โปรแกรมรวบรวมและจัดการไฟล์ล็อกเพื่อการรักษาความปลอดภัย

SECURITY LOG CONSILIDATION AND MANAGEMENT PROGRAM

ธีรวุฒิ มงคลสหกุล

ธีระพงษ์ วรรณสิงห์

วิธวัช สุวรรณฤทธิ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

ปริญญานิพนธ์ปีการศึกษา 2551

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมรวบรวมและจัดการไฟล์ล็อกเพื่อการรักษาความปลอดภัย

Security Log Consolidation and Management Program

ผู้จัดทำ

1. นายธีรวุฒิ มงคลสหกุล รหัสนักศึกษา 49015280
2. นายธีระพงษ์ วรรณสิงห์ รหัสนักศึกษา 49015281
3. นายวิธวัช สุวรรณฤทธิ์ รหัสนักศึกษา 49015299

_____อาจารย์ที่ปรึกษา

(อาจารย์อัครเดช วัชระภูพงษ์)

_____อาจารย์ที่ปรึกษา

(ผู้ช่วยศาสตราจารย์ธนา หงษ์สุวรรณ)

_____อาจารย์ที่ปรึกษา

(อาจารย์ธนัญชัย ตรีภาค)

โปรแกรมรวบรวมและจัดการไฟล์ล็อกเพื่อการรักษาความปลอดภัย

นายธีรวุฒิ	มงคลสหกุล	49015280
นายธีระพงษ์	วรรณสิงห์	49015281
นายวิธวัช	สุวรรณฤทธิ์	49015299
อาจารย์อัครเดช	วัชระภูพงษ์	อาจารย์ที่ปรึกษา
ผศ.ธนา	หงษ์สุวรรณ	อาจารย์ที่ปรึกษาร่วม
อาจารย์ธนัญชัย	ตรีภาค	อาจารย์ที่ปรึกษาร่วม

ปีการศึกษา 2551

บทคัดย่อ

โครงงานนี้เป็นโครงงานที่มุ่งเน้นศึกษาการจัดเก็บไฟล์ล็อก ซึ่งเป็นไฟล์ที่สำคัญในระบบปฏิบัติการยูนิกซ์และลีนุกซ์ซึ่งมีไว้สำหรับตรวจเช็คความปลอดภัยหรือความผิดพลาดของระบบ โดยมีประเด็นในการจัดเก็บที่อ้างอิงกับ พรบ.ว่าด้วยกระทำผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 เพื่อให้ตรงตามที่กฎหมายกำหนด และได้ทำการสร้างระบบจัดเก็บไฟล์ล็อกกลางเพื่อรวบรวมล็อกไฟล์

การพัฒนาโปรแกรมรวบรวมและจัดการไฟล์ล็อกเพื่อการรักษาความปลอดภัยนี้ มุ่งเน้นที่ การรับไฟล์ล็อกจากระบบอื่นๆมา แล้วทำการจัดเก็บอย่างปลอดภัยโดยทำการควบคุมการเข้าถึงไฟล์ล็อก ทั้งนี้ยังสามารถตรวจสอบความผิดปกติของไฟล์ล็อกที่ได้ทำการจัดเก็บ เพื่อให้ผู้ดูแลระบบทราบว่าระบบของตนปลอดภัยหรือไม่

SECURITY LOG CONSOLIDATION AND MANAGEMENT PROGRAM

Mr. Teerawut	Mongkolsahakul	49015280
Mr. Teerapong	Wannasing	49015281
Mr. Viratus	Suwannarid	49015299
Mr. Akkradach	Watcharapupong	Advisor
Asst. Prof. Thana	Hongsuwan	Co-Advisor
Mr. Tanunchai	Tripak	Co-Advisor

Academic Year 2008

ABSTRACT

This project aims at reseach about keeping log file. It is an important file for security and error checking in operation system Unix and Linux. The main point in this project is refer to computer crime act 2007. So that it abides by the law and central unit is created to keep all the log files.

The development of our program purpose concentrates on receiving logs file from other system and store it safely by controlling access to the log file. Whereby the log files which are kept can be checked for errors so the administrator can ensure that their system is safe.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้จะไม่สามารถเสร็จสมบูรณ์ได้ถ้าไม่ได้รับคำแนะนำ คำเตือนทั้งหลายจาก อาจารย์อัครเดช วัชระภูพงษ์ ผู้ช่วยศาสตราจารย์ธนา หงษ์สุวรรณ และอาจารย์ธนัญชัย ตรีภาค คณะผู้จัดทำขอขอบพระคุณอย่างยิ่งสำหรับทุกสิ่งที่อย่างที่ได้รับจากท่านทั้งสาม

นอกจากนี้ขอขอบคุณสถาบัน ภาควิชาวิศวกรรมคอมพิวเตอร์ และห้องวิจัยและพัฒนาการรักษาความปลอดภัยข้อมูล (ISAG) ที่ได้เอื้อเฟื้อสถานที่ให้คณะผู้จัดทำได้ทำการวิจัยและศึกษา ขอขอบคุณเพื่อนๆ พี่ ๆ ห้อง ISAG ที่ได้ให้คำแนะนำและให้ความช่วยเหลือในยามที่ผู้จัดทำพบกับปัญหาได้ดีเสมอมา

สุดท้ายต้องขอขอบคุณบิดา มารดาที่ได้ให้กำเนิด คอยสั่งสอน ให้การสนับสนุนการศึกษาและเป็นกำลังใจในการศึกษาเล่าเรียนเสมอ นับเป็นพระคุณอย่างยิ่ง

นายธีรวุฒิ มงคลสหกุล
นายธีระพงษ์ วรรณสิงห์
นายวิธวัช สุวรรณฤทธิ์

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญรูป

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

สารบัญตาราง

ตารางที่ **หน้า**

บทที่ 1
บทนำ

1.1 บทนำ

ในปัจจุบันการติดต่อสื่อสารทางระบบเนตเวิร์คได้มีความสำคัญต่อการดำเนินชีวิตของเรา เพิ่มมากขึ้นไม่ว่าจะเป็นการสื่อสารผ่านรูปแบบต่างๆ เช่น อีเมลล์ (E-Mail), เอ็มเอสเอ็น (MSN), คิวคิว (QQ), ยะฮู (Yahoo), ไฟร์ฟล็อก (FireFox) เป็นต้นหรือจะเป็นการโพสข้อความ เช่น ไฮไฟ (hi5) เว็บบรอด (Webboard) เป็นต้น ซึ่งการใช้งานโปรแกรมหรือบริการดังกล่าวอาจจะเกิดการกระทำ พรบ.ว่าด้วยการกระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ.2550 ขึ้นได้ ดังนั้นหากมีการกระทำผิดเกิดขึ้นและมีผู้เสียหาย อาจทำให้ไม่สามารถหาตัวผู้ที่กระทำผิดมาลงโทษได้เนื่องจากว่าภายในระบบเนตเวิร์คย่อมมีผู้ใช้งานเป็นจำนวนมากเข้ามาใช้งาน การที่จะหาบุคคลที่กระทำผิดจึงค่อนข้างจะหาตัวได้ยาก จากปัญหาที่กล่าวมานี้หากเรามีการเก็บ ไฟล์ล็อก (FileLog) จะทำให้เราสามารถทราบถึงที่มาของการเกิดการกระทำความผิดของบริการนั้นๆ ว่ามาจากที่ใด เวลาที่เกิดเมื่อไหร่ มีต้นทาง ปลายทางมาจากที่ไหน ใช้บริการอะไร เป็นต้น ซึ่งจะส่งผลดีต่อผู้ที่เสียหายที่จะใช้ข้อมูลนี้เป็นตัวหาผู้กระทำผิดต่อไป

โปรแกรมต้นแบบที่สร้างขึ้นนี้ เป็นโปรแกรมรวบรวมและจัดการไฟล์ล็อกเพื่อการรักษาความปลอดภัย (Security Log consolidation and Management Program) เป็นการพัฒนาโปรแกรมเพื่อรวบรวมและจัดการบริหารล็อกไฟล์ (บันทึกล็อกไฟล์) ที่เกิดจากเหตุการณ์ต่างๆ และแนวทางการตอบสนองเมื่อเกิดเหตุการละเมิดความปลอดภัย โดยมีซิสล็อก (Syslog) เป็นโปรโตคอลหลักที่ใช้ในการบันทึกล็อกไฟล์ต่างๆที่เกิดจากการบุกรุก การโจมตีระบบ หรือเหตุการณ์อื่นๆอันไม่พึงประสงค์ เพื่อให้สามารถบริหารและจัดการล็อกไฟล์ได้อย่างมีประสิทธิภาพ สะดวก และตรงตาม พรบ.ว่าด้วยการกระทำผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 โปรแกรมรวบรวมและจัดการไฟล์ล็อกเพื่อการรักษาความปลอดภัย ประกอบด้วยส่วนต่างๆ ซึ่งแบ่งเป็นสองส่วน คือ ส่วนแรกเซิฟเวอร์สำหรับบันทึกและจัดเก็บเหตุการณ์ (Central Log Server) เซิฟเวอร์สำหรับบันทึกเหตุการณ์ที่เกิดการบุกรุก การโจมตีระบบ หรือเหตุการณ์อื่นๆอันไม่พึงประสงค์ ใช้เป็นตัวเก็บข้อมูลล็อกไฟล์ตาม พรบ.ว่าด้วยการกระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 ใช้เป็นตัวเก็บกฎของไฟล์วอลล์และเก็บล็อกไฟล์ที่ได้จากการเฝ้าดูพฤติกรรมของผู้บุกรุก ที่กระทำภายในเครื่อง โดยมีซิสล็อกเป็นกลไกที่ใช้ในการเก็บข้อมูลล็อกของ เคอร์เนล (kernel) และ แอฟพิเคชั่น (application) ของระบบ ใช้จำกัดสิทธิ์ในการเข้าดูและเปลี่ยนแปลงล็อกไฟล์ ใช้อ้างอิงเวลาเพื่อให้เวลาของระบบเครือข่ายตรงกันโดยใช้ เอ็นทีพี (NTP:Network Time Protocol) เป็นตัวจัดการ และอีกส่วนคือ โปรแกรมรวบรวมและบริหารจัดการล็อกไฟล์ (Log Management Program) เป็นโปรแกรมรวบรวมและ

บริหารจัดการล็อกไฟล์ซึ่งเป็นโปรแกรมส่วนกลางที่ควบคุม ดูแล และจัดการให้กับการทำงานของระบบรวบรวมและจัดการล็อกไฟล์ โดยทำงานผ่าน เว็บ บราวเซอร์ (Web Browser)

