Logo

Description automatically generated

**ระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์**

**Web Server Performance Monitoring System**

**โดย**

**นายเจฟฟี่ ดาโต๊ะ รหัสนักศึกษา 6110210071**

**นายภาสวี สุวรรณโณ รหัสนักศึกษา 6110210608**

**โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตร์บัณฑิต** **(วิทยาการคอมพิวเตอร์)**

**สาขาวิทยาศาสตร์การคำนวณ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์**

**ปีการศึกษา 2563**

**ชื่อโครงงาน** ระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์

**ผู้จัดทำ** นายเจฟฟี่ ดาโต๊ะ **รหัสนักศึกษา** 6110210071

นายภาสวี สุวรรณโณ **รหัสนักศึกษา** 6110210608

|  |  |
| --- | --- |
| **อาจารย์ที่ปรึกษาโครงงาน**  ………………………………………..  (ผู้ช่วยศาสตราจารย นิธิ ทะนนท์) | **คณะกรรมการสอบ**  …………………………………….. ประธานกรรมการ  (ผู้ช่วยศาสตราจารย นิธิ ทะนนท์)  …………………………………….. กรรมการสอบคนที่ 1  (ผู้ช่วยศาสตราจารย ดร.ชินพงศ์ อังสุโชติเมธี)  ………………………………….. กรรมการสอบคนที่ 2  (อาจารย์สมศักดิ์ คงแสง) |

**ชื่อโครงงาน** ระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์

**ผู้จัดทำ** นายเจฟฟี่ ดาโต๊ะ **รหัสนักศึกษา** 6110210071

นายภาสวี สุวรรณโณ **รหัสนักศึกษา** 6110210608

**ปีการศึกษา 2564**

**บทคัดย่อ**

ในปัจจุบันแนวโน้มเทคโนโลยีด้านการรักษาความปลอดภัยไซเบอร์ถูกจับตามองมากขึ้น เพราะมีความเสี่ยงที่จะถูกโจมตีจากผู้ไม่หวังดี และมีรายงานการโจมตีโดยวิธี HTTP Flood อยู่บ่อยครั้ง โดยในโครงงานเล่มนี้จะกล่าวถึงระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ ซึ่งสามารถช่วยให้ผู้ดูแลระบบสามารถทราบถึงข้อมูลสถานะของเว็บเซิร์ฟเวอร์ ในรูปแบบเว็ปไซต์เพื่อเพิ่มประสิทธิภาพในการตรวจสอบสถานะ และนำข้อมูลสถานะไปวิเคราะห์เพื่อตรวจสอบความผิดปกติของข้อมูล รวมถึงมีการแจ้งเตือนไปยังผู้ดูแลได้อย่างรวดเร็วผ่าน Line Notify เมื่อมีค่าข้อมูลสถานะที่ผิดปกติ หรือมีความเสี่ยงว่าถูกโจมตี จากการพัฒนาพบว่า

**Project Title Web Server Performance Monitoring System**

**By Jeffy Datoh** **Student ID 6110210071**

**Academic Year 2021**

**ABSTRACT**

Nowadays, cybersecurity technology trends are progressively being scrutinized these days. Because there's a chance, you'll be attacked by someone with nefarious intentions. HTTP Flood assaults are also a common occurrence. We will discuss a web server performance monitoring system in this project. This can assist administrators in determining the web server's status. in the form of a webpage to make checking the status more efficient Furthermore, the status data is evaluated to look for any data anomalies. Quickly sending alerts to careers via Line Notify When there is an abnormal status information or are in risk of being attacked

# คำนำ

โครงงานนี้เป็นส่วนหนึ่งการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์) สาขาวิทยาศาสตร์การคำนวณ คณะวิทยาศาสตร์ ปีการศึกษา 2564โดยในโครงงานนี้ได้ทำการศึกษาในหัวข้อระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์

ข้อมูลในโครงงานฉบับนี้ เป็นการรายงานผลการดำเนินงาน การศึกษาค้นคว้าข้อมูลจากแหล่งต่าง ๆ จึงนำมารวบรวม เรียบเรียงไว้ในโครงงานฉบับนี้ หากข้อมูลผิดพลาดประการใด ผู้จัดทำขออภัยมา ณ ทีน

นายเจฟฟี่ ดาโต๊ะ

นายภาสวี สุวรรณโณ

**กิตติกรรมประกาศ**

โครงงานฉบับนี้สำเร็จได้ด้วยความช่วยเหลือและสนับสนุนจากบุคคลหลายฝ่าย ผู้จัดทำรู้สึกซาบซึ้งและขอขอบพระคุณอย่างสูงมา ณ โอกาสนี้คือ

ผู้ช่วยศาสตราจารย์ นิธิ ทะนนท์ อาจารย์ที่ปรึกษาโครงงาน ที่กรุณาให้ความรู้ คำปรึกษาแนะนำ และช่วยเหลือในการแก้ปัญหาต่างๆให้แก่ผู้จัดทำเสมอมา รวมถึงถ่ายทอดหลักการแนวคิดในการพัฒนาตัวเองและคุณธรรมสำหรับวิชาชีพแก่ผู้จัดทำมาโดยตลอด

ผู้ช่วยศาสตราจารย ดร.ชินพงศ์ อังสุโชติเมธี กรรมการสอบโครงงาน ที่ให้คำปรึกษา และกรุณาช่วยตรวจทานและแก้ไขโครงงานให้มีความสมบูรณ์

อาจารย์สมศักดิ์ คงแสง กรรมการสอบโครงงาน ที่ให้คำปรึกษา และกรุณาช่วยตรวจทานและแก้ไขโครงงานให้มีความสมบูรณ์

นอกจากนี้ยังต้องขอขอบคุณหลักสูตร วท.บ. วิทยาการคอมพิวเตอร์ สาขาวิทยาศาสตร์การคำนวณ ที่เอื้อเฟื้อสถานที่และอุปกรณ์ในการทำโครงงานเพื่อนๆ ภาควิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ ที่ให้คำปรึกษา และช่วยเหลือในการทำโครงงาน

ผู้จัดทำขอขอบพระคุณทุกท่านเป็นอย่างสูง ณ โอกาสนี้ด้วย สุดท้ายนี้ หากโครงงานชิ้นนี้มีข้อผิดพลาดประการใด ผู้พัฒนาขออภัยมา ณ ที่นี้และจะน้อมรับทุกคำติชมมาพัฒนาโครงงานให้ดียิ่งขึ้นในอนาคตต่อไป

นายเจฟฟี่ ดาโต๊ะ

นายภาสวี สุวรรณโณ

**สารบัญ**

[**บทคัดย่อ** 3](#_Toc82628449)

[**ABSTRACT** 4](#_Toc82628450)

[คำนำ 5](#_Toc82628451)

[**กิตติกรรมประกาศ** 6](#_Toc82628452)

[บทที่ 1 1](#_Toc82628453)

[1.1 ความเป็นมาของโครงงาน 1](#_Toc82628454)

[1.2 วัตถุประสงค์ 1](#_Toc82628455)

[1.3 ขอบเขตของโครงงาน 1](#_Toc82628456)

[1.4 ระยะเวลาในการดำเนินงาน 1](#_Toc82628457)

[1.5 แผนภาพระยะเวลาดำเนินงาน 2](#_Toc82628458)

[1.6 ประโยชน์ที่คาดว่าจะได้รับ 2](#_Toc82628459)

[1.7 ระเบียบวิธีโครงงาน 3](#_Toc82628460)

[1.8 สถานที่และเครื่องมือที่ใช้ทำโครงงาน 4](#_Toc82628461)

[1.9 อาจารย์ที่ปรึกษา 4](#_Toc82628462)

[1.10 ผู้จัดทำโครงงาน 4](#_Toc82628463)

[บทที่ 2 5](#_Toc82628464)

[2.1 Web Server 5](#_Toc82628465)

[2.2 Web Monitoring 5](#_Toc82628466)

[2.3 Apache HTTP Server 5](#_Toc82628467)

[2.4 Server Monitor 6](#_Toc82628468)

[2.5 HTTP 6](#_Toc82628469)

[2.6 HTTP REQUEST 6](#_Toc82628470)

[2.8 Mod\_Status 7](#_Toc82628471)

[2.9 LINE Notify 7](#_Toc82628472)

[บทที่ 3 8](#_Toc82628473)

