ลายเซ็นดิจิตอลสำหรับอุปกรณ์โมบายล์โดยใช้ปัจจัยทางตำแหน่งและเวลา

SPACE-TIME BASED DIGITAL SIGNATURE ON MOBILE INTERNET DEVICES

สันติ จารุสมบัติ

SANTI JARUSOMBAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2550

ลายเซ็นดิจิตอลสำหรับอุปกรณ์โมบายล์โดยใช้ปัจจัยทางตำแหน่งและเวลา

SPACE-TIME BASED DIGITAL SIGNATURE ON MOBILE INTERNET DEVICES

สันติ จารุสมบัติ

SANTI JARUSOMBAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2550

SPACE-TIME BASED DIGITAL SIGNATURE ON MOBILE INTERNET DEVICES

SANTI JARUSOMBAT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN COMPUTER ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2007

หัวข้อวิทยานิพนธ์	ลายเซ็นดิจิตอลสำหรับอุปกรณ์โมบายล์โดยใช้ปัจจัยทางตำแหน่งและเวลา
นักศึกษา	นายสันติ จารุสมบัติ
รหัสนักศึกษา	47060830
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
พ.ศ.	2550
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผศ.ดร.สุรินทร์ กิตติธรกุล

บทคัดย่อ

เนื่องด้วยความเจริญก้าวหน้าอย่างรวดเร็วของเทคโนโลยีการสื่อสารไร้สาย และอุปกรณ์โมบายล์ ทำให้อุปกรณ์เหล่านี้ได้กลายมาเป็นสิ่งจำเป็นในชีวิตประจำวัน ซึ่งในทุกวันนี้ได้มีการเอาเทคโนโลยี Positioning System (PS) รวมเข้าไว้กับอุปกรณ์เหล่านี้แล้ว และก็มีการใช้งานในหลายๆ แอพพลิเคชั่นที่ใช้งานอินเทอร์เน็ทผ่านอุปกรณ์โมบายล์ เช่น การโอนเงินผ่านอุปกรณ์มือถือ, การทำพาณิชย์อิเล็กทรอนิกส์, เกมส์ออนไลน์ และ จองตั๋วภาพยนต์ผ่านอุปกรณ์มือถือ เป็นต้น เพื่อความปลอดภัยของแอพพลิเคชันเหล่านั้น ควรที่จะรองรับการทำงานของลายเซ็นดิจิตอลเข้าไปด้วย โดยอุปกรณ์โมบายล์เป็นอุปกรณ์ที่มีหน่วยประมวลผลต่ำ และมีแบตเตอร์รี่ที่จำกัด ทำให้การใช้ลายเซ็นดิจิตอลแบบดั้งเดิมที่ใช้อัลกอริทึมการเข้ารหัสแบบอสมมาตร อาจจะเป็นภาระที่หนักเกินไปสำหรับอุปกรณ์โมบายล์ที่จะประมวลผลเพียงลำพัง เพราะจะทำให้ต้องใช้เวลาในการประมวลผลนาน และมีผลโดยตรงต่อเวลาในการใช้งานของแบตเตอร์รี่ อีกทั้งยังไม่ได้ใช้ประโยชน์จากเทคโนโลยี PS ที่มีอยู่อีกด้วย งานวิจัยนี้นำเสนอ "Space-Time Based Digital Signature on Mobile Devices" หรือ "ลายเซ็นดิจิตอลสำหรับอุปกรณ์โมบายล์โดยใช้ปัจจัยทางตำแหน่งและเวลา" โดยมีการนำหลักการของ Geo-Encryption และ Mobility Model มาประยุกต์ใช้ในการสร้างลายเซ็นดิจิตอลด้วย โดยทั้งนี้ยังเป็นการลดภาระในส่วนของผู้ใช้ โดยเปรียบเทียบกับการสร้าง RSA-based partially blind signature ของ Abe-Fujisaki สามารถลดเวลาในการประมวลผลได้ 97.5 % และลดการติดต่อสื่อสารกันระหว่างผู้ส่งและผู้รับด้วย เมื่อนำปัจจัยทางตำแหน่งที่ตั้งของอุปกรณ์โมบายล์เข้ามามีส่วนร่วมด้วย เป็นการเพิ่มความยากสำหรับผู้ประสงค์ร้ายที่ต้องการโจมตีระบบของเรา จึงเป็นการเพิ่มความปลอดภัยของการสร้าง และตรวจสอบลายเซ็นดิจิตอลขึ้นไปอีกด้วย

Thesis Title	Space-Time Based Digital Signature on Mobile Internet Devices
Student	Mr. Santi Jarusombat
Student ID.	47060830
Degree	Master of Engineering
Program	Computer Engineering
Year	2007
Thesis Advisor	Asst.Prof.Dr. Surin Kittitornkun

ABSTRACT

With rapid growing of wireless and mobile technologies, mobile device have become a part of our lives. Nowadays Positioning Service (PS) technology is combined with mobile devices. People do some Internet applications by using mobile devices, e.g. mobile fund transfer, e-commerce, game online and theatre booking. These applications should be supported Digital Signature Service for more security. Because mobile devices are low-computation and limited battery life, they are not suitable to use traditional digital signature protocols, which are based on asymmetric cryptographic algorithm, with mobile devices. And they do not use advantage of using PS technology. We present "Space-Time Based Digital Signature on Mobile Devices" by using space and time parameter to help Sign Sever generate Digital Signature. This method reduces the burden on mobile devices, the computation time of the requester by almost 97.5%, is compared with Abe-Fujisaki's RSA-based partially blind signature and decreases communication between sender and receiver. And we use mobile device's position for increasing security level of generating digital signature process.

กิตติกรรมประกาศ

คุณความดีอันใดที่บังเกิดจากวิทยานิพนธ์ฉบับนี้ ขอมอบแด่บิดาและมารดาของผู้วิจัย ผู้ที่คอยห่วงใย เข้าใจและให้การสนับสนุนในการศึกษามาโดยตลอด

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงลงได้ด้วยดี โดยได้รับความกรุณาอย่างยิ่งจากอาจารย์ที่ปรึกษาวิทยานิพนธ์ ผศ.ดร สุรินทร์ กิตติธรกุล อาจารย์ที่ปรึกษา ที่ได้ช่วยเหลือในการให้คำแนะนำความรู้ทางทฤษฎีต่างๆที่ใช้ และชี้แนะแนวทางในการแก้ปัญหาต่างๆอย่างทุ่มเทรวมทั้งฝึกฝนผู้วิจัยให้มีความสามารถในการทำวิจัยและพัฒนาได้อย่างมีประสิทธิภาพ ผู้วิจัยรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านและขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบคุณกรรมการสอบวิทยานิพนธ์ทุกท่านที่ได้กรุณาให้คำแนะนำในทุกๆเรื่อง ทั้งวิธีแก้ปัญหาที่เกิดขึ้นในการทำวิทยานิพนธ์และมุมมองในเชิงวิศวกรรมอื่นๆซึ่งช่วยให้ผู้วิจัยมีวิสัยทัศน์ที่กว้างไกลขึ้น

ขอขอบคุณบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้การสนับสนุนการทำวิทยานิพนธ์นี้

ขอบคุณพี่ๆเพื่อนๆและน้องๆนักศึกษาทุกคนในห้องวิจัย รวมทั้งเพื่อนๆหลายๆคนในอินเตอร์เน็ตที่ช่วยเหลือให้คำแนะนำต่างๆและให้กำลังใจแก่ผู้วิจัยตลอดมา

สันติ จารุสมบัติ

สารบัญ

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญรูป

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องด้วยเทคโนโลยีต่างๆได้พัฒนาขึ้นจนกระทั่งปัจจุบันโทรศัพท์เคลื่อนที่ได้เข้ามามีบทบาทในชีวิตประจำวันมากขึ้น ดังจะเห็นได้จากวิวัฒนาการของโทรศัพท์เคลื่อนที่ที่ได้มีการพัฒนาไปมาก จากอดีตที่มีการทำงานด้วยระบบอนาล็อก (Analog) มาสู่ยุคปัจจุบันที่ทำงานด้วยระบบดิจิตอล (Digital) หรือยุคของ 2จี (2G), 2.5จี (2.5G) และก้าวไปสู่ยุค 3จี (3G) ในที่สุด จึงทำให้โทรศัพท์เคลื่อนที่ไม่ได้เป็นแค่เพียงอุปกรณ์ที่ใช้สำหรับการสื่อสารเพียงอย่างเดียวเท่านั้น อีกทั้งเทคโนโลยีอินเทอร์เน็ท (Internet) ได้มีการใช้งานกันอย่างกว้างขวาง ทำให้อุปกรณ์โมบายล์ (Mobile Devices) สามารถเข้าถึงอินเทอร์เน็ทได้ไม่ว่าที่ไหน หรือว่า เมื่อไหร่ ซึ่งก็มีหลายๆ แอพพลิเคชันที่ต้องการความปลอดภัยสูงในการติดต่อสื่อสารผ่านอินเทอร์เน็ทเช่น ระบบ M-Commerce, การโอนเงินผ่านอุปกรณ์โมบายล์, ซื้อหนังสือออนไลน์ หรือ จองตั๋วภาพยนตร์ผ่านอุปกรณ์โมบายล์เป็นต้น

โดยลายเซ็นดิจิตอล (Digital Signature) ก็เหมือนกับลายเซ็นปกติ ที่สามารถตรวจสอบได้ว่า ผู้เซ็นคนไหนเป็นคนเซ็นลงบนเอกสารนั้น และตรวจสอบได้อีกว่า เอกสารนั้นเหมือนกับเอกสารตอนที่ผู้เซ็นได้เซ็นไว้หรือไม่ โดยลายเซ็นดิจิตอลนั้นสามารถสร้างความปลอดภัยให้กับแอพพลิเคชันต่างๆได้ ซึ่งโดยปกติจะใช้เทคนิคการเข้ารหัสแบบอสมมาตร (Asymmetric Cryptographic) ซึ่งเป็นการทำงานที่หนักเกินไปสำหรับอุปกรณ์โมบายล์ที่มีพลังงานในการคำนวณที่จำกัดและมีผลกระทบโดยตรงกับเวลาในการใช้งานของแบตเตอร์รี่

เทคโนโลยี Positioning System หรือ PS ซึ่งเป็นเทคโนโลยีที่สามารถบอกตำแหน่งที่ตั้ง ณ ขณะนั้นได้ ในปัจจุบันก็มีอยู่หลายๆระบบด้วยกัน เช่น GPS (Global Positioning System), Hutch Navi เป็นต้น ทำให้ในปัจจุบันมีการรวมเอาเทคโนโลยี PS เข้าไว้กับอุปกรณ์โมบายล์ โดยการเพิ่ม PS Reader เข้าไปในอุปกรณ์ด้วย ทำให้อุปกรณ์โมบายล์ สามารถรู้ตำแหน่งที่ตั้ง ณ ปัจจุบันได้ (Latitude, Longitude) ได้ แต่ก็ไม่ได้มีการนำเทคโนโลยี PS มาใช้ประโยชน์ในการสร้างลายเซ็นดิจิตอลแต่อย่างไร

ในงานวิทยานิพนธ์ชิ้นนี้จะมุ่งเน้นการนำเสนอวิธีในการสร้างลายเซ็นดิจิตอล ซึ่งมีการประยุกต์นำเอาประโยชน์จากตำแหน่งที่ตั้ง ณ ปัจจุบันมามีส่วนร่วมในการสร้างลายเซ็นดิจิตอล ที่เหมาะสมกับอุปกรณ์โมบายล์ ซึ่งมีความสามารถในการประมวลผลต่ำและพลังงานแบตเตอร์รี่ที่จำกัด

1.2 วัตถุประสงค์ของการศึกษา

1. ศึกษาหลักการพื้นฐานของลายเซ็นดิจิตอล
2. ศึกษาวิธีในการสร้างลายเซ็นดิจิตอลแบบต่างๆที่มีอยู่ในปัจจุบัน
3. ศึกษาหลักการทำงานของ Positioning System
4. นำเสนอวิธีการสร้างลายเซ็นดิจิตอลที่เหมาะสมกับอุปกรณ์โมบายล์ โดยมีการประยุกต์ใช้ตำแหน่งที่ตั้งเข้ามามีส่วนร่วมในการสร้างลายเซ็นดิจิตอลด้วย
5. วิเคราะห์ความปลอดภัยและประสิทธิภาพของวิธีที่นำเสนอ และทำการจำลองการหาค่า GeoLock ของอุปกรณ์โมบายล์ที่ ความเร็วและขนาด Grid Size ที่แตกต่างกัน แล้วทำการเปรียบเทียบกับวิธีการที่มีอยู่แล้ว

1.3 สมมุติฐานของการศึกษา

เพื่อให้บรรลุวัตถุประสงค์ของการศึกษา วิทยานิพนธ์ชิ้นนี้ได้นำเสนอวิธีการใช้การสร้าง SPACE-TIME BASED DIGITAL SIGNATURE สำหรับอุปกรณ์โมบายล์ ตามทฤษฎีที่เกี่ยวข้อง เพื่อลดเวลาในการประมวลผลทางฝั่งอุปกรณ์โมบายล์ โดยมี Sign Server ทำหน้าที่ในการช่วยอุปกรณ์โมบายล์สร้างลายเซ็นดิจิตอล และ คอยรับข้อมูลอัพเดทสถานะการเคลื่อนที่จากอุปกรณ์โมบายล์ โดยในที่นี้เราถือว่า Sign Server มีความสามารถในการประมวลผลสูง และมีพลังงานที่มากเมื่อเทียบกับอุปกรณ์โมบายล์ ซึ่งเมื่อเอาปัจจัยทางภูมิศาสตร์เข้ามามีส่วนร่วมในการระบวนการสร้าง ลายเซ็นดิจิตอล ซึ่งเป็นการเพิ่มความยากในการแฮ็คหรือโจมตีระบบของเรา จึงเป็นการเพิ่มความปลอดภัยทั้งในการสร้าง และตรวจสอบความถูกต้องของลายเซ็นดิจิตอลและอาจสามารถนำไปประยุกต์ใช้กับ Location Based Services ต่างๆ ต่อไปได้

1.4 ขอบเขตของงานวิจัย

วิทยานิพนธ์ชิ้นนี้ได้ศึกษาหลักการทำงานและวิธีการสร้างลายเซ็นดิจิตอล และศึกษาการทำงานของ Positioning System ต่างๆ รวมไปถึงงานวิจัยที่เกี่ยวข้องกับการสร้างลายเซ็นดิจิตอล และ Location Based Services ต่างๆด้วย แล้วจึงนำเสนอ “ลายเซ็นดิจิตอลสำหรับอุปกรณ์โมบายล์โดยใช้ปัจจัยทางตำแหน่งและเวลา” โดยมุ่งเน้นไปที่การนำเสนอแนวความคิดใหม่ในการนำปัจจัยทางภูมิศาสตร์ซึ่งเป็นจุดเด่นของอุปกรณ์โมบายล์นั้นก็คือ สามารถเคลื่อนที่ได้ และเวลา เข้ามามีส่วนร่วมในการสร้างลายเซ็นดิจิตอลได้ และลดกระประมวลผลทางฝั่งอุปกรณ์โมบายล์ เพื่อให้เหมาะสมกับอุปกรณ์โมบายล์ และเปรียบเทียบกับวิธีการอื่นๆที่มีอยู่แล้ว

1.5 ขั้นตอนการศึกษา

1. ศึกษาหลักการของลายเซ็นดิจิตอล
2. ทำการศึกษาหลักการของทำงานของ Positioning System
3. ศึกษางานวิจัยที่คล้ายคลึงหรือสอดคล้องกับงานวิจัยนี้
4. นำเสนอวิธีการสร้างลายเซ็นดิจิตอลที่เหมาะสมกับ Mobile Devices โดยใช้ปัจจัยทางภูมิศาสตร์เข้ามามีส่วนร่วม พร้อมจำลองการหาค่า GeoLock ของอุปกรณ์โมบายล์ที่ความเร็วและ ขนาด Grid Size ที่แตกต่างกัน
5. สรุปและวิเคราะห์วิธีการที่นำเสนอ

1.6 ข้อตกลงเบื้องต้น

ข้อตกลงเบื้องต้นของวิทยานิพนธ์นี้ก็คือ ทั้งผู้ส่งและผู้รับเป็นอุปกรณ์โมบายล์ และสามารถส่งตำแหน่งที่ตั้งของตนไปยัง Sign Server ได้อย่างปลอดภัย เมื่อจำเป็น และเนื่องจาก Sign Server ช่วยอุปกรณ์โมบายล์ในการสร้างลายเซ็นดิจิตอลจึงจำเป็นต้องมีช่องทางการติดต่อสื่อสารกันระหว่างอุปกรณ์โมบายล์ด้วย และในส่วนของเปรียบเทียบประสิทธิภาพของงานที่นำเสนอนั้น จะเปรียบเทียบที่เวลาในการประมวลผล จำนวนครั้งในการทำ modular exponentiation, modular multiplication, hash function และ inverse computation โดยตั้งสมมุติฐานตาม [10], [16], [17] และ [18]

1.7 รายละเอียดของวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้ เป็นการนำเสนอวิธีใหม่ในการสร้างลายเซ็นดิจิตอล ที่เหมาะสมสำหรับอุปกรณ์โมบายล์ โดยมีการนำปัจจัยทางภูมิศาสตร์เข้ามามีส่วนร่วมในกระบวนการสร้างด้วย โดยรายละเอียดต่างๆภายในวิทยานิพนธ์นี้ได้จัดแบ่งในส่วนเนื้อหาออกเป็น 5 บท ซึ่งแต่ละบทมีหัวข้อและเนื้อหาดังต่อไปนี้

บทที่ 1 "บทนำ" อธิบายถึงวัตถุประสงค์ สมมุติฐานของการศึกษา ขอบเขตและขั้นตอนการศึกษา รวมไปถึงรายละเอียดเนื้อหาโดยสรุปของแต่ละบท

บทที่ 2 "หลักการและงานวิจัยที่เกี่ยวข้อง" อธิบายหลักการพื้นฐานของการเข้ารหัสข้อมูลหรือ Cryptography รวมไปถึงหลักการทำงานของ Positioning System ที่มีอยู่ในปัจจุบัน และงานวิจัยที่เกี่ยวข้องด้วย

บทที่ 3 "งานที่นำเสนอ" บทนี้จะเสนอวิธีในการสร้าง Space-time based Digital Signature ที่นำเสนอ และอธิบายถึงพารามิเตอร์ที่ใช้ในการอัพเดทสถานะการเคลื่อนที่ การคำนวณพารามิเตอร์ต่างๆ และหลักเกณฑ์ในการอัพเดทข้อมูลสถานะการเคลื่อนที่

บทที่ 4 “วิเคราะห์งานนำเสนอ” กล่าวถึงการวิเคราะห์ความปลอดภัย และ ประสทธิภาพของงานที่นำเสนอ พร้อมจำลองการหาค่า GeoLock ของอุปกรณ์โมบายล์ทีความเร็วและ ขนาด Grid Size ที่แตกต่างกัน โดยมีเปรียบเทียบเวลาในการประมวลผลในการสร้างและตรวจสอบความถูกต้องของลายเซ็นดิจิตอล ระหว่างวิธีที่นำเสนอกับวิธีที่มีอยู่แล้ว

บทที่ 5 “บทสรุป” เป็นการสรุปและวิจารณ์ผลที่ได้จากการวิเคราะห์ของงานที่นำเสนอ

บทที่ 2
หลักการและงานวิจัยที่เกี่ยวข้อง

2.1 หลักการพื้นฐานของการเข้ารหัสข้อมูล (Cryptography)

ส่วนนี้จะอธิบายถึงความรู้พื้นฐานเกี่ยวกับการเข้ารหัสข้อมูล รวมทั้งครอบคลุมไปถึงอัลกอริทึมที่ใช้ในการเข้ารหัสที่สำคัญๆ โดยจุดประสงค์ที่สำคัญในการเข้ารหัสข้อมูลคือ

- การรักษาความลับของข้อมูล (Confidentiality) คือการรับรองว่าจะมีการเก็บข้อมูลไว้เป็นความลับ และผู้ที่มีสิทธิเท่านั้นที่จะสามารถเข้าถึงข้อมูลได้
- การรักษาความถูกต้องของข้อมูล (Integrity) เพื่อป้องกันข้อมูลในอยู่ในสภาพเดิมอย่างสมบูรณ์ คือ ผู้รับ (Receiver) จะต้องได้รับข้อมูลที่ถูกต้องตามที่ผู้ส่ง (Sender) ส่งมาให้ โดยข้อมูลจะต้องไม่มีการสูญหายหรือถูกเปลี่ยนแปลงแต่อย่างใด
- การทำให้สามารถพิสูจน์ตัวตนของผู้ส่งข้อมูลได้ (Authentication/Nonrepudiation) เพื่อให้สามารถตรวจสอบได้ว่าใครคือผู้ส่งข้อมูล หรืออีกทางก็คือ เพื่อป้องกันการแอบอ้างได้

การเข้ารหัสข้อมูล (Cryptography) การเข้ารหัสข้อมูลโดยพื้นฐานแล้วจะเกี่ยวข้องกับวิธีการทางคณิตศาสตร์เพื่อใช้ในการป้องกันข้อมูลหรือข้อความตั้งต้นที่ต้องการส่งไปถึงผู้รับ ข้อมูลตั้งต้นจะถูกแปรเปลี่ยนไปสู่ข้อมูลหรือข้อความอีกรูปแบบหนึ่งที่ไม่สามารถอ่านเข้าใจได้โดยใครก็ตามที่ไม่มีคีย์สำหรับเปิดดูข้อมูลนั้น เราเรียกกระบวนการในการแปรรูปของข้อมูลตั้งต้นว่า "การเข้ารหัสข้อมูล" (Encryption) และกระบวนการในการแปลงข้อความที่ไม่สามารถอ่าน และทำความเข้าใจให้กลับไปสู่ข้อความดั้งเดิม ว่าการถอดรหัสข้อมูล (Decryption) อัลกอริทึมในการเข้ารหัส สามารถแบ่งออกเป็น 2 ประเภทหลัก ๆ คือ

2.1.1 การเข้ารหัสแบบสมมาตร (Symmetric Encryption)

การเข้ารหัสแบบสมมาตร นั้นจะใช้คีย์ลับ (Secret Key) เพื่อใช้ในการเข้าและถอดรหัสข้อความที่ส่งไป โดยการเข้ารหัสแบบสมมาตร ยังสามารถแบ่งออกได้เป็น 2 ประเ ภทด้วยกันคือ แบบบล็อค (Block Algorithms) ซึ่งจะทำการเข้ารหัสทีละบล็อค (1 บล็อคประกอบด้วยหลายๆ ไบต์ เช่น 64 ไบต์ เป็นต้น) และ แบบสตรีม (Stream Algorithms) ซึ่งจะทำการเข้ารหัสทีละไบต์

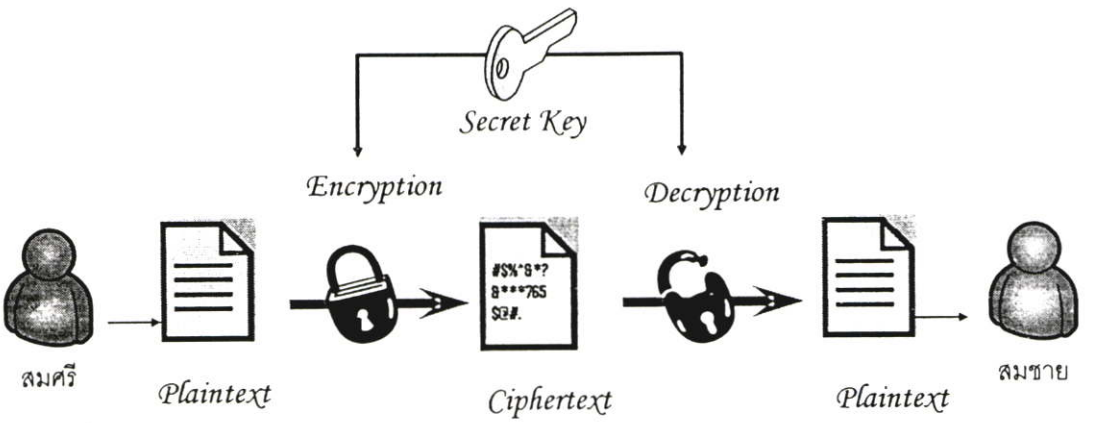
หลักการทำงานของ Symmetric Encryption

ในกระบวนการเข้ารหัสแบบสมมาตร นั้นจะมีส่วนประกอบอยู่ด้วยกันทั้งหมด 5 ส่วน คือ

- Plaintext คือ ข้อความเริ่มแรกที่ยังไม่ได้เข้ารหัส และต้องการที่จะเข้ารหัส
- Encryption Algorithm เป็นกระบวนการแปลงข้อความตั้งต้น ไปเป็นข้อความที่ไม่สามารถอ่านได้ (Ciphertext)
- Secret Key คือคีย์ใช้เพื่อป้อนเข้าไปในกระบวนการเข้ารหัส เพื่อแปลงข้อความจากข้อความตั้งต้น ไปเป็นข้อความที่ไม่สามารถอ่านได้ และต้องเก็บคีย์ เป็นความลับ
- Ciphertext คือข้อความที่ไม่สามารถอ่านได้ เป็นผลลัพธ์ที่ได้จากการเข้ารหัส โดยจะขึ้นกับข้อความตั้งต้น และคีย์ลับที่ป้อนเข้าไป ข้อความที่เหมือนกันแต่มีคีย์ ที่ต่างกัน ก็ได้ผลลัพธ์ที่ต่างกัน
- Decryption Algorithm เป็นกระบวนการย้อนกลับของการเข้ารหัส โดยเมื่อใส่ข้อความที่ไม่สามารถอ่านได้ และคีย์ลับที่ถูกต้องเข้าไปในกระบวนการถอดรหัส ก็จะต้องได้ข้อความตั้งต้นที่ถูกต้องกลับออกมา

ยกตัวอย่างการทำงานของการเข้ารหัสแบบสมมาตร เช่น เมื่อสมศรีต้องการส่งข้อความ ไปหาสมชาย โดยที่สมศรีไม่ต้องการใครอ่านข้อความนี้ได้นอกจากสมชายเท่านั้น สิ่งที่เกิดขึ้นก็คือ

1. ทั้ง สมศรี และสมชายจะต้องตกลงก่อนว่าจะใช้อัลกอริทึมอะไร
2. ทั้ง สมศรี และสมชายจะต้องตกลงคีย์ลับระหว่างทั้งสองคน
3. หลังจากนั้น สมศรีต้องนำข้อความตั้งต้น มาทำกระบวนการเข้ารหัสตามอัลกอริทึม และคีย์ที่ได้ตกลงกันไว้ เพื่อให้ได้เป็นข้อความที่อ่านไม่ได้
4. สมศรีส่งข้อความที่อ่านไม่ได้ไปหาสมชาย
5. สมชายนำข้อความที่อ่านไม่ได้ที่ได้ ไปทำกระบวนการถอดรหัสตามอัลกอริทึม และคีย์ที่ได้ตกลงกันไว้ เพื่อให้ได้ออกมาเป็นข้อความตั้งต้น แล้วจึงอ่านมัน



รูปที่ 2.1 การเข้ารหัสแบบสมมาตร ใช้คีย์ตัวเดียวกันในการเข้ารหัส และ ถอดรหัสข้อความ

อัลกอริทึมสำหรับการเข้ารหัสแบบสมมาตร (Symmetric Encryption Algorithm)

อัลกอริทึมสำหรับการเข้ารหัสแบบสมมาตร ในปัจจุบันมีเป็นจำนวนมาก โดยในวิทยานิพนธ์ฉบับนี้จะนำเสนอเพียงจำนวนหนึ่งเท่านั้น ซึ่งเป็นอัลกอริทึมที่เป็นที่รู้จักกันดีในวงการของการเข้ารหัสข้อมูล

- **Data Encryption Standard Algorithm**

DES ย่อมาจาก Data Encryption Standard อัลกอริทึมนี้ได้รับการรับรองโดยรัฐบาลสหรัฐอเมริกาในปี ค.ศ. 1977 ให้เป็นมาตรฐานการเข้ารหัสข้อมูลสำหรับหน่วยงานของรัฐทั้งหมด ในปี 1981 อัลกอริทึมนี้ยังได้รับการกำหนดให้เป็นมาตรฐานการเข้ารหัสข้อมูลในระดับนานาชาติตามมาตรฐาน ANSI (American National Standards) อีกด้วย

DES เป็นอัลกอริทึมแบบบล็อกซึ่งใช้คีย์ที่มีขนาดความยาว 56 บิตและเป็นอัลกอริทึมที่มีความแข็งแกร่ง แต่เนื่องด้วยขนาดความยาวของคีย์ที่มีขนาดเพียง 56 บิต ซึ่งในปัจจุบันถือได้ว่าสั้นเกินไป ผู้บุกรุกอาจใช้วิธีการลองผิดลองถูกเพื่อค้นหาคีย์ที่ถูกต้องสำหรับการถอด รหัสได้

ในปี 1998 ได้มีการสร้างเครื่องคอมพิวเตอร์พิเศษขึ้นมาซึ่งมีมูลค่า 250,000 เหรียญสหรัฐเพื่อใช้ในการค้นหาคีย์ที่ถูกต้องของการเข้ารหัสข้อมูลหนึ่งๆ ด้วย DES และพบว่าเครื่องคอมพิวเตอร์นี้สามารถค้นหาคีย์ที่ถูกต้องได้ภายในระยะเวลาไม่ถึงหนึ่งวัน

- **Triple-DES Algorithm**

Triple-DES เป็นอัลกอริทึมที่เสริมความปลอดภัยของ DES ให้มีความแข็งแกร่งมากขึ้น โดยใช้อัลกอริทึม DES เป็นจำนวนสามครั้งเพื่อทำการเข้ารหัส แต่ละครั้งจะใช้คีย์ในการเข้ารหัสที่แตกต่างกัน ดังนั้นจึงเปรียบเสมือนการใช้คีย์เข้ารหัสที่มีความยาวเท่ากับ 56*3 = 168 บิต Triple-DES ได้ถูก ใช้งานกับสถาบันทางการเงินอย่างแพร่หลาย รวมทั้งใช้งานกับโปรแกรม Secure Shell (SSH) ด้วย