ซึ่งเมื่อเกิดเหตุการการบุกรุก การโจมตีระบบ หรือเหตุการณ์อื่นๆอันไม่พึงประสงค์ในระบบ บนระบบยูนิกซ์หรือลินุกซ์จะมี ซิสล็อก ดีลมอล (Syslog Daemon) ทำหน้าที่บันทึกเหตุการณ์ลงในล็อกไฟล์ตามที่ได้คอนฟิกไว้ แต่ล็อกไฟล์ดังกล่าวนั้นอาจถูกเปลี่ยนแปลงทั้งโดยเจตนาและไม่เจตนาจึงได้มีเซิพเวอร์สำหรับบันทึกเหตุการ์ณ์ไว้เพื่อเก็บรวบรวมล็อกไฟล์ และเวลาที่ต้องการบริหารและจัดการไฟล์ก็สามารถทำได้แต่ไม่สะดวกเท่าที่ควร โปรแกรมรวบรวมและบริหารจัดการล็อกไฟล์จะช่วยทำให้การรวบรวมและจัดการล็อกไฟล์ที่เกิดจากเหตุการณ์ต่างๆ สะดวกและง่ายขึ้น

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อศึกษาแนวทางการตอบสนองเมื่อเกิดการละเมิดความปลอดภัย

1.2.2 เพื่อศึกษาวิธีปกปิดพฤติกรรมของผู้บุกรุกระบบคอมพิวเตอร์

1.2.3 เพื่อสร้างต้นแบบโปรแกรมรวบรวมและจัดการล็อก

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1.3.1 สามารถรวบรวมและบันทึกเหตุการณ์ที่เกิดจากการบุกรุกหรือโจมตี หรือเหตุการณ์อันไม่พึงประสงค์

1.3.2 สามารถรวบรวมและบันทึกเหตุการณ์ที่เกิดจากบุกรุกหรือโจมตี หรือเหตุการณ์อันไม่พึงประสงค์จากคอมพิวเตอร์เครื่องอื่น หรืออุปกรณ์อื่นๆได้

1.3.3 สามารถจัดการล็อกไฟล์ที่ได้จากการบุกรุกหรือโจมตี หรือเหตุการณ์อันไม่พึงประสงค์ได้ง่ายขึ้น

1.3.4 สามารถจัดเก็บล็อกไฟล์ที่ได้จากการบุกรุกหรือโจมตี หรือเหตุการณ์อันไม่พึงประสงค์ได้ตรงตาม พรบ.ว่าด้วยการกระทำผิดเกี่ยวกับคอมพิวเตอร์ 2550

1.3.5 สามารถเฝ้าดูและบันทึกพฤติกรรมการบุกรุกของผู้บุกรุกเพื่อนำไปศึกษารูปแบบการบุกรุกใหม่ ๆ ต่อไปได้

1.3.6 ล็อกไฟล์ที่ได้จากการเก็บในส่วนของเครื่องเซิพเวอร์ เครื่องไคลเอ็นต์ หรืออุปกรณ์อื่นๆ ในระบบ มีความปลอดภัยและน่าเชื่อถือมากขึ้น

1.4 ขอบเขตของโครงงาน

1.4.1 ส่วนของการบันทึกเหตุการณ์ที่สามารถทำได้ตามจุดประสงค์ที่ต้องการ

1.4.2 ส่วนของ ล็อกเซิพเวอร์ สามารถรับบันทึกเหตุการณ์จากเครื่องเซิพเวอร์ เครื่องไคลเอ็นต์ หรืออุปกรณ์อื่นๆได้

1.4.3 ส่วนของการจำกัดสิทธิ์ ในการเข้าดูและเปลี่ยนแปลงข้อมูลล็อกไฟล์

1.4.4 ล็อกไฟล์ที่เก็บนั้นเป็นการเก็บแบบ Write Append Only

1.4.5 ส่วนของการอ้างอิงเวลา เอ็นทีพี ระหว่างเครื่องล็อกเซิพเวอร์กับเครื่องเซิพเวอร์ เครื่องไคลเอ็นต์ หรืออุปกรณ์อื่นๆในระบบเครือข่าย

1.4.6 ส่วนของการบันทึกเหตุการณ์ ตรงตามที่ พรบ.ว่าด้วยการกระทำผิดเกี่ยวกับคอมพิว-เตอร์ พ.ศ.2550 ที่ได้กำหนดไว้

1.4.7 สามารถควบคุมการทำงาน หรือการตั้งค่าต่างๆ ของซิสล็อกและล็อกไฟล์ที่ได้จากการบันทึก โดยโปรแกรมรวบรวมและจัดการล็อกไฟล์ ซึ่งมีลักษณะเป็นยูสเซอร์อินเตอร์เฟส (User Interface) เพื่อให้ง่ายต่อการใช้งานและการจัดการ โดยสามารถทำงานร่วมกับเว็บ บราวเซอร์ได้

1.5 เนื้อหาของรายงาน

ในส่วนของเนื้อหาในหนังสือเล่มนี้จะแบ่งออกเป็นบทๆ ซึ่งเป็นขั้นตอนในการศึกษาเพื่อใช้ทำโครงงานนี้ โดยในส่วนของบทที่แบ่งออกนั้นจะมีเนื้อหาในแต่ละบท มีเนื้อหาดังนี้

บทที่ 1 บทนำ แบ่งย่อยออกเป็นหัวข้อดังนี้

- บทนำ
- วัตถุประสงค์ของโครงงาน
- ประโยชน์ที่คาดว่าจะได้รับ
- ขอบเขตของโครงงาน
- เนื้อหาของรายงาน

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

- บทนำ
- หลักเกณฑ์การเก็บรักษาข้อมูลจราจรทางคอมพิวเตอร์ของผู้ให้บริการ
- ระบบจัดเก็บไฟล์ล็อกส่วนกลาง
- เอ็นทีพี (NTP:Network Time Protocol)
- การตรวจสอบความคงอยู่ (Integrity Checking)
- Internet Protocol Security (IPsec)

- ระบบควบคุมจัดการ

บทที่ 3 การออกแบบและพัฒนา

- บทนำ
- การออกแบบฮาร์ดแวร์
- การออกแบบซอฟต์แวร์

บทที่ 4 การทดลองและผลการทดลอง

- บทนำ
- รับ – ส่ง ไฟล์ล็อก
- ตัวอย่างการตรวจสอบการเปลี่ยนแปลงของไฟล์

บทที่ 5 บทวิจารณ์และสรุป

- วิเคราะห์และสรุปผลการทดลอง
- ปัญหาและอุปสรรค
- แนวทางการพัฒนาต่อ

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 บทนำ

ในการจะสร้างโปรแกรมต้นแบบซึ่งเป็นโปรแกรมรวบรวมและจัดการไฟล์ล็อกเพื่อการรักษาความปลอดภัย การศึกษาทางด้านทฤษฎีเป็นเรื่องที่สำคัญอย่างยิ่ง โดยเฉพาะเรื่องที่เกี่ยวกับการจัด-เก็บไฟล์ล็อกของระบบการบริหารและจัดการไฟล์ล็อกของระบบ การส่งข้อมูลผ่านระบบเนตเวิร์ค และยังมีกฎเกณฑ์ในการจัดการเกี่ยวกับการจัดเก็บไฟล์ล็อกที่เกี่ยวข้องกับ พรบ.ว่าด้วยการกระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 อีกด้วย

ทฤษฎีต่างๆที่เกี่ยวข้องจึงเป็นสิ่งจำเป็นในการทำโครงงานชิ้นนี้ จึงต้องศึกษาให้เข้าใจอย่างละเอียดเพื่อให้โครงงานนี้มีความสมบูรณ์ และตรงตามจุดประสงค์หลักของโครงงานมากที่สุด

2.2 หลักเกณฑ์การเก็บรักษาข้อมูลจราจรทางคอมพิวเตอร์ของผู้ให้บริการ

2.2.1 หลักการในการจัดเก็บล็อกไฟล์

เนื่องด้วย พรบ.ว่าด้วยการกระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 ได้มีการกำหนดให้มีการเก็บรักษาข้อมูลจราจรทางคอมพิวเตอร์และผู้ให้บริการต้องใช้วิธีการที่มั่นคงปลอดภัย ดังต่อไปนี้

2.2.1.1 **เก็บในสื่อมีเดีย (Media)**

เก็บในสื่อมีเดียที่สามารถรักษาความครบถ้วนถูกต้องแท้จริงอินทิกรีตี้ (Integrity) และการระบุตัวบุคคล (Identification) ที่เข้าถึงสื่อดังกล่าวได้

2.2.1.2 **มีระบบการเก็บรักษาความลับของข้อมูลที่จัดเก็บ**

กำหนดชั้นความลับในการเข้าถึงข้อมูล ดังกล่าวเพื่อรักษาความน่าเชื่อถือของข้อมูล และไม่ให้ผู้ดูแลระบบสามารถแก้ไขข้อมูลที่เก็บรักษาไว้ เช่น การเก็บไว้ในเซิพเวอร์สำหรับบันทึกและจัดเก็บเหตุการณ์หรือการทำ ดาต้า อาชิฟวิ่ง (Data Archiving) หรือทำดาต้า แฮซชิ่ง (Data Hashing) เป็นต้น เว้นแต่ผู้มีหน้าที่เกี่ยวข้องที่เจ้าของหรือผู้บริหารองค์กร กำหนดให้สามารถเข้าถึงข้อมูลดังกล่าวได้ เช่น ผู้ตรวจสอบระบบสารสนเทศขององค์กร (IT Auditor) หรือบุคคลที่องค์กรมอบหมาย เป็นต้น รวมทั้งพนักงานเจ้าหน้าที่ตามพระราชบัญญัตินี้

2.2.1.3 **จัดให้มีผู้มีหน้าที่ประสานงานและให้ข้อมูลกับพนักงานเจ้าหน้าที่**

คือได้รับการแต่งตั้งตามพระราชบัญญัติว่าด้วยการกระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 เพื่อให้การส่งมอบข้อมูลนั้นเป็นไปด้วยความรวดเร็ว

2.2.1.4 **สามารถระบุรายละเอียดผู้ใช้บริการ**

ในการเก็บข้อมูลจราจรนั้น ต้องสามารถระบุรายละเอียดผู้ใช้บริการเป็นรายบุคคลได้ (Identification and Authentication) เช่น ลักษณะการใช้บริการ พอคซี่ เซิพเวอร์ (Proxy Server), เนท (Network Address Translation :NAT) หรือ พร็อกซี่ แคช (Proxy Cache) หรือ แคส เอนจิ้น (Cache Engine) หรือบริการฟรีอินเทอร์เนต (Free Internet) หรือ บริการ 1222 หรือ วายฟายฮอทสปอร์ท (Wi-Fi Hotspot) ต้องสามารถระบุตัวตนของผู้ใช้บริการเป็นรายบุคคลได้จริง

