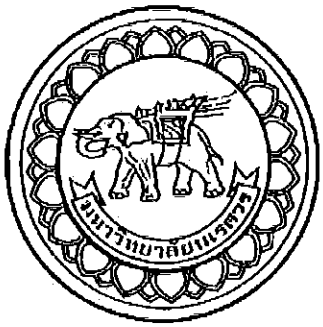


กรณีศึกษาการออกแบบโปรโตคอลเพื่อลดจำนวนอีเมล์ขยะ

A Case Study of Spam Fighting Protocol Design

นายวิพุธ คุณูปการ รหัส 46360095

นายอรรถกร อ้วนวิจิตร รหัส 46362190

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2549

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน กรณีศึกษาการออกแบบโปรโตคอลเพื่อลดปริมาณอีเมล์ขยะ

ผู้ดำเนินโครงงาน นายวิพุธ คุณูปการ รหัส 46360095

นายอรรถกร อ้วนวิจิตร รหัส 46362190

อาจารย์ที่ปรึกษา อาจารย์ภาณุพงศ์ สอนคม

สาขาวิชา วิศวกรรมคอมพิวเตอร์

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2549

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธานกรรมการ

(อาจารย์ภาณุพงศ์ สอนคม)

.....กรรมการ

(ดร.สุรเดช จิตประไพกุลศาล)

.....กรรมการ

(อาจารย์จิราพร พุกสุข)

หัวข้อโครงงาน	กรณีศึกษาการออกแบบโปรโตคอลเพื่อลดจำนวนอีเมล์ขยะ
ผู้ดำเนินโครงงาน	นายวิพุธ คุณูปการ รหัสนิสิต 46360095
	นายอรรถกร อ้วนวิจิตร รหัสนิสิต 46362190
อาจารย์ที่ปรึกษา	อาจารย์ภาณุพงศ์ สอนคม
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2549

.....

บทคัดย่อ

โครงการนี้เป็นการศึกษาเปรียบเทียบข้อดี-ข้อเสียของวิธีการต่างๆในการออกแบบโปรโตคอลเพื่อลดปริมาณอีเมล์ขยะซึ่งพบมากในการสื่อสารระบบอีเมล์ และนำผลดังกล่าวมาเป็นหลักเกณฑ์ในการออกแบบโปรโตคอลรูปแบบใหม่ ซึ่งผู้ส่งอีเมล์สามารถค้นหาคำตอบจากเว็บไซต์และได้แสตมป์ไฟล์แนบกับอีเมล์ส่งไปยังผู้รับ ปริมาณการค้นหาคำตอบจะมากหรือน้อยนั้นขึ้นอยู่กับความน่าเชื่อถือของผู้ส่ง และโปรโตคอลรูปแบบใหม่นี้สามารถนำมาใช้กับระบบอีเมล์ในปัจจุบันได้ เพราะไม่ได้ดัดแปลงโปรโตคอล SMTP ซึ่งเป็นโปรโตคอลที่ใช้เป็นมาตรฐานในการรับส่งอีเมล์

Project Title	A Case Study of Spam Fighting Protocol Design		
Name	Mr. Wiput	Kunoopakarn	ID. 46360095
	Mr. Atthakorn	Uanwichit	ID. 46362190
Project Advisor	Mr. Panupong	Sornkom	
Major	Computer Engineering		
Department	Electrical And Computer Engineering		
Academic Year	2006		

ABSTRACT

The objective of this project was to compare the advantage and disadvantage of the present protocols in order to design new protocols which can reduce e-mail wastes and find the answer to change for the stamp file for sending to the receivers. The number of finding the answer depend on the respect of the senders. The new protocols had never changed the SMTP and can use for the present e-mails system.

กิตติกรรมประกาศ

โครงงานวิศวกรรมคอมพิวเตอร์สำเร็จได้ด้วยดี ก็เนื่องจากความอนุเคราะห์จากท่านอาจารย์ที่ปรึกษา อาจารย์ ภาณุพงศ์ สอนคม ที่กรุณาให้คำปรึกษา แนะนำวิธีการในการทำงานตลอดถึงการตรวจสอบการทำงานพร้อมทั้งเสนอแนวทางการแก้ไขตลอดระยะเวลาการทำโครงงาน สุดท้ายขอขอบพระคุณดร.สุรเดช จิตประไพกุลศาลและอาจารย์จิราพร พุกสุข ที่กรุณาเสียสละเวลาอันมีค่าในการตรวจสอบเนื้อหาของโครงงานฉบับนี้ และให้ความกรุณาเป็นกรรมการในการสอบโครงงานรวมถึงเพื่อนๆ ทุกคนที่ยังไม่ได้เอ่ยนามที่คอยสนับสนุนในการทำโครงงานครั้งนี้

สารบัญ

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

หน้า

ตารางที่

สารบัญรูป

หน้า

รูปที่

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ในปัจจุบันเทคโนโลยีการสื่อสารทางอินเทอร์เน็ตมีความสำคัญต่อชีวิตประจำวันเป็นอย่างมาก ซึ่งหนึ่งในวิธีในการติดต่อสื่อสารที่ได้รับความนิยมและใช้กันอย่างแพร่หลายก็คือ การใช้อีเมล์ติดต่อสื่อสารกัน เนื่องจากการติดต่อสื่อสารด้วยระบบอีเมล์มีความสะดวกในการใช้งาน มีความรวดเร็วและแน่นอนในการรับส่งข้อมูล

ถึงแม้ว่าการสื่อสารระบบอีเมล์จะใช้กันอย่างแพร่หลายและสะดวกต่อการใช้งาน แต่ก็มีปัญหาบางประการที่ลดประสิทธิภาพในการใช้งานระบบอีเมล์ นั่นก็คือ การที่ระบบอีเมล์ไม่สามารถป้องกันอีเมล์ขยะ (Spam Mail) ได้ดีเท่าที่ควร ซึ่งอีเมล์ขยะนี้จะเป็นอีเมล์ที่ถูกส่งมาโดยที่ผู้รับอีเมล์ไม่ต้องการ และเป็นการส่งในปริมาณที่มาก ตัวอย่างของอีเมล์ขยะ เช่น การโฆษณาสินค้าต่างๆ การส่งจดหมายลูกโซ่ เป็นต้น ซึ่งก่อให้เกิดผลเสียต่อผู้ใช้งาน คือ จะทำให้สิ้นเปลืองเนื้อที่ในจัดเก็บข้อมูลในเมล์บ็อกก่อให้เกิดความรำคาญแก่ผู้ใช้งาน และจะทำให้สูญเสียเวลาในการลบอีเมล์ในเมล์บ็อกอีกด้วย

ดังนั้น จึงได้ทำการศึกษาวิธีการที่ใช้ในการป้องกันอีเมล์ขยะ เพื่อทำการออกแบบและนำเสนอโปรโตคอลที่ใช้ในการลดปริมาณอีเมล์ขยะ อีกทั้งยังเป็นแนวทางในการศึกษาเพื่อที่จะนำโปรโตคอลที่นำเสนอนี้มาพัฒนาสำหรับการใช้งานจริงต่อไป

1.2 วัตถุประสงค์

1. ทำการออกแบบโปรโตคอลเพื่อใช้ลดปริมาณอีเมล์ขยะ
2. เพื่อเป็นแนวทางสำหรับการศึกษาและพัฒนาไปสู่การใช้งานจริง

1.3 ขอบข่ายของโครงงาน

1. ศึกษาการทำงานของเทคนิคและวิธีการต่างๆที่ใช้ป้องกันอีเมล์ขยะ
2. ออกแบบการทำงานของโปรโตคอล
3. สร้างโปรแกรมเพื่อหาคำตอบของ Puzzle
4. สร้างเว็บไซต์ที่ให้บริการแสตมป์

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาเกี่ยวกับทฤษฎีและหลักการต่อไปนี้
   - ศึกษาการทำงานของระบบอีเมล์
   - ศึกษาโปรโตคอลที่ใช้ในระบบอีเมล์
   - ศึกษาลักษณะของเมล์ขยะ
   - ศึกษารูปแบบการโจมตีของแสปมเมอร์
   - ศึกษาการทำงานของเทคนิคและวิธีการต่างๆที่ใช้ในการป้องกันอีเมล์ขยะ
2. ทำการออกแบบโปรโตคอล
3. ทดสอบประสิทธิภาพของโปรโตคอลที่ได้ออกแบบขึ้นมา
4. ทำการปรับปรุงแก้ไขโปรโตคอล
5. วิเคราะห์การทดสอบและสรุปผล
6. จัดทำเป็นรูปเล่ม

1.5 แผนการดำเนินงาน

กิจกรรม	ปี 2549							ปี 2550			
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.
1. ศึกษาทฤษฎีและหลักการ	←	—	—	—	→						
2. ทำการออกแบบโปรโตคอล					←	—	—	—	→		
3. ทดสอบประสิทธิภาพ								←	—	→	
4. ปรับปรุงแก้ไขโปรโตคอล									←	—	→
5. วิเคราะห์และสรุปผล										←	— →
6. จัดทำรูปเล่ม										←	— →

1.6 ผลที่คาดว่าจะได้รับ

1. โปรโตคอลที่ออกแบบสามารถที่จะควบคุมและลดปริมาณอีเมล์ขยะได้

2. สามารถนำโปรโตคอลที่ได้ออกแบบขึ้นมาไปเป็นแนวทางการศึกษา และนำไปสู่การนำไปใช้งานจริง

1.7 งบประมาณ

- ค่าหนังสือ	1,000	บาท
- ค่าถ่ายเอกสารและการจำทำรูปเล่ม	500	บาท
- ค่าใช้จ่ายในการค้นหาข้อมูลภายนอกหนังสือ	500	บาท
รวมเป็นเงินทั้งสิ้น	2,000	บาท (สองพันบาทถ้วน)

หมายเหตุ ถัวเฉลี่ยทุกรายการ

บทที่ 2
หลักการและทฤษฎี

2.1 ลักษณะการทำงานของระบบอีเมล์

ลักษณะการทำงานของระบบอีเมล์(E-mail) โดยจะเริ่มต้นจากผู้ส่ง (Sender) จะเริ่มเขียนจดหมาย เมื่อเขียนเสร็จแล้วก็จะทำการส่งอีเมล์โดยจะถูกส่งผ่านโปรโตคอลเพื่อส่งไปยังเครื่อง Mail Server จากนั้นเครื่อง Mail Server จะส่งไปยังเครื่องที่เป็น Relay Host เนื่องจากเครื่อง Relay Host เป็นเครื่องที่สามารถติดต่อกับโลกภายนอกได้ โดยทั่วไปเครื่อง Mail Server อาจทำหน้าที่เป็นเครื่อง Relay Host ในเครื่องเดียวกันก็ได้

จากเครื่องRelay Host ต้นทางเมื่อได้รับอีเมล์มาแล้วจะติดต่อกับเครื่องRelay Host ปลายทางเพื่อส่งอีเมล์ฉบับนี้ไป และในกรณีเดียวกัน ถ้าเครื่อง Relay Host ปลายทางกับเครื่อง Mail Server ปลายทางเป็นเครื่องเดียวกัน เมื่ออีเมล์ไปถึงเครื่อง Mail Server ปลายทางเรียบร้อยแล้ว ถือเป็นการเสร็จสิ้นกระบวนการส่งอีเมล์

เมื่อผู้รับทำการเช็คอีเมล์ ก็จะต้องติดต่อกับเครื่อง Mail Server ของตนเอง เพื่อนำอีเมล์ฉบับนั้นมาอ่าน ในที่นี้เครื่อง Mail Server หรือเครื่อง Relay Host ก็จะทำหน้าที่เหมือนกับที่ทำการไปรษณีย์ในการส่งจดหมายทั่วไปนั่นเอง

2.2 สถาปัตยกรรมของระบบอีเมล์

โปรโตคอล TCP/IP มีโปรโตคอลสนับสนุนการรับ-ส่งอีเมล์หลายโปรโตคอล แต่โปรโตคอลที่ได้รับความนิยมใช้ในอินเตอร์เน็ตคือ โปรโตคอล SMTP (Simple Mail Transport Protocol) หน้าที่ของโปรโตคอล SMTP คือ กำหนดกรรมวิธีและแบบแผนการนำส่งข้อความระหว่างผู้รับและผู้ส่ง โดยโปรโตคอล SMTP อาศัย TCP เพื่อลำเลียงจดหมายผ่านพอร์ต 25

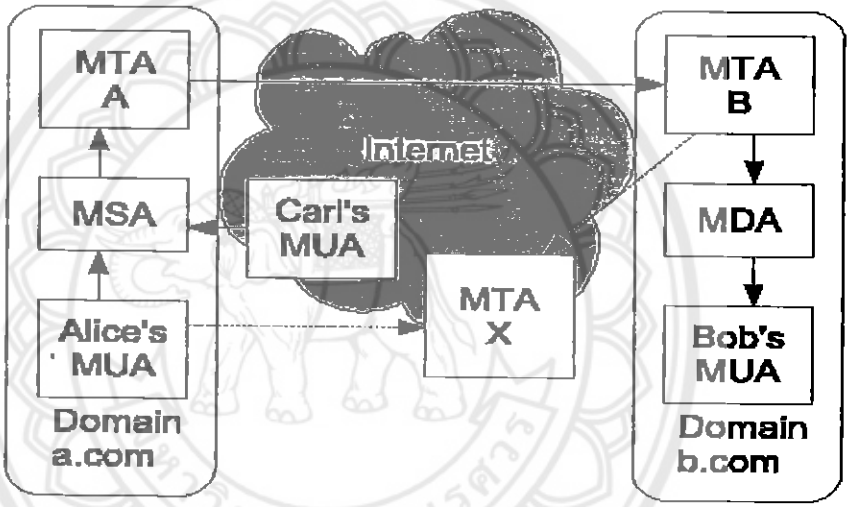
ระบบอีเมล์ที่ใช้ใน TCP/IP มีองค์ประกอบสองส่วนคือ User Agent (UA) (อาจจะเรียกว่า Mail User Agent: MUA) และ MTA (Mail Transfer Agent) ทั้ง UA และ MTA เป็นชื่อที่นำมาจากระบบ X.400 ซึ่งเป็นมาตรฐานนานาชาติกำหนดการนำส่งอีเมล์

- **User Agent (UA)** ทำหน้าที่ในการติดต่อกับผู้ใช้เพื่อรับและส่งอีเมล์โดย UA เป็นโปรแกรมอำนวยความสะดวกให้ผู้ใช้งานสามารถเขียน แก้ไข และส่งจดหมาย รวมทั้งการเปิดอ่านจดหมายที่ได้รับ และจัดเก็บจดหมายเพื่อนำมาใช้ภายหลัง

- **Mail Transport Agent (MTA)** คือส่วนที่ทำหน้าที่ในการรับและส่งอีเมล์ โดยจะรับจาก UA แล้วตรวจสอบว่าผู้รับปลายทางอยู่ในเครื่องเดียวกันหรือไม่ หากอยู่ในเครื่องเดียวกันก็จะส่งอีเมล์นั้นไว้ในเมล์บ็อก หรือโฟลเดอร์ที่เก็บอีเมล์ของผู้รับนั้น แต่หากอยู่กันคนละเครื่อง ก็จะส่ง

ให้กับอีกโปรเซสหนึ่งเพื่อทำการส่งต่อไปยังเครื่องนั้นๆได้ต่อไป (โปรเซสที่ทำหน้าที่รับส่งอีเมล์ข้ามเครื่องนั้นอาจเป็น SMTPD ที่ทำหน้าที่คอยแปลงอีเมล์ให้อยู่ในรูปของโปรโตคอล SMTP เพื่อให้สามารถส่งผ่านเครือข่าย TCP/IP ได้) ในขณะเดียวกันก็ทำหน้าที่รับอีเมล์ที่ส่งเข้ามายังผู้รับในเครื่องนั้น แล้วทำการจัดส่งให้ผู้รับแต่ละคนอย่างถูกต้องด้วย ในส่วนนี้โปรแกรมที่ได้รับความนิยม ได้แก่ Send Mail, Microsoft Mail, Microsoft Exchange

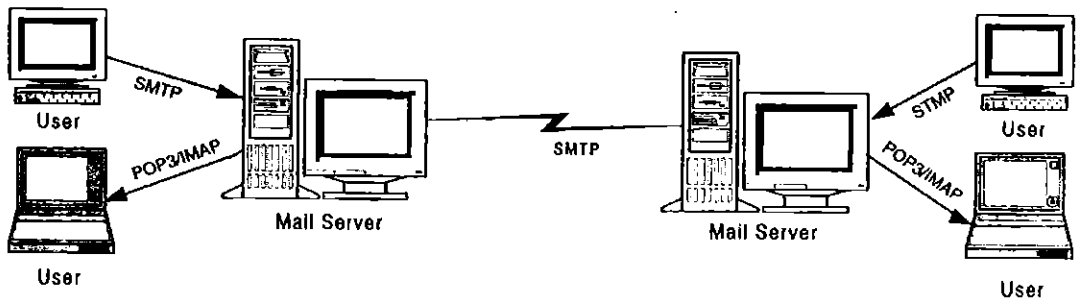
การจัดแบ่งออกเป็น UA และ MTA มีข้อดีคือ แยกงานของทั้งสองส่วนให้เป็นอิสระจากกัน หน้าที่ของ UA จะเน้นการทำงานกับผู้ใช้เพื่อให้ผู้ใช้ อ่าน เขียนจดหมายได้อย่างสะดวกโดยไม่ต้องยุ่งเกี่ยวกับการทำงานระดับล่างของโปรโตคอล ส่วน MTA ทำงานตามโปรโตคอล SMTP เช่น การตรวจสอบความถูกต้องของแอดเดรสผู้รับผู้ส่ง รวมทั้งการหาเส้นทางและนำส่งจดหมายไปยังปลายทาง



รูปที่ 2.1 สถาปัตยกรรมในการรับ-ส่งอีเมล์ [1]

2.3 โปรโตคอลที่ใช้ในการรับ-ส่งอีเมล์

การที่เครื่องคอมพิวเตอร์ 2 เครื่องจะรับส่งอีเมล์กันได้ หรือผู้ใช้จะโหลดอีเมล์ไปอ่านที่เครื่องของตนเองนั้นจำเป็นต้องมีโปรโตคอลที่ใช้คุยกันระหว่างเครื่องทั้งสองดังนี้



รูปที่ 2.2 การรับ-ส่งอีเมล์ผ่านโปรโตคอลที่สำคัญ [1]

1) โปรโตคอล SMTP

SMTP หรือ Simple Mail Transfer Protocol เป็นโปรโตคอลที่ติดต่อกันระหว่างเครื่องที่เป็น Host กับ Host โดย Host ในที่นี้ทำหน้าที่เป็น Mail Server หรือผู้ให้บริการอีเมล์ซึ่งจะมีโปรเซสที่ทำหน้าที่เป็น MTA ทำงานอยู่บนทั้ง 2 ด้าน และรับ-ส่งข้อมูลระหว่างกันโดยใช้โปรโตคอล SMTP เมื่อได้รับอีเมล์มาแล้วก็จะเก็บอีเมล์เหล่านั้นไว้ใน Directory ที่เป็นเมล์บ็อก และรอจนกว่าผู้ใช้มาเปิดอ่าน ซึ่งมีได้ 3 วิธีด้วยกันคือ

- ผู้ใช้มี Account บนเครื่อง Mail Server ก็สามารถเปิดอ่านได้โดยใช้คำสั่งต่าง ๆ ของ Linux/Unix เช่น Mail, Pine และอีเมล์ที่ถูกอ่านจะถูกย้ายไปเก็บไว้ในเมล์บ็อกของผู้ใช้แทนเมล์บ็อกของระบบได้

- ผู้ใช้อยู่บนเครื่องลูกข่าย จะต้องโหลดอีเมล์ไปไว้ในเครื่องของตัวเองก่อน แล้วจึงเปิดอ่านได้

- ผู้ใช้รับส่งอีเมล์ผ่านตัวกลางที่เป็น Web Server ซึ่งอีเมล์นี้จะยังคงถูกเก็บไว้ที่เครื่อง Mail Server

การทำงานของโปรโตคอล SMTP จะทำหน้าที่ในการกำหนดว่า MTA แต่ละตัวจะติดต่อกันได้อย่างไรผ่านทาง TCP/IP จดหมายที่ส่งไปนั้นอาจจะส่งตรงไปยัง MTA ปลายทางเลย หรือว่าผ่าน MTA หลายเครื่อง (ผ่าน Server หลายเครื่อง) ก็ได้

โปรโตคอล SMTP จะไม่สนใจว่าข้อความในจดหมายเป็นอะไรแต่จะจำกัดว่าโปรโตคอล SMTP สามารถส่งได้แต่ข้อมูลที่เป็นข้อความ ASCII เท่านั้น ไม่สามารถส่งไฟล์ที่เป็นเพลง, หนัง, รูปภาพ หรืออื่นๆ ได้ ซึ่งถ้าเราต้องการส่งไฟล์เหล่านั้นผ่านทางโปรโตคอล SMTP จะต้องแปลงไฟล์เหล่านั้นให้อยู่ในรูปของข้อความเสียก่อนและเมื่อส่งไปถึงปลายทางแล้วค่อยทำการแปลงกลับอีกที

นอกจากการใช้โปรโตคอล SMTP เพื่อรับ-ส่งอีเมล์ระหว่าง Mail Server ด้วยกันแล้ว ยังใช้ในขณะที่เป็น Client ส่งอีเมล์ไปยังเครื่องที่เป็น Mail Server ด้วย

2) โปรโตคอล POP

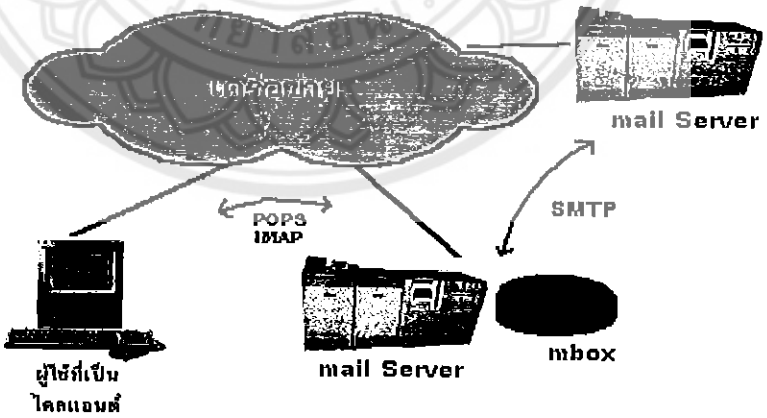
POP (Post Office Protocol) เป็นโปรโตคอลที่ออกแบบมาให้ใช้สำหรับการรับอีเมล์จากเครื่องที่เป็น Mail Server มายังเครื่องของผู้ใช้ โดยทางฝั่ง Server จะมีโปรเซสที่เป็น POP Server ขณะทางฝั่งผู้ใช้มี POP Client ซึ่งในบางโปรแกรมที่ผู้ใช้อ่านและเขียนอีเมล์นั้นจะมี POP client ฝังอยู่ในตัวอยู่แล้ว ไม่ได้แยกออกมาเป็นโปรแกรมหนึ่ง เมื่อผู้ใช้เชื่อมต่อไปที่ POP Server อีเมล์ที่อยู่บน Mail Server จะถูกส่งมาเก็บไว้ในเครื่องของผู้ใช้เลย ดังนั้นเมื่อผู้ใช้จัดการกับอีเมล์ เช่น ลบอีเมล์หรือส่งต่ออีเมล์ก็จะทำกับอีเมล์ที่อยู่บนเครื่องของผู้ใช้เอง ส่วนอีเมล์บน Mail Server จะถูกลบทิ้งไปเมื่อมีการส่งให้ผู้ใช้เรียบร้อยแล้วนอกจากได้กำหนดเพิ่มเติมไว้ที่โปรแกรม Mail Client ว่าอย่าให้ลบอีเมล์ออกจาก Server

ในปัจจุบันโปรโตคอลมีออกมาหลายแบบ แต่ที่นิยมใช้งานกัน คือ โปรโตคอลPOP เวอร์ชัน3 (POP 3) ซึ่งก็ยังมีข้อจำกัดในการใช้งาน คือขณะรับและส่งอีเมล์ฝั่งผู้ใช้จะส่งรหัสผ่านของผู้ใช้ในรูปของข้อความหรือเท็กซ์ไป ทำให้ไม่ปลอดภัยนักหากมีการลอบดักจับข้อมูล