การใช้อัลกอริทึม DES เพื่อเข้ารหัสเป็นจำนวนสองครั้งด้วยคีย์สองตัว (56*2=112 บิต) ยังถือได้ว่าไม่ปลอดภัยอย่างพอเพียง

- **Blowfish Algorithm**

Blowfish เป็นอัลกอริทึมที่มีความรวดเร็วในการทำงาน มีขนาดเล็กกระทัดรัด และใช้การเข้ารหัสแบบบล็อค ผู้พัฒนาคือ Bruce Schneier อัลกอริทึมสามารถใช้คีย์ที่มีขนาดความยาวตั้งแต่ไม่มากนักไปจนถึงขนาด 448 บิต ซึ่งทำให้เกิดความยืดหยุ่นสูงในการเลือกใช้คีย์ รวมทั้งอัลกอริทึมยังได้รับการออกแบบมาให้ทำงานอย่างเหมาะสมกับหน่วยประมวลผลขนาด 32 หรือ 64 บิต Blowfish ได้เปิดเผยสู่สาธารณะและไม่ได้มีการจดสิทธิบัตรใดๆ นอกจากนั้นยังใช้ในโปรแกรม SSH และอื่นๆ

- **International Data Encryption Algorithm**

IDEA ย่อมาจาก International Data Encryption Algorithm อัลกอริทึมนี้ได้รับการพัฒนาในประเทศสวิสเซอร์แลนด์ที่เมือง Zarich โดย James L. Massey และ Xuejia Lai และได้รับการตีพิมพ์เผยแพร่ในปี ค.ศ. 1990 อัลกอริทึมใช้คีย์ที่มีขนาด 128 บิต และได้รับการใช้งานกับโปรแกรมยอดฮิตสำหรับการเข้ารหัสและลงลายมือชื่ออิเล็กทรอนิกส์ในระบบอีเมล์ที่มีชื่อว่า PGP ต่อมา IDEA ได้รับการจดสิทธิบัตรทางด้านซอฟต์แวร์โดยบริษัท Ascom-Tech AG ในประเทศสวิสเซอร์แลนด์ ซึ่งทำให้การนำไปใช้ในงานต่างๆ เริ่มลดลง ทั้งนี้เนื่องจากติดปัญหาเรื่องลิขสิทธิ์นั่นเอง

- **RC4 Algorithm**

อัลกอริทึมนี้เป็นอัลกอริทึมแบบ Stream (ทำงานกับข้อมูลทีละไบต์) ซึ่งได้รับการพัฒนาขึ้นมาโดย Ronald Riverst และถูกเก็บเป็นความลับทางการค้าโดยบริษัท RSA Data Security ในภายหลังอัลกอริทึมนี้ได้รับการเปิดเผยใน Usenet เมื่อปี ค.ศ. 1994 และเป็นที่ทราบกันว่าเป็นอัลกอริทึมที่มีความแข็งแกร่งโดยสามารถใช้ขนาดความยาวของคีย์ที่มีขนาดตั้งแต่ 1 บิตไปจนกระทั่งถึงขนาด 2048 บิต

- **Rijndael Algorithm (AES Algorithm)**

อัลกอริทึมนี้ได้รับการพัฒนาโดย Joan Daemen และ Vincent Rijmen ในปี 2000 อัลกอริทึมได้รับการคัดเลือกโดยหน่วยงาน National Institute of Standard and Technology (NIST) ของสหรัฐอเมริกาให้เป็นมาตรฐานในการเข้ารหัสชั้นสูงของประเทศ อัลกอริทึมมีความเร็วสูงและมีขนาดกะทัดรัดโดยสามารถใช้คีย์ที่มีความยาวขนาด 128, 192 และ 256 บิต

- **One-time Pads Algorithm**

อัลกอริทึมนี้ได้รับการยอมรับว่าเป็นอัลกอริทึมที่ไม่มีใครสามารถเจาะความแข็งแกร่งของอัลกอริทึมได้ อัลกอริทึมใช้คีย์ที่มีขนาดความยาวซึ่งอาจจะมากกว่าขนาดความยาวของข้อความที่ต้องการเข้ารหัส คีย์จะถูกสร้างออกมาแบบสุ่มและโดยปกติจะถูกใช้งานแค่เพียงครั้งเดียวแล้วทิ้งไป แต่ละไบต์ของข้อความที่ต้องการส่งไปจะถูกเข้าและถอดรหัสชนิดไบต์ต่อไบต์ของคีย์ที่ถูกสร้างขึ้นมาใช้งาน เนื่องจากคีย์ที่ถูกใช้งานแต่ละครั้งจะไม่ซ้ำกันและถูกสร้างขึ้นมาแบบสุ่ม จึงเป็นการยากที่จะค้นหาคีย์ที่ถูกต้องได้

ข้อจำกัดของอัลกอริทึมนี้ คือขนาดของคีย์ที่อาจมีขนาดยาวกว่าข้อความที่ต้องการส่ง ซึ่งส่งผลให้การส่งมอบคีย์ที่มีขนาดใหญ่ทำได้ไม่สะดวกนัก รวมทั้งการสร้างคีย์ให้มีความสุ่มสูงไม่ใช่เป็นสิ่งที่ทำได้ง่ายนัก อย่างไรก็ตามอัลกอริทึมนี้ก็ยังมีการใช้งานในระบบเครือข่ายที่ต้องการความปลอดภัยสูง

ปัญหาของการเข้ารหัสแบบสมมาตร

อัลกอริทึมแบบสมมาตร นั้นมีความสำคัญไม่น้อยไปกว่าอัลกอริทึมแบบอสมมาตร ทั้งนี้เนื่องจากอัลกอริทึมแบบแรกทำงานได้รวดเร็วกว่าและง่ายต่อการใช้งานกว่าแบบหลัง อย่างไรก็ตามอัลกอริทึมแบบสมมาตรยังมีปัญหาที่สำคัญ 3 ประการ ซึ่งเป็นข้อจำกัดในการใช้งานอัลกอริทึมนี้

- ในการใช้งานอัลกอริทึมนี้ สมมุติว่า มีคนสองคนต้องการแลกเปลี่ยนข้อมูลกัน (เช่น สมศรี กับ สมชาย) จำเป็นต้องแลกเปลี่ยนคีย์ลับกันก่อน (ซึ่งอาจหมายถึงส่งมอบคีย์ลับให้กับอีกคนหนึ่ง) การแลกเปลี่ยนคีย์นั้นอาจทำได้อย่างยุ่งยากและไม่สะดวก
- ทั้งสองคนต้องรักษาคีย์ลับนั้นไว้เป็นอย่างดี ห้ามเปิดเผยให้ผู้อื่นล่วงรู้โดยเด็ดขาด การที่คีย์ถูกเปิดเผยออกไปสู่ผู้อื่น (จะโดยคนใดคนหนึ่งก็ตาม) และอีกคนหนึ่งไม่ได้รับทราบปัญหานี้ อาจก่อให้เกิดปัญหากับคนที่ไม่ทราบนี้ได้ เช่น สมศรีอาจส่งข้อความที่เป็นความลับไปให้กับยังสมชาย แต่ข้อความนี้อาจถูกเปิดเผยได้โดยใช้คีย์ลับที่ล่วงรู้โดยผู้อื่น
- สำหรับสองกลุ่มที่ต้องการติดต่อกัน จำเป็นต้องใช้คีย์ลับเป็นจำนวน 1คีย์เพื่อติดต่อกัน สมมติว่ามีผู้ที่ต้องติดต่อกันเป็นจำนวน n กลุ่ม จำนวนคีย์ลับทั้งหมดที่ต้องแลกเปลี่ยนกันคิดเป็นจำนวนทั้งหมด C 2n หรือเท่ากับ n (n-1)/2คีย์ซึ่งจะเห็นได้ว่าจำนวนคีย์มีมากมายเกินไป ซึ่งอาจก่อให้เกิดปัญหาด้านการรักษาความปลอดภัยให้กับคีย์เหล่านี้

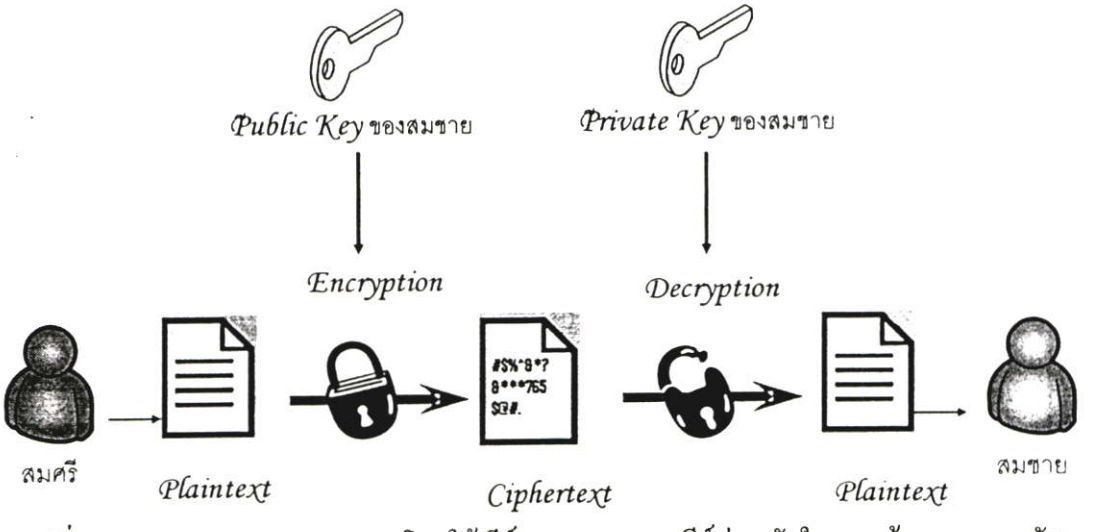
จากที่กล่าวมา จะเห็นได้ว่าความปลอดภัยของการเข้ารหัสแบบสมมาตรนั้น จะขึ้นอยู่กับการรักษาความลับของคีย์ลับไม่ใช่การรักษาความลับของอั่ลกอริทึมในการเข้ารหัส ทั้งนี้เนื่องจากการได้มาซึ่งอัลกอริทึม 1 ตัวที่มีความสามารถตามความต้องการของการเข้ารหัสแบบสมมาตรนั้นไม่ใช่เรื่องง่ายๆ ดังนั้นหากใช้วิธีให้ความลับอยู่ที่อัลกอริทึมแล้ว หากถูกเปิดเผยก็จะต้องสร้าง

อัลกอริทึมใหม่ขึ้นมา ในขณะที่ใช้วิธีรักษาความลับที่ตัวคีย์แล้ว หากคีย์มีการเปิดเผย ก็เพียงสร้างคีย์ใหม่ขึ้นมาเท่านั้น ซึ่งเป็นเรื่องที่ง่ายกว่ามาก

2.1.2 การเข้ารหัสแบบอสมมาตร (Asymmetric Encryption)

ในปี 1976 Whitfield Diffie และ Martin Hellman ได้เสนอวิธีการหนึ่งขึ้นในบทความที่ชื่อ "New Directions in Cryptography" โดยลงในหนังสือ IEEE Transaction on Information Theory ฉบับเดือนพฤศจิกายน 1976 โดยวิธีการที่นำเสนอคือ การกำหนดให้มีคีย์ทั้งหมด 2 ตัว คือ คีย์ส่วนตัว (Private Key) และ คีย์สาธารณะ (Public Key) โดยคีย์ทั้งสองต้องเป็นคู่ของกันและกัน โดยหากข้อมูลถูกเข้ารหัสด้วยคีย์ตัวหนึ่ง จะต้องสามารถถอดรหัสได้ด้วยคีย์อีกตัวหนึ่ง

และจากการที่มีคีย์อยู่ 2 ตัวนั้น เวลาใช้งานจะต้องเก็บ คีย์ส่วนตัวไว้เป็นความลับ ห้ามเปิดเผยให้ผู้อื่นทราบโดยเด็ดขาด และแจกคีย์สาธารณะให้กับผู้อื่น เช่น เพื่อนร่วมงานที่เราต้องการติดต่อด้วย หรือ แม้กระทั่งวางไว้บนเว็บไซต์เพื่อผู้อื่นสามารถดาวน์โหลดไปใช้งานได้ โดยไม่ต้องปกปิดเป็นความลับแต่อย่างใด เนื่องด้วย ข้อความใดก็ตามที่เข้ารหัสด้วย คีย์สาธารณะจะสามารถถอดรหัสได้ คีย์ส่วนตัวที่เป็นคู่กันเท่านั้น ดังนั้น เมื่อสมศรีต้องการส่งข้อความไปยังสมชายโดยที่ไม่ต้องการให้คนอื่นสามารถอ่านได้ ก็สามารถทำได้โดยการนำข้อความตั้งต้นมาเข้ารหัสด้วย คีย์สาธารณะของสมชาย เพื่อให้ได้ข้อความที่ไม่สามารถอ่านได้ มา แล้วจึงส่งไปให้สมชาย ซึ่งข้อความที่ไม่สามารถอ่านได้ที่ได้นั้นจะสามารถถอดรหัสได้ด้วย คีย์ส่วนตัวของสมชายเท่านั้น และก็มีแต่สมชายเท่านั้นที่มี คีย์ส่วนตัวอยู่ ทำให้มีแต่สมชายเท่านั้นที่สามารถอ่านข้อความที่สมศรีส่งมาหาได้ ในกรณีที่สมชายต้องการส่งข้อความหาสมศรี ก็สามารถทำได้เหมือนกัน



รูปที่ 2.2 Asymmetric Encryption โดยใช้ คีย์สาธารณะและ คีย์ส่วนตัวในการเข้าและถอดรหัส

การเข้ารหัสแบบอสมมาตรช่วยแก้ปัญหาของการเข้ารหัสแบบสมมาตรได้ทั้งหมด โดยการที่สามารถเผยแพร่ คีย์สาธารณะในสถานที่ต่างๆ ได้ ทำให้ลดความยุ่งยากในการแลกเปลี่ยนคีย์กันซึ่งเป็นปัญหาข้อแรกของการเข้ารหัสแบบการเข้ารหัสแบบสมมาตร สำหรับปัญหาที่ว่าทั้งสองฝ่ายจะต้องรักษาคีย์ลับไว้เป็นอย่างดีนั้น วิธีการของ คีย์สาธารณะจะทำให้ผู้ที่ต้องรับผิดชอบเหลือเพียงผู้เดียว กล่าวคือ ผู้ถือ คีย์ส่วนตัวซึ่งห้ามให้ผู้อื่นล่วงรู้โดยเด็ดขาด

สำหรับปัญหาที่สามที่ว่าจำนวนคีย์ลับที่จำเป็นต้องใช้มีมากมายเกินไป วิธีการของ คีย์สาธารณะจะใช้จำนวนคีย์ที่ประหยัดกว่า เนื่องจาก คีย์สาธารณะ1 ตัวของกลุ่มๆ หนึ่งจะสามารถเผยแพร่ให้กับกี่กลุ่มก็ได้ที่เราต้องการติดต่อด้วย (แทนที่จะเป็น 1คีย์ลับต่อสองกลุ่มที่ต้องการติดต่อกัน) ดังนั้นถ้ามีกลุ่มที่ต้องติดต่อกันจำนวน n กลุ่ม จำนวน คีย์ส่วนตัวที่ต้องระวังรักษาก็คือ nคีย์ซึ่งจะเห็นได้ว่าลดลงไปได้เป็นจำนวนมาก

ข้อเสียที่สำคัญของการเข้ารหัสแบบอสมมาตร คือ ต้องใช้เวลาในการคำนวณการเข้าและถอดรหัส เมื่อเทียบกับการเข้ารหัสแบบสมมาตร และอาจใช้เวลาเป็นพันเท่าของเวลาที่ใช้โดยระบบการเข้ารหัสแบบสมมาตร

นอกจากนี้การเข้ารหัสแบบอสมมาตรยังสามารถประยุกต์ใช้กับการลงลายมือชื่ออิเล็กทรอนิกส์ (ซึ่งเปรียบเสมือนการลงลายมือชื่อของเราที่ใช้กับเอกสารสำนักงานทั่วไป

อัลกอริทึมสำหรับการเข้ารหัสแบบอสมมาตร (Asymmetric Encryption Algorithm)

อัลกอริทึมสำหรับการเข้ารหัสแบบอสมมาตร ในปัจจุบันมีเป็นจำนวนมาก โดยในวิทยานิพนธ์ฉบับนี้จะนำเสนอเพียงจำนวนหนึ่งเท่านั้น ซึ่งเป็นอัลกอริทึมที่เป็นที่รู้จักกันดีในวงการของการเข้ารหัสข้อมูล

- **RSA Algorithm [14]**

อัลกอริทึม RSA ถือได้ว่าเป็นการเข้ารหัสแบบ คีย์สาธารณะที่ได้รับความนิยมมากที่สุด ซึ่งได้รับการพัฒนาขึ้นที่มหาวิทยาลัย MIT ในปี 1977 โดยศาสตราจารย์ 3 คน ซึ่งประกอบด้วย Ronald Rivest, Adi Shamir และ Leonard Adleman โดย RSA เป็นลิขสิทธิ์ของบริษัท RSA Security ชื่อของอัลกอริทึมได้รับการตั้งชื่อตามตัวอักษรตัวแรกของนามสกุลของศาสตราจารย์ทั้งสามคน อัลกอริทึมนี้สามารถใช้ในการเข้ารหัสข้อมูลรวมทั้งการลงลายมือชื่ออิเล็กทรอนิกส์ด้วย

- **DSS Algorithm [15]**

DSS ย่อมาจาก Digital Signature Standard อัลกอริทึมนี้ได้รับการพัฒนาขึ้นมาโดย National Security Agency ในประเทศสหรัฐอเมริกาและได้รับการรับรองโดย NIST (National Institute of Standard and Technology) ให้เป็นมาตรฐานกลางสำหรับการลงลายมือชื่ออิเล็กทรอนิกส์ในประเทศสหรัฐอเมริกา

ในเรื่องของความยาวคีย์นั้น หากเป็นการเข้ารหัสในแบบ Symmetric Encryption ถ้าความยาวคีย์ที่เพิ่มขึ้นทุก 1 bit จะทำให้ต้องใช้เวลาในการถอดรหัสเพิ่มขึ้นเป็น 2 เท่า เช่น สมมติว่าหากความยาวคีย์50 bit ต้องใช้เวลาในการถอดรหัส 100 ปี ดังนั้น ถ้าเพิ่มความยาวคีย์เป็น 51 bit ก็ต้องใช้เวลาในการถอดรหัส 200 ปีนั้นเอง แต่สำหรับความยาวคีย์ของ Asymmetric Encryption จะมีค่าบางค่าเท่านั้นที่สามารถเป็นคีย์ได้ เช่น ถ้าเพิ่มความยาวคีย์จาก 428 bit เป็น 429 bit อาจไม่ทำให้เวลาที่ใช้เพิ่มขึ้นเลยก็ได้ แต่สำหรับ RSA ที่มีความยาวคีย์1028 bit ขึ้นไป ถือว่ามีแข็งแกร่งมากพอในงานทุกๆด้าน

2.1.3 ความแข็งแกร่งของอัลกอริทึมสำหรับการเข้ารหัส

ความแข็งแกร่งของอัลกอริทึม หมายถึง ความยากในการที่ผู้บุกรุกจะสามารถถอดรหัสข้อมูลได้โดยปราศจากคีย์ที่ใช้ในการเข้ารหัส ซึ่งจะขึ้นอยู่กับปัจจัยดังนี้

- การเก็บคีย์ ที่ใช้ในการเข้ารหัสไว้อย่างเป็นความลับ ผู้เป็นเจ้าของคีย์ลับหรือ คีย์ส่วนตัวต้องระมัดระวังไม่ให้คีย์สูญหายหรือล่วงรู้โดยผู้อื่น
- ความยาวของคีย์เข้ารหัส ปกติคีย์ที่ใช้ในการเข้ารหัสจะมีความยาวเป็นบิต ยิ่งจำนวนบิตของคีย์ยิ่งมาก ยิ่งทำให้การเดาเพื่อสุ่มหาคีย์ที่ถูกต้องเป็นไปได้ยากยิ่งขึ้น (เช่นคีย์ขนาด 1 บิต จะสามารถแทนตัวเลขได้ 2 ค่าคือ 0 กับ 1 ส่วนคีย์ขนาด 2 บิต จะเป็นไปได้ 4 ค่าคือ 0, 1, 2, 3 เป็นต้น)
- ความไม่เกรงกลัวต่อการศึกษาหรือดูอัลกอริทึมเพื่อหารูปแบบของการเข้ารหัส อัลกอริทึมที่ดีต้องเปิดให้ผู้รู้ทำการศึกษาในรายละเอียดได้โดยไม่เกรงว่าผู้ศึกษาจะสามารถจับรูปแบบของการเข้ารหัสได้
- การมีประตูลับในอัลกอริทึม อัลกอริทึมที่ดีต้องไม่แฝงไว้ด้วยประตูลับที่สามารถใช้เป็นทางเข้าไปสู่อัลกอริทึม แล้วอาจใช้เพื่อทำการถอดรหัสข้อมูลได้ ประตูลับนี้ทำให้ไม่จำเป็นต้องใช้คีย์ในการถอดรหัส
- ความไม่เกรงกลัวต่อปัญหาการหาความสัมพันธ์ในข้อมูลที่ได้รับ กล่าวคือเมื่อผู้บุกรุกทราบข้อมูลบางอย่างที่เป็นข้อมูลตั้งต้นซึ่งยังไม่ได้เข้ารหัส รวมทั้งมีข้อมูลที่เข้ารหัสแล้ว (ของข้อมูลตั้งต้นนั้น) ผู้บุกรุกอาจจะสามารถหาความสัมพันธ์ระหว่างข้อความทั้งสองนั้นได้ ซึ่งจะเป็นวิธีการในการถอดรหัสข้อมูลได้ ปัญหานี้เรียกกันว่า Known plaintext attack
- คุณสมบัติของข้อความตั้งต้น คุณสมบัตินี้อาจใช้เป็นช่องทางในการถอดรหัสข้อมูลได้ อัลกอริทึมที่ดีต้องไม่ใช้คุณสมบัติของข้อความเป็นกลไกในการเข้ารหัสข้อมูล

คำแนะนำในการเลือกใช้อัลกอริทึมคือให้ใช้อัลกอริทึมที่ได้มีการใช้งานมาเป็นระยะเวลานานแล้ว ทั้งนี้เนื่องจากหากปัญหาของอัลกอริทึมนี้มีจริง ก็คงเกิดขึ้นมานานแล้วและก็คงเป็นที่ทราบกันแล้ว นั่นคืออย่างน้อยที่สุดจวบจนกระทั่งถึงปัจจุบัน ก็ยังไม่มีการบุกรุกที่ทำให้อัลกอริทึมนั้นไม่สามารถใช้งานได้อย่างปลอดภัยเป็นที่ประจักษ์ ดังนั้นจึงไม่ควรใช้อัลกอริทึมใหม่ๆ ที่เพิ่งได้มีการนำเสนอกันสู่สาธารณะ เพราะอาจมีช่องโหว่แฝงอยู่และยังไม่เป็นที่ทราบในขณะนี้

2.1.4 ความยาวของคีย์ที่ใช้ในการเข้ารหัส

ความยาวของคีย์ที่ใช้ในการเข้ารหัสมีหน่วยนับเป็นบิต หนึ่งบิตในคอมพิวเตอร์เป็นตัวเลขฐานสองที่ประกอบด้วยค่า 0 และ 1 คีย์ที่มีความยาว 1 บิต ตัวเลขที่เป็นไปได้เพื่อแทนคีย์นั้นจึงอาจมีค่าเป็น 0 หรือ 1 ส่วนคีย์ที่มีความยาว 2 บิต ตัวเลขที่เป็นไปได้จึงเป็น 0, 1, 2 และ 3 ตามลำดับ และคีย์ที่มีความยาว 3 บิต ตัวเลขที่เป็นไปได้จะอยู่ระหว่าง 0 ถึง 7 ดังนั้นเมื่อเพิ่มความยาวของคีย์ทุกๆ 1 บิต ค่าที่เป็นไปได้ของคีย์จะเพิ่มขึ้นเป็นสองเท่าตัว หรือจำนวนคีย์ที่เป็นไปได้จะเพิ่มขึ้นเป็น 2 เท่าตัวนั่นเอง

ฉะนั้นจะเห็นได้ว่าคีย์ยิ่งมีความยาวมาก โอกาสที่ผู้บุกรุกจะสามารถคาดเดาคีย์ที่ตรงกับหมายเลขที่ถูกต้องของคีย์จะยิ่งยากมากขึ้นตามลำดับ ในการที่ผู้บุกรุกลองผิดลองถูกกับคีย์โดยใช้คีย์ที่มีหมายเลขต่างๆ กัน เพื่อหวังที่จะพบคีย์ที่ถูกต้องและสามารถใช้ถอดรหัสข้อมูลได้ การลองผิดลองถูกนี้เราเรียกกันว่าคีย์Search หรือการค้นหาคีย์นั่นเอง ทฤษฎีได้กล่าวไว้ว่าการลองผิดลองถูกนี้โดยเฉลี่ยจะต้องทดลองกับคีย์เป็นจำนวนครึ่งหนึ่งของคีย์ทั้งหมดก่อนที่จะพบคีย์ที่ถูกต้อง

ความยาวของคีย์ที่มีขนาดเหมาะสมจึงขึ้นอยู่กับความเร็วในการค้นหาคีย์ของผู้บุกรุกและระยะเวลาที่ต้องการให้ข้อมูลมีความปลอดภัย ตัวอย่างเช่น ถ้าผู้บุกรุกสามารถลองผิดลองถูกกับคีย์เป็นจำนวน 10 คีย์ภายในหนึ่งวินาทีแล้ว คีย์ที่มีความยาว 40 บิต จะสามารถป้องกันข้อมูลไว้ได้ 3,484 ปี ถ้าผู้บุกรุกสามารถลองได้เป็นจำนวน 1 ล้านคีย์ในหนึ่งวินาที (เทคโนโลยีปัจจุบันสามารถทำได้) คีย์ที่มีความยาว 40 บิตจะสามารถป้องกันข้อมูลไว้ได้เพียง 13 วันเท่านั้น (ซึ่งอาจไม่เพียงพอสำหรับในบางลักษณะงาน) ซึ่งถ้าผู้บุกรุกสามารถทดลองได้เป็นจำนวน 1,000 ล้านคีย์ในหนึ่งวินาที คีย์ขนาด 128 บิตจะสามารถป้องกันข้อมูลไว้ได้ 1022 ปี ซึ่งพอเพียงต่อการรักษาความลับของข้อมูลเอาไว้ได้

2.1.5 การสร้างหมายเลขแบบสุ่ม (Random Number Generators)

การสร้างตัวเลขแบบสุ่มนั้น เป็นหัวข้อที่พูดถึงกันน้อยในเรื่องของการเข้ารหัส แต่จริงๆ แล้วมีความสำคัญไม่ต่างจากหัวข้ออื่นๆเลย โดยโปรโตคอลที่เกี่ยวกับการเข้ารหัสส่วนใหญ่จำเป็นจะต้องใช้การสร้างตัวเลขแบบสุ่มทั้งนั้น และความปลอดภัยของโปรโตคอลนั้นๆ ก็ขึ้นอยู่กับความสุ่มของการสร้างตัวเลขแบบสุ่มนั้น

ถ้าการสร้างตัวเลขแบบสุ่มนั้นไม่ปลอดภัย (คือ ตัวเลขที่ได้ ไม่ใช่ตัวเลขแบบสุ่มอย่างแท้จริง) ไม่ว่าจะออกแบบโปรโตคอลในการเข้ารหัสดีแค่ไหนก็ตาม ก็ไม่ใช่เรื่องยากที่จะ break โปรโตคอลนั้นๆ

2.1.6 One-way Hash Function

หลักการของฟังก์ชันทางเดียว (One-way) คือ ง่ายต่อการคำนวณหรือประมวลผล แต่ยากต่อการทำกระบวนการย้อนกลับ นั้นก็คือ ถ้าเราให้ค่า x มา สามารถหาค่า f(x) ได้อย่างง่ายดาย แต่ถ้าให้ f(x) มา ยากมากที่จะหาค่า x ได้ โดยฟังก์ชันนี้จะรับอินพุตที่มีขนาดหลากหลายและแปลงข้อมูลออกมาเป็นเอาท์พุตที่มีขนาดคงที่ ซึ่งโดยทั่วไปจะมีขนาดเล็กกว่าขนาดของข้อมูลเริ่มต้น ดังนั้น ฟังก์ชัน One-way hash จึงเป็นฟังก์ชัน Hash ที่ทำงานเพียงทิศทางเดียว ซึ่งในอีกมุมมองหนึ่งก็เหมือนกับ ลายนิ้วมือ คือ ข้อมูลเพียงเล็กน้อย สามารถเป็นตัวแทนของข้อมูลขนาดใหญ่ๆ ได้ ซึ่งความปลอดภัยของมันก็คือ การที่มันเป็นฟังก์ชันทางเดียวนั้นเอง ฟังก์ชัน One-way Hash นี้ใช้ในหลายๆแอพพลิเคชันที่ต้องการความปลอดภัย รวมไปถึง การลงลายมือชื่ออิเล็กทรอนิกส์ด้วย

2.1.7 Hash Chain

แนวความคิดเรื่อง Hash Chain นั้น นำเสนอครั้งแรงโดย Lamport ในปี 1981 และใช้เพื่อป้องกันการลักลอบดักฟังรหัสผ่าน แต่ด้วยความเรียบง่ายและเป็นเทคนิคแบบ low-cost ที่อเนกประสงค์ จึงทำให้มีหลายๆแอพพลิเคชันนำไปใช้กันอย่างกว้างขวาง