2.2.2 ข้อมูลที่ต้องทำการจัดเก็บ

2.2.2.1 ข้อมูลอินเทอร์เน็ตที่เกิดจากการเข้าถึงระบบเครือข่าย

- ข้อมูลล็อกที่มีการบันทึกไว้เมื่อมีการเข้าถึงระบบเครือข่ายซึ่งระบุถึงตัวตน และสิทธิในการเข้าถึงเครือข่าย (Access logs specific to authentication and authorization servers, such as TACACS+ or RADIUS or DIAMETER used to control access to IP routers or network access servers)
- ข้อมูลเกี่ยวกับวันและเวลาการติดต่อของเครื่องที่เข้ามาใช้บริการ และเครื่องให้บริการ (Date and time of connection of client to server)
- ข้อมูลเกี่ยวกับชื่อที่ระบุตัวตนผู้ใช้ (User ID)
- ข้อมูลหมายเลขชุดอินเทอร์เน็ต ที่ถูกกำหนดให้โดยระบบผู้ให้บริการ (Assigned IP address)
- ข้อมูลที่บอกถึงหมายเลขสายที่เรียกเข้ามา (Calling line Identification)

2.2.2.2 ข้อมูลอินเทอร์เน็ตบนเครื่องผู้ให้บริการจดหมายอิเล็กทรอนิกส์ (E-mail Servers)

2.2.2.2.1 ข้อมูลล็อกที่บันทึกไว้เมื่อเข้าถึงเครื่องให้บริการไปรษณีย์อิเล็กทรอนิกส์ (Simple Mail Transfer Protocol : SMTP Log) ซึ่งได้แก่

- ข้อมูลหมายเลขของข้อความที่ระบุในจดหมายอิเล็กทรอนิกส์ (Message ID)
- ข้อมูลชื่อที่อยู่อิเล็กทรอนิกส์ของผู้ส่ง (Sender E-mail Address)
- ข้อมูลชื่อที่อยู่อิเล็กทรอนิกส์ของผู้รับ (Receiver E-mail Address)
- ข้อมูลที่บอกถึงสถานะในการตรวจสอบ (Status Indicator) ซึ่งได้แก่จดหมายอิเล็กทรอนิกส์ที่ส่งสำเร็จ จดหมายอิเล็กทรอนิกส์ที่ส่งคืน จดหมายอิเล็กทรอนิกส์ที่มีการส่งล่าช้า เป็นต้น

2.2.2.2.2 ข้อมูลหมายเลขชุดอินเทอร์เน็ตของเครื่องคอมพิวเตอร์ของผู้ใช้บริการที่เชื่อมต่ออยู่ขณะเข้ามาใช้บริการ (IP Address of Client Connected to Server) ข้อมูลวันและเวลาการติดต่อ ของเครื่องที่เข้ามาใช้บริการและเครื่องที่ให้บริการ (Date and time of connection of Client Connected to server)

2.2.2.2.3 ข้อมูลหมายเลขชุดอินเทอร์เน็ตของเครื่องบริการจดหมายอิเล็กทรอนิกส์ที่ถูกเชื่อมต่ออยู่ในขณะนั้น (IP Address of Sending Computer)

2.2.2.2.4 ชื่อผู้ใช้งาน ยูสเซอร์ ไอดี (ถ้ามี)

2.2.2.2.5 ข้อมูลที่บันทึกการเข้าถึงข้อมูลจดหมายอิเล็กทรอนิกส์ ผ่านโปรแกรมจัดการจากเครื่องของสมาชิกหรือการเข้าถึงเพื่อดึงจดหมายอิเล็กทรอนิกส์ไปยังเครื่องสมาชิกโดยยังคงจัดเก็บข้อมูลที่บันทึกการเข้าถึงข้อมูลจดหมายอิเล็กทรอนิกส์ที่ดึงไปนั้นไว้ที่เครื่องให้บริการป๊อป3 ล็อก (POP3:Post Office Protocol version 3) หรือ ไอเอ็มเอพี4 ล็อก (IMAP4:Internet Message Access Protocol Version 4)

2.2.2.3 ข้อมูลอินเทอร์เน็ตจากการโอนแฟ้มข้อมูลบนเครื่องให้บริการโอนแฟ้มข้อมูล

- ข้อมูลล็อกที่บันทึกเมื่อมีการเข้าถึงเครื่องให้บริการโอนแฟ้มข้อมูล
- ข้อมูลวันและเวลาการติดต่อ ของเครื่องที่เข้ามาใช้บริการและเครื่องที่ให้บริการ
- ข้อมูลหมายเลขชุดอินเทอร์เน็ตของเครื่องคอมพิวเตอร์ของผู้เข้าใช้ที่เชื่อมต่ออยู่ในขณะนั้น (IP source address)
- ข้อมูลชื่อผู้ใช้งาน
- ข้อมูลตำแหน่ง (path) และชื่อไฟล์ที่อยู่บนเครื่องที่ให้บริการโอนถ่ายข้อมูลที่มีการส่งขึ้นมาบันทึกหรือให้ดึงข้อมูลออกไป (Path and filename of data object uploaded or downloaded)

2.2.2.4 ข้อมูลอินเทอร์เน็ตบนเครื่องผู้ให้บริการเว็บ

- ข้อมูลล็อกที่บันทึกเมื่อมีการเข้าถึงเครื่องผู้ให้บริการเว็บ
- ข้อมูลวันและเวลาการติดต่อ ของเครื่องที่เข้ามาใช้บริการและเครื่องที่ให้บริการ
- ข้อมูลหมายเลขชุดอินเทอร์เน็ตของเครื่องคอมพิวเตอร์ ผู้เข้าใช้ที่เชื่อมต่ออยู่ในขณะนั้น
- ข้อมูลคำสั่งการใช้งานระบบ
- ข้อมูลที่บ่งบอกถึงเส้นทางในการเรียกดูข้อมูลยูอาร์ไอ (URI : Uniform Resource Identifier) เช่น ตำแหน่งของเว็บเพจ

2.2.2.5 ชนิดของข้อมูลบนเครือข่ายคอมพิวเตอร์ขนาดใหญ่ ยูสเนท (Usenet)

- ข้อมูลล็อกที่บันทึกเมื่อมีการเข้าถึงเครือข่าย เอ็นเอ็นทีพี ล็อก (NNTP log)
- ข้อมูลวันและเวลาการติดต่อ ของเครื่องที่เข้ามาใช้บริการและเครื่องที่ให้บริการ
- ข้อมูลหมายเลขพอร์ท (port) ในการใช้งาน Protocol process ID
- ข้อมูลชื่อเครื่องให้บริการ (Host name)
- ข้อมูลหมายเลขลำดับข้อความที่ได้ถูกส่งไปแล้ว (Posted message ID)

2.2.2.6 ข้อมูลที่เกิดจากการโต้ตอบกันบนเครือข่ายอินเทอร์เน็ต

- ข้อมูลล็อก เช่น ข้อมูลเกี่ยวกับ วัน เวลาการติดต่อ ของผู้ใช้บริการ (Date and time of connection of client to server) และ/หรือข้อมูลชื่อเครื่องบนเครือข่าย และ/หรือหมายเลขเครื่องของผู้ให้บริการ ที่เครื่องคอมพิวเตอร์เชื่อมต่ออยู่ในขณะนั้น (Hostname and/or IP address) เป็นต้น

2.3 ระบบจัดเก็บไฟล์ล็อกส่วนกลาง (Central Log Server)

2.3.1 ซิสล็อก (Syslog)

ซิสล็อกเป็นกลไกที่ใช้ในการเก็บข้อมูลล็อกของเคอร์เนลและแอบพิเคชั่นบนระบบยูนิกซ์และลินุกซ์ ซึ่งเป็นดีลมอล (daemon) ที่ถูกติดตั้งมาให้พร้อมกับระบบปฏิบัติการในเกือบทุกระบบ โดยผู้ดูแลระบบสามารถปรับแต่งไฟล์ คอนฟิกกูเรชั่น (configuration) เพื่อควบคุมการทำงานของซิสล็อกได้ เช่น ให้ซิสล็อกเก็บข้อมูลไปไว้ที่ไฟล์ใดหรือให้ส่งข้อมูลล็อกนี้ไปเก็บไว้ยังเครื่องอื่นในเครือข่าย

ข้อมูลล็อกที่ควบคุมโดยซิสล็อกนั้น จะถูกกำหนดให้มีค่า facility และ priority โดยส่วนของ facility นั้น เป็นข้อมูลที่อธิบายถึงแหล่งกำเนิดของข้อมูลล็อกนั้นๆ เช่น ข้อมูลล็อกที่ส่งมาจากระบบเมล์ก็จะมี facility เป็น mail ส่วน priority นั้น จะแสดงถึงระดับความสำคัญของเหตุการณ์ที่เกิดสำหรับแต่ละ facility ทั้งนี้ข้อมูลล็อกทุกอันจำเป็นต้องมี facility และ priority เสมอ

ซิสล็อกนั้นเป็นมาตรฐานสำหรับการทำล็อกกิ้ง (logging) ของยูนิกซ์และลินุกซ์แต่ซิสล็อกกำลังจะถูกแทนที่โดย ซิสล็อก-เอ็นจี ซึ่งมีความยืดหยุ่นมากกว่าซิสล็อกมาตรฐานและสามารถเก็บข้อมูลล็อกบนพื้นฐานของ regular expression ได้

2.3.1.1ข้อเสียของซิสล็อก

2.3.1.1.1 เนื่องจากซิสล็อกเป็นการส่งข้อมูลแบบยูดีพี (UDP) ทำให้ผู้ส่งไม่ได้รับความมั่นใจว่าข้อมูลที่ส่งไปให้เครื่องเซิฟเวอร์นั้น จะไปถึงหรือไม่

2.3.1.1.2 การที่ไม่มีการระบุตัวตน เพื่อยืนยันก่อนว่าใครคือคนส่ง ก็อาจจะนำไปสู่การส่งข้อมูลแปลกปลอมปนเข้าไปยังเครื่องล็อกเซิฟเวอร์ เพื่อไปชักนำให้การตามรอยผู้บุกรุกเกิดการหักเหไปสู่ทิศทางที่ไม่ถูกต้อง

2.3.1.1.3 การส่งข้อมูล Plaintext โดยไม่มีการเข้ารหัสก่อนที่จะส่ง อาจจะทำให้ผู้ไม่หวังดีสามารถดักจับข้อมูลไปดูและอาจจะทำให้รู้ได้ว่าระบบเราลงโปรแกรมอะไรไปบ้าง มีช่องโหว่หรือจุดอ่อนอะไร ซึ่งอาจจะนำไปสู่การโจมตีระบบ

2.3.2 ซิสล็อก – เอ็นจี (Syslog – ng)

