



รายงาน Mini Project เรื่อง ระบบอีเมล (Email System)

เสนอ

อาจารย์ผู้สอน ผู้ช่วยศาสตราจารย์ดร.สวิน วงศ์ปรเมษฐ์

จัดทำโดย

นาย สรวิษฐ์ แพงเจริญ 670112418023 ปี 2 หมู่ที่ 1

โครงการนี้เป็นส่วนหนึ่งของรายวิชา เทคโนโลยีอินเทอร์เน็ต (4132203)

ภาคเรียนที่ 2/2568

สาขาวิชาเทคโนโลยีสารสนเทศ คณะวิทยาศาสตร์ มหาวิทยาลัยราชภัฏบุรีรัมย์

คำนำ

รายงานฉบับนี้เป็นส่วนหนึ่งของการจัดทำ Mini Project โดยมีวัตถุประสงค์เพื่อศึกษาและทำความเข้าใจระบบอีเมล (Email System) ซึ่งเป็นบริการสื่อสารข้อมูลรูปแบบตัวอักษรที่ถูกส่งผ่านเครือข่ายอินเทอร์เน็ตอย่างแพร่หลาย โดยระบบอีเมลเป็นเครื่องมือสำคัญที่ใช้ในการติดต่อสื่อสารทั้งในองค์กร หน่วยงานธุรกิจ และบุคคลทั่วไป

เนื้อหาในรายงานครอบคลุมถึงความหมายและประวัติความเป็นมาของอีเมล รูปแบบการใช้งานโปรโตคอลที่เกี่ยวข้อง เช่น SMTP, POP3, IMAP การทำงานร่วมกับภาษาโปรแกรม พอร์ตที่ใช้งาน รวมถึงตัวอย่างการใช้งาน Email Server ผ่าน Docker พร้อมตัวอย่างไฟล์ docker-compose และภาพประกอบ

ผู้จัดทำหวังว่ารายงานฉบับนี้จะเป็นประโยชน์ต่อผู้อ่านในการทำความเข้าใจระบบอีเมลและการประยุกต์ใช้งานในระดับพื้นฐานและระดับองค์กร

ลงชื่อผู้จัดทำ

นาย สรวิษฐ์ แพงเจริญ

(วันที่ 11 พฤศจิกายน 2568)

เนื้อหา

ระบบอีเมลคืออะไร

ระบบอีเมล (Email หรือ Electronic Mail) เป็นระบบการแลกเปลี่ยนข้อมูลหรือข้อความผ่านอินเทอร์เน็ตหรือเครือข่ายคอมพิวเตอร์ ผู้ส่งสามารถส่งข้อมูลได้ในรูปแบบข้อความ อาจแนบไฟล์ได้ เช่น รูปภาพ วิดีโอ เอกสาร หรือไฟล์ข้อมูลประเภทอื่น ทำให้เป็นวิธีการสื่อสารที่ได้รับความนิยมสูงสุดในปัจจุบัน

โครงสร้างของ Email แบ่งได้เป็น 3 ส่วนหลัก ได้แก่

1. ผู้ส่ง (Sender)
2. เซิร์ฟเวอร์ส่ง/รับ
3. ผู้รับ (Recipient)

ระบบอีเมลใช้วิธีการส่งข้อมูลแบบ Client-Server ผู้ส่งจะส่งข้อมูลไปยังเซิร์ฟเวอร์ และเซิร์ฟเวอร์จะส่งต่อไปยังปลายทาง

ประวัติความเป็นมา

- ปี 1971
Ray Tomlinson พัฒนาระบบอีเมลบนเครือข่าย ARPANET
- และคิดค้นการใช้สัญลักษณ์ “@” ในอีเมลเพื่อแบ่งชื่อผู้ใช้กับชื่อโฮสต์
- ปี 1982
SMTP ถูกพัฒนาขึ้นเป็นมาตรฐานการส่งอีเมล
- ปี 1984
มี POP (Post Office Protocol)
- ปี 1996
IMAP ถูกพัฒนาขึ้นมาเพื่อแก้ข้อจำกัดของ POP

จากนั้นระบบ Email ได้กลายเป็นระบบสื่อสารหลักในองค์กรและผู้ใช้ทั่วไป จนมาถึงยุคปัจจุบันที่ Email สามารถทำงานร่วมกับระบบคลาวด์ ความปลอดภัยผ่านการเข้ารหัส และมีการยืนยันตัวตนหลายขั้นตอนเพื่อป้องกันสแปมและการโจมตี

องค์ประกอบหลักของระบบอีเมล

1. Mail User Agent (MUA) – เช่น Gmail, Outlook
2. Mail Transfer Agent (MTA) – เช่น Sendmail, Postfix
3. Mail Delivery Agent (MDA)
4. Email Server
5. Protocol SMTP / POP3 / IMAP

โปรโตคอลที่เกี่ยวข้อง

SMTP — (Simple Mail Transfer Protocol)

ใช้สำหรับส่งอีเมล (Outgoing mail)

พอร์ตที่ใช้

- 25
- 465 (SSL/TLS)
- 587 (StartTLS)

ขั้นตอนทำงาน

1. ผู้ใช้ส่งอีเมลผ่าน MUA
2. ส่งไป MTA ผ่าน SMTP
3. ส่งต่อไปยัง MTA ปลายทาง
4. MDA ส่งต่อไปยังผู้รับ