3) โปรโตคอล IMAP

IMAP (Internet Message Access Protocol)เป็นโปรโตคอลที่เกิดหลังโปรโตคอลPOP ถูกนำมาใช้งานเพื่อแก้ไขข้อจำกัดที่เกิดจากโปรโตคอล POP นั่นเอง ทั้งนี้เพราะโปรโตคอล POP จะใช้วิธีการโหลดอีเมล์ที่อยู่บน Server มาเก็บไว้ยังเครื่องพีซีของผู้ใช้ แล้วลบอีเมล์นั้นทิ้ง (แต่ปัจจุบันโปรโตคอล POP มีการพัฒนาขึ้น คือสามารถกำหนดที่ Mail Client ได้ว่าจะให้ลบอีเมล์นั้นทิ้งหรือไม่) ทำให้ผู้ใช้นั้นไม่สามารถอ่านอีเมล์จากพีซีเครื่องอื่นๆได้อีก ต้องใช้เครื่องเดิมตลอด ซึ่งเป็นปัญหาสำหรับผู้ใช้ที่มีเครื่องพีซีที่บ้านและที่ทำงาน หรือองค์กรที่มีเครื่องให้กับพนักงานไม่ครบทุกคน

การทำงานของโปรโตคอล IMAP นั้นจะจัดการอีเมล์ที่อยู่บน Server เช่น อ่านข้อความหรือเขียนข้อความ ซึ่งอีเมล์เหล่านั้นจะยังคงอยู่บน server ทำให้ผู้ใช้จะใช้พีซีเครื่องใดอ่านอีเมล์ก็ได้หรือสั่งดาวน์โหลดอีเมล์ที่ต้องการมาเก็บในเครื่องพีซีของตนเองเหมือนกับการทำงานของโปรโตคอล POP นอกจากนี้ยังสามารถกำหนดเมล์บ็อกหนึ่งๆให้กับผู้ใช้หลายๆคนได้ โดยที่ผู้ใช้เหล่านั้นสามารถเปิด เมล์บ็อก อ่านได้พร้อมๆกัน สำหรับในกรณี Web Mail เครื่องที่เป็น Web Server ก็จะมีการติดต่อกับ Mail Server โดยผ่านโปรโตคอล IMAP เช่นกัน



รูปที่ 2.3 การรับส่งอีเมล์ผ่านเซิร์ฟเวอร์ [2]

ข้อเสียของโปรโตคอล IMAP

ข้อเสียของโปรโตคอล IMAP คือมีความซับซ้อน ยากในการ Implement และมีซอฟแวร์ที่สนับสนุนน้อยกว่าโปรโตคอล POP

2.4 อีเมล์ขยะและอีเมล์ลูกโซ่

คือลักษณะของอีเมล์ ที่สร้างความเสียหายแก่เครือข่ายอินเทอร์เน็ต และสร้างความรำคาญแก่ผู้ใช้โดยแบ่งออกเป็นประเภทใหญ่ๆได้ 2 ประเภท คือ จดหมายลูกโซ่ (Chain Mail) และการโจมตีด้วยอีเมล์ (Spam Mail)

1) อีเมล์ลูกโซ่ มีลักษณะเป็นข้อความเหมือนจดหมายลูกโซ่ ซึ่งเนื้อหาก็จะเป็นเรื่องคำเตือนต่างๆเกี่ยวกับไวรัส หรือเรื่องอื่นๆ แต่ที่สำคัญคือ บอกว่าให้ส่งข้อความนี้ให้กับคนที่รู้จักอีเมล์ลูกโซ่ จะสร้างความเสียหายกับตัว Mail Server ของผู้ให้บริการอินเทอร์เน็ตทำให้ Server Down หรือทำงานช้าลง ถ้ามีคนเชื่อและส่งต่อกันไปมา นอกจากนั้นก็ยังเป็นตัวการให้ Traffic ของเครือข่ายอินเทอร์เน็ตเกิดการติดขัดในส่วนของผู้ใช้งานอินเทอร์เน็ตเองก็จะเกิดความรำคาญถ้ามีอีเมล์ประเภทนี้เข้ามาบ่อยๆ

2) อีเมล์ขยะ คือการส่งอีเมล์จำนวนมากๆในครั้งหนึ่งหรือการทยอยส่ง แต่ส่งจำนวนมากฉบับ สำหรับวัตถุประสงค์นั้นมีหลายหลาก ตั้งแต่โฆษณาสินค้า การโจมตีระบบ การแก้แค้นส่วนตัว หรือการกลั่นแกล้ง เป็นต้น

ในปัจจุบันการติดต่อสื่อสารทางจดหมายอิเล็กทรอนิกส์หรืออีเมล์เป็นที่นิยมใช้กันอย่างแพร่หลายทั้งในระดับองค์กรเพื่อติดต่อทางธุรกิจ หรือการติดต่อสื่อสารส่วนตัว ซึ่งอีเมล์ แอดเดรสของเราก็เปรียบเสมือนที่อยู่ทางไปรษณีย์นั่นเอง แต่จะแตกต่างกันก็คืออีเมล์ แอดเดรสจะถูกส่งไปกับอีเมล์ที่เรารับ, ส่ง หรือส่งต่อ (forward) ก็หมายถึงทุกครั้งที่เรารับ, ส่ง หรือ ส่งต่ออีเมล์จะทำให้คนที่อยู่ในกลุ่มผู้รับอีเมล์ต่อจะรู้อีเมล์ แอดเดรสของเราอย่างง่ายดาย ซึ่งก็คือโอกาสที่ทำให้กลุ่มผู้สร้างอีเมล์ขยะจะสามารถรู้อีเมล์ แอดเดรสของเรา และส่งอีเมล์ขยะมาให้เราได้อย่างไม่ยากเลย

2.5 การโจมตีของ แสปมเมอร์ ที่ทำให้เกิดอีเมล์ขยะ

อีเมล์ขยะ นั้นถูกส่งมา เนื่องจากผู้ใช้นำอีเมล์ แอดเดรสทิ้งไว้ตามที่ต่าง เช่น Chat room ที่มีการทิ้งอีเมล์ แอดเดรสเพื่อติดต่อสื่อสารกัน หรือ การใช้อีเมล์ แอดเดรสในการสมัครสมาชิกของกลุ่มข่าว (newsgroup) หรือ สมัครสมาชิกของเว็บไซต์ต่างๆ ซึ่ง แสปมเมอร์ สามารถที่จะเห็นที่อยู่ของผู้ใช้และนำไปไว้ในลิสต์สำหรับส่งอีเมล์ขยะได้ รวมไปถึงอีเมล์ที่ส่งไปเฉพาะกลุ่มที่อาจถูกส่งต่อไปหาคนอื่นเรื่อยๆ ซึ่งจะเป็นการง่ายต่อการรวบรวมที่อยู่ ซึ่งก็จะกลายเป็นเป้าหมายในการโจมตีของพวก แสปมเมอร์ นอกจากนี้ยังมีโปรแกรมที่สามารถค้นหาและรวบรวมอีเมล์ แอดเดรสต่างๆจาก Search Engine และ เว็บบอร์ด ซึ่งก็จะทำให้ได้ที่อยู่จำนวนมากที่สามารถนำไปใช้ในการส่งอีเมล์ขยะ โดยบางครั้งก็ได้จากพวก Spy Ware ที่แฝงมากับพวกซอฟแวร์, แชร์แวร์, ฟรีแวร์ หรือพวกDemo ต่างๆที่นำมาจากอินเทอร์เน็ต ซึ่งสปายแวร์ จะทำหน้าที่รวบรวมพฤติกรรมการใช้งานอินเทอร์เน็ตของผู้ใช้(รวมถึงอีเมล์ ด้วย) เพื่อส่งไปรวบรวมไปที่ Server แล้วจะส่งโฆษณามาตามที่ผู้ใช้ต้องการ

1) การโจมตีโดยใช้พวก มาลแวร์

ปัจจุบันมีไวรัส (พวกมาลแวร์) จำนวนไม่น้อยที่ใช้คุณสมบัติของอีเมล์ขยะในการกระจายอีเมล์ ออกไปซึ่งเมื่อผู้ใช้เปิดออกดูก็จะทำให้สามารถที่จะติดไวรัสโดยไม่รู้ตัวซึ่งก็อาจจะทำให้เครื่องคอมพิวเตอร์ของผู้ใช้กลายเป็นผู้กระจายอีเมล์ขยะไปที่ต่างๆ รวมทั้งยังสามารถที่จะสร้างช่องทางให้ผู้ไม่ประสงค์ดีเข้ามาบุกรุกและเก็บเกี่ยวผลประโยชน์จากช่องทางที่อีเมล์ขยะ สร้างขึ้น ไวรัสที่ใช้เทคนิคอีเมล์ขยะในการแพร่ขยายนั้น สามารถกระตุ้นความสนใจให้กับผู้ใช้ทำการเปิดอีเมล์ง่ายขึ้นโดยมีวิธีการดึงดูดเช่น เป็นจดหมายแจ้งเตือนจากผู้ผลิตซอฟแวร์รายหนึ่งว่าให้อัพเดทข้อมูลซึ่งทำให้ผู้ใช้คิดว่าเป็นจริงเวลาผู้ใช้เปิดไฟล์ก็จะทำให้เครื่องคอมพิวเตอร์ของผู้ใช้ติดไวรัสโดยไม่รู้ตัวซึ่งก็สามารถทำให้เกิดความเสียหายให้กับเครื่องคอมพิวเตอร์หรือองค์กรนั้นได้

รูปแบบการโจมตีผ่านทางอีเมล์นั้น เป็นวิธีการโจมตีที่มาลแวร์ นิยมใช้มากที่สุด เนื่องจากทุกวันนี้ผู้ใช้คอมพิวเตอร์ทุกคนต้องอ่านอีเมล์เป็นประจำ โอกาสที่ผู้ใช้จะเปิดอีเมล์ที่ไม่หวังดีจึงมีความเป็นไปได้สูง การโจมตีมักจะมาในรูปแบบของการไฟล์แนบ (Attached File) หรือ มาในรูปของ Hyperlink หลอกให้ผู้ใช้คลิกเพื่อไปดาวน์โหลดมาลแวร์ ลงมาในเครื่องคอมพิวเตอร์โดยปกติแล้วไฟล์แนบดังกล่าวจะใช้นามสกุลที่เราไม่ค่อยคุ้น เช่น *.VBS, *.HTA, *.CMD, *.PIF และมักจะมาในรูป executable file เช่น *.EXE หรือ .*.COM หากเราพบไฟล์แนบนามสกุลดังกล่าว ให้สงสัยว่าเป็นมาลแวร์ ไว้ก่อน เพราะคนปกติส่วนใหญ่จะไม่ส่งไฟล์แนบโดยใช้นามสกุลไฟล์ดังกล่าว ในปัจจุบันผู้สร้างมาลแวร์ หันมานิยมใช้ไฟล์แนบนามสกุล *.ZIP ที่เรานิยมใช้กันทั่วไป ทำให้ผู้ใช้งานโดนหลอกง่ายขึ้นโดยการแต่งข้อความในอีเมล์ให้ดูน่าเชื่อถือ วิธีการนี้เรียกว่า Social Engineering เพื่อหลอกผู้อ่าน อีเมล์ให้ตายใจคิดว่าเป็นอีเมล์จากคนรู้จักก็มักจะเปิดโดยไม่ระวังทำให้ถูกมาลแวร์โจมตีได้อย่างง่ายดาย

ปัจจุบันระบบป้องกันมาลแวร์ ที่บริเวณ Internet Perimeter หรือบริเวณ Gateway นั้น สามารถดักจับอีเมล์ที่มีมาลแวร์ แอบแฝงอยู่ได้อย่างมีประสิทธิภาพในระดับหนึ่งถึงแม้จะไม่ร้อยเปอร์เซ็นต์แต่ก็เป็นสิ่งที่องค์กรต้องให้ความสำคัญกับอุปกรณ์ Anti-Spam หรือAnti-Virus ที่บริเวณดังกล่าวในลำดับต้นๆ หากองค์กรมีระบบป้องกันในบริเวณ Gateway ไม่ดีพอจะทำให้เกิดความเสี่ยงต่อถูกโจมตีโดยมาลแวร์ ค่อนข้างสูงด้วยความสามารถของอุปกรณ์ดังกล่าวทำให้ผู้ไม่หวังดีต้องหาทางอื่นในการส่งมาลแวร์ มายังระบบเครือข่ายของเราโดยวิธีการอื่นที่ไม่ใช่การส่งผ่านมาทางอีเมล์เพียงวิธีเดียวเท่านั้น

2) การโจมตีแบบ MIMT (Man In The Middle)

เทคโนโลยีสมัยใหม่ของพวกแฮคเกอร์และโปรแกรม Sniffer (โปรแกรมสำหรับดักจับข้อมูล) นั้น สามารถทำงานในสภาวะแวดล้อมที่ใช้สวิตซ์แทนฮับ ได้ โดยใช้เทคนิค "MIM" หรือ "MIMT" ย่อมาจาก "Man In The Middle Attack" ซึ่งมีการทำการปลอม "MAC Address" หลอกเครื่องต้นทาง เครื่องปลายทาง และ Switch ให้เข้าใจผิดว่าเครื่องของพวก Hacker นั้นเป็นเครื่องต้นทางและปลายทางที่กำลังติดต่อกันอยู่ เทคนิคนี้เรียกว่า "ARP Spoofing" และ "ARP Poisoning" ซึ่งอาศัยช่องโหว่ของโปรโตคอล TCP/IP ที่หลักการ ARP ซึ่งไม่ได้มีการป้องกันด้านความปลอดภัยที่ดีพอ ทำให้ Hacker สามารถปลอมค่า MAC Address ปลอมมาหลอกเครื่องของเราเมื่อไรก็ได้โดยที่เราไม่รู้ตัวและ Switch ที่ไม่ได้ถูกโปรแกรมในลักษณะ "Port Security" ก็จะไม่สามารถที่จะแก้ปัญหา ARP Spoofing และ ARP Poisoning ได้ ด้วยหลักการนี้แฮคเกอร์จึงสามารถมองเห็นข้อมูลสำคัญๆของเราเช่น User Name และ พาสเวิร์ด จากการที่เราเข้าเล่นเว็บไซต์ หรืออ่านอีเมล์ได้อย่างสบายๆ เพราะข้อมูลไม่ได้ทำการเข้ารหัส

แต่วิธีนี้ก็จะมีความลำบากในการที่จะเข้าไปควบคุม Router เพราะในตัว Router จะมีระบบความปลอดภัยภายในตัวเองอยู่แล้ว

2.6 สรุปแหล่งที่ แสปมเมอร์ มักจะหาอีเมล์ แอดเดรส ได้

1) ระบบ Mailing list โดยบางที่ไม่มีการป้องกันการดึงอีเมล์ แอดเดรส ออกจากระบบ โดยเพียงแค่ส่งอีเมล์เข้าไปในระบบ Mailing list ระบบก็จะทำการส่งอีเมล์ แอดเดรสที่มีอยู่ในระบบทั้งหมดมาให้ แสปมเมอร์

2) เว็บไซต์ ต่างๆ ในปัจจุบันนี้มีโปรแกรมที่เข้าไปดูข้อมูลในเว็บไซต์ ซึ่งสามารถแยกแยะข้อมูลและสามารถดึงออกมาเฉพาะอีเมล์ ได้ เช่นดึงออกมาจาก เว็บบอร์ด

3) โปรแกรม Web browser โดยโปรแกรมนี้จะมีการเก็บข้อมูลต่างๆไว้ภายในตัวเองซึ่งพวก แสปมเมอร์ ก็จะสามารถดูข้อมูลจากผู้ใช้ได้

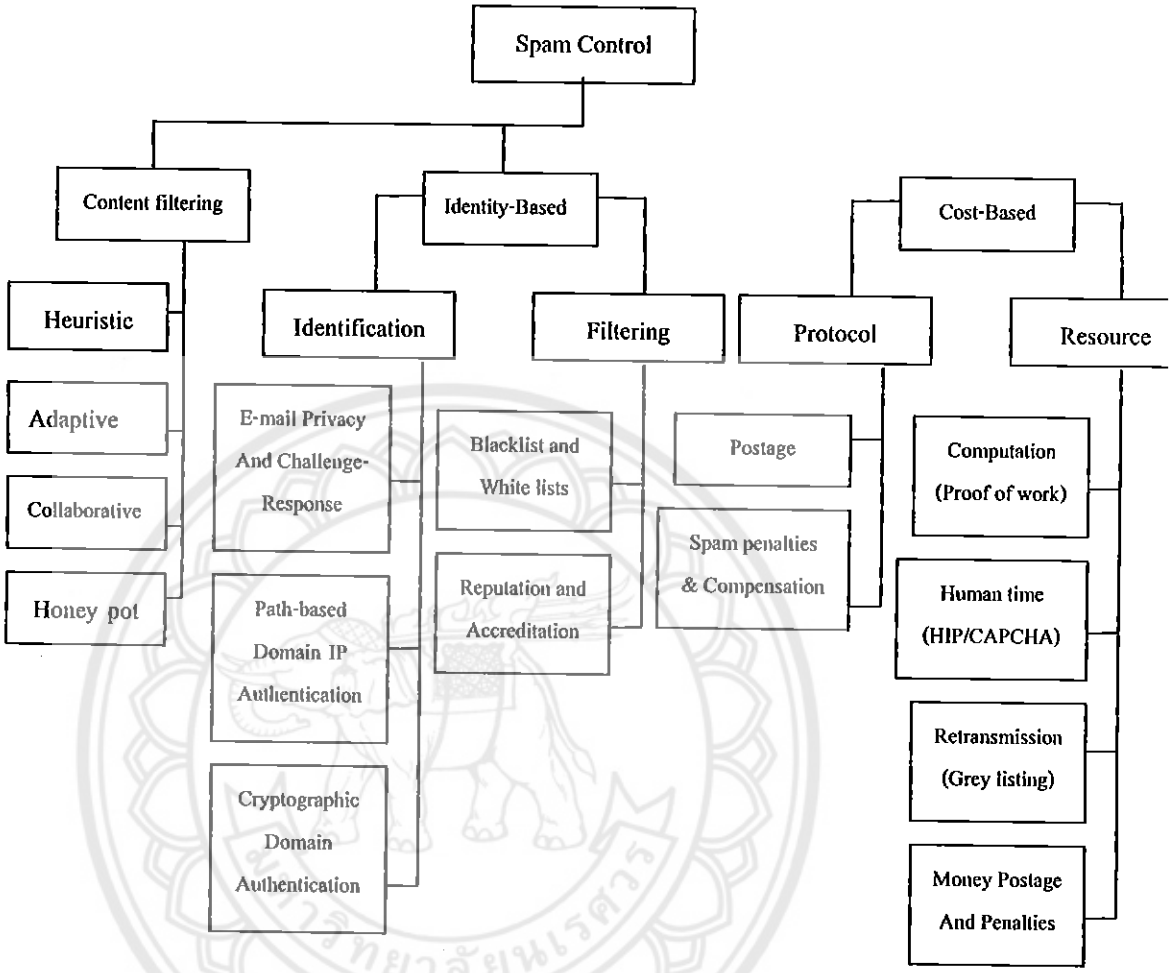
4) Chatroom โปรแกรม Chat บางตัวจะเก็บอีเมล์ แอดเดรสของผู้เข้ามาใช้ซึ่งพวก แสปมเมอร์ ก็จะใช้วิธีนี้ในการเข้ามาดูอีเมล์ แอดเดรสของของผู้ใช้

5) ข้อมูลผู้จดทะเบียนโดเมน ในการจดทะเบียนจะต้องทำการผ่านกระบวนการต่างๆซึ่งสามารถที่จะเรียกดูข้อมูลได้โดยใช้ "Who is" Command โดยเมล์ที่ได้จากวิธีนี้ส่วนใหญ่จะเป็นเมล์ที่ถูกต้อง

6) Internet Directory เว็บไซต์ที่ให้บริการรวบรวมสินค้ารวมถึงรายชื่อและอีเมล์ต่างๆที่ใช้ในการติดต่อในการสั่งซื้อสินค้า

2.7 เทคนิคและวิธีการที่ใช้ในการป้องกัน อีเมล์ขยะ

เราสามารถแยกประเภทของวิธีการควบคุม Spam 3 วิธี ได้ดังนี้



รูปที่ 2.4 เทคนิคและวิธีการในการป้องกันอีเมล์ขยะ [9]

1) Content Filtering Spam Control

1.1) Content Filtering เป็นวิธีการที่พยายามแยกแยะประเภทของอีเมล์ที่มีอย่างหลากหลายแบบ เช่น ข้อความที่มุ่งร้าย, โฆษณาในทางไม่ดี ซึ่งวิธีการนี้จะใช้ไม่ได้ถ้าผู้ส่งมีความเด่นชัดในรูปแบบมาตรฐานการส่ง เช่น ส่งหัวข้อเป็นประเภทโฆษณา โดยอยู่ในกฎของ Anti-Spam อย่างไรก็ตาม พวกข้อความแสปมจำนวนมากมักละเลยที่จะใช้ตามระเบียบแบบแผนที่มีอยู่โดย Content Filtering จะตรวจสอบข้อความอย่างละเอียดและมีกลไกหลากหลายในการตรวจสอบ ซึ่งกลไกที่มีอยู่มีดังนี้

1.2) **Heuristic filter** จะทำการตรวจดูรูปแบบภายในข้อความ เช่น มีคำว่า XXX หรือ FREE... หรือตรวจดู File ที่ส่งมาว่ามีแนวโน้มว่าจะเป็น Virus หรือไม่ แต่รูปแบบ Heuristic filter นี้ยังมีจุดอ่อนอยู่ เนื่องจากพวกแสปมเมอร์ได้รู้ทันกับรูปแบบนี้และสามารถเปลี่ยนคำหรือพลิกแพลงคำให้รอดจากการตรวจได้ เช่น V*I*R*G*A และอีกอย่างหนึ่งก็คือ เป็นการยากมากที่จะเพิ่มรูปแบบในการตรวจเข้าไป

1.3) **Adaptive (Learning , Bayesian)filter** โดยจะสะสมข้อมูล สถิติ ของข้อความซึ่งเป็นอัลกอลิทึ่ม ที่ใช้ในการตรวจสอบว่าข้อความนั้นมีลักษณะเข้าข่ายว่าเป็นSpam หรือไม่เป็นโดยอัตโนมัติ ซึ่งวิธีนี้ถือว่าใช้งานได้ดีทีเดียวเพราะจากรายงานปรากฏว่า 0.03% False Positive(เมล์ที่เป็น Spam แต่คิดว่าไม่เป็น) ส่วน 0.5% เป็น False Negative(เมล์ที่ไม่เป็นSpam แต่คิดว่าเป็น) เท่านั้น แต่อย่างไรก็ตามมันก็กลายเป็นเรื่องง่ายสำหรับพวกแสปมเมอร์ โดยพวกแสปมเมอร์ ก็จะหลีกเลี่ยงพวกข้อความที่ดูรุนแรงหรือหยาบคายหรือพวกคำที่โดนตรวจสอบได้ง่าย รวมถึงพวกแสปมเมอร์ ก็สามารถที่จะใช้ อัลกอลิทึ่มเดียวกับที่ Adaptive ใช้ เพื่อหลีกเลี่ยงการตรวจสอบของ Filter ซึ่งก็จะสามารถดัดแปลงข้อความให้พ้นจากการตรวจสอบในระบบนี้ได้

1.4) **Collaborative Filters** หรือเรียกอย่างง่ายๆก็คือ "Human Filtering" ซึ่งก็จะให้ User เป็นคนคอยดูข้อความและแยกแยะเองว่าข้อความไหนเป็น Spam แล้วก็ทำการมาร์คไว้ว่าข้อความนี้เป็น Spam

1.5) **Honey Pots** เป็นอีเมล์ที่ใช้ในการลวงแสปมเมอร์ ให้ทำการส่งอีเมล์เข้าไป โดยจะมีคำเตือนว่าห้ามคนที่มีตัวตนจริงส่งเข้าไปดังนั้นอีเมล์ที่ส่งเข้าไปก็จะมีแต่เฉพาะที่เป็น Spam เท่านั้น ซึ่งอีเมล์ แอดเดรสที่ส่งอีเมล์ขยะ มานี้ก็จะถูกเก็บไว้ใน Blacklist ซึ่งลักษณะอีเมล์เช่นนี้ก็อาจจะแสดงไว้บนเว็บไซต์ สาธารณะทั่วไปเป็นต้น

