทดสอบการโจมตีได้ทดสอบทั้งการโจมตีไปยังการสร้างการเชื่อมต่อของ Linphone และ SIPp แต่ในวิทยานิพนธ์นี้แสดงเฉพาะการโจมตีของ SIPp เนื่องจากโปรแกรม Linphone ยังไม่สามารถส่งคำร้องขอ UPDATE และ Re-INVITE ได้

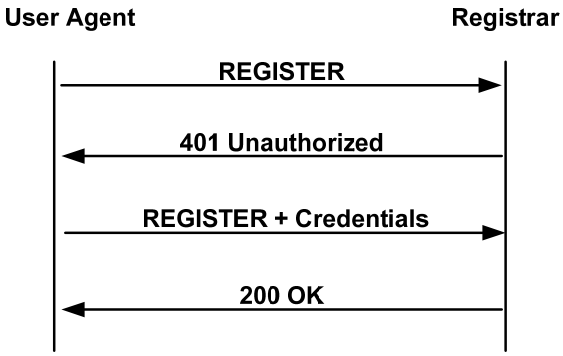
จากที่ได้กล่าวถึงเฮดเดอร์ของ SIP ที่เกี่ยวข้องกับการโจมตีสัญญาณเชื่อมต่อและมีการทดสอบการโจมตีในบทที่ 3 ในบทนี้จะกล่าวถึงการทดสอบการโจมตี SIP ที่ใช้กลไก SIPE-SAP โดยสามารถติดตามการทำงานตามปกติของ SIP และการโจมตีโดยไม่ใช้กลไก SIPE-SAP ได้ในบทที่ 3 การทดสอบการโจมตีมีรายละเอียดดังต่อไปนี้

1) Registration Hijacking

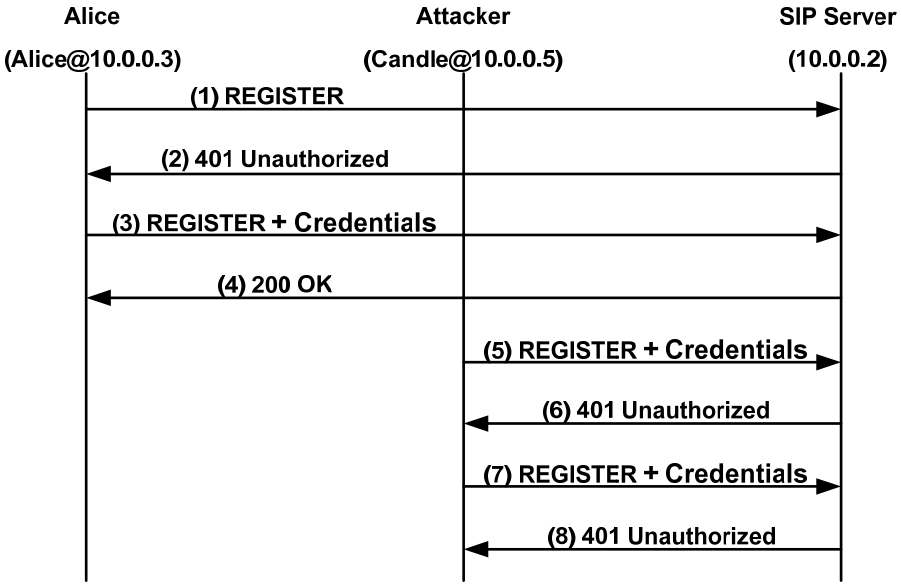
โดยปกติในขั้นตอนการลงทะเบียน UA จะส่งคำร้องขอซึ่งเป็นข้อความ REGISTER ไปยังเซิร์ฟเวอร์ เซิร์ฟเวอร์เรียกร้องให้มีการพิสูจน์ตัวตนโดยการตอบกลับด้วยข้อความ 401 Unauthorized จากนั้น UA ส่งข้อความ REGISTER พร้อมด้วยข้อมูลประจำตัวไปยังเซิร์ฟเวอร์อีกครั้ง เซิร์ฟเวอร์ใช้ข้อมูลประจำตัวนี้ในตรวจสอบตัวตน หากการพิสูจน์ตัวตนถูกต้องจะตอบกลับด้วยข้อความ 200 OK ดังภาพประกอบที่ 4.26

การโจมตีแบบ Registration Hijacking สามารถเกิดขึ้นหลังจากที่มีการลงทะเบียนเรียบร้อยแล้วดังภาพประกอบที่ 4.27 การโจมตีที่เกิดขึ้นครั้งแรกใช้ข้อความ (5) REGISTER (รายละเอียดดูภาพประกอบที่ 4.28) มีวัตถุประสงค์เพื่อยกเลิกการลงทะเบียน ข้อความนี้จะมีเฮดเดอร์ Expires และหมายเลขลำดับของเฮดเดอร์ CSeq ที่แตกต่างจากข้อความ (3) REGISTER ซึ่งกลไก SIPE-SAP มีการใช้เฮดเดอร์ CSeq ในการพิสูจน์ตัวตนและตรวจสอบความถูกต้องสมบูรณ์ของข้อมูลด้วย ดังนั้น เซิร์ฟเวอร์สามารถตรวจสอบได้ว่าข้อความนี้ถูกแก้ไขและจะร้องขอให้มีการพิสูจน์ตัวตนใหม่โดยส่งข้อความ (6) 401 Unauthorized กลับไป แต่ถ้าผู้บุกรุกไม่แก้ไข CSeq กระบวนการทำงานของ SIP จะตรวจสอบได้ว่าหมายเลขลำดับของ CSeq ไม่ถูกต้องและส่งข้อความ 400 Bad Request (รายละเอียดดูภาพประกอบที่ 4.29) ตอบกลับไปแทน

การโจมตีครั้งที่ 2 คือการลงทะเบียนด้วยตำแหน่งที่อยู่ของผู้บุกรุก ซึ่งจำเป็นจะต้องมีการแก้ไขเฮดเดอร์ Contact และแก้ไขหมายเลขลำดับของเฮดเดอร์ CSeq เซิร์ฟเวอร์จึงสามารถตรวจสอบได้เช่นเดียวกันว่ามีการโจมตี



ภาพประกอบที่ 4.26 กระบวนการลงทะเบียน (Geneiatakis และคณะ, 2006)



ภาพประกอบที่ 4.27 ตัวอย่าง Registration Hijacking

```
REGISTER sip:sipserver.cs.psu.ac.th SIP/2.0
Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1549-1-2
From: <sip:alice@sipserver.cs.psu.ac.th>;tag=1
To: <sip:alice@sipserver.cs.psu.ac.th>
Call-ID: 1-1549@10.0.0.3
CSeq: 3 REGISTER
Contact: <sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060>
Authorization: Digest username="alice", realm="sipserver.cs.psu.ac.th",
        nonce="513c6a2e8a449eccc28cb141e73ef1fd5f1bb910", uri="sip:10.0.0.2:5060",
        algorithm=MD5
Max-Forwards: 70
Expires: 0
User-Agent: SIPp/Linux
Sig-Sec: encrypted=1b6ersAcBGlgjsqFQLgXp7dw/0Il/bNB1k6fFCriEyH0LRJhPpzUvD3j3EsNKo34,
        skey=eSF3EzQC0Ff+Z9OxfiY4jPhr+Ud6jfy02uIovbqZXShjNsoMZY5ijjSgy0rxbQjBPJH
        JDATG24c1lGAvkdFUUQ==, ed_algo=AES,
        digest=6ca64b0a9a38afe809244928e6e71adf, md_algo=MD5
Content-Length: 0
```