Hash Chain ที่มีความยาว N หมายถึงการใช้ One-way Hash Function เป็นจำนวน N ครั้งบนข้อความตั้งต้น

$$K^N = h^N(x) = h(h(h(..h(x)...))) \qquad \text{(N times)}$$

โดย K^N ที่ได้นั้นมีลักษณะคล้ายๆกับ คีย์สาธารณะในการเข้ารหัสแบบ Asymmetric คือ เราสามารถประกาศ K^N ไปได้และต้องเก็บค่า x เอาไว้เป็นความลับ นั้นก็คือถ้าให้ K^N มาจะไม่

สามารถหาค่า K^{N-1} ได้โดยที่ไม่ทราบค่า x นั้นเอง ซึ่ง Hash Chain นั้นใช้คุณสมบัติของฟังก์ชัน One-way Hash โดยตรง

2.1.8 เมสเซสไดเจสต์ (Message Digest)

เมสเซสไดเจสต์ (Message Digest) หรือ เรียกสั้นว่า MD คือ ข้อความสรุปจากเนื้อหาข้อความเริ่มต้น โดยปกติข้อความสรุปจะมีความยาวน้อยกว่า ความยาวของข้อความเริ่มต้นมาก จุดประสงค์สำคัญของ MD ก็คือ สร้างข้อความสรุปที่สามารถใช้เป็นตัวแทนของข้อความเริ่มต้นได้ ซึ่งโดยทั่วไป MD จะใช้ฟังก์ชัน One-way Hash ในการสร้าง มีความยาวอยู่ระหว่าง 128 ถึง 256 บิต และจะไม่ขึ้นกับขนาดความยาวของข้อความเริ่มต้น

คุณสมบัติที่สำคัญของอัลกอริทึมสำหรับสร้างไดเจสต์มีดังนี้

- ทุกๆ บิตของ Message Digest จะขึ้นอยู่กับทุกบิตของข้อความเริ่มต้น
- ถ้าบิตใดบิตหนึ่งของข้อความเริ่มต้นเกิดการเปลี่ยนแปลง เช่น ถูกแก้ไข ทุกๆ บิตของ MD จะมีโอกาสร้อยละ 50 ที่จะแปรเปลี่ยนค่าไปด้วย ซึ่งหมายถึงว่า 0 เปลี่ยนค่าเป็น 1 และ 1 เปลี่ยนเป็น 0
- โอกาสที่ข้อความตั้งต้น 2 ข้อความใดๆ ที่มีความแตกต่างกัน จะสามารถคำนวณได้ค่า Message Digest เดียวกันมีโอกาสน้อยมาก

จากคุณสมบัติที่กล่าวมาสามารถอธิบายได้ว่าการเปลี่ยนแปลงแก้ไขข้อความเริ่มต้นโดยผู้ไม่ประสงค์ดีแม้ว่าอาจแก้ไขเพียงเล็กน้อยก็ตาม เช่น เพียง 1 บิตเท่านั้น ก็จะส่งผลให้ผู้รับข้อความทราบว่าข้อความที่ตนได้รับไม่ใช่ข้อความเริ่มต้น (โดยการนำข้อความที่ตนได้รับเข้าอัลกอริทึมเพื่อทำการคำนวณหา MD ออกมา แล้วจึงเปรียบเทียบ MD ที่คำนวณได้กับ MD ที่ส่งมาให้ด้วย ถ้าต่างกัน แสดงว่าข้อความที่ได้รับนั้นถูกเปลี่ยนแปลงแก้ไข)

อย่างไรก็ตามในทางทฤษฎีแล้ว มีโอกาสที่ข้อความ 2 ข้อความที่แตกต่างกันจะสามารถคำนวณแล้วได้ค่า MD เดียวกัน ปัญหานี้เรียกกันว่าการชนกันของ MD (Collision) อัลกอริทึมสำหรับสร้าง MD ที่ดีควรจะมีโอกาสน้อยมากๆ ที่จะก่อให้เกิดปัญหาการชนกันของ MD

อัลกอริทึมสำหรับสร้าง Message Digest ยอดนิยมมีดังนี้

อัลกอริทึม MD2

ผู้พัฒนาคือ Ronald Rivest อัลกอริทึมนี้เชื่อกันว่ามีความแข็งแกร่งที่สุดในบรรดาอัลกอริทึมต่างๆ ที่ Rivest พัฒนาขึ้นมา (ความแข็งแกร่งพิจารณาได้จากคุณสมบัติสามประการ

ข้างต้น) ข้อเสียของอัลกอริทึมนี้คือใช้เวลามากในการคำนวณ Message Digest หนึ่งๆ MD2 จึงไม่ค่อยได้มีการใช้งานกันมากนัก MD2 สร้าง Message Digest ที่มีความยาว 128 บิต

อัลกอริทึม MD4

ผู้พัฒนาคือ Rivest เช่นเดียวกับ MD2 อัลกอริทึมนี้พัฒนาขึ้นมาเพื่อแก้ปัญหาความล่าช้าในการคำนวณของ MD2 อย่างไรก็ตามในภายหลังได้พบว่าอัลกอริทึมมีข้อบกพร่องที่เกี่ยวข้องกับคุณสมบัติข้อที่สามโดยตรง กล่าวคือปัญหาการชนกันของ Message Digest มีโอกาสเกิดขึ้นได้ไม่น้อย ซึ่งผู้บุกรุกอาจใช้ประโยชน์จากจุดอ่อนนี้เพื่อทำการแก้ไขข้อความตั้งต้นที่ส่งมาให้ได้ MD4 ผลิตไดเจสต์ที่มีขนาด 128 บิต

อัลกอริทึม MD5

Rivest เป็นผู้พัฒนาเช่นกันโดยพัฒนาต่อจาก MD4 เพื่อให้มีความปลอดภัยที่สูงขึ้น ถึงแม้จะเป็นที่นิยมใช้งานกันอย่างแพร่หลาย ทว่าในปี 1996 ก็มีผู้พบจุดบกพร่องของ MD5 (เช่นเดียวกับ MD4) จึงทำให้ความนิยมเริ่มลดลง MD5 ผลิตไดเจสต์ที่มีขนาด 128 บิต

อัลกอริทึม SHA

SHA ย่อจาก Secure Hash Algorithm อัลกอริทึม SHA ได้รับแนวคิดในการพัฒนามาจาก MD4 และได้รับการพัฒนาขึ้นมาเพื่อใช้งานร่วมกับอัลกอริทึม DSS (ซึ่งใช้ในการลงลายมือชื่ออิเล็กทรอนิกส์) หลังจากที่ได้มีการตีพิมพ์เผยแพร่อัลกอริทึมนี้ได้ไม่นาน NIST ก็ประกาศตามมาว่าอัลกอริทึมจำเป็นต้องได้รับการแก้ไขเพิ่มเติมเล็กน้อยเพื่อให้สามารถใช้งานได้อย่างเหมาะสม SHA สร้างไดเจสต์ที่มีขนาด 160 บิต

อัลกอริทึม SHA-1

SHA-1 เป็นอัลกอริทึมที่แก้ไขเพิ่มเติมเล็กน้อยจาก SHA การแก้ไขเพิ่มเติมนี้เป็นที่เชื่อกันว่าทำให้อัลกอริทึม SHA-1 มีความปลอดภัยที่สูงขึ้น SHA-1 สร้างไดเจสต์ที่มีขนาด 160 บิต

อัลกอริทึม SHA-256, SHA-384 และ SHA-512

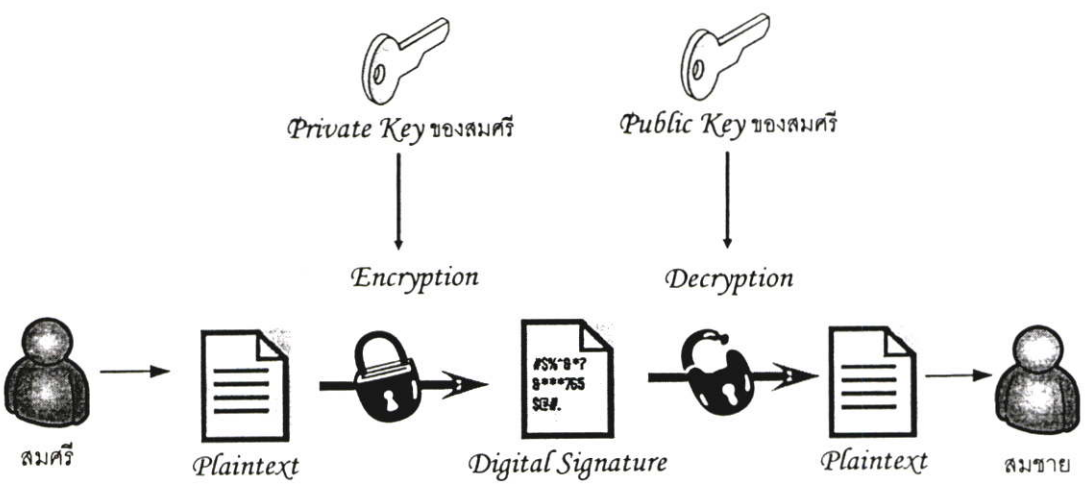
NIST เป็นผู้นำเสนออัลกอริทึมทั้งสามนี้ในปี 2001 เพื่อใช้งานร่วมกับอัลกอริทึม AES (ซึ่งเป็นอัลกอริทึมในการเข้ารหัสแบบสมมาตร) อัลกอริทึมเหล่านี้สร้างไดเจสต์ที่มีขนาด 256, 384 และ 512 บิต ตามลำดับ

2.1.9 การลงลายเซ็นดิจิตอล (Digital Signature)

ลายเซ็นดิจิตอล (ซึ่งเปรียบเสมือนการลงลายมือชื่อของเราที่ใช้กับเอกสารสำนักงานทั่วไป) การลงลายมือชื่อนี้จะเป็นการพิสูจน์ความเป็นเจ้าของและสามารถใช้ได้กับการทำธุรกรรมต่างๆ บนอินเทอร์เน็ท เช่น การซื้อสินค้า เป็นต้น สามารถตรวจสอบได้ว่า ผู้เซ็นคนไหนเป็นคนเซ็นลงบนเอกสารนั้น และ สามารถตรวจสอบได้อีกว่า เอกสารนั้นเหมือนกับเอกสารตอนที่ผู้เซ็นได้เซ็นไว้หรือไม่ โดยลายเซ็นดิจิตอล นั้นรองรับ

- Authentication
- Data Integrity
- Non Repudiation Service

โดยการเข้ารหัสแบบอสมมาตร สามารถประยุกต์ใช้ได้กับการลงลายลายเซ็นดิจิตอลได้ เนื่องจากข้อความที่เข้ารหัสโดย คีย์ส่วนตัวจะต้องถอดได้โดย คีย์สาธารณะที่เป็นคู่ของมันเท่านั้น ดังนั้นจึงใช้ในการยืนยันต้นทางว่ามาจากบุคคลนั้นจริงๆได้ เพราะหากคนนั้นไม่ได้เข้ารหัสด้วยคีย์ส่วนตัวมาแล้ว ก็จะใช้ คีย์สาธารณะของคนนั้นถอดรหัสข้อความออกมาไม่ได้ ซึ่งในที่นี้เราจะเรียกว่าการ Signs



รูปที่ 2.3 Digital Signature Algorithm

วิธีการใช้งาน ยกตัวอย่างเช่น สมศรี ต้องการส่งข้อความหา สมชาย โดยที่ต้องการยืนยันว่า ข้อความที่สมศรีส่งไปนั้น มาจากสมศรีจริงๆ สามารถทำได้โดย สมศรี ต้องทำการ Signs ข้อความที่ต้องการจะส่งไปยังสมชายด้วย คีย์ส่วนตัวของสมศรีแล้วจึงส่งข้อความที่ได้ทำการ Signs แล้วไปยังสมชาย เมื่อสมชายได้รับข้อความก็จะสามาระใช้ คีย์สาธารณะของสมศรี เพื่อ

ตรวจสอบว่าเป็นข้อความที่มาจากสมศรีหรือไม่ (เพราะสมศรีเป็นคนเดียวที่มี คีย์ส่วนตัวของสมศรีอยู่)

ในทางปฏิบัตนั้น นิยมใช้ MD เป็นส่วนหนึ่งในการลงลายมือชื่ออิเล็กทรอนิกส์ กล่าวคือ ใช้การลงลายมือชื่อกับ MD ของข้อความตั้งต้นแทนการลงลายมือชื่อกับข้อความตั้งต้นทั้งข้อความ

2.2 หลักการพื้นฐานของ Positioning System

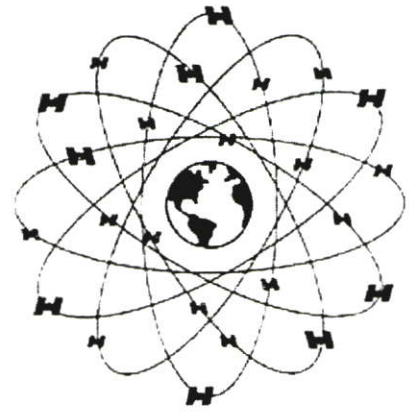
Positioning System (PS) คือ ระบบที่สามารถบอกตำแหน่งที่ตั้ง ณ ปัจจุบันบนพื้นผิวโลกได้ เพื่อนำตำแหน่งที่ตั้ง และ ข้อมูลบางอย่างไปใช้ในแอพพลิเคชันต่างๆ ได้ เช่น Location based service, GIS เป็นต้น ซึ่งนี้ที่นี้จะยกตัวอย่างและอธิบายหลักการพื้นฐานดังนี้

2.2.1 Global Positioning System

GPS (Global Positioning System) เป็นระบบหาพิกัดบนพื้นโลก โดยการอ้างอิงจากดาวเทียมที่มีความแม่นยำสูง สามารถใช้หาพิกัดใดๆบนพื้นผิวโลก ได้ทุกเวลา และ ทุกสภาพอากาศ ระบบนี้มีดาวเทียว 24 ดวง หมุนรอบโลก อยู่สูงขึ้นไป 11,000 nautical miles หรือประมาณ 20,200 กิโลเมตร ดาวเทียมหมุนรอบโลกแบ่งออกเป็น 6 ระนาบ ระนาบละ 4 ดวง โดยทำมุมเอียง 55 องศา ดาวเทียมทั้งหมดจะได้รับการควบคุมดูแล จากสถานีภาคพื้นดินทั่วโลกตลอดเวลา

ระบบที่ทำให้ GPS ทำงานได้ สามารถแบ่งได้ออกเป็น 3 ส่วนหลัก คือ ส่วนอวกาศ (Space Segment), ส่วนควบคุม (Control Segment) และส่วนผู้ใช้งาน

SPACE SEGMENT ส่วนของอวกาศ ประกอบไปด้วยเครือข่ายของดาวเทียม ระบบ GPS ทั้งระบบ ประกอบด้วย ดาวเทียม 24 ดวง โคจรรอบโลก ที่ระยะ 11,000 ไมล์อากาศ จากพื้นโลก ใช้เวลา 12 ช.ม ในการโคจรรอบโลกหนึ่งรอบ ดาวเทียมโคจรรอบโลก แบ่งเป็น 6 ระนาบ และ ทำมุมเอียง 55 องศา การวางวงโคจรเช่นนี้ทำให้ เราสามารถรับสัญญาณจากดาวเทียม ได้คราวละถึง 6 ดวง ดาวเทียมติดตั้งนาฬิกาที่เที่ยงตรงมากๆ ถึง 3 nanoseconds (ความเที่ยงตรง 0.000000003 ของวินาที หรือ 3e-9)



รูปที่ 2.4 ดาวเทียม 24 ดวง โคจรรอบโลก แบ่งเป็น 6 ระนาบ

ความเที่ยงตรงมีความสำคัญมาก สำหรับเครื่องรับ เพราะเครื่องรับจำเป็นต้องทราบเวลาที่เที่ยงตรง แน่นอน ว่าระยะเวลาเท่าไร ที่สัญญาณคลื่นจากดาวเทียมเดินทางถึงเครื่องรับ ดาวเทียมแต่ละดวง มีเชื้อเพลิง และเครื่องยนต์ขนาดเล็ก ซึ่งสามารถที่จะปรับแต่งดาวเทียม ให้อยู่ในตำแหน่งที่ถูกต้องในวงโคจร ถ้าดาวเทียมเกิดเคลื่อนออกจากตำแหน่งที่กำหนด ดาวเทียม และในแต่ละดวงมีนาฬิกา Atomic Clocks 4 อัน เป็นนาฬิกานี้มีความเที่ยงตรงถึงหนึ่งในหนึ่งพันล้านของวินาที หรือ Nanosecond โดยดาวเทียมแต่ละดวง จะส่งคลื่นสัญญาณออกมาสองคลื่นสัญญาณ หนึ่งคลื่นสำหรับการทหาร และอีกคลื่นหนึ่งสำหรับพลเรือน

คุณลักษณะ บางอย่างของดาวเทียม

- น้ำหนัก 930 kg (in orbit)
- ขนาดSize 5.1 m
- ความเร็ว ใน การโคจร 4 km/sec
- สัญญาณที่ส่ง 1575.42 MHz and 1227.60 MHz
- เครื่องรับ สัญญาณ 1783.74 MHz
- นาฬิกา 2 Cesium and 2 Rubidium
- อายุ การ ใช้งาน 7.5 year (later model BlockIIR 10 years)

CONTROL SEGMENT ส่วนควบคุมดาวเทียมประกอบด้วย:

- **Master Control Station:** สถานีควบคุมแม่ข่ายมีอยู่ 1 สถานี ทำหน้าที่รับผิดชอบในการจัดการทั่วไป และบริการสถานีลูกข่าย เป็นศูนย์กลางที่ให้การสนันสนุนการทำงานเครื่องแม่ข่ายจะคำนวนตำแหน่ง และ นาฬิกา ดูความคลาดเคลื่อนของดาวเทียมแต่ละดวงจากสถานีลูกข่ายภาคพื้น และสั่งคำสั่งแก้ไขกลับไปยังสถานีลูกข่าย เพื่อส่งไปยังดาวเทียมดวงนั้นๆ

- **Monitor Stations:** สถานีควบคุมลูกข่ายมีอยู่ 4 สถานี จะทำการตรวจสอบ ความสูง , ตำแหน่ง , ความเร็ว, และวงจร ทั่วไปของดาวเทียม สถานีควบคุมนี้ตรวจสอบดาวเทียมได้ครั้งละ 11 ดวง การตรวจสอบนี้ แต่ละสถานีกระทำวันละ 2 ครั้ง เมื่อดาวเทียมโคจรรอบโลก

Global Positioning System (GPS) Master Control and Monitor Station Network

รูปที่ 2.5 ตำแหน่งที่ตั้งของ Master Control Station และ Monitor Station บนพื้นผิวโลก

USER SEGMENT ส่วนผู้ใช้งานประกอบด้วยเครื่องรับสัญญาณ หรือเครื่อง GPS แบบมือถือที่มีใช้กันอยู่ทั่วไปนั่นเอง โดยในเครื่อง GPS นั้นจะมีโปรแกรมคอมพิวเตอร์อยู่ในตัวเครื่องเพื่อให้เครื่องทราบว่าดาวเทียมอยู่ในตำแหน่งใด ในเวลานั้น ๆ โดยเครื่อง GPS จะทำการคำนวณตรวจสอบ และถอดรหัสสัญญาณที่ได้จากดาวเทียม เพื่อให้ได้ข้อมูลมา ซึ่งข้อมูลที่ได้โดยปรกติก็มักจะถูกประมวลผลโดยโปรแกรมและส่งข้อมูลออกมาทางหน้าจอของเครื่อง GPS นั้น ๆ เพื่อให้ผู้ใช้ได้ทราบข้อมูล โดยการแสดงผลก็จะต่างกันขึ้นกับโปรแกรมในเครื่อง GPS แต่ละรุ่นและแต่ละยี่ห้อ จะเห็นได้ว่าเบื้องหลังการใช้งานเครื่อง GPS นั้น มีส่วนประกอบที่สำคัญอื่น ๆ ที่ทำให้เราสามารถใช้งานเครื่อง GPS ได้ ซึ่งในส่วนผู้ใช้งานเองแค่มีเพียง GPS Receiver เครื่องเดียวก็เพียงพอแล้ว โดยในส่วนอื่น ๆ นั้นก็จะมีหน่วยงานที่เกี่ยวข้องคอยดูแล เพื่อให้ระบบนั้นสามารถทำงานได้อย่างมีประสิทธิภาพ

หลักการของเครื่อง GPS คือการคำนวณระยะทางระหว่างดาวเทียมกับเครื่อง GPS ซึ่งจะต้องใช้ระยะทางจากดาวเทียมอย่างต่ำ 3 ดวง เพื่อให้ได้ตำแหน่งที่แน่นอน ซึ่งเมื่อเครื่อง GPS สามารถรับสัญญาณจากดาวเทียมได้ 3 ดวงขึ้นไปแล้ว (ถ้ารับสัญญาณได้ 3 ดวง จะสามารถคำนวณหาตำแหน่งได้ แต่ถ้าได้ 4 ด้วยขึ้นไป สามารถตำแหน่งในลักษณะ 3 มิต หรือ ความสูงได้) จะมีคำนวณระยะทางระหว่างดาวเทียมถึงเครื่อง GPS โดยจากสูตรคำนวณทางฟิสิกส์

2.2.2 Hutch Navi

Hutch Navi คือ รูปแบบใหม่ของการให้บริการค้นหาตำแหน่งและเส้นทาง ซึ่งได้นำเอาความสามารถของเทคโนโลยีที่ทันสมัยผสานเข้ากับแอพพลิเคชั่นอันหลากหลาย ด้วยการนำเทคโนโลยี AGPS (Assist Global Positioning System) มาใช้ในการพัฒนา ทำให้ Navi มีความโดดเด่นในเรื่องของความแม่นยำ เนื่องจากเป็นการใช้ความสามารถของเครือข่าย CDMA 2000 1X ทำงานร่วมกับระบบการระบุพิกัดดาวเทียม GPS (Global Positioning System) ทำให้ Navi สามารถระบุตำแหน่งได้ด้วยแม่นยำสูงถึงในระดับต่ำกว่า 10 เมตร และยังสามารถหาตำแหน่งได้แม้อยู่ในอาคาร (ระดับความละเอียดในการหาตำแหน่งจะขึ้นอยู่กับสภาพแวดล้อมของผู้ใช้บริการขณะนั้น)

จุดเด่นอีกประการหนึ่งของ Navi คือ ระบบการป้องกันความเป็นส่วนตัวของผู้ใช้บริการ (Privacy control) คุณมีสิทธ์เลือกที่จะอนุญาตหรือไม่อนุญาตให้ใครทราบตำแหน่งของคุณ สามารถกำหนดได้กระทั่งวันและเวลาที่ต้องการ ตลอดจนระดับความแม่นยำที่ให้ทราบตำแหน่งได้ เพื่อป้องกันการถูกค้นหาตำแหน่งจากบุคคลที่ไม่พึงประสงค์ เพราะความล้ำหน้าควรจะมาพร้อมๆ กับความปลอดภัย

ยิ่งไปกว่านั้นยังเพิ่มคุณสมบัติของความเป็น Personalize ที่ให้คุณสามารถบันทึกตำแหน่งและเส้นทางที่ใช้เป็นประจำได้ ตื่นตาด้วยการแสดงผลลัพธ์เป็นแผนที่กราฟฟิกที่นอกจากแสดงตำแหน่งหรือเส้นทางที่ค้นหาแล้วยังแสดงจุดสังเกตสำหรับการเดินทางด้วย อีกทั้งยังเพิ่มลูกเล่นของการย่อ (zoom-in), ขยาย (zoom-out) รวมทั้งปรับหมุนแผนที่ได้ถึง 360 องศา ซึ่งทำได้ง่ายๆ และรวดเร็ว

เทคโนโลยี AGPS (Assist Global Positioning System)

เทคโนโลยีในการค้นหาตำแหน่งของโทรศัพท์เคลื่อนที่สามารถแบ่งได้เป็นประเภทใหญ่ๆได้ดังนี้

- เทคโนโลยีการระบุตำแหน่งด้วยดาวเทียม GPS ในช่วงแรก เทคโนโลยี GPS เป็นเทคโนโลยีที่ใช้ในวงการทหารของสหรัฐอเมริกา โดยใช้การคำนวณระยะทางโดยเทียบกับดาวเทียม GPS 24 ดวงที่โคจรรอบโลกทุกๆ 12 ชั่วโมง ซึ่งกองทัพสหรัฐฯ ใช้ GPS เพื่อค้นหาตำแหน่งพิกัดของทหารที่อยู่ในพื้นที่ต่างๆรวมถึงในทะเล ต่อมาได้มีการนำเทคโนโลยีนี้มาใช้ในเชิงพาณิชย์และการใช้งานสำหรับบุคคลทั่วไป เช่น ระบบนำทางรถยนต์ (Car Navigation) ข้อดีของเทคโนโลยี GPS คือ ความแม่นยำในการระบุตำแหน่งซึ่งสามารถทำได้ในระดับ 20 – 100 เมตร แต่ก็มีข้อจำกัดบางเรื่อง เช่น ต้องใช้เวลานานในการค้นหาตำแหน่งแต่ละครั้ง อีกทั้งยังต้องไม่มีสิ่งบดบังระหว่างดาวเทียมกับเครื่องรับ

ด้วย ซึ่งด้วยสาเหตุดังกล่าวทำให้เกิดปัญหาเมื่อมีการใช้งานในอาคาร หรือในบริเวณที่มีตึกสูง หรือใต้ทางด่วน

- เทคโนโลยีระบุตำแหน่งด้วยเครือข่ายโทรศัพท์เคลื่อนที่ (Network based) วิธีนี้เป็นการค้นหาตำแหน่งของเครื่องรับโทรศัพท์เคลื่อนที่โดยใช้ตำแหน่งที่ตั้งของสถานีฐานของระบบเครือข่ายที่โทรศัพท์เคลื่อนที่นั้นอยู่ ณ ขณะนั้น แทนตำแหน่งของเครื่องโทรศัพท์เคลื่อนที่ วิธีนี้จะมีความคลาดเคลื่อนในการค้นหาค่อนข้างสูงเมื่อเทียบกับระบบ GPS ซึ่งความละเอียดที่ได้จะอยู่ในระดับ หลายร้อยเมตรถึงหลายกิโลเมตร ขึ้นอยู่กับขนาดของสถานีฐาน อย่างไรก็ดี ได้มีการนำเอาระบบอื่นๆมาช่วยในการเพิ่มความแม่นยำของเทคโนโลยีนี้ เช่น การวัดความแตกต่างของเวลา ซึ่งช่วยให้หาตำแหน่งได้แม่นยำขึ้น แต่อย่างไรก็ดีระดับความแม่นยำที่ได้ก็ยังแตกต่างจากความแม่นยำของระบบ GPS มาก ข้อดีของเทคโนโลยีนี้คือความรวดเร็วในการค้นหาตำแหน่งและเครื่องรับโทรศัพท์มีความซับซ้อนน้อย

สำหรับเทคโนโลยี AGPS ที่ใช้ในบริการ Navi เป็นการนำความสามารถของเครือข่าย CDMA และเครื่องเซิร์ฟเวอร์ประสิทธิภาพสูงมาช่วยในการทำงานของระบบ GPS โดยวิธีนี้จะทำให้ระบบGPSมีความแม่นยำสูงขึ้น อีกทั้งยังสามารถคำนวณตำแหน่งได้รวดเร็วขึ้น ตลอดจนสามารถทำงานได้แม้อยู่ภายในอาคาร

AGPS Terminal

เทอร์มินอลของ Hutch รุ่น Kyocera 850 และ Kyocera 830 ที่รองรับบริการ Navi ได้ถูกออกแบบมาเพื่อให้ทำงานกับระบบ AGPS ของ Hutch โดยมีเสาอากาศที่สามารถรับได้ทั้งสัญญาณโทรศัพท์ CDMA และสัญญาณจากดาวเทียม GPS ภายในยังบรรจุวงจรและโปรแกรมที่เกี่ยวข้องกับการทำงานของ AGPS ไว้ โดยที่ตัวเครื่องรับโทรศัพท์ยังคงมีขนาดเล็กเมื่อเทียบกับอุปกรณ์ GPS ทั่วไป ทำให้ผู้ใช้สามารถพกพาไปทุกที่ได้สะดวก นอกจากเรื่องการระบุตำแหน่งแล้ว Hutch ได้ใส่โปรแกรมพิเศษในเครื่องทั้งสองรุ่น ทำให้สามารถแสดงแผนที่ในแบบกราฟิกส์สี ซึ่งสามารถ ย่อ ขยาย เลื่อน และปรับหมุนแผนที่ไปตามทิศทางด้านหน้าของผู้ใช้ Naviคีย์ได้ถูกออกแบบมาเพื่อให้ผู้ใช้บริการสามารถเรียกใช้บริการ Navi ได้ง่าย เพียงกด Naviคีย์ผู้ใช้จะถูกนำเข้าสู่หน้าหลัก (Portal) ของ Navi

2.3 งานวิจัยที่เกี่ยวข้อง

ส่วนนี้จะเสนองานวิจัยที่คล้ายคลึงหรือเกี่ยวข้องกับงานวิจัยที่เราพัฒนา ซึ่งในวิทยานิพนธ์นี้สนใจงานวิจัยทางด้านลายเซ็นดิจิตอล, ความปลอดภัยสำหรับอุปกรณ์โมบายล์, Location based Service, Mobility Model