[3.1 สถาปัตยกรรมของระบบตรวจสอบประสิทธิภาพเว็บเซิร์ฟเวอร์ 8](#_Toc82628474)

[**(1) Attacker** 9](#_Toc82628475)

[**(2) Web Server** 9](#_Toc82628476)

[**(3) Performance Monitoring System** 9](#_Toc82628477)

[**(4) Administrator** 10](#_Toc82628478)

[**บทที่ 4** 12](#_Toc82628479)

[4.1 การนำเข้าข้อมูลสถานะจากเว็บเซิร์ฟเวอร์ 12](#_Toc82628480)

[4.2 การแสดงกราฟข้อมูลสถานะ 15](#_Toc82628481)

[4.3 การตรวจสอบข้อมูลสถานะ 15](#_Toc82628482)

[4.4 แจ้งเตือนสถานะไปยัง Line Notify 15](#_Toc82628483)

[**บทที่ 5** 17](#_Toc82628484)

[5.1 บทสรุป 17](#_Toc82628485)

[5.2 ปัญหาและแนวทางแก้ปัญหา 17](#_Toc82628486)

[5.3 ข้อเสนอแนะ 17](#_Toc82628487)

[เอกสารอ้างอิง 18](#_Toc82628488)

# บทที่ 1

**บทนำ**

## 1.1 ความเป็นมาของโครงงาน

HyperText Transfer Protocol (HTTP) คือ โพรโทคอลในระดับชั้นโปรแกรมประยุกต์เพื่อการแจกจ่ายและการทำงานร่วมกันกับสารสนเทศของสื่อผสม ใช้สำหรับการรับทรัพยากรที่เชื่อมโยงกับภายนอก ซึ่งนำไปสู่การจัดตั้งเวิลด์ไวด์เว็บ (world wide web) และ เว็บเซิร์ฟเวอร์ (Web Server)

หน่วยงานต่างๆ มีการติดตั้งเว็บเซิร์ฟเวอร์ (Web Server) เป็นของตนเองเพื่อใช้ในการเผยแพร่ข้อมูลข่าวสาและประชาสัมพันธ์หน่วยงาน เนื่องจากเว็บเซิร์ฟเวอร์สามารถเข้าถึงได้จากบุคคลทั่วไปซึ่งเปิดโอกาศให้ผู้ที่ไม่หวังดีสามารถทำการโจมตีเว็บเซิร์ฟเวอร์จนไม่สามารถให้บริการได้อย่างเต็มประสิทธิภาพ ดังนั้นทางผู้วิจัยจึงมีแนวคิดในการหาทางป้องกันปัญหาดังกล่าว โดยพัฒนาระบบสำหรับใช้ตรวจสอบการโจมตีเว็บเซิร์ฟเวอร์เพื่อให้สามารถป้องกันการโจมตีได้อย่างรวดเร็ว

## 1.2 วัตถุประสงค์

เพื่อวิเคราะห์และออกแบบระบบสำหรับตรวจสอบการโจมตีเว็บเซิร์ฟเวอร์

## 1.3 ขอบเขตของโครงงาน

* ผู้ดูแลระบบสามารถตรวจสอบประสิทธิภาพการทำงานของเว็บเซิร์ฟเวอร์ได้
* ผู้ดูแลระบบสามารถตรวจสอบสถานะการทำงานของเว็บเซิร์ฟเวอร์ เช่น
* การใช้งาน CPU, Network Interface
* การกําหนดค่าสถานะที่เว็บเซิร์ฟเวอร์พร้อมใช้งานอย่างมีประสิทธิภาพ

## 1.4 ระยะเวลาในการดำเนินงาน

* มกราคม 2564 - ตุลาคม 2564

## 1.5 แผนภาพระยะเวลาดำเนินงาน

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **การดำเนินงาน** | **ระยะเวลาการดำเนินงาน** | | | | | | |
| **ม.ค.** | **ก.พ.** | **มี.ค.** | **มิ.ย.** | **ก.ค.** | **ส.ค.** | **ก.ย.** |
| คิดหัวข้อโครงการ |  |  |  |  |  |  |  |
| กำหนดขอบเขต |  |  |  |  |  |  |  |
| ศึกษางานวิจัยที่เกี่ยวข้องและเทคโนโลยีที่ใช้ |  |  |  |  |  |  |  |
| วางแผนขั้นตอนการดำเนินงาน |  |  |  |  |  |  |  |
| วิเคราะห์และออกแบบระบบ |  |  |  |  |  |  |  |
| พัฒนาระบบ |  |  |  |  |  |  |  |
| ติดตั้งและทดสอบระบบ |  |  |  |  |  |  |  |
| แก้ไขข้อผิดพลาดของระบบ |  |  |  |  |  |  |  |
| จัดทำคู่มือและเอกสารประกอบโครงงาน |  |  |  |  |  |  |  |

## 1.6 ประโยชน์ที่คาดว่าจะได้รับ

1. ระบบที่พัฒนาขึ้นมาเพื่อช่วยติดตามปัญหาและตรวจสอบประสิทธิภาพความพร้อมใช้งานของเว็บเซิร์ฟเวอร์

2. มีการตอบสนองต่อปัญหาและแก้ไขได้รวดเร็วทําให้ผู้ใช้งานระบบได้รับความพอใจและเชื่อถือระบบต่อไป

## 1.7 ระเบียบวิธีโครงงาน

|  |  |  |  |
| --- | --- | --- | --- |
| **ขั้นที่** | **วัตถุประสงค์** | **การดำเนินงาน** | **สิ่งที่ได้** |
| 1. | การศึกษางานวิจัยที่เกี่ยวข้องและเทคโนโลยีที่ใช้ | | |
|  | - ศึกษาทำความเข้าใจเกี่ยวกับ  ปัญหาและกระบวนการของ  ระบบงาน | - ศึกษากิจกรรมหลักของ  ระบบงาน และข้อมูลที่เกี่ยวข้อง | - Proposal |
| 2. | การวิเคราะห์และออกแบบระบบ (Analysis) | | |
|  | - วิเคราะห์และหาความต้องการ  ของระบบเดิม | - กำหนดขอบเขตความต้องการ  และขอบเขตของฐานข้อมูล  ระบบงานใหม่  - Conceptual Data Model | - Use Case Diagram  - DFD  - ER Diagram |
| 3. | การออกแบบระบบ (Design) | | |
|  | - ออกแบบโครงสร้างระบบให้  เหมาะสมกับความต้องการ | - ออกแบบจอภาพการรับข้อมูล  เข้า, การส่งข้อมูลออก และการ  แสดงผลข้อมูล  - นำ ER Diagram มาออกแบบ  DBMS ของระบบ | - Data Dictionary  - โครงสร้างระบบ |
| 4. | การพัฒนา (Development) | | |
|  | - พัฒนาโปรแกรมด้วยชุดคำสั่ง  เพื่อสร้างระบบงาน | - พัฒนาโปรแกรมที่ได้จาการ  วิเคราะห์และออกแบบไว้ | - ชุดคำสั่งต้นแบบ  (Source Code) |
| 5.. | การทดสอบและปรับปรุงแก้ไข (Testing) | | |
|  | - ทดสอบระบบก่อนนำไปใช้งาน  จริง | - ตรวจสอบและปรับปรุงแก้ไขการ  ทำงานของโปรแกรมให้ตรงกับ  การวิเคราะห์และออกแบบไว้ | - Testing Report |
| 6. | จัดทำเอกสารประกอบ (Document) | | |
|  | - ผู้ใช้งานระบบมีคู่มือในการใช้งาน | - จัดทำคู่มือให้กับผู้ใช้งานระบบ | - คู่มือการใช้งานระบบ |

## 1.8 สถานที่และเครื่องมือที่ใช้ทำโครงงาน

สถานที่

* อาคารวิทยาการคอมพิวเตอร์ สาขาวิชาวิทยาศาสตร์การคำนวณคณะวิทยาศาสตร์ มหาวิทยาสงขลานครินทร์
* หอพักนักศึกษาที่ 11 มหาวิทยาลัยสงขลานครินทร์

เครื่องมือที่ใช้ในการทำโครงงาน

* ซอฟต์แวร์
  + Apache HTTP Server
  + Visual studio code
* ฮาร์ดแวร์
  + Microcomputer