สำหรับระบบยูนิก ส่วนใหญ่คงคุ้นเคยกับซิสล็อกมาเป็นอย่างดี เพราะซิสล็อกถือได้ว่าเป็นล็อกดีลมอลที่ใช้กันมาอย่างยาวนานและกลายเป็นมาตรฐานของการเก็บข้อมูลล็อกของระบบปฏิบัติการยูนิกในหลายๆตัว แต่อย่างไรก็ตามซิสล็อกก็ยังมีข้อเสียบางอย่างดังที่ได้กล่าวมาในหัวข้อที่แล้ว (ในหัวข้อ 2.3.1.1) ซึ่งล็อกดีลมอลตัวอื่น เช่น ซิสล็อก - เอ็นจี, เอ็มซิสล็อก สามารถแก้ไขข้อบกพร่องดังกล่าวได้ ดังนี้

- ซิสล็อก-เอ็นจี สามารถทำงานได้ทั้งบน ทีซีพี (TCP) และ ยูดีพี (UDP)
- ซิสล็อก-เอ็นจี สามารถทำการกรอง (filter) ข้อมูลได้ด้วย regular expression
- ซิสล็อก-เอ็นจี สามารถทำงานในรูปแบบที่อ้างอิง priority/facility ได้ ดังนั้น มันจึงสามารถทำงานแทนที่ซิสล็อกได้
- ซิสล็อก-เอ็นจี สนับสนุน log forwarding ซึ่งทำให้สามารถทราบได้ว่า ต้นทางของล็อกถูกส่งมาจากเครื่องใด และผ่านเครื่องใดมาบ้าง

นอกจากนี้ ซิสล็อก-เอ็นจียังมีรูปแบบของไฟล์คอนฟิกกูเรชั่น ที่ง่ายแต่มีความยืดหยุ่นสูง สามารถนำไปประยุกต์ใช้ให้ตรงความต้องการได้โดยง่าย และควรรันซิสล็อก-เอ็นจีภายหลังจากการสร้างไฟล์คอนฟิกกูเรชั่นเสร็จสิ้นแล้วเท่านั้น โดยซิสล็อก-เอ็นจีนี้มีออปชันในการรันค่อนข้างง่าย ดังตารางที่ 2.1

ตารางที่ 2.1 syslog-ng command line options

Flag	Description
-d	แสดงข้อความดีบัก
-v	แสดงข้อความดีบักมากกว่าเดิม (verbose)
-f filename	ใช้ filename เป็นไฟล์ configuration (default = /etc/syslog-ng/syslog-ng.conf)
-V	แสดงหมายเลขเวอร์ชัน
-p pidfilename	ตั้งชื่อไฟล์ proce-ID (default = /var/run/syslog-ng.pid)

2.3.3 การ Configuring Syslog-ng

การคอนฟิกกูเรชั่นของซิสล็อก-เอ็นจี จะมีความยุ่งยากมากกว่าของซิสล็อก แต่ก็ให้ประโยชน์ในแง่ของความยืดหยุ่นที่ได้และความสามารถที่มีมากกว่า หลังจากที่ทำความเข้าใจคอนฟิกกูเรชั่นแล้ว ผู้ดูแลระบบสามารถสร้างไฟล์คอนฟิกกูเรชั่นง่ายๆขึ้นมาได้ด้วยตัวเอง และสามารถปรับปรุง

ให้เหมาะสมกับระบบของตนต่อไป โดยปกติแล้วซิสล็อกจะอ่านข้อมูลคอนฟิกกูเรชั่นจากไฟล์ /etc/syslog-ng/syslog-ng.conf

```
options {
use_fqdn(no);
sync(0);
};

source s_sys { unix-stream("/dev/log"); internal(); };
source s_net { udp(); };

destination d_security { file("/var/log/security"); };
destination d_meages { file("/var/log/meages"); };
destination d_console { usertty("root"); };

filter f_authpriv { facility(auth, authpriv); };
filter f_meages { level(info .. emerg) and not facility(auth, authpriv); };
filter f_emergency { level(emerg); };

log { source(s_sys); filter(f_authpriv); destination(d_security); };
log { source(s_sys); filter(f_meages); destination(d_meages); };
log { source(s_sys); filter(f_emergency); destination(d_console); };
```

รูปที่ 2.1 การ Configuring Syslog-ng

จากรูปที่ 2.1 จะเห็นได้ว่าส่วนประกอบหลักของคอนฟิกกูเรชั่นประกอบไปด้วย 5 statement หลักคือ options{}, source{}, destination{}, filter{}, log{} ซึ่งแต่ละ statement จะคั่นด้วยเครื่องหมาย semicolon(;) จะเห็นได้ว่ารูปแบบคอนฟิกกูเรชั่นของ syslog-ng.conf จะคล้ายคลึงกันกับรูปแบบของภาษาซี C ซึ่งทุกๆ statement จะต้องลงท้ายด้วยเครื่องหมาย semicolon ส่วน whitespace หรือช่องว่างนั้นจะไม่มีผลใดๆในคอนฟิกกูเรชั่นจะใช้งานเพียงเพื่อให้สามารถอ่านได้ง่ายเท่านั้น

2.3.3.1 Global options

เป็นออปชันที่ถูกประกาศใช้งานภายใน options {} statement ซึ่งบางออปชันนั้นนอกจากจะสามารถใช้งานได้ใน option {} ของตัวเองแล้วยังสามารถใช้งานใน statement ของตัวอื่นๆ เช่น source {}, destination {}, filter {}, log {} ได้อีกด้วย

ตารางที่ 2.2 Options {}

Option	Description
chain_hostnames(yes | no)	หลังจากแสดง hostname ของเครื่องที่ส่งล็อกมายังเครื่องนี้ผ่านทาง tcp/udp แล้ว ให้แสดง hostname ของทุกเครื่องที่ข้อมูลล็อกถูก handle (โดย syslog-ng) มาตลอดทาง ซึ่งเหตุการณ์นี้จะเกิดขึ้นเมื่อล็อกถูกส่งต่อจาก syslog-ng server ไปยัง syslog-ng server อื่นๆ เป็นทอดๆ (default = yes)
keep_hostname(yes | no)	ให้เชื่อใจ (trust) ค่า hostname ที่อยู่ใน tcp/udp message (default = no)
use_fqdn(yes | no)	บันทึก full name ของเครื่องที่ส่ง tcp/udp message (default = no)
use_dns(yes | no)	ให้ resolve ค่า IP address ในข้อมูลล็อก เป็น hostname (default = yes)
use_time_recvd(yes | no)	ตั้งค่า message timestamp เป็นเวลาที่ล็อกเดินทางมาถึง ซึ่งโดยปกติแล้วจะใช้เวลาที่ระบุในล็อก (default = no)
time_reopen (NUMBER)	เมื่อมีแพ็กเกต tcp ที่สูญหายระหว่างทางหรือเหตุที่ทำให้ไม่สามารถสื่อสารได้ตามปกติ syslog-ng จะพยายามสร้างการสื่อสารใหม่ขึ้นมา โดยจะรอเวลาตามที่ระบุ (NUMBER) หน่วยเป็นวินาที (default = 60)
time_reap(NUMBER)	เมื่อ syslog-ng เปิดไฟล์ที่เป็น inactive file (ไม่มีการเขียนข้อมูลลงไฟล์) syslog-ng จะพยายามปิดไฟล์ดังกล่าว โดยจะรอเวลาตามที่ระบุ (NUMBER) หน่วยเป็นวินาที (default = 60)
log_fifo_size (NUMBER)[a]	ขนาดของ message ที่จะถูกนำไปเข้าคิวในหน่วยความจำก่อนที่จะถูกประมวลผล ถ้าคิวเต็มและ syslog-ng ไม่สามารถทำงานได้ตามปกติ (busy) ข้อความล็อกที่ส่งเข้ามาจะถูกละทิ้ง แต่หากระบุขนาด FIFO จำนวนมากเกินไปก็จะทำให้สิ้นเปลืองหน่วยความจำ (default = 100)
sync(NUMBER)[a]	จำนวนบรรทัดของ message ที่จะเขียนลงไฟล์ก่อนที่ไฟล์จะถูก synchronize (default = 0)
owner(string)[a]	ตั้งค่าชื่อ user สำหรับไฟล์ล็อกที่ syslog-ng สร้างขึ้นมาใหม่ (default = root)
group(string)[a]	ตั้งค่าชื่อ group สำหรับไฟล์ล็อกที่ syslog-ng สร้างขึ้นมาใหม่ (default = root)
perm(NUMBER)[a]	ตั้งค่า file permission สำหรับไฟล์ล็อก (default = 0600)
create_dirs (NUMBER)[a]	เป็นตัวบอกว่าจะให้ syslog-ng สร้างไดเรกทอรีใหม่ได้หรือไม่ ในกรณีที่ path ที่ระบุไม่มีอยู่จริงในระบบ (default = no)
dir_owner(string)[a]	ตั้งค่าชื่อ user สำหรับไดเรกทอรีที่ syslog-ng สร้างขึ้นมาใหม่ (default = root)
dir_group(string)[a]	ตั้งค่าชื่อ group สำหรับไดเรกทอรีที่ syslog-ng สร้างขึ้นมาใหม่ (default = root)
dir_perm(NUMBER)[a]	ตั้งค่า directory permission เมื่อ syslog-ng สร้างไดเรกทอรีใหม่ (default = 700)

[a] : ออปชันที่สามารถนำไปใช้กับ file() ใน destination{} ได้

สำหรับออปชันที่เกี่ยวข้องกับ hostname ได้แก่ chain_hostnames(), keep_hostname(), use_fqdn() และ use_dns() นั้นจะสนใจเฉพาะค่า hostname ของเครื่องที่ส่งล็อกมาเท่านั้น ไม่เกี่ยวข้องกับ hostname ที่ระบุใน message body แต่อย่างใด

2.3.3.1.1 use_dns()

เช่น หากใน syslog-ng.conf มี statement ดังต่อไปนี้

```
options { use_dns(yes); };
```

และเครื่อง joe-chong ซึ่งมีไอพีเป็น 10.0.0.7 ส่งล็อกดังต่อไปนี้มาที่ log server

Oct 13 19:56:56 s_sys@**10.0.0.7** sshd[1222]: Accepted publickey for ROOT from 10.0.0.222 port 1355 ssh2

เครื่อง log server จะทำการบันทึกล็อกดังนี้

Oct 13 19:56:56 s_sys@**joe-chong** sshd[1222]: Accepted publickey for ROOT from 10.0.0.222 port 1355 ssh2