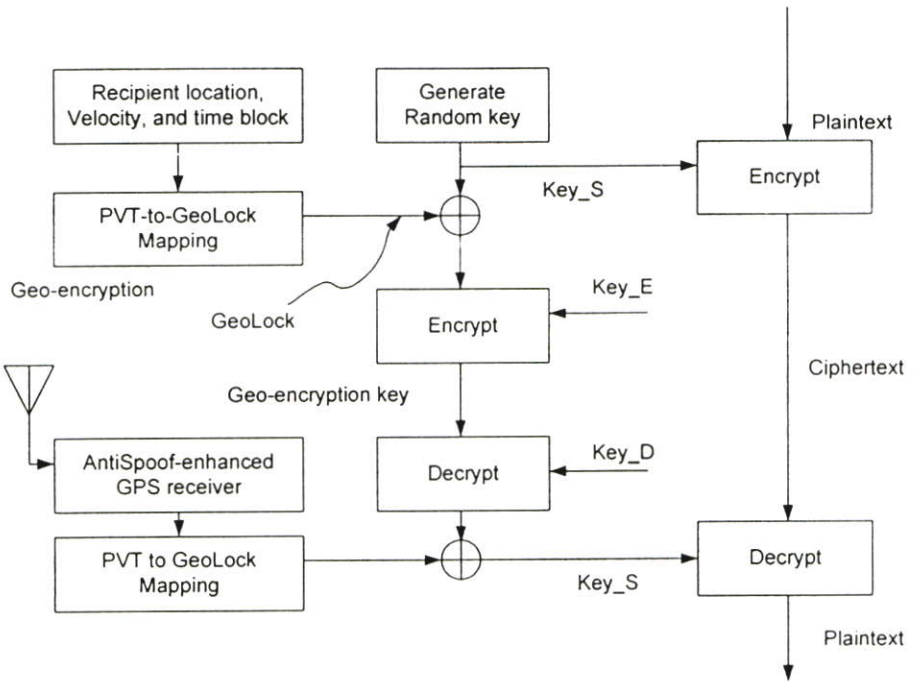
2.3.1 ลายเซ็นดิจิตอลและความปลอดภัยสำหรับอุปกรณ์โมบายล์

[1] เสนอ "Server-Supported Signature Scheme for mobile communication" ใช้ One-way Function กับวิธีในการสร้างลายเซ็นดิจิตอลแบบปกติ โดยมี Signature Servers ทำหน้าที่ในการสร้าง Digital Signature Tokens และ Certification Authority เพื่อตรวจสอบ Tokens เหล่านั้น ต่อมา [2] นำเสนอ "Server Aided Signature" ซึ่งตั้งอยู่บนพื้นฐานของ [1] โดยบทความนี้ ผู้ใช้จะมีส่วนร่วมในการสร้าง Signature Token และเพิ่มคุณสมบัติในส่วนของออนไลน์เข้ามา ซึ่งไม่เหมือนกับการสร้างลายเซ็นดิจิตอลแบบปกติ ซึ่งใช้คู่คีย์สาธารณะ/คีย์ส่วนตัวในการสร้าง ทั้ง [1], [2] ใช้ฟังก์ชัน One-way Hash ในการสร้างคีย์ลับแล้วใช้คีย์นั้นในการสร้าง Non-Repudiation Signature เพราะว่าการสร้างคู่ คีย์สาธารณะ/คีย์ส่วนตัวนั้นใช้การประมวลผลที่ละเอียดอ่อน และหนักเกินไปสำหรับอุปกรณ์โมบายล์ แต่ในวิธีการของทั้ง [1], [2] นั้นอุปกรณ์ Mobile ยังต้องการประมวลผลที่หนักในการตรวจสอบความถูกต้องของลายเซ็นดิจิตอลอยู่ดี

[3] เสนอ "Generating Digital Signatures on Mobile Devices" โดยใช้ Server ช่วยในการสร้าง Non-Repudiation Digital Signature และด้วยการนำวิธีการสร้างลายเซ็นดิจิตอลจาก [4] "RSA-Based Partially Blind Signature with Low Computation" มาใช้ทำให้สามารถหลีกเลี่ยงการคำนวณที่หนักๆ ทางฝั่งอุปกรณ์โมบายล์ ในส่วนของการสร้าง และตรวจสอบลายเซ็นดิจิตอลไปได้ และยังได้ความปลอดภัยระดับเดียวกับลายเซ็นดิจิตอลแบบปกติอีกด้วย ทำให้วิธีนี้มีประสิทธิภาพดีกว่า [1], [2]

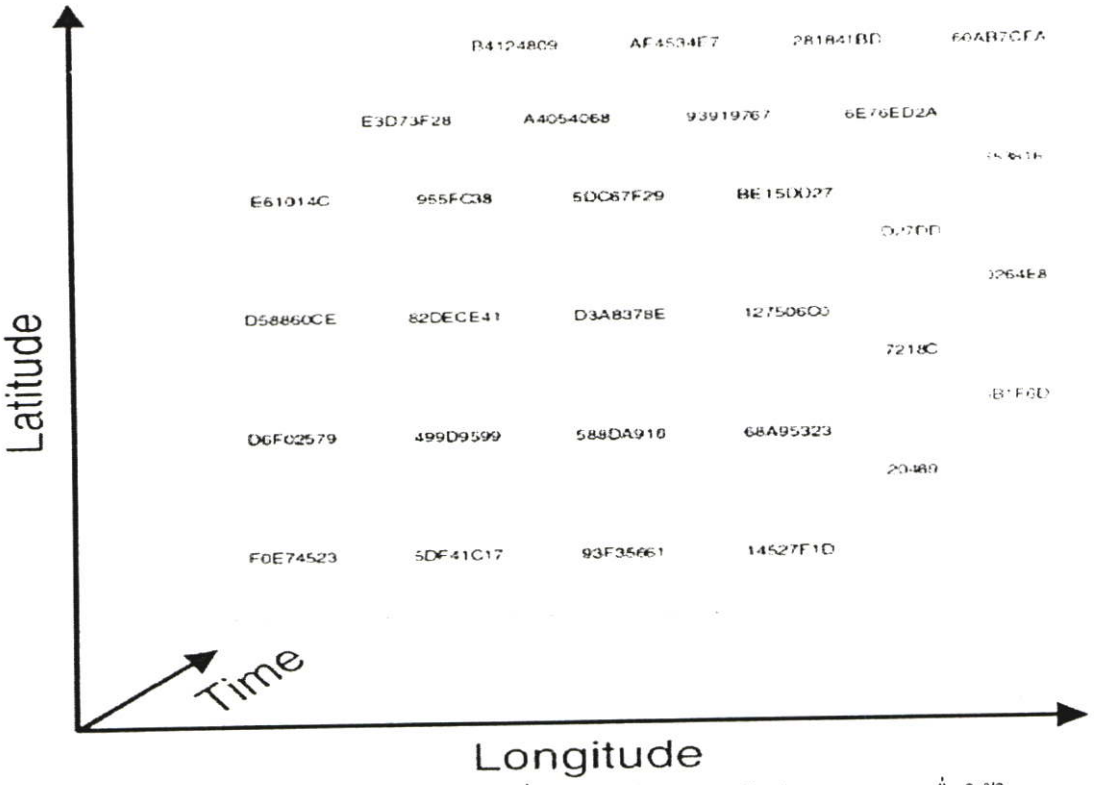
2.3.2 Location based Service และ Mobility Model

[4] เสนอ "Geo-Encryption" ด้วยความก้าวหน้าของอุปกรณ์โมบายล์ และเทคโนโลยี GPS จึงได้มีการนำประโยชน์ของ PS มาใช้ในการทำการเข้ารหัสที่เรียกว่า GEO-Encryption โดย D. E. Denning เป็นโมเดลที่การรวมเอาตำแหน่งที่ตั้ง และ เวลาเข้าไปในกระบวนการเข้ารหัสและถอดรหัส เพื่อเพิ่มระดับความปลอดภัย โดยเข้ารหัสข้อความเพื่อให้สามารถถอดรหัสข้อความได้ ถ้าผู้รับอยู่ในตำแหน่งที่กำหนด และ ภายในเวลาที่กำหนดเท่านั้น



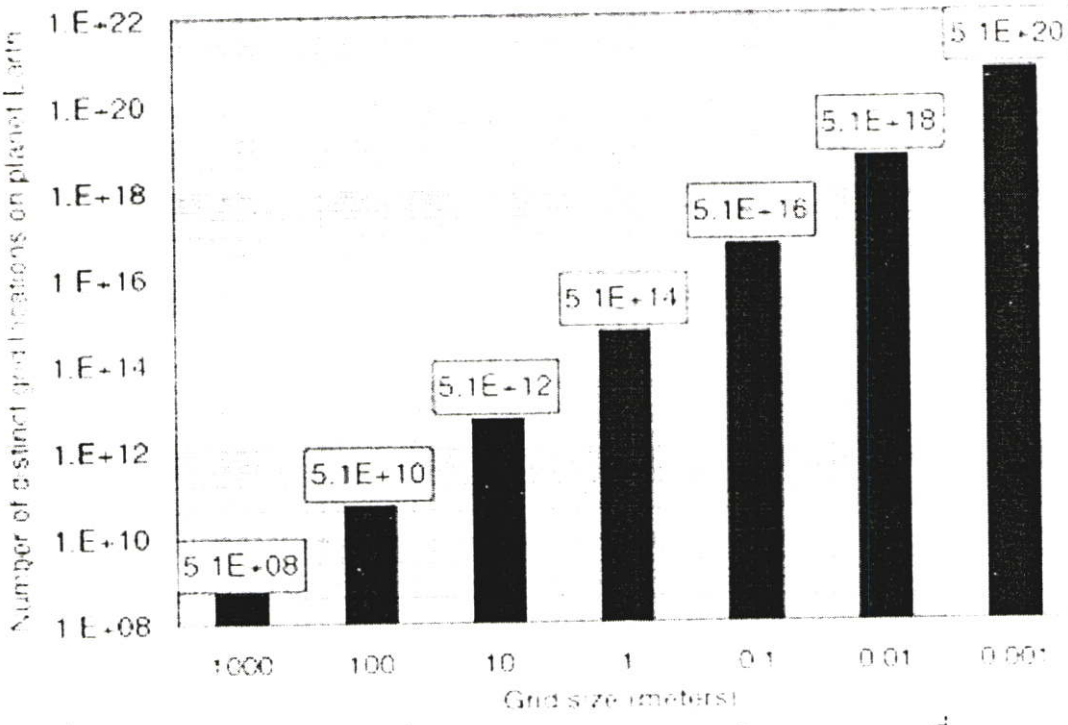
รูปที่ 2.6 อัลกอริทึมของ Geo-Encryption

โดยมีฟังก์ชัน Position-Velocity-Time (PVT) to GeoLock Mapping (ในที่นี้ขอเรียกสั้นๆ ว่า Mapping Function) รับอินพุทเป็น Latitude, Longitude และเวลา ทำหน้าที่ในการสร้างรหัสหรือค่า GeoLock ส่งไปพร้อมกับข้อความที่ผ่านการเข้ารหัสด้วย คีย์ Session โดยค่า GeoLock ใช้ในการสร้างคีย์ Geo-Secured จากทางฝั่งผู้ส่ง และใช้ถอดรหัสจากคีย์ Geo-Secured มาเป็น คีย์ Session ในฝั่งของผู้รับ โดยในที่นี้ Grid ของ Latitude, Longitude และเวลา จะถูกสร้างขึ้น และในแต่ละ Grid ก็จะมีค่า GeoLock ที่แตกต่างกัน



รูปที่ 2.7 PVT-to-GeoLock Mapping ทำหน้าที่ในการสร้างรหัสหรือค่า GeoLock เพื่อใช้ในการล็อค และปลดล็อค Session Key

ดังนั้นขนาดของ Grid นั้นก็ขึ้นอยู่กับความแม่นยำของเทคโนโลยี Positioning System ที่นำมาใช้งาน เช่น เทคโนโลยี Positioning System ที่นำมาใช้งานมีความมีความแม่นยำได้การบอกตำแหน่งสูง เราก็สามารถตั้งขนาดของ Grid ให้มีขนาดที่เล็กได้ หรือถ้าเทคโนโลยี Positioning System ที่นำมาใช้มีความคลาดเคลื่อนในการบอกตำแหน่งที่ค่อนข้างมาก เราก็ควรเลือกขนาดของ Grid ให้มีขนาดใหญ่ เพราะว่า การเลือกขนาดของ Grid ให้เหมาะสมนั้น ก็เพื่อลดความผิดพลาดในการสร้างค่า GeoLock เพื่อใช้ในการถอดรหัสทางฝั่งผู้รับนั้นเอง

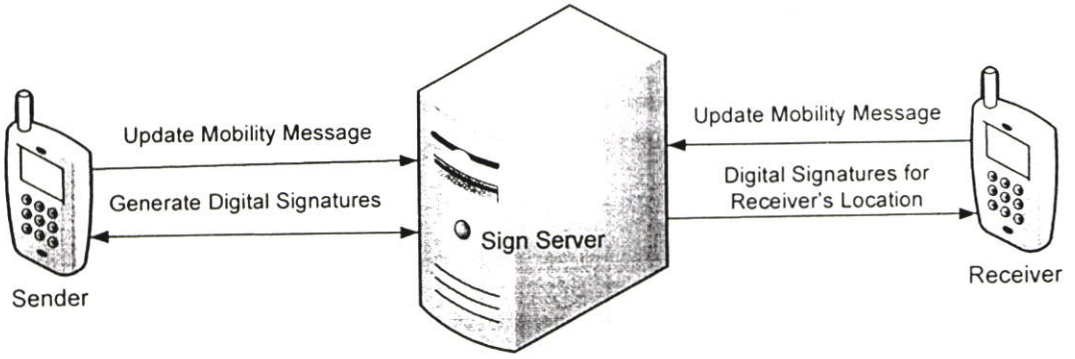


รูปที่ 2.8 กราฟแสดงความสัมพันธ์ระหว่าง ขนาดของ Grid และ จำนวนของ Grid ที่ครอบคลุมพื้นผิวโลก

[5] เสนอ "A Mobility Model for GPS-Based Encryption" เนื่องด้วยความจริงที่ว่า โมเดล Geo-Encryption [4] จะสามารถใช้งานอย่างมีประสิทธิภาพก็ต่อเมื่อ ผู้ส่งสามารถรู้ตำแหน่งที่ตั้งของผู้รับและเวลาที่ผู้รับจะไปอยู่ในตำแหน่งนั้นได้ ดังนั้นบทความนี้จึงนำเสนอ Mobility Model เพื่อให้สามารถใช้งานร่วมกับ Geo-Encryption ได้โดยให้อุปกรณ์โมบายล์แลกเปลี่ยนพารามิเตอร์ในการเคลื่อนที่ต่างๆกันได้ ทำให้ผู้ส่งสามารถทำ Geo-Encrypt ข้อความส่งไปยัง Decryption Zone ที่ประมาณว่าผู้รับจะอยู่ได้ และนำเสนอวิธีการคำนวณค่าพารามิเตอร์ในการเคลื่อนที่ต่างๆ ไว้อีกด้วย

บทที่ 3
งานวิจัยที่นำเสนอ

ในบทนี้จะกล่าวถึงวิธีที่นำเสนอ โดยจะอธิบายถึงวิธีในการสร้างลายเซ็นดิจิตอล, การนำปัจจัยทางภูมิศาสตร์ ตำแหน่งที่ตั้ง และเวลาเข้ามามีส่วนร่วมในการสร้าง ว่ามีหลักการทำงานอย่างไร โดยหลักการที่นำเสนอนี้ ตั้งอยู่บนพื้นฐานของ [3], [4] และ [5] ที่ว่าทั้งผู้ส่ง และ ผู้รับนั้นเป็นอุปกรณ์โมบายล์ และสามารถส่งตำแหน่งที่ตั้งของตนไปยังฝ่ายตรงข้ามได้อย่างปลอดภัยเมื่อจำเป็น โดยนำ Mobility Model ใน [5] มาประยุกต์ใช้กับแบบจำลองของเราเพื่ออัพเดทพารามิเตอร์ในการเคลื่อนที่ไปยัง Sign Server ซึ่งทำหน้าที่ช่วยสร้าง ลายเซ็นดิจิตอลนั้นเอง



รูปที่ 3.1 ภาพรวมของแบบจำลองที่นำเสนอโดยมี Sign Server ทำหน้าที่ช่วยในการสร้างลายเซ็นดิจิตอลและคอยรับ Mobility Message

3.1 การสร้างลายเซ็นดิจิตอล

ด้วยการนำวิธีการในการสร้างลายเซ็นดิจิตอล ใน [3] มาประยุกต์ใช้ในแบบจำลองของเราทำให้เราสามารถสร้าง ลายเซ็นดิจิตอลที่เหมาะสมสำหรับอุปกรณ์โมบายล์ ซึ่งมีระดับความสามารถในการประมวลผล และมีพลังงานที่จำกัด โดย ลายเซ็นดิจิตอล ที่ได้นั้นรองรับ Authentication, Data Integrity และ Non-Repudiation Services ได้ (ซึ่งในที่นี้สนใจที่ Non-Repudiation of Sender (NRS) และ Non-Repudiation of Receiver (NRR)) และมีระดับความปลอดภัยระดับเดียวกับการสร้างลายเซ็นดิจิตอล แบบปกติ โดยแบบจำลองของที่นำเสนอมี Sign Server ทำหน้าที่ในการช่วย อุปกรณ์โมบายล์สร้างลายเซ็นดิจิตอลอีกทั้งยังคอยรับข้อมูลอัพเดทสถานะการเคลื่อนที่ของ อุปกรณ์โมบายล์s เพื่อให้สามารถประมาณตำแหน่งของผู้รับในเวลา t ใด ๆ ได้

โดยมีการใช้ One-way Collision-resistant Hash Function เช่น SHA1 หรือ MD5 เป็นต้น เพื่อใช้ในการสร้าง K'' ของทั้งผู้ส่งและผู้รับ ซึ่งกำหนดสัญลักษณ์ให้ $h'()$ หมายถึง การนำ

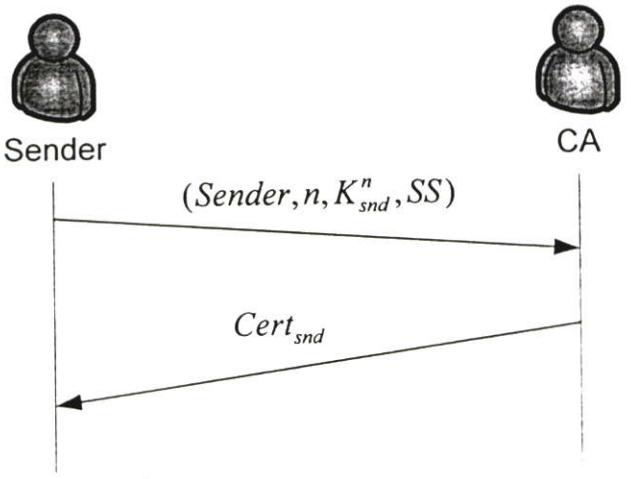
เอาท์พุตที่ได้มาทำการ Hash ซ้ำไปซ้ำมาเป็นจำนวนทั้งหมด i ครั้ง โดยผู้ใช้ทุกคนต้องสร้างค่า K_u ขึ้นมา โดยสัญลักษณ์ u แทนผู้ใช้คนนั้นๆ และด้วยการใช้ K_u เป็นอินพุทของ Hash Function แต่ละผู้ใช้จะต้องสร้าง Hash Chain $K_u^0, K_u^1, K_u^2, \ldots, K_u^n$ ซึ่ง $K_u^0 = K_u$ และ $K_u^i = h^i(K_u) = h(K_u^{i-1})$ และให้ประกาศ $K_u^n = h^n(K_u)$ ของผู้ใช้คนนั้นๆออกไป และเก็บ K_u ไว้เป็นความลับ โดยถ้าผู้ส่งต้องการสร้างลายเซ็นดิจิตอล สำหรับข้อความ a เพื่อส่งไปยังผู้รับ ต้องทำตามขั้นตอนนี้

ขั้นตอนที่ 1

ผู้ส่งต้องสร้าง K_{snd}^n ของตนขึ้นมา แล้วเลือก Sign Server (แทนด้วยสัญลักษณ์ SS) ที่ต้องการให้ช่วยสร้างลายเซ็นดิจิตอล เสร็จแล้วจึงส่งค่า Identify ของผู้ส่ง, จำนวนลายเซ็นดิจิตอลสูงสุดที่ต้องการสร้าง, K^n ของผู้ส่ง (K_{snd}^n) และ Identify ของ SS ไปยัง Certification Authority (CA) เพื่อให้ CA สร้าง Certificate ($Cert_{snd}$)

$$Cert_{snd} = SK_{CA}(Sender, n, K_{snd}^n, SS)$$

โดย SK_{CA} หมายถึง คีย์ลับของ CA หลังจากที่สร้าง $Cert_{snd}$ แล้ว จึงประกาศผ่าน Directory Services เช่น LDAP [7] เป็นต้น



รูปที่ 3.2 กระบวนการสร้าง Certificate

ขั้นตอนที่ 2

Sign Server จะต้องสร้างค่าจำนวนเฉพาะขนาดใหญ่ขึ้นมา 2 ค่า คือ p กับ q แล้วกำหนดให้ $m = p \times q$ และ $\phi(m) = (p-1) \times (q-1)$ จากนั้นหาค่า d โดยที่ค่าของ d ต้องเป็นไปตามสมการ $d \times e = 1 \pmod{\phi(m)}$ โดยที่ e ได้มาจากการสุ่มตัวเลขที่มีค่ามากกว่า 1 และน้อยกว่า $\phi(m)$ โดยมีเงื่อนไขที่ว่า ตัวหารร่วมมากของ e และ $\phi(m)$ ต้องมีค่าเท่ากับ 1 หรือ

เรียกอีกอย่างก็คือ มี e ต้องไม่มีตัวประกอบร่วมกับ $\phi(m)$ นั้นเอง โดยใช้ Euclidean Algorithm หรือเขียนในทางคณิตศาสตร์ว่า $GCD(e,\phi(m))=1$ โดยจะสามารถหาค่า d ได้ง่ายจากสมการ $d=((\partial\times\phi(m))+1)/e$ โดย ∂ คือจำนวนเต็มใดๆก็ตามที่สามารถทำให้ค่า d มีค่าเป็นจำนวนเต็มบวก หลังจากนั้นให้ SS จะเก็บ (d,p,q) ไว้เป็นคีย์ส่วนตัว แล้วประกาศ (e,m) เป็นคีย์สาธารณะของ Sign Server

ขั้นตอนที่ 3

เมื่อผู้ส่งต้องการสร้างลายเซ็นดิจิตอล สำหรับข้อความ a จะต้องเลือกตัวเลขแบบสุ่มขึ้นมา 2 ค่าคือ r,w เพื่อคำนวณหาค่า δ จาก (1)

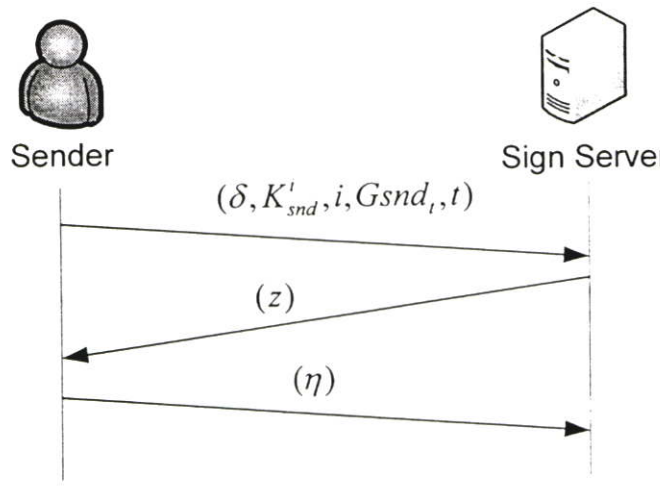
$$\delta = r^e h(a)(w^2+1) \pmod m \tag{3.1}$$

แล้วอ่านค่า Latitude, Longitude และเวลา t ขึ้นมาจาก Positioning System ที่ใช้ แทนด้วยสัญลักษณ์ (X_t,Y_t,t) เพื่อเอามาใส่ Position-Velocity-Time (PVT) to GeoLock Mapping Function [4] (ในบทความนี้ขอเรียกสั้นๆว่า Mapping Function) เพื่อให้ได้ค่า GeoLock ของผู้ส่งที่เวลา t แสดงด้วยสัญลักษณ์ $Gsnd_t$ หลังจากนั้นแล้วจึงส่งข้อความ $(\delta,K'_{snd},i,Gsnd_t,t)$ ต่อไปยัง SS เมื่อ SS ได้รับข้อความแล้วก็จะคำนวณหาค่าตำแหน่งพิกัดที่ตั้ง โดยประมาณของผู้ส่งที่เวลา t แสดงด้วยสัญลักษณ์ $(\hat{X}_t,\hat{Y}_t)$ ซึ่งสามารถคำนวณได้จาก (3.9) (ซึ่งจะอธิบายในหัวข้อถัดไป) เพื่อเอาค่ามาใส่ใน Mapping Function เพื่อให้ได้ผลลัพธ์เป็นค่า GeoLock ของผู้ส่งจากตำแหน่งที่ตั้งที่ประมาณขึ้นมาในเวลา t แสดงด้วยสัญลักษณ์ $\hat{G}snd_t$ แล้วจึงเปรียบเทียบค่า $Gsnd_t$ กับ $\hat{G}snd_t$ ว่ามีค่าตรงกันหรือไม่ ถ้าผลที่ได้ออกมาไม่เท่ากัน SS จะไม่สร้างลายเซ็นดิจิตอล ให้กับผู้ส่ง แล้วส่ง (Ack_1) กลับไปยังผู้ส่ง เมื่อผู้ส่งได้รับข้อความ (Ack_1) ก็จะคำนวณค่า $Gsnd_t$ ในขณะนั้นใหม่แล้วจึงส่งข้อความ $(\delta,K'_{snd},i,Gsnd_t,t)$ ใหม่ไปยัง SS ใหม่อีกครั้งหนึ่ง

แต่ในกรณีที่ผลการเปรียบเทียบออกมาตรงกัน SS จะเลือกค่าตัวเลขแบบสุ่มขึ้นมาคือ z โดย $z<m$ แล้วส่งค่า (z) ไปยังผู้ส่ง เมื่อผู้ส่งได้รับก็จะเลือกตัวเลขแบบสุ่มขึ้นมาอีกตัวคือ r' แล้วคำนวณค่า b จาก (3.2) แล้วคำนวณค่า η จาก (3.3) แล้วส่งค่า (η) ไปยัง SS อีกทีหนึ่ง

$$b = r\times r' \tag{3.2}$$

$$\eta = b^e\times(w-z) \pmod m \tag{3.3}$$



รูปที่ 3.3 กระบวนการคำนวณ z และ η โดยมีการตรวจสอบ $Gsnd_t$ ก่อน

ขั้นตอนที่ 4

เมื่อ SS ได้รับ η จะคำนวณค่า γ จาก (3.4) แล้ว SS ก็จะประมาณตำแหน่งที่ตั้งของผู้รับ ณ ขณะนั้น (t) จากข้อมูลอัพเดทสถานะการเคลื่อนที่ผู้รับที่มีการส่งมาอัพเดทตาม (3.9) แทนด้วยสัญลักษณ์ $(\hat{X}_t, \hat{Y}_t)$ ของผู้รับ แล้วก็นำไปใส่ใน Mapping Function เพื่อให้ได้ค่า GeoLock ของผู้รับ ณ เวลา t แทนด้วยสัญลักษณ์ $\hat{G}rcv_t$

$$\gamma = \eta^{-1} \pmod{m} \tag{3.4}$$

แล้วนำมาคำนวณหาค่า α จากสมการ (3.5) เพื่อส่ง $(\gamma, \alpha, \hat{G}rcv_t)$ ไปยังผู้ส่ง โดยเมื่อผู้ส่งได้รับข้อความแล้วจะคำนวณค่า c, s โดยสามารถคำนวณได้ตาม (3.6) และ (3.7) ตามลำดับ

$$\alpha = h(Cert_{snd} \oplus Grcv_t)^d (\delta(z^2+1)\eta^{-2})^{2d} \pmod{m} \tag{3.5}$$

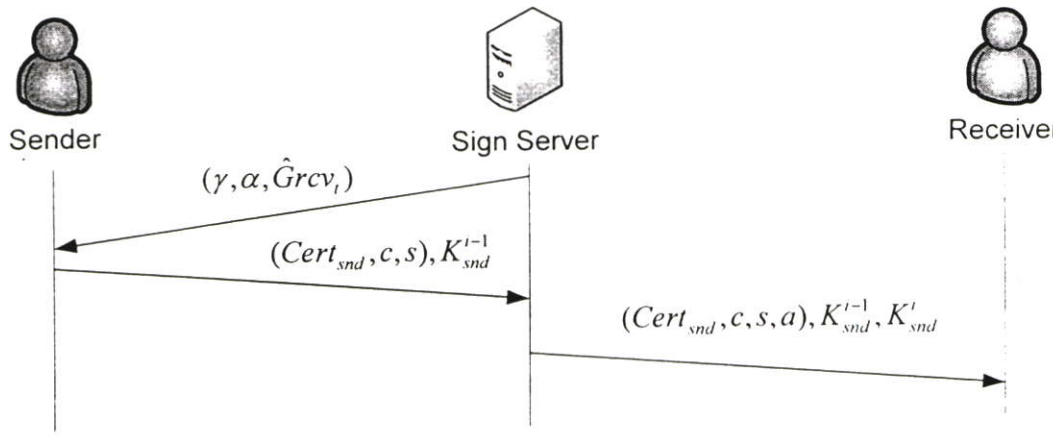
$$c = (wz+1) \times \gamma \times b^e = (wz+1)(w-z)^{-1} \pmod{m} \tag{3.6}$$

$$s = \alpha \times r^2 \times r'^4 \pmod{m} \tag{3.7}$$

โดย $(Cert_{snd}, c, s)$ ที่ได้มานี้เป็นลายเซ็นดิจิตอล ของข้อความ a ซึ่งมี Sign Server ช่วยในการสร้าง และใช้ปัจจัยทางตำแหน่งที่ตั้งของทั้งผู้ส่งและผู้รับและเวลาเข้ามามีส่วนร่วมในการสร้างด้วย ซึ่งสามารถตรวจสอบความถูกต้องของ ลายเซ็นดิจิตอล ได้ จากสมการ (3.8)

