การพัฒนา Servlet เพื่อประมวลผลคำสั่งซื้อโดยใช้มาตรฐาน MOSET

The Development of A Servlet for Electronic Payment with MOSET Standard

โดย

นางสาว อิสรียา วัฒนไพโรจน์รัตน์

รหัส 43067079

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2544
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	การพัฒนา Servlet เพื่อประมวลผลคำสั่งซื้อโดยใช้มาตรฐาน MOSET
นักศึกษา	อิสรียา วัฒนไพโรจน์รัตน์
อาจารย์ที่ปรึกษา	ดร.จันทร์บูรณ์ สถิตวิริยวงศ์
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2544

บทคัดย่อ

เนื่องจากการทำธุรกรรมผ่านทางอินเตอร์เน็ตมีการขยายตัวเพิ่มขึ้นมาก รูปแบบการซื้อขายได้รับการพัฒนาขึ้นอย่างต่อเนื่อง ทั้งนี้รวมถึงรูปแบบวิธีการในการชำระเงินทางอิเลกทรอนิกส์ ซึ่งได้รับการพัฒนาขึ้นอย่างหลากหลาย อย่างไรก็ตามรูปแบบที่ได้รับความนิยมนำมาพัฒนาใช้ คือโปรโตคอล SSL และ SET ซึ่งทั้ง 2 โปรโตคอลต่างมีข้อดีและข้อเสียแตกต่างกันไป จึงได้มีการพัฒนา MOSET ขึ้นเพื่อผสมผสานข้อดีจาก SET และ SSL เพื่อสร้างระบบการชำระเงินที่ใช้งานได้ง่ายขึ้น โครงการศึกษาระบบงานนี้จะศึกษา วิเคราะห์และพัฒนาระบบการซื้อขายสินค้าบนอินเตอร์เน็ต โดยนำหลักการของ MOSET มาเป็นมาตรฐานในการพัฒนา

Title	The Development of Web Applecation for electronic payment with MOSET standard
Student	Miss Isariya Wattanapairojerut
Advisor	Dr.Chanboon Satitwiriyawongs
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2001

ABSTRACT

As the expansion of internet using, electronic commerce is emerged and widely spreaded. All transactions require more reliability and confidentiality. There are many techniques employed to guarantee secrecy of every customers' payment data, privacy of the commercial operation of the trader as well as financial institution. Secure Socket Layer (SSL) and Secure Electronic Transactions (SET) play significant roles in electionic commerce. Each of them has its owned limitations. As they are analysed in the paper of Seminar2, there is the latest technology that integrate the usage of both SSL and SET, called MOSET. According from the study in Seminar2, this project will bring that background to implement the web application supporting the employing of MOSET by using the simulation of payment system that can support the mentioned technology as well.

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

หน้า

ตารางที่

สารบัญภาพ

หน้า

รูปที่

สารบัญภาพ (ต่อ)

หน้า

รูปที่

บทที่ 1

บทนำ

ในการทำธุรกิจบนอินเตอร์เน็ตสิ่งที่จะต้องคำนึงถึงคือความปลอดภัยของข้อมูลที่ติดต่อสื่อสารกัน ซึ่งในปัจจุบันข้อมูลในการซื้อขายสินค้าบนอินเตอร์เน็ตยังสามารถที่จะถูกเปลี่ยนแปลงแก้ไขหรือปลอมแปลงได้ เนื่องมาจากวิธีการป้องกันหรือการรักษาความปลอดภัยในการทำธุรกิจซื้อขายสินค้าบนอินเตอร์เน็ตในปัจจุบันยังไม่รัดกุมเพียงพอ ทำให้ง่ายต่อการขโมย ถอดรหัสหรือปลอมแปลงข้อมูลรายการในการทำธุรกรรมซื้อขายสินค้าได้

ข้อมูลที่สำคัญในการทำธุรกรรมทางพาณิชย์บนอินเตอร์เน็ตคือ หมายเลขบัตรเครดิต ซึ่งเป็นข้อมูลส่วนบุคคลที่ควรรู้เฉพาะเจ้าของบัตรเครดิตที่ทำรายการและบริษัทที่รับชำระบัตรเครดิตเท่านั้น ร้านค้าที่ให้บริการขายสินค้าหรือบุคคลอื่นไม่จำเป็นต้องทราบหมายเลขบัตรเครดิต แม้จะมีการพัฒนามาตรฐาน SET มาใช้เพื่อคุ้มครองความปลอดภัยของข้อมูลแล้ว แต่การใช้ SET ก็ไม่เป็นที่แพร่หลาย ทั้งนี้เนื่องจากความยุ่งยากในการติดตั้งซอฟต์แวร์สำหรับใช้งาน กล่าวคือ ในระบบ SET จะเริ่มจากผู้ถือบัตรเครดิตของธนาคารที่สนับสนุนระบบ SET ทำการดาวน์โหลดกระเป๋าสตางค์อิเลกทรอนิกส์ (E-wallet) มาติดตั้งลงบนคอมพิวเตอร์ก่อนและขอใบรับรองดิจิตอลตามมาตรฐาน SET จาก Certificate Authority ที่รองรับมาตรฐาน SET ซึ่งได้แก่ธนาคารที่ออกบัตรเครดิตนั้นๆ มาติดตั้งไว้บน Web Browser หลังจากนั้นต้องทำการกรอกแบบฟอร์มลงทะเบียนบัตรเครดิตเพื่อเข้าสู่ระบบ SET บน Web Site ของธนาคาร แล้วจึงจะสามารถชำระเงินด้วยบัตรเครดิตบนอินเตอร์เน็ตได้

ด้วยเหตุนี้จึงทำให้เกิดการพัฒนามาตรฐาน MOSET ย่อมาจากคำว่า Merchant Originated SET ซึ่งเป็นมาตรฐานสำหรับการทำงานของ Merchant Server เพื่อให้สามารถรับข้อมูลการชำระเงินจากลูกค้าที่ชำระโดยบัตรเครดิตแต่ใช้มาตรฐานอื่นเช่น SSL ได้ และทำการเปลี่ยนแปลงข้อมูลให้ตรงตามมาตรฐาน SET เพื่อทำการติดต่อกับ Payment Gateway ต่อไป ทั้งนี้เพื่ออำนวยความสะดวกให้แก่กลุ่มลูกค้า ให้สามารถใช้ประโยชน์จากระบบ SET ได้ง่ายขึ้น โดยสามารถซื้อสินค้าผ่านโปรโตคอลที่ให้ความคุ้มครองด้านความปลอดภัยอื่นๆ และสามารถใช้ประโยชน์ในเรื่องความปลอดภัยของข้อมูลผ่าน SET ได้ โดยในโครงการพัฒนาระบบงานได้นำมาตรฐาน MOSET มาใช้พัฒนาโปรแกรมสำหรับการซื้อขายสินค้าบนอินเตอร์เน็ต

1.1 วัตถุประสงค์

1 เพื่อศึกษาและเข้าใจหลักการทำงานของ MOSET (Merchant Originated SET)

2 นำความรู้ที่ได้มาประยุกต์ใช้สำหรับพัฒนาโปรแกรมรองรับการซื้อขายสินค้าบนอินเตอร์เน็ต โดยอ้างอิงมาตรฐานแบบ MOSET

1.2 ขั้นตอนการศึกษา

1. ศึกษาหลักการทำงานของ SSL (Secure Socket Layer), SET (Secure Electronic Transaction) และ MOSET (Merchant Originated SET)
2. พัฒนา web application เพื่อจำลองขั้นตอนการซื้อขายสินค้าบนอินเตอร์เน็ต โดย Merchant Server จะทำการแปลงข้อมูลที่ได้รับ ไปอยู่ในรูปแบบมาตรฐานของ SET

1.3 ขอบเขตการศึกษา

โครงการพัฒนานี้จะทำการแปลงข้อมูลจากรูปแบบของ SSL มาเป็น SET การแปลงข้อมูลดังกล่าวจะเกิดขึ้นที่ Merchant Server และเปรียบเทียบผลลัพธ์ที่ได้โดยดูจากข้อมูลที่ Payment Gateway ได้รับ ซึ่งจะทำการถอดรหัสกลับและแสดงข้อมูลที่ได้รับมา

1.4 ประโยชน์ที่ได้รับ

1. ความเข้าใจหลักการทำงานของ SSL, SET และ MOSET
2. ได้นำความรู้ที่ได้จากการศึกษามาประยุกต์ใช้ได้จริง

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 วิวัฒนาการของพาณิชย์อิเลกทรอนิกส์

พาณิชย์อิเลกทรอนิกส์เริ่มขึ้นตั้งแต่ ปี ค.ศ. 1960 โดยเริ่มจากบริษัทในสหรัฐอเมริกาได้นำการส่งเอกสารทางอิเลกทรอนิกส์ที่เรียกว่าระบบ EDI (Electronic Data Interchange : การส่งเอกสารหรือแบบฟอร์มอิเลกทรอนิกส์ที่เป็นมาตรฐานเพื่อการติดต่อทำการค้าระหว่างกัน) มาช่วยในการซื้อขายสินค้าระหว่างบริษัท นอกจากนี้สถาบันการเงินต่างๆ ได้สร้างเครือข่ายคอมพิวเตอร์ที่เรียกว่า EFT (Electronic Funds Transfer) ใช้ส่งผ่านรายการโอนเงินในเครือข่ายคอมพิวเตอร์ของสถาบันการเงิน เพื่อใช้ในการโอนเงินตราระหว่างธนาคาร ในช่วงดังกล่าวการติดตั้ง EDI จะต้องสร้างเครือข่ายสื่อสารส่วนตัวขึ้นมาเองซึ่งถือเป็นการลงทุนที่สูง และราคาแพง การใช้งานของ EDI จึงจำกัดอยู่ที่บริษัทขนาดใหญ่และสถาบันการเงินที่มีทุนทรัพย์เท่านั้น แต่ในปัจจุบันความแพร่หลายของอินเตอร์เน็ต ทำให้โลกการค้าอิเลกทรอนิกส์เปลี่ยนแปลงไป อินเตอร์เน็ตได้กลายเป็นช่องทางสื่อสารรูปแบบใหม่ที่มีการนำไปใช้งานอย่างกว้างขวาง จนทำให้ระบบการค้าอิเลกทรอนิกส์ในปัจจุบันขยายตัวอย่างรวดเร็วและไม่ได้จำกัดอยู่แต่เฉพาะสถาบันการเงินและบริษัทขนาดใหญ่เหมือนแต่ก่อน

2.2 ประเภทของพาณิชย์อิเลกทรอนิกส์

โครงสร้างของระบบการค้าอิเลกทรอนิกส์สามารถจำแนกออกเป็นประเภทต่างๆ ได้ ดังต่อไปนี้

2.2.1 การค้าอิเลกทรอนิกส์จัดแบบตามโครงสร้าง

จัดแบ่งได้ดังนี้

1. Consumer-to-Business คือ การค้าอิเลกทรอนิกส์ระหว่างผู้บริโภคกับธุรกิจ
2. Intra-Org E-commerce คือ การค้าอิเลกทรอนิกส์ภายในองค์กร เพื่อช่วยในการปรับปรุงการทำงานภายใน และให้บริการลูกค้าได้ดีขึ้น
3. Inter-Org E-commerce คือ การค้าอิเลกทรอนิกส์ระหว่างองค์กร เป็นแบบเดียวกับการค้าอิเลกทรอนิกส์ระดับ B2B (Business-to-Business) ซึ่งเป็นการค้าทางอิเลกทรอนิกส์ระหว่างองค์กรกับองค์กรด้วยกัน

2.2.2 การค้าอิเลกทรอนิกส์เชิงพาณิชย์

จัดแบ่งได้ดังนี้

1. B2B (Business-to-Business) เป็นการค้าขนาดใหญ่ระหว่างองค์กรกับองค์กร ซึ่งโดยทั่วไปจะเป็นสินค้าส่งออกหรือนำเข้าที่ต้องส่งสินค้าเป็นจำนวนมาก ซึ่งการชำระเงินจะผ่านระบบธนาคาร
2. B2C (Business-to-Consumer) เป็นการค้าปลีกไปยังผู้บริโภคทั่วโลก หรือภายในท้องถิ่นของตน ในส่วนนี้อาจจะรวมการค้าปลีกแบบจำนวนมากไว้ด้วย ซึ่งการชำระเงินโดยส่วนใหญ่จะเป็นการชำระเงินผ่านระบบบัตรเครดิต แต่อย่างไรก็ตามการค้าแบบ B2C นี้ก็มักจะทำให้เกิดการค้าแบบ B2B ได้ในอนาคต และหลายบริษัทจะทำกิจกรรมสองอย่างนี้ในคราวเดียวกัน
3. C2C (Consumer-to-Consumer) เป็นการค้าปลีกระหว่างบุคคลทั่วไป หรือระหว่างผู้ใช้อินเตอร์เน็ตด้วยกัน เช่น การขายสินค้าที่ใช้งานแล้ว รวมถึงการขายซอฟต์แวร์ด้วย ซึ่งปัจจุบันมีเป็นจำนวนมาก โครงสร้างไม่ซับซ้อนมากนัก

2.3 ระบบการชำระเงินบนอินเตอร์เน็ต

การทำธุรกิจบนอินเตอร์เน็ตแบบเต็มรูปแบบจะรวมถึงการจัดการการชำระเงินบนอินเตอร์เน็ตด้วย ซึ่งระบบการชำระเงินที่ดีมีประสิทธิภาพ จะต้องพิจารณาถึงประเด็นต่อไปนี้ :

1. ต้นทุนต่ำ เพื่อรองรับการซื้อขายสินค้ามูลค่าต่ำด้วย
2. มีความปลอดภัย เพราะการสื่อสารบนอินเตอร์เน็ตเป็นแบบเปิด
3. ให้การคุ้มครองความเป็นส่วนตัวแก่ทุกฝ่ายที่เกี่ยวข้องในระดับที่เหมาะสม เช่นระบบต้องไม่มีการบันทึกรายการสั่งซื้อของบุคคลใดบุคคลหนึ่ง
4. ลูกค้าต้องสามารถตรวจสอบความคืบหน้าหรือผลการทำงานของระบบได้ ควรเป็นการทำงานโดยใช้ software เพราะการสร้างระบบการชำระเงินที่ต้องลงทุนด้วย hardware จะทำให้ต้นทุนการลงทุนสูงกว่า ซึ่งไม่คุ้มค่าต่อการลงทุน

วิธีการชำระเงินบนอินเตอร์เน็ตนั้นจะเป็นเพียงแค่การส่งข้อมูลการตัดชำระเงินโดยใช้สื่อทางอิเลกทรอนิกส์แทนการใช้เงินสด บัตรเครดิต หรือเช็คเท่านั้น ความแตกต่างที่สำคัญคือสื่อแทนเงินอันใหม่นี้เป็นข้อมูลดิจิตอลที่ไม่สามารถจับต้องได้ โดยจะมีการใช้ระบบอิเลกทรอนิกส์มาประมวลผลการจ่ายหรือรับเงินแทนการรับเหรียญหรือธนบัตรจากมือหรือกระเป๋าเงิน การที่ข้อมูลทั้งหมดเป็นดิจิตอลทำให้ระบบชำระเงินต่างๆ ที่เกิดขึ้นบนอินเตอร์เน็ตมีพื้นฐานความคิดของระบบ

ที่คล้ายคลึงกัน เพียงแต่มีองค์กร ผู้พัฒนาและใช้ซอฟต์แวร์ที่แตกต่างกันไปเท่านั้น โดยสามารถแบ่งการชำระเงินออกได้เป็นแบบต่างๆ ดังนี้

2.3.1 การชำระเงินด้วยบัตรเครดิต

สำหรับการใช้บัตรเครดิตเพื่อใช้ในการชำระเงินบนอินเตอร์เน็ตนั้นจะมีลักษณะเช่นเดียวกับการชำระเงินด้วยบัตรเครดิตทั่วไป แต่จะมีการเพิ่มขั้นตอนเกี่ยวกับการรรักษาความปลอดภัยของการส่งข้อมูลการทำรายการระหว่างลูกค้ากับร้านค้า รวมทั้งเพิ่มระบบที่ใช้ตรวจสอบว่าผู้ทำรายการซื้อขายเป็นบุคคลที่มีสิทธิ์จริง บัตรเครดิตสามารถนำมาใช้ซื้อสินค้าบนอินเตอร์เน็ตได้ 2 แบบ คือ แบบการส่งผ่านข้อมูลของบัตรเครดิตโดยตรงโดยไม่มีการเข้ารหัสในการส่งข้อมูล และการเข้ารหัสก่อนแล้วจึงส่งข้อมูลไปให้ร้านค้า โดยการเข้ารหัสจะสามารถทำบางส่วนของข้อมูล ขึ้นอยู่กับข้อตกลงระหว่างกันว่าจะเข้ารหัสข้อมูลส่วนใด ถ้าลูกค้าเข้ารหัสข้อมูลทั้งหมดอย่างน้อยร้านค้าก็ต้องสามารถถอดรหัสส่วนที่เป็นรายละเอียดของสินค้าที่สั่งซื้อได้ จึงจะสามารถประมวลผลรายการของลูกค้านั้น และเพื่อป้องกันร้านค้าหรือผู้อื่นที่ไม่เกี่ยวข้องนำเอาข้อมูลหรือเลขที่บัตรเครดิตของลูกค้าไปใช้ ข้อมูลส่วนที่เป็นรายละเอียดของบัตรเครดิตจะถูกเข้ารหัสและถูกส่งต่อไปยังธนาคารหรือบริษัทที่ออกบัตรเครดิตเพื่อถอดรหัสและตรวจสอบความถูกต้อง แล้วจึงทำการตัดบัญชีต่อไป

2.3.2 การชำระเงินด้วยเช็คอิเลกทรอนิกส์

เป็นระบบที่พัฒนาขึ้นเพื่อให้ลูกค้าสามารถสั่งจ่ายเงินได้เหมือนกับการใช้เช็คสั่งจ่ายโดยตรงให้กับร้านค้าบนเครือข่ายอินเตอร์เน็ต ระบบเช็คอิเลกทรอนิกส์จะมีลักษณะเดียวกับเช็คปกติเกือบทุกประการเพียงแต่เอกสารบนกระดาษถูกเปลี่ยนเป็นข้อมูลทางอิเลกทรอนิกส์แทน ในการซื้อสินค้าหรือโอนเงิน ผู้สั่งจ่ายจะส่งข้อมูลไปให้ร้านค้าหรือผู้รับเงิน ทางร้านก็จะส่งผ่านข้อมูลนี้ต่อไปยังธนาคารหรือสถาบันการเงินเพื่อทำการโอนเงินเข้าบัญชี จากนั้นข้อมูลก็จะถูกส่งกลับยังผู้สั่งจ่ายเพื่อบอกว่าได้โอนเงินเรียบร้อยแล้ว