POP3 — (Post Office Protocol v3)

ใช้ดึงอีเมลจากเซิร์ฟเวอร์มาเก็บที่เครื่อง

พอร์ต

- 110
- 995 (SSL/TLS)

ข้อดี: เร็ว, เก็บข้อมูลในเครื่อง

ข้อเสีย: ไม่ sync หลายอุปกรณ์

IMAP — Internet Message Access Protocol

ใช้เก็บอีเมลบนเซิร์ฟเวอร์

พอร์ต

- 143
- 993 (SSL/TLS)

ข้อดี: sync ได้หลายอุปกรณ์, จัดโฟลเดอร์ได้

การใช้งานร่วมกับภาษาโปรแกรม

ตัวอย่าง Python ส่งอีเมล

```
import smtplib
```

```
server = smtplib.SMTP("smtp.gmail.com", 587)
```

```
server.starttls()
```

```
server.login("your_email@gmail.com", "password")
```

```
server.sendmail("your_email@gmail.com", "target@example.com", "Hello")
```

```
server.quit()
```

ตัวอย่างนี้ใช้ SMTP ส่งข้อความผ่าน Gmail SMTP Server

พอร์ตที่ใช้งาน

โปรโตคอล	พอร์ต
SMTP	25 / 465 / 587
POP3	110 / 995
IMAP	143 / 993

ภาพประกอบ :

ภาพ: โครงสร้างระบบ Email ระหว่างผู้ส่งและผู้รับ

ผู้ส่ง → SMTP Server → MTA → MDA → POP3/IMAP Client → ผู้รับ

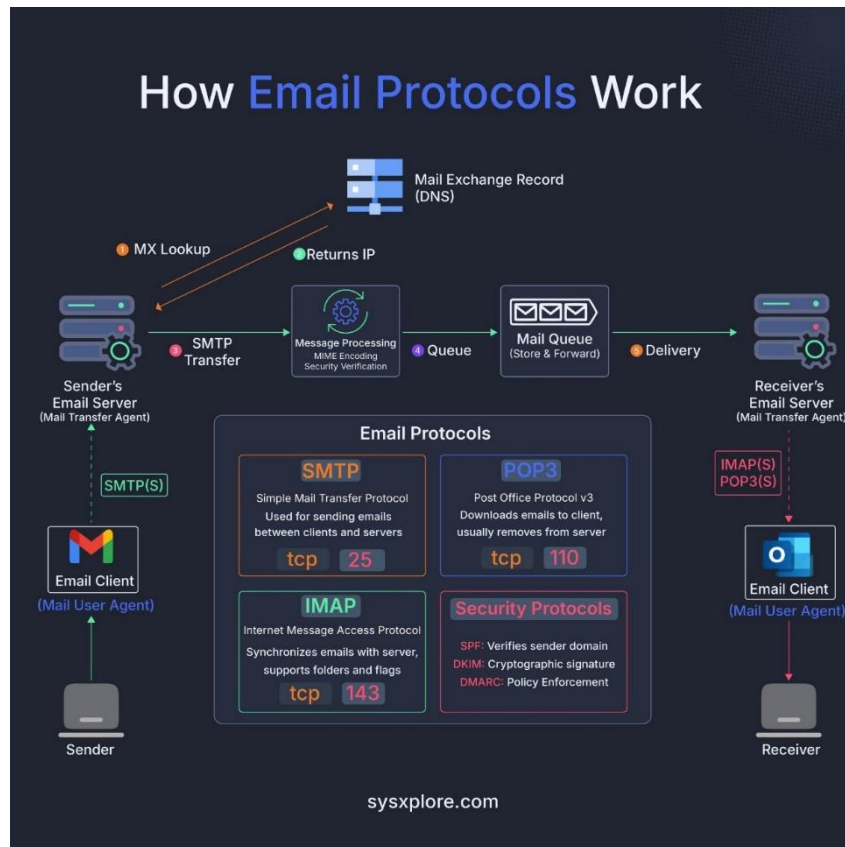


Image บน Docker Hub

ตัวอย่าง Mail Server Image

Image

รายละเอียด

mailserver/docker-mailserver

ใช้งานง่าย รองรับ
SMTP/IMAP/POP3

posteo/poste.io

Web UI สะดวก

ตัวอย่างการ pull image

docker pull mailserver/docker-mailserver

ตัวอย่าง docker-compose

version: '3.8'

services:

mailserver:

image: mailserver/docker-mailserver:latest

hostname: mail.example.com

container_name: mailserver

ports:

- "25:25"
- "465:465"
- "587:587"
- "993:993"

volumes:

- ./data:/var/mail
- ./config:/tmp/docker-mailserver/

environment:

- ENABLE_SPAMASSASSIN=1
- ENABLE_CLAMAV=1

restart: always

คำอธิบาย

- เปิดพอร์ต SMTP/IMAP
- ใช้ volume เก็บข้อมูล Email

อ้างอิง

1. <https://www.ietf.org/>
2. <https://mailserver.github.io/>
3. <https://www.rfc-editor.org>
4. <https://www.sendmail.com/>
5. <https://docs.docker.com/>