พวกแสปมเมอร์ มีการหลีกเลี่ยงกลไกของ Content Filtering โดยการส่งข้อความสั้นๆหรือส่งเป็นพวกรูปภาพซึ่งจะเป็นการยากต่อการตรวจสอบ หรือจะใช้มาลแวร์ เป็นเครื่องมือในการส่งอีเมล์ที่เป็นspam โดยผู้ใช้อาจจะติดพวกมาลแวร์ ซึ่งก็จะทำให้ส่งข้อความหรืออีเมล์ออกไปสู่เป้าหมายของพวกแสปมเมอร์โดยที่ไม่รู้ตัว

ข้อจำกัดที่ยังมีอยู่ของ Mail Filter คือยังไม่สามารถระบุตัวตนได้ดีเท่าที่ควร และพวก แสปมเมอร์สามารถที่จะเลี่ยงข้อความที่ถูก Filter ได้ง่ายเช่นเขียนคำผิดแต่ยังสามารถอ่านให้เข้าใจได้ ซึ่งสิ่งเหล่านี้เป็นปัญหาที่อยู่ใน Content Filtering ในการควบคุมแสปมและวิธีนี้ยังเป็นการล่วงละเมิดสิทธิด้วยเพราะจะต้องเข้าไปอ่านข้อความในตัวอีเมล์ซึ่งบางข้อความนั้นอาจเป็นความลับสำคัญ

2) Identity-Based Spam Control

เป็นวิธีการควบคุม spam ที่อยู่บนพื้นฐานของการระบุตัวตน โดยปกติแล้วการส่งอีเมล์จะเริ่มต้นส่งจากผู้ส่งผ่านทาง MTA ของDomain ผู้ส่ง ซึ่งจะใช้กลไกของ Identification and Authenication ในขณะที่ผู้รับก็จะจะใช้ Identity-Based Filtering โดยจะเป็นกลไกที่ทำให้มีความน่าเชื่อถือจากอีเมล์ของผู้ส่ง

ปัญหาในการส่งอีเมล์

- อีเมล์ แอดเดรสของผู้ส่งจะอยู่ใน Header Field ซึ่งจะส่งไปที่ไหนก็ได้ โดยจะเป็นสิ่งที่ควรจะทำให้ชัดเจนใน HELO/EHLO SMTP
- ในการส่งอีเมล์จะมีการเพิ่มหรือเปลี่ยน Header Field ตามลักษณะของข้อความซึ่งก็มี Form, Sender, Mail และ Resend Form
- MTAs ก็จะอนุญาตให้เพิ่มหรือดัดแปลง Header Field รวมถึงข้อความด้วย
- Mail server ก็จะรับอีเมล์จากที่ที่ไม่รู้Domainได้

มีการโต้ตอบกันเกี่ยวกับความแตกต่างกันของกลไก Identification หรือ Authentication ในความน่าเชื่อถือของแต่ละข้อความในอีเมล์ กลไกนี้จะช่วยให้หลีกเลี่ยง Spamของผู้ส่งในด้านการโจมตีแบบ "spoofed"ได้

2.1) Identification based on Email Path Security

ได้มีการนำเสนอระบบความปลอดภัยเพื่อให้เป็นการยากที่ Attacker จะขโมยข้อความที่จะส่งไปยังที่อยู่ของผู้ที่ถูกโจมตี

- **Identification by sender email address**

กลไกที่จะระบุที่อยู่ของผู้ส่งซึ่งถูกระบุใน Header Field โดยผู้โจมตีสามารถที่จะใช้ที่อยู่ปลอมได้ กลไกนี้จะใช้ได้เมื่อผู้โจมตีไม่รู้ที่อยู่ของผู้ส่งที่แน่นอน ตัวอย่างคือ ระบบจะอนุญาตให้เฉพาะอีเมล์ของผู้ส่งที่อยู่ในMailing list ของผู้ใช้ แต่พวกแสปมเมอร์ จะไม่ใช้ที่อยู่ที่เจาะจง และนี่เป็นข้อสันนิษฐานที่จะใช้รักษาความปลอดภัย ซึ่งผู้โจมตีจะไม่สามารถที่จะเข้าไปดูข้อมูลภายในที่ส่งจากผู้ใช้ได้ ซึ่งก็จะช่วยป้องกันไม่ให้ แสปมเมอร์ สามารถรู้ที่อยู่ของผู้ส่งได้

ระดับความปลอดภัยที่มียังไม่สูงพอคือ ไม่สามารถที่จะป้องกันกลไกของพวกZombieได้ แต่สามารถทำให้ลดน้อยลงได้เพราะAttackerจะใช้การส่งข้อความหลอกจากที่อยู่ที่พบจากข้อความในอีเมล์อื่นมากกว่าการใช้แบบZombie

เนื่องจากยังมีปัญหาต่อการควบคุมซึ่งก็ยังไม่สามารถระบุตัวตนได้แน่นอนจึงได้มีการเสนอกลไกเพิ่มขึ้นมาซึ่งก็คือ

- **Challenge-response validation**

ผู้รับจะส่งข้อความกลับผู้ส่งถ้าเกิดมีข้อสงสัยว่าเป็น Spam โดยจะมีคำถามสั้นๆให้ผู้ส่งได้ตอบคำถามโดยอาจจะมีการบอกเล็กน้อยในการแก้ปัญหาและก็ให้ผู้ส่งตอบกลับ

เป้าหมายของระบบนี้ก็คือ ในการรับเมล์ที่ถูกส่งมาต้องมีความถูกต้องในการรับ 'bounce' (Address ในการตอบกลับ) ซึ่งปกติจะถูกระบุอยู่ใน Mail From : SMTP header line ซึ่งแสปมเมอร์ส่วนใหญ่จะใช้ที่อยู่ปลอมหรือใส่อะไรไปก็ได้ในการส่งดังนั้นจึงไม่สามารถที่จะมีการตอบกลับได้

เทคนิคนี้ไม่สามารถช่วยได้ถ้าพวกแสปมเมอร์ ใช้รหัสอีเมล์ที่ถูกต้องดังเช่นว่าถ้าใช้การควบคุมการส่งด้วยใช้ Zombie ในการทำงาน ในกลไก Challenge-response จะบล็อกอีเมล์ที่ส่งมาจากเครื่อง Zombie ถ้าพวกนี้ไม่สามารถตอบคำถามได้ถูกต้อง แต่อย่างไรก็ตามก็ยังง่ายต่อการแก้ไขปัญหาเพราะมีโปรแกรมที่สามารถช่วยตอบได้อย่างอัตโนมัติ ซึ่งถ้าเป็นอย่างนี้ก็จะทำให้กลไกไร้ประสิทธิภาพลงไป

แต่วิธีการ Challenge-response นั้นมีคนขัดแย้งมากมายเพราะทำให้การส่งเป็นไปได้ยากซึ่งก็จะเป็นที่น่ารำคาญแก่พวกที่ไม่ใช่เป็นแสปมเมอร์

เพื่อการหลีกเลี่ยงความน่ารำคาญจึงได้มีการรวม Challenge-response ไปอยู่ในส่วนของ MUA ซึ่งก็จะช่วยให้มีการตอบอัตโนมัติเพื่อความสะดวกขึ้น แต่ถึงอย่างไรก็ตามวิธีนี้ก็ไม่สามารถใช้กับZombie ได้ รวมถึงผู้ส่งก็ลงโปรแกรมที่ช่วยให้มีการส่งได้ต่อเนื่องหลายครั้ง ซึ่งสุดท้ายนี้ระบบนี้ก็น่าจะมีการเพิ่มRetransmissions, Synchronizeหรือการเปลี่ยนที่อยู่

- **Unlisted recipient email address**

ในขั้นตอนง่ายๆของ "Unlisted recipient email address" โดยผู้รับจะให้อีเมล์ที่รู้จักและการติดต่อที่น่าเชื่อถือโดยหวังว่าจะไม่ถูกใช้โดย แสปมเมอร์ ต่อมาได้มีการรวม 'disposable'ซึ่งก็สามารถจัดการกับอีเมล์ที่ถูกใช้โดยแสปมเมอร์ได้

แต่วิธีการนี้ยังมีระบบความปลอดภัยไม่มากพอ ก็คือ อย่างแรกผู้โจมตีสามารถค้นหาชื่อผู้ใช้ โดยการสแกนใน Mail Server ได้ซึ่งที่เรียกกันว่า dictionary attack อย่างที่สองอีเมล์สามารถส่งข้อความที่เหมือนกันไปสู่ผู้รับหลายๆคนได้

ในแบบ Advanced ผู้รับจะใช้เฉพาะ Unlisted Address ของผู้ส่งโดยเทคนิคนี้จะรวม 'Return Address validation'และ 'Unlisted Email Address' โดยเป้าหมายก็คือการแสดงที่อยู่ที่มีตัวตนกลับมาโดยใช้เฉพาะการรับจดหมายของผู้ส่งครั้งแรก โดยผู้รับก็จะตอบสนองไปยังผู้ส่ง ซึ่งอีเมล์ แอดเดรสก็มาอยู่ที่ผู้รับซึ่งก็จะเก็บรายละเอียดของผู้รับในลักษณะเฉพาะโดยที่อยู่ผู้ส่งก็จะถูก generate อัตโนมัติหรือบางทีก็ใช้กลไกของการถอดรหัสด้วย

2.2) Path-based Authentication of IP-address

อีเมล์ก็จะส่งโดยใช้โปรโตคอล SMTP โดยในการส่งจะต้องมีการตอบสนองคุณสมบัติของข้อความจากการรับอีเมล์มาโดยโปรโตคอล SMTP เป็นตัวติดต่อพื้นฐานโดยจะส่งข้อความผ่านโปรโตคอล TCPและโปรโตคอล TCP ก็จะให้ผู้ส่งตอบ packet จากผู้รับ โปรโตคอล TCPจะส่ง Packetโดยใช้โปรโตคอล IP (Internet Protocol) ซึ่ง IP addressของแต่ละpacket ก็จะไปถึงจุดหมาย เพราะฉะนั้นเราสามารถระบุการส่งโดยใช้ IP address ได้มี 2 ทางคือ

2.2.1 ทันทีที่ ได้รับอีเมล์ก็จะรู้ IP address ที่ใช้ส่งอีเมล์นั้นทันที

2.2.2 การใช้ 'track information'อยู่ใน email header

การพิสูจน์ตัวตนของ IP address นั้นจะไม่ปลอดภัยกับ MITM ซึ่งสามารถจะรับและตอบได้ซึ่งลักษณะจะเป็นตัวกลางระหว่างผู้สนทนาและหลอกว่าเป็นผู้สนทนาด้วยโดยจะหลอกผู้รับปลายทางว่าเป็นผู้ส่งและหลอกผู้ส่งว่าเป็นผู้รับปลายทาง

2.3) Path-based Authentication of Domain Name

วิธีการนี้จะควบคุมทางฝั่งของ Mail server ที่ทำการส่งอีเมล์ออกไปโดยหลักในการที่ส่งคือจะไม่ยอมส่งอีเมล์ที่มี MAIL FROM จาก Domain ของผู้รับที่ไม่ได้อยู่ใน list ของตัว Server ผู้ส่งซึ่งใน list นั้นสามารถที่จะแบ่งระดับได้ 3 ระดับคือ

-all อยู่ใน list ทั้งหมด IP addressจะปรากฏรายชื่อที่สามารถส่งเมล์ได้ในนามของโดเมน (อนุญาตให้ใช้โดเมน)

all อยู่ใน list ไม่ทั้งหมดโดยที่มีบางserverที่ให้ใช้ชื่อโดเมนได้เนื่องจากอาจมีอยู่ในlist ไม่ทั้งหมด

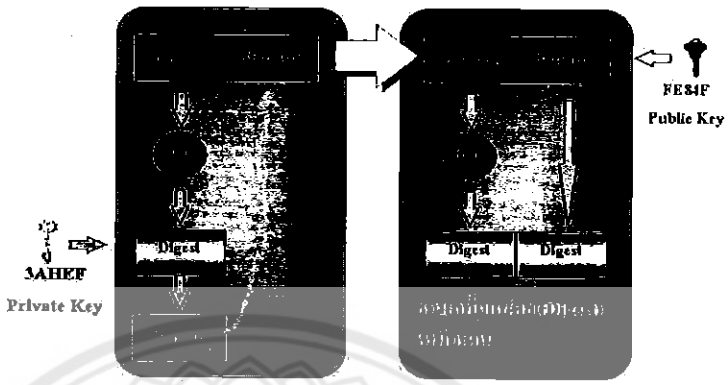
+all ไม่มีอยู่ใน list มีหนึ่งหรือหลาย serverที่ให้ใช้โดเมนโดยไม่มีชื่ออยู่ในlist

ซึ่งการพิสูจน์ตัวตนของวิธีนี้สามารถป้องกันพวก Mail Spoofingได้โดยการสร้าง Mail Server ที่จะส่งข้อความที่ได้รับอนุญาตของผู้ส่งที่ได้ระบุชนิดของDomain

- Cryptographic Email Authentication

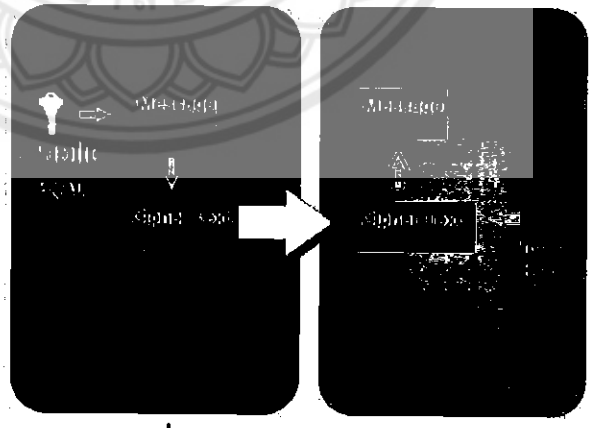
Cryptography คือ การทำให้ข้อมูลที่จะส่งผ่านไปทางเครือข่ายอยู่ในรูปแบบที่ไม่สามารถอ่านออกได้ด้วยการเข้ารหัส (Encryption) ทำให้ข้อมูลนั้นเป็นความลับ ซึ่งผู้ที่มีสิทธิ์จริงเท่านั้นจะสามารถอ่านข้อมูลนั้นได้ด้วยการถอดรหัส (Decryption) นั่นคือ สามารถรักษาข้อมูลให้เป็นความลับ(Confidentiality) และ กำหนดผู้มีสิทธิ์ได้ (Authentication & Authorization) สำหรับการเข้ารหัส และ ถอดรหัสนั้นจะอาศัยสมการทางคณิตศาสตร์ที่ซับซ้อน และต้องอาศัยกุญแจซึ่งอยู่ในรูปของพารามิเตอร์ที่กำหนดไว้ (สำหรับตัวกุญแจนั้นจะมีความยาวเป็น บิต(bit) และยิ่งกุญแจมีความยาวมาก ยิ่งปลอดภัยมาก เนื่องจากจะต้องใช้เวลานานมากขึ้นในการคาดเดากุญแจโดยผู้โจมตี)

คีย์ส่วนบุคคลหรือคีย์เดี่ยว (Private Keys หรือ Symmetric Keys) ผู้ส่งจะทำการเข้ารหัสข้อมูลด้วยคีย์ตัวใดตัวหนึ่งที่ กำหนดขึ้นเมื่อผู้รับได้รับข้อมูลก็จะต้องถอดรหัสข้อมูล โดยใช้คีย์ที่ส่งมาจากผู้ส่งที่ต้นทาง ซึ่งเป็นคีย์เดียวกันกับที่ใช้ในการเข้ารหัส ดังนั้นถ้ามีการดักจับคีย์ที่ผู้ส่งต้องส่งให้กับในผู้รับ ก็จะทำให้ข้อมูลไม่เป็นความลับอีกต่อไป



รูปที่ 2.5 ระบบคีย์เดี่ยว [6]

คีย์สาธารณะ (Public Keys หรือ Asymmetric Key) ระบบนี้ออกแบบมาเพื่อแก้ปัญหาของคีย์ส่วนบุคคล เพราะจะไม่มีการส่งคีย์กับใครทั้งสิ้น ระบบคีย์นี้ ได้ออกแบบให้แต่ละคนมีคีย์ 1 คู่ ประกอบด้วย Public Keys และ Private Keys โดย Public Keys ทุกคนสามารถรู้ได้ แต่ Private Keys จะต้องรักษาเป็นความลับ ผู้ส่งจะเข้ารหัสข้อมูลต้นฉบับด้วย Public Keysจากนั้นผู้รับจะใช้คีย์ของตนเอง (Private Keys) ในการถอดรหัสให้ได้ข้อมูลต้นฉบับเดิมอีก ครั้งหนึ่ง



รูปที่ 2.6 ระบบคีย์สาธารณะ [6]

- รูปแบบที่ใช้ในการ Authentication mail

Digital Signature

เป็นการเข้ารหัสข้อมูลเพื่อตรวจสอบเพื่อยืนยันว่าใครเป็นเจ้าของข้อความที่ส่งมาโดยใช้หลักการการเข้ารหัสข้อมูลด้วย Private Key ซึ่งจะมีเพียงผู้ส่งนั้นที่รู้ Private Key เมื่อผู้รับได้รับจะถอดรหัสได้ก็ต่อเมื่อใช้ Public Key ซึ่งเป็นคู่กันกับ Private Key ของผู้ส่งเท่านั้นถึงจะสามารถถอดรหัสลับนั้นได้ ซึ่งจะทำให้มั่นใจได้ว่าข้อมูลที่ถูกส่งมานั้นถูกส่งมาจากผู้ส่งที่มีตัวตนจริงที่ทราบ Private Key แน่นอน โดยจะใช้ร่วมกับ Message Digest เพื่อที่ใช้ในการตรวจสอบข้อมูลที่ได้รับ ทั้งนี้เพื่อป้องกันการแอบอ้างของพวก Attacker ที่คอยดักจับข้อมูลไปเปลี่ยนแปลงแก้ไข

การสร้าง Digital Signature

การสร้างกุญแจคู่ (Key Pairs) ก่อนการสร้าง Digital Signature นั้น ต้องมีการสร้างกุญแจคู่ขึ้นมาเสียก่อน ด้วยกระบวนการทางคณิตศาสตร์ โดยเจ้าของกุญแจคู่ จะต้องเก็บกุญแจแรกที่เรียกว่า "Private Key" ไว้เป็นความลับเพื่อให้ตนเองเท่านั้น สามารถใช้ Private Key ได้แต่ผู้เดียว ยกเว้นในกรณีของการมอบอำนาจให้บุคคลอื่นใช้ หรือ ในกรณีของนิติบุคคลซึ่งต้อง กระทำการผ่านบุคคลผู้มีอำนาจกระทำการแทน และ โดยปกติการเก็บรักษา "Private Key" นั้นก็มักจะบันทึกและ เก็บไว้ในสมาร์ทการ์ด ส่วน "Public Key" ก็จะเปิดเผยไว้ในระบบ ฐานข้อมูล ของผู้ประกอบการรับรอง (Certification Authority) เพื่อให้สามารถตรวจสอบตัวบุคคลได้โดยง่าย

ขั้นตอนการ Hash หรือย่อย (Hash Function) เป็นการนำข้อมูลในอีเมล์ที่ผู้ส่งส่งข้อมูลให้ผู้รับโดย นำมาคำนวณด้วยกระบวนการทางคณิตศาสตร์ (อัลกอลิทึ่ม) ที่เรียก ว่า "ขั้นตอนการ Hash (Hash Function)" หรือ One-way cryptography หรือ One-way hash function เพื่อ ทำให้ข้อมูล มีขนาดเล็กลงอัน จะทำให้ง่ายต่อการคำนวณทางคณิตศาสตร์และการจัดส่งให้ผู้รับข้อมูล ผลลัพธ์ที่ได้จากขั้นตอนการแฮช จะทำให้ได้ข้อมูลที่ย่อย (Message Digest) ซึ่งมีขนาดเล็กลง และ คงที่ (Fixed Length)

หลังจากนั้นก็นำกุญแจส่วนตัวมาทำการเข้ารหัสกับข้อมูลที่ Hash หรือ ย่อย (Message Digest) ซอฟต์แวร์ก็จะ ทำการแปลงข้อมูลเหล่านั้น ให้เป็น Digital Signatureและ Digital Signature นั้นก็จะมีลักษณะเฉพาะ ที่สัมพันธ์กับข้อมูล Hash และ Private Key กล่าวคือ ทุกครั้งที่ข้อมูล Hash หรือ Private Key เปลี่ยนแปลงไปจากเดิมDigital Signature ที่ได้ก็จะเปลี่ยนแปลงตามไปด้วย Digital Signature จึงไม่มีโอกาสซ้ำกันเลย

หลังจากสร้าง Digital Signature แล้ว ซอฟต์แวร์ก็จะทำการนำ Digital Signature ที่ได้นั้นไปแนบไว้ท้ายข้อความที่อยู่ใน รูปของข้อมูลอิเล็กทรอนิกส์ เพื่อใช้ส่งให้กับผู้รับข้อมูลต่อไป และเพื่อประโยชน์ในการตรวจสอบตัวบุคคล โดยปกติซอฟต์แวร์ก็จะถูกตั้ง โปรแกรมให้แนบกุญแจสาธารณะ และ ใบรับรองกุญแจสาธารณะของผู้ส่งข้อมูล ไปกับข้อมูลอิเล็กทรอนิกส์ พร้อมด้วย Digital Signature ด้วย เพื่อความสะดวกของผู้รับข้อมูล ในการตรวจสอบ Digital Signature นั้น

ดังนั้น ในการส่งข้อมูลอิเล็กทรอนิกส์โดยแนบ Digital Signature ไปด้วยนั้น ก็จะประกอบด้วยข้อมูลอิเล็กทรอนิกส์ถึง 3 ส่วน ได้แก่ ส่วนแรก คือ ข้อมูลอิเล็กทรอนิกส์ที่มีข้อความเดิม ซึ่งใช้ในการติดต่อสื่อสารระหว่างบุคคล อันเป็นข้อความที่อ่านออก และ เข้าใจได้ ส่วนที่สอง เป็น Digital Signature และ ส่วนสุดท้าย จะเป็น Private Key พร้อมใบรับรองกุญแจสาธารณะของผู้ลง Digital Signature

การตรวจสอบ Digital Signature

เมื่อผู้รับข้อมูลได้ข้อมูลอิเล็กทรอนิกส์ ที่มีการใช้ Digital Signature เพื่อยืนยันตัวผู้ส่งข้อมูลมาด้วย หากผู้รับข้อมูลอยากจะตรวจสอบข้อมูล ก็ทำได้โดยนำ Private Key ของผู้ส่งข้อมูลมาเข้ารหัสกับ Digital Signature และ เนื่องจาก Public Key นั้น มีความสัมพันธ์กับ Private Key เมื่อดำเนินการตามกระบวนการทางคณิตศาสตร์ ก็จะถอดรหัสออกมา และได้ผลลัพธ์ในรูปของ "ข้อมูล Hash" ในขณะเดียวกัน ข้อความที่ส่งมาในรูปของข้อมูลอิเล็กทรอนิกส์นั้น ก็จะถูก Hash ด้วยกระบวนการทางคณิตศาสตร์เช่นกัน ซึ่งจะได้ผลลัพธ์เช่นกัน คือ "ข้อมูล Hash" หากว่า "ข้อมูล Hash" ออกมา ตรงกันก็เป็นบทพิสูจน์ว่า บุคคลที่ส่งมาเป็นเจ้าของกุญแจส่วนตัว ซึ่งตรวจสอบได้ว่าเป็นผู้นั้นจริง

นอกจากประโยชน์ในการระบุตัวบุคคล และ ตรวจสอบตัวบุคคลข้างต้นแล้ว ประโยชน์อีกประการในการใช้เทคโนโลยีชนิดนี้ ก็คือ การตรวจสอบได้ว่ามีการแก้ไข เปลี่ยนแปลงข้อมูลหรือข้อความในอีเมล์หรือไม่ ซึ่งจะต้องมีขั้นตอนในการย่อยข้อความที่อ่านออกและเข้าใจได้เพื่อให้ได้ Message Digest ที่ได้จาก Private Key และ นำมาเปรียบเทียบกับ Message Digest ซึ่งเกิดจากการใช้ Public Key เข้ารหัสกับ Digital Signature นั้นต้องได้ข้อมูลที่ย่อยเหมือนกัน เสมอหากได้ค่าไม่เหมือนกัน แสดงว่ามีการเปลี่ยนแปลงแก้ไขข้อความนั้น