ข้อดีของเช็คอิเลกทรอนิกส์ คือ สามารถป้องกันการปลอมเช็คได้โดยการเข้ารหัสเลขที่บัญชีในแบบที่ธนาคารเท่านั้นที่จะสามารถถอดรหัสได้ ส่วนร้านค้าหรือผู้รับเช็คไปขึ้นเงินจะไม่สามารถทราบเลขที่บัญชีได้เลย นอกจากนี้ในการส่งผ่านข้อมูลอาจมีการใช้โปรโตคอล SET และใบรับรองดิจิตอลเข้ามาช่วยในการตรวจสอบตัวตนที่แท้จริงของผู้สั่งจ่ายเช็คหรือธนาคารจริงๆ

2.3.3 การชำระเงินด้วยเงินสดดิจิตอล (Digital Cash)

เงินสดดิจิตอลหรือ E-cash หรือ Digital Cash คือการนำข้อมูลดิจิตอลมาแทนการใช้เงินสด โดยระบบนี้จะเหมาะสำหรับการซื้อขายที่มีมูลค่าการซื้อขายน้อย และเป็นการขายสินค้าที่ลูกค้ารับสินค้าได้ทันทีบนอินเตอร์เน็ต เช่นการซื้อขายโปรแกรม ข้อมูลภาพ หรือข้อมูลข่าวสาร

ในระบบ Digital Cash ค่าของเงินจะเป็นเพียงชุดข้อมูลดิจิตอลเท่านั้น ทางธนาคารจะทำหน้าที่ออกชุดข้อมูลเหล่านี้ที่เรียกว่า Tokens และจะตัดเงินในบัญชีเป็นจำนวนเท่ากับมูลค่าของ Tokens ที่ถูกจ่ายไป ในการออก Digital Cash นั้นทางธนาคารจะตรวจสอบข้อมูลของจำนวนเงินพร้อมกับเพิ่มรายละเอียดในชุดข้อมูลของ Digital Cash ว่าสามารถชำระเงินจากชุดข้อมูลนี้ได้จริง จากนั้นจึงส่งชุดข้อมูลดังกล่าวมาให้ที่เครื่องคอมพิวเตอร์ของลูกค้า

เมื่อลูกค้าต้องการใช้ E-cash จะทำการส่งข้อมูลดังกล่าวให้กับร้านค้า ทางร้านค้าจะนำชุดข้อมูล E-cash ที่ได้ไปตรวจสอบกับทางธนาคาร ถ้าทุกอย่างเรียบร้อย ทางธนาคารจะโอนเงินไปเข้าบัญชีของร้านค้าตามจำนวน E-cash ชุดหนึ่งจะสามารถใช้ได้เพียงครั้งเดียวโดยจะมีหมายเลขประจำแต่ละชุดข้อมูลที่จะไม่ซ้ำกัน

2.4 ปัญหาของพาณิชย์อิเลกทรอนิกส์

ระบบการค้าอิเลกทรอนิกส์จะสมบูรณ์ได้ต้องมีขั้นตอนของการชำระเงิน ซึ่งขั้นตอนการชำระเงินดังกล่าวมีความแตกต่างกันไปตามประเภทของการค้าแต่ละชนิด วิธีการชำระเงินจึงถูกออกแบบมาให้เหมาะกับลักษณะการค้าของแต่ละธุรกิจ เช่น การชำระเงินแบบ On-line Internet EDI (Electronic Data Interchange) เหมาะกับการค้าขนาดใหญ่ (Business-to-Business) ส่วนการชำระเงินสำหรับการค้าขนาดเล็กส่วนมากจะใช้บัตรเครดิตเป็นหลัก ซึ่งสามารถใช้ได้ทั้งในและระหว่างประเทศ ซึ่งในการซื้อขายแบบปกติผู้ซื้อสามารถมั่นใจในสินค้าหรือบริการและตัวผู้ขายได้มากกว่าบนระบบออนไลน์ เนื่องจากสามารถมองเห็นและตรวจสอบได้ก่อนตัดสินใจซื้อ และเมื่อมีการชำระเงินด้วยบัตรเครดิต ผู้ขายจะได้รับลายเซ็นของผู้ซื้อบน Slip ซึ่งทำให้มั่นใจได้ว่าจะได้รับเงินอย่างแน่นอน แต่ในการชำระเงินด้วยบัตรเครดิตบนระบบการค้าอิเลกทรอนิกส์หากไม่มีการคุ้มครองความปลอดภัย ก็อาจนำไปสู่ปัญหาของทั้งผู้ซื้อและผู้ขายได้ ทำให้มีความจำเป็นที่จะต้องมีการรักษาความปลอดภัยสำหรับการชำระเงินแบบออนไลน์ ซึ่งสามารถที่จะช่วยตรวจสอบและยืนยันการสั่งซื้อสินค้า ตลอดจนการรักษาความถูกต้องของข้อมูลได้

2.5 เทคโนโลยีในการคุ้มครองความปลอดภัยบนอินเตอร์เน็ต

การรักษาความปลอดภัยของข้อมูล คือ เทคนิคและวิธีการต่างๆ ที่ใช้ป้องกันการแอบดูข้อมูล การเปลี่ยนแปลง การแทน หรือการทำลายข้อมูล โดยที่ไม่ได้รับอนุญาตจากเจ้าของข้อมูลนั้นๆ

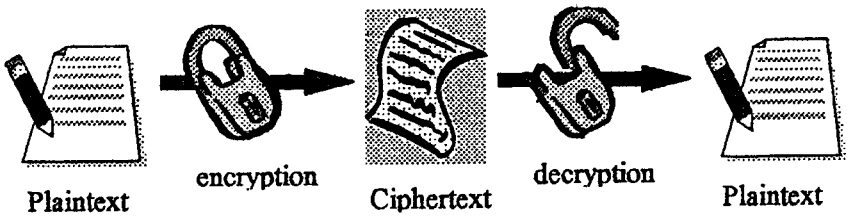
โดยทั่วไปแล้วในการให้บริการด้านความปลอดภัยของข้อมูล เทคโนโลยีต่างๆ ที่จะนำมาใช้ควรจะสามารถรองรับองค์ประกอบดังต่อไปนี้

1. การรักษาความลับของข้อมูล (Confidentiality) คือ การรักษาความลับของข้อมูลที่เก็บไว้หรือความลับของข้อมูลที่ส่งผ่านทางเครือข่าย โดยป้องกันไม่ให้ผู้อื่นที่ไม่มีสิทธิ์ ลักลอบดูได้ (เปรียบเทียบได้กับ การปิดผนึกซองจดหมาย การใช้ซองจดหมายที่ทึบแสง การเขียนด้วยหมึกที่มองไม่เห็น เป็นต้น)
2. การรักษาความถูกต้องของข้อมูล (Data Integrity) คือ การป้องกันไม่ให้ข้อมูลถูกแก้ไขโดยตรวจสอบไม่ได้ (เปรียบเทียบได้กับ การเขียนด้วยหมึกซึ่งถ้าถูกลบแล้วจะก่อให้เกิดรอยลบขึ้น การใช้โฮโลแกรมกำกับบนบัตรเครดิต เป็นต้น)
3. การระบุตัวบุคคล และการกำหนดสิทธิ (Authentication & Authorization) คือ การระบุตัวบุคคลที่ติดต่อว่าเป็น บุคคลตามที่ได้กล่าวอ้างไว้จริงหรือไม่ และการกำหนดสิทธิตามที่ได้รับอนุญาต (เปรียบเทียบได้กับ การแสดงตัวด้วยบัตรประจำตัวซึ่งมีรูปติดอยู่ด้วย การใช้ระบบล็อกซึ่งผู้ที่จะเปิดได้จะต้องมีกุญแจอยู่เท่านั้น เป็นต้น)
4. การป้องกันการปฏิเสธ หรือ อ้างปัดความรับผิดชอบ (Non-repudiation) คือ การป้องกันการปฏิเสธว่าไม่ได้มีการส่ง หรือรับข้อมูลจากฝ่ายต่างๆ ที่เกี่ยวข้อง หรือการป้องกันการอ้างที่เป็นเท็จว่าได้ รับ หรือส่งข้อมูล (เปรียบเทียบได้กับ การส่งจดหมายลงทะเบียน เป็นต้น)

วิธีการป้องกันเหตุการณ์ดังกล่าวสามารถทำได้ ดังนี้

2.5.1 ระบบเข้ารหัส (Cryptography System)

ระบบเข้ารหัส (Cryptography System) เป็นเทคโนโลยีหนึ่งที่ถูกนำมาใช้ในการรักษาความปลอดภัยของข้อมูล โดยเทคโนโลยีนี้จะแปรรูปข้อมูลอิเลกทรอนิกส์ธรรมดา (Plain Text) ซึ่งอาจเป็นตัวอักษรหรือรูปภาพ ให้อยู่ในรูปของข้อมูลที่ไม่สามารถอ่านให้เข้าใจได้ ซึ่งในที่นี้เราจะเรียกข้อมูลแบบนี้ว่า "ข้อมูลที่เข้ารหัส" (Ciphertext) ด้วยเทคโนโลยีนี้เจ้าของข้อมูลสามารถที่จะมั่นใจได้ว่าข้อมูลของตนจะได้รับความปลอดภัยเนื่องจากต้องใช้วิธีการแปรรูปที่ถูกต้องเท่านั้นเพื่อให้สามารถอ่านข้อมูลที่ถูกเข้ารหัสไว้ได้ ดังรูปที่ 2.1



รูปที่ 2.1 แสดงการเข้ารหัสและถอดรหัส (Encryption and Decryption)

การเข้ารหัสข้อมูลประกอบด้วยส่วนประกอบที่สำคัญ 2 ส่วน คือ Algorithm และ Key โดยนำข้อมูลที่ต้องการส่ง (Plain Text) กับกุญแจ (Key) ซึ่งเป็นตัวอักษรหรือตัวเลขสุ่ม มาผ่านกระบวนการทางคณิตศาสตร์ (Algorithm) ผลที่ได้ก็คือข้อมูลที่เข้ารหัส ขั้นตอนที่กล่าวมานี้จะเรียกว่า "การเข้ารหัส" (Encryption) และเมื่อผู้รับได้รับข้อมูลจะนำเอาข้อมูลที่เข้ารหัส (Ciphertext) กับกุญแจมาผ่านกระบวนการทางคณิตศาสตร์อันเดิม ผลลัพธ์ที่ได้ก็คือข้อมูลดั้งเดิม (Plain Text) ซึ่งขั้นตอนนี้จะเรียกว่า "การถอดรหัส" (Decryption) การเข้ารหัสที่ใช้ระบบ Key เป็นพื้นฐานนี้จะมีข้อดีคือ

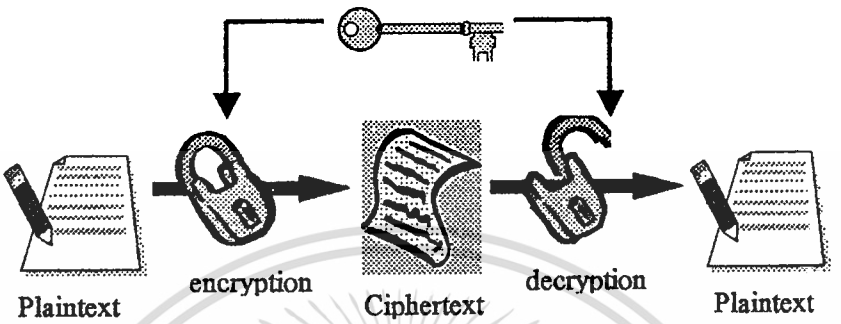
1. Algorithm ที่ใช้ในการเข้ารหัสนั้นยากที่จะคิดขึ้นใหม่และไม่จำเป็นต้องทำการสร้าง Algorithm ขึ้นมาใหม่ทุกครั้งที่ต้องการติดต่อสื่อสารกับคู่สนทนารายใหม่ แต่สามารถใช้ Algorithm เดิมเพียงแค่เปลี่ยน Key ที่แตกต่างกันออกไปตามกลุ่มบุคคลที่ติดต่อด้วย
2. ถ้ามีคนต้องการที่จะลักลอบถอดรหัสข้อมูลที่เราเข้ารหัสไว้ เราสามารถเปลี่ยน Key ใหม่มาใช้ กับ Algorithm เดิมได้

จะเห็นได้ว่ากุญแจเป็นตัวแปรสำคัญสำหรับระบบเข้ารหัส ดังนั้นจึงมีการแบ่งวิธีการเข้ารหัสข้อมูลด้วย Key ออกได้เป็น 2 ประเภทด้วยกันคือ

1. ระบบเข้ารหัสแบบกุญแจสมมาตร (Symmetric-key Cryptography)
2. ระบบเข้ารหัสแบบกุญแจอสมมาตร(Asymmetric-key Cryptography)

2.5.1.1 ระบบเข้ารหัสแบบกุญแจสมมาตร (Symmetric-key Cryptography)

รูปแบบของการเข้ารหัสข้อมูลแบบ Symmetric Key นี้ ทั้งผู้ส่งและผู้รับจะใช้ Key ที่เหมือนกันในการเข้าและถอดรหัสข้อมูล ดังรูปที่ 2.2



รูปที่ 2.2 แสดงการเข้าและถอดรหัสโดยวิธี Symmetric Cryptography

การเข้ารหัสข้อมูลด้วยวิธีนี้จะมีอุปสรรคเกิดขึ้น คือ ทั้งผู้ส่งและผู้รับจะต้องใช้ Key ร่วมกัน ดังนั้น Key ที่ใช้จะต้องเป็นความลับระหว่างผู้ติดต่อทั้ง 2 ฝ่าย เมื่อมีคู่ติดต่อเพิ่มมากขึ้น จำนวน Key ที่ต้องสร้างก็มีมากขึ้น เนื่องจากจะต้องมี Key 1 อันสำหรับแต่ละคู่สนทนา ถ้ามีการส่งข้อความถึงผู้รับจำนวน N คน จะต้องเก็บ Key เท่าจำนวนของผู้ที่เราติดต่อด้วย และหากใช้ Key หนึ่งอันสำหรับติดต่อกับคนมากกว่า 1 คน บุคคลอื่นที่ใช้ Key ร่วมกัน ก็สามารถเข้าหรือถอดรหัสได้เช่นกัน ทำให้เกิดปัญหากับการตรวจสอบตัวตนที่แท้จริง (Authentication) ของผู้ที่เราติดต่อด้วย และยังทำให้ข้อมูลไม่เป็นความลับ (Inconfidentiality) ปัญหาที่เกิดขึ้นจากการเข้ารหัสวิธีนี้ คือ การบริหารจัดการ Key การเก็บรักษา การแจกจ่ายและการควบคุมจะทำได้ลำบาก

Algorithm ที่ใช้วิธีการเข้ารหัสแบบ Symmetric Key ได้แก่

- DES (Data Encryption Standard) เป็นการเข้ารหัสข้อมูลที่ถูกสร้างโดย IBM และได้รับการรองรับจากรัฐบาลสหรัฐฯในปี 1977 ความยาวของ Key ที่ใช้มีขนาด 56 บิต แต่การเข้ารหัสข้อมูลจะกระทำบนขนาด 64 บิต ข้อดีของ DES คือทำงานได้รวดเร็วและง่ายในการใช้งานเหมาะกับการส่งข้อมูลที่มีขนาดใหญ่ไปในครั้งเดียว
- Triple DES ลักษณะการทำงานจะอยู่บนพื้นฐานการเข้ารหัสด้วย Algorithm DES โดยจะเข้ารหัสข้อมูลเป็นจำนวน 3 ครั้งด้วย Key ที่ต่างกัน 2 Keys หลักการทำงาน คือ จะทำการเข้ารหัสข้อมูลด้วย Key1 หลังจากนั้นจะทำการถอดรหัสด้วย Key2 และสุดท้ายจะเข้ารหัสอีกครั้งด้วย Key1 การรหัสแบบ Triple DES เป็นอีกทาง

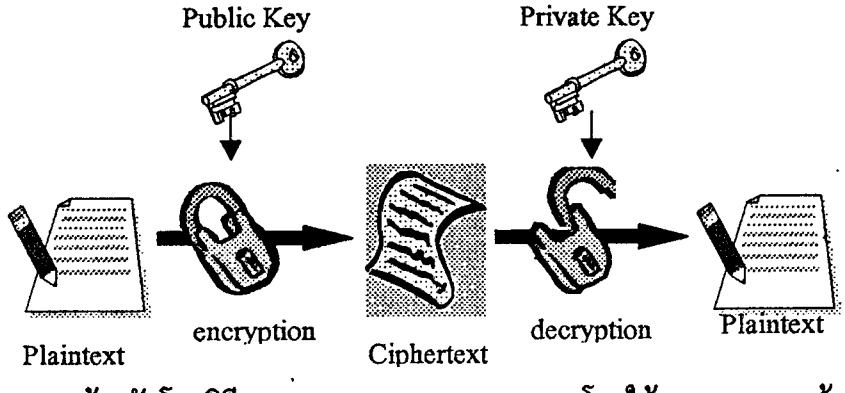
เลือกหนึ่งของการเข้ารหัสแบบ DES เพราะปัจจุบันการลักลอบถอดรหัสโดยวิธีของ DES สามารถทำได้ง่ายและเร็วขึ้น

- RC2 และ RC4 (Rivert's Code#2 และ #4) เป็น Algorithm เฉพาะที่ถูกสร้างขึ้นโดย RSA Data Security ขนาดของ Key ที่ใช้ในการเข้ารหัสจะไม่คงที่ โดยมีขนาดสูงสุดที่ 2048 บิต RC2 จะมีลักษณะของ Key เป็นแบบกลุ่มของรหัสเหมือน DES และ RC4 จะมีลักษณะของ Key เป็นแบบสายของรหัส Algorithm นี้นิยมใช้ในการเข้ารหัส Web Browser และ Server
- IDEA (International Data Encryption Algorithm) เป็น Algorithm ที่ถูกสร้างขึ้นในปี 1991 มีความปลอดภัยมากกว่า DES เพราะขนาดของ Key ที่ใช้ในการเข้ารหัส คือ 128 บิต IDEA ถูกนำไปใช้กับซอฟต์แวร์ที่เข้ารหัส E-mail เช่น PGP (Pretty Good Privacy)

2.5.1.2 ระบบเข้ารหัสแบบกุญแจอสมมาตร (Asymmetric-key Cryptography)