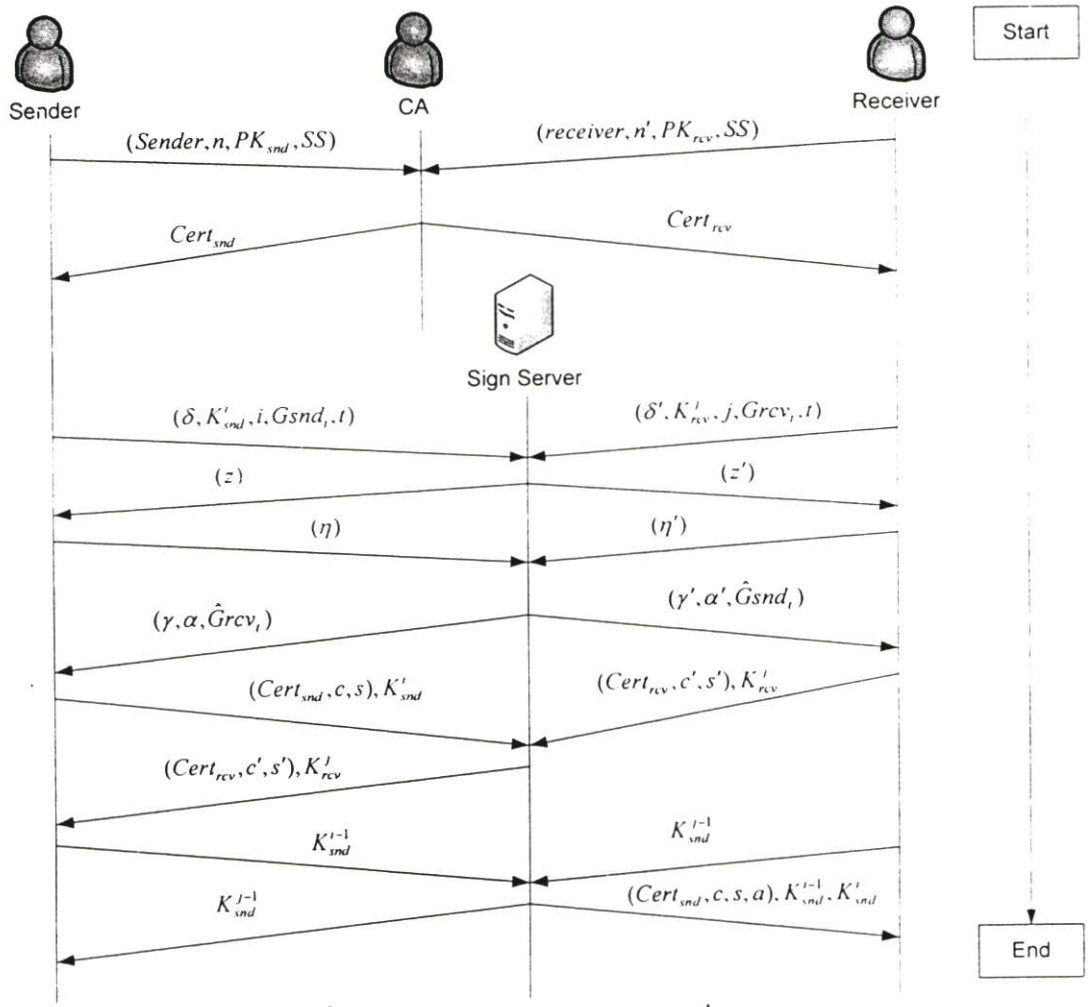
$$s^e \equiv h(Cert_{snd} \oplus Grcv_t) h(a)^2 (c^2+1)^2 \pmod{m} \tag{3.8}$$

หลังจากที่ผู้ส่งตรวจสอบความถูกต้องของลายเซ็นดิจิตอล ตาม (3.8) แล้วจึงส่ง ลายเซ็นดิจิตอล ที่สร้างขึ้นมานี้และ K_{snd}^{i-1} ไปยัง SS เมื่อ SS ได้รับข้อความ $(Cert_{snd}, c, s), K_{snd}^{i-1}$ ก็จะตรวจสอบความถูกต้องของลายเซ็นดิจิตอลตาม (3.8) อีกครั้ง และตรวจสอบว่า $K_{snd}^{i} = h(K_{snd}^{i-1})$ หรือไม่ ถ้าผลออกมาถูกต้องจึงส่ง $(Cert_{snd}, c, s, a), K_{snd}^{i-1}, K_{snd}^{i}$ ไปยังผู้รับ



รูปที่ 3.4 Sign Server ช่วยในการสร้าง Signature แล้วส่งไปยัง Receiver

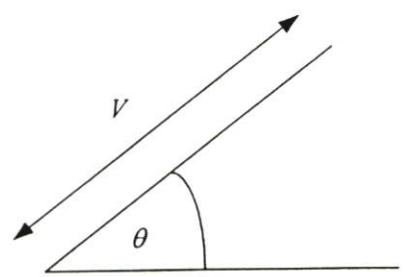
ตามขั้นตอนที่ผ่านมาเป็นอันเสร็จสิ้นการสร้างลายเซ็นดิจิตอล ซึ่งสามารถรองรับ Non-Repudiation of Sender (NRS) ถ้าต้องการจะให้รองรับ Non-Repudiation of Receiver (NRR) สามารถทำได้โดยผู้รับทำการสร้าง $(Cert_{rcv}, c', s')$ ได้เช่นกันตามขั้นตอนที่ 1-4 แล้วส่ง $(Cert_{rcv}, c', s'), K_{rcv}^{j}$ ไปยัง SS แล้วให้ SS ส่งต่อไปยังผู้ส่งอีกทีหนึ่ง ก่อนที่ผู้ส่งจะส่งค่า K_{snd}^{i-1} ไปยัง SS โดยเมื่อผู้ส่งได้รับข้อความ $(Cert_{rcv}, c', s'), K_{rcv}^{j}$ จะตรวจสอบ $(Cert_{rcv}, c', s')$ แล้วจึงส่งข้อความ $(Cert_{snd}, c, s), K_{snd}^{i-1}$ ไปยัง SS ต่อมาหลังจากที่ SS ได้รับทั้ง K_{snd}^{i-1} จากผู้ส่งและ K_{snd}^{j-1} แล้ว จึงส่ง $(Cert_{snd}, c, s, a), K_{snd}^{i-1}, K_{snd}^{i}$ ไปยังผู้รับ และส่ง K_{snd}^{j-1} ไปยังผู้ส่ง เป็นอันเสร็จสิ้นการสร้าง Space-time based Digital Signature สำหรับอุปกรณ์โมบายล์ ซึ่งรอบรับทั้ง NRS และ NRR



รูปที่ 3.5 Work Flow ของแบบจำลองที่นำเสนอ

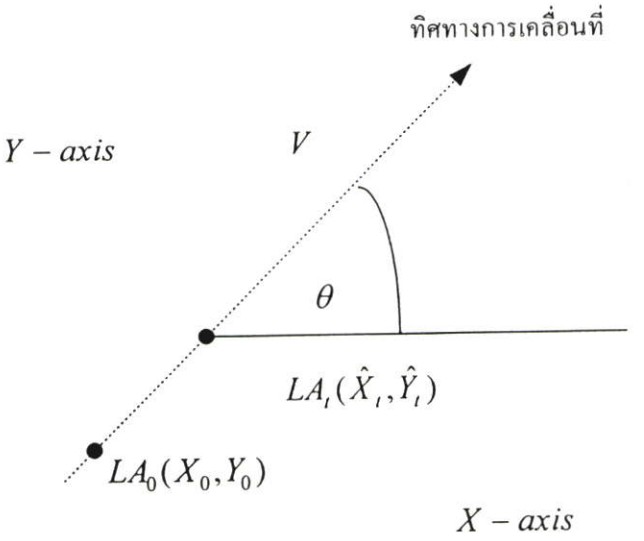
3.2 พารามิเตอร์ในการเคลื่อนที่ และ การอัพเดท

โดยในรูปที่ 5 แสดงพารามิเตอร์ในการเคลื่อนที่ของอุปกรณ์โมบายล์ ที่ใช้เป็นข้อมูลในการอัพเดทการเคลื่อนที่เพื่อส่งไปยัง SS ซึ่ง V คือความเร็วเฉลี่ยของอุปกรณ์โมบายล์ ในขณะนั้นๆ สามารถคำนวณได้จากระยะทางที่เคลื่อนที่ได้ในหนึ่งช่วงเวลา และ θ เป็นองศาที่มีค่าเป็นบวกระหว่าง ทิศทางของความเร็ว กับ แนวแกน x-axis ในระบบ Cartesian Coordinate ซึ่งทั้ง θ และ V นี้ อุปกรณ์โมบายล์ จะเป็นตัวคำนวณเองโดยอัตโนมัติ ไม่ได้มีการกำหนดค่าจากผู้ใช้



รูปที่ 3.6 พารามิเตอร์ในการเคลื่อนที่ ความเร็ว (V) และ ทิศทาง (θ)

ทั้งผู้รับและผู้ส่งต้องอัพเดทข้อมูลการเคลื่อนที่ของตนไปยัง Sign Server ตั้งแต่เริ่มต้นมีการติดต่อกัน ซึ่งจากข้อมูลอัพเดทสถานะการเคลื่อนที่ที่ส่งมานี้ทำให้ Sign Server สามารถประมาณตำแหน่งที่ตั้งของอุปกรณ์โมบายล์ได้ เช่น ถ้าให้ A เป็น อุปกรณ์โมบายล์ ตำแหน่งที่ตั้งของ A ที่เป็น Longitude และ Latitude ที่เวลา t_0 แสดงด้วยสัญลักษณ์ $LA_0(X_0, Y_0)$ โดยอุปกรณ์โมบายล์ต้องอ่านค่าตำแหน่งที่ตั้งของตนผ่าน Positioning System ได้เป็น $LA_1(X_1, Y_1)$ ที่เวลา t โดย $t = t_1, t_2, t_3, \ldots$ ให้ $t_i = t_0 + i\omega$ เมื่อ i = 1, 2, 3, ... และ ω คือค่าคงที่เพื่อกำหนดความกว้างของช่วงเวลาในการอ่านค่าตำแหน่งที่ตั้ง ทิศทางในการเคลื่อนที่ของ อุปกรณ์โมบายล์จะทำมุม θ กับแนวเส้น Latitude ดังนั้นที่เวลา t ใดๆ Sign Server สามารถที่จะประมาณตำแหน่งของอุปกรณ์โมบายล์ได้ ดังรูปที่ 6



รูปที่ 2.7 ตำแหน่งที่ประมาณขึ้นจาก ค่า $V_0, \theta, LA_0(X_0, Y_0)$

ด้วยข้อมูลอัพเดทสถานะการเคลื่อนที่ที่อุปกรณ์โมบายล์ส่งมานั้น Sign Server สามารถประมาณตำแหน่งของอุปกรณ์โมบายล์ในเวลาที่ t $(\hat{X}_t, \hat{Y}_t)$ ได้จากสมการ (3.9)

$$\begin{aligned} \hat{X} &= X_0 + (t - t_0)V\cos\theta \\ \hat{Y} &= Y_0 + (t - t_0)V\sin\theta \end{aligned} \tag{3.9}$$

ค่าความเร็วเฉลี่ยและทิศทางของ Mobile ที่เวลา t แสดงด้วยสัญลักษณ์ V_t, θ_t ตามลำดับ อุปกรณ์โมบายล์ต้องคำนวณเองโดยอัตโนมัติ สามารถหาได้จากสมการ (3.10)

$$V_t = \left(\frac{X_t - X_0}{t - t_0}\right)^2 + \left(\frac{Y_t - Y_0}{t - t_0}\right)^2$$
$$\theta_t = \arctan\left(\frac{Y_t - Y_0}{X_t - X_0}\right) \quad (3.10)$$

โดยทุกครั้งที่ อุปกรณ์โมบายล์อ่านค่า Latitude, Longitude จาก Positioning System เพื่อเอามาคำนวณค่าความเร็ว (V) และ ทิศทาง (θ) นั้น อุปกรณ์โมบายล์ต้องตัดสินใจว่าจะอัพเดทข้อมูลที่ได้มาล่าสุดหรือไม่ โดยการเปรียบเทียบค่า V_t, θ_t ณ เวลา t กับค่าเก่า V_0, θ_0 ว่าค่ามีการเปลี่ยนแปลงเกินค่า Threshold ที่กำหนดไว้หรือไม่ ถ้าไม่เกินก็จะไม่มีการส่งอัพเดทแต่อย่างใด แต่ถ้ามีการเปลี่ยนแปลงเกินค่า Threshold ที่กำหนดก็จะอัพเดทข้อมูลการเคลื่อนที่ไปยัง Sign Server แล้วเก็บค่า V_t, θ_t ล่าสุดไว้แทนที่ค่าเก่า V_0, θ_0 แทนที่ค่า t_0 ด้วยค่า t แทนค่า X_0, Y_0 ด้วยค่า X_t, Y_t ตามลำดับ โดย Update Message คือ

$$\{V_t, \theta_t, LA_t, t\}$$

ถ้ายิ่งค่า Threshold มีขนาดเล็กเท่าไหร่ ก็ยิ่งจะทำให้ อุปกรณ์โมบายล์อัพเดทสถานะการเคลื่อนที่บ่อยครั้งขึ้นเท่านั้น ดังนั้นการเลือกค่า Threshold ที่เหมาะสมนั้นจึงเป็นกุญแจสำคัญ โดยการเลือกค่า Threshold ที่เหมาะสมขึ้นอยู่กับความสมดุลกันระหว่างความน่าจะเป็นที่ค่า Geo-Lock ที่ได้มาจากตำแหน่งของ อุปกรณ์โมบายล์ที่ SS คำนวณนั้น กับค่า Geo-Lock ที่ได้จากการอ่านค่า ตำแหน่งที่ตั้งจาก Positioning System ของ อุปกรณ์โมบายล์มีค่าเท่ากัน กับ ความถี่ในการส่งข้อมูลอัพเดทสถานะการเคลื่อนที่ โดยในที่นี้ก็ขึ้นกับความแม่นยำของ Positioning System ที่ใช้ และขนาดของ Grid ด้วย

ทฤษฎี 1. ถ้าให้ $(Cert_{snd}, c, s)$ เป็น ลายเซ็นดิจิตอล ของข้อความ a ที่สร้างขึ้นมาจากวิธีการที่นำเสนอ ผู้รับจะสามารถตรวจสอบความถูกต้องได้จากสมการที่ (3.8) ซึ่งต่อไปนี้จะพิสูจน์ว่าสมการ (3.8) ที่ว่า $s^e \equiv h(Cert_{snd} \oplus Grcv_t)h(a)^2(c^2+1)^2 \pmod{m}$ เป็นจริง

$$s^e \equiv (\alpha \times r^2 \times r'^4)^e \pmod{m}$$

$$s^e \equiv \left[(h(Cert_{snd} \oplus Grcv_t)^d \times (\delta(z^2+1)\eta^{-2})^{2d} \times r^2 \times r'^4\right]^e \pmod{m}$$

$$s^e \equiv \left[\begin{array}{l}(h(Cert_{snd} \oplus Grcv_t)^d \times (r^e h(a)(w^2+1)(z^2+1)\cdot r^{-2e} \\ \cdot r'^{-2e}(w-z)^{-2})^{2d} \times r^2 \times r'^4\end{array}\right]^e \pmod{m}$$

$$s^e \equiv \left[\begin{array}{l}(h(Cert_{snd} \oplus Grcv_t)^d \times (r^e h(a)(w^2z^2+w+z+1) \\ \cdot r^{-2e} \cdot r'^{-2e}(w-z)^{-2})^{2d} \times r^2 \times r'^4\end{array}\right]^e \pmod{m}$$

$$s^e \equiv \left[\begin{array}{l}(h(Cert_{snd} \oplus Grcv_t)^d \times (r^e h(a)((wz+1)^2+(w-z)^2) \\ (w-z)^{-2} \cdot r^{-2e} \cdot r'^{-2e})^{2d} \times r^2 \times r'^4\end{array}\right]^e \pmod{m}$$

$$s^e \equiv \left[\begin{array}{l}(h(Cert_{snd} \oplus Grcv_t)^d \times (r^e h(a)(c^2+1)\cdot r^{-2e} \cdot r'^{-2e})^{2d} \\ \times r^2 \times r'^4\end{array}\right]^e \pmod{m}$$

$$s^e \equiv \left[\begin{array}{l}(h(Cert_{snd} \oplus Grcv_t) \times (h(a)(c^2+1)\cdot r^{-e} \cdot r'^{-2e})^2 \\ \times r^{2e} \times r'^{4e}\end{array}\right] \pmod{m}$$

$$s^e \equiv h(Cert_{snd} \oplus Grcv_t)h(a)^2(c^2+1)^2 \pmod{m}$$

บทที่ 4
การวิเคราะห์งานวิจัยที่นำเสนอ

ในบทนี้จะกล่าวถึงวิเคราะห์งานวิจัยที่นำเสนอ ทั้งในด้าน ความปลอดภัย, ประสิทธิภาพ, การใช้ปัจจัยทางภูมิศาสตร์ และ วิเคราะห์ในเรื่องของ Non-Repudiation Service

4.1 การวิเคราะห์ความปลอดภัย (Security Analysis)

4.1.1 การวิเคราะห์ตำแหน่งที่ตั้ง (Location based Analysis)

วิธีการที่นำเสนอมี Sign Server ทำหน้าที่ช่วยในการสร้างลายเซ็นดิจิตอล และรับข้อมูลอัพเดทสถานะการเคลื่อนที่ของอุปกรณ์โมบายล์ โดยทั้งผู้ส่งและผู้รับต้องส่งข้อมูลอัพเดทมายัง Sign Server เมื่อมีการเริ่มติดต่อกัน โดย Sign Server จะไม่สร้าง ลายเซ็นดิจิตอล ให้ ถ้า $Gsnd_t$ ที่ผู้ส่งส่งมาให้มีค่าไม่ตรงกับ $\hat{G}snd_t$ ที่ Sign Server คำนวณขึ้นมา และผู้รับจะไม่สามารถตรวจสอบความถูกต้องได้ ถ้าไม่อยู่ในบริเวณและเวลาที่กำหนดไว้ ความถูกต้องของวิธีการที่นำเสนอนั้น ก็ขึ้นอยู่กับความแม่นยำของ Positioning System ที่ใช้ เพื่อใช้ในการกำหนดขนาดของ Grid และค่า Threshold เพื่อให้อุปกรณ์โมบายล์ใช้ในการตัดสินใจว่าจะอัพเดทข้อมูลสถานะการเคลื่อนที่ไปยัง Sign Server ดีหรือไม่ด้วย ซึ่งทำให้วิธีที่ใช้ในการสร้างลายเซ็นดิจิตอลนั้นสามารถเพิ่มระดับความปลอดภัยจากการสร้างลายเซ็นดิจิตอล แบบปกติภายในความยาวคีย์เท่าเดิม และสามารถลดเวลาในการประมวลผลของฝั่งผู้ส่งได้ถึงประมาณ 97.5% เมื่อเทียบกับโมเดลของ Abe-Fujisaki "How to date blind signature" [11] ภายใต้ 1,024 bit modulus n ถ้าผู้ประสงค์ร้ายสามารถที่จะโจมตีวิธีในการสร้างลายเซ็นดิจิตอลของเราได้แสดงว่าเขาสามารถที่จะแฮ็คอัลกอริทึม SHA หรือ MD5 และ RSA ได้ โดยเมื่อมีการนำเอาตำแหน่งที่ตั้งเข้ามามีส่วนร่วมในกระบวนการสร้างและตรวจสอบความถูกต้อง ลายเซ็นดิจิตอล เป็นการเพิ่มระดับความยากในการที่จะโจมตีระบบของเรา จึงเป็นการเพิ่มระดับความปลอดภัยให้กับการสร้างลายเซ็นดิจิตอลให้มากขึ้นนั้นเอง เพื่อนำไปประยุกต์ใช้กับแอพพลิเคชันประเภท e-commerce หรือ Location Based Services ต่าง ๆ ต่อไป

4.1.2 การวิเคราะห์เรื่องการแอบอ้าง (Non-Repudiation Analysis)

วิธีการที่นำเสนอนี้สามารถรองรับทั้ง Non-Repudiation of Sender หรือ เรียกสั้นๆว่า NRS และ Non-Repudiation of Receiver หรือ NRR ในกรณีที่ผู้ส่งอ้างว่าไม่ได้มีการส่งข้อความ (NRS) ผู้รับสามารถส่ง $(Cert_{snd}, c, s, a), K_{snd}^{i-1}, K_{snd}^{i}, Grcv_t$ ไปให้ผู้ตรวจสอบ เพื่อให้ตรวจสอบให้ โดยเริ่มต้นจากการตรวจสอบ $Cert_{snd}$ ซึ่งสร้างโดย CA ซึ่งประกอบไปด้วย

Identify ของผู้รับ, K^n ของผู้ส่ง K^n_{snd} และ Identify ของ SS เป็นต้น หลังจากนั้นก็ตรวจสอบลายเซ็นดิจิตอล จาก (3.8) หลังจากนั้นก็ตรวจสอบดูว่า K^n_{snd} สามารถหาได้โดยการใช้ One-way Hash Function กับ K^{i-1}_{snd}, K^i_{snd} หรือไม่ และ $K^i_{snd} = h(K^{i-1}_{snd})$ หรือไม่ ถ้าทุกอย่างถูกต้อง ผู้ตรวจสอบสามารถตัดสินได้เลยว่า ผู้ส่งได้ส่งข้อความนี้ไปยังผู้รับจริง ซึ่งในกรณี NRS ก็ใช้วิธีเดียวกับการตรวจสอบ ในกรณีที่ผู้ส่งอ้างเช่นกัน ทั้งนี้ทั้งนั้นเพราะคุณสมบัติของ One-way Hash Function ที่ว่า "ง่ายต่อการประมวลผล แต่ยากมากต่อการทำกระบวนการย้อนกลับ" และ $K^i_{snd} = h(K^{i-1}_{snd})$ ซึ่งนั้นก็หมายความว่า ถ้าผู้ประสงค์ร้ายสามารถโจมตีวิธีที่นำเสนอได้ แปลว่าเค้าสามารถโจมตีฟังก์ชัน One-way Hash เช่น SHA1, MD5 ได้โดยสมบรูณ์ ซึ่งทำได้ยากมาก

4.1.3 การวิเคราะห์คุณสมบัติการสุ่ม (Randomization)

ในวิธีการสร้างลายเซ็นดิจิตอล ที่นำเสนอนี้ Sign Server ต้องสร้างตัวเลขแบบสุ่มขึ้นมาตัวหนึ่งคือ z เพื่อใช้ในกระบวนการ Sign ลงบนข้อความของผู้ส่ง ซึ่งด้วยคุณสมบัติของการ Random [8] นี้ ทำให้วิธีการที่นำเสนอนี้ปลอดภัยจากการโจมตีแบบ Chosen-text Attack [9] ได้ ซึ่งก็มีหลาย ๆ งานวิจัยที่มีคุณสมบัตินี้ เช่น [3], [6], [8], [10] แต่ก็มีบางงานวิจัยที่ไม่มีคุณสมบัตินี้ เช่น [11], [12] เป็นต้น

ในขั้นตอนที่ 3 ของการสร้างลายเซ็นดิจิตอล ที่นำเสนอนี้ เมื่อผู้ส่งต้องการสร้าง ลายเซ็นดิจิตอล ของข้อความ $(\delta, K^i_{snd}, i, Gsnd_t, t)$ มาให้ Sign Server หลังจากที่ Sign Server ได้ตรวจสอบค่า $Gsnd_t$ ที่ได้รับมา ว่ามีค่าตรงกับค่า $\hat{G}snd_t$ ที่ทาง Sign Server คำนวณขึ้นหรือไม่ ถ้าถูกต้อง Sign Server จะต้องทำการเลือกตัวเลขแบบสุ่มขึ้นมาหนึ่งตัวคือ z เพื่อใช้ในกระบวนการสร้างลายเซ็นดิจิตอล ซึ่งถ้าผู้ส่งต้องการสร้างค่า α โดยไม่สนใจค่า z ผู้ส่งจะต้องคำนวณค่า η' โดยที่ $\eta'^2 \equiv (z^2+1) \pmod{m}$ ในขั้นตอนที่ 3 ถ้าแม้ว่าจะทราบค่า z และ m ก็ไม่สามารถที่จะค่านวณค่า η' โดยที่ไม่ทราบตัวประกอบของ m ได้ ซึ่งเป็นที่รู้กับว่ายากมาก [13] ดังนั้นจะเห็นได้ว่า วิธีการที่นำเสนอนี้ ผู้ส่งไม่สามารถที่จะเอาค่า z ออกจากกระบวนการสร้างลายเซ็นดิจิตอล $(Cert_{snd}, c, s)$ ได้

4.1.4 การวิเคราะห์คุณสมบัติ Partially Blindness

คุณสมบัติของ Partially Blindness การันตีว่า ทุกๆลายเซ็นดิจิตอล ที่สร้างโดย Sign Server จะมีการฝังค่า $Cert_{snd}$ และ $\hat{G}rcv_t$ ที่ Sign Server คำนวณขึ้นเอาไว้ ผู้ส่งจะไม่สามารถเปลี่ยนแปลงข้อมูลที่ฝังอยู่ภายในได้ ถ้ายังต้องการให้สามารถตรวจสอบความถูกต้องของ ลายเซ็นดิจิตอล ได้

เพื่อที่จะลบหรือเปลี่ยนแปลงค่าที่ฝังอยู่ใน ลายเซ็นดิจิตอล $(Cert_{snd}, \hat{G}rcv_t)$ ผู้ส่งจะต้องคำนวณค่า η' หรือ δ' ขึ้นมาใหม่ ซึ่งทั้งสองค่าสามารถหาได้จากสมการต่อไปนี้คือ $\eta'^4 \equiv h(Cert_{snd} \oplus Grcv_t) \pmod{m}$ และ $\delta'^2 \equiv h(Cert_{snd} \oplus Grcv_t)^{-1} \pmod{m}$

ซึ่งจะไม่สามารถหาค่าได้โดยที่ไม่ทราบค่าตัวประกอบของ m ดังนั้นในวิธีการที่นำเสนอนี้ ผู้ส่งจะไม่สามารถลบหรือเปลี่ยนแปลงข้อมูลที่ฝังอยู่ภายในได้

4.1.5 การวิเคราะห์คุณสมบัติ Unforgeability

ในกรณีที่ผู้ประสงค์ร้ายต้องการจะปลอมแปลงลายเซ็นดิจิตอลที่สร้างจากวิธีการที่นำเสนอลายเซ็นดิจิตอลที่ถูกต้อง ถ้าต้องการสามารถตรวจสอบความถูกต้องได้โดยใช้สมการ $s^e \equiv h(Cert_{snd} \oplus Grcv_t)h(a)^2(c^2+1)^2 \pmod m$ ผู้ส่งต้องคำนวณหาค่า s จากสมการที่ (11)

$$s \equiv h(Cert_{snd} \oplus Grcv_t)^d h(a)^{2d}(c^2+1)^{2d} \pmod m \tag{11}$$

ซึ่งถ้าผู้ประสงค์ร้ายไม่รู้ตำแหน่งที่ตั้งของผู้รับ ก็จะไม่สามารถสร้าง หรือ ปลอมแปลงลายเซ็นดิจิตอลปลอมได้ หรือต่อให้ทราบค่า $h(Cert_{snd} \oplus Grcv_t)$ และ $h(a), c$ ก็ไม่สามารถหาค่า d จาก e, m โดยที่ไม่ทราบตัวประกอบของ m ได้ตามอัลกอริทึมของ RSA [14] หรือในอีกทางหนึ่งถ้าทราบค่า $h(Cert_{snd} \oplus Grcv_t)$ และ $h(a), s$ ก็ไม่สามารถคำนวณค่า c จากสมการที่ (12) ได้ โดยที่ไม่ทราบค่าตัวประกอบของ m

$$c^2 \equiv \left((s^e h(Cert_{snd} \oplus Grcv_t)^{-1} h(a)^{-3})^{1/2} - 1 \right) \pmod m \tag{12}$$

4.1.6 คุณสมบัติเรื่อง Unlinkability

ในขั้นตอนที่ 3 ของวิธีการที่นำเสนอนี้ ผู้ส่งจะต้องส่งค่า η และ δ ไปยัง Sign Server เพื่อให้ Sign Server สร้างลายเซ็นดิจิตอลสำหรับข้อความ a ซึ่งสามารถคำนวณจากสมการที่ (1) และ (3) คือ $\delta = r^e h(a)(w^2+1) \pmod m$ และ $\eta = b^e \times (w-z) \pmod m$ ตามลำดับ ต่อมาผู้ส่งจะต้องคำนวณค่า s, c จากสมการที่ (7) และ (6) คือ $s = \alpha \times r^2 \times r'^4 \pmod m$ และ $c = (wz+1) \times \gamma \times b^e = (wz+1)(w-z)^{-1} \pmod m$ ซึ่งภายใต้ข้อความที่ฝังอยู่ในลายเซ็นดิจิตอล คือ $Cert_{snd}$ และ $\hat{G}rcv_t$ เดียวกัน Sign Server จะไม่สามารถที่จะหาความสัมพันธ์ระหว่าง (δ, η) กับ $(c, s, Cert_{snd}, \hat{G}rcv_t)$ ได้ เพราะค่าของ r, w, r' จะถูกสร้างขึ้นแบบสุ่มและเก็บไว้เป็นความลับโดยผู้ส่งเท่านั้น และนี้คือคุณสมบัติ Unlinkability ภายใต้ $Cert_{snd}$ และ $\hat{G}rcv_t$ เดียวกัน

4.2 การวิเคราะห์ประสิทธิภาพ (Performance Analysis)