จากตัวอย่างจะเห็นว่าไอพี 10.0.0.7 นั้นถูกแยกแยะ (resolve) ให้เป็น joe-chong แต่ข้อมูลไอพีอื่นที่อยู่ใน message body คือ 10.0.0.222 นั้นไม่ได้ถูกแยกแยะไปด้วย ดังนั้นจึงสรุปได้ว่าออปชัน use_dns(yes) นั้นจะทำการแยกแยะเฉพาะ hostname ที่อยู่ในส่วนต้นบรรทัดของ message เท่านั้น

นอกจากนี้ออปชันบางตัวที่เกี่ยวข้องกับไฟล์และไดเรกทอรี ยังสามารถใช้งานได้ทั้งใน global options () และ destination () ซึ่งก็คือ modifier ของออปชัน file () เช่น owner (), group ()

เป็นต้น ทั้งนี้หากมีการระบุค่าออปชันบางตัวที่ซ้ำกันใน options() section และ section อื่นๆ ค่าที่ระบุใน section อื่นๆ จะถูกนำไปใช้แทนที่ค่าใน options() section

2.3.3.1.2 keep_hostname ()

เป็นออปชันที่ใช้งานค่อนข้างมากซึ่งจะตั้งค่าดีฟอลต์เป็น no หมายถึง ซิสล็อกจะไม่ใช้ค่า hostname ที่ส่งมา แต่มันจะทำการแยกแยะหา hostname จาก source IP address ของแพ็กเกตที่ส่งล็อกเข้ามา เพื่อป้องกันการปลอม hostname จากเครื่องที่ส่งล็อกเข้ามาซึ่งจะแตกต่างจากซิสล็อกซึ่งใช้ค่า hostname ตามที่ได้รับมาจาก log message

2.3.3.1.3 chain_hostname ()

โดยดีฟอลต์มีค่าเป็น yes ซึ่งหมายถึง ซิสล็อกจะทำการแสดงรายชื่อ host ทุก host ที่ message ถูกส่งต่อมา (relayed by syslog-ng) โดย host ดังกล่าวต้องเป็น host ที่ติดตั้ง ซิสล็อก-เอ็นจี และทำหน้าที่ redirect ข้อมูลล็อกมายัง log server (ไม่ใช่ host ที่เป็น network host ตามปกติ เช่น เราท์เตอร์ (router), ไฟย์วอล (firewall)

จากรูปที่ 2.2 แสดงผลของการใช้งาน keep_hostname() และ chain_hostnames () ซึ่งทั้งสองค่าถูกตั้งค่าดีฟอลต์ให้เป็น yes โดยในตัวอย่างข้อมูลล็อกจะถูกสร้างขึ้นโดยเครื่องปัจจุบัน (locally) จากนั้นจะถูกส่งต่อไปยัง host1 ซึ่งมี hostname จริงๆ เป็น "linux" ซึ่งจะส่งข้อมูลล็อกต่อไปยัง host2 โดย host2 จะทำหน้าที่ตรวจสอบ hostname ผ่านทาง ดีเอ็นเอ็ส (DNS) จากนั้นล็อกจึงจะถูกส่งต่อไปยัง host3 ต่อไป

```
Original log entry on host1:
Oct 9 23:57:16 s_loc@linux syslog-ng[1656]: syslog-ng version 1.4.13 starting

Entry as sent to and recorded by host2:
Oct 9 23:57:16 s_loc@linux/host1 syslog-ng[1656]: syslog-ng version 1.4.13 starting

Same log entry as relayed from host2 to host3:
Oct 9 23:57:16 s_loc@linux/host1/host2 syslog-ng[1656]: syslog-ng version 1.4.13 starting
```

รูปที่ 2.2 แสดงตัวอย่างล็อกที่ถูกส่งต่อผ่านโฮสต์

ซึ่งสิ่งที่น่าสนใจจากรูปที่ 2.2 คือ

- เมื่อ host2 บันทึกข้อมูลล็อก ตัวซิสล็อกได้ตรวจสอบข้อมูลจากดีเอ็นเอสแล้วพบว่า จริงๆแล้ว host1 นั้นมี DNS name เป็น linux แต่ซิสล็อกเองก็ยังไม่มั่นใจจึงเพิ่ม hostname "linux" ต่อท้าย hostname "host1" (host1 อาจจะเป็นชื่อที่ปลอมมา)
- timestamp ที่ระบุในล็อกทั้งสามชุดมีเวลาที่ตรงกัน ซึ่งหมายถึง เวลาที่เห็นนั้นถูกสร้างขึ้นจากเครื่องที่ให้กำเนิดล็อกแล้วจึงส่งล็อกต่อไปเรื่อยๆ ผ่านโฮสต์

ต่างๆ ซึ่งโฮสต์เหล่านั้นไม่ได้ตั้งค่า use_time_recvd() ให้เป็น yes โฮสต์ต่างๆ จึงไม่ได้แก้ไขข้อมูล timestamp จึงมีผลให้เวลาทั้งสามจุดตรงกันหมด

- จากข้อมูลล็อกที่ host1 จะพบคำว่า s_loc ปรากฏอยู่ ซึ่งค่าดังกล่าวเป็นค่า source{} ของ ซิสล็อก-เอ็นจี ที่อยู่บน host1

```
options{};
source s_loc {unix-stream("/dev/log"); internal(); };
destination d_host2 { udp("host2" port(514)); };
destination d_local { file("/var/log/messages"); };
log { source(s_loc); source(s_net); destination(d_host2); destination(d_local); };
```

รูปที่ 2.3 แสดง Configuration ของ Syslog-ng บนเครื่อง host1

2.3.3.2 Sources

จากรูปที่ 2.3 มีการประกาศค่า source{} หนึ่งครั้ง โดยข้อมูลภายใน source{} ซึ่งก็คือ source driver ทำหน้าที่ระบุถึงแหล่งที่มาของข้อมูลล็อก ทั้งนี้ใน syslog-ng.conf หนึ่งๆ สามารถประกาศ source{} ได้ไม่จำกัดครั้งซึ่งภายใน source{} แต่ละตัวนั้นสามารถบรรจุ driver ได้ไม่จำกัดเช่นกัน รูปแบบการประกาศ source{}

source sourcelabel1 { drivers([options]); drivers([options]); etc. };

โดย sourcelabel1 หมายถึง string ที่ใช้เพื่ออ้างอิงกลุ่มของ source driver เพื่อให้สามารถนำไปใช้งานต่อได้อย่างสะดวก เช่น

source s_loc { unix-stream("/dev/log"); internal(); };

จากบรรทัดด้านบน s_loc เป็นชื่อที่ถูกใช้เพื่ออ้างอิงถึงข้อมูลล็อกที่ถูกดึงมาจาก /dev/log และข้อมูลล็อกที่รับมาจากซิสล็อก-เอ็นจี ดีลมอลเอง

ซิสล็อก-เอ็นจี มีความหยืดหยุ่นอย่างมากในการใช้งาน source driver ซึ่งสามารถรับข้อมูลล็อกได้จาก Unix socket เช่น /dev/log หรือล็อกจากซิสล็อกเอง รวมทั้งล็อกที่ส่งมาจากเครื่องอื่นผ่านทาง ทีซีพี, ยูดีพี โพรโตคอลและยังสามารถรับล็อกจากไฟล์พิเศษเช่น ไฟล์ใน /proc ได้อีกด้วย

ตารางที่ 2.3 Source drivers

Source	Description
`internal()`	ล็อกที่รับมาจาก syslog-ng daemon เอง
`file("filename" [options])`	ล็อกที่อ่านมาจากไฟล์ที่ระบุไว้ เช่น `/proc/kmsg`
`pipe("filename")`	ล็อกที่รับมาจาก name pipe
`unix-stream("filename" [options])`	ล็อกที่รับมาจาก Unix socket หรืออยู่ในโหมด connection-oriented stream เช่น `/dev/log` (maximum concurrent connections default = 100)
`unix-dgram("filename" [options])`	ล็อกที่รับมาจาก Unix socket หรืออยู่ในโหมด connectionless datagram เช่น ล็อกของ klogd จาก `/dev/log`
`tcp([ip(address)] [port(#)] [max-connections(#)] )`	ล็อกที่รับมาจากเครื่องอื่นที่ส่งข้อมูลผ่านทาง TCP ตามหมายเลขพอร์ตที่ระบุ (default = 514) โดยรับข้อมูลจาก local network interface (default = all) และสามารถระบุจำนวน concurrent connections ได้ (default = 10)
`udp([ip(address)] [port(#)])`	ล็อกที่รับมาจากเครื่องอื่นที่ส่งข้อมูลผ่านทาง UDP ตามหมายเลขพอร์ตที่ระบุ (default = 514) โดยรับข้อมูลจาก local network interface (default = all)

2.3.3.2.1 internal ()

ซิสล็อก-เอ็นจี เองจะส่งข้อมูลล็อก เช่น startup message, errors หรือล็อกอื่นๆ ไปยัง internal() ดังนั้นหากต้องการรับล็อกของตัวโปรแกรมซิสล็อก จะต้องระบุ internal() ไว้ใน source{} ด้วย

2.3.3.2.2 file ()

file() ใช้เพื่อระบุชื่อไฟล์ที่ต้องการให้ซิสล็อก-เอ็นจีไปดึงข้อมูลล็อกมา เช่น ไฟล์ /proc/kmsg ซึ่งเป็นไฟล์ข้อมูลล็อกของเคอร์เนลหากต้องการให้ซิสล็อกดึงข้อมูลล็อกจาก text file ปกติ เช่น ล็อกของ httpd นั้น จะต้องสร้างสคริปต์ขึ้นมาเพิ่มเติม เพื่อทำหน้าที่ไปป์ (pipe) ผลลัพธ์ของคำสั่ง tail -f [filename] ไปยัง logger (ดูรายละเอียดเพิ่มเติมเกี่ยวกับการใช้งาน logger ได้จากคำสั่ง # man logger)

2.3.3.2.3 unix-stream(), unix-dgram()

เป็น source driver ที่สำคัญ โดยจะรับข้อมูลจากการเชื่อมต่อแบบ connection-oriented และ connectionless Unix socket สำหรับลีนุกซ์ที่ใช้เคอร์เนลเวอร์ชัน 2.4.1 หรือสูงกว่านั้น จะใช้งาน Unix datagram socket ดังนั้นหากต้องการเก็บข้อมูลล็อกของ /dev/log จะต้องใช้ unix-dgram("/dev/log") เท่านั้น จึงจะสามารถได้รับล็อกตามปกติ เช่น

```
source s_loc { unix-dgram("/dev/log"); internal(); };
```

แต่หากใช้ลีนุกซ์ที่มีเวอร์ชันของเคอร์เนลเป็น 2.4.0 หรือต่ำกว่า จะต้องใช้ unix-stream() ในการเก็บข้อมูลล็อกจาก /dev/log

2.3.3.2.4 tcp(), udp()