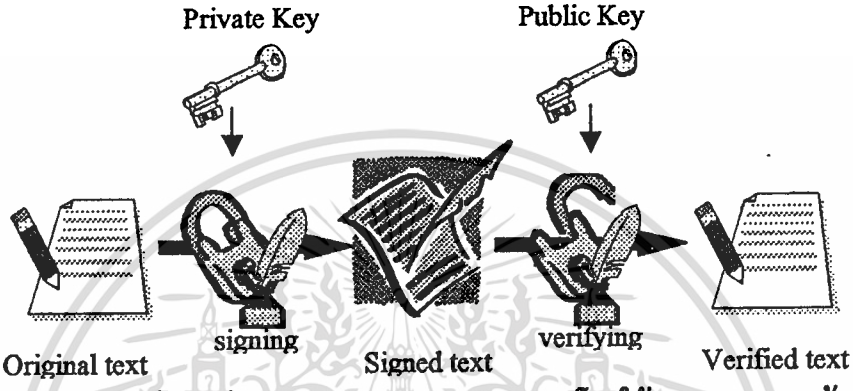
วิธีนี้เรียกอีกชื่อหนึ่งว่า "ระบบเข้ารหัสแบบกุญแจสาธารณะ" (Public-key Cryptography) การเข้ารหัสแบบ Asymmetric-key Cryptography นี้จะอยู่บนพื้นฐานแนวความคิดของ Key ที่เป็นคู่ คือ Private Key และ Public Key โดยที่ Key ทั้งสองจะแตกต่างกัน Key หนึ่งทำหน้าที่เข้ารหัส อีกอันใช้ในการถอดรหัส Key ที่เป็นคู่กันเท่านั้นจึงจะสามารถนำมาใช้ถอดรหัสข้อมูลที่ถูกเข้ารหัสด้วย Key คู่ของมัน ลักษณะการใช้งานของ Asymmetric-key Cryptography นี้สามารถทำได้ 2 ลักษณะ ดังนี้

1. การเข้ารหัสข้อมูลเพื่อปกปิดข้อมูลทั้งหมดเป็นความลับ (Confidentiality) โดยผู้ส่งใช้ Public Key ในการเข้ารหัสข้อมูล ซึ่งผู้รับก็จะใช้ Private Key ในการถอดรหัส วิธีนี้ใช้เพื่อยืนยันว่าข้อมูลเป็นความลับ และมีเพียงผู้รับที่มี private key คู่กันเท่านั้นจึงจะสามารถเปิดอ่านได้ ดังรูปที่ 2.3



รูปที่ 2.3 แสดงการเข้ารหัสโดยวิธี Asymmetric Cryptography โดยใช้ Public Key เข้ารหัส

2. การลงลายมือชื่อดิจิตอล (Digital Signature) โดยผู้ส่งใช้ Private Key ของตนเองเข้ารหัสและผู้รับใช้ Public Key ถอดรหัส เพื่อยืนยันตัวตนที่แท้จริงของผู้ส่ง (Authentication) นอกจากนี้ยังสามารถป้องกันการปลอมแปลงแก้ไขเอกสาร (Data Integrity) และป้องกันไม่ให้ผู้ออกเอกสารปฏิเสธเอกสารใดๆ ที่ส่งไปให้ผู้รับ (Non-repudiation) ดังรูปที่ 2.4 ซึ่งเป็นหลักการของการทำ Digital Signature



รูปที่ 2.4 แสดงการเข้ารหัสโดยวิธี Asymmetric Cryptography โดยใช้ Private Key เข้ารหัส

Algorithm ที่ใช้วิธีการเข้ารหัสแบบ Asymmetric Key ได้แก่

- Diffie-Hellman ออกแบบเพื่อใช้ในการติดต่อระหว่างบุคคล 2 ฝ่าย โดยแต่ละฝ่ายจะมี Secret Key เป็นของตนเอง และทำการแลกเปลี่ยนข้อมูล Secret Key ของแต่ละฝ่าย ในการแลกเปลี่ยนข้อมูลนั้นจะทำการสร้าง Session Key ที่ได้มาจาก Secret Key ของทั้ง 2 ฝ่าย และเข้ารหัสข้อมูลด้วย Session Key นั้น ดังนั้นถ้าผู้รับไม่ใช่บุคคลที่ทำการแลกเปลี่ยน Secret Key กันก็จะไม่สามารถสร้าง Session Key ที่จะใช้ในการติดต่อกันต่อไปได้
- RSA (Ronald Rivert, Adi Shamir and Leonard Adelman) เป็น Algorithm ที่ใช้ในการทำ Digital Signature โดยจะทำการเข้ารหัสข้อมูลด้วย Private Key และถอดรหัสข้อมูลด้วย Public Key และขนาดความยาวของ Key ที่ใช้ไม่คงที่ โดยจะอยู่ในช่วงตั้งแต่ 512 บิต จนถึงมากกว่า 1,024 บิต

2.5.1.3 การเปรียบเทียบวิธีการเข้ารหัส

ระบบการเข้ารหัสด้วยวิธีใดวิธีหนึ่งนั้นไม่สามารถแก้ปัญหาได้ทั้งหมด โดยการเข้ารหัสแต่ละวิธีจะมีข้อดีข้อเสีย ดังตารางที่ 2.1

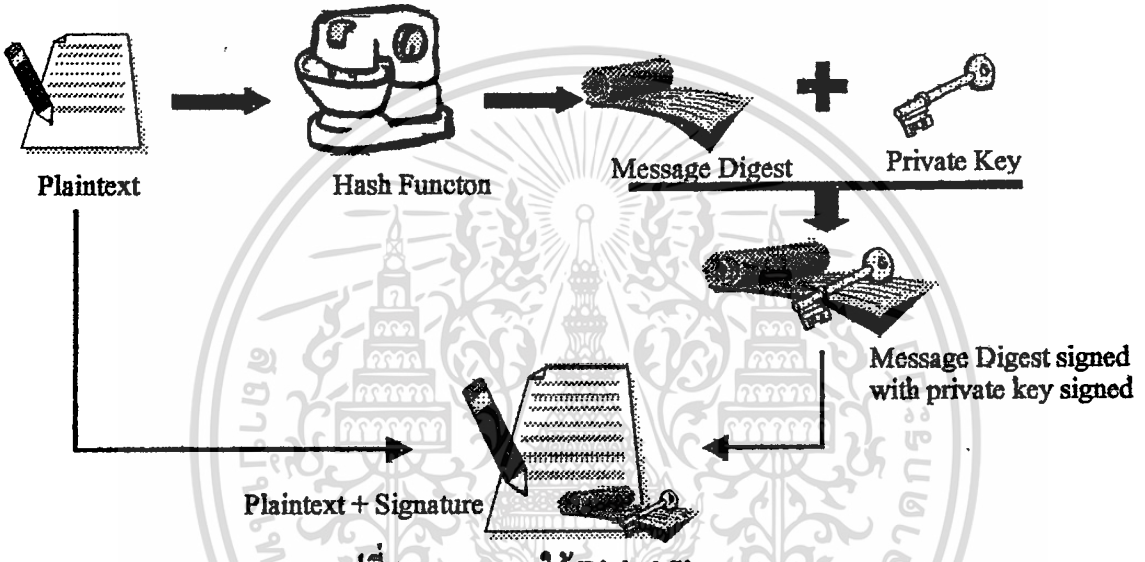
ประเภทของการเข้ารหัส	ข้อดี	ข้อเสีย
Symmetric Cryptography	- รวดเร็ว - ง่ายในการนำไปใช้งาน	- Key ที่ใช้ต้องเหมือนกันทั้งผู้รับและผู้ส่ง - ยุ่งยากในการจัดเก็บ Key เพราะต้องใช้Key เท่ากับจำนวนกลุ่มที่ติดต่อด้วย - มีความยุ่งยากในการที่จะส่ง Key ออกไปให้คนที่ติดต่อด้วย - ไม่สามารถใช้กับการทำ Digital Signature ได้
Asymmetric Cryptography	- ใช้สอง Key ที่แตกต่างกันในการเข้าและถอดรหัส - การจัดส่ง Key ทำได้ง่าย - มีความน่าเชื่อถือและสนับสนุนการส่งข้อมูลโดยใช้ Digital Signature	- ช้าและการคำนวณค่อนข้างยากและซับซ้อนกว่า

ตารางที่ 2.1 แสดงข้อดีและข้อเสียของการเข้ารหัสในแต่ละวิธี

ในทางปฏิบัติ การเข้ารหัสแบบ Asymmetric Cryptography และ Symmetric Cryptography สามารถทำร่วมกันได้ โดยเข้ารหัสข้อมูลโดยวิธี Symmetric ก่อนแล้วจึงเข้ารหัสอีกชั้นด้วยแบบ Asymmetric ซึ่งการเข้ารหัสแบบ Symmetric จะมีประสิทธิภาพมากสำหรับการส่งข้อมูลขนาดใหญ่ แต่จะมีปัญหาในเรื่องความรัดกุมในด้านความปลอดภัยในการส่ง Key ไปให้คู่สนทนา ในขณะที่การเข้ารหัสแบบ Asymmetric ให้ความปลอดภัยในเรื่องการจัดส่ง Key แต่ไม่เหมาะกับการส่งข้อมูลขนาดใหญ่ ดังนั้นจึงนำกลไกของ Symmetric มาใช้ในการเข้ารหัสข้อมูลที่จะส่ง พร้อมทั้งแนบ Key ที่ใช้ในการเข้ารหัสไปด้วย และใช้ Asymmetric มาเข้ารหัสตัว Key (ของ Symmetric) ด้วย Public Key ของผู้รับ ดังนั้นผู้ที่จะสามารถถอดรหัสเอา Key ที่ใช้เข้ารหัสข้อมูลจริงได้ จะต้องเป็นผู้ที่มี Private Key ของผู้รับเท่านั้น

2.6 Digital Signature

ในการส่งข้อมูลไปยังผู้รับถ้าข้อมูลมีขนาดใหญ่จะไม่สะดวกในการเข้ารหัสและถอดรหัสข้อมูล จึงเกิดความคิดที่จะส่งข้อมูลที่จะทำการเข้ารหัสให้มีขนาดเล็กและสามารถยืนยันถึงผู้ส่งได้ คือ การทำ Digital Signature หรือลายเซ็นดิจิตอล ที่เป็นลายเซ็นทางอิเลกทรอนิกส์ที่ใช้ในการรับประกันว่าข้อความที่ส่งไปนั้นจะไม่ถูกเปลี่ยนแปลงในระหว่างการส่งและสามารถยืนยันถึงผู้ส่งได้ และผู้รับสามารถมั่นใจได้ว่าข้อมูลที่ได้รับมีความถูกต้อง



รูปที่ 2.5 แสดงการใช้ Digital Signature

การทำ Digital Signature ดังรูปที่ 2.5 เป็นการนำเอาข้อมูลที่ต้องการส่งมาผ่านฟังก์ชันทางคณิตศาสตร์ คือ Hash Function ได้เป็น Message Digest ออกมา ขนาดของ Message Digest ที่ได้จะขึ้นอยู่กับฟังก์ชันที่ใช้ เช่น ถ้า Hash Function มีขนาด 60 บิต จะได้ Message Digest ขนาด 60 บิต และทำการเข้ารหัส Message Digest นั้นด้วย Private Key ของผู้ส่งได้เป็นลายเซ็นดิจิตอล นำลายเซ็นดิจิตอลของผู้ส่งนี้แนบไปกับข้อมูลที่ต้องการส่ง เมื่อผู้รับได้รับข้อมูลจะทำการตรวจสอบลายเซ็นดิจิตอลนั้น โดยการถอดรหัสลายเซ็นดิจิตอลนั้นด้วย Public Key ของผู้ส่ง ได้เป็น Message Digest แล้วไปผ่าน Hash Function ที่เป็นฟังก์ชันเดียวกับที่ผู้ส่งใช้ ผลลัพธ์ที่ได้จะนำไปเปรียบเทียบกับข้อมูลที่ส่งมาพร้อมลายเซ็นฯ เพื่อตรวจสอบความถูกต้อง

ปัญหาที่เกี่ยวกับการส่งข้อมูลด้วยลายเซ็นดิจิตอล คือ ข้อมูลจะถูกส่งมาในลักษณะที่สามารถอ่านได้ ไม่มีการป้องกันการเรียกดูข้อมูลจากผู้ที่ไม่ได้รับอนุญาต ดังนั้นจึงควรนำวิธีการเข้ารหัสข้อมูลมาใช้ควบคู่กับการทำลายเซ็นดิจิตอลด้วย

2.7 Digital Certificate

ใบรับรองดิจิตอล (Digital Certificate) เป็นชุดของข้อมูลที่ใช้รับรองบุคคลหรือองค์กรว่าเป็นผู้ส่งเอกสารนั้นจริง โดยมีตัวกลางที่ไว้ใจได้เป็นผู้ออกใบรับรองฯ ให้ ตัวกลางดังกล่าว คือ องค์กรรับรองความถูกต้อง หรือ Certificate Authority (CA)

ประเภทของใบรับรองดิจิตอล แบ่งออกเป็น 3 ประเภท คือ

1. ใบรับรองเครื่องแม่ข่าย
2. ใบรับรองตัวบุคคล
3. ใบรับรองสำหรับ CA

การสร้างใบรับรองดิจิตอลขั้นตอนแรก เครื่องคอมพิวเตอร์ของผู้ขอใบรับรองฯ จะสร้าง Key ขึ้นมา 1 คู่ โดยเก็บ Private Key ไว้กับตัวเอง แล้วส่ง Public Key พร้อมกับข้อมูลส่วนตัวที่ต้องการให้ปรากฏบนใบรับรองให้กับ CA จากนั้น CA จะสร้างใบรับรองฯขึ้น โดยในใบรับรองฯ จะมีรายละเอียดดังต่อไปนี้

- ข้อมูลส่วนตัวของผู้ที่ได้รับการรับรอง เช่น ชื่อ องค์กร ที่อยู่
- ข้อมูลระบุผู้ออกใบรับรอง ได้แก่ ลายมือชื่อดิจิตอลขององค์กรที่ออกใบรับรอง หมายเลขประจำตัวของผู้ออกใบรับรอง หมายเลขประจำตัวของผู้ออกใบรับรอง
- กุญแจสาธารณะของผู้ได้รับการรับรอง
- วันหมดอายุของใบรับรองดิจิตอล
- ระดับชั้นของใบรับรองดิจิตอล ซึ่งมีทั้งหมด 4 ระดับ ในระดับ 4 จะมีกระบวนการตรวจสอบเข้มงวดที่สุด และต้องการข้อมูลมากที่สุด
- หมายเลขประจำตัวของใบรับรองดิจิตอล

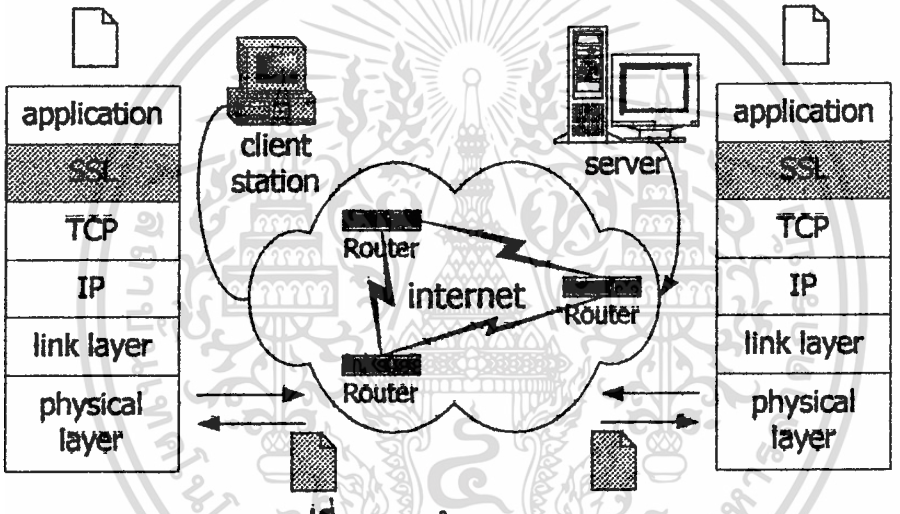
เมื่อ CA สร้างใบรับรองขึ้นแล้ว จะทำการเข้ารหัสใบรับรองด้วย Public Key ของผู้ขอใบรับรองฯ แล้วส่งกลับไปให้ผู้ขอ สุดท้ายผู้ขอจะต้องใช้เครื่องคอมพิวเตอร์เครื่องเดิมซึ่งมี Private Key ที่ใช้ขอใบรับรองฯ ทำการถอดรหัสใบรับรองฯ ที่ได้รับ

จากวิธีการดังกล่าว จะเห็นได้ว่าใบรับรองดิจิตอล คือ ชุดของข้อมูลที่ Public Key ของบุคคลหรือองค์กรที่ได้รับการลงนาม (signed) โดยผู้รับรอง ใบรับรองดิจิตอลจะถูกใช้ในการแลกเปลี่ยนข้อมูลบนอินเตอร์เน็ตในกรณีที่ต้องการยืนยันตัวตนที่แท้จริงของผู้ส่งข้อมูล โดยผู้ส่งสามารถทำสำเนาใบรับรองฯของตนเองแนบไปกับข้อมูลได้ หากปราศจากใบรับรองฯจะไม่สามารถแน่ใจได้ว่า Public Key นั้นเป็นของบุคคลหรือองค์กรที่อ้างความเป็นเจ้าของจริงหรือไม่ จึงเห็นได้ว่าใบรับรองฯมีประโยชน์อยู่ 2 ประการ คือ ช่วยให้ผู้รับข้อมูลมั่นใจได้ว่าข้อมูลนั้นถูกส่งจากผู้ส่งตัวจริงตามที่อ้าง และผู้ส่งไม่สามารถบอกปัดความรับผิดชอบต่อข้อมูลที่ส่งไปนั้นได้

2.8 Transaction Security Protocols

2.8.1 Secure Socket Layer (SSL)

SSL (ปัจจุบันพัฒนาถึงเวอร์ชั่น 3.0) ถูกพัฒนาโดย Netscape สำหรับทำงานในเว็บเบราเซอร์ทั่วไป SSL ทำงานอยู่ระหว่าง Application Layer และ Transportation Layer โดยจะเข้ารหัสข้อมูลก่อนจะส่งไปให้ TCP เว็บไซต์ที่สนับสนุน SSL จะมีเครื่องหมายแสดงบน Web Browser ว่า Web Page นั้นได้รับการรักษาความปลอดภัย เช่น กรณีของ Netscape Navigator จะแสดงด้วยรูปกุญแจที่มุมล่างด้านซ้าย SSL ถูกนำไปใช้อย่างแพร่หลายทั้งบน Intranet, Internet Server และ Browser การทำงานของ SSL แสดงในรูปที่ 2.6 ดังนี้



รูปที่ 2.6 การทำงานของ SSL

การเข้ารหัสสำหรับ SSL จะใช้วิธีการเข้ารหัสทั้งแบบ Symmetric และ Asymmetric Cryptography โดย Asymmetric จะใช้ในการทำ Authentication ระหว่าง Server กับ Client ในช่วงแรกของการติดต่อ และใช้ Symmetric Cryptography เพื่อเข้ารหัสข้อความที่รับส่งกันตลอดช่วงของแต่ละ Session ที่ใช้ในการติดต่อกันระหว่าง Server กับ Client ซึ่งเรียกว่า Session Key