## 1.9 อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์ นิธิ ทะนนท์

## 1.10 ผู้จัดทำโครงงาน

นายเจฟฟี่ ดาโต๊ะ รหัสนักศึกษา 6110210071

นายภาสวี สุวรรณโณ รหัสนักศึกษา 6110210608

นักศึกษาสาขาวิชาวิทยาศาสตร์การคำนวณ (วิทยาการคอมพิวเตอร์) ชั้นปีที่ 3

คณะวิทยาศาสตร์ มหาวิทยาสงขลานครินทร์

# บทที่ 2

**ทฤษฎีและหลักการที่เกี่ยวข้อง**

เนื้อหาในส่วนนี้เป็นการกล่าวถึงทฤษฎีและหลักการที่เกี่ยวข้องในการพัฒนาระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ประกอบด้วยเนื้อหาดังต่อไปนี้

## 2.1 Web Server

เว็บเซิร์ฟเวอร์ (Web Server) คือ เครื่องคอมพิวเตอร์ ที่ทำการติดตั้ง Software เกี่ยวกับ Server ทำหน้าที่บริการและส่งข้อมูลให้กับ Client Request ข้อมูล และ Web Server จะ Response เพื่อส่งผลลัพธ์ไปยัง Client โดย Web Server ทั่ว ๆ ไปจะติดตั้ง Software ที่ชื่อว่า IIS (Internet Information Services) และ Apache โดย Service เหล่านี้จะบริการข้อมูลผ่าน Port : 80 และรับส่งข้อมูลผ่าน Web Browser

## 2.2 Web Monitoring

การตรวจสอบการทำงานของเว็บไซต์ว่ามีการทำงานอยู่ หรือหยุดทำงานไป (up/down- time) การเปลี่ยนแปลงหน้าเว็บไซต์ (Defacement) และตรวจสอบสิ่งผิดปกติบนเว็บไซต์ นอกจากนี้ยังสามารถแจ้งเตือนไปยังผู้ดูแลระบบได้ผ่านช่องทางกำหนด เช่น SMS, Email อื่น ๆ

## 2.3 Apache HTTP Server

Apache HTTP Server เป็นซอฟต์แวร์เซิร์ฟเวอร์ที่ใช้กันอย่างแพร่หลาย Apache พัฒนาและดูแลโดย Apache Software Foundation ซึ่งเป็นซอฟต์แวร์โอเพนซอร์สที่สามารถใช้งานได้ฟรี โดยมีการใช้โดยรวมประมาณ 67% ของเว็บเซิร์ฟเวอร์ทั้งหมดในโลก ซึ่งรวดเร็วเชื่อถือได้และปลอดภัย สามารถปรับแต่งได้เพื่อตอบสนองความต้องการของสภาพแวดล้อมที่หลากหลาย โดย สามารถเพิ่ม function พิเศษที่เป็น module plugin ได้โดยง่าย

## 2.4 Server Monitor

การตรวจสอบประสิทธิภาพของเซิร์ฟเวอร์เป็นกระบวนการตรวจสอบทรัพยากรของเซิร์ฟเวอร์เช่น การใช้งาน CPU, การใช้งานหน่วยความจำ, การใช้งานดิสก์ เป็นต้น โดยการตรวจสอบประสิทธิภาพของเซิร์ฟเวอร์จะใช้งานซอฟต์แวร์สำหรับตรวจสอบการทำงาน ซึ่งช่วยให้สามารถตรวจสอบเซิร์ฟเวอร์ได้อัตโนมัติ และยังเป็นกระบวนการที่ช่วยวางแผนการใช้งานเซิร์ฟเวอร์ นอกจากนี้การตรวจสอบประสิทธิภาพของเซิร์ฟเวอร์ยังช่วยในการระบุปัญหาอื่น ๆที่เกี่ยวข้องกับประสิทธิภาพ เช่น การตอบสนองการให้บริการ,การใช้งานทรัพยากร และอื่น ๆ

## 2.5 HTTP

เป็น โปรโตคอล ที่ใช้งานสำหรับเผยแพร่ข้อมูล และ เป็นสื่อการสำหรับการสื่อสารระหว่าง web browser กับ web server อีกทั้งเป็นจุดกำเนิดของ World Wide Web ซึ่งมีโครงสร้างเป็นตัวอักษรและตัวเลข (text) ใช้สำหรับเป็น link เชื่อมระหว่าง ข้อมูล text อื่น ๆ และถูกใช้ในการแลกเปลี่ยน ข้อมูลในรูปแบบ multimedia สามารถเเรียกใช้งานผ่าน web browser เช่น Firefox, Google Chrome, Safari, Opera และ Microsoft Internet Explorer ซึ่งจะไปทำการดึงและแลกเปลี่ยนข้อมูลกับทาง Server ที่ระบุใน URL ข้อมูลที่ส่งไปจะอยู่ในรูป plain text ไม่มีการเข้ารหัส ทำให้สามารถถูกดักจับและอ่านข้อมูลได้ง่าย โดยส่ง request ไปและรอจนกว่าจะได้ response จากเครื่องแม่ข่ายกลับมาจึงไม่ปลอดภัย

## 2.6 HTTP REQUEST

คือการส่งคำขอไปยังเชิิฟเวอร์เพื่อต้องการข้อมูล โดยจากฝั่ง Client ไปยังฝั่ง Server เพื่อให้ส่งข้อมูลที่ต้องการกลับมา แบ่งได้ 4 แบบ

* Request-Line จะเริ่มด้วย token method ตามด้วย Request-URI และ version ของ protocol จบด้วย CRLF ทุกส่วนจะถูกคั่นด้วยช่องว่าง
* Request Method จะกำหนด action ที่จะกระทำต่อข้อมูล method จะมีลักษณะเป็นตัวพิมพ์ใหญ่ทั้งหมดเสมอโดย method ที่สำคัญมี 4 ตัวคือ GET, POST, PUT, DELETE
  + GET — เป็นการเรียกรับข้อมูลจาก URI ที่กำหนด method GET ควรใช้ในการดึงข้อมูลเท่านั้นและต้องไม่มีผลกระทบใด ๆ กับข้อมูล
  + POST — ใช้สำหรับการสร้างข้อมูลใหม่โดยส่งข้อมูลผ่าน body
  + PUT — ใช้สำหรับแทนที่ข้อมูลที่มีทั้งหมดด้วยข้อมูลใหม่ที่ส่งขึ้นไป
  + DELETE — ใช้สำหรับลบข้อมูลที่มีอยู่ ของเป้าหมายที่กำหนดโดย URI
* Request Header Fields

จะให้ ผู้ใช้ สามารถส่งผ่านข้อมูลเพิ่มเติมเกี่ยวกับ Request ของผู้ใช้ เช่น เป็นภาษาอะไร, ชนิดข้อมูลเป็นแบบไหน และนี่คือรายชื่อของ header ที่สำคัญบางอย่างที่สามารถเลือกใช้ได้

* + Accept-Charset
  + Accept-Encoding
  + Accept-Language
  + Connection
  + Content-Length
  + Content-Type
  + Cookie
  + Host
  + User-Agent
* Request message

จะเป็นส่วนที่ใช้สำหรับส่งข้อมูลให้กับ server อาจจะมีหรือไม่มีการส่งก็ได้ขึ้นอยู่กับบริการที่เราเรียกด้วย

## 2.8 Mod\_Status

mod\_status เป็นโมดูล Apache ที่ช่วยตรวจสอบการโหลดของเว็บเซิร์ฟเวอร์และการเชื่อมต่อ http ในปัจจุบันด้วยอินเทอร์เฟซ HTML ที่สามารถเข้าถึงได้ผ่านเว็บเบราว์เซอร์ และ mod\_status ของ Apache นั้นแสดงหน้า HTML ที่มีข้อมูลเกี่ยวกับสถิติปัจจุบันของเว็บเซิร์ฟเวอร์

## 2.9 LINE Notify

LINE Notify คือ บริการที่คุณสามารถได้รับข้อความแจ้ง เตือนจากเว็บเซอร์วิสต่าง ๆ ที่คุณสนใจได้ทาง LINE โดย หลังเสร็จสิ้นการเชื่อมต่อกับทางเว็บเซอร์วิสแล้ว คุณจะ ได้รับการแจ้งเตือนจากบัญชีทางการของ “LINE Notify” ซึ่งให้บริการโดย LINE นั่นเอง คุณสามารถ เชื่อมต่อกับบริการที่หลากหลาย และยังสามารถรับ การ แจ้งเตือนทางกลุ่มได้อีกด้วย ซึ่งบริการหลักๆ ที่สามารถ เชื่อมต่อได้แก่ GitHub, IFTTT เป็นต้น