สรุปกระบวนการทำ Digital Signature

- ผู้ส่งเตรียมข้อมูลหรือข้อความเพื่อสร้าง Digital Signature
- นำข้อมูลข้างต้นมาย่อยหรือผ่าน Hash Function ให้ได้ค่าหนึ่ง
- ข้อมูลที่ผ่านการ Hash จะเรียกว่า Message Digest
- นำMessage Digest มาเข้ารหัสโดยใช้ Private Key ของผู้ส่ง
- ได้Digital Signature เพื่อเตรียมส่งให้ผู้รับ
- ส่งDigital Signature และข้อมูลตั้งต้นไปด้วยกัน
- เมื่อผู้รับได้รับDigital Signature และข้อมูลตั้งต้น
- ผู้รับจะถอดรหัสDigital Signature โดยใช้ Public Key ของผู้ส่ง
- จะได้Message Digest จากการถอดรหัส
- นำข้อมูลตั้งต้นที่ได้รับจากผู้ส่งมาทำการ Hash

- จะได้Message Digest ของข้อมูลตั้งต้นมาอีกหนึ่งชุด

- นำMessage Digest ทั้งสองมาเปรียบเทียบกันหากตรงกันก็แสดงว่าผู้ที่ส่งข้อมูลมามีตัวตนจริงที่รู้ private Key

Message Digest (MD)

มีลักษณะคล้ายกับการสรุปข้อความซึ่งก็จะมีความยาวน้อยกว่าข้อความข้างต้น โดยค่าที่ออกมาจะมีเป็นค่าๆหนึ่งที่มีขนาดเป็นบิต เพื่อใช้ระบุถึงข้อความตั้งต้นที่นำมาสร้าง ไดเจสต์ โดยค่าที่ได้จากการใช้ Hash function และการสร้างไดเจสต์จะไม่ขึ้นอยู่กับขนาดของข้อมูลตั้งต้นแต่ถ้าหากข้อมูลมากก็ต้องใช้เวลานาน จึงแก้ปัญหาโดยใช้MD ที่สูงกว่าเพราะว่าจะทำให้รวดเร็วมากขึ้น ซึ่งจุดประสงค์ของMD ก็เพื่อสร้างสรุปข้อมูลตั้งต้นเพื่อใช้เป็นตัวแทนของข้อมูลตั้งต้นนั้นๆซึ่งถ้าหากข้อมูลตั้งต้นต่างกันก็จะทำให้ค่าของ MD ที่ได้มีค่าต่างกันด้วย

คุณสมบัติ Message Digest (MD) ทุกบิตของ MD จะขึ้นอยู่กับข้อความตั้งต้นซึ่งถ้าข้อความตั้งต้นเปลี่ยนแปลง 1 บิตก็จะทำให้ค่าที่ได้ออกมาเปลี่ยนไปด้วยซึ่งก็จะทำให้ผู้รับรู้ทันทีว่าข้อความที่ได้รับไม่เหมือนกับข้อความตั้งต้น

MD5

MD5 เป็น Hashing อัลกอลิทึ่ม เป็นลักษณะของการนำข้อมูลมาเข้า Hash functionในการสร้าง Digest ขึ้นมาซึ่ง Digest จะเป็นข้อมูลที่มีความยาวคงที่ 128 bits ไม่ว่าจะมีข้อมูลมากเท่าไหร่ก็ตาม โดยMD5 เป็นฟังก์ชันในลักษณะของ One-way Function ซึ่งคำตอบย้อนกลับทำวิธีเดิมในลักษณะตรงกันข้าม หรือย้อนกลับเข้า Input อีกรอบ ก็จะไม่ได้ค่าเดิม

ลักษณะการทำงานของ MD5 มีลักษณะคือ ข้อมูล 1 ตัว เมื่อนำไปผ่าน MD5 จะได้ Digest ขึ้นมา 1 ตัว เช่น Thaiflashdev จะมี Digest เป็น 9e1afdaa0bac2ace1c692d711af10b6c และถ้าลองเปลี่ยนข้อมูลให้เป็น Thaiflashdef ก็จะได้ Digest เป็น a405441b6ed5edafcef1062cac13da33

จากตัวอย่างนั้นแสดงให้เห็นว่าการเปลี่ยนแปลงข้อมูลเพียงแค่ 1 บิต Digest ที่ได้จากMD5 ก็จะมีการเปลี่ยนแปลงไปอย่างมากซึ่งจะเป็นการยากในการแปลงข้อมูล

2.4) Blocking แสปมเมอร์ using Blacklist

เป็นเทคนิคสำหรับการระบุแหล่งที่มาของเมล์หรือของกลุ่มคนที่ส่งเมล์ให้แน่นอนซึ่งจะใช้ในการรับหรือปฏิเสธการรับเมล์ซึ่งจะใช้การ Blacklist ผู้ส่งหรือDomainที่มีความไม่น่าเชื่อถือหรือน่าสังสัยว่าจะเป็น แสปมเมอร์

- **Sender Blacklists:** ผู้ส่งที่อยู่ใน Black list นั้นก็จะเป็นที่อยู่เมล์ที่เป็นของพวกแสปมเมอร์ อย่างไรก็ตาม การBlack list ก็ยังมีข้อจำกัดในด้านพื้นที่เนื่องจากพวกแสปมเมอร์ มีการเปลี่ยนอีเมล์แอดเดรส อยู่ตลอดเพราะมีหลายเว็บที่ให้บริการ อีเมล์แอดเดรส ฟรีอยู่มากมายซึ่งเป็นการยากที่จะทำการ Blacklist

- **Domain Blacklist:** เทคนิคนี้จะว่าเมล์ที่ส่งมานั้นอยู่ใน Blacklists ว่าเป็นพวกแสปมหรือไม่โดยจะบันทึก Domain Nameที่แสปมเมอร์ส่งมาแต่ปัญหาก็คือในการขอ Domain Name เป็นเรื่องง่ายและใช้ค่าใช้จ่ายที่ต่ำซึ่งเป็นการยากที่จะตรวจสอบแต่อย่างไรก็ตามผู้ให้บริการด้าน Domain Name ก็มีการป้องกันอย่างดีเพื่อไม่ให้ Domain Name ของตัวเองต้องอยู่ใน Blacklist เช่น พวก Hotmail , gmail

วิธีการ Blacklist จำนวนมากใช้ Domain Name System (DNS) โดย list พวก Spamming domain

ซึ่งง่ายต่อการดำเนินการและง่ายต่อการพิจารณาลักษณะกลไกอื่นๆที่จะใช้ควบคู่กันไป

วิธีนี้ Userต้องแน่ใจว่า list ถูกต้องจริงๆโดยต้องหลีกเลี่ยง False Positive และ False Negativeเพราะถ้าเกิดข้อผิดพลาดก็จะทำให้เกิดความเสียหายได้ซึ่งก็ไม่ง่ายเลยในการที่จะพิจารณาเพราะพวก Attacker สามารถที่จะหาเทคนิคต่างๆที่เรียกว่า "Poison" ในการที่จะทำให้ Server ไม่คิดว่าเป็นเมล์ที่เป็นแสปมซึ่งBlacklists นั้นก็ต้องมีความน่าเชื่อถือด้วย ซึ่งการ Blacklists นั้นก็จะมีกฎเกณฑ์ในการใช้ที่แตกต่างกันอย่างเช่น

- list เฉพาะที่มีการส่งข้อมูลที่ส่งไปที่ต่างๆ
- list IP Addressซึ่งถูกใช้ในการส่งพวกแสปม
- list IP Address ที่ถูก Block เนื่องจากมีการใช้ส่งเมล์ที่เป็นแสปม
- list Block IP Address ที่ติดต่อซึ่งไม่คาดว่าจะมีการดำเนินการใน Mail Server

ซึ่งวิธีนี้ก็ไม่สามรถที่จะหลีกเลี่ยงการโจมตีของพวก Zombie ได้ทั้งหมดรวมถึงเป็นวิธีที่ยังไม่ค่อยมีความแม่นยำในการตรวจสอบโดยบางBlacklist นั้นเก็บ IP Block ไว้เป็นเนื้อที่ขนาดใหญ่ซึ่งจะใช้เนื้อที่เยอะซึ่งก็เป็นเหตุในการทำลายพวก Userที่ไม่ใช่คนส่งพวกแสปม

2.5) Flitering based on White-list , Reputation and Accreditation Services

- Sender White-list เป็นเทคนิคที่ผู้รับเก็บ อีเมล์แอดเดรส ของผู้ส่งที่ไม่ใช้พวกแสปมเมอร์ ไว้(ซึ่งปราศจากการผ่าน filtering)ซึ่งการใช้เทคนิคนี้ก็จะช่วยลดความเสี่ยงจากการเกิด False Positive ลงได้เมื่อมีการใช้ Content filteringช่วย อย่างไรก็ตาม เทคนิคนี้จะไม่มีประสิทธิภาพถ้าผู้ส่งทำการเปลี่ยนแปลง Account กับอีเมล์แอดเดรส ที่ถูกต้องที่ถูกบันทึกถูกไวรัสจากพวกแสปมเมอร์ซึ่งก็จะทำให้เกิดการส่งไปโดยที่ผู้ใช้ไม่รู้ตัว ตัวอย่างของวิธีนี้เช่น Bob ติดต่อกับ Alice ซึ่งในการเพิ่มALICE@WONDERLAND.NET โดยจะทำการตรวจสอบว่าเคยอยู่ใน Blacklistหรือไม่รวมถึงตรวจสอบโดยใช้Content Filtering ซึ่งจะต้องได้รับการอนุญาตก่อนที่จะมีการส่งข้อความไปสู่ผู้รับ

Whitelisting จะช่วยป้องทำให้ปลอดภัยจากการใช้ Spoofing (ใช้ชื่อปลอม) ถ้ามีการใช้พวก Public Keys เช่น Bob สามารถใช้ Public Keyของ Alice เพื่อเป็นการระบุตัวตนด้วย

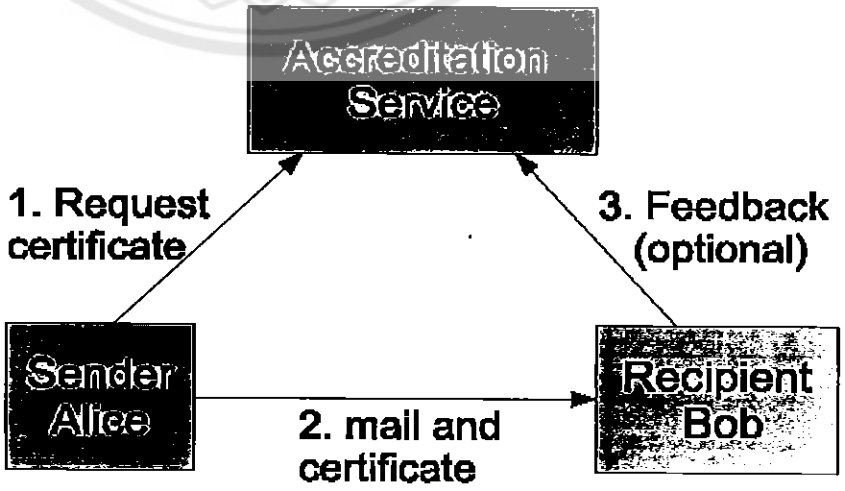
- **Reputation** ผู้รับจะให้ความน่าเชื่อถือและตอบรับการติดต่อเมื่อผู้ส่งทำการ Proof of Reputation จากผู้ให้บริการโดยผู้รับควรจะมี Reputation service อย่างน้อยหนึ่งที่ก่อนจะให้ความเชื่อถือ การพิสูจน์ของ Reputation นั้นปกติจะใช้ Digital Signatureโดย Reputation Serviceซึ่งอยู่บน Reputation Record RRกล่าวคือ $Sign_{RS,s}(RR)$โดย Sign ก็คือSignature อัลกอลิทึ่ม และ RR คือ Signature Key ซึ่งเป็นความลับของระดับของ RS (Rating Service)ซึ่งระบบนี้น่าจะมี method อื่นในการที่จะทำให้ผู้รับบันทึกได้อย่างน่าเชื่อถือ เช่นการใช้สัญลักษณ์ทาง digital , การป้องกันโดยการตั้งคำถามของฝั่งผู้รับหรือการใช้ลักษณะของ Reputation Record ทาง Domain Name System (DNS)

Reputation Record RR นั้นจะระบุบันทึกของผู้ส่ง เช่น คีย์สาธารณะ, อีเมล์ แอดเดรส หรือ/และ IP addressซึ่งก็จะมีการจัดระดับและประวัติของข้อมูล เช่นถ้าอยู่ในความน่าเชื่อถือระดับสูงก็แสดงว่าไม่ใช่แสปมเมื่อ Public Keys ที่อยู่ในการบันทึกนั้นก็จะเป็นที่น่าเชื่อถือซึ่งในการส่งข้อความก็จะไม่มีการเปลี่ยนแปลงและจะต้องมีการตรวจสอบSpam และดูว่ากลุ่มต่างๆที่ติดต่ออยู่ในข่ายหรือไม่ เช่น blacklist

สิ่งที่ควรเลี่ยงของวิธีนี้ก็คือ False Positive ซึ่งระบบควรจะมีการจัดการอย่างมีประสิทธิภาพเพื่อให้เกิดความน่าเชื่อถือต่อผู้ใช้บริการ

- **Accreditation Service for High-Risk Senders** มีจุดมุ่งหมายว่าผู้ส่งที่ดูไม่น่าเชื่อถือหรือเป็นพวกส่ง Spam จะต้องมีการเสียค่าใช้จ่ายมากขึ้นในการที่จะส่ง Spam

ซึ่งขั้นตอนก็คือผู้ส่งจะทำการเพื่อแสดงความน่าเชื่อถือต่อ Accreditation Service ซึ่งผู้ให้บริการก็จะทำการตรวจสอบว่ามีความน่าเชื่อถือหรือไม่ก่อนที่จะให้ส่งข้อความไปหาผู้รับซึ่งวิธีนี้ผู้รับก็สามารถแสดง Feedback กลับไปยังผู้ให้บริการได้ว่าเมล์ที่ส่งมาจากผู้ส่งนั้นเป็น Spam หรือไม่ถ้าเป็นหรือเข้าข่ายก็แจ้งให้ทำการ Blacklist ไว้เลยเป็นต้น



รูปที่ 2.7 กลไกของ Accreditation Service [9]

3) Cost-Based Spam Controls

เนื่องจากการรับส่งอีเมล์เป็นกระบวนการติดต่อสื่อสารที่มีราคาถูก และมีความสะดวกในการใช้งาน จึงเป็นเหตุผลหนึ่งที่ แสปมเมอร์ ใช้การสื่อสารด้วยอีเมล์เพื่อผลประโยชน์ทางเศรษฐกิจ ซึ่ง Cost-Based Spam Controls นั้นเป็นวิธีทางหนึ่งที่จะควบคุม Spam ได้คือ การเรียกร้องค่าใช้จ่าย ซึ่งอาจจะเป็นการจ่ายเงินจริงหรือค่าใช้จ่ายในแบบอื่นๆสำหรับการส่งข้อความ

อย่างไรก็ตามผู้ใช้ไม่เห็นด้วยที่จะเสียค่าใช้จ่ายในการส่งอีเมล์ซึ่งโดยปกติจะเป็นการใช้ฟรี เราอาจจะแก้ปัญหานี้ได้โดยการใช้ Protocol ที่รับรองความปลอดภัยหรือทำให้มั่นใจว่า ผู้ใช้จะเสียค่าใช้จ่ายเฉพาะอีเมล์ที่เป็น Spam เท่านั้น หรืออาจจะมีการจ่ายคืนให้สำหรับอีเมล์ที่ไม่ใช่ Spam ซึ่งวิธี Cost-Based Spam Controls จะแบ่งออกเป็น 3 วิธีดังนี้

3.1) Non – Monetary Cost Mechanisms

Non – Monetary Cost Mechanisms เป็นกระบวนการที่ได้รับการยอมรับจากผู้ใช้ เพราะว่าไม่ได้คิดค่าใช้จ่ายเป็นเงินแต่คิดในรูปแบบอื่นคือ

- Cost of Human Effort and Human Interaction Proofs (HIP)

เหตุผลหนึ่งที่ทำให้การ Spam มีราคาไม่แพง เพราะว่าจะเป็นการส่งในรูปแบบไปรษณีย์ขนาดใหญ่ ซึ่งส่งโดยอัตโนมัติโดยโปรแกรม ดังนั้นจึงมีวิธีการที่จะควบคุม Spam นั่นก็คือให้มีการใช้เวลาในการทำงานบางอย่างก่อนที่จะมีการส่งอีเมล์ซึ่งจะเป็นการป้องกันการส่งอีเมล์ไปหลายๆ ผู้รับ

การควบคุม Spam นี้จะทำให้ผู้ส่งต้องดำเนินการบางอย่างซึ่งเป็นสิ่งที่ง่ายสำหรับมนุษย์ แต่ยากที่จะดำเนินการโดยการใช้โปรแกรมคอมพิวเตอร์ ซึ่งการดำเนินการนี้จะถูกเรียกว่า Human Interaction Proofs (HIP) ซึ่งเป็นวิธีการที่ต้องการความพยายามในการทำงานของมนุษย์มากกว่าการทำงานอัตโนมัติโดยโปรแกรม

Human Interaction Proofs (HIP) ลักษณะของ HIP ที่รู้จักกันดีคือ CAPTCHA ซึ่งจะเป็นรูปภาพที่ประกอบด้วยตัวอักษรซึ่งถูกแสดงในรูปแบบที่บิดงอ ซึ่งเป็นรูปแบบที่มนุษย์สามารถเข้าใจได้ แต่คอมพิวเตอร์ไม่สามารถเข้าใจได้ ดังตัวอย่างต่อไปนี้

รูปที่ 2.8 รูปที่ใช้ในการพิสูจน์ตัวตนโดยมนุษย์ [9]

โดยเป้าหมายของการใช้ HIP ในการต่อต้าน Spam คือ การทำให้ แสปมเมอร์ใช้เวลาค่อนข้างมากในการทำงานให้เสร็จ เนื่องจากต้องใช้การจัดการโดยมนุษย์ ดังนั้นการSpamming จึงไม่เกิดผลประโยชน์ใด

ซึ่งโดยปกติของ CAPTCHA แล้วจะเป็นลักษณะที่เกี่ยวข้องกับความสามารถในการจำแนกออกด้วยการมองเห็นของมนุษย์ โดยอุปสรรคหรือข้อบกพร่องที่เป็นไปได้ของวิธีนี้คือ การทำให้เกิดความยุ่งยากสำหรับผู้ส่ง ซึ่งบางคนอาจจะปฏิเสธหรือไม่สามารถทำงานนี้ให้สำเร็จได้ เช่น การขาดความสารถในการตอบสนองที่ถูกต้องต่อ HIP ได้ของผู้ใช้ที่ตาบอดหรือมีสติปัญญาไม่สมบูรณ์ มากกว่านั้นมันสามารถทำการปรับปรุงโปรแกรมหรือ อัลกอลิทึ่มซึ่งอาจจะทำให้สารถแก้ปัญหา HIP โดยอัตโนมัติ โดยไม่ต้องการการตอบสนองของมนุษย์ได้

- **Cost of Computation, or Proof of Computational Work**

เป็นวิธีการที่ต้องการการตรวจสอบประวัติการใช้งานหรือตรวจสอบแหล่งที่มาของผู้ใช้ก่อนการอนุญาตให้ใช้บริการ เช่น ก่อนกระบวนการรับข้อความหรือการเปิด Free Web mail Account ซึ่งเป็นการตรวจสอบแหล่งหรือข้อมูลที่ผู้โจมตีใช้ เช่น จำนวนข้อความที่เป็น spam ที่ถูกส่งโดยผู้โจมตีหรือจำนวน account ที่เปิด Web Mail โดยผู้โจมตี (ข้อความที่ส่งก็จะมีแต่ละ account ในการส่ง) ซึ่งจะเป็นการการป้องการการนำกลับมาใช้ใหม่ควรจะมีการตั้งคำถาม(Challenge) โดยเลือกจากผู้ให้บริการ ซึ่ง Challenge นี้อาจจะเป็นข้อความในอีเมล์ง่ายๆซึ่งรวมถึงผู้รับและวันเวลาที่ส่ง ซึ่งอนุญาตให้ปลายทางสามารถป้องกันการนำกลับมาใช้ใหม่ได้ หรือ อาจจะเลือกโดยผู้ให้บริการ เช่น Web Mail Server

เพื่อที่จะให้ได้ประสิทธิภาพสูงสุดของวิธีการนี้ จะต้องใช้เนื้อที่ในการเก็บข้อมูลขนาดใหญ่ ซึ่งไม่เหมาะสมที่จะเก็บในแรม ผลก็คือ ในการคำนวณนี้จะจัดเก็บไว้ในหน่วยความจำสำรอง เช่น disk โดยเมื่อข้อมูลมีขนาดใหญ่ขึ้นเรื่อยๆความเร็วในการทำงานก็จะลดลงไปด้วยและอีกปัญหาหนึ่งของ Cost of computation และ Cost of Human effort คือ ผู้ส่งที่ไม่ใช่แสปมเมอร์ จะทำการส่งข้อความน้อยลงเนื่องจากต้องเสียค่าใช้จ่าย รวมถึงต้องเสียเวลาในการต้องพิสูจน์การใช้งานของตนเอง

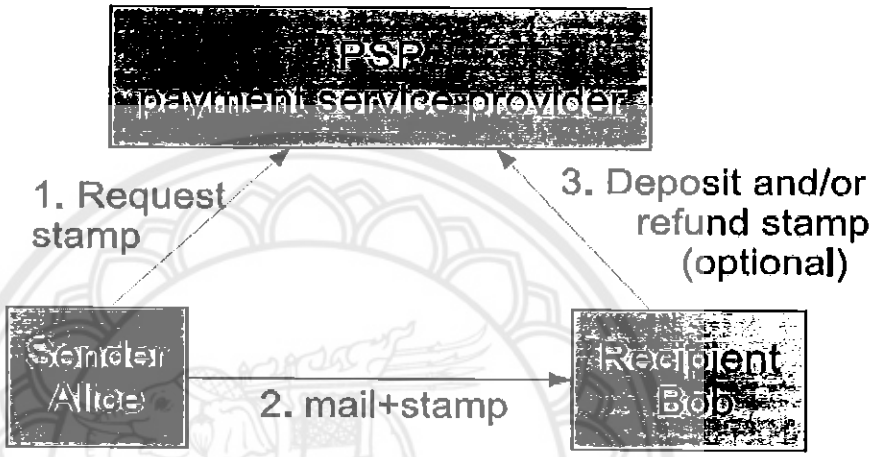
3.2) Postage Protocols

เมื่อแสปมเมอร์ สู้ที่จะเสียค่าใช้จ่ายในการส่งอีเมล์ขยะ จึงมีวิธีการที่จะควบคุมอีเมล์ขยะดังกล่าวได้ 2 วิธีคือ

วิธีการแรกคือการคิดค่าใช้จ่ายเฉพาะอีเมล์ขยะ (ใช้ Penalty Protocol) เป็นวิธีที่ได้รับความนิยมมากกว่า เนื่องจาก จะไม่ทำการเก็บค่าปรับจากผู้ส่งอีเมล์ที่ไม่ใช่แสปม แต่ก็ต้องมีความมั่นใจว่าข้อความไหนคืออีเมล์ขยะ

วิธีการที่สองคือ การเรียกเก็บค่าใช้จ่ายที่ผู้ส่งสำหรับแต่ละข้อความ ซึ่งจะคล้ายกับค่าไปรษณีย์ แต่จะมีการคืนค่าใช้จ่ายหรืออย่างน้อยก็ทำการลดค่าใช้จ่ายสำหรับข้อความที่ถูกระบุว่าไม่ใช่อีเมล์ขยะ ซึ่งเราเรียกวิธีการนี้ว่า Refundable Postage Protocol

ได้มีการนำเสนอการออกแบบอย่างง่ายของ Postage-based spam control ดังรูปต่อไปนี้



รูปที่ 2.9 กลไกของ Postage Protocols [9]