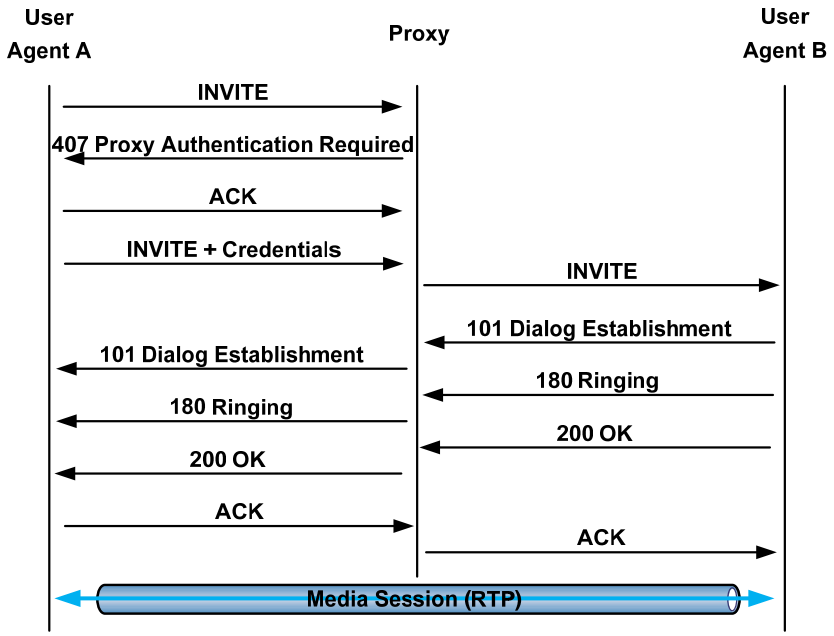
ภาพประกอบที่ 4.28 Registration Hijacking: (5) REGISTER

```
SIP/2.0 400 Bad Request
Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1540-1-2;
        received=10.0.0.5
From: <sip:alice@sipserver.cs.psu.ac.th>;tag=1
To: <sip:alice@sipserver.cs.psu.ac.th>;tag=96c42fb4eaeb131e386501201d85a818.bffb
Call-ID: 1-1540@10.0.0.3
CSeq: 2 REGISTER
Contact: <sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060>;expires=1759
P-Registrar-Error: Invalid CSeq number
Server: OpenSIPS (1.6.4-2-tls (i386/linux))
Content-Length: 0
```

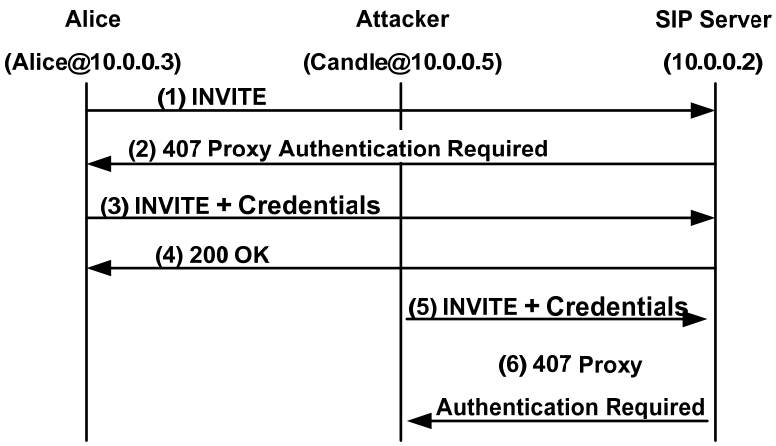
ภาพประกอบที่ 4.29 Registration Hijacking: 400 Bad Request

2) Invite Replay Billing Attack

ขั้นตอนการเชื่อมต่อผู้ใช้ตามปกติแสดงดังภาพประกอบที่ 4.30 การโจมตีเริ่มขึ้นเมื่อผู้บุกรุกมีการดักจับข้อความ (3) INVITE ดังภาพประกอบที่ 4.31 แล้วแก้ไข Request-URI หมายเลขลำดับของ CSeq และหมายเลขไอพีในส่วนของข้อมูลที่ใช้อธิบายเซสชัน ได้เป็นข้อความ (5) INVITE (รายละเอียดดูภาพประกอบที่ 4.32) แล้วส่งไปยังเซิร์ฟเวอร์ การตรวจสอบข้อมูล digest จากเฮดเดอร์ Sig-Sec จะทำให้เซิร์ฟเวอร์ทราบได้ว่าข้อความนี้ถูกเปลี่ยนแปลงแก้ไขและมีการร้องขอให้มีการพิสูจน์ตัวตนใหม่



ภาพประกอบที่ 4.30 การเชื่อมต่อผู้ใช้ใน SIP



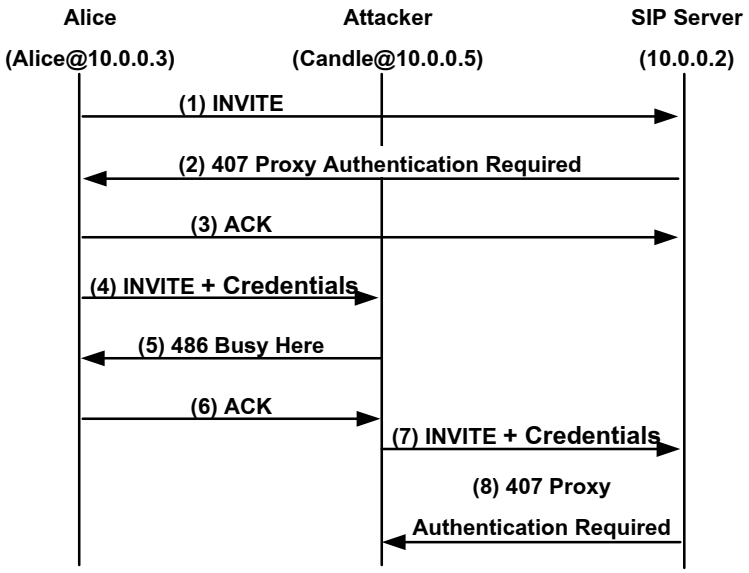
ภาพประกอบที่ 4.31 ตัวอย่าง Invite Replay Billing Attack

```
INVITE sip:dan@sipserver.cs.psu.ac.th SIP/2.0
Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1596-1-3
From: <sip:alice@sipserver.cs.psu.ac.th>;tag=1
To: <sip:bob@sipserver.cs.psu.ac.th>
Call-ID: 1-1596@10.0.0.3
CSeq: 3 INVITE
Contact: <sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060>
Proxy-Authorization: Digest username="alice", realm="sipserver.cs.psu.ac.th",
        nonce="514b0296cd669d01aaf33e25464daedb35794e77", uri="sip:10.0.0.2:5060",
        algorithm=MD5
Content-Type: multipart/mixed; boundary=boundary1
Mime-Version: 1.0
Max-forwards: 70
Sig-Sec:
        encrypted=714n/4vpiCTtIa4yPCgSHL/cP3svwW6DWNpI2eEVUcln4s4LrO9K1q/HyaRs
        HXcZ, skey=djb2tbuv97tI2DYE7jTn98HbHAO2Qj46yOBVohS/4izWo+rUdxiE36n1Yge
        H88uEifYKKQSmLWuNJvCW5ODjOw==, ed_algo=AES,
        digest=02afa3850853f31b0865120453809761, md_algo=MD5,
        signature=PxD3Ji2aT5tvdGU1Guxgg+9Y/Qonm7nnrgKlVwqLjcVqYfQD6NcSX/4Uipau
        w4Xm2aC+vmUg5FUsIm6LA2iQUQ==
Content-Length:  1367

--boundary1
content-type: application/sdp
v=0
o=alice 53655765 2353687637 IN IP4 10.0.0.5
…
a=fmtp:101 0-11,16

--boundary1
content-type: text/plain
-----BEGIN CERTIFICATE-----
MIIC1zCCAb+gAwIBAgIBAjANBgkqhkiG9w0BAQUFADBpMRIwEAYDVQQDFAlZb3Vy
…
9wZRPQE7IsI92G0=
-----END CERTIFICATE-----
--boundary1--
```

ภาพประกอบที่ 4.32 Invite Replay Billing Attack: (5) INVITE

3) Call Establishment Hijacking

วิธีการโจมตีนี้จะต้องมีการแก้ไข Request-URI และหมายเลขไอพีในส่วนของข้อมูลที่ใช้อธิบายเซสชันของข้อความ (4) INVITE เช่นเดียวกับ Invite Replay Billing Attack แต่ไม่จำเป็นต้องแก้ไขหมายเลขลำดับของ CSeq ซึ่งการแก้ไขหมายเลขไอพีก็เพียงพอแล้วที่จะทำให้กลไก SIPE-SAP ตรวจสอบได้ว่ามีการโจมตีเกิดขึ้น ดังภาพประกอบที่ 4.33