ขั้นตอนการทำงานของ SSL คือ เมื่อ Client เข้ามายัง Web Page ที่รักษาความปลอดภัยด้วย SSL Client จะขอติดต่อกับ Server โดยส่งข้อความไปยัง Server เมื่อ Server ได้รับข้อความจาก Client จะตอบกลับข้อความนั้นไปยัง Client ถ้า Client รองรับ SSL ก็จะตอบกลับมายัง Server เพื่อตอบรับเริ่มต้นการรับส่งข้อมูลโดยใช้ SSL ซึ่งเป็นการเข้าสู่ขั้นตอนที่เรียกว่า SSL Handshake โดย Server และ Client จะแลกเปลี่ยนข้อมูลการรักษาความปลอดภัยซึ่งกันและกัน ในข้อมูลที่ Client ส่งกลับมายัง Server จะระบุหมายเลขของการติดต่อครั้งนี้ Algorithm ที่จะใช้ในการเข้ารหัส และวิธี

การบีบอัดข้อมูล จากนั้น Server จะติดต่อกลับไปยัง Client และทำการส่งสำเนาใบรับรองดิจิตอลให้กันและกันเพื่อตรวจสอบ โดย Server จะส่ง Public Key เฉพาะสำหรับการติดตั้งครั้งนั้น (Session Key) ไปยัง Client ด้วย Client จะใช้ Key ที่ได้รับทำการเข้ารหัสข้อมูลก่อนส่งไปยัง Server และ Client สามารถติดต่อกันได้อย่างปลอดภัย ทั้งนี้ในการติดต่อกับแต่ละ Client จะใช้ Key ที่ต่างกัน และ Key ที่สร้างขึ้นเฉพาะ session นั้นจะหมดอายุภายใน 24 ช.ม.โดยอัตโนมัติ โดยสรุป SSL ได้จัดเตรียมบริการด้านความปลอดภัยพื้นฐานเอาไว้ให้ 3 อย่าง ดังตารางที่ 2.2 ที่แสดงถึงบริการด้านความปลอดภัยพื้นฐานของระบบ SSL

บริการที่ SSL มีให้	เทคโนโลยีที่ใช้	สิ่งที่ป้องกันได้
Server Authentication	ใบรับรองดิจิตอลตามมาตรฐาน X.509	การแอบอ้างโดยบุคคลอื่น และการปัดความรับผิดชอบในภายหลัง
Data Security	การเข้ารหัส	การแอบอ่าน , แก้ไข, ขโมยข้อมูล
Client Authentication (Optional)	ใบรับรองดิจิตอลตามมาตรฐาน X.509	การแอบอ้างโดยบุคคลอื่น และการปัดความรับผิดชอบในภายหลัง

ตารางที่ 2.2 แสดงบริการด้านความปลอดภัยพื้นฐานของ SSL

ความปลอดภัยของข้อมูล แบ่งได้เป็น 2 ด้าน คือ ความเป็นส่วนตัวของข้อมูล และ ความถูกต้องของข้อมูล โดยความเป็นส่วนตัวของข้อมูล เกิดจากการใช้การเข้ารหัสทั้งแบบ Symmetric Cryptography ร่วมกับแบบ Asymmetric Cryptography เพื่อให้สามารถทำการเข้ารหัสและถอดรหัสได้อย่างรวดเร็ว และในขณะเดียวกันก็ยังคงไว้ซึ่งความปลอดภัยในระดับสูงโดยที่ข้อมูลทุกอย่างที่สงไปมาระหว่าง Server และ Client จะถูกเข้ารหัสโดยใช้ Key และ Algorithm การเข้ารหัสที่ตกลงกันไว้ในช่วง SSL Handshake ทำให้แม้ผู้ลักลอบอ่านข้อความจะใช้อุปกรณ์ตรวจจับกลุ่มข้อมูลไอพี (IP Packet Sniffer) มาอ่านข้อความก็จะเห็นแต่ข้อความที่ถูกเข้ารหัสไว้

ความถูกต้องของข้อความ บริการนี้ช่วยให้สามารถมั่นใจได้ว่าข้อมูลจะไม่ถูกแก้ไขในระหว่างทางที่ส่งไปมาระหว่าง Server และ Client โดยอาศัย Hash Function ประกอบกัน

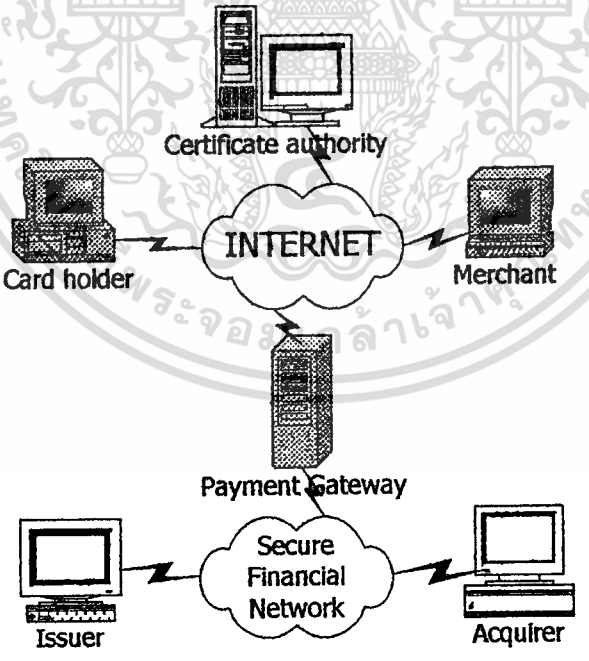
การตรวจสอบซึ่งกันและกัน Client สามารถตรวจสอบใบรับรองดิจิตอลของ Server และบน SSL หากผู้ใช้ทางด้าน Client มีใบรับรองดิจิตอล Server ก็สามารถจะขอตรวจสอบผู้ใช้ได้ด้วยเช่นกัน ในการแลกเปลี่ยนใบรับรองดิจิตอลจะเกิดขึ้นในขั้นตอน SSL Handshake เพื่อให้มั่นใจว่าฝ่ายที่แสดงใบรับรองดิจิตอลเป็นเจ้าของใบรับรองนั้นจริง และต้องมีลายเซ็นดิจิตอล (Digital Signature) กำกับข้อมูลทุกอย่างที่ส่งให้อีกฝ่ายหนึ่ง ในขั้นตอน SSL Handshake ข้อมูลที่ถูกเซ็น

กำกับจะต้องมีใบรับรองด้วย เพื่อป้องกันไม่ให้ผู้อื่นปลอมแปลงเพราะจะมีเพียงผู้ที่มี Private Key ที่คู่กับ Public Key บนใบรับรองเท่านั้นที่สามารถเซ็นกำกับข้อมูลได้อย่างถูกต้อง

2.8.2 Secure Electronic Transaction (SET)

พัฒนาขึ้นโดย VISA และ Master Card และพัฒนาร่วมกับ MicroSoft, CyberCash, GTE, IBM และ Netscape SET มีความแตกต่างจาก SSL อย่างมาก มีความรัดกุมในการคุ้มครองความปลอดภัยสูงกว่า SSL โดยมีการตรวจสอบ 3 ฝ่ายหลักๆ คือ ลูกค้า ผู้ขาย และธนาคาร (ธนาคารเป็นตัวกลางในการทำรายการชำระเงิน) ผู้ขายจะไม่ทราบหมายเลขบัตรเครดิตของผู้ซื้อ เนื่องจากข้อมูลเกี่ยวกับบัตรเครดิตจะถูกส่งไปยังธนาคารของผู้ขายโดยตรวจสอบความถูกต้องข้อมูลบัตรเครดิตและวงเงินกับธนาคารผู้ออกบัตร เมื่อได้รับการอนุมัติวงเงิน ธนาคารผู้ขายก็จะนำเงินเข้าสู่บัญชีผู้ขาย

SET เป็นโปรโตคอลแบบ transaction-oriented ซึ่งมีการเข้ารหัสแบบ DER (Distinguished Encoding Rules) ตามรูปแบบ ASN.1 (Abstract Syntax Notation 1) รูปที่ 2.7 แสดงภาพการติดต่อระหว่าง entities ในระบบ SET



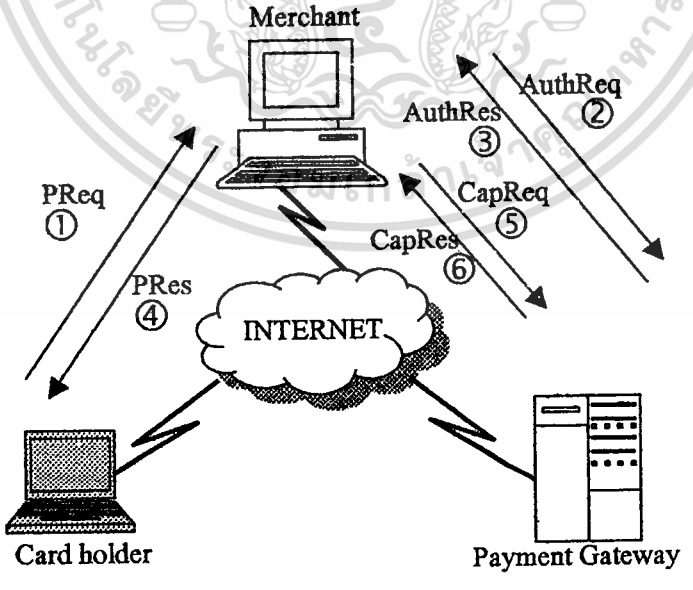
รูปที่ 2.7 โครงสร้างการติดต่อในระบบ SET

การส่งโดยใช้ SET นี้สามารถแบ่งลักษณะข้อมูลได้เป็น 2 กลุ่ม คือ กลุ่มข้อมูลการรับรอง (certification message) และกลุ่มข้อมูลในการชำระเงิน แต่จากทั้ง 2 กลุ่ม จะมีข้อมูลพื้นฐานที่ต้องมี ดังแสดงในตารางที่ 2.3

ข้อมูล	ผู้ส่ง – ผู้รับ	ความหมาย
PReq	C → M	Purchase Request
PRes	M → C	Response to Preq
AuthReq	M → G	Authorization Request
AuthRes	G → M	Response to AuthReq
CapReq	M → G	Request for compensation
CapRes	G → M	Response to CapReq

ตารางที่ 2.3 ข้อมูลพื้นฐานที่ต้องมีในการชำระเงินผ่าน SET

จากตาราง PReq และ PRes เป็นข้อมูลที่ส่งจากผู้ซื้อ (C คือ Cardholder) ไปให้ ผู้ขาย (M คือ Merchant) ในขณะที่ข้อมูลอื่นเป็นการติดต่อระหว่างผู้ขายและ Payment Gateway ดังที่แสดงในรูปที่ 2.8



รูปที่ 2.8 แสดงการแลกเปลี่ยนข้อมูลพื้นฐานใน SET

การค้าอิเลกทรอนิกส์โดยระบบ SET จะเริ่มจากผู้ถือบัตรเครดิตของธนาคารที่สนับสนุนระบบ SET ทำการดาวน์โหลดกระเป๋าสตางค์อิเลกทรอนิกส์ (E-wallet) มาติดตั้งลงบนคอมพิวเตอร์และขอใบรับรองดิจิตอลตามมาตรฐาน SET จาก CA ที่สนับสนุน SET ซึ่งได้แก่ธนาคารที่ออกบัตรเครดิตนั้นๆ มาติดตั้งไว้บน Web Browser หลังจากนั้นจะกรอกแบบฟอร์มลงทะเบียนบัตรเครดิตเพื่อเข้าสู่ระบบ SET บน Web Site ของธนาคาร การที่จะมั่นใจได้ว่า Web Site นั้นเป็นของธนาคารผู้ออกบัตรเครดิตให้จริงหรือไม่กระทำโดยการตรวจสอบใบรับรองของ Web Site ธนาคารที่ส่งมายัง Web Browser โดยผ่าน SSL เมื่อระบบของทางธนาคารตรวจสอบข้อมูลกับฐานข้อมูลแล้วตรงกัน ก็จะสามารถ Download ใบรับรองมาติดตั้งบนกระเป๋าสตางค์อิเลกทรอนิกส์ พร้อมที่จะชำระเงินด้วยบัตรเครดิตบนอินเตอร์เน็ต

ผู้ซื้อซึ่งถือใบรับรองของธนาคารและเข้าไปเลือกซื้อสินค้าหรือบริการบนเว็บไซต์ที่สนับสนุน SET หลังจากเลือกสินค้าหรือบริการใส่ลงในรถเข็นหรือตะกร้าอิเลกทรอนิกส์แล้ว จึงคลิกปุ่มสั่งซื้อทาง Server ของผู้ขายจะส่งใบสั่งซื้อที่ระบุรายการสินค้าหรือบริการพร้อมจำนวนและราคามาแสดงบนหน้าจอผู้ซื้อ ให้ผู้ซื้อกรอกที่อยู่สำหรับส่งสินค้าลงไป แล้วเลือกวิธีการชำระเงิน หากเลือกชำระเงินผ่านระบบ SET โปรแกรมกระเป๋าเงินอิเลกทรอนิกส์จะถูกเรียกขึ้นมาโดยอัตโนมัติ ผู้ซื้อป้อนรหัสผ่านแล้วจึงเลือกบัตรเครดิตที่จะใช้ชำระเงิน คำสั่งซื้อของลูกค้าจะถูกเข้ารหัสโดยใช้กุญแจส่วนตัว (private key) ของผู้ซื้อ ส่วนข้อมูลบัตรเครดิตก็จะถูกเข้ารหัสก่อนส่งให้ธนาคารของผู้ขายเช่นกัน โดยข้อมูลทั้งสองส่วนจะมีการเซ็นลายเซ็นดิจิตอล (Digital Signature) กำกับข้อมูลที่เข้ารหัสแล้วทั้งสองส่วนไว้ด้วยเพื่อยืนยันตัวลูกค้าก่อนจะส่งให้กับผู้ขาย ผู้ขายจะตรวจสอบลายเซ็นดิจิตอลของผู้ซื้อบนใบสั่งซื้อ และส่งต่อข้อมูลการชำระเงินไปยังธนาคารของผู้ขาย เพื่อขออนุมัติการชำระเงิน ธนาคารของผู้ขายจะตรวจสอบกับธนาคารผู้ออกบัตรฯ ว่าข้อมูลบัตรเครดิตถูกต้องและมีวงเงินพอจ่ายหรือไม่ และส่งผลการตรวจสอบไปยังธนาคารของผู้ขาย จากนั้นธนาคารของผู้ขายจะแจ้งผลไปยังผู้ขายอีกทอดหนึ่ง ซึ่งหากข้อมูลถูกต้องและมีวงเงินพอจ่าย ธนาคารผู้ออกบัตรจะบันทึกรายการชำระเงินของเจ้าของบัตรเพื่อเรียกเก็บเงินต่อไป ส่วนธนาคารของผู้ขายจะโอนเงินเข้าสู่บัญชีของผู้ขาย และสุดท้ายผู้ขายจะส่งใบเสร็จให้ผู้ซื้อเก็บไว้ในกระเป๋าเงินอิเลกทรอนิกส์ ในทุกขั้นตอนจะมีการเข้ารหัสข้อมูลระหว่างการส่งทั้งหมด โดยใช้วิธีการเข้ารหัสแบบ DES ซึ่ง DES key จะถูกส่งไปให้ผู้รับในรูปของการเข้ารหัสแบบ RSA โดยใช้ public key ของผู้รับ ทั้งนี้การส่งข้อมูลการชำระเงินใน SET ฝ่ายผู้ซื้อจะมี public/private key เพียงชุดเดียว ในขณะที่ฝ่ายผู้ขายและ payment gateway จะใช้ public/private key 2 ชุด ดังนั้นทั้งผู้ขายและ payment gateway จะต้องมี certificate 2 อัน อันหนึ่งเพื่อใช้ในการแลกเปลี่ยน key และอีกอันเพื่อใช้ในการทำ digital signature

จากการศึกษาการทำงานของ SET และ SSL พบว่าการใช้ SET จะมีขั้นตอนที่มากกว่าและใช้เวลานานกว่า รวมถึงปริมาณข้อมูลที่ต้องส่งระหว่างกัน ทั้งนี้เพราะต้องมีการตรวจสอบความมีตัวตนของแต่ละฝ่าย และวิธีการเข้ารหัสของ SET จะมีความซับซ้อนมากกว่า นอกจากนี้ผู้ที่จะชำระเงินผ่าน SET จะต้องเปิดบัญชีไว้ล่วงหน้าก่อนแล้ว ในขณะที่ SSL เป็นโปรโตคอลที่มีอยู่ในเบราเซอร์ทั่วไป จึงก่อให้เกิดความสะดวกในการใช้ การทำงานของ SSL จะใช้เวลาน้อยกว่า ข้อมูลขนาดเล็กกว่า และปริมาณงานที่น้อยกว่า อย่างไรก็ตาม SSL มีข้อเสียในประเด็นของการปกป้องข้อมูลของลูกค้า เพราะธนาคารจะสามารถอ่านข้อมูลการชำระเงินทุกอย่างได้หมด ในขณะที่ SET จะให้การปกป้องข้อมูล โดยผู้ขายและธนาคารจะได้รับเพียงข้อมูลเท่าที่จำเป็นเท่านั้น ในตารางที่ 2.3 แสดงข้อมูลเปรียบเทียบการทำงานของ SET และ SSL ในประเด็นต่างๆ ดังนี้

ประเด็นที่ใช้เปรียบเทียบ	SSL	SET
จำนวนฝ่ายที่เกี่ยวข้อง	- 2 ฝ่าย (Server และ Browser)	- 3 ฝ่าย (ผู้ซื้อ, ผู้ขาย และธนาคาร)
ใบรับรองดิจิตอล	- มีเฉพาะฝั่ง Server	- ทุกฝ่ายที่เกี่ยวข้องต้องมี
การตรวจสอบ	- Server กับ Browser ต่างตรวจสอบซึ่งกันและกัน	- ต้องมี CA ตรวจสอบทุกฝ่ายที่เกี่ยวข้อง
การป้อนข้อมูลบัตรเครดิต	- ผู้ซื้อต้องป้อนข้อมูลทุกครั้ง	- ข้อมูลบัตรเครดิตเก็บไว้ในE-wallet จึงป้อนเพียงครั้งเดียว
การจำกัดการเข้าถึง	- สามารถควบคุมการเข้าถึง Server Directory, File และบริการต่างๆ	- ธนาคารผู้ออกบัตรไม่ทราบรายละเอียดการซื้อ - ผู้ขายไม่ทราบข้อมูลบัตรเครดิต
การใช้ข้อมูลร่วมกัน	- Browser สามารถใช้ข้อมูลร่วมกับ Server และป้องกันบุคคลที่ 3 ไม่ให้เข้าถึงข้อมูลได้	- คำสั่งซื้อที่เข้ารหัสแล้วถูกส่งให้ผู้ขาย ส่วนข้อมูลบัตรฯที่เข้ารหัสแล้วถูกส่งให้ธนาคารผู้ออกบัตรฯ
การป้องกันข้อมูล	- Server สร้างกุญแจสำหรับการสั่งซื้อแต่ละครั้ง แล้วส่งให้ Browser เพื่อเข้ารหัสคำสั่งซื้อแล้วส่งกลับมา	- ข้อมูลถูกเข้ารหัสด้วยกุญแจส่วนตัวของผู้ซื้อ
การพิสูจน์ตัวตนของลูกค้าและยอดเครดิตแบบทันที	- ไม่สนับสนุน แต่สามารถทำได้โดยเขียนโปรแกรมสำหรับจัดการได้	- สนับสนุน
การเข้ารหัสข้อมูลบัตรเครดิต	- เข้ารหัสรายละเอียดคำสั่งซื้อกับข้อมูลบัตรฯรวมกัน จึงมีความปลอดภัยน้อยกว่า SET	- เข้ารหัสคำสั่งซื้อและข้อมูลบัตรฯ แยกจากกันและเนื่องจากข้อมูลบัตรฯมีขนาดตายตัว จึงเข้ารหัสได้แข็งแรงกว่า