ซึ่งจะมีวิธีการดังต่อไปนี้

1. ผู้ส่งหรือ Mail Agent ของผู้ส่งร้องขอแสตมป์สำหรับการส่งอีเมล์จาก Payment Service Provider (PSP) ของผู้ส่ง ซึ่งแสตมป์คือ สิ่งที่ยืนยันการจ่ายข้อความจาก PSP ซึ่งจะทำให้ผู้รับมีความมั่นใจว่าข้อความนี้ถูกส่งโดยผู้ส่งและต้องการส่งถึงผู้รับจริงๆ
2. Mail Agent ของผู้ส่งจะส่งอีเมล์พร้อมกับแสตมป์ไปที่ผู้รับ ซึ่งผู้รับหรือ Mail Agent ของผู้รับจะทำการตรวจสอบอีเมล์ที่ส่งมาเมื่อไม่ใช่อีเมล์ขยะก็จะทำการยอมรับข้อความกับค่าไปรษณีย์จาก PSP และผู้รับก็จะทำการบันทึกว่าผู้ส่งคนนี้ไม่ได้อยู่ในข่ายของบุคคลที่เป็นแสปมเมอร์
3. จากนั้นผู้รับจะทำการติดต่อกับ PSP เพื่อที่จะบอกกลับไปว่าข้อความที่ส่งมานั้นเป็นอีเมล์ขยะหรือไม่ ถ้าไม่ใช่ก็จะทำการคืนแสตมป์ไปที่ PSP

หลังจากนั้น Mail Agent ของผู้รับจะแสดงข้อความที่ผู้รับ ซึ่ง Mail Agent ของผู้รับจะรู้ว่าข้อความนี้เป็นข้อความจริงหรือเป็นอีเมล์ขยะ ถ้าข้อความที่ส่งมาเป็นอีเมล์ขยะผู้รับจะทำการจัดผู้ส่ง

ให้อยู่ใน Blacklist ดังนั้นอีเมล์ที่ถูกส่งมาจากผู้ส่งจะถูกปฏิเสธ (แสตมป์ที่ถูกแนบมาจะถูกป้องกันการใช้ซ้ำสำหรับแสปมเมอร์ ในขณะที่จะอนุญาตให้ใช้ซ้ำได้สำหรับผู้ที่ไม่ใช่แสปมเมอร์)

และถ้าข้อความนี้ไม่ใช่ อีเมล์ขยะ Agent ของผู้รับจะทำการติดต่อกับ PSP และทำการคืนค่าใช้จ่ายให้กับผู้ส่ง ซึ่งแบบแผนที่ผู้รับทำการคืนค่าใช้จ่ายให้ผู้ส่งที่ไม่ใช่แสปมเมอร์ จะถูกเรียกว่า Refundable อย่างไรก็ตามผู้ส่งอาจได้คืนค่าใช้จ่ายโดยรับเป็นอีเมล์ข้อความจากผู้รับ

บทที่ 3
วิธีการดำเนินงาน

ในบทที่ 2 เราได้กล่าวถึงหลักการและทฤษฎีที่เกี่ยวข้องกับขั้นตอนของการส่งอีเมล์ รูปแบบการโจมตีของพวกแสปมเมอร์ และรูปแบบของโปรโตคอลที่มีการใช้ในการติดต่อสื่อสารกันอยู่ในปัจจุบัน โดยในบทนี้จะกล่าวถึงวิธีการดำเนินการในการออกแบบโปรโตคอลซึ่งแบ่งเป็นขั้นตอนได้ดังนี้

1. ศึกษาขั้นตอนการส่งอีเมล์ และศึกษารูปแบบการโจมตีของแสปมเมอร์
2. ศึกษาโปรโตคอลรูปแบบต่างๆ เพื่อศึกษาวิธีการทำงาน, ข้อดีข้อเสีย สำหรับนำมาใช้เป็นหลักเกณฑ์ในการออกแบบโปรโตคอล
3. ทำการออกแบบโปรโตคอล
4. ตรวจสอบปัญหาและจุดบกพร่องของโปรโตคอลที่ออกแบบ
5. ปรับปรุงโปรโตคอลและแก้ไขจุดบกพร่อง
6. ออกแบบลักษณะของ Puzzle
7. เสนอการทดลองให้เห็นเป็นตัวอย่างในการใช้จริง

3.1 ศึกษาขั้นตอนการทำงานของระบบอีเมล์

จากการศึกษาขั้นตอนในการส่งอีเมล์ พบว่า ปัญหา อีเมล์ขยะ เกิดจากการที่การส่งอีเมล์นั้นเสียค่าใช้จ่ายน้อยมากถ้าเทียบกับการส่งจดหมายทั่วไปและสามารถส่งได้ที่ละจำนวนมากในเวลาที่รวดเร็วดังนั้นจึงเป็นสาเหตุให้มีการส่งอีเมล์ เช่น โฆษณาขายสินค้าต่างๆ, จดหมายลูกโซ่รวมถึงจดหมายในเชิงไม่สร้างสรรค์ เป็นต้น ซึ่งสร้างปัญหาและก่อความรำคาญต่อผู้ใช้อย่างมาก

3.2 ศึกษารูปแบบการโจมตีของแสปมเมอร์

โดยรูปแบบในการโจมตีนั้นมีหลายแบบซึ่งแสปมเมอร์ จะโจมตีช่องโหว่ของเทคนิคต่างๆ ที่ใช้ป้องกันอีเมล์ขยะ เช่น การใช้พวกมาลแวร์ ซึ่งเป็นการทำให้เครื่องผู้ใช้ติดไวรัสและส่งข้อความโดยไม่รู้ตัวซึ่งก็คล้ายกับว่าเป็นผู้ส่งอีเมล์ที่เป็นพวกแสปมเมอร์และยังมีอีกหลายวิธีเช่น MIMT รวมถึงการดักจับข้อมูลในขณะที่มีการส่งอีเมล์กัน

3.3 ศึกษาการทำงานของเทคนิคและวิธีการต่างๆที่ใช้ป้องกันอีเมล์ขยะ

ทำการค้นหาและศึกษาจากผลงานทางวิชาการ ที่ได้มีการนำเสนอซึ่งจะมีรูปแบบต่างๆของโปรโตคอลที่มีใช้กันอยู่ในการติดต่อสื่อสารรวมถึงเทคนิควิธีการต่างๆในการป้องกันอีเมล์ขยะซึ่งความรู้เหล่านี้สามารถช่วยในการออกแบบโปรโตคอลและมีการเปรียบเทียบข้อดีข้อเสียในแต่ละแบบเพื่อช่วยในการตัดสินใจว่าจะนำแบบไหนมาเป็นส่วนช่วยในการปรับปรุงพัฒนาโปรโตคอลที่ออกแบบให้ดียิ่งขึ้น เพื่อทำให้โปรโตคอลที่เราทำการออกแบบนั้นมีข้อบกพร่องในการที่จะถูกโจมตีของพวกแสปมเมอร์ ให้น้อยที่สุดหรือทำให้เสียเวลาในขั้นตอนการส่งข้อความให้มากที่สุดสำหรับพวกที่มีความเชื่อว่าเป็นอีเมล์ขยะซึ่งเราได้ทำตารางเปรียบเทียบในแต่ละรูปแบบได้ดังนี้

จากการศึกษาเทคนิคที่ใช้ในการควบคุมอีเมล์ขยะ พบว่าสามารถแบ่งเทคนิคต่าง ๆ ได้เป็น 3 ประเภท ซึ่งสามารถวิเคราะห์เปรียบเทียบ ข้อดี – ข้อเสีย ของเทคนิคแต่ละประเภทได้ดังนี้

ตารางที่ 3.1 เปรียบเทียบข้อดี-ข้อเสีย ของวิธีการที่ใช้ป้องกันแสปมเมล์

รูปแบบในการป้องกัน	ข้อดี	ข้อเสีย
Content Filtering	- สะดวกในการส่งอีเมล์ของผู้รับ- ผู้ส่งเพราะไม่ต้องผ่านขั้นตอนใดๆเพิ่มเติมโดยเป็นหน้าที่ของทาง Server ในการตรวจสอบอย่างเดียว - ป้องกันการโจมตีแบบพวก Zombie ได้	- มีประสิทธิภาพน้อยในการตรวจสอบถ้ามีการส่งข้อความประเภทพวกรูปภาพ - สามารถใช้ข้อความที่หลีกเลี่ยงการตรวจของ Filter ได้ เช่น VIRGA ก็สามารถที่จะเปลี่ยนเป็น V*I*R*G*A ได้ - เป็นวิธีการที่ละเมิดสิทธิส่วนบุคคลเพราะต้องเข้าไปทำการอ่านข้อความเพื่อรวบรวมข้อมูลของผู้ใช้
Identity-based	- สามารถเพิ่มความน่าเชื่อถือให้กับผู้ส่งได้ว่าเป็นอีเมล์ที่ส่งจากบุคคลจริงและที่อยู่จริงไม่ใช่เป็นการปลอมอีเมล์มา	- ไม่สามารถป้องกันวิธีการโจมตีแบบ MITM(Man In The Middle) ได้อย่าง มีประสิทธิภาพ - สามารถถูกดักจับข้อมูลหรือถูกจับ

ตารางที่1(ต่อ) เปรียบเทียบข้อดี-ข้อเสีย ของวิธีการที่ใช้ป้องกันแสปมเมล์

รูปแบบในการป้องกัน	ข้อดี	ข้อเสีย
	- ไม่เป็นการละเมิดสิทธิส่วนบุคคลเพราะไม่ได้ไปทำการอ่านข้อมูลในตัวอีเมล์	Packet ได้ซึ่งก็สามารถเข้าไปเปลี่ยนข้อความในอีเมล์ - ยังไม่สามารถป้องกันอีเมล์ที่ถูกส่งมาจากเครื่องที่ติดมาลแวร์ หรือที่เรียกว่า Zombie - ยากต่อการนำไปปฏิบัติจริง - บางเทคนิคในรูปแบบนี้ต้องมีการบันทึกข้อมูล เช่น การบันทึกพวก Blacklist White list ซึ่งจะทำให้สิ้นเปลืองเนื้อที่ในการจัดเก็บถ้าข้อมูลเริ่มมากขึ้นเรื่อยๆ
Cost-Based	- ไม่เป็นการละเมิดสิทธิส่วนบุคคลเพราะไม่ได้ไปทำการอ่านข้อมูลในตัวอีเมล์ - ทำให้การส่งอีเมล์Spam ต้องมีการเสียค่าใช้จ่ายมากขึ้นซึ่งอาจทำให้การส่งอีเมล์ที่เป็น Spam ลดน้อยลง - สามารถลดการส่งอีเมล์ที่ไร้สาระได้เพราะต้องผ่านขั้นตอนต่างๆก่อนการส่ง - มีการยืนยันว่ามนุษย์เป็นผู้ใช้งาน ไม่ใช่โปรแกรมคอมพิวเตอร์ โดยการใช้ CAPTCHA	- ไม่สะดวกในการส่งอีเมล์เพราะต้องผ่านขั้นตอนต่างๆก่อนการส่ง - อาจโดนแกล้งจากผู้รับถ้ามีการ Feedback กลับไปว่าผู้ส่งเป็นผู้ที่ส่งอีเมล์ที่เป็นSpam ซึ่งก็จะไม่ได้ค่าใช้จ่ายคืน - สร้างความลำบากต่อผู้ที่เกิดความบกพร่องทางกาย เช่น ตาบอดซึ่งไม่สามารถมองเห็นภาพได้ - ไม่สามารถป้องกันการดักจับข้อมูลระหว่างการส่งข้อมูลได้

เทคนิคที่นำมาใช้เป็นรูปแบบในการออกแบบโปรโตคอล

จากการเปรียบเทียบข้อดี-ข้อเสียของเทคนิคต่างๆนั้นเราได้เลือกวิธี Cost-Based มาใช้เป็นส่วนหลักในการออกแบบโปรโตคอลเพราะว่า ต้องการให้ในการส่งข้อความหรืออีเมล์ในแต่ละครั้งนั้นต้องมีการทำงานก่อนการส่งหรือเสียค่าใช้จ่ายก่อนการส่งเพื่อที่จะทำให้การส่งแต่ละครั้งมีความยากลำบากมากขึ้นด้วยซึ่งในเทคนิคนี้สามารถแบ่งได้ดังนี้

1. Non – Monetary Cost Mechanisms

ข้อดี

1.ให้มีการทำงานก่อนการส่งเพื่อที่จะได้มีการเสียค่าใช้จ่ายและเสียเวลาซึ่งก็จะทำให้การส่งนั้นยุ่งยากมากขึ้นซึ่งอย่างน้อยก็สามารถที่จะลดการส่งอีเมล์ที่ไม่มีสาระได้รวมถึงเมล์ที่เป็นSpamด้วย

2. จะใช้รูปภาพที่ประกอบด้วยตัวอักษรซึ่งถูกแสดงในรูปแบบที่บิดงอ (CAPTCHA) ซึ่งเป็นรูปแบบที่มนุษย์เข้าใจได้แต่คอมพิวเตอร์ไม่สามารถเข้าใจได้ ซึ่งถือว่าเป็นการพิสูจน์ว่าเป็นบุคคลจริง

ข้อเสีย

1. วิธีนี้อาจมีปัญหาเนื่องจากต้องใช้คอมพิวเตอร์มีประสิทธิภาพสูงจึงจะสามารถทำงานได้รวดเร็วโดยวิธีนี้ส่วนหนึ่งก็จะขึ้นอยู่กับทรัพยากรในการส่งด้วย

2. วิธีนี้อาจมีปัญหากับผู้บกพร่องทางร่างกาย เช่น ตาบอด ซึ่งไม่สามารถมองเห็นรูปภาพที่ใช้ในการตรวจสอบได้

3. วิธีนี้อาจสร้างความรำคาญกับผู้ส่งอีเมล์ที่ไม่ใช่พวก แสปมเมอร์ เนื่องจากต้องเสียเวลาในการที่จะส่งอีเมล์ในแต่ละครั้ง

4. อาจถูกปรับปรุงโปรแกรมให้คอมพิวเตอร์สามารถทำการเรียนรู้ลักษณะของ CAPTCHA ได้

2. Postage Protocols

ข้อดี

1. การขอ Stamp นั้นก็จะเป็นการยืนยันว่าเป็นบุคคลจริงอีกทางหนึ่งซึ่งจะเสียค่าใช้จ่ายในการส่งถ้าเป็นSpam ซึ่งก็จะช่วยให้การส่ง Spam ลดน้อยลง

2. ผู้รับอีเมล์สามารถทำการfeedback กลับมาให้ผู้บริการได้ว่าอีเมล์ที่ส่งมาจากผู้ส่งนั้นๆเป็น Spam หรือไม่

ข้อเสีย

1. แสตมป์นั้นอาจถูกกลับนำมาใช้ซ้ำได้ซึ่งวิธีนี้ยังไม่ได้มีการป้องกัน

2. แสตมป์นั้นอาจถูกดักจับข้อมูลซึ่งผู้ดักจับที่เป็นแสปมเมอร์ อาจนำแสตมป์นั้นมาใช้ส่งอีเมล์ที่เป็น Spam ได้

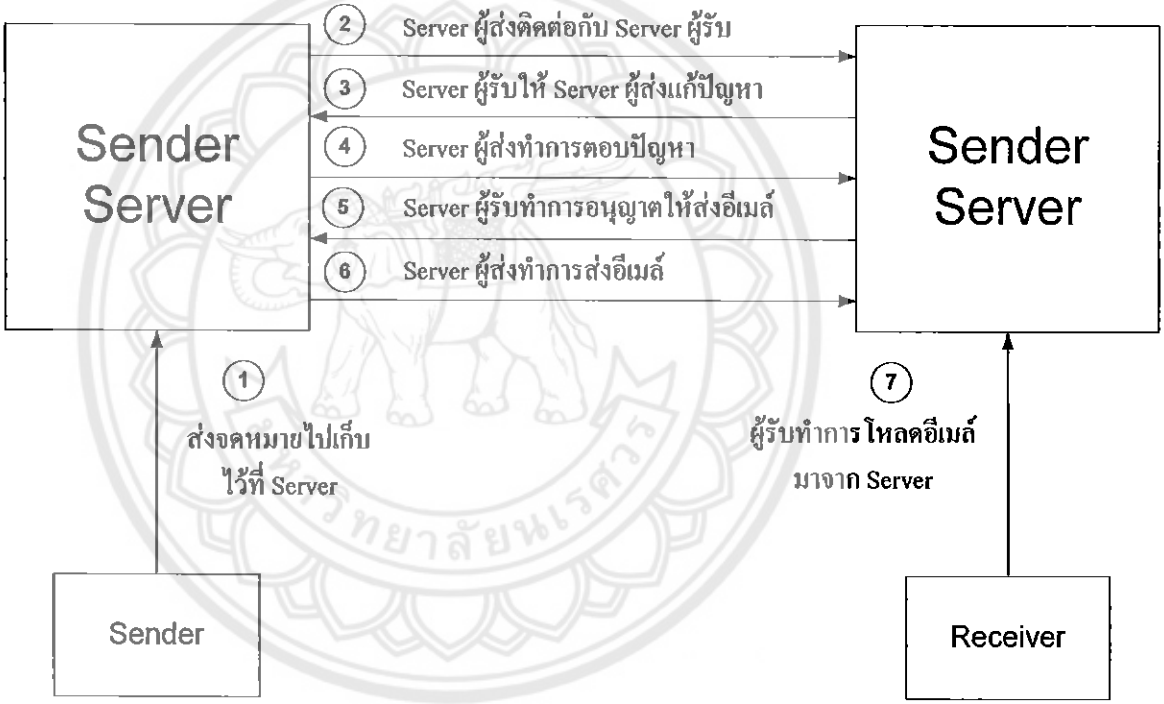
3. เสียเวลาในการขอแสตมป์ก่อนที่จะส่งข้อความ

3.4 ทำการออกแบบโปรโตคอล

ในการออกแบบโปรโตคอลเราจะนำความรู้ที่ได้จากการศึกษามาช่วยในการออกแบบซึ่งจะมีการออกแบบโปรโตคอลไว้ทั้งหมด2 แบบซึ่งมีลักษณะดังนี้

แบบที่1

รูปแบบโปรโตคอลที่ 1 มีแผนภาพการทำงานดังนี้



รูปที่ 3.1 การทำงานของโปรโตคอลรูปแบบที่ 1

1. เข้าไปใช้รูปแบบการติดต่อสื่อสารของโปรโตคอล SMTP โดยใช้ Server ของผู้รับและผู้ส่งในการติดต่อสื่อสารกันในการรับ-ส่งอีเมล์

2. ก่อนการส่งอีเมล์ทาง Server ของผู้ส่งจะต้องทำการแก้ปัญหา (Puzzle) ซึ่งใช้รูปแบบแนวคิดมาจากวิธี Challenge-Response ขั้นตอนการแก้ปัญหาคือ

- Server ของผู้รับจะให้ Server ของผู้ส่งทำการแก้ปัญหาโดยใช้โปรแกรมซึ่งจะให้ Server ผู้ส่งทำการหา พาสเวิร์ด ที่ถูกต้องโดย Server ของผู้รับจะมีการบอกใบ้พาสเวิร์ดให้บางส่วนตามความน่าเชื่อถือซึ่งจะปรับโดยตัว Server ของผู้รับซึ่งจะดูว่าผู้ส่งมีการติดต่อกันอย่างต่อเนื่องหรือไม่ หรือมีความผิดปกติในการส่งอีเมล์หรือไม่ เช่น มีการส่งอีเมล์มามากผิดปกติอย่างต่อเนื่องซึ่งก็จะทำการปรับความน่าเชื่อถือให้มีระดับต่ำลงซึ่งทาง Server ของผู้รับก็จะบอกใบ้พาสเวิร์ดในจำนวนที่น้อยลง

รูปแบบของระดับความน่าเชื่อมีดังนี้

ระดับที่ 1 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 1 ตัว

ระดับที่ 2 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 2 ตัว

ระดับที่ 3 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 3 ตัว

ระดับที่ 4 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 4 ตัว

ระดับที่ 5 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 5 ตัว

ระดับที่ 6 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 6 ตัว

ระดับที่ 7 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 7 ตัว

ระดับที่ 8 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 8 ตัว

ระดับที่ 9 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 9 ตัว

ระดับที่ 10 จะทำการเก็บอีเมล์ แอดเดรสไว้ใน Blacklist หรือ บล็อกอีเมล์นั้น

รูปแบบของ Puzzle มีลักษณะดังนี้

- จะเป็นการสุ่ม(Random) ข้อมูลที่เป็น String ทั้งหมด 256 bit
- โดยใช้โจทย์ซึ่งประกอบด้วย
  - h(m) ซึ่งก็คือ Hash ของข้อความ
  - m ก็คือข้อความบางส่วนที่บอกมาให้ (ตามระดับความน่าเชื่อถือ)

    เช่น บอกข้อความ 240 บิต ก็แสดงว่าจะต้องทำการเดาอีก 16 บิตที่เหลือ

    ซึ่ง คำตอบก็คือ m ทั้งหมด

3. เมื่อ Server ของผู้ส่งทำการหาพาสเวิร์ดที่ถูกต้องได้แล้วก็สามารถที่จะส่งอีเมล์ได้

3.5 ตรวจสอบปัญหาและจุดบกพร่องของโปรโตคอลที่ออกแบบ

ตรวจสอบดูว่าโปรโตคอลที่เราออกแบบนั้นมีจุดอ่อนหรือข้อบกพร่องคือ

1. โปรโตคอลที่ออกแบบนั้นจะต้องเข้าไปยุ่งเกี่ยวโปรโตคอล SMTP ซึ่งเป็นโปรโตคอลที่ใช้กันเป็นมาตรฐานทั่วโลกในการติดต่อรับ-ส่งอีเมล์ดังนั้นถ้าเราไปเปลี่ยนแปลงในตัวโปรโตคอลก็จะเกิดความยุ่งยากมากมายรวมทั้งต้องเสียค่าใช้จ่ายมากมายในการที่จะเปลี่ยนแปลง

2. เป็นการเพิ่มการทำงานให้กับ Server เพราะต้องทำการแก้ปัญหาในการส่งซึ่งจะทำให้การทำงานของ Server ทั้งฝ่ายรับ-ส่งทำงานหนักเกินไปซึ่งอาจจะทำให้เกิดการผิดพลาดในการรับ-ส่งอีเมล์

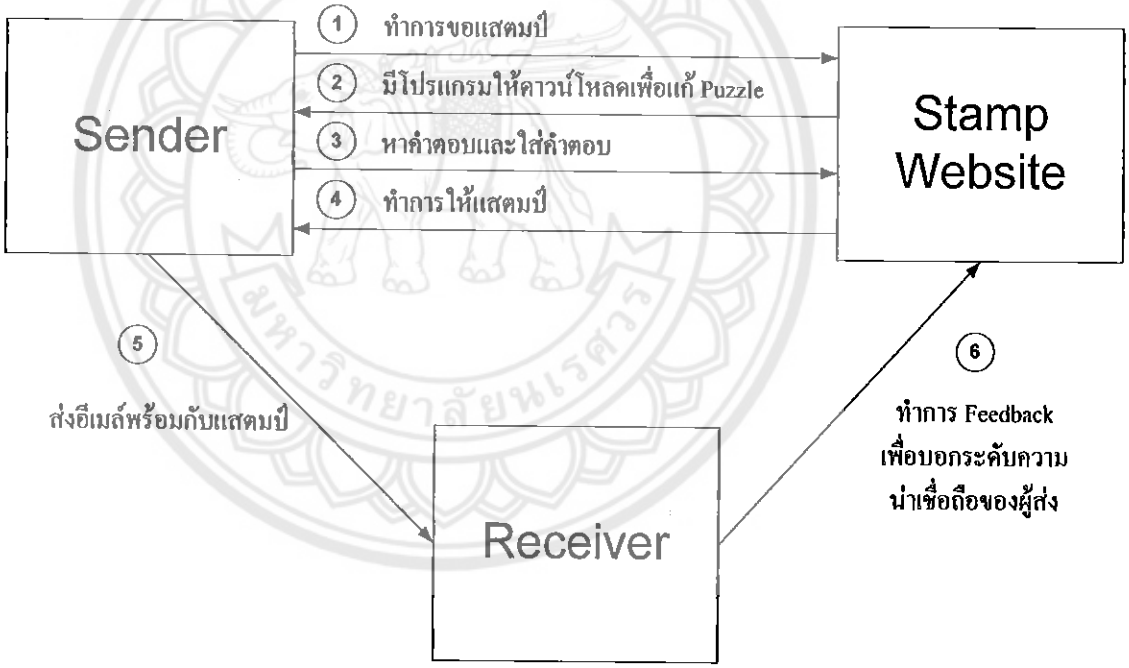
3. เนื่องจากในการแก้ปัญหานั้นจะไม่ได้ใช้บุคคลในการแก้ปัญหาแต่จะเป็นตัว Server ดังนั้นจึงไม่รู้ที่จะทำขั้นตอนการทำงานอย่างไรเพื่อที่จะใช้ในการปฏิบัติจริง