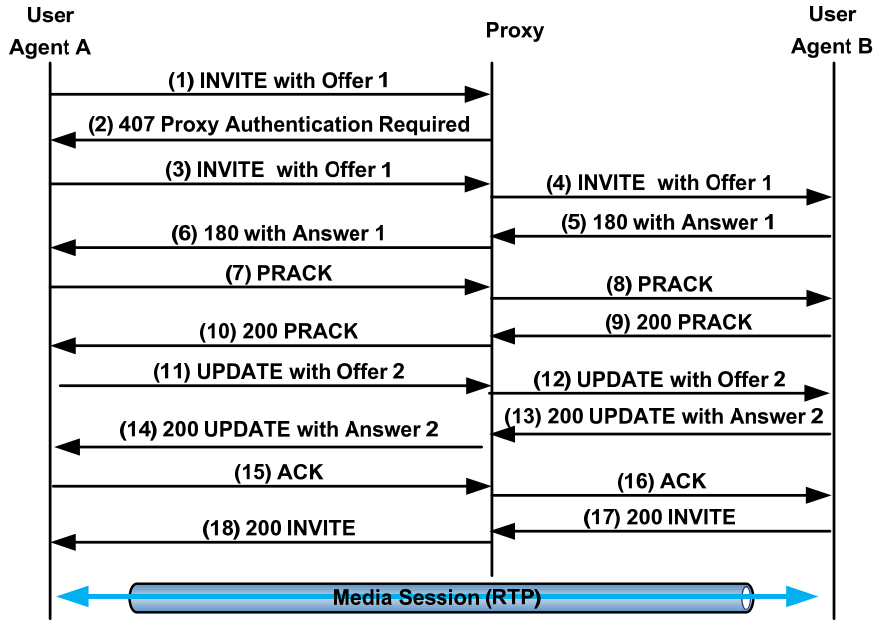


ภาพประกอบที่ 4.33 ตัวอย่าง Call Establishment Hijacking

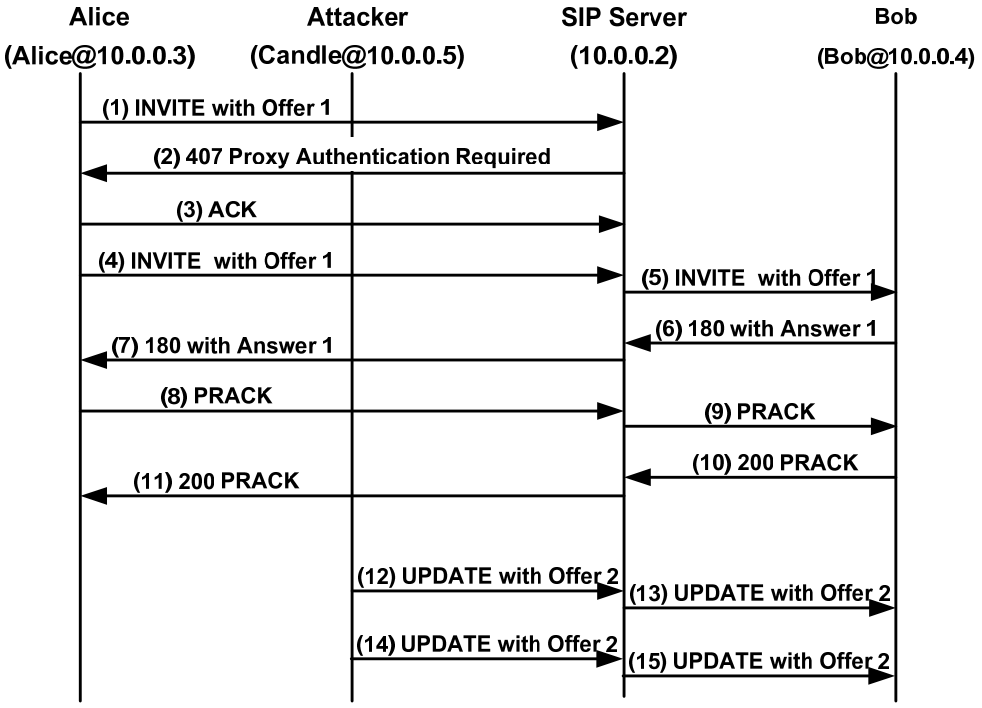
4) UPDATE Attack

การเปลี่ยนแปลงรายละเอียดของเซสชันมีกระบวนการตามภาพประกอบที่ 4.34 และภาพประกอบที่ 4.35 แสดงตัวอย่างการโจมตีแบบ UPDATE Attack การสร้างข้อความ (12) UPDATE (รายละเอียดดูภาพประกอบที่ 4.36) นอกจากต้องใช้ Request-URI, Via, Route, From, To, Call-ID และ CSeq แล้ว ยังต้องมีเฮดเดอร์ Sig-Sec ตามกลไกที่ได้ออกแบบไว้อีกด้วย เฮดเดอร์ Sig-Sec ของข้อความ UPDATE ต้องประกอบด้วย digest ในตัวอย่างนี้ได้มีการคัดลอก digest จากข้อความ (4) INVITE มาใส่ในข้อความ UPDATE ซึ่งเมื่อ UAS ตรวจสอบจะทราบได้ทันทีว่ามีการโจมตี เนื่องจากข้อความ (4) INVITE และ UPDATE จะมีหลายเฮดเดอร์ที่แตกต่างกัน เช่น CSeq และข้อมูล tag ของเฮดเดอร์ To การสร้าง digest จึงได้ผลลัพธ์ออกมาไม่เหมือนกัน

นอกจากนี้ ผู้บุกรุกไม่สามารถสร้าง digest ขึ้นมาเองได้ เพราะต้องใช้กุญแจเซสชันในการสร้าง เมื่อได้รับข้อความ UPDATE นี้ UAS จะเพิกเฉย ไม่ดำเนินการปรับปรุงข้อมูลรายละเอียดของเซสชัน



ภาพประกอบที่ 4.34 การเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชันโดยใช้คำร้องขอ UPDATE