ประเด็นที่ใช้เปรียบเทียบ	SSL	SET
ข้อเสีย	- การเข้ารหัสจะใช้เพียง 40 บิต หรือสูงสุดคือ 128 บิต - รองรับการติดต่อแบบ point-to-point เท่านั้น แต่การชำระค่าสินค้าผ่านบัตรเครดิตจะมีผู้เกี่ยวข้องอย่างน้อย 3 ฝ่าย คือ ผู้ซื้อ, ผู้ขาย และธนาคาร - ผู้ขายสามารถดูข้อมูลบัตรเครดิตของลูกค้าได้ - ไม่มีการยืนยันตัวตนได้ ดังนั้นผู้ขายจึงมีความเสี่ยงในการรับชำระค่าสินค้าผ่าน SSL	- มีค่าใช้จ่ายสูง - การตรวจสอบและรับชำระค่าสินค้าแต่ละรายการต้องใช้เวลามาก - ลูกค้าส่วนใหญ่ยังไม่นิยมใช้ wallet - การนำ SET ที่ไม่ได้ถูกพัฒนามาด้วยกันมาทำงานร่วมกัน ยังอยู่ในระหว่างการทดลอง

ตารางที่ 2.4 เปรียบเทียบคุณสมบัติของ SSL และ SET ในประเด็นต่างๆ

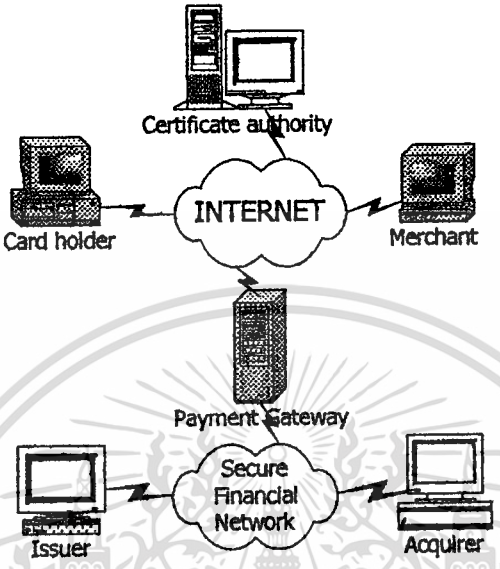
2.8.3 Merchant Originated SET (MOSET)

จากวิวัฒนาการด้านความปลอดภัยในทางอิเลกทรอนิกส์ แม้ว่า SET จะให้การคุ้มครองข้อมูลได้ดีกว่า SSL แต่ข้อเสียของ SET ก็คือมีค่าใช้จ่ายสูงกว่า จึงทำให้ผู้ขายไม่สามารถใช้ประโยชน์จาก SET ได้เท่าที่ควร และเนื่องจากลูกค้าส่วนใหญ่บนอินเตอร์เน็ตยังรู้สึกว่าการชำระค่าสินค้าผ่าน wallet เป็นเรื่องที่ยุ่งยากมากกว่าการชำระผ่านบัตรเครดิตธรรมดา

ระบบ MOSET (หรือ MIA: Merchant Initiated Authorization of Non-SET Orders)เป็นการประยุกต์โดยนำระบบ SET บางส่วนมาใช้ ระบบนี้ถูกคิดค้นและพัฒนาจาก IBM โดยมีวัตถุประสงค์เพื่อให้การซื้อขายทางอิเลกทรอนิกส์เป็นไปอย่างสะดวกมากขึ้น ระบบ MOSET กำหนดให้การรับส่งข้อมูลระหว่าง Web Server ของผู้ขายและ SET Gateway (ธนาคาร) จะเป็นแบบ SET แต่การติดต่อระหว่างผู้ซื้อและผู้ขายไม่ได้กำหนดโปรโตคอลไว้ชัดเจน

จากหัวข้อก่อนหน้านี้ จะเห็นได้ว่าการชำระผ่าน MOSET ถูกพัฒนาขึ้นเพื่อนำข้อดีจากทั้ง SSL และ SET มาผสมผสานกัน ก่อให้เกิดระบบการชำระเงินที่ช่วยให้ลูกค้าสามารถชำระเงินโดยใช้บัตรเครดิตได้เหมือนการชำระผ่านระบบ SSL เดิม (ซึ่งมีอยู่ในเว็บเบราเซอร์ทั่วไป) ทำให้ได้รับความสะดวกในการซื้อ แต่ในขณะเดียวกันการโอนเงินระหว่างธนาคารและผู้ขายจะทำผ่านระบบ SET และมีการตรวจสอบซึ่งกันและกัน (authentication) โดยใช้ประโยชน์จากใบรับรองดิจิตอล ซึ่งเป็นการให้ความคุ้มครองข้อมูลของลูกค้าเหมือนการชำระเงินผ่าน SET ด้วย ดังนั้นในระบบการ

ชำระเงินแบบ MOSET จะมีองค์ประกอบต่างๆ เช่นเดียวกับระบบ SET แต่กระบวนการทำงานและข้อมูลที่รับส่งกันภายในจะแตกต่างออกไป รูปที่ 2.9 แสดงโครงสร้างการติดต่อในระบบ MOSET



รูปที่ 2.9 โครงสร้างการติดต่อในระบบ MOSET

ซึ่งรายละเอียดเกี่ยวกับการทำงานของระบบ MOSET จะถูกอธิบายโดยละเอียดในบทถัดไป

บทที่ 3

หลักการทำงานของระบบ MOSET

3.1 วัตถุประสงค์ของการพัฒนา MOSET

เพื่อให้ผู้ค้าสามารถใช้งานมาตรฐาน SET ได้เหมือนเดิม ในขณะที่สามารถรับคำสั่งซื้อจากลูกค้าที่ไม่ได้ใช้ SET ได้ด้วย

3.2 การทำงานของ MOSET

การชำระเงินในระบบ MOSET นี้ จะใช้ทรัพยากรของระบบโดยรวมเช่นเดียวกับระบบ SET ยกเว้นลูกค้าที่ไม่ต้องมี SET Wallet ติดตั้งที่เครื่องของตน ก็สามารถซื้อสินค้าผ่านระบบนี้ได้ และผู้ขายจะต้องเพิ่มเติมโปรแกรมของตนเพื่อให้สามารถรองรับการชำระเงินจากลูกค้านอกระบบ SET ได้

เมื่อระบบได้รับคำสั่งซื้อจากลูกค้าผ่านการติดต่อแบบ SSL ระบบของผู้ขายจะทำการสร้างรายละเอียดการจ่ายเงินตามมาตรฐาน SET (SET Payment Instruction : PI) จากข้อมูลที่ลูกค้าส่งมา

Payment Instruction ถือเป็นส่วนที่สำคัญที่สุดใน SET ข้อมูลที่อยู่ภายใน PI จะเป็นข้อมูลที่ธนาคารหรือ Payment Gateway ใช้ในการตรวจสอบ (authorize) สิทธิ์ในการจ่ายเงินของผู้ถือบัตร ในระบบ SET ปกติ ข้อมูลส่วน PI จะถูกสร้างและเข้ารหัสจากฝั่งผู้ถือบัตร ส่งผ่านผู้ขายเพื่อทำการส่งต่อไปให้ธนาคารของผู้ขาย (Acquirer) โดยที่ผู้ขายจะไม่สามารถรู้รายละเอียดภายในได้ ยกเว้นธนาคารของผู้ขายจะส่งกลับมาใหม่ให้กับผู้ขายเอง

ตามมาตรฐาน SET นั้น PI จะถูกแบ่งออกเป็น 3 ประเภท คือ

1. PIUnsigned : ถูกสร้างขึ้นโดยผู้ถือบัตรที่ไม่มี signature certificate ดังนั้นความถูกต้องของข้อมูลจะถูกปกป้องโดยการนำ Hash Function เข้ามาช่วย แต่ก็ไม่ได้รับรองตัวตนที่แท้จริงของผู้ทำการสั่งซื้อ
2. PIDualSigned : สร้างขึ้นโดยผู้ถือบัตรที่มี signature certificate ซึ่ง PI ประเภทนี้ให้ความคุ้มครองในเรื่องความปลอดภัยของข้อมูล รวมถึงความน่าเชื่อถือของข้อมูลว่ามาจากผู้ถือบัตรที่แท้จริงได้
3. AuthToken : สร้างขึ้นที่ Payment Gateway ซึ่งเป็นลักษณะ PI ที่ถูกสร้างขึ้นสำหรับการสั่งซื้อที่จะได้รับสินค้าเป็นงวดๆ

โครงสร้างของ Payment Instruction ประกอบด้วยข้อมูลหลายส่วน (ดูภาคผนวก 1.) แต่ส่วนที่สำคัญใน MOSET คือฟิลด์ PIExtensions ซึ่งอยู่ในส่วน PIHead ทำหน้าที่เป็น flag ที่แสดงให้รู้ว่ารายการชำระเงินนี้เป็นแบบ MOSET คือ เป็น SET Message ที่ถูกสร้างขึ้นโดยผู้ขายเอง เมื่อ Payment Gateway ได้รับรายการชำระเงิน จะสามารถรู้ได้ว่า PI นี้สร้างโดยผู้ขายเอง

การสร้าง PI โดยผู้ขาย จะทำในลักษณะเดียวกับ PI ที่สร้างโดยฝั่งผู้ถือบัตร แต่จะมีการเพิ่ม message extension เข้าไปใน Application ของผู้ขาย เพื่อใช้ในการกำหนด flag

Message Extension ที่กำหนดตามมาตรฐาน MOSET มีดังนี้

```
MIAuthExtension DEFINITIONS EXPLICIT TAGS : := BEGIN

-- EXPORTS ALL;
IMPORTS
   AlgorithmIdentifier {}, ALGORITHM-IDENTIFIER
         FROM SetAttribute
EXTENSION, id-set-msgExt
         FROM SetCertificateExtensions;
id-set-miAuth OBJECT IDENTIFIER : := { id-set-msgExt miAuth(3) }
miAuth EXTENSION : := {
   SYNTAX              MIAuth
   IDENTIFIED BY       id-set-miAuth
}
MIAUTH : := SEQUENCE {
   version      INTEGER {miAuthVer1(1) } (miAuthVer1),
   transMethod TransMethod,
   transCrypto TransCrypto OPTIONAL
}
TransMethod : := ENUMERATED {
   channelEncryption          (0),
   unencryptedWWW             (1),
   encryptedEMail             (2),
   unencryptedEMail           (3),
   otherElectronic            (4),
   mail                       (5),
   telephone                  (6),
   fax                        (7),
   faceToFace                 (8),
   otherNonElectronic         (9),
...
}
TransCrypto : := SEQUENCE {
   encrypted          [0] AlgInfo OPTIONAL,
   digitallySigned    [1] AlgInfo OPTIONAL
}
AlgInfo : := SEQUENCE {
   algorithm   AlgorithmIdentifier {(AnyAlgorithm)},
```

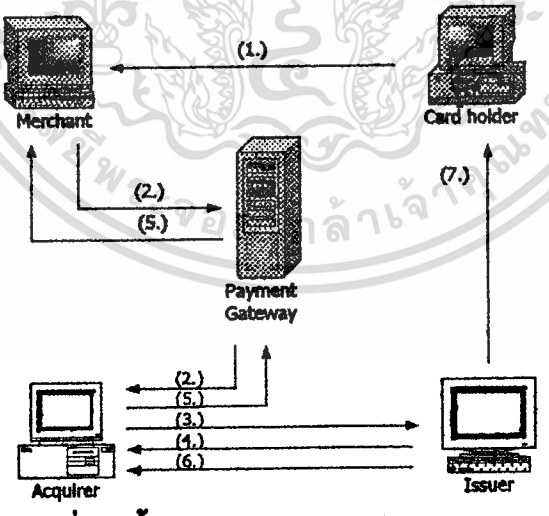
```
    keyLength    INTEGER(1..MAX) OPTIONAL        -- number of bits
}
AnyAlgorithm ALGORITHM-IDENTIFIER : := {
...
}
END
```

3.3 ตัวอย่างระบบที่ใช้มาตรฐาน MOSET

MOSET ถูกพัฒนาขึ้นเพื่อเปลี่ยนแปลงรูปแบบการใช้งานระบบ SET ให้ได้ง่ายขึ้น โดยการสั่งซื้อของลูกค้าจะไม่ได้ถูกจำกัดอยู่เฉพาะ SET ลูกค้าสามารถส่งคำสั่งซื้อผ่านมาตรฐานการชำระเงินในรูปแบบอื่นๆ (ไม่กำหนดตายตัว) ซึ่งในโครงการพัฒนาระบบฯนี้จะใช้การติดต่อผ่านช่องทางของ SSL สำหรับการส่งข้อมูลจากลูกค้าไปยังผู้ขาย โดยระดับของการเข้ารหัสจะขึ้นอยู่กับเว็บเบราเซอร์ของลูกค้าแต่ละราย ซึ่งปัจจุบันระดับสูงสุด คือ 128 บิต แต่ในส่วนของการติดต่อระหว่างผู้ขายกับธนาคาร (หรือ Payment Gateway) จะใช้ SET เป็นมาตรฐานในการรักษาความปลอดภัย เพื่อคงความปลอดภัยในเรื่องข้อมูลการสั่งซื้อสินค้า ที่ธนาคารจะไม่สามารถทราบได้

ผู้เกี่ยวข้องในระบบจะต้องมีใบรับรองดิจิตอลเพื่อยืนยันตัวตนของตนเองกับอีกฝ่าย ยกเว้นลูกค้าที่อาจมีหรือไม่มีก็ได้ เพราะซื้อสินค้าโดยใช้ช่องทางการติดต่อแบบ SSL

รูปที่ 3.1 แสดงตัวอย่างการนำมาตรฐาน MOSET ไปใช้ในระบบรับชำระเงินจริง



รูปที่ 3.1 ขั้นตอนการชำระเงินผ่าน MOSET

จากรูปที่ 3.1 การชำระเงินผ่าน MOSET จะเริ่มจาก

1. ลูกค้าส่งรายละเอียดคำสั่งซื้อและรายละเอียดบัตรเครดิตไปยังผู้ขาย ซึ่งข้อมูลจากผู้ซื้อมาที่ผู้ขายจะถูกส่งโดยเข้ารหัสตามมาตรฐานของ SSL

2. Merchant Server จะส่งข้อมูลบัตรเครดิตไปให้กับธนาคารของผู้ขาย (Acquirer) โดยข้อมูลจะถูก เข้ารหัส ตามมาตรฐานของ SET ผ่าน Payment Gateway
3. ธนาคารผู้ขายจะทำการตรวจสอบบัตรเครดิตจากธนาคารผู้ออกบัตร (Issuer)
4. ธนาคารผู้ออกบัตรแจ้งผลการตรวจสอบให้แก่ธนาคารของผู้ขายว่าบัตรเครดิตนั้นถูกต้องและสามารถนำมาใช้ชำระเงินได้
5. ธนาคารของผู้ขายจะแจ้งผลการตรวจสอบไปให้ผู้ขายทราบ ผ่านPayment Gateway เมื่อผู้ขายรับทราบผลการตรวจสอบบัตรเครดิตว่าถูกต้องแล้ว จะยอมรับคำสั่งซื้อนั้น เพื่อนำไปดำเนินการต่อ
6. ธนาคารผู้ออกบัตรจะทำการโอนเงินเข้าบัญชีของผู้ขาย
7. ผู้ซื้อจะได้รับรายงานการใช้จ่ายผ่านบัตรเครดิตจากธนาคารผู้ออกบัตร

จากกระบวนการทำงานใน MOSET จะพบว่าการติดต่อระหว่างฝั่งผู้ขาย (Merchant Server) กับธนาคารของผู้ขาย (Acquiring Bank) จะยังคงผ่านระบบ SET ดังนั้นข้อมูลที่ผู้ขายได้รับจากผู้ซื้อจึงต้องถูกแปลงให้อยู่ในรูปแบบเดียวกับการชำระเงินผ่านทาง SET ดั้งเดิม โดยจะมีการเพิ่มหน้าที่ของ Merchant Server ในการทำหน้าที่ดังกล่าว เมื่อข้อมูลถูกส่งมาที่ Payment Gateway จะตรวจสอบลักษณะของข้อมูลว่าเป็นการชำระผ่าน SET จริง (Cardholder Initiated Transaction) หรือเป็นเป็น MOSET (คือเป็น PI ที่ถูกสร้างขึ้นโดยผู้ขายเอง) โดย Payment Gateway จะตรวจสอบจาก extension ของ transaction แต่ละรายการ

บทที่ 4

การพัฒนาระบบงาน

4.1 ปัญหาหรืออุปสรรคในระบบการชำระเงินปัจจุบัน

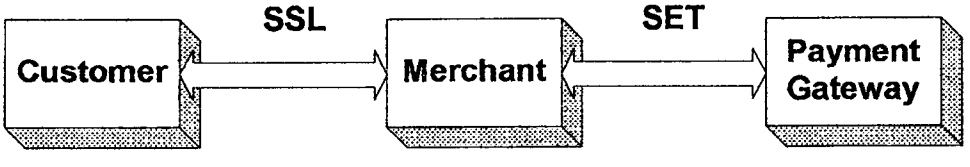
การซื้อขายทางอิเลกทรอนิกส์ส่วนใหญ่ในปัจจุบันให้การคุ้มครองความปลอดภัย ผ่านช่องทางของ SSL หรือ SET ซึ่งมาตรฐานทั้ง 2 ต่างมีจุดอ่อนคนละแบบ คือ SSL ทำได้ง่าย แต่ไม่สามารถให้การคุ้มครองความเป็นส่วนตัวแก่ลูกค้าได้ ส่วน SET แม้จะให้ความปลอดภัยสูง แต่ก็ทำได้ยาก และไม่สะดวก เพราะลูกค้าจะต้องเปิดบัญชีไว้ก่อนจึงจะซื้อสินค้าผ่านระบบ SET ได้