สำหรับการเปรียบเทียบประสิทธิภาพกับวิธีการอื่นๆนั้น งานวิจัยนี้ตั้งอยู่บนสมมติฐานเดียวกับ [10], [16] ที่ว่าที่ modulus n นั้น การทำการคำนวณ modular exponentiation ใช้เวลาในการประมวลผลเท่ากับ 0.3246|n| เท่าของเวลาในการประมวลผล modular multiplication และเวลาในการคำนวณหาค่า inverse ใช้เวลาเท่าๆกับการหาค่า modular exponentiation และจาก [17] ที่ว่าเวลาในการประมวลผล Hash Function กินเวลาน้อยกว่าการทำ modular multiplication และจาก [18] Ghanem และ Wahab กล่าวไว้ว่าการทำ XOR นั้นมีความปลอดภัยและ ใช้เวลาในการประมวลผลเร็วมากๆ

4.2.1 การประมวลผลทางฝั่งผู้ส่ง (Computation Load for Sender)

มีหลายวิธีในการสร้าง ลายเซ็นดิจิตอล ที่ได้ถูกนำเสนอมาแล้วที่ ทางฝั่งของอุปกรณ์โมบายล์ต้องทำการประมวลผล modular exponentiation และ inverse computation ซึ่งออกจะเป็นการประมวลผลที่หนักเกินไปสำหรับอุปกรณ์โมบายล์ที่มีความสามารถในการประมวลผลและพลังงานที่จำกัด แต่ใน [6] Hung-Yu Chien, Jinn-Ke Jan และ Yhu-Min Tseng ได้นำเสนอ Partially Blind Signature โดยใช้หลักการพื้นฐานของ RSA ซึ่งผู้ส่งใช้การประมวลผลแค่ 21 modular multiplication และ 2 hash เท่านั้น ซึ่งเมื่อเปรียบเทียบกับ [11] Abe-Fujisaki นำเสนอ Partially Blind Signature ซึ่งใช้หลักการของ RSA ใช้การประมวลผลทั้งหมด 2 modular exponentiation, 1 inverse computation, 4 hash และ 4 modular multiplication ทำให้สามารถลดการประมวลผลทางฝั่งผู้ส่งไปได้ประมาณ 98 % ซึ่งในวิธีการที่นำเสนอนี้ ได้นำวิธีในการสร้างลายเซ็นดิจิตอล ใน [6] มาประยุกต์ใช้ โดยในตอนแรกสุดนั้น ผู้ใช้แต่ละคนต้องสร้าง Hash Chain ก่อน หลังจากนั้นก็จะมีการคำนวณค่าอัพเดทข้อมูลสถานะการเคลื่อนที่ $\{V_t, \theta_t, LA_t, t\}$ ซึ่งเป็นการคำนวณแบบง่าย ไม่มีความซับซ้อน จึงไม่เป็นภาระสำหรับอุปกรณ์ Mobile Device และจาก [4] ฟังก์ชัน PVT-to-GeoLock ที่ใช้ในการหาค่า GeoLock ของผู้ใช้นั้น เป็นฟังก์ชันที่รับอินพุทเป็นตำแหน่งที่ตั้งของอุปกรณ์โมบายล์ และเวลา เพื่อให้ได้ค่า GeoLock ออกมาซึ่งใช้การประมวลผลเท่ากับการทำ Hash Function ดังนั้นในวิธีการที่นำเสนอนี้ ใช้การประมวลผลทางฝั่งผู้ใช้ทั้งสิ้น 21 modular multiplication ,4 hash และ 2 XOR ซึ่งจากที่ได้กล่าวมา [18] การทำ XOR นั้นถือว่าเป็นการคำนวณที่เร็วมาก ทำให้เมื่อเปรียบเทียบกับ [11] โดยใช้สูตร $\%\, computation\, reduce = \frac{T_{old} - T_{new}}{T_{old}} \times 100\%$ สามารถลดการประมวลผลทางฝั่งผู้ใช้ไปได้ทั้งหมดประมาณ 97.5 % ที่ modulus n เท่ากับ 1024 bit

4.2.2 การประมวลผลทางฝั่ง Sign Server (Computation Load for Sign Server)

ถึงการประมวลผลทางฝั่ง Sign Server จะไม่เป็นประเด็นหลักของงานที่นำเสนอ แต่ในหัวข้อนี้ก็จะทำการเปรียบเทียบให้ได้ทราบกัน โดย [11] ซึ่งใช้การประมวลผลทางฝั่ง Server ทั้งหมด 1 inverse computation และ 1 modular exponentiation และ [6] ใช้การประมวลผลทั้งหมด 1 inverse computation, 2 modular exponentiation และ 6 modular multiplication

ในวิธีการที่นำเสนอ Sign Server ทำหน้าที่ช่วยในการสร้าง ลายเซ็นดิจิตอล และคอยรับข้อมูลอัพเดทสถานะการเคลื่อนที่จากอุปกรณ์โมบายล์ โดยเริ่มต้นตั้งแต่ การสร้างคู่ RSA Key และประกาศออกไป หลังจากนั้นก็ใช้อัลกอริทึมอสมมาตร ในการสร้าง ลายเซ็นดิจิตอล จะเห็นได้ว่า วิธีการที่นำเสนอไม่ได้มีการลดการประมวลผลทางฝั่ง Sign Server แต่อย่างใดเมื่อเปรียบเทียบกับ [11], [6] โดยส่วนที่เพิ่มขึ้นมาก็คือ การคำนวณหาค่า $\hat{G}rcv_t$, $\hat{G}snd_t$ ของทั้งผู้รับและผู้ส่ง, การตรวจสอบความถูกต้องของ ลายเซ็นดิจิตอล อีกรอบก่อนส่งถึงผู้รับ และการตรวจสอบความถูกต้องของ NRS และ NRR แต่อย่างไรก็ตาม เนื่องจากในงานวิจัยนี้ ถือว่า Sign Server มีความสามารถในการประมวลผลที่สูง และ มีพลังงานที่มาก เมื่อเทียบกับทางฝั่ง Mobile Device ซึ่งมีความสามารถในการประมวลผลที่ต่ำ และมีพลังงานที่จำกัด ดังนั้นจึงเห็นว่า การลดการประมวลผลทางฝั่งอุปกรณ์โมบายล์มีความจำเป็นกว่า

4.2.3 การติดต่อสื่อสาร (Communication Cost)

ในการทำการลงลายมือชื่อบนข้อความนั้น ผู้ส่งต้องเชื่อถือ Sign Server เพื่อให้ Sign Server ช่วยในการประมวลผลที่หนักๆ ในการสร้าง ลายเซ็นดิจิตอล อีกทั้ง Mobile Device ยังต้องส่งข้อมูลอัพเดทสถานะการเคลื่อนที่มาให้ Sign Server เมื่อจำเป็น ดังนั้นจึงจำเป็นต้องมีช่องทางการสื่อสารระหว่าง Sign Server และผู้ส่ง ในระหว่างกระบวนการสร้างลายเซ็นดิจิตอล

โดยจะเห็นได้ว่าเมื่อเปรียบเทียบกับการสร้าง ลายเซ็นดิจิตอล แบบทั่วไป [15] นั้น Over Head ที่เกิดจากการติดต่อสื่อสารผ่านเน็ทเวิร์ค, การสร้าง $K_u^n = h^n(K_u)$ ของแต่ละผู้ใช้ และการส่งข้อมูลอัพเดทสถานะการเคลื่อนที่ของแต่ผู้ใช้นั้น อาจจะเป็นการเพิ่มค่าใช้จ่ายให้กับระบบ แต่วิธีการที่เรานำเสนอนั้นเหมาะสมกับอุปกรณ์โมบายล์ซึ่งมีทรัพยากร และความสามารถในการประมวลผลที่จำกัด อีกทั้งยังใช้ข้อดีของอุปกรณ์โมบายล์นั้นก็คือ ตำแหน่งที่ตั้ง ซึ่งสามารถเคลื่อนที่ได้ ไม่จำเป็นต้องอยู่กับที่ มามีส่วนร่วมในการสร้างลายเซ็นดิจิตอล และมีการนำ Mobility Model มาประยุกต์ใช้ด้วย จึงเป็นการเพิ่มระดับความปลอดภัยให้กับการสร้างลายเซ็นดิจิตอล อีกด้วย โดยตารางที่ 1 แสดงคุณสมบัติต่างๆ และเวลาที่ใช้ในการประมวลผลบนฝั่งผู้ใช้โดยเปรียบเทียบระหว่างวิธีที่นำเสนอ, [6], [8], [11] และ [12]

ตารางที่ 4.1 เปรียบเทียบคุณสมบัติต่างๆ และ เวลาในการประมวลผลที่ใช้ทางฝั่งผู้ใช้

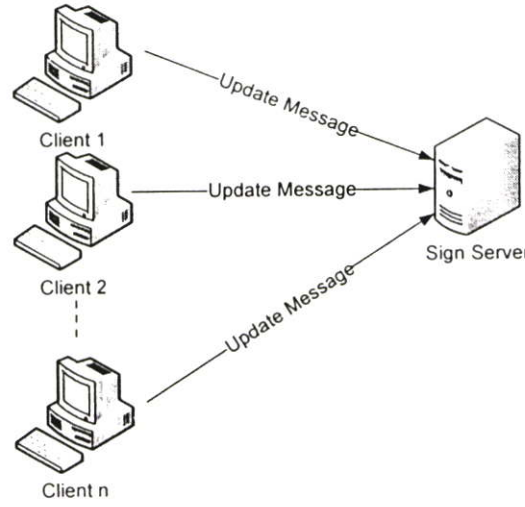
	Our proposed scheme	Ferguson's scheme [8]	Abe-Fujisaki's scheme [11]	Chaum's Scheme [12]
Mathematical foundation	RSA	RSA	RSA	RSA
Randomization property	Yes	Yes	No	Yes
Unlinkability property	Yes	Yes	Yes	No
Partial blindness property	Yes	No	Yes	No
Unforgeability property	Yes	Yes	Yes	Yes
Computations for the user	$21T_m + 4T_h + 2T_{XOR}$	$4T_e + 1T_i + 2T_h + 3T_m$	$2T_e + 1T_i + 4T_h + 4T_m$	$2T_e + 1T_i + 2T_h + 2T_m$
Computations Reduced from [11] **	97.5 %	- 66 %	0 %	0.4 %

* T_e : เวลาที่ใช้ในการทำ 1 modular exponentiation; T_i : เวลาที่ใช้ในการทำ 1 inverse computation; T_m : เวลาที่ใช้ในการทำ 1 modular multiplication; T_h : เวลาที่ใช้ในการทำ 1 hash function; T_{XOR} : เวลาที่ใช้ในการทำ 1 XOR operation

** ที่ modulus n เท่ากับ 1024 bits และจาก [16] T_e=0.3246|n| T_m

4.3 การจำลองการเคลื่อนที่ของอุปกรณ์โมบายล์

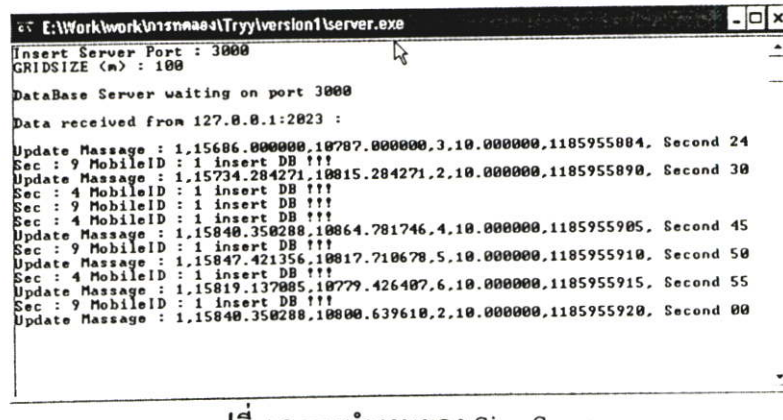
ในงานวิจัยนี้ ได้ทำการจำลองการเคลื่อนที่ของอุปกรณ์โมบายล์ เพื่อตรวจสอบความถูกต้องของค่า GeoLock ที่อุปกรณ์โมบายล์ทำการคำนวณ กับค่า GeoLock ที่ได้จากการประมาณตำแหน่งที่ตั้งของอุปกรณ์โมบายล์ที่ Sign Server ได้ทำการคำนวณขึ้น ว่ามีเปอร์เซ็นต์ความถูกต้อง (ในที่นี้เรียกค่านี้ว่าค่า Tolerance) มากน้อยเพียงใด ภายใต้ขนาดของ Grid Size และความเร็วของอุปกรณ์โมบายล์ที่แตกต่างกัน



รูปที่ 4.1 สภาพแวดล้อมของการทดลอง มี Sign Server รับข้อมูลอัพเดทจากอุปกรณ์โมบายล์จำลอง

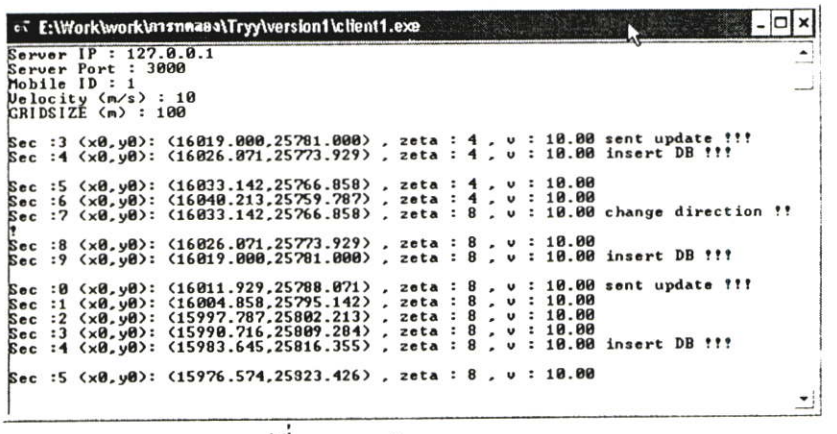
โดยองค์ประกอบของการจำลองการเคลื่อนที่ของอุปกรณ์โมบายล์ประกอบไปด้วย 2 ส่วน คือ

- Sign Server เป็นแอพพลิเคชันที่พัฒนาจากภาษา C++ โดยใช้ Microsoft Visual Studio 6.0 บนแพลตฟอร์ม Window ทำหน้าที่เป็น Sign Server คอยรับข้อมูลอัพเดทสถานะการเคลื่อนที่ของอุปกรณ์โมบายล์ต่างๆ เพื่อนำมาทำการประมาณตำแหน่งที่ตั้งของอุปกรณ์โมบายล์นั้นๆ ตามสมการ (3.9) แล้วจึงนำตำแหน่งที่ตั้งที่ประมาณได้มาคำนวณค่า GeoLock ของอุปกรณ์โมบายล์ ณ เวลาต่างๆเก็บไว้ เพื่อนำใช้ในคำนวณเปอร์เซ็นต์ความถูกต้องภายหลัง



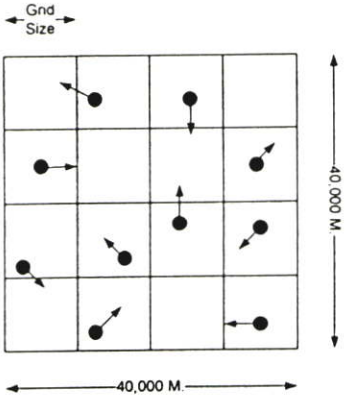
รูปที่ 4.2 การทำงานของ Sign Server

- Client เป็นเป็นแอพพลิเคชันที่พัฒนาจากภาษา C++ โดยใช้ Microsoft Visual Studio 6.0 บนแพลตฟอร์ม Window ทำหน้าที่เป็นจำลองตนเองเป็นอุปกรณ์โมบายล์ โดยจะกำหนดจุดตั้งต้นขึ้นแบบสุ่ม ภายในขอบเขตของที่กำหนดไว้ และสุ่มทิศทางในการเคลื่อนที่ด้วย ซึ่งสามารถเคลื่อนที่ได้ทั้งหมด 8 ทิศทางด้วยกัน และจะส่งข้อมูลอัพเดทสถานะการเคลื่อนที่ของตนไปยัง Sign Server เมื่อจำเป็นนั้นก็คือ เมื่อมีการเปลี่ยนทิศทางนั้นเอง และทำการเก็บค่า GeoLock ที่ อุปกรณ์โมบายล์จำลอง คำนวณได้ที่เวลาต่างๆ เก็บไว้อีกด้วย



รูปที่ 4.3 การทำงานของ Client

โดยในการจำลองการเคลื่อนที่ของอุปกรณ์โมบายล์นี้ เราทดลองอยู่บนพื้นที่ 40*40 ตารางกิโลเมตร โดยทำการจำลองอุปกรณ์โมบายล์จำนวน 10 เครื่อง โดยสุ่มตำแหน่งเริ่มต้นภายในพื้นที่ที่กำหนดไว้ และจะทำการส่งข้อความอัพเดทสถานะการเคลื่อนที่ของตนไปยัง Sign Server เมื่อจำเป็น โดยอุปกรณ์โมบายล์จำลองนั้นจะเคลื่อนที่ด้วยความเร็วที่ต่างกัน (โดยในการจำลองนี้ใช้ความเร็วที่ 1, 5, 10 เมตรต่อวินาที) และภายใต้ขนาด Grid Size ที่แตกต่างกัน (โดยในการจำลองนี้กำหนดขนาด Grid Size ที่ 10, 50, 100, 500, 1000, 5000, 10,000 เมตร) โดย ณ ช่วงเวลาหนึ่งๆแต่ละ Grid จะให้ค่า GeoLock ที่แตกต่างกัน ดังนั้นฟังก์ชัน GeoLock Mapping จึงเป็นฟังก์ชันที่รับอินพุทเป็น ตำแหน่งที่ตั้ง และเวลา คือ f(X, Y, t) = GeoLock โดย X คือตำแหน่งที่ตั้งของอุปกรณ์โมบายล์ในแนวนอน, Y คือตำแหน่งที่ตั้งในแนวตั้ง และ t คือเวลาในขณะนั้น



รูปที่ 4.4 ภาพจำลองสมมติฐานของการทดลอง

หลังจากนั้นจึงทำการนำค่า GeoLock ที่ได้จากทั้งอุปกรณ์โมบายล์จำลอง และ Sign Server จำลองนำมาเปรียบเทียบเพื่อหาค่า Tolerance โดยในงานวิจัยนี้ ได้ทำการทดลองทั้งหมด 3 การทดลองดังนี้

การทดลองที่ 1 :

ทำการรัน Client จำนวน 10 Client โดยกำหนดจุดตั้งต้นแบบสุ่มภายในขอบเขตที่กำหนดไว้ และแต่ละ Client จะส่งข้อมูลอัพเดทสถานะการเคลื่อนที่ของตนไปยัง Sign Server เมื่อมีการเปลี่ยนแปลงทิศทางการเคลื่อนที่ โดยกำหนดความเร็วของ Client เป็น 1 เมตรต่อวินาที ได้ผลดังนี้

ตารางที่ 4.2 ผลการทดลองเมื่อกำหนดความเร็ว Client เป็น 1 เมตรต่อวินาที

Grid Size (m)	Percent Tolerance (%)
10	78.8345244€
50	95.09803922
100	97.31417864
500	98.80829016
1000	98.98024219
5000	99.84301413
10000	99.73958333

การทดลองที่ 2 :

ทำการรัน Client จำนวน 10 Client โดยกำหนดจุดตั้งต้นแบบสุ่มภายในขอบเขตที่กำหนดไว้ และแต่ละ Client จะส่งข้อมูลอัพเดทสถานะการเคลื่อนที่ของตนไปยัง Sign Server เมื่อมีการเปลี่ยนแปลงทิศทางการเคลื่อนที่ โดยกำหนดความเร็วของ Client เป็น 5 เมตรต่อวินาที ได้ผลดังนี้

ตารางที่ 4.3 ผลการทดลองเมื่อกำหนดความเร็ว Client เป็น 5 เมตรต่อวินาที

Grid Size (m)	Percent Tolerance (%)
10	55.00454959
50	78.15183009
100	88.22016461
500	97.7443609
1000	98.95120839
5000	98.99912968
10000	99.74293059

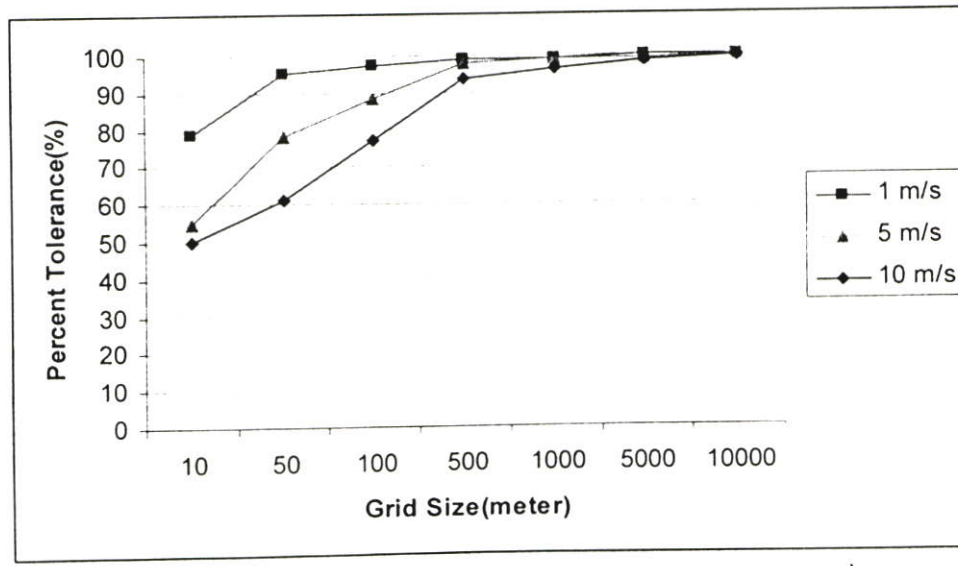
การทดลองที่ 3 :

ทำการรัน Client จำนวน 10 Client โดยกำหนดจุดตั้งต้นแบบสุ่มภายในขอบเขตที่กำหนดไว้ และแต่ละ Client จะส่งข้อมูลอัพเดทสถานะการเคลื่อนที่ของตนไปยัง Sign Server เมื่อมีการเปลี่ยนแปลงทิศทางการเคลื่อนที่ โดยกำหนดความเร็วของ Client เป็น 10 เมตรต่อวินาที ได้ผลดังนี้

ตารางที่ 4.4 ผลการทดลองเมื่อกำหนดความเร็ว Client เป็น 10 เมตรต่อวินาที

Grid Size (m)	Percent Tolerance (%)
10	50.05506608
50	61.01344364
100	77.17842324
500	93.87159533
1000	96.1737332
5000	98.49974133
10000	99.72779857

จากผลการทดลองสามารถนำมาสร้างกราฟแสดงความสัมพันธ์ระหว่าง Percent Tolerance และ Grid Size ที่ความเร็วต่างๆ ดังรูปที่ 4.3 จะเห็นได้ว่า ขนาด Grid Size และความเร็วของอุปกรณ์โมบายล์มีผลต่อค่า Percent Tolerance นั้นก็คือ ถ้าอุปกรณ์โมบายล์เคลื่อนที่ด้วยความเร็วสูง แต่เรากำหนดค่า Grid Size ให้มีขนาดเล็กเกินไป ก็จะทำให้ Percent Tolerance ที่ได้มีค่าต่ำ ส่งผลให้การเข้ารหัส และ ถอดรหัสของ Space-Time based Digital Signature เกิดความผิดพลาดขึ้นได้ ดังนั้นการเลือกขนาดของ Grid Size ให้เหมาะสมกับการใช้งานจริง จึงเป็นอีกหนึ่งปัจจัยที่สำคัญ เพื่อให้สามารถสร้าง Space-Time based Digital Signature ได้อย่างมีประสิทธิภาพ



รูปที่ 4.5 กราฟแสดงความสัมพันธ์ระหว่าง Percent Tolerance และ Grid Size ที่ความเร็วต่างๆ

บทที่ 5
สรุปงานวิจัยที่นำเสนอ

สรุปงานวิจัยที่นำเสนอ

ด้วยความเจริญก้าวหน้าของเทคโนโลยี การสื่อสารไร้สาย และ เทคโนโลยี Positioning System ทำให้อุปกรณ์อุปกรณ์สามารถใช้งานอินเทอร์เน็ทได้ทุกที่ทุกเวลา อีกทั้งยังสามารถทราบตำแหน่งพิกัดที่ตั้งของตนได้ทุกที่ทุกเวลาอีกด้วย ซึ่งก็มีหลายๆแอพพลิเคชันที่ต้องการความปลอดภัยในการส่งข้อความสำคัญต่าง ๆ จึงควรที่จะรองรับลายเซ็นดิจิตอลด้วย แต่เนื่องด้วยอุปกรณ์โมบายล์เป็นอุปกรณ์ที่มีความสามารถในการประมวลผลน้อย และ มีพลังงานที่จำกัด การทำการคำนวณหรือประมวลผลที่มีความสลับซับซ้อนสูงบนอุปกรณ์อุปกรณ์โมบายล์จึงเป็นภาระที่หนักเกินไป อาจจะทำให้ต้องใช้เวลาในการประมวลผลนาน และเป็นสาเหตุทำให้แบตเตอร์รี่หมดอย่างรวดเร็ว

งานวิจัยนี้ นำเสนอ Space-Time Based Digital Signature หรือ ลายเซ็นดิจิตอลสำหรับอุปกรณ์โมบายล์โดยใช้ปัจจัยทางภูมิศาสตร์และเวลาเข้ามามีส่วนร่วม ซึ่งใช้หลักการของ RSA และออกแบบมาให้เหมาะสมกับ Mobile Devices เพราะไม่มีการประมวลผลที่ซับซ้อน เช่น modular exponentiation หรือ inverse computation บนฝั่งของอุปกรณ์โมบายล์เลย อีกทั้งงานที่นำเสนอนี้ยังได้นำประโยชน์จากเทคโนโลยี Positioning System มาใช้ ซึ่งเป็นการนำข้อดีของอุปกรณ์โมบายล์นั้นก็คือ "สามารถเคลื่อนที่ได้ ไม่อยู่กับที่ตายตัว" มาใช้ให้เป็นประโยชน์นั้นเอง โดยมีการประยุกต์ใช้ Geo-Encryption และ Mobility Model มามีส่วนร่วมในการสร้างลายเซ็นดิจิตอลนั้นเอง

วิธีการที่นำเสนอมี Sign Server ทำหน้าที่ช่วยอุปกรณ์โมบายล์ในการสร้างลายเซ็นดิจิตอล และยังคอยรับข้อมูลอัพเดทสถานะของอุปกรณ์โมบายล์ เพื่อคำนวณตำแหน่งที่ตั้งโดยประมาณของอุปกรณ์โมบายล์นั้นๆอีกด้วย โดยลายเซ็นดิจิตอลที่ได้จากวิธีที่นำเสนอนี้สามารถรองรับ Authentication, Data Integrity, Non Repudiation Service ซึ่งในที่นี้สนใจที่ Non-Repudiation of Sender (NRS) และ Non-Repudiation of Receiver (NRR) และมีคุณสมบัติ Randomization, Unlinkability, Partial blindness และ Unforgeability

วิธีการที่นำมาใช้ในการสร้างลายเซ็นดิจิตอลมีความปลอดภัยเทียบเท่ากับ RSA ซึ่งเมื่อนำ Space-Time Based เข้ามาเกี่ยวข้องด้วยก็ยิ่งเพิ่มความยากลำบากในการแฮ็คหรือโจมตีระบบของเราเพิ่มขึ้นไปอีก โดยเมื่อเปรียบเทียบกับ "RSA-based partially blind signature" ของ Abe-Fujisaki [11] วิธีการที่นำเสนอสามารถลดเวลาในการประมวลผลทางฝั่งอุปกรณ์โมบายล์ ได้ถึง 97.5 % ที่ modulus n เท่ากับ 1024 bit และอาจสามารถนำไปประยุกต์ใช้แอพพลิเคชันต่างๆ รวมไปถึง Location Based Services ต่างๆ ต่อไปได้ เช่น mobile payment system, e-ticket system และ electronic wallet เป็นต้น

เอกสารอ้างอิง

[1] N. Asokan, G. Tsudik, M.Waidner, **Server-supported signatures**, Journal of Computer Security, Voulme 5, Issue 1, pages 91-108, January 1997.