3.6 ปรับปรุงโปรโตคอลเพื่อแก้ไขจุดบกพร่อง

เนื่องจากโปรโตคอลแบบแรกที่ทำการออกแบบนั้นยังมีจุดบกพร่องอยู่หลายประการ เราจึงทำการออกแบบโปรโตคอลรูปแบบใหม่ขึ้นมา ดังนี้

แบบที่ 2

รูปแบบโปรโตคอลที่ 2 มีแผนภาพการทำงานดังนี้



รูปที่ 3.2 การทำงานของโปรโตคอลรูปแบบที่ 2

1. เข้า เว็บไซต์ที่ให้บริการแสตมป์โดยให้กรอก Username และ พาสเวิร์ดซึ่งข้อมูลเหล่านี้จะเก็บบันทึกไว้ในฐานข้อมูลเมื่อทำการกรอกเสร็จแล้วก็ทำการกด ยืนยัน

2. เมื่อทำการกด [ยืนยัน] จากข้อที่ 1 แล้วจะให้ผู้ใช้ทำการใส่ข้อมูลที่เป็นอีเมล์ แอดเดรสของผู้ส่งและของผู้รับ ซึ่งก็จะทำการเก็บข้อมูลที่กรอกไว้ลงไปในฐานข้อมูลซึ่งจะเก็บไว้ในข้อมูลเดียวกับ Username และ พาสเวิร์ด ของผู้ใช้

3. เมื่อทำการกด [ยืนยัน] จากข้อ2 แล้วก็จะให้ทำการแก้ Puzzle ซึ่งจะมีการบอกใบ้ พาสเวิร์ด ให้ตามลำดับความน่าเชื่อถือของผู้ส่งอีเมล์นั้น ซึ่งจะมีทั้งหมด 10 ระดับ ดังนี้

ระดับที่ 1 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 1 ตัว
ระดับที่ 2 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 2 ตัว
ระดับที่ 3 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 3 ตัว
ระดับที่ 4 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 4 ตัว
ระดับที่ 5 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 5 ตัว
ระดับที่ 6 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 6 ตัว
ระดับที่ 7 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 7 ตัว
ระดับที่ 8 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 8 ตัว
ระดับที่ 9 จะบอกคำตอบให้ทำการค้นหาส่วนที่เหลืออีก 9 ตัว
ระดับที่ 10 จะทำการเก็บอีเมล์ แอดเดรสไว้ใน Blacklist หรือ บล็อกอีเมล์นั้น

จะมีการบอกข้อความย่อย(Message Digest) ของ พาสเวิร์ดทั้งหมดซึ่งได้จากการนำเข้า Hash functionที่เป็นแบบ MD5 และจะมีโปรแกรมให้ทำการ ดาวน์โหลด ซึ่งจะต้องใส่ค่า พาสเวิร์ดที่บอกใบ้ให้รวมถึง Message Digest ของ พาสเวิร์ด ทั้งหมดที่มีให้ลงในตัวโปรแกรมซึ่งโปรแกรมก็จะทำการค้นหาข้อมูลจนได้พาสเวิร์ด ที่ถูกต้อง ซึ่งการทำทั้งหมดนี้เรียกว่าการแก้ "Puzzle"

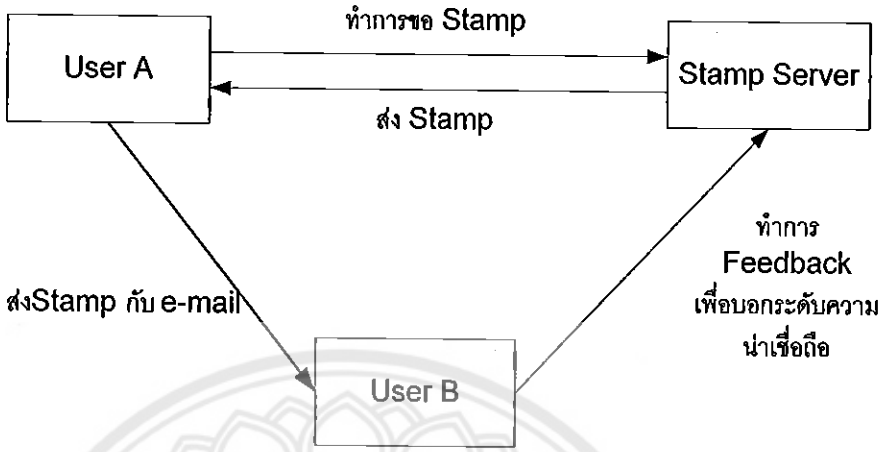
ลักษณะของโปรแกรม ก็คือจะเขียนด้วยภาษา Java ซึ่งจะเป็นโปรแกรมที่ทำการวนหาคำตอบ พาสเวิร์ด ที่ถูกต้องโดยจะต้องใช้ พาสเวิร์ด ที่ทำการบอกใบ้ให้และ Message Digest ของพาสเวิร์ด ทั้งหมดเพื่อทำการเทียบกันซึ่งก็จะทำการค้นหาจนเจอคำตอบ (พาสเวิร์ด) ทั้งหมด

รูปแบบ Puzzle

- จะเป็นการสุ่ม(Random) ข้อมูลที่เป็น String ทั้งหมด 256 bit
- โดยใช้โจทย์ซึ่งประกอบด้วย
  - h(m) ซึ่งก็คือ Hash ของข้อความ
  - m ก็คือข้อความบางส่วนที่บอกมาให้ (ตามระดับความน่าเชื่อถือ)
    เช่น บอกข้อความ 240 บิต ก็แสดงว่าจะต้องทำการเดาอีก 16 บิตที่เหลือ
    ซึ่ง คำตอบก็คือ m ทั้งหมด

4. เมื่อทำการเปรียบเทียบ พาสเวิร์ด ถูกต้องแล้วทาง เว็บไซต์ ที่ให้บริการก็จะทำการให้แสตมป์ แก่ผู้ใช้ที่เข้ามาขอโดยลักษณะของแสตมป์จะเป็นข้อมูลที่แสดงถึงผู้รับ-ส่งรวมถึงแสดงว่าแสตมป์นี้เป็นการขอครั้งที่เท่าไหร่

ซึ่งแผนภาพการดำเนินการของโปรโตคอลที่ออกแบบมาจะมีลักษณะดังนี้



รูปที่ 3.3 แผนภาพการดำเนินการของโปรโตคอลที่ออกแบบ

หมายเหตุ : โดยที่ A เป็นผู้ส่ง และ B เป็นผู้รับ

5. เมื่อดำเนินการมาถึงข้อ4 ของแบบแรกแล้วสำหรับแสตมป์นั้นโดยจะใช้หลักการของลายเซ็นดิจิตอล (Digital Signature)

6. ฝ่ายรับต้องทำการเช็คได้ว่าแสตมป์เป็นของจริงหรือไม่

6.1 ในการที่จะได้แสตมป์มานั้นต้องได้จาก Server ที่ให้แสตมป์จริง

ใช้ความรู้เกี่ยวกับการใช้ ลายเซ็นดิจิตอล เพื่อทำให้ระบุว่าได้มาจาก Server จริงๆ และระว่ามีตัวตนจริงของผู้ส่งมีลักษณะคือ

6.2 เนื้อหาของแสตมป์คือ ข้อมูล X

Stamp $= X, \{X\}_{pri\text{-}s}$ โดยที่ X เป็นข้อมูลเดิม , {X}pri-s คือข้อมูลที่ถอดรหัสโดยใช้ Private Key ของผู้ส่ง

ทางฝ่ายรับก็จะทำการตรวจสอบข้อมูลโดยใช้ Public Key ของฝ่ายส่ง เมื่อได้ข้อมูลมาก็ทำการเปรียบเทียบกับข้อมูล X จากผู้ส่ง

Check X $= \{\{X\}_{pri\text{-}s}\}_{pub\text{-}s}$; ซึ่งจะได้ค่า X ซึ่งสามารถนำมาเปรียบเทียบกับข้อมูลX ในแสตมป์จากผู้ส่งว่าตรงกันหรือไม่ซึ่งทำให้สามารถตรวจสอบได้ว่าผู้ส่งนั้นมีตัวตนจริงโดยเนื้อหาจะต้องมีการระบุว่า ผู้ส่งคือ A และผู้รับคือ B

$X = A \| B \| N$; A คือ ผู้ส่ง, B คือ ผู้รับ และ N คือจำนวนครั้งที่ขอ Stamp

6.3 ทำการตรวจสอบว่าแสตมป์นี่ต้องไม่เคยถูกใช้มาก่อน

$$Stamp = A \parallel B \parallel N, \{A \parallel B \parallel N\}_{pri\text{-}s}$$

โดยค่า N จะทำให้เราสามารถตรวจสอบได้ว่าเป็นแสตมป์ที่ใช้แล้วหรือแสตมป์เก่าหรือไม่ โดยสามารถทำการเปรียบเทียบค่า N ครั้งล่าสุดที่ส่งมา ซึ่งเราสามารถใช้โปรแกรมตรวจสอบได้

โดยใช้คำสั่ง If ($N_{old} < N_{new}$) ก็จะทำการให้ผ่านแต่ถ้าไม่ก็จะไม่รับอีเมล์จากผู้ส่งนั้น

7. เมื่ออีเมล์ส่งถึงผู้รับแล้วผู้รับสามารถที่จะมี Feedback กลับมายังผู้ให้บริการแสตมป์ได้ว่าให้ปรับระดับความน่าเชื่อถือของผู้ส่งเป็นระดับที่เท่าไหร่แต่ถ้าไม่มีการ Feedback และมีการติดต่อไปอยู่เสมอทางระบบก็จะปรับความน่าเชื่อถือของผู้ส่งให้อยู่ในระดับที่ดีขึ้น

3.7 ออกแบบลักษณะของ Puzzle

ซึ่งเป็นโปรแกรมที่ให้ดาวน์โหลด ในเว็บไซต์ เพื่อนำไปแก้ปัญหา (Puzzle) ในการทำงาน โดยแบ่งออกเป็นฝ่ายรับ และ ฝ่ายส่ง ลักษณะโปรแกรมมีดังนี้

ฝ่ายส่ง (Sender)

เป็นโปรแกรมที่ทำการวนหาคำตอบของพาสเวิร์ด ทั้งหมดโดยจะต้องใส่ค่า Message Digest ของพาสเวิร์ด ทั้งหมดด้วยเพื่อเป็นการตรวจสอบว่าถูกต้อง

Code ในส่วนที่สำคัญในการทำงาน

```
import java.math.BigInteger;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
```

เป็นคำสั่ง Import เพื่อจะได้เรียกใช้ฟังก์ชั่นต่างๆที่เกี่ยวข้อง

```
//---- input parameter ------
String ppwd = args[0];
String pwdLen = args[1];
String mdpwd = args[2];

System.out.println("patial password = "+ppwd);
System.out.println("password length = "+pwdLen);
System.out.println("md password = "+mdpwd);
```

```
int bitSpace = (Integer.parseInt(pwdLen)-ppwd.length())*4;
System.out.println("bitSpace = "+bitSpace+" bit");
// ----------------------------
```

เป็นการใส่ค่าของ Password ที่บอกให้บางส่วน(ppwd) , ความยาวของ Password ทั้งหมด (pwdLen) , ค่าของ Message Digest ของ Password ทั้งหมด (mdpwd)

โดยค่าที่ต้องหานั้นจะได้จาก (ความยาวของ Password ทั้งหมด - Password ที่บอกให้บางส่วน)*4 ; ที่คูณด้วย 4 เพราะ 1 char มี 4 บิต ในเลขฐาน16

```
// ------------------------------------
while(guess.length()<(bitSpace/4))
{
guess = "0"+guess;
}
System.out.println("guess = "+guess);
String candidate = ppwd + guess;
//-----------------------------------
```

เป็นการวนหาค่าตัวที่เหลือโดยจะเพิ่มค่าขึ้นเรื่อยโดยจะมีการแสดงว่าวนถึงตัวไหนแล้วบ้างแล้วก็นำค่านั้นไปบวกกับค่าของ Password ที่บอกให้มาบางส่วน

```
//-----------------------------------
MessageDigest algorithm = MessageDigest.getInstance("MD5");
//-----------------------------------
```

เป็นการเรียกใช้ฟังก์ชั่น MD5 เพื่อนำมาเทียบกับตัวที่ถูกต้อง

```
//-----------------------------------
if(mdpwd.equalsIgnoreCase(result))
{
System.out.println("message digest of password = "+mdpwd);
System.out.println("Finish");
System.out.println("password = "+candidate);
break;
//-----------------------------------
```

เมื่อค่าของ MD5 ของการวนหาคำตอบ กับ ค่า MD5 ของคำตอบจริงเท่ากันก็จะหยุดทำการวนหาคำตอบแล้วก็จะแสดงข้อมูลทั้งหมดแล้วแสดงคำว่า finish

```
//-------------------------------------
```

ฝ่ายผู้ให้บริการ Stamp

โดยฝ่ายผู้ให้บริการแสตมป์จะบอกpassword บางส่วนและจะให้ Message Digest ของ password ทั้งหมด

```
$num_show=getNumberPass($level);  //จำนวนของ pass word ที่จะแสดง
$pass_rand_show=substr($pass_rand,0,$num_show); //  pass word ที่จะแสดง
$generate=md5($pass_rand); // ใช้ MD5 ของ Password ทั้งหมด
//-------------------------------------------------------------------------
if($dbarr['pass_rand']==$respond)
        {
                $result="<br><br><center><font size='3' face='MS Sans Serif'><b>รหัสถูกต้อง</b></font></center>" ;
                print "<meta http-equiv=refresh content=2;URL=page_4.php>";
        }
        else
        {
                $result="<br><br><center><font size='3' face='MS Sans Serif'><b>รหัสไม่ถูกต้อง กรุณาลองใหม่อีกครั้ง</b></font></center>" ;
        }
}
//-------------------------------------------------------------------------
```

เป็นการตรวจคำตอบโดยถ้าคำตอบที่หามาเท่ากับ Password ทั้งหมดก็จะแสดงข้อความว่า "รหัสถูกต้อง" แต่ถ้าไม่ใช่ก็จะแสดงว่า "รหัสไม่ถูกต้อง"

```
//-------------------------------------------------------------------------
```

บทที่ 4

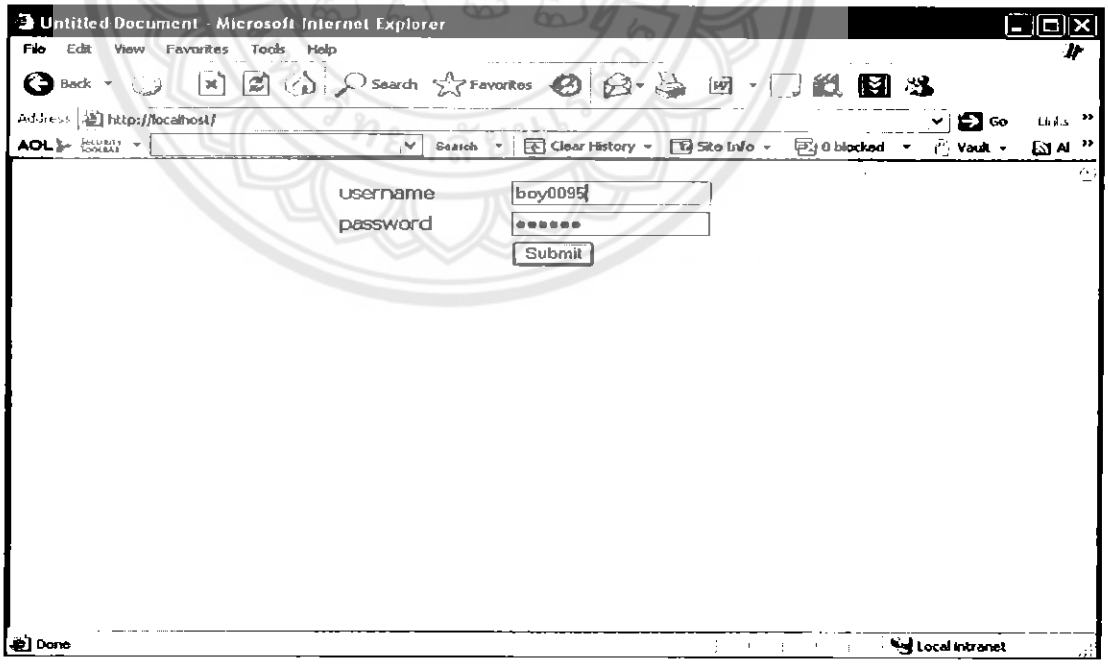
ผลการทดลอง

4.1 สรุปการเลือกใช้รูปแบบโปรโตคอล

เราได้เลือกใช้การออกแบบรูปแบบโปรโตคอลที่ 2 ซึ่งเป็นลักษณะที่ให้มีการทำงาน(มากหรือ น้อยตามระดับความน่าเชื่อถือ) ก่อนที่จะมีการให้แสตมป์เพื่อแนบไปกับข้อความก่อนการส่ง ซึ่งวิธีนี้จะสามารถช่วยลดอีเมล์ขยะรวมถึงอีเมล์ที่ส่งมาแบบไร้สาระได้เพราะก่อนการส่งจะต้องเสียค่าใช้จ่าย เช่น เปลืองค่าไฟและเสียเวลา เป็นต้น ส่วนวิธีที่ 1 นั้นจะยุ่งยากเพราะไปเกี่ยวข้องกับโปรโตคอล SMTP ซึ่งเป็นโปรโตคอลที่ใช้กันอย่างแพร่หลายซึ่งถ้าเกิดการเปลี่ยนแปลงก็จะทำให้ต้องคำนินเรื่องต่างๆมากมายรวมถึงวิธีนี้จะทำให้ Server ทำงานหนักเกินไป

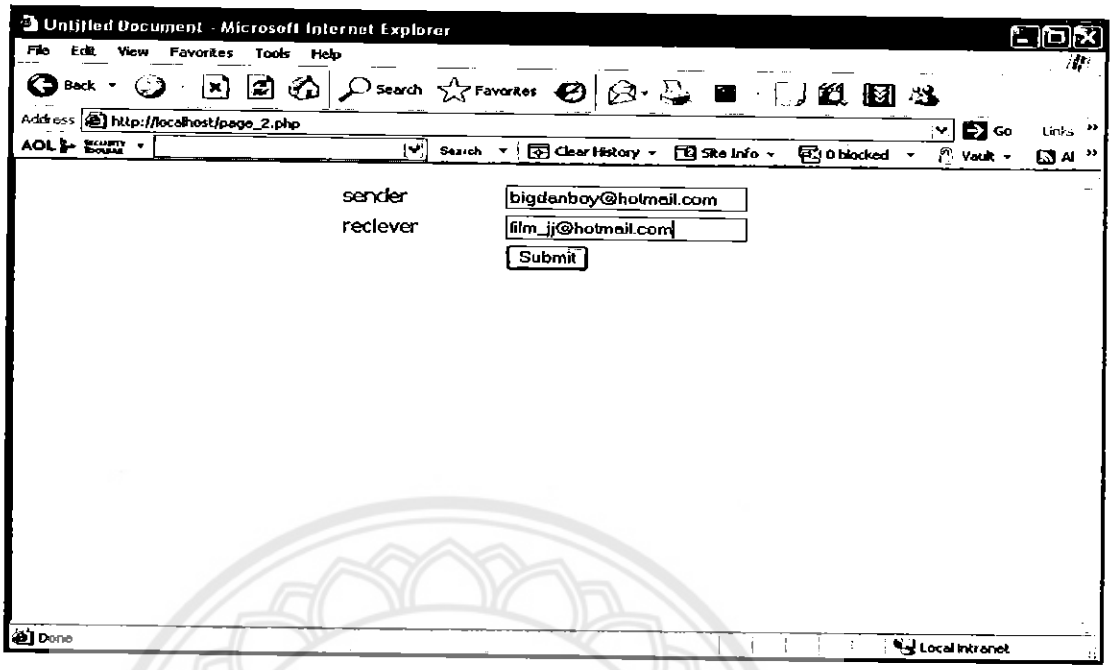
4.2 การทดสอบการทำงานของเว็บไซต์ที่ให้บริการแสตมป์

1. ทำการเข้าเว็บไซต์ที่ให้บริการแสตมป์
2. ทำการกรอก Username และ พาสเวิร์ด เพื่อเก็บไว้ในฐานข้อมูล
3. หลังจากทำการกรอกข้อมูลแล้วก็ทำการกด Submit



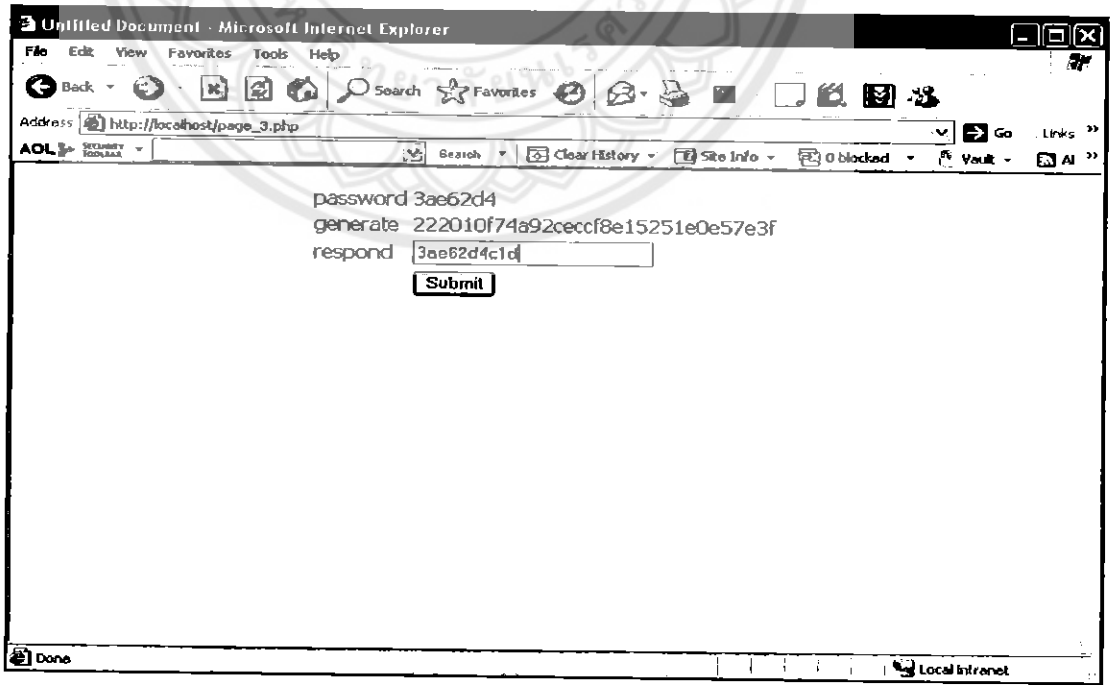
รูปที่4.1 เว็บไซต์ที่ให้บริการแสตมป์หน้า Log in