ทั้ง tcp () และ udp () จะรับข้อมูลล็อกจาก remote host ผ่านทาง ทีซีพี โพรโตคอล (connection-oriented) และ ยูดีพี (connectionless) โดยทั้งคู่สามารถตั้งให้รอรับข้อมูลล็อกผ่านทางไอพี แอสเดส (IP address) และ พอร์ต (port) ที่ระบุได้ โดยดีฟอลต์แล้วซิสล็อกจะรอรับการเชื่อมต่อที่ 0.0.0.0:514 ซึ่งหมายถึง "รอรับการเชื่อมต่อที่ทุก network interface, port 514"

การระบุไอพี แอสเดส มีประโยชน์สำหรับโฮสต์ที่มี network interface มากกว่าหนึ่ง และต้องการเปิดพอร์ตรอรับล็อกจากบางอินเตอร์เฟส (interface) เท่านั้น ดังรูปที่ 2.3.3.2.4

```
source s_tcpmessages { tcp( ip(192.168.1.19) port(10514) ); };
source s_udpmessages { ucp(); };
```

รูปที่ 2.4 ตัวอย่างการระบุ ip, port ใน source{}

จากรูปที่ 2.4 ซึ่งกำหนดให้ s_tcpmessages รับข้อมูลล็อกทุกอันที่ส่งมายัง network interface ที่มีไอพีเป็น 192.178.1.19 TCP port 10514 ส่วน s_udpmessages นั้นรอรับข้อมูลล็อกทุกอันผ่านทาง ยูดีพี พอร์ต 514 ในทุกๆ local network interface

2.3.3.2.5 ip(), port(), max_connections()

นอกเหนือจาก ip() และ port() แล้ว ยังมี max_connections() ซึ่งใช้ร่วมกับ tcp() เพื่อจำกัดจำนวนการเชื่อมต่อพร้อมกันสูงสุด ซึ่งการใช้งานออปชันนี้ต้องใช้ค่าที่เหมาะสมกับระบบ เพราะหากกำหนดค่าที่มากไปอาจจะมีผลให้ล็อกบางส่วนถูกทิ้ง (drop) ไปเมื่อเซิร์ฟเวอร์ทำงานเกินพิกัด หากกำหนดน้อยเกินไปและมีการเชื่อมต่อเพื่อส่งล็อกถึงขีดที่กำหนดไว้ จะมีผลให้ข้อมูลล็อกถูกตัดทิ้งไป จนกระทั่งจะมีช่องว่างเพียงพอที่จะสร้างการเชื่อมต่อ

```
source s_tcpmessages { tcp( ip(192.168.1.19) port(10514) max-connections(100)); };
```

รูปที่ 2.5 ตัวอย่างการใช้งาน max-connections()

ค่าดีฟอลต์ของ max-connections() สำหรับ unix-stream() มีค่าเป็น 100 และสำหรับ tcp() มีค่าเป็น 10

2.3.3.3 Destinations

ซิสล็อก-เอ็นจี สามารถเก็บข้อมูลล็อกในรูปแบบเดียวกันกับที่ซิสล็อกเก็บได้ไม่ว่าจะเป็น ASCII file, name pipe, remote host (ผ่านทางยูดีพี) และแสดงผลออกทาง ทีทีวาย (TTY) นอกจากนี้ ซิสล็อก-เอ็นจียังสามารถส่งข้อมูลล็อกไปยัง Unix socket, remote host (ผ่านทีซีพี) และส่งต่อไปยัง standard input ของโปรแกรมอื่นได้ด้วย

ตารางที่ 2.4 Destination drivers

Driver	Description
file("filename [$MACROS]")	เก็บข้อมูลล็อกลง Ascii file ตามปกติ หาก syslog-ng ไม่พบไฟล์ตามที่ระบุ มันจะสร้างให้โดยอัตโนมัติ ส่วน MACRO นั้น ใช้เพื่อกำหนดชื่อไฟล์แบบ dynamic เช่น ตั้งชื่อไฟล์ตาม facility ของข้อมูลล็อก (โปรดอ่านรายละเอียดเพิ่มเติม ที่เอกสารเผยแพร่เรื่อง "ทำความรู้จักกับ syslogd")
tcp("address" [port(#);])	ส่งข้อมูลล็อกไปยัง IP address หรือ hostname ที่ระบุผ่านทาง TCP port ที่ระบุ (default port = 514)
udp("address" [port(#);])	ส่งข้อมูลล็อกไปยัง IP address หรือ hostname ที่ระบุผ่านทาง UDP port ที่ระบุ (default port = 514)
pipe("pipename")	ส่งข้อมูลล็อกไปยัง name pipe เช่น /dev/xconsole
unix-stream("filename" [options])	ส่งข้อมูลล็อกไปยัง Unix socket แบบ connection-oriented เช่น /dev/log
unix-dgram("filename" [options])	ส่งข้อมูลล็อกไปยัง Unix socket แบบ connectionless เช่น /dev/log
usertty(username)	ส่งข้อมูลล็อกไปยัง console ของ user ที่ระบุ
program("/path/to/program")	ส่งข้อมูลล็อกเพื่อนำไปเป็น standard input ของโปรแกรมที่ระบุ

ซิสล็อก-เอ็นจี สามารถเก็บข้อมูลลงไฟล์ได้และมีความสามารถมากกว่าซิสล็อก ตรงที่มีการใช้งานมาโคร ซึ่งมาโครนี้จะช่วยให้สามารถตั้งชื่อไฟล์ที่ใช้เก็บข้อมูลล็อกได้อย่างน่าดู เช่น ตั้งชื่อไฟล์ตามปีเดือนวัน หรือตั้งชื่อไฟล์ตาม facility, priority

```
destination d_dailylog { file("/var/log/messages.$WEEKDAY"); };
```

รูปที่ 2.6 ตัวอย่างการใช้งานมาโคร

จากรูปที่ 2.6 ตัวอย่างคอนฟิกกูเรชั่น ด้านบนเมื่อซิสล็อกต้องการเขียนข้อมูลล็อกลงไฟล์ มันจะสร้างไฟล์ชื่อ /var/log/messages.Tues, /var/log/messages.Wed ซึ่งขึ้นกับวันที่เก็บข้อมูลล็อก-ดังกล่าว

ตารางที่ 2.5 Macros supported in file() destination

Macro	Expands to
Program	ชื่อของโปรแกรมที่ส่งล็อกเข้ามา
HOST	ชื่อโฮสต์ที่เป็นจุดกำเนิดล็อก
FACILITY	facility ของล็อกที่ถูกส่งเข้ามา
PRIORITY or LEVEL	priority ของล็อกที่ถูกส่งเข้ามา
YEAR	ปีปัจจุบัน[a]
MONTH	เดือนปัจจุบัน[a]
DAY	วันที่ปัจจุบัน[a]
WEEKDAY	วันปัจจุบัน[a] เช่น Monday
HOUR	ชั่วโมงปัจจุบัน[a]
MIN	นาทีปัจจุบัน[a]
SEC	วินาทีปัจจุบัน[a]

[a] : หากออปชัน use_time_recvd() ถูกตั้งค่า yes แล้ว ข้อมูลเวลาจะอ้างอิงจาก local system ขณะที่ล็อกเดินทางมาถึง แต่หาก use_time_recvd() มีค่าเป็น no ก็จะอ้างอิงเวลาจากเวลาที่ปรากฏในข้อมูลล็อก

ซิสล็อก-ล็อก จะสร้างไฟล์ขึ้นมาใหม่หากไฟล์ที่ระบุใน file() ไม่มีอยู่จริง นอกจากนี้ ซิสล็อก-เอ็นจี ยังสามารถกำหนดออปชันบางตัวในระดับทั่วไป (general rule) คือให้มีผลกับ คอนฟิกกูเรชั่นทั้งไฟล์ได้ ขณะเดียวกันก็สามารถกำหนดออปชันในระดับ per-log-file ได้ ซึ่งการกำหนดออปชันชนิดหลังนี้จะเป็นการ overridden ออปชันในระดับ general rule

```
destination d_mylog { file("/var/log/ngfiles/mylog" create_dirs(yes)\
dir_owner(root) dir_group(root) dir_perm(700)); };
```

รูปที่ 2.7 การควบคุม file()

จากรูปที่ 2.7 เป็นการระบุออปชัน dir_owner(), dir_group(), dir_perm() ใน destination{} ซึ่งค่าที่ระบุนี้จะมีผลแทนที่ค่าที่ระบุใน options{} โดยอัตโนมัติ นอกจากนี้ยังสามารถระบุออปชัน owner(), group(), perm() ได้เช่นเดียวกันกับออปชันด้านบน

โดยปกติ ซิสล็อก-เอ็นจี จะสร้างไฟล์ล็อกที่ไม่มีอยู่ในระบบโดยอัตโนมัติ เว้นเสียแต่ว่าไฟล์ที่ระบุดังกล่าวจะอยู่ใน path ที่ไม่มีอยู่จริงและออปชัน create_dirs() ถูกตั้งค่าเป็น no

sync() ถูกใช้เพื่อจำกัดความถี่ในการ synchronize ไฟล์ล็อก หากมีค่าสูงๆ จะทำให้ข้อมูลล็อกถูกนำไปเก็บไว้ที่แคช (cache) เป็นจำนวนมากก่อนที่จะถูก synchronize หรือบันทึกลงไฟล์ล็อกต่อไป หาก sync() มีค่าต่ำ ก็เป็นการลดความเสี่ยงในการสูญเสียข้อมูล เพราะข้อมูลที่ถูกประมวลผลแล้วจะถูกบันทึกลงไฟล์ล็อกทันที

ทั้งนี้โดยดีฟอลต์แล้ว ค่าล็อกถูกตั้งค่าเป็นศูนย์ซึ่งหมายถึงให้บันทึกข้อมูลล็อกทุกอันในทันที โดยปกติค่า sync() ต่ำๆ จะเหมาะสำหรับระบบที่ข้อมูลล็อกไม่เยอะมาก ส่วนระบบที่มีข้อมูลล็อกจำนวนมากควรใช้ค่า sync() สูง ซึ่งค่าระหว่าง 100 ถึง 1000 นั้นถือว่ามีค่าสูงพอสมควร ซึ่งผู้ดูแลระบบจะต้องทดสอบเพื่อหาค่าที่เหมาะสมกับระบบของตนต่อไป

อย่างไรก็ตามหากระบบที่ติดตั้งซิสล็อก ได้ติดตั้งโปรแกรมจำพวก log monitoring tool เช่น Swatch แล้วไม่ควรตั้งค่า sync() ไว้สูงมากนักเพราะอาจจะทำให้ไม่สามารถแจ้งเตือนผู้ดูแลระบบได้ในกรณีที่ไฟล์ล็อกโดนลบ