[2] Xuhua Ding, Daniele Mazzocchi, Gene Tshdik, **Experimenting with Server-Aided Signatures**, In Proceedings of Network and Distributed System Security Symposium (NDSS'2002), San Diego, 2002

[3] Yu Lei; Deren Chen and Zhongding Jiang, **Generating digital signatures on mobile devices**, in Advanced Information Networking and Applications, 2004. AINA 2004. 18th International Conference on Volume 2, 2004 Page(s):532 – 535

[4] L. Scott and D. Denning, "**Geo-encryption: using GPS to enhance data security.**" GPS World, 1 Apr. 2003, http://www.gpsworld.com/gpsworld/article/articleDetail.jsp ?id=57975

[5] Ala Al-Fuqaha and Omar Al-Ibrahim and Joe Baird, **A Mobility Model for GPS-Based Encryption**, in Global Telecommunications Conference, 2005. GLOBECOM '05, 2005 Volume: 3, On page(s): 1721-1725

[6] Hung-Yu Chien and Jinn-Ke Jan and Yuh-Min Tseng, **RSA-based partially blind signature with low computation**, in Proceedings of International Conference on Parallel and Distributed Systems (ICPADS'2001), 2001

[7] S. Boeyen, T. Hows, P. Richard, **Inernet x.509 public key infrastructure operational protocols-LDAPv2,** RFC 2559, 1999

[8] N. Ferguson, **Sigle term off-line coins,** Advances in Cryptology-EUROCRYPT'93, LNCS 765, Springer Verlag, On page(s): 318-328, 1993

[9] A. Shamir, and C. P. Schnorr, **Cryptanalysis of certain variants of Rabin' signature scheme**, Information Processing Letters, vol. 19, On page(s): 113-115, 1984

[10] C-I. Fan, and C.-L. Lei, **Low-computation partially blind signatures for electronic cash**, IEICE Trans. Fundamentals, vol. E-81-A, no. 5 , On page(s):818-824, May 1998

[11] M. Abe and E. Fujisaki, **How to date blind signatures**, Advances in Cryptology-ASIACRYPT'96, LNCS 1163, 1996, pp. 224-251

[12] D. Chaum, **Blind signature systems**, Advances in Cryptilogy-CRYPTO'83, Plenum, On page(s): 153, 1983

[13] M. O. Rabin, **Digitalized signatures and public-key functions as intractable as factorization**, Technical Report, MIT/LCS/TR212, MIT Lab., Computer Science, Cambridge, Mass., Jan. 1979

[14] Public Key Cryptography Standards (PKCS), No.1, **RSA Encryption standard**, http://www.rsasecurity.com/rsalabs/pkcs

[15] National Institute for Standards and Technology, **Digital Signature Standard (DSS)**, Technical Report 169, August 30, 1991

[16] C.Y. Chen, C.C. Chang and W.P. Yang, Hybrid method for modular exponentiation with precomputation, Electron. Lett., vol. 32, no.6, On page(s): 540-541, 1996

[17] G. J. Simmons, **Contemporary Cryptology: The Science of Information Integrity**, IEEE Press, N.Y., 1992

[18] S. M. Ghanem, H. A. Wahab, **A simple XOR-based technique for distributing group key in secure multicasting**, Proceedings for the Fifth IEEE Symposium on computers and Communications, ISCC 2000, 3-6 July, 2000, On page(s): 166-171

ภาคผนวก

ภาคผนวก ก.
ผลงานวิจัยในระหว่างการศึกษาที่ได้รับการตีพิมพ์เผยแพร่

[1] S. Jarusombat and S Kittitornkun, "Digital Signature on Mobile Devices based on Location", International Symposium on Communications and Information Techonologies, 2006 (ISCIT 2006), Bangkok, Thailand

[2] S. Jarusombat and S Kittitornkun, "Location Based Digital Signature on Mobile Devices", 29th Electrical Engineering Conference (EECON #29), 2006, Pattaya, Thailand

Introduction
Organization
Important Dates
Call for Papers
Paper Submission
Technical Program
Exhibition
Registration
Accommodation
Sponsors
Links
General Information
Contact Us

EECON29 WEBBOARD

ประกาศรางวัล "ผู้นำเสนอบทความดี"

เป็นรางวัลที่มอบให้แก่นักศึกษาที่ทำการนำเสนอบทความได้ดีในการประชุม
โดยเจ้าภาพจะส่งรางวัลให้แก่ สถาบัน

ตรวจดูรายชื่อได้ ที่นี่

ภาพบรรยากาศงาน EECON29

ณ โรงแรมแอมบาสเดอร์ ซิตี้ จอม เทียนพัทยา UPDATE!!

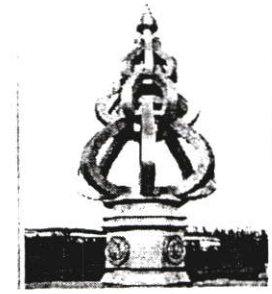
EECON29
Important Dates

Full Paper Submission Date
July 10, 2006

Acceptance Notification
September 4, 2006

Manuscript Due
September 21, 2006

Conference Date
November 9-10, 2006



ลายเซ็นต์ดิจิตอลสำหรับอุปกรณ์โมบายโดยใช้ปัจจัยทางภูมิศาสตร์

Location Based Digital Signature on Mobile Devices

สันติ จารุสมบัติ และ สุรินทร์ กิตติธรกุล
ภาควิชาวิศวกรรมไฟฟ้า สาขาคอมพิวเตอร์ คณะวิศวกรรมศาสตร์,
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,
กรุงเทพมหานคร 10520 ประเทศไทย
Onepiece_com@hotmail.com, kksurin@kmitl.ac.th

บทคัดย่อ

เนื่องด้วยความเจริญก้าวหน้าอย่างรวดเร็วของเทคโนโลยีการสื่อสารไร้สาย และ Mobile Devices ทำให้อุปกรณ์เหล่านี้ได้กลายมาเป็นสิ่งจำเป็นในชีวิตประจำวัน ปัจจุบันมีการเอาเทคโนโลยี GPS (Global Positioning Service) รวมเข้าไว้กับอุปกรณ์เหล่านี้แล้ว ได้มีการใช้งาน Internet ในหลายๆรูปแบบผ่าน Mobile Devices เพื่อความปลอดภัยของข้อมูลแอพพลิเคชันเหล่านั้นควรรองรับการทำงานของ Digital Signature เข้าไปด้วย Mobile Devices เป็นอุปกรณ์ที่มีหน่วยประมวลผลต่ำ และแบตเตอรี่ที่จำกัด จึงไม่เหมาะสมกับ Digital Signature ที่ใช้วิธีอัลกอริทึมแบบ Asymmetric Cryptographic อีกทั้งยังไม่ได้ใช้ประโยชน์จากเทคโนโลยี GPS ที่มีอยู่อีกด้วย บทความนี้จึงขอเสนอ "Location Based Digital Signature on Mobile Devices" โดยนำหลักการของ Geo-Encryption มาประยุกต์ใช้ในการสร้าง Digital Signature ด้วย โดยทั้งนี้ยังเป็นการลดภาระในส่วนของ Client, โดยเปรียบเทียบกับการสร้าง RSA-based partially blind signature ของ Abe-Fujisaki สามารถลดเวลาในการประมวลผลได้เกือบ 98 % และลดการสื่อสารกันระหว่างผู้ส่งและผู้รับ

คำสำคัญ : ลายเซ็นต์ดิจิตอล, โมบาย, ปัจจัยทางภูมิศาสตร์

Abstract

With rapid growing of wireless and mobile technologies, mobile devices are part and parcel of our lives. Nowadays GPS (Global Positioning Service) technology is combined with mobile devices. People use some Internet applications, e.g. mobile fund transfer, e-commerce, game online and theatre booking. These applications should support Digital Signature Service for more security. Because mobile device is low-computation and limited battery life, it is not suitable to use digital signature protocols, which are based on asymmetric cryptographic algorithm, with mobile devices. And it does not use advantage of GPS technology. We present "Location Based Digital Signature on Mobile Devices" with Geo-Encryption to generate Digital Signature. This method reduces the burden on mobile devices, the computation time of the requester by almost 98%, compared with Abe-Fujisaki's RSA-based partially blind signature and decreases communication between sender and receiver.

1. คำนำ

ปัจจุบันเทคโนโลยี Internet ได้มีการใช้งานกันอย่างกว้างขวาง บวกกับความเจริญก้าวหน้าของ Wireless Communication และ Mobile Devices ทำให้เราสามารถใช้ Mobile Devices เข้าถึง Internet ได้ไม่ว่าที่ไหนหรือ เมื่อไหร่ แล้วก็มีหลายแอพพลิเคชันที่ต้องการความปลอดภัยในการติดต่อสื่อสาร เช่น ระบบ m-commerce

Digital Signature ก็เหมือนกับลายเซ็นปกติ ที่สามารถตรวจสอบว่า ผู้เซ็นคนไหนเป็นคนเซ็นลงบนเอกสารนั้น และตรวจสอบได้อีกว่า เอกสารนั้นเหมือนกับเอกสารตอนที่ผู้เซ็นได้เซ็นไว้หรือไม่ โดย Digital Signature นั้นสามารถสร้างความปลอดภัยให้กับแอพพลิเคชันต่างๆได้ โดยปกติจะใช้เทคนิค Asymmetric Cryptographic ซึ่งเป็นการทำงานที่หนักสำหรับ Mobile Devices อาจทำให้ใช้เวลานาน และกินแบตเตอรี่หมดอย่างรวดเร็ว

โดยมีหัวข้อดังต่อไปนี้ ในหัวข้อที่ 2 นำเสนองานวิจัยที่เกี่ยวข้อง หัวข้อที่ 3 นำเสนอรายละเอียดของวิธีในการสร้าง Location Based Digital Signature ของเรา ในหัวข้อต่อมา จะวิเคราะห์ประสิทธิภาพ และความปลอดภัยของโมเดลที่นำเสนอ และหัวข้อสุดท้ายคือ สรุป

2. งานวิจัยที่เกี่ยวข้อง

มีหลายๆบทความที่กล่าวถึง การสร้าง Digital Signatures สำหรับ Mobile Devices หนึ่งในนั้นคือ [1] มี Server ช่วยในการสร้าง Non-Repudiation Digital Signature โดยนำวิธีการสร้าง Digital Signature จาก [4] มาใช้ทำให้สามารถหลีกเลี่ยงการคำนวณที่หนักๆในส่วนของการตรวจสอบความถูกต้อง Digital Signatures ไปได้ และยังได้ความปลอดภัยระดับเดียวกับการสร้าง Digital Signatures แบบปกติ อีกด้วย

ในบทความ [3] เสนอโมเดลที่นำเอาตำแหน่ง และ เวลามามีส่วนร่วมในการเข้าและถอดรหัสเพื่อเพิ่มความปลอดภัยในการส่งข้อมูล โดยเข้ารหัสข้อความเพื่อให้สามารถถอดรหัสได้ เฉพาะเมื่อผู้รับอยู่ในตำแหน่งและเวลาที่กำหนดเท่านั้น ฟังก์ชัน Geo-Locking ทำหน้าที่ในการรวมตำแหน่งที่ตั้ง (L) กับ Encryption Key เพื่อให้ได้ Geo-Secured Key ส่งไปพร้อมกับข้อความที่เข้ารหัสแล้ว โดยฟังก์ชัน Geo-Locking จะเรียกใช้ฟังก์ชัน PVT-to-GeoLock Mapping (Mapping Function) [3] ฟังก์ชันนี้จะรับอินพุทเป็น Latitude, Longitude และ time เพื่อสร้างค่า Geo-Lock ใช้ในการสร้าง Geo-Secured Key และถอดรหัสจาก Geo-Secured Key เป็น Session Key ในฝั่งของผู้รับ โดยโมเดล Geo-Encryption จะมีประสิทธิภาพก็ต่อเมื่อ ผู้ส่ง

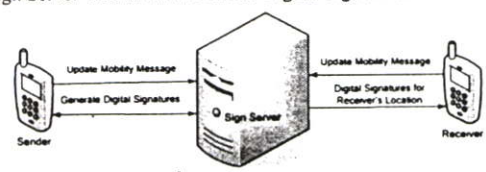
CP10

สามารถรู้ตำแหน่งที่ตั้งของผู้รับและเวลาที่ผู้รับจะไปอยู่ในตำแหน่งนั้น ในบท ความ [2] เสนอ Mobility Model เพื่อให้สามารถใช้งานร่วมกับ Geo-Encryption ได้โดยให้ Mobile Devices ทำการแลกเปลี่ยนพารามิเตอร์ในการเคลื่อนที่ต่างๆกันได้ ทำให้ผู้ส่งสามารถทำ Geo-Encrypt ข้อความส่งไปยัง Decryption Zone ที่ประมาณว่าผู้รับจะอยู่ได้ และนำเสนอวิธีการคำนวณค่าพารามิเตอร์ในการเคลื่อนที่ไว้อีกด้วย

จากที่ได้กล่าวมาข้างต้น คงจะเป็นการดี ถ้านำประโยชน์จากเทคโนโลยี GPS มาใช้ในการสร้าง Digital Signatures จึงเสนอ "Location Based Digital Signature on Mobile Devices" เป็นการประยุกต์ใช้ Geo-Encryption เพื่อเพิ่มระดับความปลอดภัยในการสร้าง Digital Signatures ที่เหมาะสมสำหรับ Mobile Devices ให้สามารถสร้าง Digital Signatures ที่มีระดับความปลอดภัยเทียบเท่ากับแบบปกติได้ อีกทั้งยังนำระบบ Location Based มาใช้ให้เป็นประโยชน์อีกด้วย

3. โมเดลที่นำเสนอ

บทความนี้ ตั้งอยู่บนพื้นฐานของ [1], [2] และ [3] โดยที่ทั้งผู้ส่ง และ ผู้รับนั้น เป็น Mobile Device และสามารถส่งที่ตั้งไปยังฝ่ายตรงข้ามได้อย่างปลอดภัยเมื่อจำเป็น โดยนำ Mobility Model ใน [2] มาประยุกต์ใช้กับโมเดลของเราเพื่ออัพเดทพารามิเตอร์ในการเคลื่อน ที่ไปยัง Sign Server ซึ่งทำหน้าที่ช่วยสร้าง Digital Signatures นั้นเอง



รูปที่ 1 ภาพรวมของโมเดล

3.1 การสร้าง Digital Signature

ด้วยวิธีการสร้าง Digital Signature สำหรับ Mobile Devices ใน [1] ซึ่งรองรับ Authentication, Data Integrity และ Non-Repudiation Services ได้ (สนใจที่ Non-Repudiation of Sender (NRS) และ Non-Repudiation of Receiver (NRR)) โดยมี Sign Server ช่วย Mobile Devices สร้าง Digital Signatures อีกทั้งยังคอยรับข้อมูลอัพเดทสถานะการเคลื่อนที่ของ Mobile Devices เพื่อให้สามารถประมาณตำแหน่งของ Mobile Devices ในเวลา t ใดๆ ได้

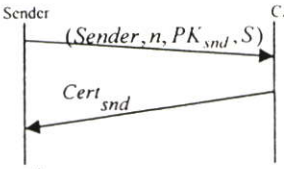
โดยมีการใช้ One-way Collision-resistant Hash Function เช่น SHA1 หรือ MD5 เป็นต้น เพื่อใช้ในการสร้าง Public Keys ของทั้งผู้ส่งและผู้รับ ซึ่งกำหนดสัญลักษณ์ให้ $h^i()$ หมายถึง Hash ทั้งหมด i ครั้ง โดยผู้ใช้ต้องสร้าง Secret Key K_u โดย u หมายถึง user ด้วยการใช้ K_u เป็นอินพุทของ Hash Function แต่ละผู้ใช้มี Hash Chain $K_u^1, K_u^2, K_u^3, \ldots, K_u^n$ ซึ่ง $K_u^0 = K_u$ และ $K_u^i = h^i(K_u) = h(K_u^{i-1})$ และให้ $PK_u = K_u^n$ เป็น Public Key ของผู้ใช้ u ถ้าผู้ส่งต้องการสร้าง Signature สำหรับข้อความ a เพื่อส่งไปยังผู้รับต้องทำตามขั้นตอนนี้

ขั้นตอนที่ 1

ผู้ส่งสร้าง PK_{snd} ของตนขึ้นมา แล้วเลือก Sign Server (S) ที่ต้องการให้ช่วยสร้าง Signatures เสร็จแล้วจึงส่งค่า Identify ของผู้ส่ง, จำนวน Signatures สูงสุดที่ต้องการสร้าง (n), PK_{snd} และ S ไปยัง Certification Authority (CA) เพื่อให้ CA สร้าง Certificate ($Cert_{snd}$)

$$Cert_{snd} = SK_{CA}(Sender, n, PK_{snd}, S)$$

โดย SK_{CA} หมายถึง Secret Key ของ CA หลังจากที่สร้าง $Cert_{snd}$ แล้ว จึงประกาศผ่าน Directory Services เช่น LDAP



รูปที่ 2 กระบวนการสร้าง Certificate

ขั้นตอนที่ 2

Sign Server ต้องสร้างค่าจำนวนเฉพาะขนาดใหญ่ขึ้นมา 2 ค่า คือ p กับ q และให้ $m = p \times q$ และ $\phi(m) = (p-1) \times (q-1)$ และหาค่า d โดยที่ค่าของ d ต้องเป็นไปตามสมการ $e \times d = 1 \bmod \phi(m)$ ซึ่ง $e = 3$ หลังจากนั้นให้ S เก็บ (d, p, q) ไว้เป็นความลับแล้วประกาศ (e, m) ไป

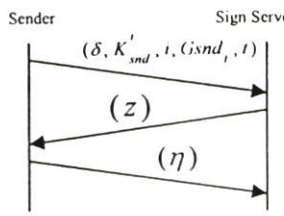
ขั้นตอนที 3

ถ้าผู้ส่งต้องการสร้าง Signature สำหรับข้อความ a จะต้องเลือกตัวเลขแบบสุ่มขึ้นมา 2 ค่าคือ r, w แล้วจึงคำนวณค่า δ จาก (1)

$$\delta = r^e h(a)(w^2 + 1) \bmod m \qquad (1)$$

แล้วอ่านค่า Latitude, Longitude และเวลา t (X_t, Y_t, t) จาก GPS เพื่อเอามาใส่ Position-Velocity-Time (PVT) to GeoLock Mapping Function [3] เพื่อให้ได้ค่า GeoLock ของผู้ส่งที่เวลา t ($Gsnd_t$) มาแล้วส่ง $(\delta, K'_{snd}, i, Gsnd_t, t)$ ไปยัง S เมื่อ S ได้รับข้อความแล้วก็จะคำนวณหาค่า $(\hat{X}_t, \hat{Y}_t)$ ที่เวลา t จาก (9) เพื่อเอาค่ามาใส่ใน Mapping Function ให้ได้ผลลัพธ์ $\hat{G}snd_t$ (อธิบายในหัวข้อถัดไป) แล้วเปรียบเทียบค่า $Gsnd_t$ กับ $\hat{G}snd_t$ ว่ามีค่าเท่ากันหรือไม่ ถ้าผลที่ได้ออกมาไม่เท่ากัน S จะไม่สร้าง Signature ให้กับผู้ส่ง แล้วส่ง (Ack_1) กลับไปยังผู้ส่ง เมื่อผู้ส่งได้รับข้อความ (Ack_1) ให้คำนวณค่า $Gsnd_t$ ในขณะนั้นใหม่แล้วจึงส่งข้อความ $(\delta, K'_{snd}, i, Gsnd_t, t)$ ใหม่ไปยัง S อีกครั้ง

แต่ในกรณีที่ผลลัพธ์ออกมาเหมือนกัน S จะเลือกค่าตัวเลขแบบสุ่มขึ้นมาคือ z โดย $z < m$ แล้วส่งค่า (z) ไปยังผู้ส่ง เมื่อผู้ส่งได้รับก็จะเลือกตัวเลขแบบสุ่มขึ้นมาอีกตัวคือ r' แล้วคำนวณค่า b จาก (2) หลังจากนั้นผู้ส่งคำนวณค่า η จาก (3) แล้วส่งค่า (η) ไปยัง S อีกทีหนึ่ง



CP10

รูปที่ 3 กระบวนคำนวณ z และ η โดยมีการตรวจสอบ $Gsnd_t$ ก่อน

$$b = r \times r' \tag{2}$$

$$\eta = b^e \times (w - z) \tag{3}$$

ขั้นตอนที่ 4

เมื่อ S ได้รับ η จะคำนวณค่า γ จาก (4) และประมาณตำแหน่งของผู้รับ ณ ขณะนั้น t จากข้อมูลอัพเดทสถานะการเคลื่อนที่ ที่ผู้รับส่งมาอัพเดทตาม (9) หลังจากได้ค่า $(\hat{X}_t, \hat{Y}_t)$ ของผู้รับ แล้วก็นำไปใส่ใน Mapping Function เพื่อให้ได้ค่า $\hat{Grcv}_t$ มาแล้วนำมาคำนวณหาค่า α เพื่อส่ง $(\gamma, \alpha, \hat{Grcv}_t)$ ไปยังผู้ส่ง โดยค่า α คำนวณได้จากสมการ (5) โดยเมื่อผู้ส่งได้รับข้อความแล้วจะคำนวณค่า c, s ตาม (6) และ (7)

$$\gamma = \eta^{-1} \bmod m \tag{4}$$

$$\alpha = h(Cert_{snd} \oplus Grcv_t)^d (\delta(z^2+1)\eta^{-2})^{2d} \bmod m \tag{5}$$

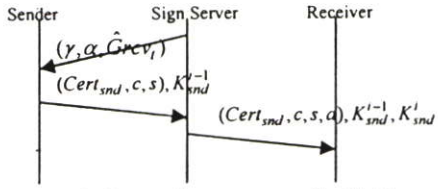
$$c = (wz+1) \times \gamma \times b^e = (wz+1)(w-z)^{-1} \bmod m \tag{6}$$

$$s = \alpha \times r^2 \times r'^4 \bmod m \tag{7}$$

โดย $(Cert_{snd}, c, s)$ เป็น Digital Signature ของข้อความ a ซึ่งสามารถตรวจสอบความถูกต้องได้ตามวิธีใน [4] จากสมการ (8)

$$s^e = h(Cert_{snd} \oplus Grcv_t)h(a)^2(c^2+1)^2 \bmod m \tag{8}$$

หลังจากผู้ส่งตรวจสอบความถูกต้องของ Signature ตาม (8) แล้วจึงส่ง Signature ที่สร้างขึ้นมานี้กับ K_{snd}^{t-1} ไปยัง S เมื่อ S ได้รับข้อความ $(Cert_{snd}, c, s), K_{snd}^{t-1}$ ก็จะตรวจสอบความถูกต้องของ Signature ตาม (8) และตรวจสอบว่า $K_{snd}^t = h(K_{snd}^{t-1})$ หรือไม่ ถ้าผลออกมาถูกต้องจึงส่ง $(Cert_{snd}, c, s, a), K_{snd}^{t-1}, K_{snd}^t$ ไปยังผู้รับ

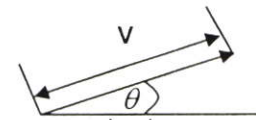


รูปที่ 4 Sign Server ช่วยในการสร้าง Signature แล้วส่งไปยัง Receiver

ตามขั้นตอนที่ผ่านมาเป็นอันเสร็จสิ้นการสร้าง Digital Signature ซึ่งสามารถรองรับ NRS ถ้าต้องการจะให้รองรับ NRR ตามในขั้นตอนที่ 1-4 ผู้รับก็สามารถสร้าง $(Cert_{rcv}, c', s')$ ได้เช่นกันแล้วส่ง $(Cert_{rcv}, c', s'), K_{rcv}^J$ ไปยัง S แล้วให้ S ส่งต่อไปยังผู้ส่งอีกทีหนึ่ง ก่อนที่ผู้ส่งจะส่งค่า K_{snd}^{t-1} ไปยัง S โดยเมื่อผู้ส่งได้แล้วผู้ส่งจึงส่ง $(Cert_{snd}, c, s), K_{snd}^{t-1}$ ไปยัง S หลังจากที่ได้มีการตรวจสอบ $(Cert_{rcv}, c', s')$ แล้ว หลังจากที่ Sign Server ได้รับทั้ง K_{snd}^{t-1} และ K_{rcv}^{J-1} (ส่งโดยผู้รับ) แล้ว จึงส่ง $(Cert_{snd}, c, s, a), K_{snd}^{t-1}, K_{snd}^t$ ไปยังผู้รับ และส่ง K_{rcv}^J ไปยังผู้ส่ง เป็นอันเสร็จสิ้นการสร้าง Location Based Digital Signature สำหรับอุปกรณ์ Mobile Devices

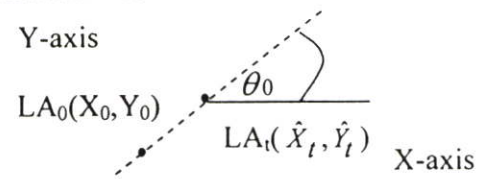
3.2 พารามิเตอร์ในการเคลื่อนที่ และ การอัพเดท

โดยในรูปที่ 5 แสดงพารามิเตอร์ในการเคลื่อนที่ของ Mobile Devices ที่ใช้ในการเป็นข้อมูลเพื่อส่งไปยัง S ซึ่ง V คือความเร็วเฉลี่ยของ Mobile Devices ในขณะนั้นๆ สามารถคำนวณได้จากระยะทางที่เคลื่อนที่ได้ในหนึ่งช่วงเวลา และ θ เป็นองศาที่มีค่าเป็นบวกระหว่าง ทิศทางของความเร็ว กับ แนวแกน x-axis ในระบบ Cartesian coordinate ซึ่งทั้ง θ และ V นี้ Mobile Devices จะเป็นตัวคำนวณเองโดยอัตโนมัติ ไม่ได้มีการกำหนดค่าจากผู้ใช้



รูปที่ 5 พารามิเตอร์ในการเคลื่อนที่ ความเร็ว(V) และ ทิศทาง (θ)

ผู้รับและผู้ส่งต้องอัพเดทข้อมูลการเคลื่อนที่ไปยัง Sign Server จากข้อมูลที่ส่งมาทำให้ Sign Server ประมาณตำแหน่งของ Mobile Devices ได้ ให้ A เป็น Mobile Device ตำแหน่งของ A ที่เป็น Longitude และ Latitude ที่เวลา t_0 แสดงด้วยสัญลักษณ์ $LA_0(X_0, Y_0)$ โดย Mobile Devices ต้องอ่านค่าตำแหน่งของตนผ่าน GPS Reader เป็น $LA_t(X_t, Y_t)$ ที่เวลา t ซึ่ง $t = t_0, t_1, t_2, \ldots$ ให้ $t_i = t_0 + i\omega$ เมื่อ $i = 1, 2, 3, \ldots$ และ ω คือค่าคงที่ไว้เพื่อกำหนดความกว้างของช่วงเวลา โดยทั้งผู้ส่งและผู้รับต้องเริ่มส่ง Mobility Message ให้ Sign Server ตอนที่ทั้งผู้ส่งและผู้รับเริ่มติดต่อกัน แล้วส่งข้อมูลอัพเดทไปเรื่อยๆ ทิศทางในการเคลื่อนที่ของ Mobile Device จะทำมุม θ กับแนวเส้น Latitude ที่เวลา t ใดๆ Sign Server สามารถที่จะประมาณตำแหน่งของ Mobile Devices ได้ ดังรูปที่ 6



รูปที่ 6 แสดงตำแหน่งที่ประมาณขึ้นจาก ค่า V_t, θ, LA_t

ด้วยข้อมูลที่ส่งมานั้น Sign Server สามารถประมาณตำแหน่งของ Mobile Devices ในเวลาที่ t ได้จากสมการ (9)

$$\begin{aligned} \hat{X}_t &= X_0 + (t - t_0)V\cos\theta \\ \hat{Y}_t &= Y_0 + (t - t_0)V\sin\theta \end{aligned} \tag{9}$$

และค่า V และ θ นั้น Mobile Devices คำนวณเองโดยอัตโนมัติ ซึ่งความเร็วและทิศทางของ Mobile Devices ที่เวลา t สามารถหาได้จากสมการ (10)

$$\begin{aligned} V_t &= \sqrt{\left(\frac{X_t - X_0}{t - t_0}\right)^2 + \left(\frac{Y_t - Y_0}{t - t_0}\right)^2} \\ \theta_t &= \arctan\left(\frac{Y_t - Y_0}{X_t - X_0}\right) \end{aligned} \tag{10}$$

โดยทุกครั้งที่ Mobile Devices อ่านค่า Latitude, Longitude จาก GPS เพื่อเอามาคำนวณค่าความเร็ว (V) และ ทิศทาง (θ) นั้น Mobile Devices ต้องตัดสินใจว่าจะอัพเดทข้อมูลที่ได้มาล่าสุดหรือไม่ โดยการเปรียบเทียบค่า V_t, θ_t ณ เวลา t กับค่าเก่า V_0, θ_0 ว่าค่ามีการเปลี่ยนแปลง

CP10

เกินค่า threshold ที่กำหนดไว้หรือไม่ ถ้าไม่เกินก็จะไม่มีการส่งอัพเดทแต่อย่างใด แต่ถ้ามีการเปลี่ยนแปลงเกินค่า threshold ที่กำหนดก็จะอัพเดทข้อมูลการเคลื่อนที่ไปยัง Sign Server แล้วเก็บค่า V_t, θ_t ล่าสุดไว้แทนที่ค่าเก่า V_0, θ_0 แทนที่ค่า t_0 ด้วยค่า t แทนค่า X_0, Y_0 ด้วยค่า X_t, Y_t ตามลำดับ โดย Update Message คือ

$$\{V_t, \theta_t, LA_t, t\}$$

ยิ่งค่า threshold มีขนาดเล็กจะทำให้มีการอัพเดทบ่อยครั้งขึ้น ซึ่งการเลือกค่า threshold ที่เหมาะสมนั้นขึ้นอยู่กับความสมดุลกันระหว่างความน่าจะเป็นที่ค่า Geo-Lock ที่ได้มาจากตำแหน่งของ Mobile Devices ที่ S คำนวณนั้น มีค่าเท่ากับค่า Geo-Lock ที่ได้จากการอ่านค่า GPS ของ Mobile Devices กับ ความถี่ในการส่งข้อมูลอัพเดทสถานะการเคลื่อนที่

4. วิเคราะห์ประสิทธิภาพ

4.1 Location Based Analysis

โมเดลนี้มี Sign Server ทำหน้าที่ช่วยในการสร้าง Signature และรับข้อมูลอัพเดทสถานะการเคลื่อนที่ของ Mobile Devices โดยทั้งผู้ส่งและผู้รับต้องส่งข้อมูลอัพเดทมายัง S เมื่อมีการเริ่มติดต่อกัน โดย S จะไม่สร้าง Signature ให้ ถ้า $Gsnd$, ที่ผู้ส่งส่งมาให้มีค่าไม่เท่ากับ $\hat{G}snd$, ที่ S คำนวณขึ้นมาและผู้รับจะไม่สามารถตรวจสอบความถูกต้องได้ ถ้าไม่อยู่ในบริเวณและเวลาที่กำหนดไว้ ซึ่งวิธีที่ใช้ในการสร้าง Signature นั้นมีระดับความปลอดภัยเหมือนกับการสร้าง Signature แบบปกติ และสามารถลดเวลาในการประมวลผลของฝั่งผู้ส่งลงถึง 98% [4] เมื่อเทียบกับโมเดลของ Abe-Fujisaki "RSA-based partially blind signature" [5] ถ้าผู้ประสงค์ร้ายสามารถที่จะโจมตีวิธีในการสร้าง Signature ของเราได้ แสดงว่าเขาสามารถที่จะแฮ็คอัลกอริทึม SHA หรือ MD5 และ RSA ได้ โดยเมื่อมีการเอา Location Based เข้ามามีส่วนร่วม จึงเป็นการเพิ่มระดับความปลอดภัยให้กับการสร้าง Digital Signature ให้มากขึ้นนั่นเอง เพื่อนำไปประยุกต์ใช้กับแอพพลิเคชันประเภท e-commerce ต่าง ๆ ต่อไป