4.2 วิเคราะห์ความต้องการสำหรับการพัฒนาโปรแกรม

จากปัญหาดังกล่าว จะสามารถนำมาใช้กำหนดความต้องการของโปรแกรม ได้ดังนี้

- ผู้ซื้อสามารถซื้อสินค้าทางอินเตอร์เน็ตได้สะดวกเหมือนการใช้มาตรฐาน SSL โดยไม่ต้องมีการเปิดบัญชีไว้ก่อนล่วงหน้าเหมือนกับการใช้ SET
- ผู้ซื้อได้รับการคุ้มครองความปลอดภัยของข้อมูล
- สามารถให้ความคุ้มครองความเป็นส่วนตัวของข้อมูลของลูกค้าได้มากกว่าการใช้ SSL ซึ่งตามมาตรฐานของ MOSET จะให้ความคุ้มครองความเป็นส่วนตัวของลูกค้าในส่วนของข้อมูลการสั่งซื้อสินค้า โดยธนาคารจะไม่สามารถรู้รายละเอียดในการใช้จ่ายเงินของผู้ถือบัตร

สำหรับในโครงการพัฒนาระบบงานนี้ มีข้อจำกัดในเรื่องเวลาและทรัพยากรต่างๆ ในการพัฒนา ดังนั้นโปรแกรมในโครงการนี้จะครอบคลุมเพียงการประมวลคำสั่งซื้อที่ได้รับจากผู้ถือบัตรตามมาตรฐานของ SSL และประมวลผลข้อมูลที่ได้รับเพื่อจัดให้ใช้ได้กับระบบ SET ที่ใช้ติดต่อกับ Payment Gateway ซึ่งแสดงในรูป 4.1



รูปที่ 4.1 แสดงมาตรฐานที่ใช้ในการรับส่งข้อมูลระหว่างผู้เกี่ยวข้อง

4.3 ข้อกำหนดสำหรับโปรแกรม

4.3.1 Input สำหรับโปรแกรม

- คำสั่งซื้อ ซึ่งประกอบด้วยข้อมูลสินค้าที่สั่งซื้อ (OI) และรายละเอียดการชำระเงิน (PI)

4.3.2 หน้าที่ของโปรแกรม

- สร้าง Authorization Request โดยใช้ Order Information แล้วเซ็นลายเซ็นดิจิตอลกำกับไว้ โดยใช้ Private Key ของผู้ขาย : $[AuthReq]_{PR\_M}$
- เข้ารหัส Authorization Request และลายเซ็นดิจิตอล โดยใช้ Public key ของ Payment Gateway : $[[AuthReq]_{PR\_M}]_{PB\_P}$
- สร้าง PI จากข้อมูลการสั่งซื้อที่ได้รับ และเพิ่ม Extension เพื่อแจ้งให้ Payment Gateway รู้ว่าเป็น PI ที่สร้างขึ้นโดยผู้ขาย : PI
- สร้างลายเซ็นดิจิตอลคู่เพื่อแนบไปกับ PI โดย
  - ย่อย OI, PI ใช้ขนาด 160 bits digest : {OI} , {PI}
  - รวม {OI} และ {PI} เข้าด้วยกันแล้วย่อยอีกครั้ง จากนั้นเข้ารหัสโดย Private Key ของผู้ขาย : $\{\{OI\},\{PI\}\}_{PR\_M}$
- เข้ารหัสข้อมูลการชำระเงินและลายเซ็นดิจิตอล โดยใช้ Public key ของ Payment Gateway : $[PI, \{\{OI\},\{PI\}\}_{PR\_M}]_{PB\_P}$

4.3.3 Output จากโปรแกรม

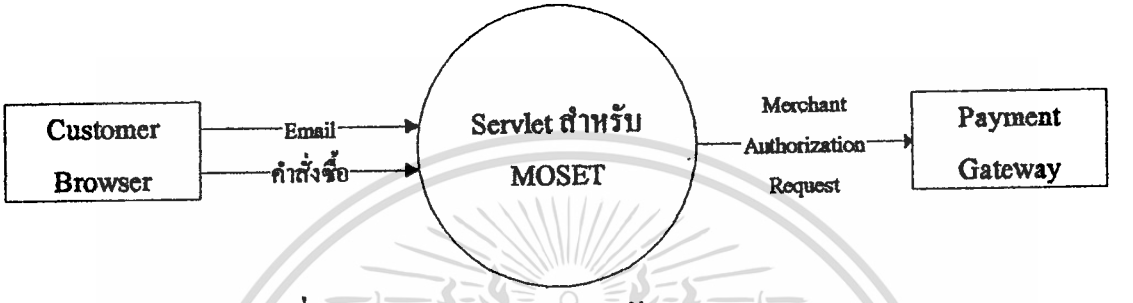
- Merchant Authorization Request ซึ่งประกอบด้วย
  - $[AuthReq , [AuthReq]_{PR\_M}]_{PB\_P}$

    Authorization Request ที่มีลายเซ็นดิจิตอลกำกับและเข้ารหัสโดย Public Key ของ Payment Gateway
  - $[PI, \{\{OI\},\{PI\}\}_{PR\_M}]_{PB\_P}$

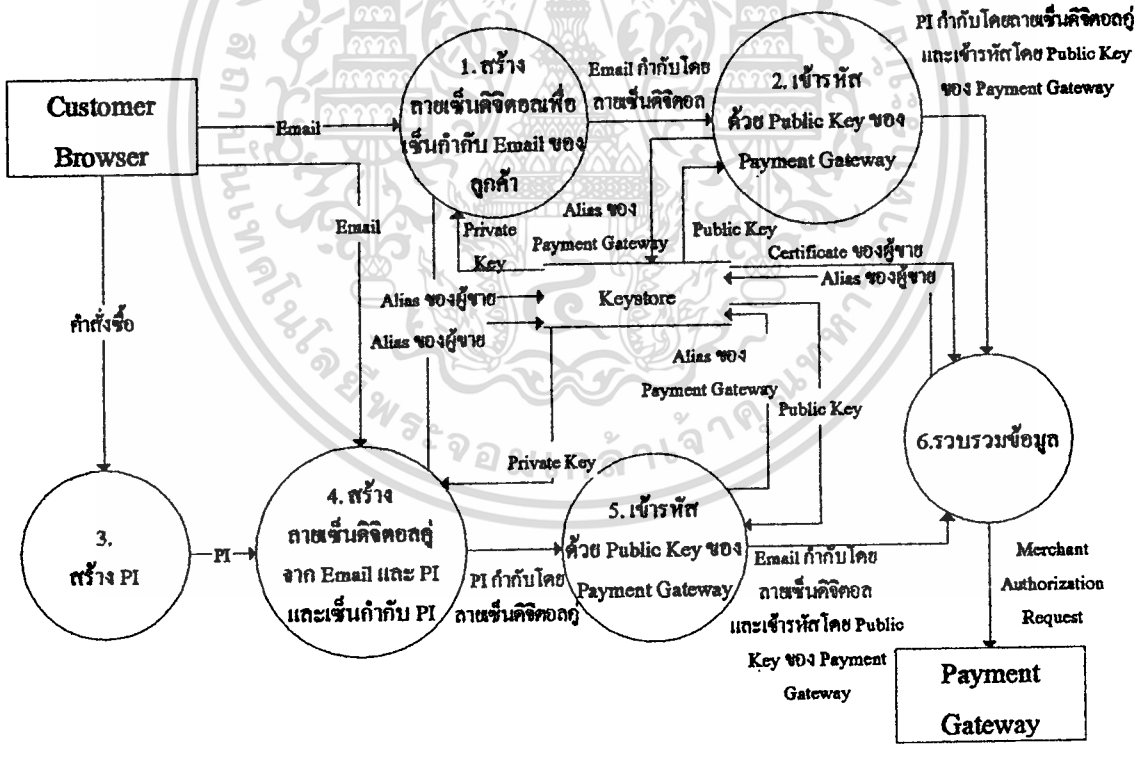
    PI ที่มีลายเซ็นดิจิตอลคู่กำกับและเข้ารหัสโดย Public Key ของ Payment Gateway

4.4 การออกแบบโปรแกรม

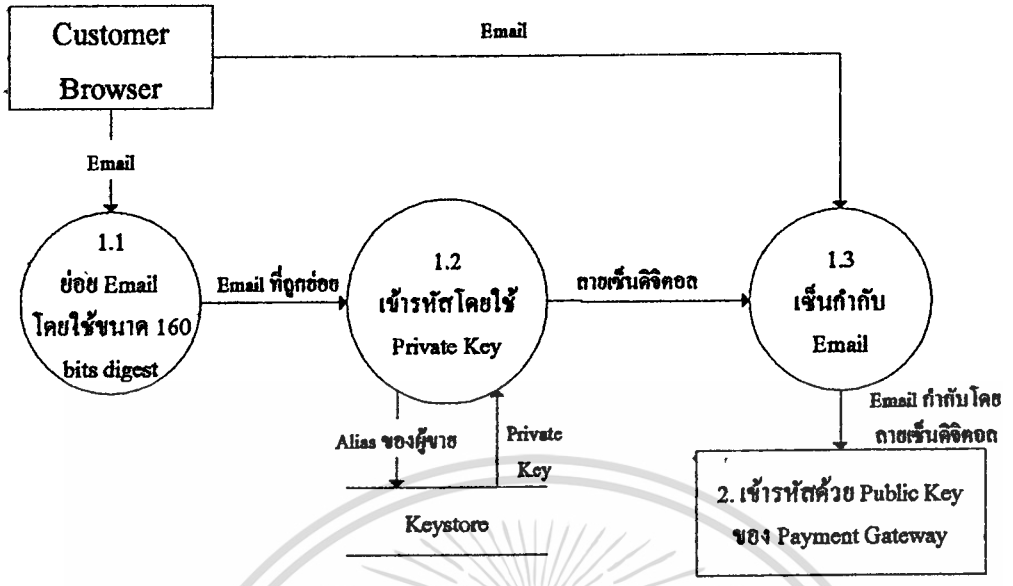
การทำงานของโปรแกรมถูกออกแบบโดยแสดงไว้ใน Context Diagram และ Data Flow Diagrams (DFD) ดังนี้



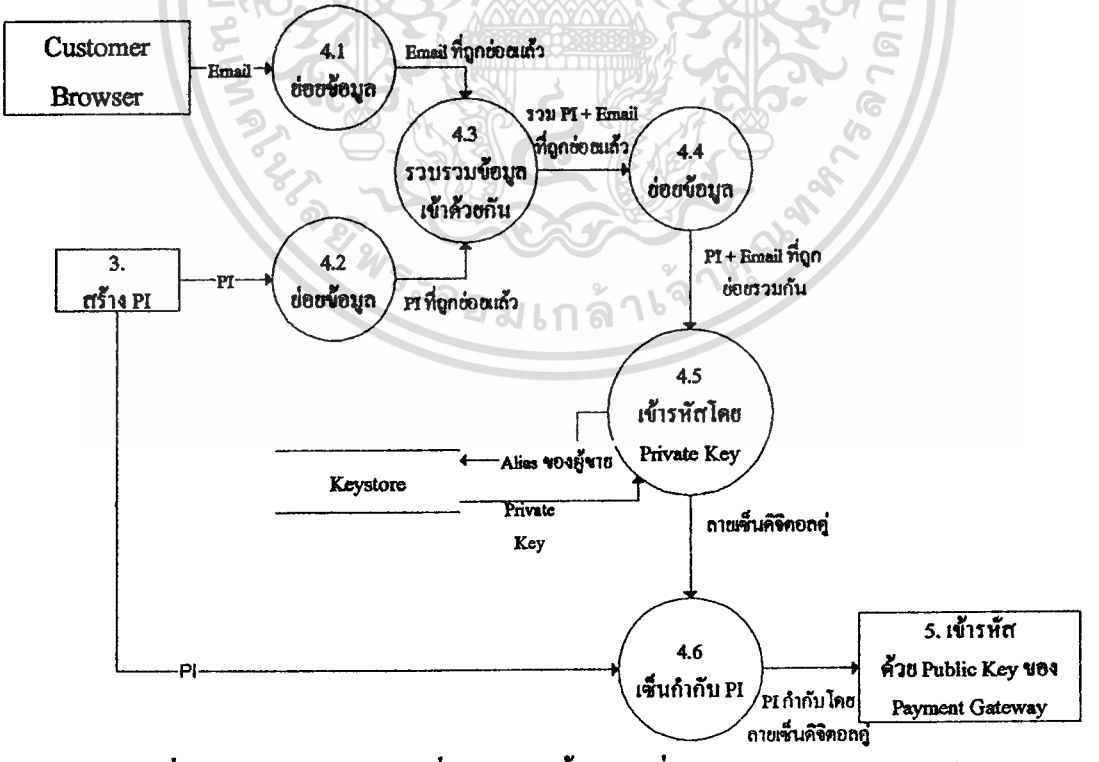
รูปที่ 4.2 Context Diagram สำหรับ MOSET Servlet



รูปที่ 4.3 Data Flow ระดับที่ 1 สำหรับ MOSET Servlet



รูปที่ 4.4 Data Flow ระดับที่ 2 สำหรับขั้นตอนที่ 1. สร้างลายเซ็นดิจิตอลเพื่อเซ็นกำกับ Email ของลูกค้า



รูปที่ 4.5 Data Flow ระดับที่ 2 สำหรับขั้นตอนที่ 4. สร้างลายเซ็นดิจิตอลคู่จาก Email และ PI และเซ็นกำกับ PI

4.5 การพัฒนาโปรแกรม

ในการพัฒนาโปรแกรมในโครงการนี้ สามารถสรุปลักษณะงานที่ต้องพัฒนาขึ้นได้เป็น 4 ส่วน คือ

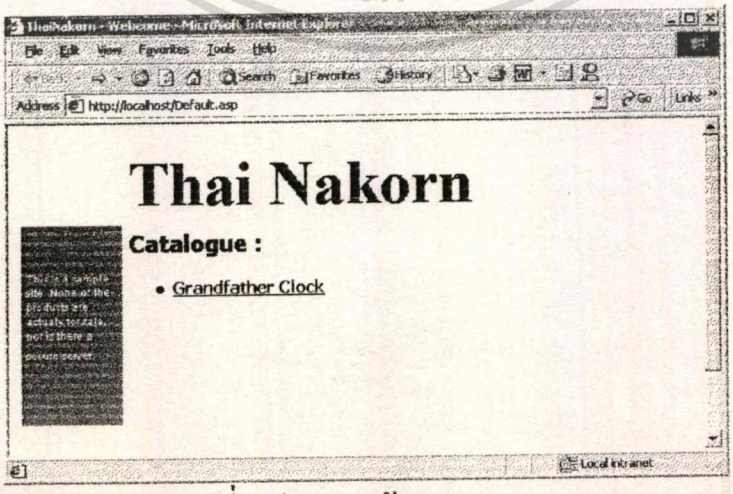
1. Front-end จำลองร้านค้าบนอินเตอร์เน็ต เพื่อรับข้อมูลการสั่งซื้อสินค้า (OI) และข้อมูลการชำระเงิน (PI) จากลูกค้าผ่าน Web Browser รวมทั้งฐานข้อมูลที่เกี่ยวข้อง
2. Servlet เพื่อประมวลผลข้อมูลและส่งต่อให้กับ Payment Gateway
3. สร้างใบรับรองดิจิตอล 2 ใบ สำหรับ Merchant Server และ Payment Gateway Server
4. สร้างโปรแกรมสำหรับรับข้อมูลจาก Servlet และถอดรหัสข้อมูลที่ถูกส่งมา โดยทำหน้าที่เสมือน Payment Gateway จำลองเฉพาะขั้นตอนการรับข้อมูลการชำระเงิน (PI) จากผู้ขาย

ในการพัฒนาโครงงานนี้ จะใช้เครื่องคอมพิวเตอร์ 2 เครื่องเชื่อมโยงถึงกัน แต่ละเครื่องทำหน้าที่เป็น Merchant Server และ Payment Gateway Server สำหรับติดต่อส่งข้อมูลการชำระเงินระหว่างกัน โดยที่ Server ทั้ง 2 จะทำงานบนระบบปฏิบัติการ Windows2000 และใช้บริการ IIS พร้อมกับติดตั้งโปรแกรม Jakarta Tomcat เพื่อให้เครื่องทำหน้าที่เป็น Server ที่สนับสนุนการทำงานของ JAVA Servlet ได้ โดยก่อนการติดตั้ง Jakarta Tomcat ลงในเครื่องนั้น จะต้องมี JDK ติดตั้งไว้อยู่ก่อนแล้ว จึงจะสามารถใช้งาน Tomcat ได้

4.5.1 Front-end ของร้านค้าจำลองที่พัฒนาขึ้น

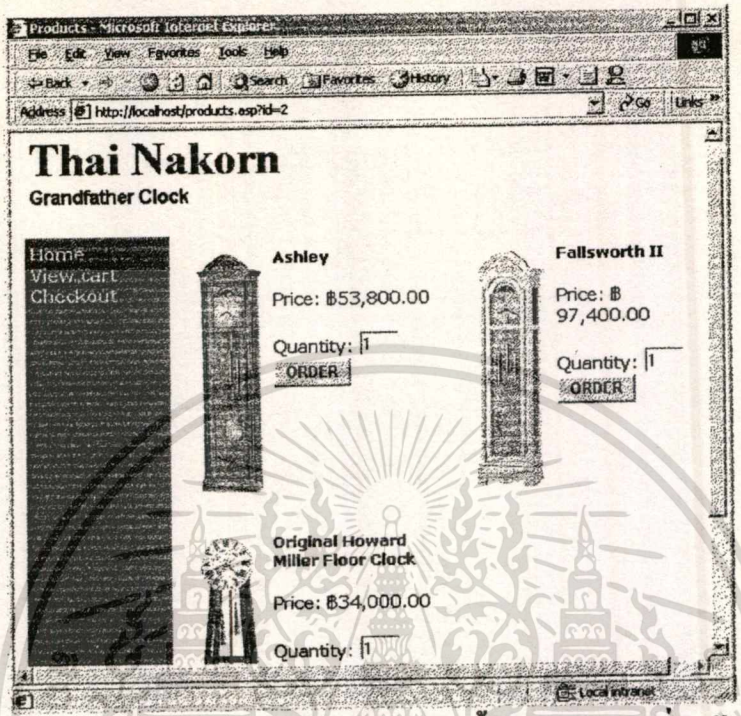
Web Site จำลองถูกพัฒนาขึ้นจากการเขียน ASP ซึ่งมีไฟล์หลักที่สำคัญ 5 ไฟล์ ได้แก่

