การเพิ่มประสิทธิภาพอุปกรณ์ Electronic Data Capture ด้วย SSL Protocol สำหรับระบบชำระเงิน

Electronic Data Capture Improvement by SSL Protocol for Payment Systems.

ยุทธนา สรวลสรรค์[1]

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลพระนคร

yutthna@rmutp.ac.th[1]

ศักดิ์ชัย ทิพย์จักษุรัตน์[2] *สมศักดิ์ มิตะถา*[3]

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

{ktsakcha[2]*,kmsomsak*[3]*}@kmitl.ac.th*

บทคัดย่อ

ในบทความนี้ ได้ทำการวิเคราะห์ประสิทธิภาพภายหลังจากที่ได้เพิ่มระบบความปลอดภัย (SSL Protocol) เข้ากับชุดข้อมูลใน TCP/ IP บนอุปกรณ์ EDC (Electronic Data Capture) ต้นทาง สิ่งที่ดำเนินการนั้นได้ทำการส่งข้อมูลที่มีลักษณะเป็น Plaintext ที่ไม่มีการป้องกันข้อมูลเปรียบเทียบกับ Ciphertext บน SSL ที่แสดงให้เห็นว่าสามารถป้องกันการลักลอบดักข้อมูลได้ เพื่อความเข้าใจถึงผลกระทบที่มีต่อประสิทธิผลของอุปกรณ์ EDC ความปลอดภัยที่ได้ดำเนินการนั้น ได้ทำการตรวจวัดประสิทธิภาพอัตราการส่งผ่านข้อมูลที่มีความปลอดภัยในหลายรูปแบบของการเข้ารหัสข้อมูล

Abstract

In this paper, we have analyzed the performance of combining security system (SSL Protocol) with current TCP/IP module by porting security shareware into the EDC (Electronic Data Capture) devices. Finally, in order to understand the impact on the EDC's performance, we test the throughput of the EDC before and after applying security with various encryption algorithms.

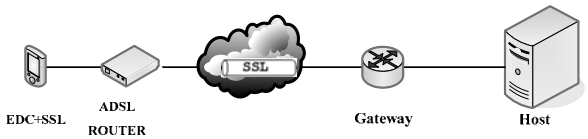
Key words: SSL, EDC, Plaintext, Ciphertext

1. บทนำ

การดำรงชีพในปัจจุบันนั้นกระแสไฟฟ้า น้ำประปา รวมไปถึง โทรศัพท์พื้นฐานเป็นตัวอย่างของสิ่งจำเป็นในการดำเนินชีวิตของมนุษย์ในปัจจุบัน ทั้งนี้ภายหลังจากที่เราได้ใช้บริการของระบบสาธารณูปโภคแล้ว เราก็จำเป็นต้องฝ่าวิกฤตจราจรเพื่อดำเนินการชำระค่าบริการดังกล่าว แต่ด้วยภารกิจอันจำเป็นของแต่ละบุคคล ทำให้เหลือเวลาดำเนินการด้านนี้น้อยมาก จึงมีผู้ให้บริการอำนวยความสะดวกด้วยการรับชำระค่าบริการหรือสาธารณูปโภค โดยการชำระเงินผ่านระบบบัญชีธนาคาร, ชำระผ่านบัตรเครดิต, ชำระที่เคาน์เตอร์บริการต่างระบบหรือแม้แต่ระบบอิเล็กทรอนิกส์ [1] เป็นต้น แต่ในบางครั้งก็ได้รับความสะดวกยังไม่เพียงพอ กับการที่ต้องรอคิวเป็นเวลานาน ๆ หรือแม้กระทั่งชำระผ่านตู้ ATM ซึ่งบางครั้ง

อาจจะไม่มีตู้ ATM ที่เราถือบัตรอยู่ในบริเวณนั้น ก็เกิดเป็นปัญหาในเรื่องความไม่สะดวกได้เช่นกัน จากปัญหาที่ได้กล่าวมานั้น จึงมีแนวความคิดหาวิธีการใหม่ ซึ่งเป็นทางเลือกอีกทางให้สามารถนำไปใช้ได้ เพราะการเดินทางไกลๆ ทำให้เสียเวลาหรือบางครั้งอาจจะไม่มีตู้ ATM ที่เราถือบัตรอยู่ในบริเวณนั้น วิธีการที่จะนำเสนอคือการใช้อุปกรณ์ที่เรียกว่า EDC (Electronic Data Capture) เพื่อเป็นสื่อในการชำระผ่านระบบเครือข่าย