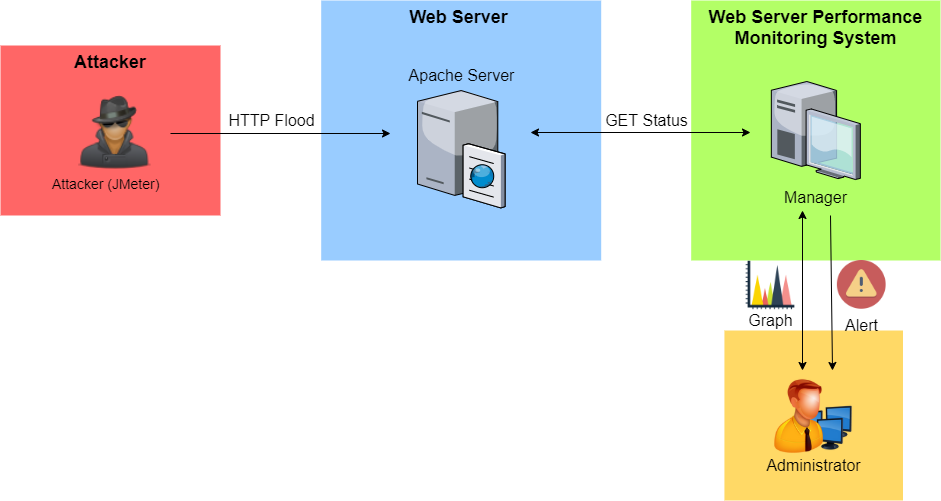
# บทที่ 3

**การวิเคราะห์และออกแบบระบบ**

เนื้อหาในบทนี้จะกล่าวถึงการออกแบบและสถาปัตยกรรมของระบบตรวจสอบประสิทธิภาพเว็บเซิร์ฟเวอร์ โดยมีรายละเอียดตามลำดับดังนี้

## 3.1 สถาปัตยกรรมของระบบตรวจสอบประสิทธิภาพเว็บเซิร์ฟเวอร์

จากบทที่ 2 ได้กล่าวถึงและอธิบายเกี่ยวกับเว็บเซิร์ฟเวอร์และเทคโนโลยีเกี่ยวกับเว็บเซิร์ฟเวอร์ในการพัฒนาระบบ ทางผู้จัดจึงได้ออกแบบสถาปัตยกรรมของระบบตรวจสอบประสิทธิภาพเว็บเซิร์ฟเวอร์ ดังตัวอย่างภาพประกอบ 3.1



**ภาพประกอบที่ 3.1** **สถาปัตยกรรมของระบบตรวจสอบประสิทธิภาพเว็บเซิร์ฟเวอร์**

จากภาพประกอบที่ 3.1 สามารถจำแนกส่วนประกอบหลักของระบบตรวจสอบประสิทธิภาพเว็บเซิร์ฟเวอร์ออกเป็น 4 ส่วน คือ (1) Attacker, (2) Web Server, (3) Performance Monitoring System และ (4) Administrator โดยแต่ละส่วนมีลักษณะการทำงานดังนี้

### **(1) Attacker**

Attacker จะทำการโจมตีไปยัง Web Server โดยใช้โปรแกรม JMeter ซึ่งเป็นเครื่องมือจาก Apache ด้วยรูปแบบ HTTP Flood Attack โดยทำการสร้าง HTTP Request จำนวนมากส่งเข้าไปยัง Web Server เป้าหมาย หากทำการโจมตีสำเร็จ Web Server จะไม่สามารถให้บริการแก่ผู้ใช้งานระบบอื่น ๆ ที่ต้องการใช้บริการจริง ๆ

### **(2) Web Server**

Web Server คือ เครื่องคอมพิวเตอร์ที่ทำการติดตั้งโปรแกรม Apache Web Server โดยเปิดใช้ mod\_status ซึ่งเป็นโมดูลสำหรับแสดงสถานะการทำงานในรูปแบบเอกสาร HTML

### **(3) Performance Monitoring System**

ในส่วนนี้เป็นโปรแกรมที่ผู้วิจัยได้พัฒนาขึ้นมาใหม่ด้วยภาษา PHP ซึ่งจะทำการดึงค่าข้อมูล Total Accesses, Total kBytes, CPU System, CPU Load, Uptime, RequsetPerSecond, Bytes PerSecond, BytesPerRequest และ DurationPerRequest จาก Web server ผ่านทางโมดูล mod\_status ดังแสดงในภาพประกอบที่ 3.2 และในรูปแบบตารางดังแสดงในภาพประกอบที่ 3.3

Text, letter

Description automatically generated

**ภาพประกอบที่ 3.2 แสดงรายละเอียดของ log file**

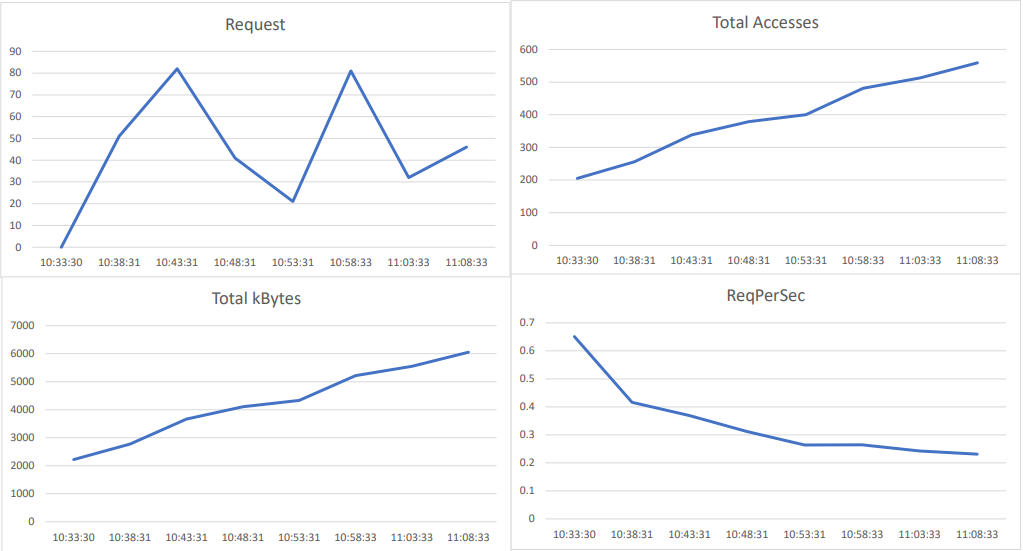
จากภาพประกอบที่ 3.2 จะแสดงให้เห็นรายละเอียดข้อมูลค่าสถานะของ Web Server โดยแต่ละบรรทัดมีความหมายดังนี้

* วันเวลาที่บันทึกได้
* Total Accesses: จำนวน Request ทั้งหมดที่เข้ามายัง Web Server
* Total kBytes: จำนวนไบต์ทั้งหมดของ Web Server
* CPU System: ปริมาณที่ CPU ใช้งาน
* CPU Load: ปริมาณงานที่ให้ CPU ทำงาน
* Up Time: ระยะเวลาเวลาทำงานของ Server
* ReqPerSec: จำนวน Request ต่อวินาที
* BytesPerSec: จำนวนไบต์ต่อวินาที
* BytesPerReq: จำนวนไบต์ต่อ Request
* DurationPerReq :ระยะเวลาต่อ Request