ภาพประกอบที่ 4.35 ตัวอย่าง UPDATE Attack

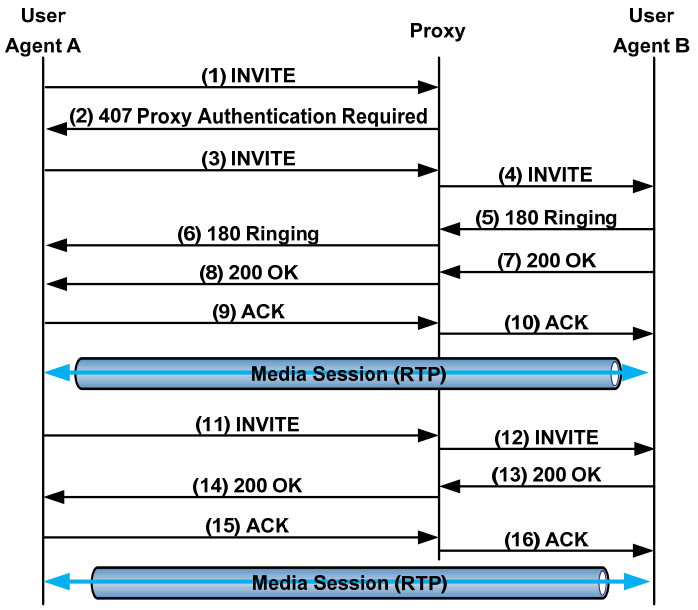
```
UPDATE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2154-1-7
Route: <sip:10.0.0.2;lr=on;did=9a7.cf305261>
From: <sip:alice@sipserver.cs.psu.ac.th>;tag=1
To: <sip:bob@sipserver.cs.psu.ac.th>;tag=1
Call-ID: 1-2154@10.0.0.3
CSeq: 4 UPDATE
Contact: sip:candle@candleclient.sipserver.cs.psu.ac.th:5060
Max-Forwards: 70
Sig-Sec: digest=30b7c251c7328e180a1b8fdc6c6f7df8, md_algo=MD5
Content-Type: application/sdp
Content-Length:   141

v=0
o=candle 53655765 2353687637 IN IP4 10.0.0.5
s=A conversation
c=IN IP4 10.0.0.5
t=0 0
m=audio 6000 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

ภาพประกอบที่ 4.36 UPDATE Attack: (12) UPDATE

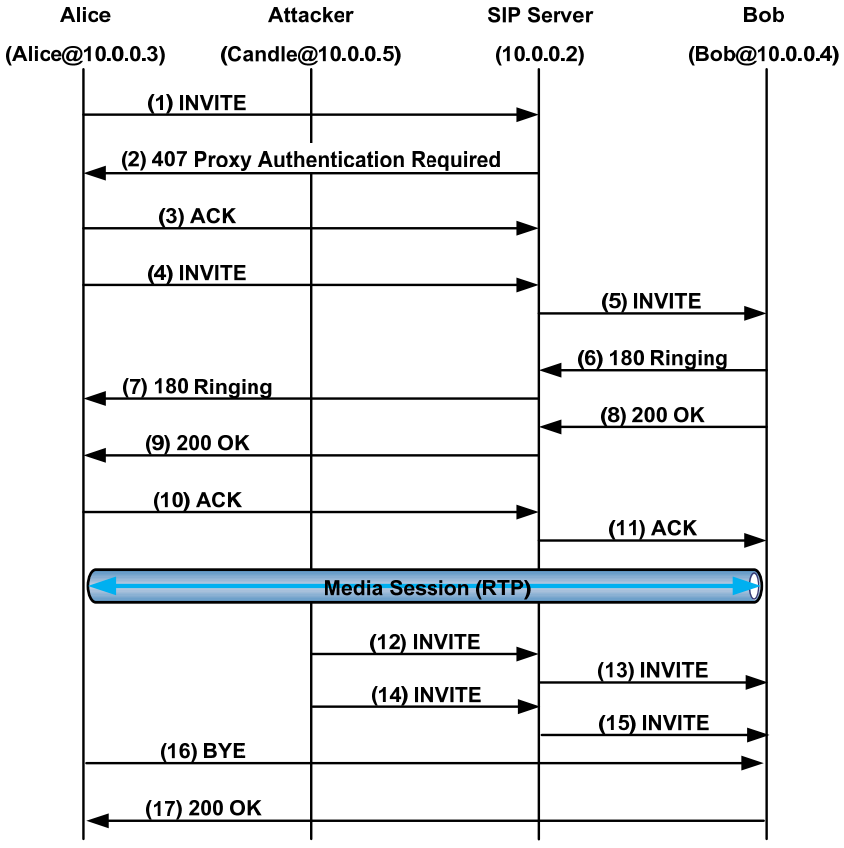
5) Re-INVITE Attack

คำร้องขอ Re-INVITE คือคำร้องขอ INVITE ที่มีการส่งอีกครั้งหลังจากการเชื่อมต่อสำเร็จแล้วเพื่อเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชัน ดังภาพประกอบที่ 4.37

การขอเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชันโดยใช้ข้อความ Re-INVITE มีลักษณะเดียวกันกับการใช้ข้อความ UPDATE ดังภาพประกอบที่ 4.38 คือข้อความ (12) INVITE (รายละเอียดดูภาพประกอบที่ 4.39) ต้องประกอบด้วยเฮดเดอร์ Sig-Sec หากผู้บุกรุกไม่มีกุญแจเซสชันก็จะไม่สามารถสร้าง digest ที่ถูกต้องขึ้นมาได้ เมื่อ UAS ได้รับข้อความ Re-INVITE จึงไม่ดำเนินการปรับปรุงข้อมูลรายละเอียดของเซสชัน



ภาพประกอบที่ 4.37 การเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชันโดยใช้คำร้องขอ INVITE



ภาพประกอบที่ 4.38 Re-INVITE Attack

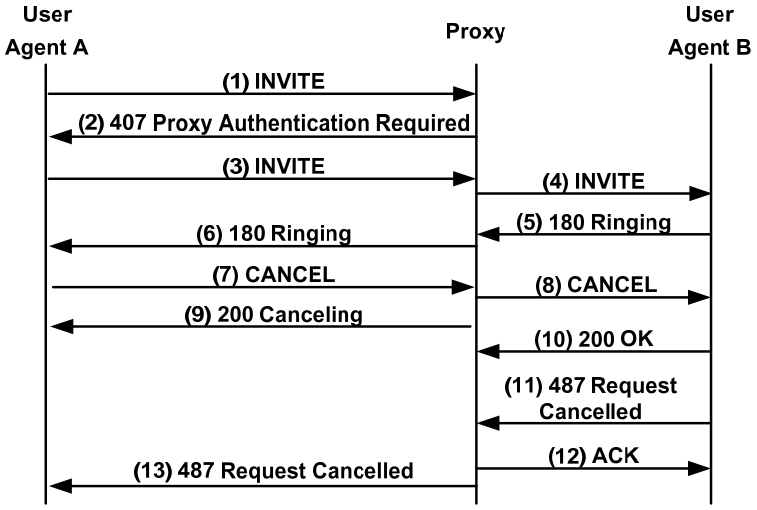
```
INVITE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-2949-1-8
Route: <sip:10.0.0.2;lr=on;did=086.961c0262>
From: <sip:alice@sipserver.cs.psu.ac.th>;tag=1
To: <sip:bob@sipserver.cs.psu.ac.th>;tag=1
Call-ID: 1-2949@10.0.0.3
CSeq: 3 INVITE
Contact: sip:candle@candleclient.sipserver.cs.psu.ac.th:5060
Max-Forwards: 70
Sig-Sec: digest=615c346f431aa89dafb4581846fdc1b4, md_algo=MD5
Content-Type: application/sdp
Content-Length:   141

v=0
o=candle 53655765 2353687637 IN IP4 10.0.0.5
s=A conversation
c=IN IP4 10.0.0.5
t=0 0
m=audio 6000 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

ภาพประกอบที่ 4.39 Re-INVITE Attack: (12) INVITE

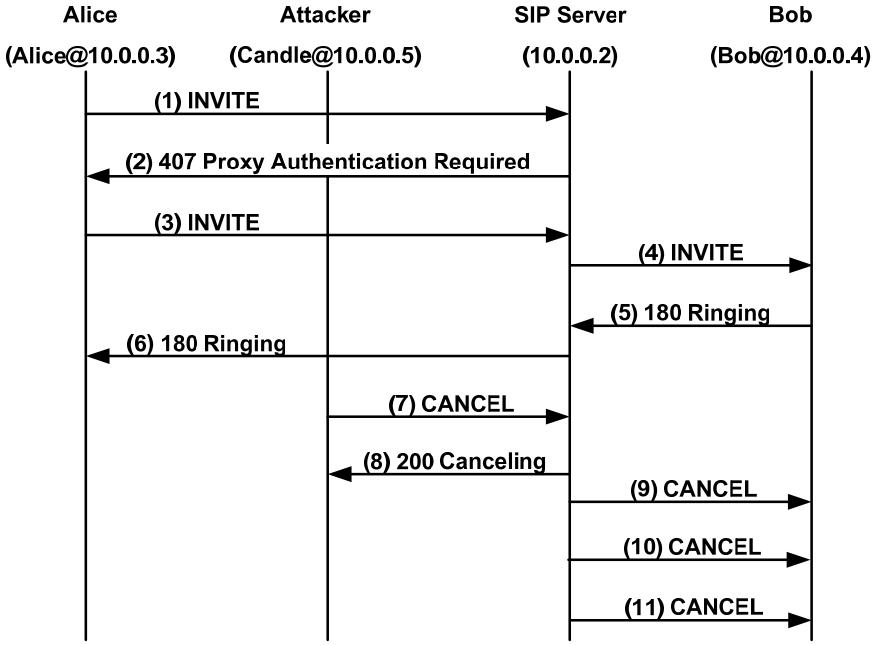
6) CANCEL Attack

คำร้องขอ CANCEL ใช้สำหรับยกเลิกคำร้องขอที่กำลังดำเนินการอยู่ เช่น การยกเลิกคำร้องขอ INVITE จะต้องดำเนินการก่อนที่จะมีการเชื่อมต่อเสร็จสิ้น (ก่อน UAS ส่งข้อความ 200 OK) ดังภาพประกอบที่ 4.40

ตัวอย่างการโจมตีโดยใช้คำร้องขอ CANCEL แสดงดังภาพประกอบที่ 4.41 ผู้บุกรุกต้องใช้ Request-URI, Via, From, To, Call-ID และ CSeq ที่สอดคล้องกับข้อความ (3) INVITE เพื่อปลอมแปลงข้อความ (7) CANCEL (รายละเอียดดูภาพประกอบที่ 4.42) แต่เฮด-เดอร์ CSeq ของทั้งสองข้อความไม่เหมือนกัน ข้อความ (7) CANCEL ไม่มีข้อมูลรายละเอียดของเซสชัน นอกจากนี้ กุญแจเซสชันที่ใช้สร้าง digest ยังไม่เหมือนกันอีกด้วย การคัดลอก digest จากข้อความ (3) INVITE มาใช้เพื่อโจมตีตามตัวอย่างจึงไม่ประสบความสำเร็จ