4. ทำการกรอกที่อยู่อีเมล์ (อีเมล์แอดเดรส) ของผู้ส่งและผู้รับ แล้วกด Submit



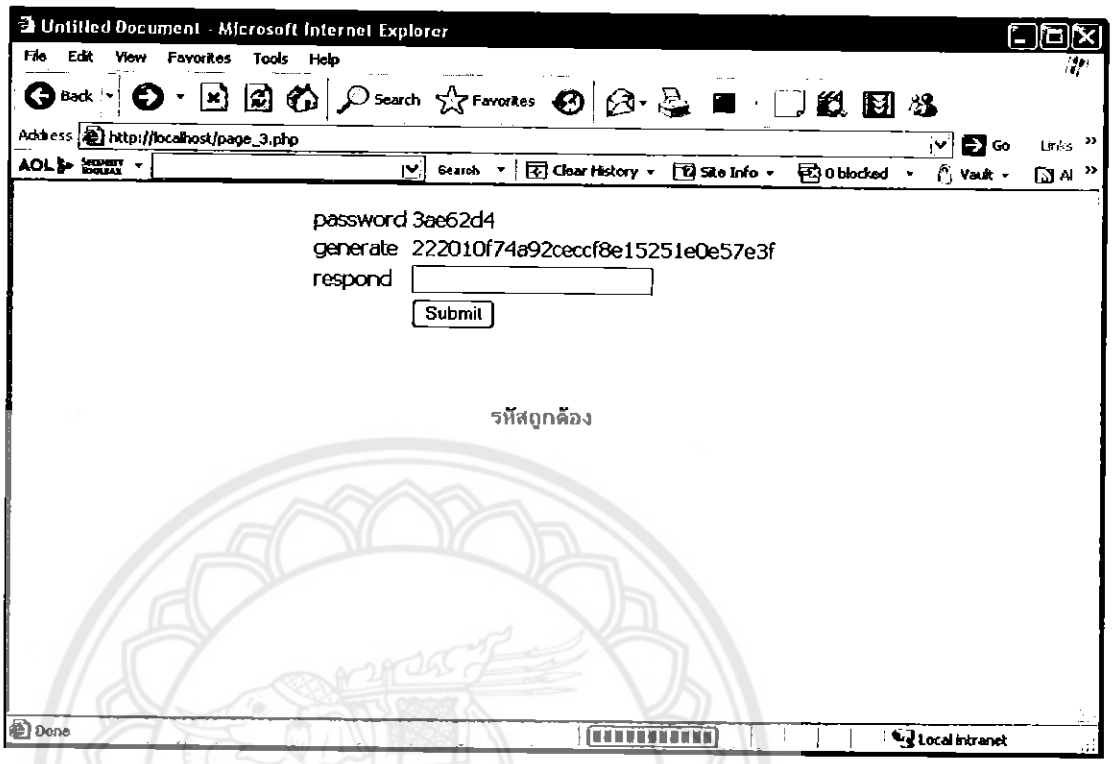
รูปที่4.2 เว็บไซต์ที่ให้บริการแสดงมป์หน้ากรอกอีเมล์ ผู้รับ-ผู้ส่ง

5. เข้าสู่หน้าที่ให้ทำการตอบคำตอบทั้งหมดโดยให้พาสเวิร์ด บางส่วนและMDของพาสเวิร์ด ทั้งหมดรวมถึงให้มีการดาวน์โหลด โปรแกรมในการค้นหาคำตอบ



รูปที่4.3 เว็บไซต์ที่ให้บริการแสดงมป์หน้าที่ให้ใส่คำตอบทั้งหมด

6. เมื่อกรอกคำตอบถูกต้องระบบก็จะแสดงว่า "รหัสถูกต้อง"



รูปที่ 4.4 เว็บไซต์ที่ให้บริการแสดงปีเมื่อตอบคำตอบถูกต้อง

ในการแก้ พาสเวิร์ด เพื่อหาคำตอบทั้งหมดนั้นจะต้องใช้โปรแกรมที่มีให้ซึ่งจะมีลักษณะการใช้งานของโปรแกรมดังนี้

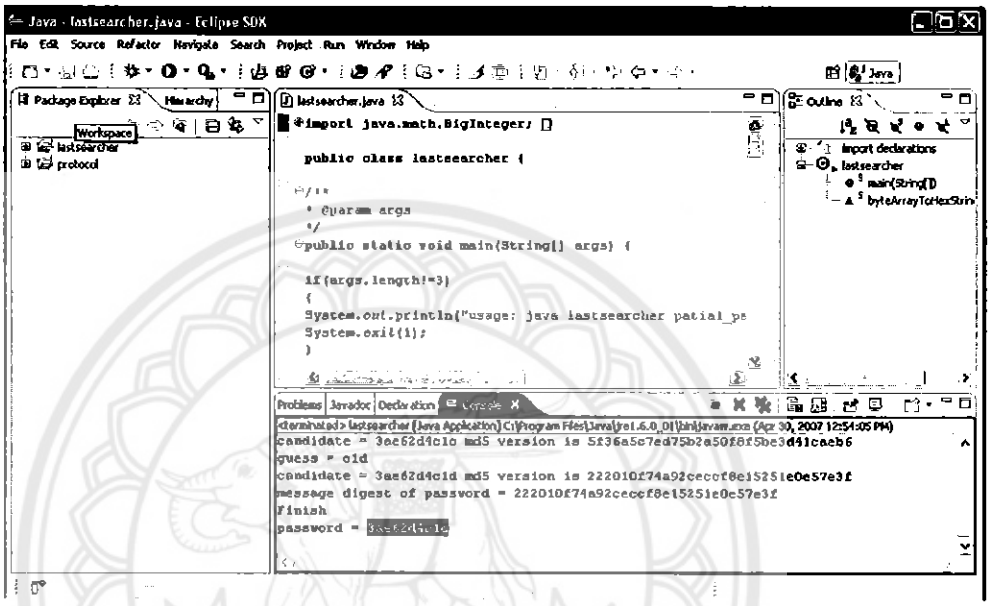
```
//---- input parameter ------
String ppwd = args[0];
String pwdLen = args[1];
String mdpwd = args[2];
System.out.println("patial password = "+ppwd);
System.out.println("password length = "+pwdLen);
System.out.println("md password = "+mdpwd);
int bitSpace = (Integer.parseInt(pwdLen)-ppwd.length())*4;
System.out.println("bitSpace = "+bitSpace+" bit");
```

เป็นการใส่ค่าของพาสเวิร์ดที่บอกให้บางส่วน (ppwd), ความยาวของพาสเวิร์ดทั้งหมด (pwdLen) , ค่าของ Message Digest ของพาสเวิร์ดทั้งหมด (mdpwd)

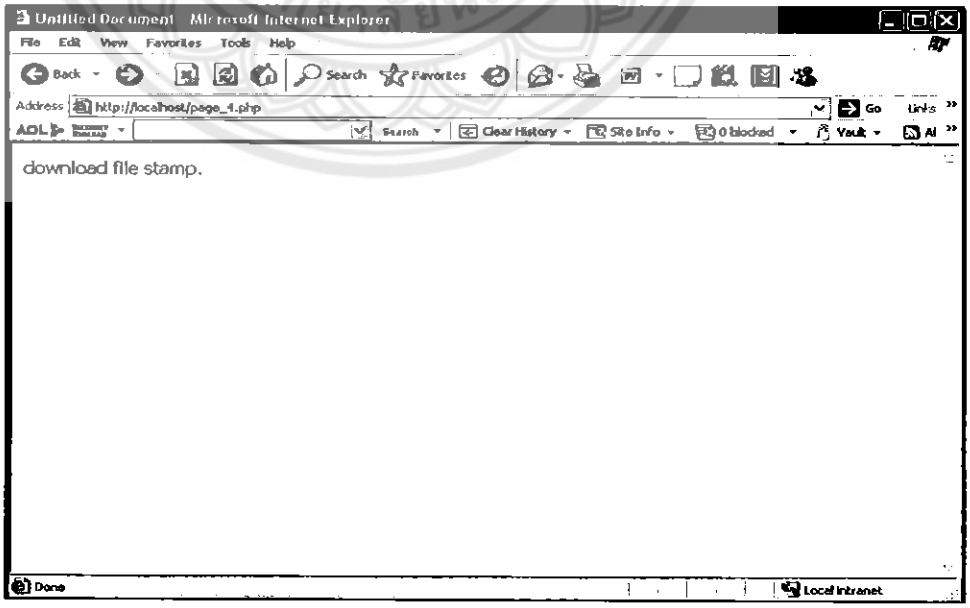
โดยค่าที่ต้องหานั้นจะได้จาก (ความยาวของพาสเวิร์ด ทั้งหมด - พาสเวิร์ดที่บอกให้บางส่วน)*4 ; ที่คูณด้วย 4 เพราะ 1 char มี 4 บิต ในเลขฐาน16

เมื่อทำการใส่ค่าทั้งหมดแล้วเมื่อทำการ Run โปรแกรมจะมีลักษณะดังนี้

- สมมติว่าผู้ใช้อยู่ในระดับความน่าเชื่อถือระดับ 7 (ไม่ค่อยน่าเชื่อถือ)



รูปที่ 4.5 รูปแบบการทำงานของโปรแกรมเพื่อหาค่าพาสเวิร์ด หรือคำตอบทั้งหมด

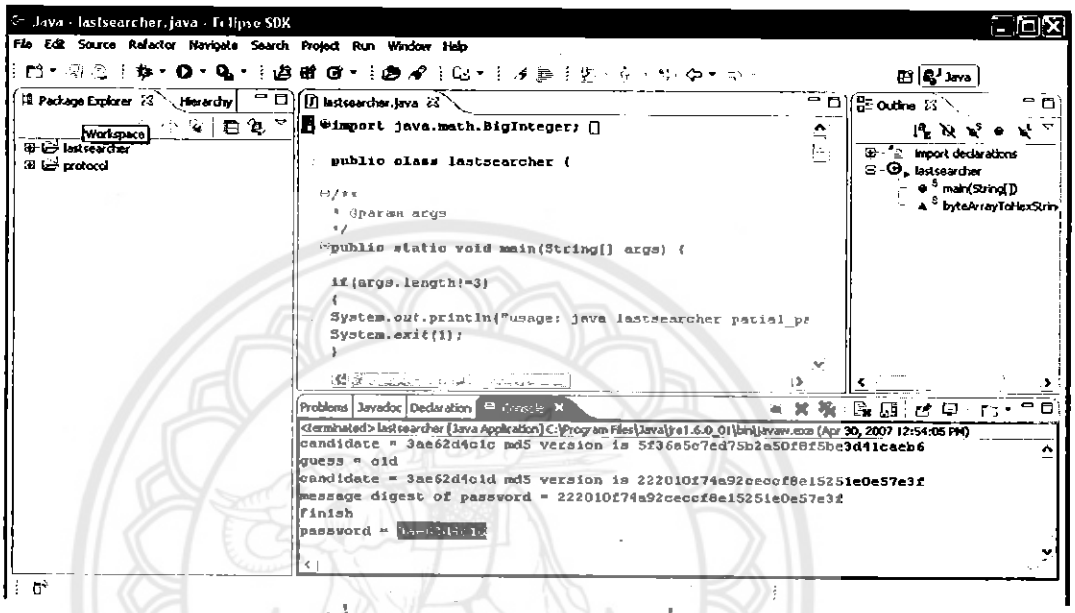


รูปที่ 4.6 เว็บไซต์ที่ให้บริการแสตมป์หน้าที่ให้ดาวน์โหลดแสตมป์

เมื่อผ่านขั้นตอนทุกอย่างก็จะมี File Stamp ให้โหลดโดยในการส่งจะต้องแนบไปกับอีเมล์ทุกครั้งซึ่ง Stamp ในแต่ละครั้งนั้นจะมีข้อมูลไม่เหมือนกัน

การทดสอบในการแก้ Puzzle เพื่อหาระดับความน่าเชื่อถือ

ตัวอย่างการทดลองการหาคำตอบโดยใช้โปรแกรมมีรูปแบบดังนี้



รูปที่ 4.7 ผลการทดลองความน่าเชื่อถือระดับ 1

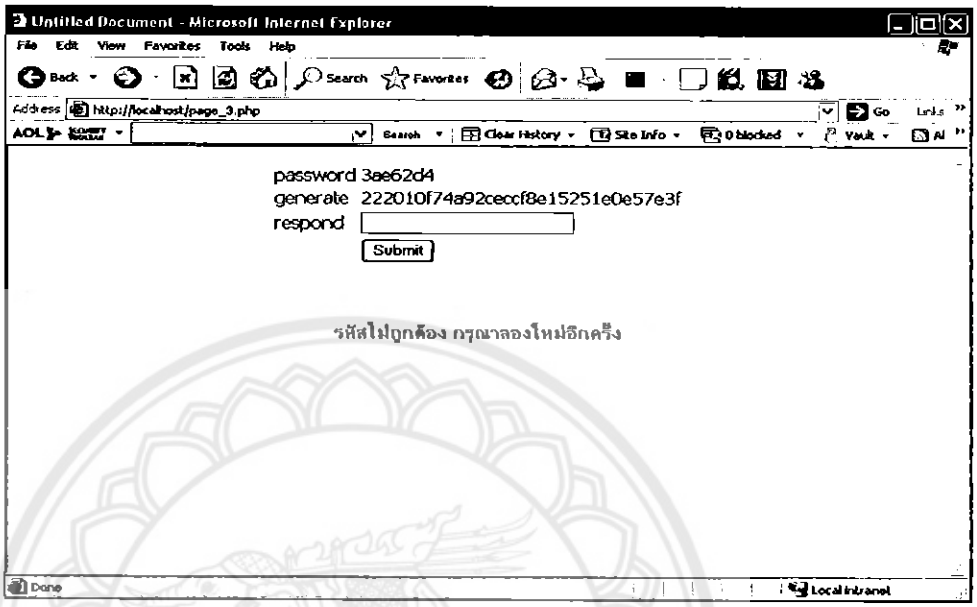
จากการทดลองได้สรุปเป็นตารางเวลาในการแก้ Puzzle แต่ละระดับความน่าเชื่อถือ

ตารางที่ 4.1 ผลการทดลองหาระยะเวลาในการแก้ Puzzle

ระดับความน่าเชื่อถือ	จำนวนบิตที่ใช้หาคำตอบ	ระยะเวลาในการหาคำตอบ
ระดับ 1	1	30 millisec
ระดับ 2	2	230 millisec
ระดับ 3	3	641 millisec
ระดับ 4	4	3094 millisec
ระดับ 5	5	36783 millisec
ระดับ 6	6	455145 millisec
ระดับ 7	7	13137782 millisec
ระดับ 8	8	ประมาณ 3 วัน12 ชม.
ระดับ 9	9	ประมาณ 7 วัน
ระดับ 10	10	ทำการ Block อีเมล์

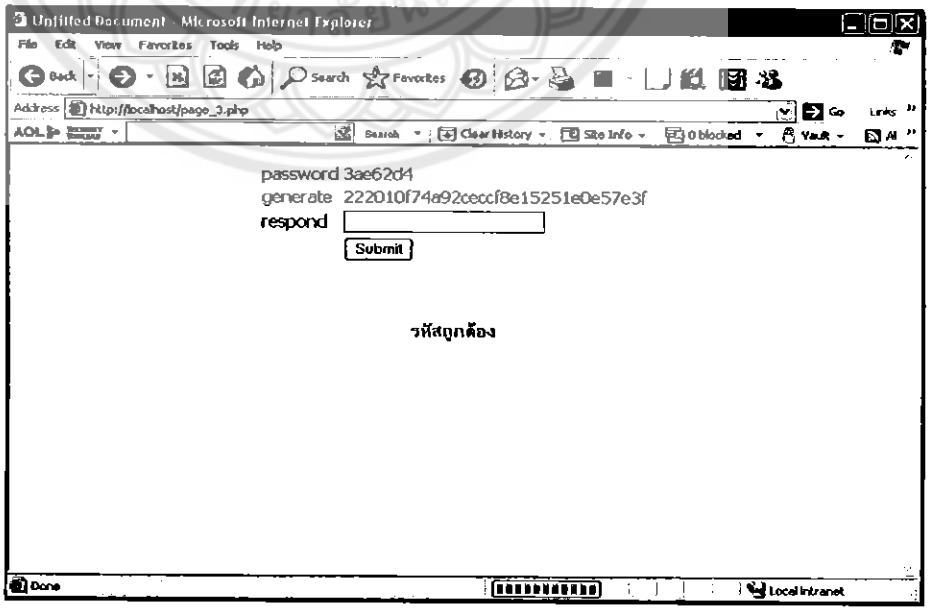
ผลการทดสอบเมื่อทำการใส่คำตอบ

เมื่อใส่คำตอบที่ไม่ถูกต้องทางเว็บไซต์ที่บริการจะขึ้นว่า "รหัสไม่ถูกต้อง" ซึ่งจะไม่เข้าไปที่หน้าของการดาวโหลดแสตมป์

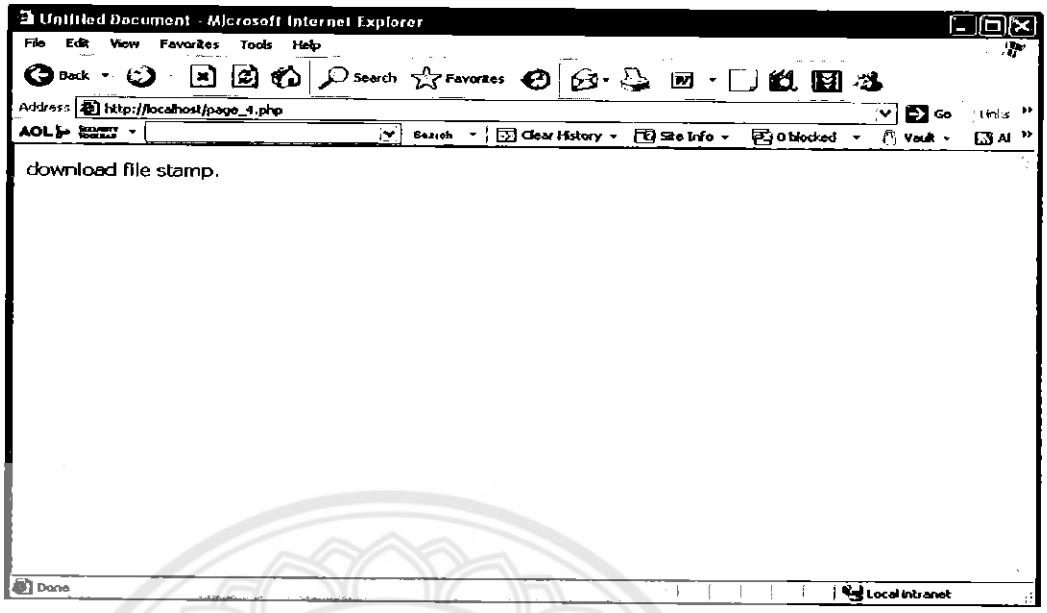


รูปที่ 4.8 เว็บไซต์ที่ให้บริการแสตมป์เมื่อใส่คำตอบไม่ถูกต้อง

เมื่อใส่คำตอบที่ถูกต้องทางเว็บไซต์ที่บริการจะขึ้นว่า "รหัสถูกต้อง" และจะเข้าไปที่หน้าของการดาวโหลดแสตมป์



รูปที่ 4.9 เว็บไซต์ที่ให้บริการแสตมป์เมื่อใส่คำตอบที่ถูกต้อง



รูปที่ 4.10 เว็บไซต์ที่ให้บริการแสตมป์เมื่อเข้าไปทำการ ดาวน์โหลด แสตมป์

โดยลักษณะของแสตมป์จะอยู่ในรูปของไฟล์ซึ่งเวลาส่งอีเมล์นั้นจะต้องทำการแนบไฟล์ที่ได้จากเว็บไซต์ผู้ให้บริการไปด้วยในการส่งซึ่งไฟล์แสตมป์นี้จะใช้ความรู้ทางด้าน Digital Signature เพื่อยืนยันว่าเป็นแสตมป์ที่มาจากผู้ให้บริการเว็บไซต์จริง

4.3 การทดสอบประสิทธิภาพของโปรโตคอล

กรณีที่1 เมื่อผู้ส่งทำการส่งอีเมล์ครั้งแรกมาให้ผู้รับ

เมื่อผู้ส่งทำการส่งอีเมล์ครั้งแรกมาให้ผู้รับ เว็บไซต์ที่ให้บริการแสตมป์จะปรับระดับความน่าเชื่อถือของผู้ส่งให้อยู่ในระดับปานกลาง คือ ระดับที่ 5

กรณีที่2 ผู้รับทำการ Feedback กลับไปที่ผู้ให้บริการแสตมป์

เมื่อผู้รับได้รับอีเมล์จากผู้ส่งแล้ว จะต้องทำการ Feedback กลับไปที่เว็บไซต์ที่ให้บริการแสตมป์ว่าผู้ส่งอยู่ในระดับความน่าเชื่อถือใด และเว็บไซต์ที่ให้บริการแสตมป์จะทำการปรับระดับตามที่ผู้รับ Feedback มา

กรณีที่3 เมื่อผู้รับไม่มีการแสดง Feedback กลับมายังผู้ให้บริการแสตมป์

ผู้รับจะต้องทำการ Feedback กลับไปให้เว็บไซต์ที่ให้บริการแสตมป์ เมื่อผู้รับไม่มีการ Feedback กลับมา ทางเว็บไซต์ก็จะให้ระดับความน่าเชื่อถือของผู้ส่งอยู่ในระดับเดียวกับการขอครั้งแรก แต่ถ้าผู้ส่งมีการขอแสตมป์อยู่อย่างต่อเนื่องก็จะทำการปรับระดับความน่าเชื่อถือให้มีระดับความน่าเชื่อถือมากขึ้น

<u>กรณีที่4</u> เมื่อผู้ส่งใช้แสตมป์ที่ไม่ได้ขอจาก Server ส่งมา

ทางผู้รับจะใช้ Public Keys ทำการถอดรหัสข้อมูลจาก private keys ของผู้ส่งซึ่งเมื่อทำการถอดรหัสแล้วจะได้ข้อมูลออกมาซึ่งลักษณะจะเป็นดังนี้

Stamp $= X, \{X\}_{pri\text{-}s}$ โดยที่ X เป็นข้อมูลเดิม , {X}pri-s คือข้อมูลที่ถอดรหัสโดยใช้ Private Key ของผู้ส่ง

ทางฝ่ายรับก็จะทำการตรวจสอบข้อมูลโดยใช้ Public Key ของฝ่ายส่ง เมื่อได้ข้อมูลมาก็ทำการเปรียบเทียบกับข้อมูล X จากผู้ส่ง

Check X $= \{\{X\}_{pri\text{-}s}\}_{pub\text{-}s}$; ซึ่งจะได้ค่า X ซึ่งสามารถนำมาเปรียบเทียบกับข้อมูลX ในแสตมป์จากผู้ส่งว่าตรงกันหรือไม่ซึ่งทำให้สามารถตรวจสอบได้ว่าผู้ส่งนั้นมีตัวตนจริงโดยเนื้อหาจะต้องมีการระบุว่า ผู้ส่งคือ A และผู้รับคือ B

$X = A \| B \| N$; A คือ ผู้ส่ง, B คือ ผู้รับ และ N คือจำนวนครั้งที่ขอ Stamp

<u>กรณีที่5</u> เมื่อผู้ส่งใช้แสตมป์ที่เคยใช้แล้วมาแนบไฟล์ส่ง

ทางผู้รับจะทำการปฏิเสธการรับอีเมล์นี้และสามารถที่จะแสดง Feedback กลับไปยังผู้ให้บริการแสตมป์เพื่อปรับระดับความน่าเชื่อถือให้ลดลง ซึ่งจะมีวิธีการตรวจสอบโดยการเช็คจากจำนวนครั้งของการขอแสตมป์ซึ่งจะมีข้อมูลนี้อยู่ในแสตมป์ คือ ข้อมูลในแสตมป์ที่เราถอดรหัสมาที่ได้จากใน กรณีที่4 ข้อมูลจะอยู่ในรูปแบบ A|| B || N โดย A คือ ผู้ส่ง B คือผู้รับและN คือ จำนวนครั้งที่ทำการขอแสตมป์ซึ่งผู้รับจะต้องทำการตรวจสอบว่าค่า N นั้นเคยถูกใช้มาก่อนหรือไม่ซึ่งผู้รับจะต้องบันทึกว่ารับแสตมป์ครั้งที่เท่าไหร่มาบ้างซึ่งถ้าเป็นของเก่าเราก็จะไม่รับอีเมล์นั้น

<u>กรณีที่6</u> เมื่อผู้ส่งใช้แสตมป์เดียวกันส่งไปหาผู้รับอื่น

ทางผู้รับตรวจสอบได้จากเนื้อหาของแสตมป์ซึ่งจะระบุผู้รับและผู้ส่งรวมถึงจำนวนครั้งที่ทำการขอแสตมป์ซึ่งจะใช้ Public keys ของผู้ส่งในการถอดข้อมูลโดยข้อมูลในแสตมป์ที่เราถอดรหัสมาเราจะได้อยู่ในรูปแบบ A|| B || N ซึ่งเราสามารถตรวจสอบได้จาก A คือที่อยู่ของผู้ส่ง ส่วน B คือที่อยู่ของผู้รับ ซึ่งเราก็สามารถที่จะดูข้อมูลภายในได้ว่าแสตมป์นี้ทำการส่งหาใคร