โลกอินเตอร์เน็ตในปัจจุบันจะเห็นว่าความปลอดภัยของเครือข่ายเป็นสิ่งที่มีความสำคัญในระดับต้นๆ แม้ว่ามีหลากหลายแนวทางในการจัดการสภาพแวดล้อมของระบบเครือข่ายอย่างปลอดภัย แต่สถาปัตยกรรมความปลอดภัยของ IP (SSL/IPSec)[2] เป็นกลไกความปลอดภัยของเครือข่ายยอดนิยมและเครือข่ายที่ทันสมัยที่สุด การชำระเงินผ่านระบบเครือข่าย โดยใช้อุปกรณ์ EDC (Electronic Data Capture) เป็นเครื่องมือส่งผ่านข้อมูลโดยการรูดบัตร หรือการสแกนบาร์โค้ด ซึ่งวงจรควบคุมที่อยู่ภายในอุปกรณ์ EDC นั้นเป็นไมโครคอนโทรลเลอร์ และที่นำมาใช้ในงานวิจัยนี้ ใช้ชิปตระกูล ARM7 ซึ่งสามารถทำการโปรแกรม และสามารถแก้ไขโปรแกรมได้ อุปกรณ์ EDC ที่ใช้นั้นสามารถพิมพ์ใบเสร็จรับเงินเก็บไว้เป็นหลักฐานในการชำระค่าสาธารณูปโภคได้ สามารถติดต่อสื่อสารกับระบบเครือข่ายด้วย พอร์ต Modem, RS-232 และ Ethernet ในการวิจัยครั้งนี้จะทดลองกับพอร์ต Ethernet ติดต่อระบบเครือข่าย ดังแสดงระบบการเชื่อมต่อในรูปที่ 1 ซึ่งในการต่อเข้าระบบเครือข่ายนั้นเรื่องสำคัญอีกประการคือ เรื่องความปลอดภัยของข้อมูลในการรับ-ส่งข้อมูลต้องคำนึงถึงความปลอดภัย ดังนั้นจำเป็นต้องมีการเข้ารหัสข้อมูลเพื่อความปลอดภัย โดยงานวิจัยนี้ได้ดำเนินการศึกษาและเพิ่มระบบความปลอดภัยด้วย SSL Protocol ลงไปบนตัวอุปกรณ์ EDC และทำการวิเคราะห์หาอัลกอริธึมที่เหมาะสมกับตัวอุปกรณ์ต่อไป

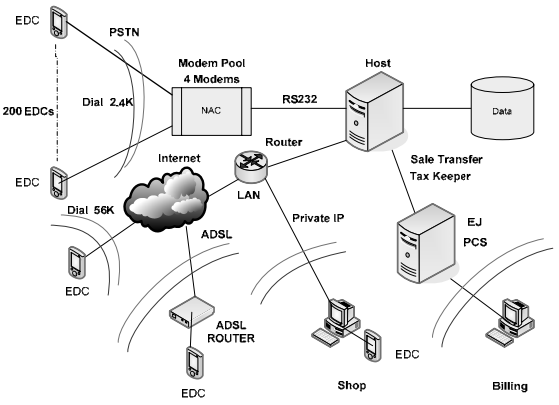


รูปที่ 1: ผังวงจรสื่อสาร

เนื้อหาของบทความในหัวข้อที่ 2 ได้นำเสนอโครงสร้างของระบบชำระเงิน อธิบายหลักการของ SSL Protocol และวิธีการต่างๆ ที่นำมาใช้ หัวข้อที่ 3 ขั้นเตรียมความพร้อม แสดงคุณสมบัติของเครื่องมือ หัวข้อที่ 4 วิธีการทดลอง หัวข้อที่ 5 ผลการทดลอง และหัวข้อที่ 6 เป็นข้อสรุป