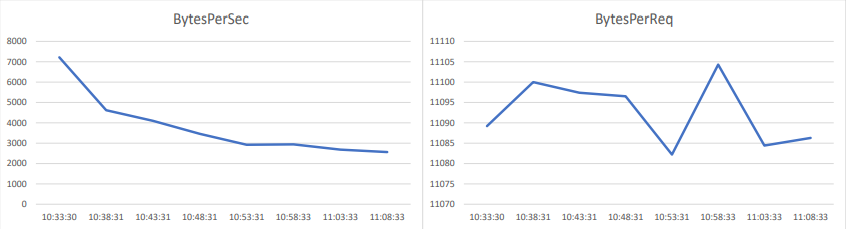
ในส่วนของการแจ้งเตือน จะทำตรวจสอบค่าสถานะต่าง ๆ ที่สนใจของ Web Server หากค่าสถานะที่ตรวจสอบมีความผิดปกติ เช่น จำนวน Request ที่เข้ามายัง Web Server คือ 82 Request ภายใน 5 นาทีจากจำนวน Request ที่เก็บได้ก่อนหน้านี้มีจำนวน 51 Request จะเห็นว่ามีจำนวน Request เพิ่มขึ้นผิดปกติ แสดงว่า Web Server ถูกโจมตี และทำการแจ้งเตือนผ่าน Line Notify เพื่อส่งข้อความแจ้งเตือนไปยังผู้ดูแลระบบ

### **(4) Administrator**

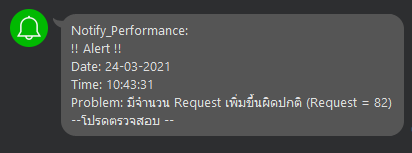
Administrator คือ ผู้ดูแลระบบสามารถเรียกดูสถานะการทำงานของ Web Server ในรูปแบบกราฟ ดังแสดงในภาพประกอบที่ 3.4 และ 3.5 ในกรณีที่ Web Server ถูกโจมตีจะมีการแจ้งเตือนผ่าน Line Notify ดังแสดงในภาพประกอบที่ 3.6

**ภาพประกอบที่ 3.4 แสดงข้อมูลค่าสถานะ Request, Total Accesses,**

**Total kBytes**, **ReqPerSec และ BytesPerReq**



**ภาพประกอบที่ 3.5 แสดงข้อมูลค่าสถานะ BytesPerSec และ BytesPerReq**



**ภาพประกอบที่ 3.6 แสดงข้อความการแจ้งเตือนผ่าน****แอพพลิเคชั่น Line**

**บทที่ 4**

**การพัฒนาระบบ**

ในบทนี้จะกล่าวถึงการพัฒนาระบบตรวจสอบประสิทธิภาพเว็บเซิร์ฟเวอร์ สำหรับการพัฒนาระบบจะประกอบด้วย การนำเข้าข้อมูลสถานะจากเว็บเซิร์ฟเวอร์, การแสดงกราฟข้อมูลสถานะ, การตรวจสอบข้อมูลสถานะ และแจ้งเตือนสถานะไปยัง Line Notify

## 4.1 การนำเข้าข้อมูลสถานะจากเว็บเซิร์ฟเวอร์

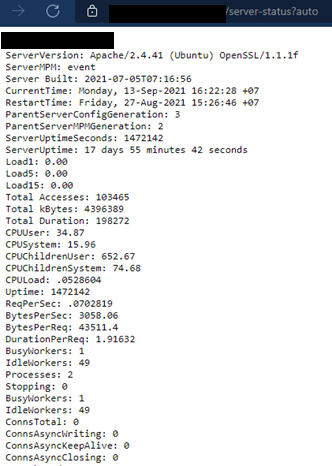
โดยทางผู้จัดทำได้การนำเข้าข้อมูลสถานะจาก Apache Server Status ซึ่งจำเป็นต้องเปิดและตั้งค่า Apache Extended Status เพื่อให้สามารถดูสถานะการทำงานของ Apache ได้ ดังในภาพ 3.7

Text

Description automatically generated

**ภาพประกอบที่ 4.1 แสดงรายละเอียดการตั้งค่าเพื่อดูสถานะการทำงานของ server**

เมื่อตั้งค่าสำเร็จจะสามารถดูสถานะของ server ได้ โดยสามารถดูได้จาก Web Browser ดังภาพที่ 3.8



**ภาพประกอบที่ 4.2 แสดงรายละเอียดการข้อมูลสถานะการทำงานของ server**

ซึ่งทางผู้จัดทำมีได้นำข้อมูลสถานะที่น่าสนใจเพื่อใช้ตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ ดังนี้ Total Accesses, Total kBytes, CPU System, CPU Load, Uptime, RequsetPerSecond, Bytes PerSecond, BytesPerRequest และ DurationPerRequest โดยใช้ภาษา PHP ในการดึงข้อมูลสถานะและเก็บข้อมูลในรูปแบบ JSON เพื่อสะดวกในการใช้ข้อมูล ดังภาพประกอบที่ 4.3 และ 4.4 ตามลำดับ

Text

Description automatically generated

**ภาพประกอบที่ 4.3 การดึงข้อมูลจากเว็บเซิร์ฟเวอร์และเก็บในรูปแบบ JSON**

Text, letter

Description automatically generated

**ภาพประกอบที่ 4.4 ข้อมูลที่ถูกจัดเก็บในรูปแบบ JSON**

## 4.2 การแสดงกราฟข้อมูลสถานะ

การแสดงผลของข้อมูลสถานะที่ผู้จัดทำในรูปแบบของกราฟเส้นตรง โดยเลือกใช้ Chart.js ซึ่งเป็น Javascript library สำหรับการสร้างกราฟเพื่อรายงานข้อมูลสถานะ ซึ่งใช้คู่กับ Vue.js ที่เป็น Javascript Framework ในการเขียน Code แสดงผลกราฟข้อมูลและโครงสร้าง รวมถึงใช้ Axois เป็น HTTP client libraryสำหรับดึงข้อมูลสถานะมาใช้สำหรับทำรายงานข้อมูล

Chart, histogram

Description automatically generated

**ภาพประกอบที่ 4.4 รายงานข้อมูลของ Total Accesses ที่แสดงในรูปแบบกราฟเส้น**

## 4.3 การตรวจสอบข้อมูลสถานะ

## 4.4 แจ้งเตือนสถานะไปยัง Line Notify

ในส่วนนี้ผู้จัดทำมีความต้องการที่จะแจ้งเตือนไปยังผู้ดูแล หากมีข้อสถานะที่ผิดปกติ จึงเลือกใช้ Line Notify สำหรับการแจ้งเตือน

**บทที่ 5**

**บทสรุปและข้อเสนอแนะ**

จากการพัฒนาระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ สามารถสรุป ชี้แจงปัญหาและอุปสรรค ข้อจำกัดของระบบ และข้อเสนอแนะ ได้ดังนี้

## 5.1 บทสรุป

โครงงานฉบับนี้ นำเสนอเกี่ยวกับระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ เนื่องจากในปัจจุบันพบว่าแนวโน้มเทคโนโลยีด้านการรักษาความปลอดภัยไซเบอร์ถูกจับตามองมากขึ้น เพราะมีความเสี่ยงที่จะถูกโจมตีจากผู้ไม่หวังดี และมีรายงานการโจมตีโดยวิธี HTTP Flood อยู่บ่อยครั้ง โดยในโครงงานเล่มนี้จะกล่าวถึงระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ ซึ่งสามารถช่วยให้ผู้ดูแลระบบสามารถทราบถึงข้อมูลสถานะของเว็บเซิร์ฟเวอร์ ในรูปแบบเว็ปไซต์เพื่อเพิ่มประสิทธิภาพในการตรวจสอบสถานะ และนำข้อมูลสถานะไปวิเคราะห์เพื่อตรวจสอบความผิดปกติของข้อมูล รวมถึงมีการแจ้งเตือนไปยังผู้ดูแลได้อย่างรวดเร็วผ่าน Line Notify เมื่อมีค่าข้อมูลสถานะที่ผิดปกติ หรือมีความเสี่ยงว่าถูกโจมตี จากการพัฒนาพบว่า

## 5.2 ปัญหาและแนวทางแก้ปัญหา

1) เป็นการพัฒนาระบบครั้งแรกของผู้จัดทำจึงต้องใช้เวลาในการทำความเข้าใจภาษาและเครื่องมือต่หาง ๆ

2) ความรู้ในทางด้านการวิเคราะห์ข้อมูลมีน้อยทำให้มีตัวเลือกสมการในการวิเคราะห์น้อย

## 5.3 ข้อเสนอแนะ

1) ควรออกแบบระบบให้รองรับเว็บเซิร์ฟเวอร์ที่หลากหลาย

2)เพื่มความสามารถในการตรวจจับที่มีประสิทธิ์ภาพมากขึ้น

# เอกสารอ้างอิง

[1] สุรศักดิ์ กาบเงิน. เว็บเซิร์ฟเวอร์ (Web Server). [ออนไลน์]. เข้าถึงได้จาก: เว็บเซิร์ฟเวอร์ (Web Server) – Elearningsurasak (wordpress.com) (วันที่สืบค้นข้อมูล: 16 มกราคม 2564)