บทที่ 5
สรุปผลโครงงาน

5.1 สรุปผลโครงงาน

การจัดทำโครงงานนี้จะเริ่มต้นจากการศึกษาการทำงานของเทคนิคและวิธีการต่างๆที่ใช้ในการป้องกันอีเมล์ขยะ หลังจากนั้นจึงได้ทำการออกแบบโปรโตคอล โดยโปรโตคอลดังกล่าวได้อาศัยข้อดี-ข้อเสีย ของวิธีการที่ใช้ในการป้องกันอีเมล์ขยะมาเป็นหลักเกณฑ์ในการออกแบบ ซึ่งส่วนหนึ่งได้นำหลักการทำงานของวิธี Cost Based Spam Control มาใช้ โดยการทำให้ผู้ที่จะทำการส่งอีเมล์ต้องทำงานบางอย่างก่อนการส่งอีเมล์ เพื่อต้องการให้ผู้ส่งอีเมล์เสียเวลาในการทำงานซึ่งงานดังกล่าวของโปรโตคอลใหม่นี้คือ การหาคำตอบของ Puzzle รวมถึงได้นำวิธีการ Postage Protocol มาใช้เป็นหลักเกณฑ์ในการออกแบบและสร้างเว็บไซต์ที่ให้บริการแสตมป์ ซึ่งมีหลักการคือ ผู้ที่จะทำการส่งอีเมล์จะต้องทำการขอแสตมป์จากเว็บไซต์ที่ให้บริการแสตมป์ก่อน โดยเว็บไซต์ที่ให้บริการแสตมป์จะให้ผู้ที่จะส่งอีเมล์ทำการแก้ปัญหาซึ่งลักษณะคือจะมีพาสเวิร์ดมาให้บางส่วนตามระดับความน่าเชื่อถือของผู้ส่งคือถ้าผู้ส่งอยู่ในระดับที่ดีก็จะทำการสุ่มหาพาสเวิร์ดตัวที่เหลือน้อยลงและทางเว็บไซต์จะให้ดาวโหลดโปรแกรมที่ทำการหาคำตอบทั้งหมดโดยโปรโตคอลนี้สามารถที่นำมาใช้ได้เลยเพราะไม่ได้เข้าไปยุ่งกลับโปรโตคอลที่ใช้เป็นมาตรฐานในการรับ-ส่งอีเมล์ เช่น SMTP โปรโตคอล

ผลที่ได้จากการทดสอบโปรโตคอลที่ได้ทำการออกแบบเราสามารถสรุปในแต่ละกรณีต่างๆได้ดังนี้

1. เมื่อผู้ส่งทำการส่งอีเมล์ครั้งแรกมาให้ผู้รับผู้ให้บริการแสตมป์จะปรับระดับความน่าเชื่อถืออยู่ที่ระดับ 5
2. เมื่อผู้รับทำการ Feedback กลับไปที่ผู้ให้บริการแสตมป์โดยทำเพื่อที่จะปรับระดับความน่าเชื่อถือของผู้ส่ง
3. เมื่อผู้รับไม่มีการแสดง Feedback กลับมายังผู้ให้บริการแสตมป์ทางผู้ให้บริการก็จะคงให้ระดับความน่าเชื่อถือของผู้ส่งคงเดิมแต่ถ้ามีการติดต่ออยู่เป็นประจำก็จะปรับระดับให้ดีขึ้น
4. เมื่อผู้ส่งใช้แสตมป์ที่ไม่ได้ขอจาก Server ส่งมาทางผู้รับจะใช้ Public Keys ทำการถอดรหัสข้อมูลจาก private keys ของผู้ส่งซึ่งเมื่อทำการถอดรหัสแล้วจะได้ข้อมูลออกมา
5. เมื่อผู้ส่งใช้แสตมป์ที่เคยใช้แล้วมาแนบไฟล์ส่งโดยจะทำการตรวจสอบข้อมูลที่ได้จากการถอดรหัสว่ามีจำนวนครั้งของแสตมป์ครั้งนั้นเคยส่งมาแล้วหรือไม่

6. เมื่อผู้ส่งใช้แสตมป์เดียวกันส่งไปหาผู้รับอื่นโดยดูได้จากข้อมูลที่ได้จากการถอดรหัสว่าส่งจากใครถึงใคร

ตารางที่ 5.1 สรุปตารางเปรียบเทียบความแตกต่างโปรโตคอลที่ออกแบบกับโปรโตคอลที่มีอยู่ในปัจจุบัน

หัวข้อเรื่อง	โปรโตคอล			
	ออกแบบใหม่	Content filtering	Identity-Based	Cost-Based
1. ผู้ส่งต้องมีการทำงานก่อนการที่จะส่งอีเมล์	✓			✓
2. ผู้ส่งมีระดับการทำงานไม่เท่ากันในแต่ละบุคคล	✓			
3. ผู้รับสามารถปรับระดับการทำงานของผู้ส่ง	✓			
4. มีการเปลี่ยนแปลงโปรโตคอลที่ใช้ในการรับ-ส่งอีเมล์ในปัจจุบัน	✓	✓	✓	✓
5. สามารถนำมาใช้งานได้ทันที	✓	✓	✓	✓
6. เข้าไปดูข้อความในตัวจดหมาย		✓		
7. ผู้รับต้องมี Public Keys เพื่อใช้ในการยืนยันตัวตนก่อนการส่งอีเมล์			✓	

5.2 ปัญหาและอุปสรรค

เนื่องจากเอกสารและบทความที่เกี่ยวข้องกับการรูปแบบและการทำงานของโปรโตคอลแบบต่างๆนั้นมีภาษาไทยค่อนข้างน้อยจึงทำให้ต้องทำการศึกษาจากบทความที่เป็นภาษาอังกฤษส่งผลให้มีปัญหาในการอ่านพอสมควรเนื่องจากเป็นเรื่องที่เข้าใจค่อนข้างยากจึงจำเป็นที่จะต้องใช้เวลามากในการศึกษาให้เข้าใจรวมถึงการออกแบบโปรโตคอลต้องใช้เวลานานในการคิดเนื่องมาจากขาดการวางแผนก่อนการทำให้ดีทำให้ต้องมีการแก้ไขอยู่เป็นประจำรวมถึงศึกษา

รูปแบบโปรโตคอลและการโจมตีต่างๆต้องใช้ความรู้พื้นฐานทางด้านความปลอดภัยในระบบคอมพิวเตอร์จึงทำให้มีข้อบกพร่องต่างๆในการออกแบบในช่วงแรก

5.3 แนวทางการแก้ไขปัญหา

ปัญหาที่เกิดขึ้นนั้นเกิดจากความขาดทักษะทางด้านการอ่านภาษาอังกฤษทำให้ในการอ่านข้อมูลที่เป็นภาษาอังกฤษซึ่งเป็นเนื้อหาส่วนใหญ่ที่ต้องใช้นั้นใช้เวลานานและใช้เวลาในการทำความเข้าใจมากซึ่งควรที่จะทำการอ่านภาษาอังกฤษและจับใจความเป็นประจำ รวมถึงในการออกแบบโปรโตคอลนั้นเราควรที่จะมีกรวางแผนในแต่ละขั้นตอนให้ดีและศึกษาถึงปัญหาต่างๆเรื่องเมล์ขยะที่มีในปัจจุบันให้เข้าใจเพื่อที่จะได้นำสิ่งที่เราศึกษามาช่วยในการออกแบบโปรโตคอลซึ่งจะช่วยให้สามารถลดหรือป้องกันเมล์ขยะพวกนี้ได้จริง

5.4 ข้อเสนอแนะ

ในโครงงานนี้เป็นการออกแบบรูปแบบโปรโตคอลเพื่อใช้ในการป้องกันเมล์ขยะ ซึ่งผู้จัดทำคาดหวังว่าจะสามารถเป็นแนวทางการศึกษา เพื่อนำไปพัฒนาต่อได้ดั้งนี้

- นำโปรโตคอลที่ออกแบบมาทำงานจริงโดยทำเว็บไซต์ที่ให้บริการแสตมป์
- นำโปรโตคอลนี้ไปเผยแพร่เพื่อให้ได้ใช้กันอย่างแพร่หลายในการส่งอีเมล์
- แก้ปัญหาการโจมตีที่คอยดักจับข้อมูลและเปลี่ยนแปลงโดยใช้เทคนิคที่มีการคิดค้นใหม่มาประยุกต์ใช้เข้าด้วยกันเช่น ใช้เทคนิค ความพัวพันเชิงควอนตัม (Quantum Entanglement)ลักษณะการทำงานคือเมื่อมีคนดักฟังหรือจับข้อมูลก็จะไปทำการรบกวนพฤติกรรมของอนุภาคโฟตรอนที่เป็นสื่อนำข้อมูลในระบบทำให้อนุภาคตัวอื่นๆถูกกระทบจึงทำให้ผู้ส่งและรับข้อมูลรู้ได้ว่ามีคนดักจับข้อมูลอยู่
- นำรูปแบบไปพัฒนาเพื่อให้สามารถขอแสตมป์เป็นกลุ่มได้เพื่อจะได้มีความสะดวกของผู้ส่งข้อความที่เป็นประโยชน์ จะสามารถที่จะส่งข้อความจำนวนมากได้ซึ่งลักษณะของแสตมป์นั้นจะมีลักษณะข้อมูลเฉพาะที่แสดงว่าเป็นแสตมป์เพื่อส่งข้อความที่เป็นกลุ่ม

เอกสารอ้างอิง

1. ระบบเมล์ (Mail System [homepage on the Internet]. Bangkok: ITWizard; date unknown [date unknown; cited 2006 Nov 15]. Available from: http://www.itwizard.info/technology/general/Mail%20System.htm

2. Spam-mail [homepage on the Internet]. Bangkok: นิตยสารสาระน่ารู้ประจำสัปดาห์ (ฉบับที่79): 23-29 ก.ค. 2544 [date unknown; cited 2006 Dec 20]. Available from: http://www.ku.ac.th/magazine_online/spam_mail.html

3. อีเมล์ขยะและChain Mailคืออะไร[homepage on the Internet]. Bangkok: ห้องสมุด ELIB; date unknown [date unknown; cited 2006 Nov 22]. Available from: http://www.elib-online.com/computers/spam.html

4. Spam Mail (Junk Mail) [homepage on the Internet]. Bangkok: สถาบันพัฒนาและฝึกอบรมแรงงานต้นแบบ; date unknown [date unknown; cited 2006 Nov 7]. Available from: http://www.pdti.kmutt.ac.th/learn/spam/

5. POP3 กับ IMAPคืออะไร [homepage on the Internet]. Bangkok: คอม-ไทย ดอทเน็ต; date unknown [date unknown; cited 2006 Dec 9]. Available from: www.dld.go.th/ict/article/general/gen08.html

6. ความปลอดภัยของสารสนเทศ บทที่4 [homepage on the Internet]. Petchaboon : โปรแกรมวิชาวิทยาการคอมพิวเตอร์ สถาบันราชภัฏเพชรบูรณ์; date unknown [date unknown; cited 2006 Dec 9]. Available from: http://202.29.34.95/wbi/InformationSecurity/4/charpter4.12.html

7. การเข้ารหัสข้อมูล [homepage on the Internet]. Bangkok:สำนักบริการคอมพิวเตอร์ มหาวิทยาลัยเกษตรศาสตร์; date unknown [date unknown; cited 2006 Dec 23]. Available from: http://www.ku.ac.th/e-magazine/august44/it/encryp.html

8. เทคโนโลยีด้านความปลอดภัย [homepage on the Internet]. Bangkok: เว็บไซต์ที่เน้นการพิสูจน์; date unknown [revised 2006 Nov 10; cited 2006 Nov 17]. Available from: http://www.khajorn.com/Contents/Wireless/Wireless/5.htm

9.Amir Herzberg.Cryptographic protocols to prevent spam. 2005 Sep 21. http://Amirherzberg.com.

ภาคผนวก

โปรแกรมส่วนของผู้ให้บริการแสตมป์

Code หน้าของ Config เพื่อใช้ในการติดต่อฐานข้อมูลและเพื่อให้หน้าต่างเรียกใช้

```
<meta http-equiv="content-type" content="text/html;charset=tis-620">
<?php
$host = "localhost" ;
$username = "boy" ; // ชื่อในการติดต่อ mysql
$password = "251039" ; // password ของคุณในการเชื่อมต่อกับฐานข้อมูล
$db = "boy" ;  // ชื่อฐานข้อมูลของคุณ กรุณาระบุให้ครบถ้วนนะครับ
$number_pass=10; //จำนวนของ password ที่ทำการ gen ออกมา
$connect = mysql_connect($host,$username,$password) ;
mysql_select_db($db) ;
mysql_query("SET CHARACTER SET tis620");
mysql_query("SET collation_connection = tis620_thai_ci");
// ฟังก์ชั่นหาจำนวนของ password ที่จะแสดงให้ user เห็น
function getNumberPass($level)
{
        //$number = จำนวนที่จะแสดง
        if($level==1){$number=9;}
        else if($level==2){$number=8;}
        else if($level==3){$number=7;}
        else if($level==4){$number=6;}
        else if($level==5){$number=5;}
        else if($level==6){$number=4;}
        else if($level==7){$number=3;}
        else if($level==8){$number=2;}
        else if($level==9){$number=1;}
        else if($level==10){$number=0;}
        return $number;
}
?>
```

Code หน้าที่1 ของเว็บไซต์ผู้ให้บริการแสตมป์

```
<?php
include("config.php") ;
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-874" />
<title>Untitled Document</title>
</head>
<body>
<form id="form1" name="form1" method="post" action="page_2.php">
 <table width="300" border="0" align="center">
  <tr>
   <td width="121">username</td>
   <td width="169"><input name="username" type="text" id="username" /></td>
  </tr>
  <tr>
   <td>password</td>
   <td><input name="password" type="password" id="password" /></td>
  </tr>
  <tr>
   <td> </td>
   <td><label>
    <input type="submit" name="Submit" value="Submit" />
   </label></td>
  </tr>
 </table>
</form>
</body>
</html>
```

Code หน้าที่ 2 ของผู้ให้บริการแสตมป์

```
<?php
include("config.php") ;
$username=$_POST['username'];
$password=$_POST['password'];
/*
//----------------------------------------ส่วนของการ login เข้าฐานข้อมูล ----------------------------
-
        $sql_login="select username,password from boy where username='$username' and
password='$password'";
        //echo $sql_login;
        $result_login = mysql_query($sql_login) ;
        $num_login = mysql_num_rows($result_login) ;
        if($num_login<=0) {
                echo "<br><br><center><font size='3' face='MS Sans Serif'><b>รหัสผ่าน
ไม่ถูกต้องครับ</b></font></center>" ;
                print "<meta http-equiv=refresh content=2;URL=index.php>";
                exit() ;
        }
  else {
        echo "<meta http-equiv='refresh' content='2 ;url=page_2.php'>" ;
        $_REQUEST['username']=$username;
        exit() ;
*/
        $sql_add="insert into boy(username,password) values('$username','$password')";
        $result_add=mysql_query($sql_add);
                //$_SESSION['username']=$username;
?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-874" />
<title>Untitled Document</title>
</head>

<body>
<form id="form1" name="form1" method="post" action="page_3.php">
 <table width="300" border="0" align="center">
  <tr>
   <td width="121">sender</td>
   <td width="169"><input name="sender" type="text" id="sender" size="25" /></td>
  </tr>
  <tr>
   <td>reciever</td>
   <td><input name="reciever" type="text" id="reciever" size="25" />
        <input name="username" type="hidden" value="<?php echo $username; ?>" />
        </td>
  </tr>
  <tr>
   <td> </td>
   <td><label>
    <input type="submit" name="Submit" value="Submit" />
   </label></td>
  </tr>
 </table>
</form>
</body>
</html>
```

Code หน้าที่ 3 ของผู้ให้บริการแสตมป์

```
<?php
include("config.php") ;
// ----------------------รับค่า----------------------------------------
$sender=$_POST['sender'];
$reciever=$_POST['reciever'];
$username=$_POST['username'];
$Submit3=$_POST['Submit3'];
$respond=$_POST['respond'];
$generate=$_POST['generate'];
$pass_rand_show=$_POST['pass_rand_show'];
//echo $sender.$reciever.$Submit;
if($Submit3=="") // เมื่เปิดหน้านี้มาครั้งแรก
{
//----------------------------------------ส่วนของการเพิ่ม sender recieverเข้าฐานข้อมูล ----------
----------
        $sql_add[1]="update boy set sender = '$sender' where username = '$username'";
        $sql_add[2]="update boy set reciever = '$reciever' where username = '$username'";
        for($i=1;$i<=2;$i++)
        {
                $result_add = mysql_query($sql_add[$i]);
        }
//-------------------------------------------------------------------------------
        //-----------------------------------generate password ---------------------------------
        $sql_level="select level from boy where username='$username'";
        //echo $sql_level."<br>";
        $result_level=mysql_query($sql_level);
        $dbarr_level=mysql_fetch_array($result_level);
        $level=$dbarr_level['level'];
        $num_show=getNumberPass($level); //จำนวนของ pass word ที่จะแสดง
        //echo $num_show."<br>";
        for($i=1;$i<=$number_pass;$i++)
```

```
        {
                $rand=dechex(rand(0,16)); // แปลงตัวเลขฐาน 10 เป็น 16
                $pass_rand =$pass_rand.$rand;
        }
                //echo $pass_rand."<br>";
        $pass_rand_show=substr($pass_rand,0,$num_show);//  pass word ที่จะแสดง
        //echo $pass_rand_show."<br>";
        // -------------------------บันทึกค่าของ passwod ที่ random มา----------------------
        $result_add=mysql_query("update boy set pass_rand = '$pass_rand' where username
= '$username'");
        $_SESSION['pass_rand']=$pass_rand; // บันทึกค่า pass_rand ใน session
        $generate=md5($pass_rand);
}
else
{
        $get_pass_rand = mysql_query("select pass_rand from boy where
username='$username'");
        $dbarr=mysql_fetch_array($get_pass_rand);
        //echo $sql_login;
        if($dbarr['pass_rand']==$respond)
        {
                        $result="<br><br><center><font size='3' face='MS Sans
Serif'><b>รหัสถูกต้อง</b></font></center>" ;
                        print "<meta http-equiv=refresh
content=2;URL=page_4.php>";
        }
        else
        {
                        $result="<br><br><center><font size='3' face='MS Sans
Serif'><b>รหัสไม่ถูกต้อง กรุณาลองใหม่อีกครั้ง</b></font></center>" ;
        }
}
```

```
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-874" />
<title>Untitled Document</title>
</head>

<body>
<form id="form1" name="form1" method="post" >
 <table width="300" border="0" align="center">
  <tr>
   <td width="121">password     </td>
   <td width="169"><?php echo $pass_rand_show;?>
        <input name="pass_rand_show" type="hidden" value="<?php echo
$pass_rand_show; ?>" /></td>
  </tr>
  <tr>
   <td>generate</td>
   <td><?php echo $generate;?>
   <input name="generate" type="hidden" value="<?php echo $generate; ?>" />
   <input name="username" type="hidden" value="<?php echo $username; ?>" />
        </td>
  </tr>
  <tr>
   <td>respond</td>
   <td><input name="respond" type="text" id="respond" size="25" /></td>
  </tr>
  <tr>
   <td> </td>
```

```
      <td><label>
        <input name="Submit3" type="submit" id="Submit3" value="Submit" />
      </label></td>
    </tr>
  </table>
</form>
<?php echo $result; ?>
</body>
</html>
```

Code หน้าที่ 4 ของผู้ให้บริการแสตมป์

```
<?php
include("config.php") ;
$generate=$_POST['generate'];
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-874" />
<title>Untitled Document</title>
</head>

<body>
download file stamp.
<p> </p>
</body>
</html>
```

โปรแกรมที่ใช้ในการแก้ Puzzle ของผู้ส่งใช้ Java

```
import java.math.BigInteger;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

public class lastsearcher {

/**
* @param args
*/
public static void main(String[] args) {

if(args.length!=3)
{
System.out.println("usage: java lastsearcher patial_password password_length
message_digest_of_password");
System.exit(1);
}

String ppwd = args[0];
String pwdLen = args[1];
String mdpwd = args[2];

System.out.println("patial password = "+ppwd);
System.out.println("password length = "+pwdLen);
System.out.println("md password = "+mdpwd);

int bitSpace = (Integer.parseInt(pwdLen)-ppwd.length())*4; // one char = 4 bit

System.out.println("bitSpace = "+bitSpace+" bit");

BigInteger bigI = BigInteger.ZERO;
```

```
BigInteger end = new BigInteger("2");
end = end.pow(bitSpace);

System.out.println("Maximum iterations = "+end.toString());
for(bigI.equals(BigInteger.ZERO);bigI.compareTo(end)==-1;bigI =
bigI.add(BigInteger.ONE))
{
// convert bigI to hexString
String guess = bigI.toString(16);
// zero fill
while(guess.length()<(bitSpace/4))
{
guess = "0"+guess;
}
System.out.println("guess = "+guess);

String candidate = ppwd + guess;
//System.out.println("candidate = "+candidate);
String result = "";

byte[] defaultBytes = candidate.getBytes();
try{
   MessageDigest algorithm = MessageDigest.getInstance("MD5");
   algorithm.reset();
   algorithm.update(defaultBytes);
   byte messageDigest[] = algorithm.digest();

   /*
   StringBuffer hexString = new StringBuffer();
   for (int i=0;i<messageDigest.length;i++) {
      hexString.append(Integer.toHexString(0xFF & messageDigest[i]));
   }
```

```
        result = hexString.toString();
        */

        result = byteArrayToHexString(messageDigest);

        System.out.println("candidate = "+candidate+" md5 version is "+result);
       // System.out.println(result.length());

    }catch(NoSuchAlgorithmException nsae){
    nsae.printStackTrace();
    }

    if(mdpwd.equalsIgnoreCase(result))
    {
    System.out.println("message digest of password = "+mdpwd);
       System.out.println("Finish");
       System.out.println("password = "+candidate);
       break;
    }

    }

    }

    /**
    * Convert a byte[] array to readable string format.
    * This makes the "hex" readable!
    * @return result String buffer in String format
    * @param in byte[] buffer to convert to string format
    */

    static String byteArrayToHexString(byte in[]) {
```

```
    byte ch = 0x00;
    int i = 0;
    if (in == null || in.length <= 0)
        return null;

    String pseudo[] = {"0", "1", "2",
"3", "4", "5", "6", "7", "8",
"9", "a", "b", "c", "d", "e",
"f"};
    StringBuffer out = new StringBuffer(in.length * 2);
    while (i < in.length) {
        ch = (byte) (in[i] & 0xF0); // Strip off high nibble
        ch = (byte) (ch >>> 4); // shift the bits down
        ch = (byte) (ch & 0x0F); // must do this is high order bit is on!

        out.append(pseudo[ (int) ch]); // convert the nibble to a String Character

        ch = (byte) (in[i] & 0x0F); // Strip off low nibble
        out.append(pseudo[ (int) ch]); // convert the nibble to a String Character
        i++;
    }
    String rslt = new String(out);
    return rslt;

}
}
```

ประวัติผู้เขียนโครงงาน

นายวิพุธ คุณูปการ

ภูมิลำเนา 7/81 ถนนพระร่วง ตำบลในเมือง อำเภอเมือง จังหวัดพิษณุโลก

ประวัติการศึกษา

- สำเร็จการศึกษาระดับมัธยมศึกษาจากโรงเรียนพิษณุโลกพิทยาคม จังหวัดพิษณุโลก
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4 สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: bigdanboy@hotmail.com

นายอรรถกร อ้วนวิจิตร

ภูมิลำเนา 777/193 หมู่9 ตำบลอรัญญิก อำเภอเมือง จังหวัดพิษณุโลก

ประวัติการศึกษา

- สำเร็จการศึกษาระดับมัธยมศึกษาจากโรงเรียนพิษณุโลกพิทยาคม จังหวัดพิษณุโลก
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4 สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: atthakornu@hotmail.com