2. โครงสร้างระบบชำระเงิน

จากการศึกษาระบบชำระ ดังรูปที่ 2 เป็นระบบที่ได้ทำการศึกษารูปแบบ แบ่งเป็นหลายแนวทางส่วนแนวทางที่จะนำมาทำการวิจัย คือการที่ EDC ติดต่อกับระบบชำระผ่านระบบเครือข่ายอินเตอร์เน็ต ซึ่งมีหลักการทำงานโดยพอร์ตสื่อสารนั้นจะใช้พอร์ต Ethernet ที่สามารถติดต่ออินเตอร์เน็ตผ่านทาง ADSL Modem และในส่วนนี้ก็ต้องคำนึงถึงความปลอดภัยของข้อมูลด้วย ดังนั้นในการวิจัยก็จะทำการศึกษาในเรื่องความปลอดภัยของข้อมูลเป็นหลักสำคัญ



รูปที่ 2: ผังระบบชำระแบบสมบูรณ์

2.1 SSL Protocol

SSL (Secure Socket Layer) [3],[4],[5] เป็นโปรโตคอลที่อยู่ในชั้น Application SSL ส่วนใหญ่แล้วจะป้องกันการแลกเปลี่ยนข้อมูลของ HTTP และเพื่อจุดประสงค์อื่นเช่น IMAP และ POP3 เป็นต้น SSL เป็น application ที่ทำงานอยู่บนชั้น TCP แต่บางอย่างที่มีการปรับเปลี่ยนก็สามารถเป็น application ที่ทำงานอยู่บน SSL เมื่อไม่นานมานี้มีการพัฒนาซอฟต์แวร์ที่เรียกว่า

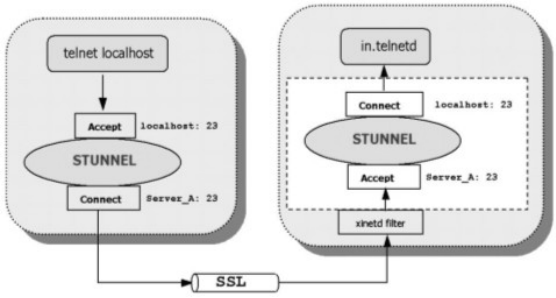
Stunnel [6],[7] สำหรับการใช้งานที่ไม่ค่อยสลับซับซ้อน SSL ประกอบไปด้วยโปรโตคอลดังนี้

1. Handshake protocol
2. Change Cipher Spec protocol
3. Alert protocol
4. Application Data protocol

Handshake protocol ใช้สำหรับการ authentication และ key exchange, Change Cipher Spec protocol ใช้แสดงคีย์ที่ถูกเลือกใช้, Alert protocol ใช้ส่งสัญญาณความผิดพลาด และการปิดงาน และ Application Data protocol ใช้รับ-ส่งข้อมูลที่เข้ารหัส

2.2 หลักการทำงานของ Stunnel

Stunnel ใช้หลักการทำงานของลูกข่าย-แม่ข่าย (Client-Server Model) ดังนั้น Stunnel จะสามารถทำงานได้สองโหมดคือ Client Mode และ Server Mode การทำงานทั้งสองแบบจะอาศัยหลักการของ Port Forwarding เช่นเดียวกับ SSH Tunneling เพื่อเปลี่ยนแปลงการเชื่อมต่อของโปรโตคอล หรือโปรแกรมบริการพื้นฐานให้อยู่ภายใต้การทำงานของการเข้ารหัสแบบ SSL ของโปรแกรม Stunnel ศึกษาตัวอย่างต่อไปนี้เพื่อความเข้าใจกระบวนการทำงานของ Stunnel ที่เพิ่มมากขึ้น ถ้าผู้ดูแลระบบที่มีความจำเป็นต้องเปิดให้บริการ telnet กับเครื่องลูกข่าย แต่ผู้ดูแลระบบรู้ดีว่าการให้บริการ telnet นั้นอาจจะไม่ปลอดภัยจากการถูกดักจับรหัสผ่าน ดังนั้นผู้ดูแลระบบจึงป้องกันการดักจับรหัสโดยกำหนดให้การเชื่อมต่อของ telnet อยู่ภายใต้การทำงานของโปรแกรม Stunnel รูปแบบการทำงานของระบบนี้แสดงได้ดังภาพที่ 3



รูปที่ 3: แสดงการทำงานของโปรแกรม Stunnel [7]