[2] ศุภกิจ อรรณพพรชัย. Apache Webserver คืออะไร โปรแกรมรองรับคำขอเพื่อเชื่อมต่อ application. [ออนไลน์]. เข้าถึงได้จาก: Apache Webserver คืออะไร โปรแกรมรองรับคำขอเพื่อเชื่อมต่อ application - Saixiii (วันที่สืบค้นข้อมูล: 16 มกราคม 2564)

[3] “An In-Depth Guide to Web Server Monitoring”. [ออนไลน์]. เข้าถึงได้จาก: https://blog.appoptics.com/an-in-depth-guide-to-web-server-monitoring/ (วันที่สืบค้นข้อมูล: 16 มกราคม 2564)

[4] “Web Monitoring”[ออนไลน์]. เข้าถึงได้จาก: https://www.catcyfence.com/it-security/services/web-monitoring/ (วันที่สืบค้นข้อมูล: 20 มกราคม 2564)

[5] “How to Monitor Apache Web Server Load and Page Statistics” [ออนไลน์]. เข้าถึงได้จาก:https://www.tecmint.com/monitor-apache-web-server-load-and-page-statistics/ (วันที่สืบค้นข้อมูล: 13 มีนาคม 2564)

[6] “Basic Http Request part 1” [ออนไลน์]. เข้าถึงได้จาก: https://rithikied.medium.com/basic-http-request-part-1-4d5dbf62272/ (วันที่สืบค้นข้อมูล: 14 มีนาคม 2564)

[7] “What is Server Monitoring?” [ออนไลน์]. เข้าถึงได้จาก: https://www.manageengine.com/network-monitoring/server-monitoring.html (วันที่สืบค้นข้อมูล: 15 มีนาคม 2564)

[8] “CPU load average คืออะไร” [ออนไลน์]. เข้าถึงได้จาก: https://thiti.dev/blog/6200/ (วันที่สืบค้นข้อมูล: 12 กันยายน 2564)

[9] “CPU Load คืออะไร แตกต่างกับ CPU Usage % อย่างไร...แล้วทำไม zabbix template จึงไม่ทำ CPU Usage” [ออนไลน์]. เข้าถึงได้จาก: https://blog.zabbixinthailand.com/2019/03/cpu-load-cpu-usage-zabbix-template-cpu.html (วันที่สืบค้นข้อมูล: 12 กันยายน 2564)

**ภาคผนวก ก**

**โปสเตอร์**

**ภาคผนวก ข**

**บทความวิชาการ**

**ระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์Web Server Performance Monitoring System**

เจฟฟี่ ดาโต๊ะ ภาสวี สุวรรณโณ และ นิธิ ทะนนท์

สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสงขลานครินทร์

E-mail: 6110210071@email.psu.ac.th, 6110210608@psu.ac.th, nithi.t@psu.ac.th

บทคัดย่อ

ในปัจจุบันแนวโน้มเทคโนโลยีด้านการรักษาความปลอดภัยไซเบอร์ถูกจับตามองมากขึ้น เพราะมีความเสี่ยงที่จะถูกโจมตีจากผู้ไม่หวังดี และมีรายงานการโจมตีโดยวิธี HTTP Flood อยู่บ่อยครั้ง โดยในโครงงานเล่มนี้จะกล่าวถึงระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ ซึ่งสามารถช่วยให้ผู้ดูแลระบบสามารถทราบถึงข้อมูลสถานะของเว็บเซิร์ฟเวอร์ ในรูปแบบเว็ปไซต์เพื่อเพิ่มประสิทธิภาพในการตรวจสอบสถานะ และนำข้อมูลสถานะไปวิเคราะห์เพื่อตรวจสอบความผิดปกติของข้อมูล รวมถึงมีการแจ้งเตือนไปยังผู้ดูแลได้อย่างรวดเร็วผ่าน Line Notify เมื่อมีค่าข้อมูลสถานะที่ผิดปกติ หรือมีความเสี่ยงว่าถูกโจมตี จากการพัฒนาพบว่า

Abstract

Nowadays, cybersecurity technology trends are progressively being scrutinized these days. Because there's a chance, you'll be attacked by someone with nefarious intentions. HTTP Flood assaults are also a common occurrence. We will discuss a web server performance monitoring system in this project. This can assist administrators in determining the web server's status. in the form of a webpage to make checking the status more efficient Furthermore, the status data is evaluated to look for any data anomalies. Quickly sending alerts to careers via Line Notify When there is an abnormal status information or are in risk of being attacked

1. บทนำ

HyperText Transfer Protocol (HTTP) คือ โพรโทคอลในระดับชั้นโปรแกรมประยุกต์เพื่อการแจกจ่ายและการทำงานร่วมกันกับสารสนเทศของสื่อผสม ใช้สำหรับการรับทรัพยากรที่เชื่อมโยงกับภายนอก ซึ่งนำไปสู่การจัดตั้งเวิลด์ไวด์เว็บ (world wide web) และ เว็บเซิร์ฟเวอร์ (Web Server)

หน่วยงานต่างๆ มีการติดตั้งเว็บเซิร์ฟเวอร์ (Web Server) เป็นของตนเองเพื่อใช้ในการเผยแพร่ข้อมูลข่าวสาและประชาสัมพันธ์หน่วยงาน เนื่องจากเว็บเซิร์ฟเวอร์สามารถเข้าถึงได้จากบุคคลทั่วไปซึ่งเปิดโอกาศให้ผู้ที่ไม่หวังดีสามารถทำการโจมตีเว็บเซิร์ฟเวอร์จนไม่สามารถให้บริการได้อย่างเต็มประสิทธิภาพ ดังนั้นทางผู้วิจัยจึงมีแนวคิดในการหาทางป้องกันปัญหาดังกล่าว โดยพัฒนาระบบสำหรับใช้ตรวจสอบการโจมตีเว็บเซิร์ฟเวอร์เพื่อให้สามารถป้องกันการโจมตีได้อย่างรวดเร็วบื้องต้นได้

**2.1 ทฤษฎีหลักการที่เกี่ยวข้อง**

**2.1.1 Web Server**

เว็บเซิร์ฟเวอร์ (Web Server) คือ เครื่องคอมพิวเตอร์ ที่ทำการติดตั้ง Software เกี่ยวกับ Server ทำหน้าที่บริการและส่งข้อมูลให้กับ Client Request ข้อมูล และ Web Server จะ Response เพื่อส่งผลลัพธ์ไปยัง Client โดย Web Server ทั่ว ๆ ไปจะติดตั้ง Software ที่ชื่อว่า IIS (Internet Information Services) และ Apache โดย Service เหล่านี้จะบริการข้อมูลผ่าน Port : 80 และรับส่งข้อมูลผ่าน Web Browser

**2.1.3 Web Monitoring**

การตรวจสอบการทำงานของเว็บไซต์ว่ามีการทำงานอยู่ หรือหยุดทำงานไป (up/down- time) การเปลี่ยนแปลงหน้าเว็บไซต์ (Defacement) และตรวจสอบสิ่งผิดปกติบนเว็บไซต์ นอกจากนี้ยังสามารถแจ้งเตือนไปยังผู้ดูแลระบบได้ผ่านช่องทางกำหนด เช่น SMS, Email อื่น ๆ

**2.1.4 Apache HTTP Server**

Apache HTTP Server เป็นซอฟต์แวร์เซิร์ฟเวอร์ที่ใช้กันอย่างแพร่หลาย Apache พัฒนาและดูแลโดย Apache Software Foundation ซึ่งเป็นซอฟต์แวร์โอเพนซอร์สที่สามารถใช้งานได้ฟรี โดยมีการใช้โดยรวมประมาณ 67% ของเว็บเซิร์ฟเวอร์ทั้งหมดในโลก ซึ่งรวดเร็วเชื่อถือได้และปลอดภัย สามารถปรับแต่งได้เพื่อตอบสนองความต้องการของสภาพแวดล้อมที่หลากหลาย โดย สามารถเพิ่ม function พิเศษที่เป็น module plugin ได้โดยง่าย