ภาพประกอบที่ 4.40 ตัวอย่างการขอยกเลิกการเชื่อมต่อโดยใช้ CANCEL



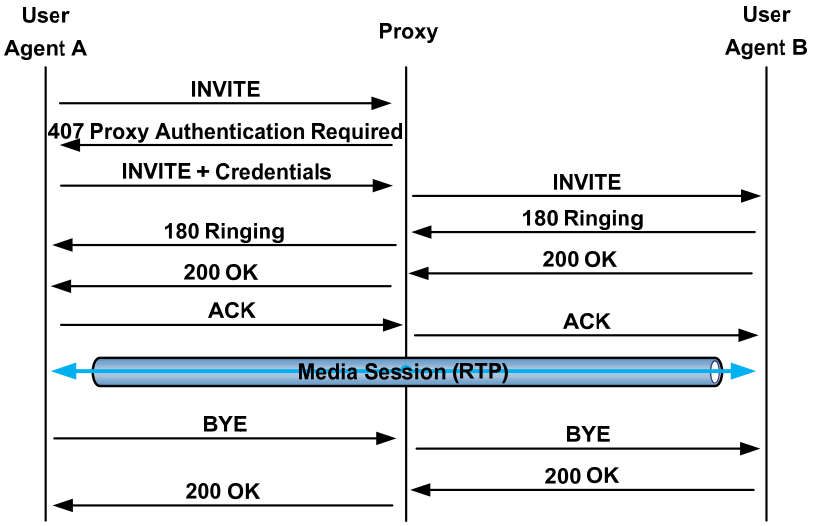
ภาพประกอบที่ 4.41 ตัวอย่าง CANCEL Attack

```
CANCEL sip:bob@sipserver.cs.psu.ac.th:5060 SIP/2.0
Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1631-1-3
From: <sip:alice@sipserver.cs.psu.ac.th>;tag=1
To: <sip:bob@sipserver.cs.psu.ac.th>
Call-ID: 1-1631@10.0.0.3
CSeq: 2 CANCEL
Sig-Sec: digest=04c44131baf621d6fe5426b2b8e5fb28, md_algo=MD5
Content-Length: 0
```

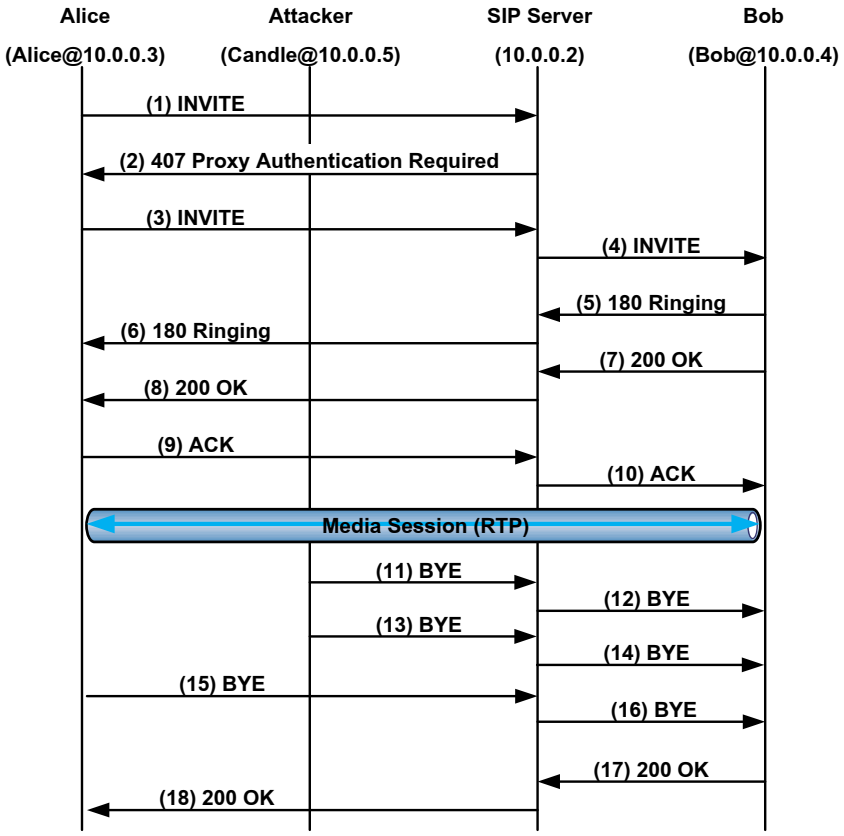
ภาพประกอบที่ 4.42 CANCEL Attack: (7) CANCEL

7) BYE Attack

คำร้องขอ BYE ใช้ยกเลิกการเชื่อมต่อที่สร้างขึ้นมาแล้ว ดังภาพประกอบที่ 4.43 ส่วนภาพประกอบที่ 4.44 คือตัวอย่างการโจมตีโดยการส่งคำร้องขอ BYE ตัวอย่างนี้มีการคัดลอกเฮดเดอร์ Sig-Sec จากข้อความ BYE อื่นๆ ที่เคยส่งระหว่างเป้าหมายที่ต้องการโจมตีมาใช้สร้างข้อความ (11) BYE (รายละเอียดดูภาพประกอบที่ 4.45) แต่ก็ไม่สามารถโจมตีได้สำเร็จ เพราะ branch, Call-ID และกุญแจเซสชันเป็นข้อมูลที่ได้จากการสุ่ม จึงมีโอกาสน้อยมากที่ข้อมูลเหล่านี้ในแต่ละข้อความ BYE จะเหมือนกัน การสร้าง digest จึงได้ผลลัพธ์ไม่เท่ากัน



ภาพประกอบที่ 4.43 การเชื่อมต่อผู้ใช้ใน SIP



ภาพประกอบที่ 4.44 ตัวอย่าง BYE Attack

```
BYE sip:bob@bobclient.sipserver.cs.psu.ac.th:5060 SIP/2.0
Via: SIP/2.0/UDP aliceclient.sipserver.cs.psu.ac.th:5060;branch=z9hG4bK-1673-1-8
Route: <sip:10.0.0.2;lr=on;did=0a4.d9770194>
From: <sip:alice@sipserver.cs.psu.ac.th>;tag=1
To: <sip:bob@sipserver.cs.psu.ac.th>;tag=1
Call-ID: 1-1673@10.0.0.3
CSeq: 3 BYE
Contact: <sip:alice@aliceclient.sipserver.cs.psu.ac.th:5060>
Max-forwards: 70
Sig-Sec: digest=4b90108adbfec0a9f16160a49ff53528, md_algo=MD5
Content-Length: 0
```

ภาพประกอบที่ 4.45 BYE Attack: (11) BYE

4.3.4 การวิเคราะห์ความปลอดภัย

ในขั้นตอนการพิสูจน์ตัวตน ข้อมูลพิสูจน์ตัวตนถูกเข้ารหัสทำให้ผู้บุกรุกไม่สามารถนำไปวิเคราะห์หารหัสผ่านได้ หากผู้บุกรุกต้องการโจมตีโดยวิธีการ Replay หรือ Hijacking จะต้องมีการแก้ไขข้อความ ซึ่งเซิร์ฟเวอร์สามารถตรวจสอบได้จากโมดูล MD และหากจะแก้ไขข้อความโดยไม่ให้เซิร์ฟเวอร์ตรวจสอบได้ ผู้บุกรุกต้องใช้กุญแจเซสชัน เนื่องจากกุญแจเซสชันถูกเข้ารหัสด้วยกุญแจสาธารณะของเซิร์ฟเวอร์ ดังนั้น มีเพียงเซิร์ฟเวอร์เท่านั้นที่สามารถถอดรหัสได้ นอกจากนี้ กุญแจเซสชันที่ใช้ไม่จำเป็นต้องเป็นกุญแจที่ใช้เพียงครั้งเดียว (One-time Key) เนื่องจากเซิร์ฟเวอร์ใช้ค่า nonce เพื่อป้องกันการโจมตีแบบ Replay อยู่แล้ว

การปลอมแปลงข้อความเพื่อใช้โจมตีจำเป็นต้องใช้กุญแจเซสชันในการสร้างเฮดเดอร์ Sig-Sec ซึ่งกระบวนการแลกเปลี่ยนกุญแจได้มีการเข้ารหัสด้วยกุญแจสาธารณะ ผู้บุกรุกจึงไม่สามารถปลอมแปลงได้