จากรูปที่ 3 เครื่องเซิร์ฟเวอร์เปิดให้บริการ telnet ผ่านพอร์ต 23 และเข้ารหัสข้อมูลด้วยโปรแกรม Stunnel การขอใช้บริการ telnet จากเครื่องลูกข่ายจะต้องผ่านกระบวนการทำงานของโปรแกรม Stunnel ก่อนมิเช่นนั้นกระบวนการขอใช้บริการจะไม่เกิดขึ้น เนื่องจากข้อมูลที่ติดต่อระหว่างเครื่องลูกข่าย และเครื่องเซิร์ฟเวอร์นั้นถูกเข้ารหัสแบบ SSL ทำให้โปรแกรม telnet ธรรมดาไม่เข้าใจข้อมูลที่ถูกเข้ารหัสเหล่านั้น

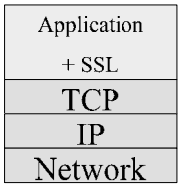
2.3 การเข้ารหัสข้อมูล (Cryptography)

การเข้ารหัสข้อมูลโดยพื้นฐานแล้วจะเกี่ยวข้องกับวิธีการทางคณิตศาสตร์เพื่อใช้ในการป้องกันข้อมูลหรือข้อความตั้งต้นที่ต้องการส่งไปถึงผู้รับ ข้อมูลตั้งต้นจะถูกแปรเปลี่ยนไปสู่ข้อมูลหรือข้อความอีกรูปแบบหนึ่งที่ไม่สามารถอ่านเข้าใจได้โดยใครก็ตามที่ไม่มีกุญแจสำหรับเปิดดูข้อมูลนั้น เราเรียกกระบวนการในการแปรรูปของข้อมูลตั้งต้นว่า "การเข้ารหัสข้อมูล" (Encryption) และกระบวนการในการแปลงข้อความที่ไม่สามารถอ่าน และทำความเข้าใจให้กลับไปสู่ข้อความดั้งเดิม ว่าการถอดรหัสข้อมูล (Decryption) อัลกอริธึมในการเข้ารหัสข้อมูล คือ

อัลกอริธึมแบบอสมมาตร (Asymmetric key algorithms) อัลกอริธึมนี้จะใช้กุญแจสองตัวเพื่อทำงานตัวหนึ่งใช้ในการเข้ารหัสและอีกตัวหนึ่งใช้ในการถอดรหัสข้อมูลที่เข้ารหัสมาโดยกุญแจตัวแรก อัลกอริธึมกลุ่มสำคัญในแบบอสมมาตรนี้คือ อัลกอริธึมแบบกุญแจสาธารณะ (Public keys algorithms) ซึ่งใช้กุญแจที่เรียกกันว่า กุญแจสาธารณะ (Public keys) ในการเข้ารหัสและใช้กุญแจที่เรียกกันว่า กุญแจส่วนตัว (Private keys) ในการถอดรหัสข้อมูล [4], [8], [9]

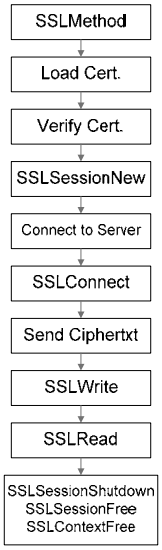
3. ขั้นเตรียมการ

ในรูปที่ 4 นั้นแสดงโครงสร้างของระบบเครือข่ายที่มี SSL Protocol อยู่ในชั้นของ TCP/IP



รูปที่ 4: ระบบเครือข่ายที่มี SSL Protocol

สร้างฟังก์ชันการทำงานของ SSL บนอุปกรณ์ EDC จากการที่ได้ศึกษาโดยแสดงหลักการทำงานตามผังการทำงานดังรูปที่ 5



รูปที่ 5: แสดงผังการทำงานของโปรแกรม

ตัวอย่างชุดคำสั่งในการเลือกเวอร์ชันของ SSL

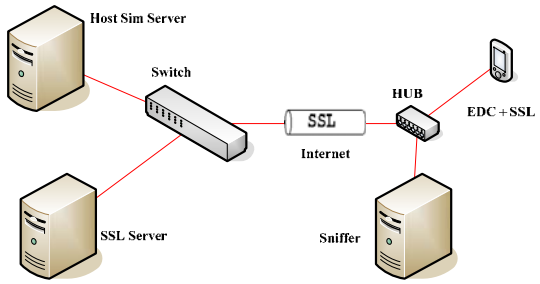
```
SSL_METHOD* method = SSLv3_client_method();
SSL_CTX*    ctx = SSL_CTX_new(method);
```