1. *default.asp* ซึ่งเป็นหน้าร้านให้ลูกค้าเลือกกลุ่มสินค้าที่ต้องการซื้อ กลุ่มสินค้าแต่ละกลุ่มจะมี ID กำกับ เพื่อส่งค่าดังกล่าวไปให้หน้าต่อไป โดยใช้ method GET



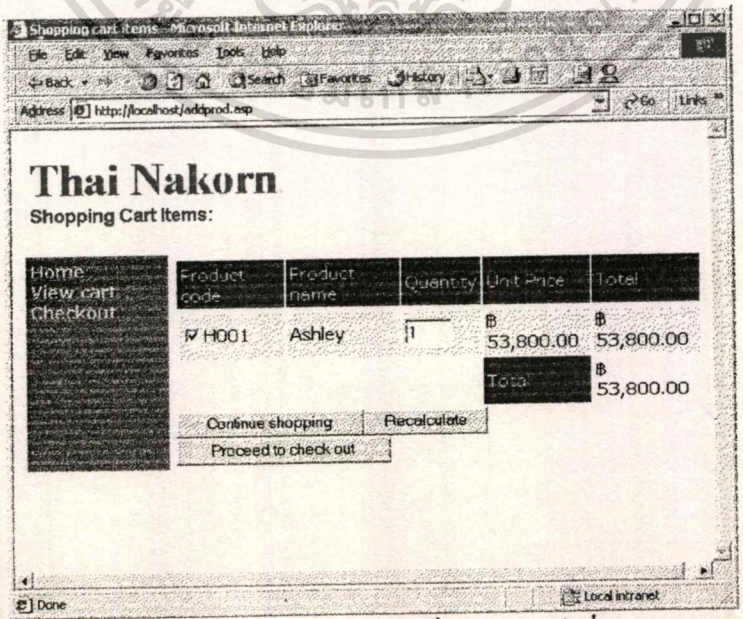
รูปที่ 4.6 แสดงหน้า default.asp

2. *product.asp* รับค่า ID จากหน้า default.asp แล้วแสดงสินค้าภายในกลุ่มที่ลูกค้าเลือกไว้



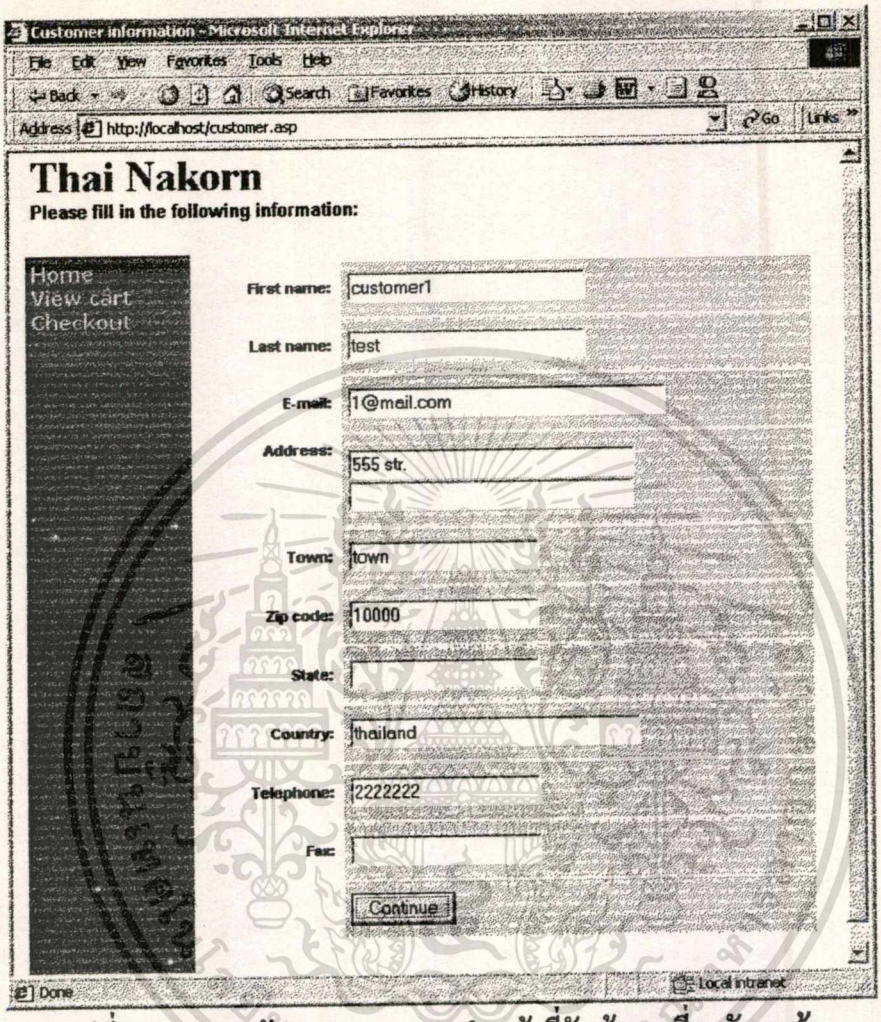
รูปที่ 4.7 แสดงหน้า product.asp สำหรับเลือกซื้อสินค้าในกลุ่มที่ลูกค้าเลือก

3. *addprod.asp* แสดงตะกร้าสินค้าที่ลูกค้าได้เลือกสินค้าไว้ พร้อมทั้งคำนวณมูลค่าของสินค้าตามจำนวนที่ลูกค้าระบุ



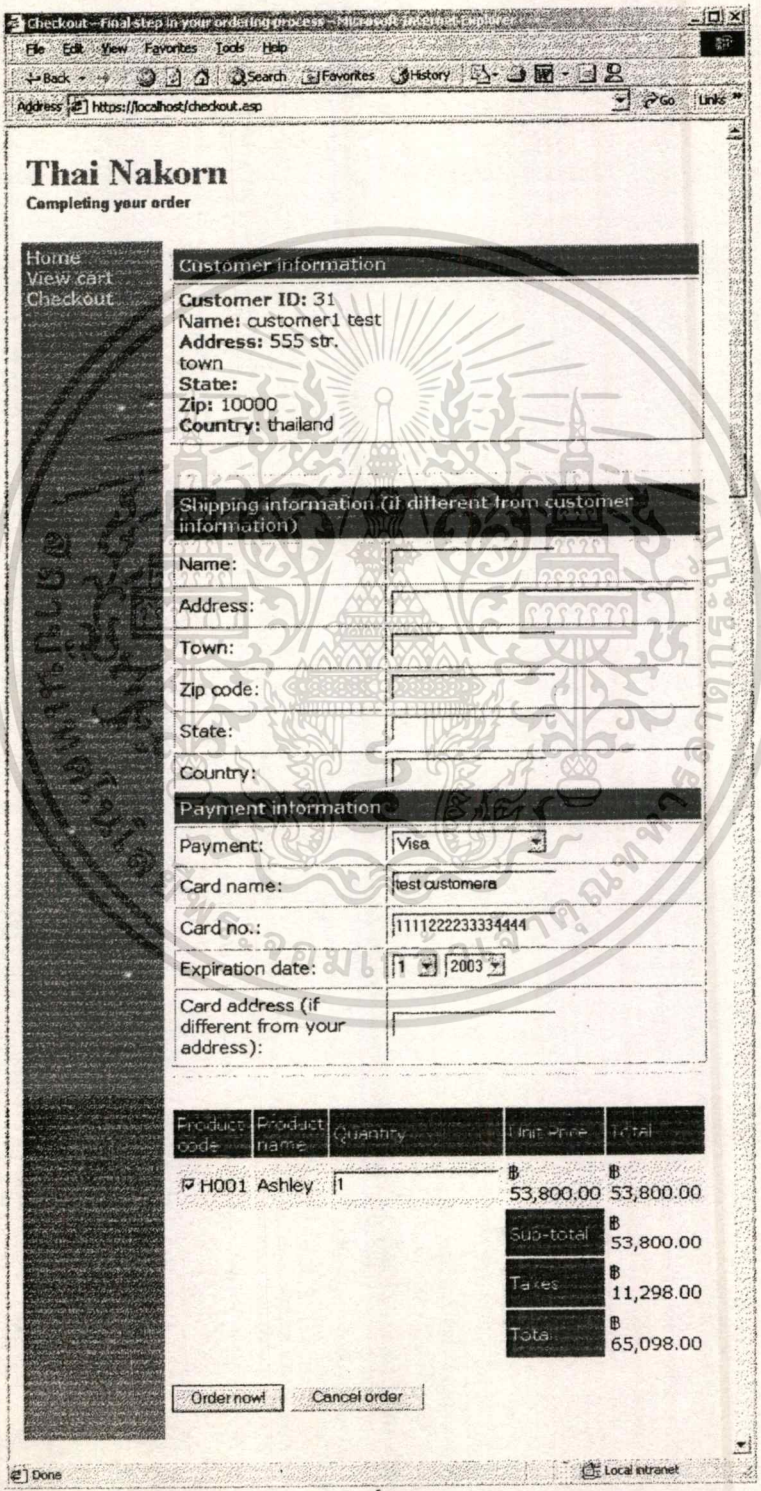
รูปที่ 4.8 แสดงหน้า addprod.asp เพื่อสรุปสินค้าที่ถูกเลือกไว้

4. *customer.asp* ทำหน้าที่รับข้อมูลเบื้องต้นของลูกค้าเพื่อส่งไปเก็บยังฐานข้อมูลลูกค้า



รูปที่ 4.9 แสดงหน้า customer.asp ทำหน้าที่รับข้อมูลเกี่ยวกับลูกค้า

5. *checkout.asp* แสดงข้อมูลที่ลูกค้าป้อนเข้ามาจากหน้าที่แล้ว และให้ลูกค้ากรอกข้อมูลในการชำระค่าสินค้า(PI) เพื่อส่งข้อมูลดังกล่าวไปให้ Servlet ประมวลผลต่อ นอกจากนี้จะเก็บ Email address ของลูกค้าที่ได้รับจากหน้าที่แล้วสำหรับใช้เป็น OI เพื่อส่งให้กับ Servlet เช่นกัน



รูปที่ 4.10 แสดงหน้า checkout.asp เพื่อรับข้อมูลการชำระเงิน (PI) จากลูกค้า

ข้อมูลในแบบฟอร์มจะถูกส่งต่อให้กับ Servlet เพื่อทำการประมวลผลและส่งต่อไปให้ยัง Payment Gateway ต่อไป โดยกำหนดโปรโตคอลที่ใช้ในการส่งข้อมูลไปให้ Servlet ผ่านช่องทางของ Secure Socket Layer หรือ SSL ไว้ใน tag FORM ดังนี้

<FORM action="https://localhost:8443/examples/servlet/EncryptToSet" method=post name="cform" onSubmit="return validate(cform)">

ข้อมูลที่ถูกกำหนดให้ส่งทั้งหมด เมื่อผ่านการตรวจสอบความถูกต้องตามข้อกำหนดใน script แล้วจะถูกส่งไปให้ Servlet ชื่อ EncryptToSet ผ่านโปรโตคอล https ไปยังตำแหน่ง path ตามที่ถูกระบุไว้ คือ localhost:8443/examples/servlet/ โดยใช้ method POST สำหรับการส่งข้อมูล

4.5.2 Servlet ที่ถูกพัฒนาขึ้นบน Merchant Server

ขั้นตอนการทำงานของ Servlet จะอ้างถึง Data Flow Diagrams ที่แสดงไว้ในหัวข้อ 4.3 โดยมี Input ของโปรแกรมได้แก่

- Order Information : ใช้ Email ของลูกค้า
- Payment Instruction ประกอบด้วย
  - ชนิดของบัตรเครดิตที่ใช้ (Payment Method)
  - ชื่อผู้ถือบัตรเครดิต (Card Name)
  - หมายเลขบัตรเครดิต (Card Number)
  - วันหมดอายุของบัตรเครดิต (Expiration Date)
  - ที่อยู่ของผู้ถือบัตรเครดิต (Card Address)

เนื่องจากการทำงานของ Servlet ดังกล่าวจะต้องทำหน้าที่ในการย่อยข้อมูล เข้ารหัสแบบ Asymmetric ดังนั้นจึงต้องมีการ import package ชื่อ java.security.* ซึ่งเป็น package ที่รองรับการทำงานดังกล่าว โดยสามารถดาวน์โหลด package ดังกล่าวได้จากเว็บไซต์ของบริษัท Sun Microsystem ไฟล์ที่ดาวน์โหลดมาได้จะอยู่ในรูปแบบไฟล์ชนิด .jar ซึ่งนำไปเก็บไว้ในไดเรกทอรี่ย่อยชื่อ Security ภายในไดเรกทอรี่ของ jdk ที่ติดตั้งไว้ในเครื่องได้เลย

ผลลัพธ์ที่ได้ในแต่ละขั้นตอน จะเป็นดังนี้

ขั้นตอนที่ 1 : สร้างลายเซ็นดิจิตอลเพื่อเซ็นกำกับ Order Information (OI)

การพัฒนาโปรแกรมในโครงการนี้จะใช้ Email ของลูกค้าเป็น OI ขั้นตอนนี้จะประกอบด้วยการทำงาน 2 ขั้นตอนย่อย (อ้างถึงรูปที่ 4.4) คือ

- 1.1 ย่อย Email

โดยในโครงการนี้จะกำหนด algorithm ในการย่อยข้อมูลเป็น SHA-1 ซึ่งเป็น default algorithm อยู่แล้วสำหรับ package java.security.* โดยมี Provider คือ "Sun" แต่

หากต้องการใช้ algorithm อื่นจะต้องมีการติดตั้ง provider ตัวใหม่ที่สนับสนุนวิธีการย่อยข้อมูลในรูปแบบอื่น

ตัวอย่าง source code สำหรับ class ชื่อ toDigest() ซึ่งทำหน้าที่ในการย่อยข้อมูล โดยการรับข้อมูลมาจาก main() ในรูปของ String แล้วส่งกลับผลลัพธ์ในรูปของ String เป็นดังนี้

```
public String toDigest(String n) {
        //convert string input into byte array.
        byte[] pByte = n.getBytes();

        //digest Payment Instruction by using SHA-1 algorithm.
        if (pByte != null) {
                MessageDigest algorithm;
                try {
                        algorithm = MessageDigest.getInstance("SHA-1");
                } catch (NoSuchAlgorithmException e) {
                        return null;
                }
                //assign the string to be digested.
                algorithm.update (pByte);
                byte[] pDigest = algorithm.digest();
                StringBuffer hexString = new StringBuffer();
                int pDigestLength = pDigest.length;
                for (int i=0;i<pDigestLength;i++) {
                        hexString.append (hexDigit(pDigest[i]));
                        hexString.append (" ");
                }
                return hexString.toString();
        } else {
                return "";
                }
}
}
```

จากตัวอย่างข้างต้น จะเห็นได้ว่า class ชื่อ toDigest จะมีการเรียกใช้ class อีกตัวชื่อ hexDigit ซึ่งจะทำหน้าที่ในการนำ input string ที่ได้รับมาคำนวณทางคณิตศาสตร์ เพื่อย่อยข้อมูลนั่นเอง ผลลัพธ์ที่ได้จากจาก toDigest() เป็นดังรูปที่ 4.11 ข้างล่างนี้

```
Oldigested : f2 a5 b7 e3 15 5f 2c 77 f6 b0 b7 d3 18 5b e8 72 15 d6 ed f7
```

รูปที่ 4.11 แสดงตัวอย่างข้อมูล OI ที่ถูกย่อย

- 1.2 เข้ารหัสโดยใช้ Private Key

  Private Key ที่นำมาใช้ในการเข้ารหัสในขั้นตอนนี้จะนำมาจากไฟล์ .keystore ภายในเครื่อง Merchant Server โดยแต่ละ private key จะมีชื่อเรียก (alias) ที่ใช้ในการอ้างถึง

  ตัวอย่าง source code สำหรับขั้นตอนการเซ็นกำกับข้อมูล OI เป็นดังนี้

```
/*Sign data using private key from keystore
        //create object of signature
        Signature signObj = Signature.getInstance("SHA1withDSA");
        //retrieve private key from keystore using alias "merchant".
        Key priv = pair.getKey(merchant);
        //provide key for signing
        signObj.initSign(priv);

        //Assign data to be signed by the private key
        byte[] dataOI = Oldigested.getBytes();
        signObj.update(dataOI);
        byte[] sig = signObj.sign();
```

Input สำหรับส่งเข้ามาเซ็นลายเซ็นดิจิตอลจะต้องถูกแปลงให้อยู่ในรูป byte array ก่อน จึงจะสามารถนำไปใช้ทำลายเซ็นดิจิตอลได้ เช่นเดียวกันกับผลลัพธ์ที่จะได้จากการทำลายเซ็นดิจิตอล ก็จะอยู่ในรูปของ byte array เช่นกัน

ผลลัพธ์ที่ได้ เรียกว่า ลายเซ็นดิจิตอล เพื่อส่งไปคู่กับ OI จากขั้นตอนที่ 1.1 เพื่อใช้ตรวจสอบความถูกต้องของข้อมูล

```
[B@493f
```

รูปที่ 4.12 แสดงตัวอย่างลายเซ็นดิจิตอลที่ถูกสร้างขึ้น

ขั้นตอนที่ 2 : เข้ารหัสด้วย Public Key ของ Payment Gateway

นำ OI และลายเซ็นดิจิตอลที่ได้มาเข้ารหัสอีกครั้งด้วย Public Key ของ Payment Gateway โดยดึงค่า Public Key มาจากใบรับรองดิจิตอลที่เก็บไว้ในเครื่อง คำสั่งที่ใช้ในการเซ็นกำกับข้อมูลจะเป็นเช่นเดียวกับในขั้นตอนที่ 1 ในการเซ็นกำกับข้อมูลโดยใช้ Private Key แต่จะเปลี่ยนจาก Private Key มาใช้เป็น Public Key ในการเข้ารหัสแทน

```
[B@6a8f
```

รูปที่ 4.13 แสดง OI และลายเซ็นดิจิตอลที่ถูกเข้ารหัสโดย Public Key ของ Payment Gateway

ขั้นตอนที่ 3 : สร้าง Payment Instruction (PI)

ข้อมูลที่ได้รับจากลูกค้าจะถูกคัดเลือกและจัดเรียงใหม่ เพื่อให้ Payment Gateway สามารถนำไปใช้ประมวลผลต่อได้

Parameter Name	Parameter Value(s)
expMonth	1
paymentm	Visa
cardaddress	No Value
Cardno	1111222233334444
cardname	test customera
expYear	2003
Extention	MOSET

รูปที่ 4.14 แสดงตัวอย่าง Payment Instruction

ขั้นตอนที่ 4 : สร้างลายเซ็นดิจิตอลคู่จาก Email และ PI และเซ็นกำกับ PI

ขั้นตอนนี้จะประกอบด้วยการทำงาน 6 ขั้นตอนย่อย (อ้างถึงรูปที่ 4.5) คือ

- 4.1 ย่อยข้อมูล (OI)