4.2 Repudiation Analysis

ในกรณีที่ผู้ส่งอ้างว่าไม่ได้มีการส่งข้อความ (NRS) ผู้รับสามารถส่ง $(Cert_{snd}, c, s, a), K_{snd}^{t-1}, K_{snd}^{t}, Grcv_t$ ไปให้ผู้ตรวจสอบตรวจสอบ โดยเริ่มต้นจากการตรวจสอบ $Cert_{snd}$ ซึ่งสร้างโดย CA ซึ่งประกอบไปด้วย Identify ของผู้รับ, Public Key และ Identify ของ S เป็นต้น หลังจากนั้นก็ตรวจสอบ Signature จาก (8) หลังจากนั้นก็ตรวจสอบดูว่า $PK_u = K_u^n$ สามารถหาได้โดยการใช้ Hash Function กับ $K_{snd}^{t-1}, K_{snd}^{t}$ หรือไม่ ถ้าทุกอย่างถูกต้อง ผู้ตรวจสอบสามารถตัดสินได้เลยว่า ผู้ส่งได้ส่งข้อความนี้ไปยังผู้รับจริง ซึ่งในกรณี NRS ก็ใช้วิธีเดียวกับการตรวจสอบ ในกรณีที่ผู้ส่งอ้างเช่นกัน

4.3 Security Analysis

ในกรณีนี้ที่ผู้ประสงค์ร้ายต้องการสร้าง Signature ปลอม เพื่อให้สามารถผ่านสมการ (8) ผู้บุกรุกต้องสร้างค่า s จาก (11) ซึ่งถ้าผู้บุกรุกไม่สามารถรู้ตำแหน่งของผู้รับในขณะนั้น ก็จะไม่สามารถสร้าง Signature ได้ อีกทั้งถึงแม้ว่าผู้บุกรุกจะรู้ค่า $h(Cert_{snd} \oplus Grcv_t)$ และ $h(a), c$ ก็แทบจะเป็นไปไม่ได้ที่ผู้บุกรุกจะสามารถคำนวณหาค่า d จาก e, m โดยที่ไม่ทราบค่า p, q หรือถ้าในกรณีที่ผู้บุกรุกรู้ค่า $s, h(Cert_{snd} \oplus Grcv_t), h(a)$ มันก็เป็นการยากที่จะคำนวณค่า c (12) จากสมการ โดยที่ไม่รู้ค่า p, q อีกเช่นกัน

$$s \equiv h(Cert_{snd} \oplus Grcv_t)^d \times h(a)^{2d} \times (c^2+1)^{2d} \bmod m \qquad (11)$$

$$c^2 \equiv \left(s^e \times h(Cert_{snd} \oplus Grcv_t)^{-1} \times h(a)^{-3}\right)^{1/2} - 1 \bmod m \qquad (12)$$

5. สรุป

ในบทความนี้ เสนอ Location Based Digital Signature สำหรับ Mobile Devices ซึ่งเป็นอุปกรณ์ที่ความความสามารถในการประมวลผลต่ำ และมีแบตเตอรีที่จำกัด จึงไม่เหมาะสมกับการสร้าง Digital Signature แบบธรรมดา โดยบทความนี้นำประโยชน์จากเทคโนโลยี GPS มาใช้ในการสร้าง Digital Signatures ประยุกต์ใช้ Geo-Encryption และ Mobility Model เพื่อเพิ่มความปลอดภัยในการสร้าง Signatures ที่เหมาะสมสำหรับ Mobile Devices สามารถสร้าง Signatures ที่มีระดับความปลอดภัยเทียบเท่ากับแบบปกติได้ และลดเวลาในการประมวลผลของฝั่งผู้ส่งลงประมาณ 98% [4] เมื่อเทียบกับ "RSA-based partially blind signature" ของ Abe-Fujisaki [5] อีกทั้งยังนำระบบ Location Based มาใช้ให้เป็นประโยชน์อีกด้วย เพื่อนำไปใช้ในชีวิตประจำวันต่อไป เช่น M-Commerce เป็นต้น

6. เอกสารอ้างอิง

[1] Yu Lei; Deren Chen and Zhongding Jiang, "Generating digital signatures on mobile devices", in *Advanced Information Networking and Applications 2004*. 2004 Page(s):532 - 535

[2] Ala Al-Fuqaha and Omar Al-Ibrahim and Joe Baird, "A Mobility Model for GPS-Based Encryption", in *Global Telecommunications Conference, 2005. GLOBECOM '05*. 2005 Volume: 3, On page(s): 5

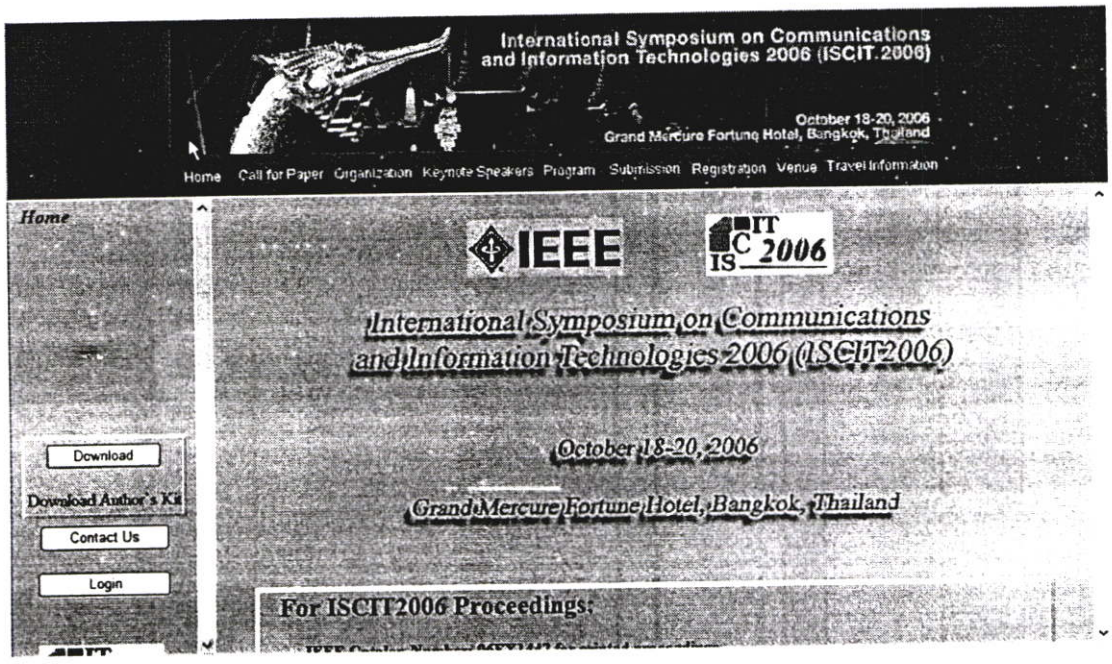
[3] L. Scott and D. Denning, "Geo-encryption: using GPS to enhance data security," GPS World, 1 Apr. 2003. http://www.gpsworld.com

[4] Hung-Yu Chien and Jinn-Ke Jan and Yuh-Min Tseng, "RSA-based partially blind signature with low computation" in *Proceedings of International Conference on Parallel and Distributed Systems (ICPADS'2001)*, 2001

[5] M. Abe and E. Fujisaki, How to date blind signatures, *Advances in Cryptology-ASIACRYPT'96, LNCS 1163*, 1996, pp. 224-251

นายสันติ จารุสมบัติ จบการศึกษาระดับปริญญาตรีวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปัจจุบันกำลังศึกษาต่อในระดับปริญญาโท ในคณะวิศวกรรมศาสตร์ สาขาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง สนใจในงานวิจัยด้าน Security on Mobile, Cryptography, Mobility Model

CP10


International Symposium on Communications and Information Technologies 2006 (ISCIT 2006)
October 18-20, 2006
Grand Mercure Fortune Hotel, Bangkok, Thailand
Home
Call for Paper
Organization
Keynote Speakers
Program
Submission
Registration
Venue
Travel Information
Home
Download
Download Author's Kit
Contact Us
Login
IEEE
ISCIT 2006
International Symposium on Communications and Information Technologies 2006 (ISCIT 2006)
October 18-20, 2006
Grand Mercure Fortune Hotel, Bangkok, Thailand
For ISCIT 2006 Proceedings:

Digital Signature on Mobile Devices based on Location

Santi Jarusombat and Surin Kittitornkun
Dept. of Computer Engineering,
Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang,
Bangkok, 10520 Thailand
Email: Onepiece_com@hotmail.com, kksurin@kmitl.ac.th

Abstract-- **With rapid growing of wireless and mobile technologies, mobile devices are part and parcel of our lives. Nowadays GPS (Global Positioning Service) technology is combined with mobile devices. People use some Internet applications, e.g. mobile fund transfer, e-commerce, game online and theatre booking. These applications should support Digital Signature Service for more security. Because mobile device is low-computation capability and limited battery life, it is not suitable to use digital signature protocols, which are based on asymmetric cryptographic algorithm, with mobile devices. And it does not use advantage of GPS technology. We present "Location Based Digital Signature on Mobile Devices" with Geo-Encryption to generate Digital Signature. This method reduces the burden on mobile devices, the computation time of the requester by almost 98%, compared with Abe-Fujisaki's RSA-based partially blind signature and decreases communication between sender and receiver.**

Keyword: digital signature, mobile, location-based, security, m-commerce.

I. INTRODUCTION

Nowadays Internet technology is used for many activities. With rapid growing of wireless and mobile technologies, we can use mobile devices to access Internet whenever and wherever we are. Many applications, such as m-commerce system and e-ticket system or electronic wallet, need security for their communication.

People create digital signature to act much as physical signature does on a written document. Digital signature can help them prove the document source and confirm integrity of transmitted message. The purposes of employing the digital signatures are: authenticating the validity of users, ensuring the integrity of the message and non-repudiation services.

Conventional digital signature schemes were based on asymmetric cryptographic algorithm. It is not suitable for mobile devices with low-computing capability and short battery life. If mobile device uses those schemes, it would be blocked for a period of time and drain batteries quickly.

The remaining sections of this paper are arranged as follows: Section 2 introduces the related work about digital signature and geo-encryption. Section 3 presents our location-based digital signature on mobile devices. Section 4 analyzes the security of our scheme. Conclusion is drawn in the last section.

II. LITERATURE REVIEWS

There are some works on digital signature on mobile devices. One of them is [1] "Generating Digital Signatures on Mobile Devices" that was proposed by Yu Lei, Deren Chen and Zhongding Jiang. They present a Server Based Signature (SBS) digital signature scheme for mobile devices. Servers were responsible for generating non-repudiation digital signature. They use an adopted signature way in [4],"RSA-Based Partially Blind Signature with Low Computation", to avoid the expensive computation when user verifying signature generated from server without losing any security strength.

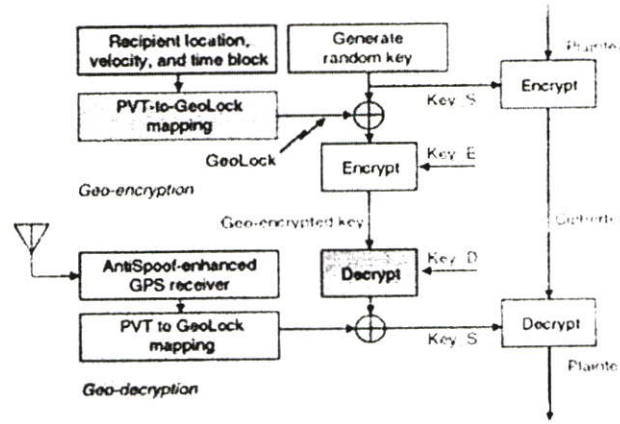


Fig 1. Geo-Encryption Algorithm [3]

With advances of mobile devices and GPS technology, D. E. Denning take advantage of GPS technology to do GPS-based encryption. It is called Geo-Encryption [3]. It is an encryption scheme that integrates position and time into the encryption and decryption processes in the way that provides an additional layer of security beyond that provided by conventional cryptography. It allows message to be encrypted for limited location or area. The recipient can decrypt messages if he stays in specific area and limited time. A geo-locking function is employed during encryption process to combine the recipient's geographic location (L), time and an encryption key to produce geo-secured key for transmission with the

0-7803-9740-X/06/$20.00 ©2006 IEEE

message. The message can only be decrypted if the recipient is physically positioned at location L. Geo-lock function creates geo-lock value by using position-velocity-time (PVT) to GeoLock mapping function which latitude, longitude and time constitute the inputs. Geo-Lock value is used to generate geo-secured key from session key and recover session key from geo-secured key.

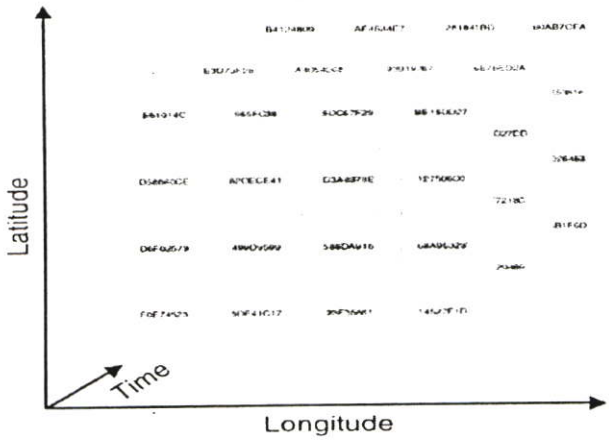


Fig 2. PVT-to-GeoLock Mapping is used to generate GeoLock value. [3]

Geo-Encryption is effective when the sender knows the receiver's location and the time that receiver will be there. "A Mobility Model for GPS-Based Encryption" [2] proposes a mobility model for geo-encryption techniques that allow mobile nodes to exchange movement parameters, therefore the sender can geo-encrypt to the receiver's estimated location. And finally they present methods for estimating the node's movement parameters.

As we have discussed, it's a good idea to take advantage of GPS technology to help mobile device generates digital signature. We propose "Location Based Digital Signature on Mobile Devices". We adopted geo-encryption to add layer of security for digital signature's security. Our model is suitable for mobile device, which has low-computation capability and short battery life. And receiver can verify digital signature that generated form server without losing any security strength. Furthermore it takes advantage of GPS technology.

III. LOCATION BASED DIGITAL SIGNATURE MODEL

In this paper based on [1], [2] and [3] that both of sender and receiver are mobile devices and can securely send their location to the others whenever necessary. Mobility model [2] is applied in case that mobile device has to update its mobility parameters to sign server. The sign server is used for helping mobile devices generate digital signatures. Fig.3 show location based digital signature architecture.

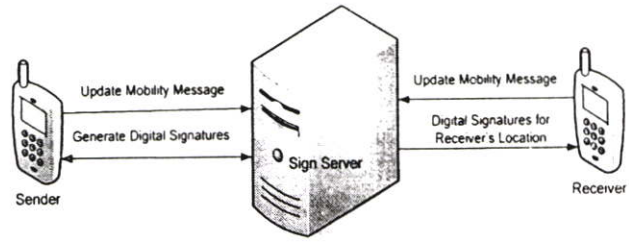


Fig 3. Model architecture with sign server

A. Generate Digital Signature

Based on [1], our model can generate digital signature that provides authentication, data integrity and non-repudiation of what have been done, in this paper interests in 2 cases: Non-Repudiation of Sender (NRS) and Non-Repudiation of Receiver (NRR). NRS guarantees that the sender can not later deny having sent that message. And NRR guarantees that receiver can't deny having received that message.

Sign server assists mobile devices create digital signature and receives mobility parameters from mobile devices. It allow sign server to estimate the mobile devices' location at any moment.

One-way collision-resistant hash functions, such as SHA1 or MD5 [7], use for generating public key of both mobile devices (sender, receiver). $h^i()$ denotes hash function with index i (i times hash operation). Each user has to generate secret key K_u, u means the user. By using K_u as input, each user can create hash chain $K_u^0, K_u^1, K_u^2, \ldots K_u^n$, $K_u^0 = K_u$ and $K_u^i = h^i(K_u) = h(K_u^{i-1})$. And $PK_u = K_u^n$ is the user's public key. If sender wants to generate digital signature from message "a", sender has to do following steps.

Step I: Generate $Cert_{snd}$

Sender creates his public key PK_{snd} and selects sign server (S). Then he has to send sender's identify, max numbers of digital signature (n) and S to the certification authority (CA) to create the certificate ($Cert_{snd}$).

$$Cert_{snd} = SK_{CA}(Sender, n, PK_{snd}, S)$$

SK_{CA} represents CA's secret key. After CA generated the certification, $Cert_{snd}$ will be made public via some directory services, such as LDAP [8].

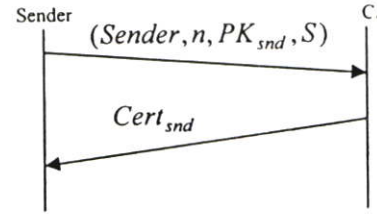


Fig 4. Certification's generate process. (Step I)

Step II: S selects p and q

First sign server (S) has to select 2 large prime numbers p and q, then computes $m = p \times q$ and $\phi(m) = (p-1) \times (q-1)$. After that, S selects number d from $d \times e = 1 \bmod \phi(m)$ where e = 3 in [1]. Then S keeps (d, p, q) as its secret key and publishes (e, m)

Step III: Generate δ, Z, η

If sender wants to generate digital signature of message a, sender selects 2 random numbers r, w and computes δ that is given by (1)

$$\delta = r^e h(a)(w^2 + 1) \bmod m \quad (1)$$

Sender has to read his latitude, longitude and t (X_t, Y_t, t) from GPS reader and use it as input for PVT-to-GeoLock mapping function, called mapping function. Mapping function will generate "GeoLock" value of sender at time t, called $Gsnd_t$. And then sender sends $(\delta, K'_{snd}, i, Gsnd_t, t)$ to S. After S received message, S will estimate sender's location at time t, $(\hat{X}_t, \hat{Y}_t)$, from (9) that will be discussed in section 3.2. The results are input for mapping function to generate $\hat{G}snd_t$. And then compare between $\hat{G}snd_t$ and $Gsnd_t$, if the result doesn't equal, sender won't generate digital signature of message a and sends (Ack_1) back to sender. When sender gains message, sender has to calculate new $Gsnd_t$ and send $(\delta, K'_{snd}, i, Gsnd_t, t)$ to S again. In case $Gsnd_t$ equals $\hat{G}snd_t$, S randomly selects positive number Z, $Z < M$, and send it to sender. Then sender randomly chooses integer r' and computes b from (2). Then it compute η from (3) and send (η) to S.

$$b = r \times r' \quad (2)$$

$$\eta = b^e \times (w - z) \quad (3)$$

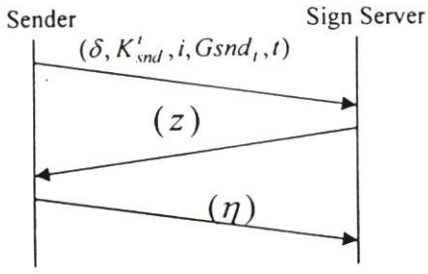


Figure 5. η and Z are generated after verified $Gsnd_t$ (Step III)

Step IV: Generate digital signature and verify.

After S gained η, S calculates γ from (4). Then S estimates receiver's location, $(\hat{X}_t, \hat{Y}_t)$, at time t by using formula (9) and mobility update message that receiver sent to S. S computes $\hat{G}rcv_t$ by using mapping function. And then it calculates α from (5) and send $(\gamma, \alpha, \hat{G}rcv_t)$ to sender. Sender has to compute c, s from (6) and (7)

$$\gamma = \eta^{-1} \bmod m \quad (4)$$

$$\alpha = h(Cert_{snd} \oplus Grcv_t)^d (\delta(z^2 + 1)\eta^{-2})^{2d} \bmod m \quad (5)$$

$$c = (wz + 1) \times \gamma \times b^e = (wz + 1)(w - z)^{-1} \bmod m \quad (6)$$

$$s = \alpha \times r^2 \times r'^4 \bmod m \quad (7)$$

$(Cert_{snd}, c, s)$ represents digital signature of message a and digital signature can verify by using formula (8).

$$s^e \equiv h(Cert_{snd} \oplus Grcv_t) h(a)^2 (c^2 + 1)^2 \bmod m \quad (8)$$

After sender verified digital signature, sender send $(Cert_{snd}, c, s), K'^{-1}_{snd}$ to S. S will verify digital signature and $K'_{snd} = h(K'^{-1}_{snd})$. If the results are correct, S sends message $(Cert_{snd}, c, s, a), K'^{-1}_{snd}, K'_{snd}$ to receiver.

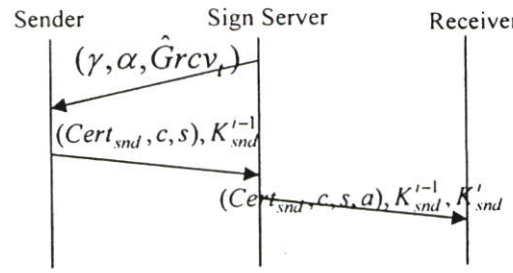


Figure 6. Sign server helps mobile devices to generate digital signature and send it to receiver. (Step IV)

By doing this step, we can generate digital signature that can support non-repudiation of sender (NRS).

It is easy to extend this model for supporting non-repudiation of receiver (NRR). As those processes described from step I to step VI, the receiver also generates a digital signature $(Cert_{rcv}, c', s')$. The receiver has to send $(Cert_{rcv}, c', s'), K^{J}_{rcv}$ to S and S transmit it to sender. Sender verified $(Cert_{rcv}, c', s')$, then sender sends K'^{-1}_{snd} to S. After S gained both K^{J-1}_{rcv} and K'^{-1}_{snd}, it trasmits $(Cert_{snd}, c, s, a), K'^{-1}_{snd}, K'_{snd}$ to the receiver and K^{J-1}_{rcv} to sender. Finally we can generate location based digital

signature on mobile devices that is suitable for mobile devices and support location based too.

B. Mobility parameters and Update process

Fig. 7 shows mobility parameters that mobile devices have to update to sign server whenever necessary. V represents average speed of mobile device. V is determined from observing the distance traveled during a specified time unit. θ represents the direction in which the mobile device is traveling and is measured as the positive angle between the positive x-axis. Both of V and θ are automatically calculated by the mobile device, not specified by user.

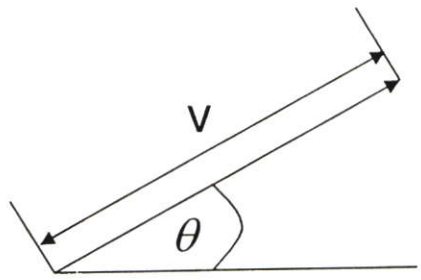


Figure 7 Mobility parameters: velocity (V), direction (θ)

Mobile devices (sender and receiver) have to send their update messages to S. From this information, S can estimate mobile device's location. Let A be mobile device, LA_0 (X_0, Y_0) represents A's location (latitude, longitude) at time t_0 that can read from GPS reader. LA_t (X_t, Y_t) represents A's location at time t and $t = t_1, t_2, t_3, \ldots, t_n$, such that $t_i = t_0 + i\omega$ where ω is a fixed time unit interval. First sender and receiver start sending update messages when they started the communication and keep on sending it whenever necessary. The line of movement makes an angle θ_0 with the positive direction of the latitude. At time t sign server can estimate mobile device's location as fig. 8

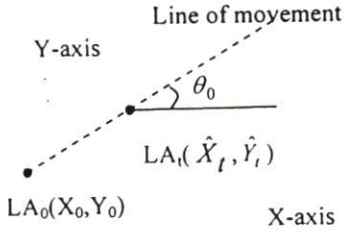


Fig 8. Mobile device's location at time t depends on V_0, θ_0, LA_0

With this information, sign server can estimate mobile device's location at time t by using formula (9)

$$\hat{X}_t = X_0 + (t - t_0)V\cos\theta$$
$$\hat{Y}_t = Y_0 + (t - t_0)V\sin\theta \qquad (9)$$

V and θ are automatically calculated by mobile device by using (10)

$$V_t = \sqrt{\left(\frac{X_t - X_0}{t - t_0}\right)^2 + \left(\frac{Y_t - Y_0}{t - t_0}\right)^2}$$
$$\theta_t = \arctan\left(\frac{Y_t - Y_0}{X_t - X_0}\right) \qquad (10)$$

Every time that mobile device reads latitude, longitude from GPS reader to calculate V and θ , mobile device has to determine whether or not to replace the old values of the parameters with new values and sends them to sign server. The X_0, Y_0 are replaced with the X_t and Y_t and set $t_0 = t$ only when the difference between the old and the new values of a parameter exceeds the threshold. Finally mobile device has to send update message to sign server. Update message is given by:

$$\{V_t, \theta_t, LA_t, t\}$$

The smaller threshold values for the parameters, the more often parameters are updated. So choosing optimal threshold values is the key to optimizing update process in order to achieve a balance between probability that Geo-Lock value which is estimated by sign server equals the Geo-Lock value which is estimated by mobile device and the frequency of sending update message.

IV. ANALYSIS OF THE PROPOSED METHOD

A. Location Based Analysis

This model uses sign server to help mobile device generate digital signature, receive update parameters from mobile device. First sender and receiver have to start update process at the beginning of communication. Sign server won't generate digital signature if $Gsnd_t$, that received from sender, doesn't equal to $\hat{G}snd_t$, that is estimated by sign server. And receiver can not verify digital signature correctly if recipient's location does not stay in specific area and time. Using an adopted digital signature way in [1], this model avoids the expensive computation when user verifying signature that was generated by sign server without losing any security strength and reduces the amount of computation time of the sender by almost 98% [4] compared with Abe-Fujisaki's model [6]. Our model generates digital signature that provides authentication, data integrity and non-repudiation services. If the intruder can break our signature scheme, he can also hack the SHA or MD5 and RSA cryptographic algorithms. In this model, mobile device's location participates in digital signature generated process. It also adds layer of digital signature's

security and cans apply in many applications, such as m-commerce etc.

B. Repudiation Analysis

If sender claimed he didn't send message that is generated by our model, receiver should send $(Cert_{snd}, c, s, a), K_{snd}^{t-1}, K_{snd}^{t}, Grcv_t$ to an arbiter. First the arbiter verifies $Cert_{snd}$ that generated from CA that consisted of sender's identify, maximum number of signatures, public root key of sender and sign server. Then the arbiter has to verify digital signature by using (8) and $PK_{snd} = K_{snd}^{n}$ that can be derived from $K_{snd}^{t-1}, K_{snd}^{t}$ by using one-way collision-resistant hash function. If all of results are correct, arbiter can believe that the sender has sent the message to the receiver. In case that the receiver claimed he didn't receive message from the sender. The arbiter can also verify in the same way as NRS.

C. Security Analysis

In case the intruder wants to generate fake digital signature. To successfully pass the verification equation (8), the intruder has to calculate s from formula (11).

$$s \equiv h(Cert_{snd} \oplus Grcv_t)^d \, h(a)^{2d} (c^2 + 1)^{2d} \bmod m \qquad (11)$$

If the intruder didn't know receiver's location, he can not generate fake signature. Even though he knows $h(Cert_{snd} \oplus Grcv_t)$ and $h(a), c$,he can not compute d without the factorization of m (p, q). Or if he knows $s, h(Cert_{snd} \oplus Grcv_t), h(a)$, it is intractable to compute c from (12) without p, q .

$$c^2 \equiv \left((s^e h(Cert_{snd} \oplus Grcv_t)^{-1} h(a)^{-3})^{1/2} - 1 \right) \bmod m \qquad (12)$$

V. CONCLUSION

This paper proposes location based digital signature on mobile device that has low-computation capability and short battery life. Mobile device is not suitable for conventional digital signature schemes. We take advantage of GPS technology by apply geo-encryption and mobility model in process of digital signature generation. Sign server is used to receive update message from mobile device and generate digital signature to reduce burden on mobile device. This model is suitable for mobile device and can apply to use in many application, such as m-commerce etc.

REFERENCES

[1] Yu Lei; Deren Chen and Zhongding Jiang, "Generating digital signatures on mobile devices", in Advanced Information Networking and Applications, 2004. AINA 2004. 18th International Conference on Volume 2, 2004 Page(s):532 - 535

[2] Ala Al-Fuqaha and Omar Al-Ibrahim and Joe Baird, "A Mobility Model for GPS-Based Encryption", in Global Telecommunications Conference, 2005. GLOBECOM '05, 2005 Volume 3, On page(s) 1721-1725

[3] L. Scott and D. Denning, "Geo-encryption: using GPS to enhance data security." GPS World, 1 Apr. 2003. http://www.gpsworld.com/gpsworld/article/articleDetail.jsp?id=57975

[4] Hung-Yu Chien and Jinn-Ke Jan and Yuh-Min Tseng, "RSA-based partially blind signature with low computation" in Proceedings of International Conference on Parallel and Distributed Systems (ICPADS'2001), 2001

[5] National Institute for Standards and Technology, "Digital Signature Standard (DSS)", Technical Report 169, August 30 1991

[6] M. Abe and E. Fujisaki, How to date blind signatures, Advances in Cryptology-ASIACRYPT'96, LNCS 1163, 1996, pp. 224-251

[7] B. Schneier. Applied Cryptography: Protocols, Algorithms, and Source Code in C. Join Wiley & Sons, Inc. 1996

[8] S. Boeyen, T. Hows, P. Richard. Internet x.509 public key infrastructure operational protocols-LDAPv2. RFC 2559, 1999.

ประวัติผู้เขียน

ชื่อ-นามสกุล	นาย สันติ จารุสมบัติ
ประวัติการศึกษา	สำเร็จการศึกษาระดับปริญญาตรี หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ จากคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2546
งานวิจัยที่สนใจ	Digital Signature, Mobile Agent, Cryptography, Location-based Encryption