คุณสมบัติเครื่องมือ

1. PC: SSL Server : Stunnel 4.20
   -Pentium 4, 1.8 GHz, RAM 512MB
   -NIC 100Mbps
2. PC : Host Sever
   -Pentium 4, 2.4 GHz, RAM 512MB
   -NIC 100Mbps
3. PC : Sniffer : Ethereal 0.99.0
   -Pentium 4, 2.4 GHz, RAM 512MB
4. EDC
   -Powerful 32-bit ARM processor
   -8MB Flash RAM and 1MB SRAM
   -128x64 dots graphics LCD and keyboard
   -18 keys with printing protected by long lasting transparent epoxy
   -2" fast and silent thermal printer
   -Magnetic stripe card reader and smart card reader
   -TCP/IP interface and high speed modem (up to 56K bps)

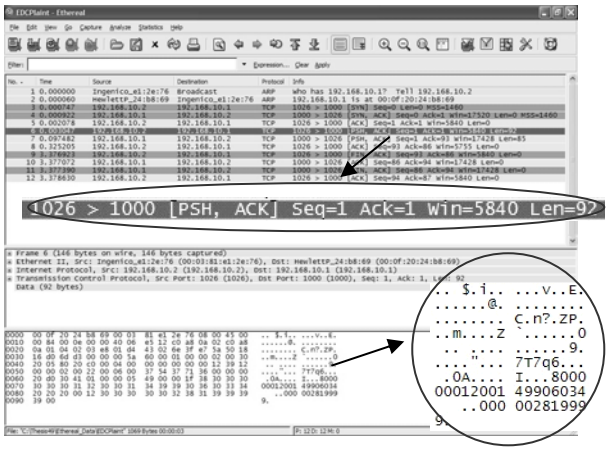
4. การทดลอง

ทำการเชื่อมต่อระบบเครือข่ายในขั้นการทดลองเก็บข้อมูลแสดงดังรูปที่ 6



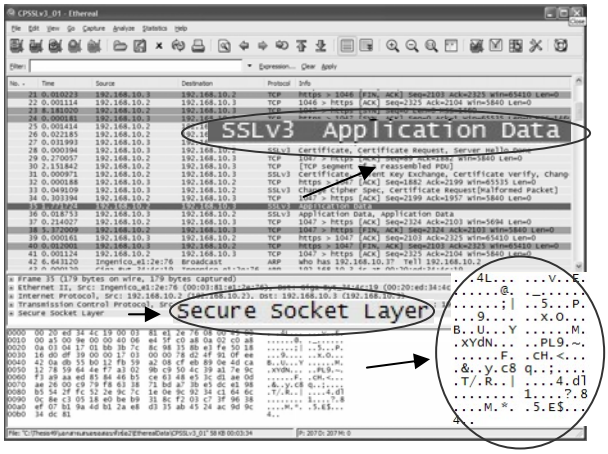
รูปที่ 6: ระบบที่มีการลักลอบดักข้อมูล

ทำการส่งข้อมูลที่เป็น Plaintext ซึ่งเป็นข้อมูลตามมาตรฐาน ISO8583 จากอุปกรณ์ EDC ไปยังเครื่อง Host Sim Server และทำการลักลอบดักข้อมูลโดยเครื่องคอมพิวเตอร์ Sniffer และแสดงคุณลักษณะของข้อมูลด้วยโปรแกรม Ethereal จากรูปที่ 7 สามารถมองเห็นข้อมูลที่ส่งจาก EDC ไปยังเครื่อง Host Sim Server



รูปที่ 7: ข้อมูล Plaintext จากการลักลอบดัก

หลังจากนั้นดำเนินการแบบเดิม แต่ในครั้งนี้ทำการเข้ารหัสข้อมูลบน SSL Protocol ที่มีกระบวนการสร้างการสื่อสารแบบ RC4-MD5 [10] ทำการเข้ารหัสข้อมูลแบบ RC4 และแสดงให้เห็นว่ามีขบวนการสร้างความปลอดภัยของข้อมูลดังแสดงในรูปที่ 8