  ผลลัพธ์ที่ได้คือ OIdigested ซึ่งแสดงไว้โดยรูปที่ 4.11

- 4.2 ย่อยข้อมูล (PI)

  ผลลัพธ์ที่ได้คือ PIdigested

```
PIdigested : df 8b bc 36 55 ee 5c ff 87 31 e5 70 80 1d 0d 82 90 7b 91 1d
```

รูปที่ 4.15 แสดงตัวอย่างข้อมูล PI ที่ถูกย่อย

- 4.3 รวบรวมข้อมูลเข้าด้วยกัน

  โปรแกรมจะทำการรวมผลลัพธ์ที่ได้จากข้อ 4.1 และ 4.2

- 4.4 ย่อยข้อมูล

  ทำการย่อยผลลัพธ์ที่ได้จากข้อ 4.3 อีกครั้ง

```
OIPIdigested : a2 43 e6 da f5 b4 21 02 8b 10 cd 5a 03 64 5f c1 9f 08 53 a6
```

รูปที่ 4.16 แสดงตัวอย่างข้อมูลที่ถูกย่อยรวมกันอีกครั้ง

- 4.5 เข้ารหัสโดย Private Key

  ผลลัพธ์ที่ได้ คือ ลายเซ็นดิจิตอลคู่

```
[B@5579
```

รูปที่ 4.17 แสดงตัวอย่างลายเซ็นดิจิตอลที่ถูกสร้างขึ้น

ขั้นตอนที่ 5 : เข้ารหัสด้วย Public Key ของ Payment Gateway

นำ PI และลายเซ็นดิจิตอลคู่ที่ได้มาเข้ารหัสอีกครั้งด้วย Public Key ของ Payment Gateway โดยดึงค่า Public Key มาจากใบรับรองดิจิตอลที่เก็บไว้ในเครื่อง คำสั่งที่ใช้ในการเซ็นกำกับข้อมูลจะเป็นเช่นเดียวกับในขั้นตอนที่ 1 ในการเซ็นกำกับข้อมูลโดยใช้ Private Key แต่จะเปลี่ยนจาก Private Key มาใช้เป็น Public Key ในการเข้ารหัสแทน

ขั้นตอนที่ 6 : รวมรวมข้อมูลและส่งให้ Payment Gateway

ข้อมูลที่ส่งให้กับ Payment Gateway เป็นผลลัพธ์ที่ได้จากขั้นตอนที่ 5 และ 2 และใบรับรองดิจิตอลของผู้ขายอีก 1 ใบ

4.5.3 การสร้างใบรับรองดิจิตอล

ใบรับรองดิจิตอลที่สร้างขึ้นจะถูกนำไปใช้ในการเข้า/ถอดรหัสข้อมูลเพื่อใช้ในการติดต่อระหว่าง Merchant Server กับ Payment Gateway โดย Private Key ของตนเองและใบรับรองดิจิตอลของผู้อื่นจะถูกเก็บไว้ที่ไฟล์ .Keystore ของแต่ละเครื่อง ซึ่งข้อมูลที่ถูกเก็บไว้ในไฟล์นี้จะถูกเข้ารหัสเอาไว้และจะสามารถเข้าถึง Private Key ของตนเอง หรือ Public Key ของผู้อื่นได้โดยระบุ alias ของกุญแจแต่ละชุดด้วย

```
Sun DSA Private Key
parameters:
    p:
    fd7f5381 1d751229 52df4a9c 2eece4e7 f611b752 3cef4400 c31e3f80 b6512669
    455d4022 51fb593d 8d58fabf c5f5ba30 f6cb9b55 6cd7813b 801d346f f26660b7
    6b9950a5 a49f9fe8 047b1022 c24fbba9 d7feb7c6 1bf83b57 e7c6a8a6 150f04fb
    83f6d3c5 1ec30235 54135a16 9132f675 f3aa2b61 d72aeff2 2203199d d14801c7
    q:
     9760508f 15230bcc b292b982 a2eb840b f0581cf5
    g:
    f7e1a085 d69b3dde cbbcab5c 36b857b9 7994afbb fa3aea82 f9574c0b 3d078267
    5159578e bad4594f e6710710 8180b449 167123e8 4c281613 b7cf0932 8cc8a6e1
    3c167a8b 547c8d28 e0a3ae1e 2bb3a675 916ea37f 0bfa2135 62f1fb62 7a01243b
    cca4f1be a8519089 a883dfe1 5ae59f06 928b665e 807b5525 64014c3b fecf492a

x: 10f1899472c3a41ef562aef8290004bdd51aca5b
```

รูปที่ 4.18 แสดงตัวอย่าง Private Key ของใบรับรองดิจิตอลที่ถูกสร้างขึ้น

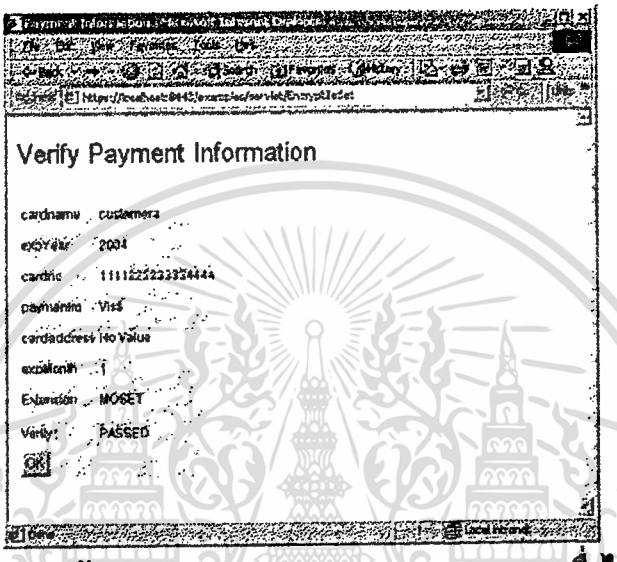
```
Sun DSA Public Key
    Parameters:
    p:
    fd7f5381 1d751229 52df4a9c 2eece4e7 f611b752 3cef4400 c31e3f80 b6512669
    455d4022 51fb593d 8d58fabf c5f5ba30 f6cb9b55 6cd7813b 801d346f f26660b7
    6b9950a5 a49f9fe8 047b1022 c24fbba9 d7feb7c6 1bf83b57 e7c6a8a6 150f04fb
    83f6d3c5 1ec30235 54135a16 9132f675 f3ae2b61 d72aeff2 2203199d d14801c7
    q:
    9760508f 15230bcc b292b982 a2eb840b f0581cf5
    g:
    f7e1a085 d69b3dde cbbcab5c 36b857b9 7994afbb fa3aea82 f9574c0b 3d078267
    5159578e bad4594f e6710710 8180b449 167123e8 4c281613 b7cf0932 8cc8a6e1
    3c167a8b 547c8d28 e0a3ae1e 2bb3a675 916ea37f 0bfa2135 62f1fb62 7a01243b
    cca4f1be a8519089 a883dfe1 5ae59f06 928b665e 807b5525 64014c3b fecf492a

y:
    b1389c17 745a8973 a71791bb 9fad7fc7 1b1c7ce8 6ee81cb9 d7900f05 07183ccc
    356cc76d c11c22ef a3bcded9 99a5d514 377d6695 25faa738 98051dbe a481c6fd
    607845a9 e8ee439d f4cfee06 0ef309d6 2bbcadfd 950980c8 99474ac0 a75070a6
    2f7d7733 9e62c95e 2c5662cb cb306fc3 20b4c4dd 91cb3aee d525b179 e30a4476
```

รูปที่ 4.19 แสดงตัวอย่าง Public Key ของใบรับรองดิจิตอลที่ถูกสร้างขึ้น

4.5.4 การทำงานของ Servlet ที่ Payment Gateway Server

การทำงานของโปรแกรมที่ Payment Gateway Server สำหรับโครงการพัฒนาระบบนี้จะทำหน้าที่เพียงรับข้อมูลที่ผู้ขายส่งมาให้, ตรวจสอบใบรับรองดิจิตอลที่ได้รับเพื่อตรวจสอบตัวตนของผู้ขายก่อน จากนั้นจะทำการถอดรหัสข้อมูลเพื่อดูรายละเอียดของ Payment Instruction สำหรับนำไปใช้ประมวลผลต่อไป

โปรแกรมถูกพัฒนาโดยใช้ภาษาจาวาอยู่ในรูปของ Servlet ซึ่งจะทำหน้าที่ย้อนกลับขั้นตอนของ Servlet ในฝั่ง Merchant Server แต่จะมีการตรวจสอบใบรับรองดิจิตอลของผู้ขายที่ถูกแนบมาพร้อมกับข้อมูลก่อน ว่าเป็นใบรับรองที่ถูกต้อง จากนั้นจึงจะทำการถอดรหัสข้อมูล



รูปที่ 4.20 แสดงหน้าจอของ Server ของ Payment Gateway เมื่อได้รับข้อมูล

4.6 ความแตกต่างของระบบที่พัฒนากับการใช้งานจริง

4.6.1 ความสามารถในการรองรับรูปแบบการติดต่อกับลูกค้า

ระบบจริง : ผู้ขายสามารถรับชำระเงินผ่านช่องทางอื่นๆได้อีก โดยอาจควบคู่กับการชำระเงินผ่านระบบ SET เดิมก็ได้ โดยจะมีตัวเลือกให้ลูกค้าก่อนการสั่งซื้อ เพื่อให้ลูกค้าสามารถเลือกระบบชำระเงินได้ว่าจะชำระเงินผ่าน SET Wallet หรือผ่าน SSL

ระบบที่พัฒนา : รองรับการชำระเงินผ่าน SSL จากลูกค้าเท่านั้น

4.6.2 ข้อมูลการสั่งซื้อสินค้าที่ส่งให้ Payment Gateway

ระบบที่พัฒนา : ใช้ Email ของลูกค้าเป็น OI

4.6.3 การเข้ารหัสข้อมูล

ระบบจริง : สร้าง Symmetric Key ขึ้นเพื่อเข้ารหัสข้อมูลและลายเซ็นดิจิตอลร่วมกัน ก่อนจะนำ Symmetric Key ที่ใช้ไปเข้ารหัสอีกครั้งด้วย Public Key ของ Payment Gateway

ระบบที่พัฒนา : ไม่มีการสร้าง Symmetric Key แต่ทำการเข้ารหัสข้อมูลและลายเซ็นดิจิตอลร่วมกันโดยใช้ Public Key ของ Payment Gateway

4.7 ข้อจำกัดของระบบที่พัฒนา

เนื่องจากโปรแกรมถูกพัฒนาโดยใช้ภาษาจาวา ดังนั้นจึงจำเป็นต้องใช้ Server ที่สนับสนุนการทำงานของ Servlet นอกจากนี้ Payment Gateway ที่จำลองขึ้นยังทำหน้าที่ไม่ได้เหมือนกับ Payment Gateway จริงๆ ซึ่งมีการทำงานที่ซับซ้อนกว่านี้ โดยลักษณะการทำงานและรูปแบบข้อมูลของ Payment Gateway แต่ละแห่งก็จะแตกต่างกันออกไปด้วย

บทที่ 5
สรุป

5.1 ผลการพัฒนา Application

โครงการพัฒนาระบบงานนี้อ้างอิงมาตรฐานของ MOSET ซึ่งเป็นมาตรฐานในระบบการชำระเงินทางอิเลกทรอนิกส์มาใช้ในการพัฒนาโปรแกรมสำหรับรับชำระเงินโดยบัตรเครดิตจากลูกค้าผ่าน SSL

โปรแกรมในโครงการนี้ถูกพัฒนาขึ้นโดยใช้ภาษาจาวา และนำ Package ต่างๆ ในด้านความปลอดภัยมาใช้สำหรับการย่อยข้อมูล, สร้าง Asymmetric Key, การเข้า/ถอดรหัสข้อมูล ซึ่งโปรแกรมที่พัฒนาขึ้นมาสามารถจำลองการทำงานของระบบการชำระเงินจริงได้เพียงบางส่วน และทำงานในส่วนของผู้ขาย ในขั้นตอนการขอตรวจสอบการชำระเงินจาก Payment Gateway เท่านั้น การให้ความคุ้มครองความปลอดภัยรวมถึงรูปแบบข้อมูลและความซับซ้อนในการทำงานยังมีไม่มากเท่าระบบจริง

5.2 ข้อเสนอแนะ

โครงการพัฒนาระบบงานนี้ยังควรได้รับการปรับปรุงการทำงานเพื่อคุ้มครองความปลอดภัยให้รัดกุมมากยิ่งขึ้น โดยการเพิ่มการเข้ารหัสโดยใช้ Symmetric Key สำหรับการส่งคำร้องขอตรวจสอบการชำระค่าสินค้า และข้อมูลการชำระเงินหลังจากเซ็นกำกับด้วยลายเซ็นดิจิตอลและลายเซ็นดิจิตอลคู่แล้ว ทั้งนี้เพื่อให้เป็นไปตามรูปแบบการทำงานร่วมกับระบบ SET จริงๆ นอกจากนี้ควรมีการสร้างสภาพแวดล้อมสำหรับการทดลองการทำงานโปรแกรมโดยใช้สภาพแวดล้อมที่เป็นจริงมากกว่านี้ เช่น การพัฒนาโปรแกรมเพื่อให้ทำงานได้บนระบบ SET แบบเต็มรูปแบบ ที่เดิมสามารถรองรับการชำระค่าสินค้าผ่าน SET Wallet ได้อยู่แล้ว

บรรณานุกรม

เปรียบเทียบวิธีการทำงานระหว่าง SSL และ SET. มกราคม 2544. [Online].
เข้าถึงได้จาก : http://www.solaris2000.f2s.com/knowledge/sslvsset.php

E-Commerce 101. [Online]. Available : http://www.ipeir.com/ecom101.html

Corebus e-business solutions. 2001. **E-Business Glossary.** [Online].
Available : http://www.corebus.com.au/glossary.htm

Sun Microsystems, Inc.. January 2002. **Fundamentals of Java Security.** [Online]
Available : http://developer.java.sun.com/developer/onlineTraining/Security/ Fundamentals/ Security.html

Goncalves, M. et. al. 1997. **Internet Privacy Kit.** Indiana : Que Corporation.

MyEG. February 2001. **I will never buy online!.** [Online].
Available : http://www.myeg.com.my/article.php?sid=10

Messmer, E.. March 1999. **MasterCard, Visa trade strong security for ease of use.** Network World. [Online]. Available : http://www.nwfusion.com/archieve/1999/61423_03-22-1999.html

SETco. 2001. **Merchant Initiated Authorization of Non-SET Orders.** [Online].
Available : http://www.setco.org/download/mia-v2.doc

ADFM. 2001. **PayDirect utilizes MOSET.** [Online].
Available : http://www.adfm.org.my/adfmnew//adfm_middleframe_MOSET.htm

Sun Microsystems, Inc.. January 2001. **Security in Java 2 SDK 1.2.** [Online]
Available : http://java.sun.com/docs/books/tutorial/security1.2/index.html

Sherif, M.. **"SET and SSL: Electronic payments on the Internet".** 353-358. In the third IEEE Symposium. 1998.

Merkow, S. Mark. 2000. **SET For A New Horizon.** [Online].
Available : http://news.sawaal.com/expertsays/guest/index105.htm

SSL vs. SET: Private Lives and Public Keys. [Online].
Available : http://www.sanford.com/ism4480/notes/sslvssset.htm

Strategic Recommendations. *Project 1: Requirements*. [Online].
Available : http://www.sims.berkeley.edu/courses/is224/s99/GroupB/project1/report/section5.htm

Tylee, L. Alan. Dec1997. **Virtual Cash – Payments on the Internet.** [Online].
Available : http://www.law.usyd.edu.au/~alant/netpay.html

ภาคผนวก

โครงสร้างข้อมูลของ Payment Instruction ตามมาตรฐาน SET

* หมายเหตุ : รายละเอียด จะอธิบายเพียงส่วนที่เกี่ยวข้องโดยตรงกับ Message Extension ของ MOSET เท่านั้น ข้อมูลเพิ่มเติมสามารถดูได้จาก http://www.setco.org

PI Data

PI	< PIUnsigned, PIDualSigned,AuthToken > - ผู้ถือบัตรสร้าง PIUnsigned หรือ PIDualSigned - Payment Gateway สร้าง AuthToken สำหรับการส่งสินค้าเป็นงวดๆ ซึ่งมีการชำระเงินเป็นงวดๆด้วย และผู้ขายจะทำการเก็บ PI ลักษณะนี้ไว้สำหรับการตรวจสอบในการชำระเงินครั้งต่อๆไป
PIUnsigned	EXH(P, PI-OILink, PANToken)
PISualSigned	{PISignature, EX(P, PI-OILink, PANData)}
AuthToken	ดูตาราง AuthToken
PI-OILink	L{PIHead, OIData} ดูตาราง PIHead
PISignature	SO(C, PI-TBS)
PI-TBS	{HPIData, HOIData}
HPIData	DD(PIData)
HOIDataa	DD(OIData)
PIData	{PIHead, PANData} ดูตาราง PIHead

PIHead

PIHead	{TransIDs, Inputs, MerchantsID, [InstallRecurData], TransStain, SWIdent, [AccqBackKeyData], [PIExtension]}
TransIDs	บอกข้อมูลที่ใช้ระบุ transaction แต่ละรายการ
Inputs	(HOD, PurchAmt}
MerchantID	เอามาจากใบรับรองอิเลกทรอนิกส์ของผู้ขาย
InstallRecurData	ข้อมูลส่วนที่ทำให้ผู้ถือบัตรสามารถอนุญาตการชำระเงิน หรือสามารถทำการชำระเงินเป็นงวดๆได้
TransStain	HMAC(XID, CardSecret)
SWIdent	ระบุรายละเอียดของโปรแกรม (ผู้ผลิตและเวอร์ชั่น)ที่ใช้ในการเริ่มต้นคำขอสั่งซื้อ เพื่อให้ Payment Gateway รู้ว่าผู้ถือบัตรใช้โปรแกรมใด
AcqBackKeyData	{AcqBackAlg, AcqBackKey}
PIExtensions	ข้อมูลเพิ่มเติมที่เกี่ยวข้องกับรายละเอียดการชำระเงิน มีผลกับการตรวจสอบสิทธิ์ในการชำระเงินของเจ้าของบัตร โดย Payment Gateway, สถาบันการเงิน หรือธนาคารผู้ออกบัตร

ประวัติผู้เขียน

ชื่อผู้แต่ง	**นางสาว อิสรียา วัฒนไพโรจน์รัตน์**
วันเดือนปีเกิด	**18 มกราคม 2521**
สถานที่เกิด	**กรุงเทพมหานคร**
วุฒิการศึกษาระดับปริญญาตรี	**บริหารธุรกิจบัณฑิต**
สถานที่สำเร็จการศึกษา	**จุฬาลงกรณ์มหาวิทยาลัย**