เมื่อเปรียบเทียบบริการด้านความปลอดภัยกับงานวิจัยอื่นๆ ตามตารางที่ 4.3 แม้ว่า SIPE-SAP จะเน้นการรักษาความลับเฉพาะข้อมูลที่ใช้พิสูจน์ตัวตน และให้ความสำคัญกับการรักษาความปลอดภัยของข้อความ SIP เพียงบางส่วน แต่การรักษาความปลอดภัยนี้จะเป็นแบบ End-to-End และจากการทดสอบการโจมตีพบว่ามีความปลอดภัยค่อนข้างเพียงพอต่อการป้องกันการโจมตีสัญญาณเชื่อมต่อ ดังตารางที่ 4.4 คือไม่สามารถป้องกันการโจมตีแบบ Call Termination Hijacking ได้เท่านั้น เพราะการโจมตีวิธีนี้จะอาศัยการสร้างแพ็กเก็ต RTP ปลอมขึ้นมา แต่ SIPE-SAP เน้นการรักษาความปลอดภัยให้กับ SIP หากต้องการป้องกันการโจมตีอาจใช้กุญแจเซสชันที่ได้จากการแลกเปลี่ยนกุญแจของ SIPE-SAP มาใช้เข้ารหัส RTP เพื่อไม่ให้ผู้บุกรุกปลอมแปลงแพ็กเก็ต RTP ได้

งานวิจัยที่เกี่ยวข้อง

[1] = SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments (Wu และคณะ, 2004)

[2] = Providing Response Identity and Authentication in IP Telephony (Cao และ Jennings, 2006)

[3] = Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP) (Peterson และ Jennings, 2006)

[4] = Holistic VoIP Intrusion Detection and Prevention System (Nassar และคณะ, 2007)

[5] = A Lightweight Protection Mechanism against Signaling Attacks in a SIP-Based VoIP Environment (Geneiatakis และ Lambrinoudakis, 2008)

[6] = A Novel Approach to Avoid Billing Attack on VOIP System. World Academy of Science, Engineering and Technology (Shekokar และ Devane, 2010)

ตารางที่ 4.3 บริการด้านความปลอดภัยที่สนับสนุนโดยกลไกป้องกันการโจมตีสัญญาณเชื่อมต่อ

Security Services	**งานวิจัย/RFCs**						
	[1]	**[2]**	**[3]**	**[4]**	**[5]**	**[6]**	**SIPE-SAP**
Integrity	-	Partial	-	-	Hop-by-Hop	Hop-by-Hop	Partial
Availability	Partial	-	Partial	Partial	Hop-by-Hop (Partial)	Hop-by-Hop (Partial)	Partial
Authenticity	-	Partial	Partial	-	Hop-by-Hop	Hop-by-Hop	Partial
Confidentiality	-	-	-	-	-	Hop-by-Hop	Partial
Non-Repudiation	-	Partial	Partial	-	-	-	Partial

ตารางที่ 4.4 การป้องกันการโจมตีที่สนับสนุนโดยกลไกป้องกันการโจมตีสัญญาณเชื่อมต่อ

การโจมตี (Attack Mechanisms)	**งานวิจัย/RFCs**						
	[1]	**[2]**	**[3]**	**[4]**	**[5]**	**[6]**	**SIPE-SAP**
1. Registration Hijacking	-	-	-	-	✓	✓	✓
2. Invite Replay Billing Attack	-	-	✓	-	✓	✓	✓
3. Call Establishment Hijacking	-	-	✓	-	✓	✓	✓
4. Call Termination Hijacking	-	-	-	-	-	-	-
5. UPDATE Attack	-	✓	✓	-	✓	✓	✓
6. Re-INVITE Attack	✓	✓	✓	-	✓	✓	✓
7. CANCEL Attack	-	✓	-	✓	✓	✓	✓
8. BYE Attack	✓	✓	-	✓	✓	✓	✓

4.4 สรุป

ในบทนี้ได้กล่าวถึงการพัฒนาและทดสอบระบบ โปรแกรมสำคัญที่ใช้สำหรับพัฒนาระบบและทดสอบระบบคือ OpenSIPS, Linphone, SIPp และ Ettercap จากการทดสอบประสิทธิภาพของระบบพบว่า SIPE-SAP ระยะเวลาที่ใช้ในการลงทะเบียนและการเชื่อมต่อการโทรเฉลี่ยของ SIPE-SAP เพิ่มขึ้นคิดเป็น 3.3 และ 5.53 เท่าตามลำดับ เมื่อเปรียบเทียบกับการใช้ SIP ร่วมกับ HTTP Digest และใช้เวลาลดลงคิดเป็น 9.28% และ 12.96% ตามลำดับ เมื่อเปรียบเทียบกับการใช้ SIP ร่วมกับ TLS โดยที่ SIPE-SAP ยังสามารถป้องกันการโจมตีสัญญาณเชื่อมต่อได้เช่นเดียวกันกับ TLS ดังแสดงในการทดสอบการโจมตี สำหรับการสรุปผลการวิจัยและปัญหาในการดำเนินการ จะกล่าวถึงในบทที่ 5 ต่อไป

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 บทนำ

บทนี้เป็นการกล่าวสรุปงานวิจัยและผลที่ได้จากการวิจัยครั้งนี้ รวมถึงอุปสรรคและปัญหาที่เกิดขึ้นในระหว่างการทำวิจัย สุดท้ายเป็นข้อเสนอแนะสำหรับที่ผู้สนใจจะนำงานวิจัยนี้ไปพัฒนาต่อไป

5.2 สรุปผลการวิจัย

งานวิจัยนี้เป็นการเสนอกลไกที่ช่วยสร้างความมั่นคงปลอดภัยสำหรับสัญญาณเชื่อมต่อของ SIP ซึ่งเป็นโพรโทคอลสำคัญที่ใช้งานในเทคโนโลยี VoIP กลไกที่นำเสนอคือ SIP Extension for Signaling Attacks Protection หรือ SIPE-SAP

การออกแบบกลไกเริ่มต้นจากการวิเคราะห์การโจมตีสัญญาณเชื่อมต่อ ได้แก่ Registration Hijacking, Invite Replay Billing Attack, Call Establishment Hijacking, Call Termination Hijacking, UPDATE Attack, Re-INVITE Attack, CANCEL Attack และ BYE Attack ว่ามีรูปแบบการโจมตีอย่างไรและจำเป็นต้องใช้ส่วนใดของข้อความ SIP เพื่อให้การโจมตีสำเร็จ ส่วนการโจมตีแบบ Broken Handshaking ที่กำหนดไว้ในขอบเขตการวิจัยนั้น ปัจจุบันพบว่าเซิร์ฟเวอร์ต่างๆ ซึ่งเป็นซอฟต์แวร์รหัสเปิดที่ได้รับความนิยมนำมาใช้เป็นพร็อกซีเซิร์ฟเวอร์ทั้ง OpenSIPS และ Asterisk ได้มีการแก้ไขปัญหานี้แล้ว งานวิจัยนี้จึงไม่จำเป็นต้องแก้ปัญหาการโจมตีแบบ Broken Handshaking ผลที่ได้สามารถแบ่งออกเป็น 3 กลุ่ม คือ

1) เฮดเดอร์ที่จำเป็นสำหรับการปลอมแปลงข้อความ SIP ขึ้นมาใหม่ ได้แก่ พารามิเตอร์ branch ของ Via, From, To, Call-ID และ Cseq

2) เฮดเดอร์ที่ประกอบด้วยข้อมูลประจำตัว ได้แก่ Authorization และ Proxy-Authorization

3) เฮดเดอร์ที่อาจถูกแก้ไข ได้แก่ CSeq และ Contact ส่วนของ Request-URI และ SDP Body ของ SIP

เมื่อได้ส่วนต่างๆ ของ SIP ที่เกี่ยวข้องกับการโจมตีแล้ว จึงมีการออกแบบเฮด-เดอร์ Sig-Sec เพื่อรองรับกลไกป้องกันการโจมตี โดยมีกระบวนการการแลกเปลี่ยนกุญแจ การนำข้อมูลประจำตัว (ข้อมูล “response”) ของเฮดเดอร์ Authorization และ Proxy-Authorization มาเข้าและถอดรหัส และนำส่วนต่างๆ ของ SIP ที่ได้วิเคราะห์แล้วมาผ่านกระบวนการย่อยข้อความพร้อมกับกุญแจเซสชัน เพื่อใช้ในการพิสูจน์ตัวตนและตรวจสอบความถูกต้องสมบูรณ์ของข้อมูล