**2.1.5 Server Monitor**

การตรวจสอบประสิทธิภาพของเซิร์ฟเวอร์เป็นกระบวนการตรวจสอบทรัพยากรของเซิร์ฟเวอร์เช่น การใช้งาน CPU, การใช้งานหน่วยความจำ, การใช้งานดิสก์ เป็นต้น โดยการตรวจสอบประสิทธิภาพของเซิร์ฟเวอร์จะใช้งานซอฟต์แวร์สำหรับตรวจสอบการทำงาน ซึ่งช่วยให้สามารถตรวจสอบเซิร์ฟเวอร์ได้อัตโนมัติ และยังเป็นกระบวนการที่ช่วยวางแผนการใช้งานเซิร์ฟเวอร์ นอกจากนี้การตรวจสอบประสิทธิภาพของเซิร์ฟเวอร์ยังช่วยในการระบุปัญหาอื่น ๆที่เกี่ยวข้องกับประสิทธิภาพ เช่น การตอบสนองการให้บริการ,การใช้งานทรัพยากร และอื่น ๆ

**2.1.6 HTTP**

เป็น โปรโตคอล ที่ใช้งานสำหรับเผยแพร่ข้อมูล และ เป็นสื่อการสำหรับการสื่อสารระหว่าง web browser กับ web server อีกทั้งเป็นจุดกำเนิดของ World Wide Web ซึ่งมีโครงสร้างเป็นตัวอักษรและตัวเลข (text) ใช้สำหรับเป็น link เชื่อมระหว่าง ข้อมูล text อื่น ๆ และถูกใช้ในการแลกเปลี่ยน ข้อมูลในรูปแบบ multimedia สามารถเเรียกใช้งานผ่าน web browser เช่น Firefox, Google Chrome, Safari, Opera และ Microsoft Internet Explorer ซึ่งจะไปทำการดึงและแลกเปลี่ยนข้อมูลกับทาง Server ที่ระบุใน URL ข้อมูลที่ส่งไปจะอยู่ในรูป plain text ไม่มีการเข้ารหัส ทำให้สามารถถูกดักจับและอ่านข้อมูลได้ง่าย โดยส่ง request ไปและรอจนกว่าจะได้ response จากเครื่องแม่ข่ายกลับมาจึงไม่ปลอดภัย

**2.1.7** Mod Status

mod\_status เป็นโมดูล Apache ที่ช่วยตรวจสอบการโหลดของเว็บเซิร์ฟเวอร์และการเชื่อมต่อ http ในปัจจุบันด้วยอินเทอร์เฟซ HTML ที่สามารถเข้าถึงได้ผ่านเว็บเบราว์เซอร์ และ mod\_status ของ Apache นั้นแสดงหน้า HTML ที่มีข้อมูลเกี่ยวกับสถิติปัจจุบันของเว็บเซิร์ฟเวอร์

**2.1.8 Line Notify**

LINE Notify คือ บริการที่คุณสามารถได้รับข้อความแจ้ง เตือนจากเว็บเซอร์วิสต่าง ๆ ที่คุณสนใจได้ทาง LINE โดย หลังเสร็จสิ้นการเชื่อมต่อกับทางเว็บเซอร์วิสแล้ว คุณจะ ได้รับการแจ้งเตือนจากบัญชีทางการของ “LINE Notify” ซึ่งให้บริการโดย LINE นั่นเอง คุณสามารถ เชื่อมต่อกับบริการที่หลากหลาย และยังสามารถรับ การ แจ้งเตือนทางกลุ่มได้อีกด้วย ซึ่งบริการหลักๆ ที่สามารถ เชื่อมต่อได้แก่ GitHub, IFTTT เป็นต้น

**3. การวิเคราะห์และออกแบบระบบ**

การออกแบบและสถาปัตยกรรมของระบบตรวจสอบประสิทธิภาพเว็บ

เซิร์ฟเวอร์

**3.1 สถาปัตยกรรมของระบบตรวจสอบประสิทธิภาพเว็บเซิร์ฟเวอร์**

Diagram, waterfall chart

Description automatically generated

รูปที่ 3.1 สถาปัตยกรรมของระบบตรวจสอบประสิทธิภาพเว็บเซิร์ฟเวอร์

จากภาพประกอบที่ 3.1 สามารถจำแนกส่วนประกอบหลักของระบบตรวจสอบประสิทธิภาพเว็บ

เซิร์ฟเวอร์ออกเป็น 4 ส่วน คือ (1) Attacker, (2) Web Server, (3) Performance Monitoring System

และ (4) Administrator โดยแต่ละส่วนมีลักษณะการทำงานดังนี้

**3.1.1 Attacker**

Attacker จะทำการโจมตีไปยัง Web Server โดยใช้โปรแกรม JMeter ซึ่งเป็นเครื่องมือจาก Apache ด้วยรูปแบบ HTTP Flood Attack โดยทำการสร้าง HTTP Request จำนวนมากส่งเข้าไปยังWeb Server เป้าหมาย หากทำการโจมตีสำเร็จ Web Server จะไม่สามารถให้บริการแก่ผู้ใช้งานระบบอื่น ๆ ที่ต้องการใช้บริการจริง

**3.1.2 Web Server**

Web Server คือ เครื่องคอมพิวเตอร์ที่ทำการติดตั้งโปรแกรม Apache Web Server โดยเปิดใช้ mod\_status ซึ่งเป็นโมดูลสำหรับแสดงสถานะการทำงานในรูปแบบเอกสาร HTML

**3.1.3 Performance Monitoring System**

Text, letter

Description automatically generated

รูปที่ 3.1.3 แสดงละเอียดข้อมูล

**3.1.4 Administrator**

Administrator คือ ผู้ดูแลระบบสามารถเรียกดูสถานะการทำงานของ Web Server ในรูปแบบกราฟ ดังแสดงในภาพประกอบที่ 3.4 และ 3.5 ในกรณีที่ Web Server ถูกโจมตีจะมีการแจ้งเตือนผ่าน Line Notify ดังแสดงในภาพประกอบที่ 3.6

Chart, line chart

Description automatically generated

รูปที่ 3.1.4.1 แสดงข้อมูลค่าสถานะ

Graphical user interface, text, application

Description automatically generated

รูปที่ 3.1.4.2 แสดงการแจ้งเตือนผ่าน Line Notify

**4. ผลการพัฒนาระบบ**

**4.1 การนำเข้าข้อมูลสถานะจากเว็บเซิร์ฟเวอร์**

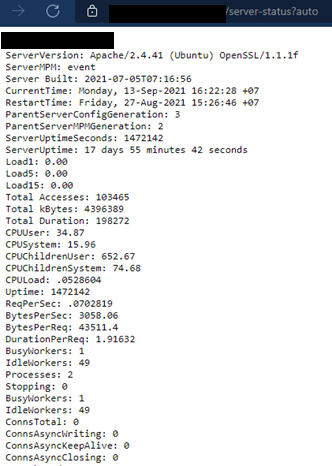
ระบบตรวจวัดอุณหภูมิและความชื้น DHT11 Sensor ใช้ในการตรวจจับ คือ อุณหภูมิและความชื้น โดยทางผู้จัดทำได้การนำเข้าข้อมูลสถานะจาก Apache Server Status ซึ่งจำเป็นต้องเปิดและตั้งค่า Apache Extended Status เพื่อให้สามารถดูสถานะการทำงานของ Apache ได้

Text

Description automatically generated

รูปที่ 4.1.1 แสดงรายละเอียดการตั้งค่าเพื่อดูสถานะการทำงานของ server

เมื่อตั้งค่าสำเร็จจะสามารถดูข้อมูลสถานะของ server ได้ โดยสามารถดูได้จาก Web Browser



รูปที่ 4.1.2 แสดงรายละเอียดการข้อมูลสถานะการทำงานของ server

ซึ่งทางผู้จัดทำมีได้นำข้อมูลสถานะที่น่าสนใจเพื่อใช้ตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ ดังนี้ Total Accesses, Total kBytes, CPU System, CPU Load, Uptime, RequsetPerSecond, Bytes PerSecond, BytesPerRequest และ DurationPerRequest โดยใช้ภาษา PHP ในการดึงข้อมูลสถานะและเก็บข้อมูลในรูปแบบ JSON เพื่อสะดวกในการใช้ข้อมูล