รูปที่ 8: แสดงข้อมูลที่มีการเข้ารหัสบน SSL Protocol

ทำการส่งข้อมูลเข้ารหัสด้วย SSLv2 และ SSLv3 เป็นจำนวน 10, 20, 30, 40 และ 50 ครั้ง เพื่อหาค่าเฉลี่ยเวลาเปรียบเทียบของทั้ง 2 เวอร์ชัน ทำการหาค่าเฉลี่ยโดยการใช้สมการที่ (1)

$$T = \frac{1}{N}\sum_{i=1}^{N} t_i \qquad (1)$$

โดยที่ T คือค่าเฉลี่ยของเวลาที่ใช้รับ-ส่งข้อมูลทั้งหมด

N คือจำนวนครั้งที่ทดลอง

t_i คือค่าของเวลาที่ใช้รับ-ส่งข้อมูลแต่ละครั้ง

$$\text{Throughput} = \frac{Data}{AvageTime} \qquad (2)$$

สมการที่ (2) เป็นการหาค่าอัตราการส่งผ่านข้อมูลเฉลี่ย

5. ผลการทดลอง

ผลการทดลองรับส่งข้อมูลโดยใช้รูปแบบการเข้ารหัสแบบ RC4-MD5 ผลการทดลองส่ง-รับข้อมูล ที่เป็น Plaintext, SSLv2 และ SSLv3 ขนาด 92 ไบต์ เฉพาะช่วงเวลาส่งและรับข้อมูล จำนวน 10, 20, 30, 40 และ 50 ครั้ง ได้ค่าเฉลี่ยตามตารางที่ 1 ถึง 4 และรูปที่ 9 และ 10

ตารางที่ 1 แสดงค่าเฉลี่ยเวลา (S)ในการส่งข้อมูลด้วยการเข้ารหัสแบบ RC4-MD5

จำนวนครั้ง	Plaintext	SSLv2	SSLv3
10	0.096030	0.095246	0.101498
20	0.100082	0.098872	0.096506
30	0.092624	0.097386	0.094983
40	0.095043	0.092942	0.098876
50	0.094251	0.098226	0.094487

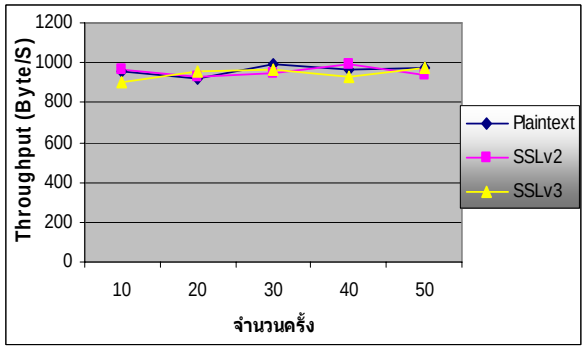
ตารางที่ 2 แสดงค่าเฉลี่ยเวลา (S) ในการรับข้อมูลด้วยการเข้ารหัสแบบ RC4-MD5

จำนวนครั้ง	Plaintext	SSLv2	SSLv3
10	0.065120	0.150940	0.130676
20	0.067751	0.153371	0.162724
30	0.069205	0.139709	0.153815
40	0.072084	0.156546	0.141860
50	0.066563	0.129560	0.148475

ตารางที่ 3 แสดงค่าเฉลี่ยของ throughput ในการส่งข้อมูลด้วยการเข้ารหัสแบบ RC4-MD5

จำนวนครั้ง	Plaintext (Byte/Sec)	SSLv2 (Byte/Sec)	SSLv3 (Byte/Sec)
10	958.03	965.92	906.42
20	919.25	930.50	953.30
30	993.27	944.69	968.59
40	967.98	989.86	930.46
50	976.12	936.62	973.68

จากค่าเวลาเฉลี่ยในตารางที่ 3 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 9

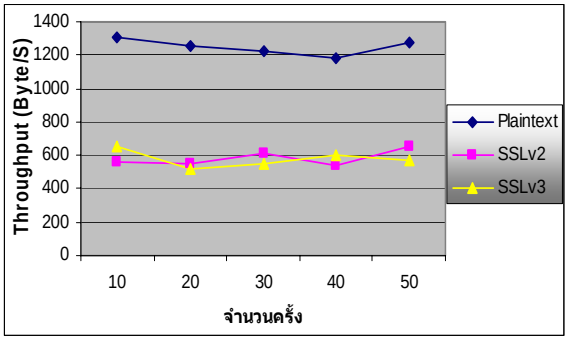


รูปที่ 9 แสดงกราฟค่าเฉลี่ยในการส่งข้อมูลด้วยการเข้ารหัสแบบ RC4-MD5

ตารางที่ 4 แสดงค่าเฉลี่ยของ throughput ในการรับข้อมูลด้วยการเข้ารหัสแบบ RC4-MD5

จำนวนครั้ง	Plaintext (Byte/Sec)	SSLv2 (Byte/Sec)	SSLv3 (Byte/Sec)
10	1305.28	563.14	650.46
20	1254.59	554.21	522.36
30	1228.24	608.41	552.61
40	1179.19	542.97	599.18
50	1276.99	656.07	572.49