การทดสอบระบบได้มีการนำมาทดลองใช้งานกับโปรแกรมจำลองโทรศัพท์ Linphone และโปรแกรมสร้างการจราจรในเครือข่าย SIPp พร้อมทั้งทดสอบประสิทธิภาพของระบบ การทดสอบประสิทธิภาพแบ่งเป็น 2 ส่วนคือ การลงทะเบียนและการเชื่อมต่อการโทร โดยเปรียบเทียบจากระยะเวลาการตอบสนอง (Response Times) ผลการทดสอบปรากฏว่าระยะเวลาที่ใช้ในการลงทะเบียนและการเชื่อมต่อการโทรเฉลี่ยของ SIPE-SAP เพิ่มขึ้นคิดเป็น 3.3 และ 5.53 เท่าตามลำดับ เมื่อเปรียบเทียบกับการใช้ SIP ร่วมกับ HTTP Digest และใช้เวลาลดลงคิดเป็น 9.28% และ 12.96% ตามลำดับ เมื่อเปรียบเทียบกับการใช้ SIP ร่วมกับ TLS

ในด้านการทดสอบการโจมตี SIP ที่ใช้กลไก SIPE-SAP ผลปรากฏว่ากลไก SIPE-SAP สามารถช่วยป้องกัน Registration Hijacking, Invite Replay Billing Attack, Call Establishment Hijacking, UPDATE Attack, Re-INVITE Attack, CANCEL Attack และ BYE Attack ได้ เนื่องจากการโจมตีเหล่านี้จำเป็นต้องมีการแก้ไขข้อความ SIP โดยการโจมตีแบบ Registration Hijacking ต้องมีการแก้ไขข้อมูล Cseq และ Contact การโจมตีแบบ Invite Replay Billing Attack ต้องมีการแก้ไขข้อมูล Request-URI, Cseq และ SDP Body การโจมตีแบบ Call Establishment Hijacking ต้องมีการแก้ไขข้อมูล Request-URI และ SDP Body การโจมตีแบบ UPDATE Attack, Re-INVITE Attack, CANCEL Attack และ BYE Attack ต้องใช้ข้อมูล พารามิเตอร์ branch ของ Via, From, To, Call-ID, CSeq และ SDP Body ที่ดักจับได้เพื่อสร้างข้อความใหม่ และต้องแก้ไขข้อมูล Cseq และ SDP Body ซึ่ง SIPE-SAP ได้นำข้อมูล Request-URI, พารามิเตอร์ branch ของ Via, From, To, Call-ID, Cseq, Contact และ SDP Body มาย่อยข้อความเพื่อใช้ตรวจสอบการแก้ไขข้อมูล ถ้ามีการแก้ไขข้อมูลเหล่านี้แสดงว่ามีการโจมตีเกิดขึ้น ดังนั้น UAC, UAS และ SIP Server จะไม่ดำเนินการตามคำร้องขอที่ส่งมา ส่งผลให้สามารถป้องกันการโจมตีรูปแบบต่างๆ ที่กล่าวมาได้

การโจมตีแบบ Call Termination Hijacking ผู้บุกรุกต้องดักจับข้อความ SIP และแพ็กเก็ต RTP แล้วใช้ข้อมูลจาก RTP ในการปลอมแปลงแพ็กเก็ตเพื่อใช้โจมตี โดยที่ SIPE-SAP สามารถป้องกันการโจมตีที่ดำเนินการกับข้อความ SIP เท่านั้น SIPE-SAP จึงไม่สามารถป้องกันการโจมตีด้วยวิธีการนี้ได้ แต่ถ้านำ SIPE-SAP มาใช้กับข้อความ 200 OK จะช่วยให้ตรวจสอบได้ว่ามีการโจมตีเกิดขึ้น ผู้ใช้งานอาจแจ้งไปยังผู้ให้บริการว่าเกิดการโจมตีและให้ยกเลิกการเชื่อมต่อนั้นๆ นอกจากนี้ SIPE-SAP มีกระบวนการแลกเปลี่ยนกุญแจที่สามารถ

นำมาใช้เข้ารหัสแพ็กเก็ต RTP เพื่อป้องกันการโจมตีแบบ Call Termination Hijacking ได้ ดังนั้น SIPE-SAP สามารถป้องกันการโจมตีสัญญาณเชื่อมต่อได้เช่นเดียวกันกับ TLS โดยที่ใช้เวลาในการทำงานน้อยกว่า

นอกจากนี้ ค่า response ซึ่งได้จากการนำรหัสผ่านมาใช้ในการคำนวณยังถูกเข้ารหัสลับโดยใช้กลไก SIPE-SAP ผู้บุกรุกจึงไม่สามารถคำนวณรหัสผ่านจากค่า response ได้ และการใช้ใบรับรองเข้ามาช่วยในการกระจายกุญแจ ช่วยให้ SIPE-SAP สามารถรองรับการทำงานเมื่อผู้ใช้อยู่คนละโดเมนกันได้

ดังนั้น กลไกสร้างความมั่นคงสำหรับสัญญาณเชื่อมต่อของ SIP ที่ได้นำเสนอนอกจากสามารถป้องกันการโจมตีโดยการเปลี่ยนแปลงข้อมูลรายละเอียดของเซสชันโดยผู้ที่ไม่มีสิทธิ์ และการโจมตีการคิดค่าบริการที่อาศัยจุดอ่อนของการส่งสัญญาณเชื่อมต่อด้วย SIP ตามวัตถุประสงค์ของงานวิจัยแล้ว กลไกที่นำเสนอยังสามารถป้องกันการปลอมแปลงข้อความ SIP เพื่อโจมตีให้เกิดการปฏิเสธการให้บริการไปยังผู้ใช้งานรายอื่นได้

5.3 อุปสรรคและปัญหา

ปัญหาที่เกิดขึ้นในการดำเนินงานวิจัยมีดังนี้

1) การพัฒนาระบบค่อนข้างมีความยุ่งยาก เนื่องจากต้องมีการประสานการทำงานของระบบทั้ง 3 ส่วนคือ พร็อกซีเซิร์ฟเวอร์ UAC และ UAS ให้สอดคล้องกัน นอกจากนี้มีการใช้พร็อกซีเซิร์ฟเวอร์เป็นเซิร์ฟเวอร์สำหรับแจกใบรับรองด้วย ดังนั้น เมื่อมีการคอมไพล์โปรแกรมฝ่ายพร็อกซีเซิร์ฟเวอร์แต่ละครั้ง ต้องมีการแจกใบรับรองใหม่ให้กับพร็อกซีเซิร์ฟเวอร์ UAC และ UAS ด้วย

2) แต่ละโปรแกรมที่ใช้พัฒนาและทดสอบระบบมีโครงสร้างข้อมูลและการทำงานของโปรแกรมที่แตกต่างกันออกไป และเป็นโปรแกรมที่มีขนาดค่อนข้างใหญ่ มีการทำงานซับซ้อน การศึกษาโปรแกรมเพื่อแก้ไขหรือเพิ่มเติมการทำงานจึงใช้เวลานาน

5.4 ข้อเสนอแนะ

ข้อเสนอแนะสำหรับการดำเนินการเกี่ยวกับการนำระบบไปใช้งานมีดังนี้

1) ในการทดสอบระบบ ผู้วิจัยได้ส่งใบรับรองแนบไปกับข้อความ REGISTER หรือ INVITE ทุกครั้ง ซึ่งทำให้ข้อความมีขนาดใหญ่ และโดยปกติใบรับรองจะมีการกำหนดอายุการใช้งานอยู่แล้ว จึงไม่จำเป็นต้องมีการส่งใบรับรองไปทุกครั้งที่มีการเชื่อมต่อ ดังนั้น การ

นำไปใช้งานจริงอาจใช้วิธีการส่งเป็นตำแหน่งที่อยู่ที่สามารถดาวน์โหลดใบรับรองนี้ได้ เพื่อลดขนาดของข้อความลง

2) งานวิจัยนี้เน้นการรักษาความปลอดภัยให้กับ SIP แต่การโจมตีแบบ Call Termination Hijacking อาศัยการสร้างแพ็กเก็ต RTP ปลอมขึ้นมา เพื่อให้เซิร์ฟเวอร์คิดว่าผู้ใช้ยังมีการติดต่อสื่อสารอยู่ งานวิจัยนี้จึงไม่สามารถป้องกันการโจมตีแบบนี้ได้โดยตรง แต่สามารถนำมาปรับใช้เพื่อป้องกันการโจมตีโดยการใช้กุญแจเซสชันที่ได้จากการแลกเปลี่ยนกุญแจตามกลไก SIPE-SAP มาเข้ารหัส RTP เพื่อไม่ให้ผู้บุกรุกปลอมแปลงแพ็กเก็ต RTP ได้