2.3.3.4 Filters

filter หรือการกรองข้อมูลเป็นส่วนที่มีความสำคัญส่วนหนึ่ง นอกเหนือจากการกรองข้อมูลโดยใช้ facility, priority แล้วซิสล็อก-เอ็นจียังสามารถตรวจสอบชื่อโปรแกรมที่ส่งข้อมูลล็อกมา ชื่อเครื่องที่ทำหน้าที่ส่งต่อล็อกมา และยังสามารถกรองข้อมูลล็อกตาม regular expression ที่ตั้งไว้อีกด้วย

filter{} statement ประกอบไปด้วย ลาเบล (label) (ชื่อเรียกของ filter{} ชุดนั้นๆ) และคำสั่งในการกรองข้อมูลอย่างน้อย 1 คำสั่ง โดยสามารถใช้ and, or, not ในการเชื่อมคำสั่งในการกรองข้อมูลได้

ตารางที่ 2.6 filter{} funtions

Function (criteria)	Description
`facility`(facility-name)	facility ที่ต้องการ
`priority`(priority-name) `priority`(priority-name1, priority-name2, etc,) `priority`(priority-name1 .. priority-name2)	ระดับของ priority ที่ต้องการ - สามารถใช้เครื่องหมาย comma (,) คั่น หากต้องการมากกว่าหนึ่งระดับได้ - สามารถใช้เครื่องหมาย .. แทน priority ที่ต้องการระหว่าง priority ที่กำหนดได้ เช่น info .. warn
`level`(priority-name)	เช่นเดียวกันกับ priority
`program`(program-name)	ชื่อโปรแกรมที่สร้างล็อกขึ้นมา
`host`(hostname)	ชื่อ host ที่ล็อกนี้ถูกสร้าง
`match`(regular-expression)	regular expression ที่จะถูกนำไปเปรียบกับกับส่วน body ของล็อก
`filter`(filter-name)	ชื่อ filter อื่นที่ต้องการนำมากรองอีกครั้ง

จากรูปที่ 2.6 แสดงsyslog-ng.conf ในระบบปฏิบัติการลินุกซ์ยูบันทู 8.10(Ubuntu 8.10)

```
filter f_mail { facility(mail); };
filter f_debug { not facility(auth, authpriv, news, mail); };
filter f_messages { level(info .. warn) and not facility(auth, authpriv, cron, daemon, mail, news); };
filter f_cother { level(debug, info, notice, warn) or facility(daemon, mail); };
```

รูปที่ 2.8 ตัวอย่างการใช้งาน filter{}

บรรทัดแรกในรูปที่ 2.3.3.4 filter f_mail กรองได้ข้อมูลล็อกทุกอันที่อยู่ใน facility mail

บรรทัดที่สอง filter f_debug กรองได้ข้อมูลล็อกทุกอันยกเว้น facility auth, authpriv, news, และ mail

บรรทัดที่สาม filter f_messages กรองได้ข้อมูลล็อกทุกอันที่มี priority ตั้งแต่ info จนถึง warn ยกเว้นข้อมูลล็อกที่มี facility เป็น auth, authpriv, cron, daemon, mail, news

บรรทัดสุดท้าย filter f_cother กรองข้อมูลล็อกที่มี priority เป็น debug, info, notice และ warn หรือ ข้อมูลล็อกที่มี facility เป็น daemin และ mail

2.3.3.5 Log statements

หลังจากที่ทำความเข้าใจส่วนประกอบต่างๆ คือ sources, filters และ destinations แล้ว ก็จะนำส่วนประกอบทั้งหมดมารวมไว้ใน log{}

```
source s_loc { unix-stream("/dev/log"); internal(); };
source s_tcpmessages { tcp( ip(192.168.1.19); port(10514);); };

destination d_dailylog { file("/var/log/messages.$WEEKDAY"); };
destination d_untlog { file("/var/log/untlog" owner(unt)) perm(0600)); };

filter f_mail { facility(mail); };
filter f_messages { level(info .. warn) and not facility(auth, authpriv, cron, daemon, mail, news); };

log { source(s_tcpmessages); destination(d_untlog); };
log { source(s_loc); filter(f_mail); destination(d_untlog); };
log { source(s_loc); filter(f_messages); destination(d_dailylog); };
```

รูปที่ 2.9 ตัวอย่าง Syslog-ng.conf

จาก log statement บรรทัดแรกนั้น จะทำให้ข้อมูลล็อกทุกอันที่มาจากเครื่อง 192.168.1.19 จะถูกบันทึกลงในไฟล์ /var/log/untlog

บรรทัดที่สองจะทำให้ข้อมูลล็อกของเมล์ (facility mail) ของ localhost ถูกบันทึกลงในไฟล์ /var/log/untlog

บรรทัดที่สามจะทำให้ข้อมูลล็อกของ localhost ที่ผ่านการกรองของ filter f_messages ถูกบันทึกลงในไฟล์ /var/log/messages.**$WEEKDAY** เช่น /var/log/Mon, /var/log/Sun

จากรูปที่ 2.9 อาจจะเกิดข้อสงสัยว่าล็อกบางส่วนที่ไม่ได้ถูกจัดเก็บโดย log{} statement ทั้งสามตัวนั้นจะถูกจัดเก็บไว้ที่ใด ซิสล็อก-เอ็นจี มีค่า filter(DEFAULT) ซึ่งสามารถใช้ระบุในตอนท้ายเพื่อสั่งให้ ซิสล็อก-เอ็นจี บันทึกข้อมูลล็อกที่ไม่ได้ถูกจัดเก็บโดย log{} ก่อนหน้านี้ได้ ดังรูปที่ 2.10

```
log { source(s_tcpmessages); destination(d_untlog); };
log { source(s_loc); filter(f_mail); destination(d_untlog); };
log { source(s_loc); filter(f_messages); destination(d_dailylog); };
log { source(s_loc); filter(DEFAULT); destination(d_dailylog); };
```

รูปที่ 2.10 ตัวอย่าง Syslog-ng.conf

2.3.4 Advanced Configuration

จากรูปที่ 2.10 แสดงการใช้ซิสล็อก-เอ็นจี เพื่อคอยเฝ้าดูข้อมูลล็อกที่ต้องการ (log monitoring)

```
source s_local { unix_stream("/dev/log"); internal(); };
filter f_denials { match("[Dd]enied|[Ff]ail"); };
destination d_mail { program("/usr/local/sbin/mail.sh"); };
log { source(s_local); filter(f_denials); destination(d_mail); };
```

รูปที่ 2.11 ตัวอย่างแสดงการใช้ Syslog-ng

จากรูปที่ 2.11 เป็นตัวอย่าง script ที่ใช้สำหรับส่งอี-เมล์

```
#!/usr/bash
while read line;
do
echo $line |mail -s "Weirdness on that Linux box" your_email@yourcompany.com
done
```

รูปที่ 2.12 ตัวอย่างการส่งอี-เมลล์

จุดที่น่าสนใจในรูปที่ 2.12 คือ match("[Dd]enied|[Ff]ail") ซึ่งหมายถึง ข้อมูลล็อกใดก็ตามที่มีคำว่า denied, Denied, Fail หรือ fail ปรากฏอยู่ ก็จะถูกส่งในรูปแบบอี-เมล์ไปยัง your_email@yourcompany.com โดย shell script ที่ชื่อ /usr/local/sbin/mail.sh

ข้อควรระวังในการใช้งานดังรูปที่ 2.12 คือ การใช้ program() นั้นเป็นการเรียกใช้งานโปรแกรมที่ระบุ โดยโปรแกรมนั้นจะยังคงรันอยู่จนกว่า ซิสล็อก-เอ็นจี จะหยุดการทำงานหรือเริ่มการทำงานใหม่ ดังนั้นผู้ดูแลระบบควรไตร่ตรองก่อนการใช้งานออปชันดังกล่าว เช่น หากรัน bash process ก็จะทำให้เกิดการสิ้นเปลืองงานทรัพยากร นอกจากนี้หากรันโปรแกรมในฐานะ root ก็จะเป็นการเพิ่มความเสี่ยงให้กับระบบอีกด้วย นอกจากนี้การใช้ระบบเตือนภัยผ่านทางอี-เมล์ดังรูปที่ 2.12 ยังก่อให้เกิดความเสี่ยงที่ทำให้ระบบถูกโจมตีแบบ Denial of Service ได้ เช่น ทำให้ mailbox ของผู้ดูแลระบบเต็ม

2.4 NTP (Network Time Protocol)

เป็นโพรโตคอลที่ใช้ในการซิงโครไนซ์ (Synchronous) เวลาของเครื่องบนเครือข่ายโดยมีกลไกรักษาและควบคุมเวลาได้ในระดับมิลลิวินาที เวลาที่เอ็นพีทีส่งออกไปจะเป็นเวลามาตรฐาน

Universal Time Coordinate (ซึ่งเป็นเวลาเดียวกับ Greenwich Mean Time) เครื่องที่ได้รับข้อมูลไปจะต้องปรับค่าของเวลาตาม Time Zone, Daylight Saving Time หรือรูปแบบอื่นๆที่ตนเองใช้เอง โพรโตคอลจะกำหนดค่าให้กับเครื่องแต่ละเครื่องโดยแบ่งเป็น 16 ระดับ เครื่องที่เป็นระดับ 1 จะสามารถเข้าถึงเวลาได้ในระดับสัญญาณนาฬิกา เครื่องคอมพิวเตอร์ที่มีค่าระดับสูงกว่าจะสอบถามเวลาที่เซิร์ฟเวอร์ที่มีค่าระดับต่ำกว่า โดยไคลเอนท์สามารถขอใช้บริการเซิร์ฟเวอร์ได้หลายเครื่อง ทั้งนี้ก็เพื่อประโยชน์ในการตรวจสอบโดยอาศัยความซ้ำซ้อน (Redundancy) และเพื่อความคงทน (Robust) ของระบบ โพรโตคอล เอ็นพีที รุ่นหนึ่งมีชื่อว่า Simplified NTP (SNTP) พัฒนาขึ้นในปี 1995 เพื่อใช้ในเครื่องพีซี (PC) ซึ่งก่อนหน้านั้นได้มีการกำหนด เอ็นพีที ไว้แล้ว 3 เวอร์ชั่น คือ เอ็นพีที เวอร์ชั่น 1 ,เอ็นพีที เวอร์ชั่น 2 ,เอ็นพีที เวอร์ชั่น3 ซึ่งโพรโตคอลเอ็นพีทีถูกพัฒนาให้มีทั้งบนระบบปฏิบัติการ ยูนิก (Unix) และ วินโดวส์ (Windows)