จากค่าเวลาเฉลี่ยในตารางที่ 4 นำมาสร้างเป็นรูปกราฟดังแสดงในรูปที่ 10



รูปที่ 10 แสดงกราฟค่าเฉลี่ยในการรับข้อมูลด้วยการเข้ารหัสแบบ RC4-MD5

6. สรุป

การเข้ารหัสข้อมูลบนอุปกรณ์ Electronic Data Capture ด้วย SSL Protocol สำหรับระบบชำระเงิน ได้ดำเนินการเพิ่มโปรโตคอลความปลอดภัย SSL บนอุปกรณ์ EDC ในกรณีการรับข้อมูลแบบ SSLv2 และ SSLv3 อุปกรณ์ EDC ต้องเพิ่มขบวนการ ในการถอดรหัส ดังนั้นค่าเฉลี่ยในการรับข้อมูลจึงลดลงอยู่ในช่วง 570-650 Byte/Sec แสดงว่าผลจากการเพิ่มขบวนการในการเข้ารหัสเข้าไปในอุปกรณ์ EDC นั้นทำให้ประสิทธิภาพของการรับ-ส่งข้อมูลลดลงไปบ้าง แต่มีข้อดีเพิ่มขึ้นคือมีขบวนการในการพิสูจน์ตัวตนและการเข้ารหัสข้อมูลทำให้มีความปลอดภัยเพิ่มขึ้น จากการวัดประสิทธิภาพ และวิเคราะห์การส่งข้อมูล ของ EDC จะเห็นว่า ในการทดลองได้ทำการแลกเปลี่ยนกุญแจเข้ารหัส โดยที่เซิร์ฟเวอร์สามารถร้องขอ Certificate จากไคลเอ็นต์เพื่อตรวจสอบความถูกต้องของไคลเอ็นต์ ในกรณีที่มีการจำกัดเฉพาะไคลเอ็นต์ที่ต้องการเท่านั้น ซึ่ง SSL สนับสนุนการตรวจสอบได้จากทั้งเซิร์ฟเวอร์และไคลเอ็นต์ ขึ้นอยู่กับการเลือกใช้งาน

7. กิตติกรรมประกาศ

งานวิจัยนี้ได้รับการสนับสนุนอุปกรณ์จากบริษัท Ingenico (Thailand) Co.,Ltd.

8. เอกสารอ้างอิง

[1] M.H.Sherif, A.Serhrouchni, A.Y.Gaid, and F.Farazmandnia, “SET and SSL :Electronic payments on the Internet” Computers and Communications, 1998. ISCC '98. Proceedings. Third IEEE Symposium on 30 June-2 July 1998, pp.353 - 358

[2] A. Alshamsi, and T. Saito , “A Technical Comparison of IPSec and SSL”, Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference, vol.2, March 2005, pp. 395-398.

[3] http://www.openssl.org/docs

[4] Stephen A. Thomas, “SSL and TLS Essentials Securing the Web” John Wiley and Sons Inc.

[5] John Viega, Matt Messier and Pravir Chandra, “Network Security with OpenSSL” O’Reilly

[6] http://www.stunnel.org/faq/stunnel.html #description

[7] http://www.thaicert.nectec.or.th/paper/encryption/stunnel.php

[8] http://www.thaicert.nectec.or.th/paper/encryption/intro_crypt.php

[9] Wolfgang Ranki and Wolfgang Effing ,“Smart Card Handbook” 3rd John Wiley and Sons Inc.

[10] D. Berbecaru, “On Measuring SSL-based Data Transfer with Handheld Devices” *ISWCS-2005: IEEE International Symposium on Wireless Communication Systems*, Siena (Italy), September 5-9, 2005, 5 pages