3) การนำระบบไปใช้งานจริง ต้องมีการพัฒนาระบบในส่วนของเซิร์ฟเวอร์ คือ Registrar และ Proxy และส่วนของไคลเอนต์ทั้ง UAC และ UAS โดยเมื่อได้รับข้อความ SIP เข้ามาหรือต้องการส่งข้อความ SIP ออกไป ต้องเรียกใช้ฟังก์ชันการทำงานของ SIPE-SAP ก่อน การเรียกใช้งาน SIPE-SAP ทางฝ่ายเซิร์ฟเวอร์ (OpenSIPS) ต้องมีการแก้ไขไฟล์ api.c และ challenge.c ในโมดูล auth ส่วนฝ่ายไคลเอนต์ที่ใช้งานไลบรารี GNU oSIP ต้องมีการแก้ไขไฟล์ osip.h เพื่อเพิ่มเฮดเดอร์ Sig-Sec เข้าไปในโครงสร้างข้อมูลเฮดเดอร์ของข้อความ SIP และแก้ไขไฟล์ osip_event.c, nist.c และ osip_message_parse.c เพื่อเรียกใช้ฟังก์ชันการทำงานของ SIPE-SAP

4) งานวิจัยในอนาคตสามารถสร้างแอปพลิเคชันสำเร็จรูปเพื่อเรียกใช้งานไลบรารี GNU oSIP ที่มีการเพิ่มเติมการทำงานของ SIPE-SAP ได้ หากต้องการปรับปรุงการทำงานของ SIPE-SAP สามารถทำได้โดยปรับปรุงแก้ไขไฟล์ osip_sipesap.c

บรรณานุกรม

Antunes, J. 2009. Academic Instant Messaging System. TERENA's 2009 Network Conference (TNC 2009). Spain, June 8-11, 2009.

Brunner, S. and Ali, A.A. 2004. Voice Over IP 101 Understanding VoIP Networks. Juniper Networks, Inc.: USA.

Butcher, D., Li, X. and Guo, J. 2007. Security Challenge and Defense in VoIP Infrastructures. IEEE Transactions on Systems, Man, and Cybernetics-Part c: Applications and Reviews. 37(6): 1152-1162.

Cao, F. and Jennings, C. 2006. Providing Response Identity and Authentication in IP Telephony. Proceedings of First International Conference on Availability, Reliability and Security (ARES'06). Austria, April 20-22, 2006. pp. 198-205.

Cisco Systems. 2002. Understanding Packet Voice Protocols. Cisco Systems, Inc.

Davidson, J., Peters, J., Bhatia, M., Kalidindi, S. and Mukherjee, S. 2006. Voice over IP Fundamentals, Second Edition. Cisco Press.: USA.

Donovan, S. The SIP INFO Method. IETF RFC 2976, October 2000.

Edelson, E. 2005. Voice over IP: Security pitfalls. Network Security. 2005(2): 4–7.

Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A. and Stewart, L. HTTP authentication: Basic and Digest Access Authentication. RFC 2617, June 1999.

Gayraud, R., Jacques, O., Day, R. and Wright, C.P. 2013. SIPp Reference Documentation. http://sipp.sourceforge.net/doc3.2/reference.html. (accessed 01/05/2013).

Geneiatakis, D., Dagiuklas, T., Kambourakis, G., Lambrinoudakis, C. and Gritzalis, S. 2006. Survey of Security Vulnerabilities in Session Initiation Protocol. IEEE Communications Surveys & Tutorials. 8(3): 68-81.

Geneiatakis, D., Kambourakis, G. and Lambrinoudakis, C. 2008. A Mechanism for Ensuring the Validity and Accuracy of the Billing Services in IP Telephony. Proceedings of the 5th international conference on Trust, Privacy and Security in Digital Business. Italy, September 4-5, 2008. pp. 59-68.

Geneiatakis, D., Kambourakis, G. and Lambrinoudakis, C. 2008. SIP Security: Threats, Vulnerabilities and Countermeasures. SIP Handbook: Services, Technologies, and Security of Session Initiation Protocol. CRC Press: UAS.

Geneiatakis, D. and Lambrinoudakis, C. 2008. A Lightweight Protection Mechanism against Signaling Attacks in a SIP-Based VoIP Environment. Telecommunication Systems. 36(4): 153-159.

Goncalves, F.E. 2008. Building Telephony Systems with OpenSER, Packt Publishing Ltd.: UK.

Goncalves, F.E. 2010. Building Telephony Systems with OpenSIPS 1.6, Packt Publishing Ltd.: UK.

Handley, M., Schulzrinne, H., Columbia, U. and Rosenberg, J. SIP: Session Initiation Protocol. IETF RFC 2543, March 1999.

Hazlett, Paul., Miles, Simon. and V.Teigre, Greger. SER - Getting Started.

Housley, R., Ford, W., Polk, W. and Solo, D. Internet X.509 Public Key Infrastructure Certificate and CRL Profile. IETF RFC 2459, January 1999.

International Telecommunication Union. Packet based multimedia communication system. Telecommunication Standardization Sector of ITU. Recommendation H.323. February 1998.

Liao, Y.P. and Wang S.S. 2010. A New Secure Password Authenticated Key Agreement Scheme for SIP using Self-Certified Public Keys on Elliptic Curves. Computer Communications. 33: 372-380.

Nassar, M., Niccolini S., State, R. and Ewald, T. 2007. Holistic VoIP Intrusion Detection and Prevention System. IPTCOMM '07. New York, July 19-20, 2007.

Olejniczak, S.P. 2009. VoIP Deployment For Dummies. Wiley Publishing, Inc.: USA.

Ornaghi, A., Valleri, M., Escobar, E. and Milam, E. 2012. Ettercap Home Page. http://ettercap.github.io/ettercap/index.html. (accessed 01/05/2013).

Peterson, J. and Jennings, C. Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP). IETF RFC 2006; 4474.

Palmieri, F. and Fiore, U. 2009. Providing True End-to-End Security in Converged Voice over IP Infrastructures. Computer and Security. 28(6): 433-449.

Porter, T., Kanclirz, J., Zmolek, A., Rosela, A., Cross, M., Chaffin, L., Baskin, B. and Shim, C. 2006. Practical VoIP Security. Syngress Publishing, Inc.: Canada.

Ramsdell, B. Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Message Specification. IETF RFC 3851, July 2004.

Rescorla , E. 2001. SSL and TLS: Designing and Building Secure Systems, Addison Wesley: Boston.

Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and Schooler, E. Sip: Session Initiation Protocol. IETF RFC 3261, June 2002.

Rosenberg, J. and Schulzrinne, H. Reliability of Provisional Responses in the Session Initiation Protocol (SIP). IETF RFC 3262, June 2002.

Rosenberg, J. The Session Initiation Protocol (SIP) UPDATE Method. IETF RFC 3311, September 2002.

Russell, T. 2008. Session Initiation Protocol (SIP): Controlling Convergent Networks. The McGraw-Hill Companies, Inc.: USA.

Shekokar, N.M. and Devane, S.R. 2010. A Novel Approach to Avoid Billing Attack on VOIP System. World Academy of Science, Engineering and Technology. 62: 993-997.

Sparks, R. The Session Initiation Protocol (SIP) Refer Method. IETF RFC 3515, April 2003.

Thermos, P. and Takanen, A. 2007. Securing VoIP Networks. Pearson Education, Inc.: USA.

Tiller, J.S. 2000. A Technical Guide to IPSec Virtual Private Networks. Auerbach Publications: New York.

Tindal, S. 2009. VoIP Hackers Strike Perth Business. http://www.zdnet.com/voip-hackers-strike-perth-business-1339294515. (accessed 01/05/2013).

Wu, Y.S., Bagchi, S., Garg, S. and Singh, N. 2004. SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments. Proceedings of the 2004 International Conference on Dependable Systems and Networks (DSN'04). Italy, 2004. pp. 433-442.

Wu, L., Zhang, Y., and Wang, F. 2009. A New Provably Secure Authentication and Key Agreement Protocol for SIP using ECC. Computer Standards & Interfaces. 31: 286-291.

Zhang, R., Wang, X., Yang, X. and Jiang, X. 2007. Billing Attacks on SIP-Based VoIP Systems. Proceedings of the First USENIX Workshop on Offensive Technologies (WOOT2007). Boston, August 6-10, 2007. Article No. 4.

Zhang, R., Wang, X., Yang, X. and Jiang, X. 2010. On the Billing Vulnerabilities of SIP-Based VoIP Systems. Computer Networks. 54 (11): 1837-1847.