Text, letter

Description automatically generated

รูปที่ 4.1.3 ข้อมูลที่ถูกจัดเก็บในรูปแบบ JSON

**4.2 การแสดงกราฟข้อมูลสถานะ**

การแสดงผลของข้อมูลสถานะที่ผู้จัดทำในรูปแบบของกราฟเส้นตรง โดยเลือกใช้ Chart.js ซึ่งเป็น Javascript library สำหรับการสร้างกราฟเพื่อรายงานข้อมูลสถานะ ซึ่งใช้คู่กับ Vue.js ที่เป็น Javascript Framework ในการเขียน Code แสดงผลกราฟข้อมูลและโครงสร้าง รวมถึงใช้ Axois เป็น HTTP client libraryสำหรับดึงข้อมูลสถานะมาใช้สำหรับทำรายงานข้อมูล

Chart, histogram

Description automatically generated

รูปที่ 4.2.1 รายงานข้อมูลของ Total Accesses ที่แสดงในรูปแบบกราฟเส้น

**4.3 การตรวจสอบข้อมูลสถานะ**

**4.4 แจ้งเตือนสถานะไปยัง Line Notify**

ในส่วนนี้ผู้จัดทำมีความต้องการที่จะแจ้งเตือนไปยังผู้ดูแล หากมีข้อสถานะที่ผิดปกติ จึงเลือกใช้ Line Notify สำหรับการแจ้งเตือน

**5. การทดสอบการใช้งาน**

จากการทดสอบข้างต้น NodeMCU สามารถส่งข้อมูลไปยังFirebaseมีการส่งข้อมูลมายัง NodeMCU ได้และมีการส่งแจ้งเตือนไปยังLine Notify ซึ่งผลการทดสอบนี้ตรงตามวัตถุประสงค์ของผู้จัดทำที่กำหนดไว้

**6. บทสรุป**

โครงงานฉบับนี้ นำเสนอเกี่ยวกับระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ เนื่องจากในปัจจุบันพบว่าแนวโน้มเทคโนโลยีด้านการรักษาความปลอดภัยไซเบอร์ถูกจับตามองมากขึ้น เพราะมีความเสี่ยงที่จะถูกโจมตีจากผู้ไม่หวังดี และมีรายงานการโจมตีโดยวิธี HTTP Flood อยู่บ่อยครั้ง โดยในโครงงานเล่มนี้จะกล่าวถึงระบบตรวจสอบประสิทธิภาพของเว็บเซิร์ฟเวอร์ ซึ่งสามารถช่วยให้ผู้ดูแลระบบสามารถทราบถึงข้อมูลสถานะของเว็บเซิร์ฟเวอร์ ในรูปแบบเว็ปไซต์เพื่อเพิ่มประสิทธิภาพในการตรวจสอบสถานะ และนำข้อมูลสถานะไปวิเคราะห์เพื่อตรวจสอบความผิดปกติของข้อมูล รวมถึงมีการแจ้งเตือนไปยังผู้ดูแลได้อย่างรวดเร็วผ่าน Line Notify เมื่อมีค่าข้อมูลสถานะที่ผิดปกติ หรือมีความเสี่ยงว่าถูกโจมตี จากการพัฒนาพบว่า

**6.1 แนวทางการพัฒนาต่อ**

- ควรออกแบบระบบให้รองรับเว็บเซิร์ฟเวอร์ที่หลากหลาย

- เพื่มความสามารถในการตรวจจับที่มีประสิทธิ์ภาพมากขึ้น

**7. กิตติกรรมประกาศ**

โครงงานฉบับนี้สำเร็จได้ด้วยความช่วยเหลือและสนับสนุนจากบุคคลหลายฝ่าย ผู้จัดทำรู้สึกซาบซึ้งและขอขอบพระคุณอย่างสูงมา ณ โอกาสนี้คือ

ผู้ช่วยศาสตราจารย์ นิธิ ทะนนท์ อาจารย์ที่ปรึกษาโครงงาน ที่กรุณาให้ความรู้ คำปรึกษาแนะนำ และช่วยเหลือในการแก้ปัญหาต่างๆให้แก่ผู้จัดทำเสมอมา รวมถึงถ่ายทอดหลักการแนวคิดในการพัฒนาตัวเองและคุณธรรมสำหรับวิชาชีพแก่ผู้จัดทำมาโดยตลอด

ผู้ช่วยศาสตราจารย ดร.ชินพงศ์ อังสุโชติเมธี กรรมการสอบโครงงาน ที่ให้คำปรึกษา และกรุณาช่วยตรวจทานและแก้ไขโครงงานให้มีความสมบูรณ์

อาจารย์สมศักดิ์ คงแสง กรรมการสอบโครงงาน ที่ให้คำปรึกษา และกรุณาช่วยตรวจทานและแก้ไขโครงงานให้มีความสมบูรณ์

นอกจากนี้ยังต้องขอขอบคุณหลักสูตร วท.บ. วิทยาการคอมพิวเตอร์ สาขาวิทยาศาสตร์การคำนวณ ที่เอื้อเฟื้อสถานที่และอุปกรณ์ในการทำโครงงานเพื่อนๆ ภาควิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ ที่ให้คำปรึกษา และช่วยเหลือในการทำโครงงาน

ผู้จัดทำขอขอบพระคุณทุกท่านเป็นอย่างสูง ณ โอกาสนี้ด้วย สุดท้ายนี้ หากโครงงานชิ้นนี้มีข้อผิดพลาดประการใด ผู้พัฒนาขออภัยมา ณ ที่นี้และจะน้อมรับทุกคำติชมมาพัฒนาโครงงานให้ดียิ่งขึ้นในอนาคตต่อไป

**8. เอกสารอ้างอิง**

[1] สุรศักดิ์ กาบเงิน. เว็บเซิร์ฟเวอร์ (Web Server). [ออนไลน์]. เข้าถึงได้จาก: เว็บเซิร์ฟเวอร์ (Web Server) – Elearningsurasak (wordpress.com) (วันที่สืบค้นข้อมูล: 16 มกราคม 2564)

[2] ศุภกิจ อรรณพพรชัย. Apache Webserver คืออะไร โปรแกรมรองรับคำขอเพื่อเชื่อมต่อ application. [ออนไลน์]. เข้าถึงได้จาก: Apache Webserver คืออะไร โปรแกรมรองรับคำขอเพื่อเชื่อมต่อ application - Saixiii (วันที่สืบค้นข้อมูล: 16 มกราคม 2564)

[3] “An In-Depth Guide to Web Server Monitoring”. [ออนไลน์]. เข้าถึงได้จาก: https://blog.appoptics.com/an-in-depth-guide-to-web-server-monitoring/ (วันที่สืบค้นข้อมูล: 16 มกราคม 2564)

[4] “Web Monitoring”[ออนไลน์]. เข้าถึงได้จาก: https://www.catcyfence.com/it-security/services/web-monitoring/ (วันที่สืบค้นข้อมูล: 20 มกราคม 2564)

[5] “How to Monitor Apache Web Server Load and Page Statistics” [ออนไลน์]. เข้าถึงได้จาก:https://www.tecmint.com/monitor-apache-web-server-load-and-page-statistics/ (วันที่สืบค้นข้อมูล: 13 มีนาคม 2564)

[6] “Basic Http Request part 1” [ออนไลน์]. เข้าถึงได้จาก: https://rithikied.medium.com/basic-http-request-part-1-4d5dbf62272/ (วันที่สืบค้นข้อมูล: 14 มีนาคม 2564)

[7] “What is Server Monitoring?” [ออนไลน์]. เข้าถึงได้จาก: https://www.manageengine.com/network-monitoring/server-monitoring.html (วันที่สืบค้นข้อมูล: 15 มีนาคม 2564)

[8] “CPU load average คืออะไร” [ออนไลน์]. เข้าถึงได้จาก: https://thiti.dev/blog/6200/ (วันที่สืบค้นข้อมูล: 12 กันยายน 2564)

[9] “CPU Load คืออะไร แตกต่างกับ CPU Usage % อย่างไร...แล้วทำไม zabbix template จึงไม่ทำ CPU Usage” [ออนไลน์]. เข้าถึงได้จาก: https://blog.zabbixinthailand.com/2019/03/cpu-load-cpu-usage-zabbix-template-cpu.html (วันที่สืบค้นข้อมูล: 12 กันยายน 2564)