เป็นที่เข้าใจกันดีแล้วว่าอุปกรณ์คอมพิวเตอร์ หรืออุปกรณ์เครือข่ายต่างๆ ในระบบสารสนเทศนั้น มีความสามารถของการรักษาความเที่ยงตรงและความแม่นยำของเวลาได้แตกต่างกัน ทั้งนี้ขึ้นอยู่กับปัจจัยหลายด้าน เช่น วัสดุที่ให้สร้างวงจรเวลาของอุปกรณ์คอมพิวเตอร์, อุณหภูมิ, ความชื้น, คลื่นแม่เหล็กไฟฟ้า หรือความสม่ำเสมอของพลังงานที่จ่ายให้กับวงจรเวลา เป็นต้น ส่งผลให้อุปกรณ์เหมือนกันหรือต่างกันอาจจะให้ค่าเวลาที่แตกต่างกัน

หากอุปกรณ์คอมพิวเตอร์หรืออุปกรณ์เครือข่ายในระบบสารสนเทศมีค่าเวลาที่แตกต่างกันแล้วนั้นจะส่งผลให้เกิดปัญหากับผู้ใช้งาน รวมทั้งผู้ดูแลระบบในการปฏิบัติงานต่างๆได้ เช่น

- ความคาดเคลื่อนของเวลาในการการแจ้งปัญหาของระบบสารสนเทศระหว่าง ผู้ใช้งานและผู้ดูแลระบบ
- ความสับสนในการตรวจสอบ และวิเคราะห์เหตุการณ์ต่างๆ เช่น เหตุการณ์การบุกรุก เหตุการณ์ของปัญหาด้านเครือข่าย หรือระบบคอมพิวเตอร์
- ผู้พัฒนามีความสับสนในเวอร์ชันของโค้ดระหว่างการพัฒนา
- มีการใช้งานไฟล์ข้อมูล หรือฐานข้อมูลที่ซ้อนทับกัน

จากตัวอย่างปัญหาข้างต้นจะเห็นว่าผู้ดูแลระบบและผู้ใช้งานระบบสารสนเทศ มีความจำเป็นต้องทำให้อุปกรณ์คอมพิวเตอร์และอุปกรณ์เครือข่ายของระบบสารสนเทศในองค์กรมีค่าเวลาเที่ยง-ตรงและแม่นยำเหมือนกัน

Network Time Protocol เป็นโพรโตคอลในระดับ Application Layer ของระบบเครือข่ายแบบทีซีพี/ไอพี (TCP/IP) ที่ทำหน้าที่ในการเทียบเวลาระหว่างอุปกรณ์คอมพิวเตอร์ ซึ่งอ้างอิงจาก RFC หมายเลข RFC 778, RFC 891, RFC 956, RFC 958, และ RFC 1305 การทำงานของโพรโตคอลชนิดนี้จะต้องอาศัยเครื่องให้บริการที่เปิดพอร์ตหมายเลข 123 ชนิดยูดีพีในการรอรับข้อมูลร้องขอการเทียบเวลาจากเครื่องลูกข่ายลักษณะการแจกจ่ายเวลาของ เอ็นทีพีนั้นจะอยู่ในรูปแบบลำดับชั้นที่เรียกว่า “Clock Strata” โดยแบ่งลำดับชั้นของการเทียบเวลาดังนี้

2.4.1 Stratum 1

เป็นอุปกรณ์ของแหล่งกำเนิดเวลา เช่น Atomic clocks, จีพีเอส (GPS) เป็นต้น ซึ่งอุปกรณ์แต่ละชนิดมีข้อดีและข้อเสียแตกต่างกัน เช่น การประยุกต์ใช้ จีพีเอส จะมีต้นทุนที่ต่ำกว่า Atomic clock มาก แต่จะมีเสถียรภาพที่น้อยกว่า หากสภาพอากาศไม่เหมาะสม จีพีเอสจะไม่สามารถรับสัญญาดาวเทียมได้ เป็นต้น

2.4.2 Stratum 1

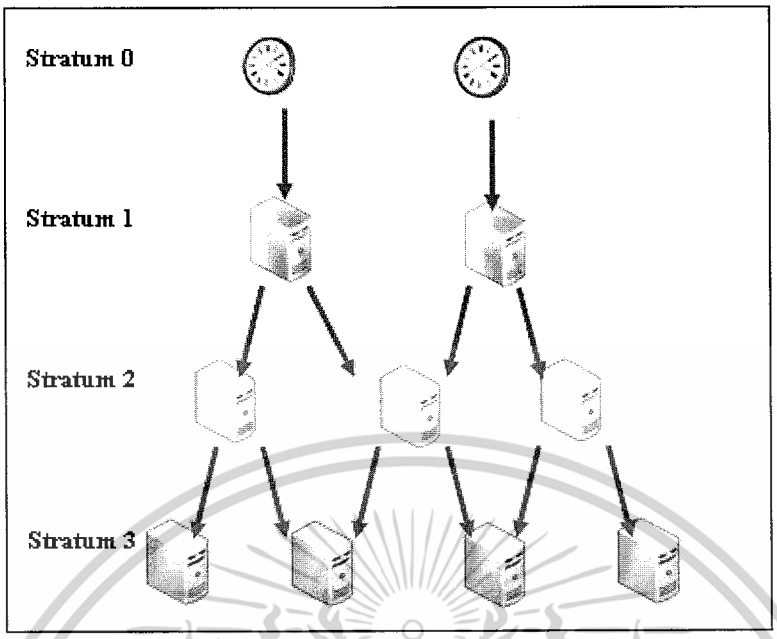
เป็นเครื่องคอมพิวเตอร์แม่ข่ายที่เชื่อมต่อกับ stratum 0 ได้รับค่าเวลามาจาก stratum 0 โดยตรงผ่านการเชื่อมต่อในระบบคอมพิวเตอร์ เช่น RS-232 เป็นต้น

2.4.3 Stratum 2

เป็นเครื่องคอมพิวเตอร์ที่ร้องขอการเทียบเวลาจากเครื่องคอมพิวเตอร์แม่ข่าย stratum 1 ผ่านระบบเครือข่าย ทีซีพี/ไอพี ด้วยการใช้งาน เอ็นพีทีเครื่องคอมพิวเตอร์ในระดับนี้อาจจะร้องขอการเทียบเวลาจาก stratum 1 ได้มากกว่า 1 แหล่ง เพื่อรองรับการทำงานแบบทดแทนกันเมื่อไม่สามารถเข้าถึง stratum 1 ตัวใดตัวหนึ่งก็จะสามารถร้องขอการเทียบเวลาจาก stratum 1 ตัวอื่นได้ต่อไป นอกจากนี้เครื่องคอมพิวเตอร์ใน stratum 2 สามารถเทียบเคียงเวลาระหว่างกันแบบ peer-to-peer เพื่อรักษาเวลาให้เทียบเท่ากันในระดับเดียวกัน

2.4.4 Stratum 3

เป็นเครื่องคอมพิวเตอร์ที่ร้องขอการเทียบเวลาจากเครื่องคอมพิวเตอร์แม่ข่าย stratum 2 ผ่านระบบเครือข่าย ทีซีพี/ไอพี ด้วยการใช้งาน เอ็นทีพีเครื่องคอมพิวเตอร์ในระดับนี้จะสามารถอ้างอิง stratum 2 ได้มากกว่า 1 แหล่ง และสามารถทำงานในรูปแบบ peer-to-peer ได้เช่นเดียวกัน และเอ็นทีพี นั้นสามารถรองรับระดับของการเทียบเวลาได้ถึง 16 ระดับ



รูปที่ 2.13 การอ้างเวลาในแต่ละ Stratum

2.5 การตรวจสอบความคงอยู่ (Integrity Checking)

2.5.1 แฮช ฟังก์ชั่น (Hash Function)

ในกระบวนการเข้ารหัสหรือสร้างดิจิตอล ซิกแนล (Digital Signature) ของข้อมูลต่างๆจะมีการใช้งานฟังก์ชั่นแฮชร่วมด้วยเรื่องจากข้อมูลที่จะทำการเติมดิจิตอล ซิกแนล นั้นจะมีความยาวแตกต่างกันอบางข้อมูลที่มีความยาวสูงมากๆจะใช้เวลาในการสร้างดิจิตอล ซิกแนล นาน นอกจากนี้ข้อมูลที่มีความซ้ำซ้อนสูงและซับซ้อนต่ำเมื่อทำการเข้ารหัสจะส่งผลให้การถอดรหัสทำได้ง่าย ดังนั้นเพื่อเพิ่มความซับซ้อนของข้อมูลและลดขนาดข้อมูลให้มีขนาดเล็กลงจึงจำเป็นต้องใช้ฟังก์ชั่นแฮชร่วมด้วย

คุณสมบัติของฟังก์ชั่นแฮชที่ดีควรมีดังต่อไปนี้

- ผลลัพธ์ของฟังก์ชั่นแฮชควรมีผลลัพธ์เฉพาะตัวกับข้อมูลที่ทำการแฮชนั้นๆ ข้อมูลแต่ละตัวเมื่อผ่านการแฮชแล้วไม่ควรมีผลลัพธ์ที่เหมือนกัน
- สามารถทำงานได้อย่างรวดเร็ว
- มีการกระจายตัวสูง การนำข้อมูลใดๆมาแฮช ควรได้รับผลลัพธ์ที่อยู่ในช่วงที่กำหนดไว้แต่ละตำแหน่ง มีความเป็นไปได้ในการเกิดเท่าๆ กัน
- รหัสแฮช (Hash Code) ที่ได้ ไม่ควรแก้กลับเป็นข้อมูลได้

2.5.1.1 เอ็มดีไฟร์ (md5)

เอ็มดีไฟร์ (Message-Digest Algorithm 5) เป็นอัลกอริทึมที่ใช้ในการสร้างแฮช หรือ digest ของข้อมูลเพื่อใช้ในการตรวจสอบความถูกต้องของข้อมูล สำหรับเอ็มดีไฟร์ถูกนำไปใช้ในการ

ทำงานที่หลากหลายเช่นการตรวจสอบความถูกต้องของไฟล์ ที่แชร์กันในอินเตอร์เน็ต การตรวจสอบความถูกต้องของการบีบอัดและขยายข้อมูล เป็นต้น สำหรับเอ็มดีไฟร์จะทำการสร้างข้อมูลที่เป็นตัวแทน หรือเป็นข้อมูลในการตรวจสอบข้อมูลต่างๆ ซึ่งจะมีลักษณะเป็นตัวเลขฐานสิบหกจำนวน 32 ตัว เอ็มดีไฟร์ เป็นอัลกอริทึมที่คิดค้นขึ้นโดย Ron Rivest ในปี 1991 ซึ่งมาใช้ทดแทนเอ็มดีโฟร์ (MD4)

2.6 แอคเซสคอนโทรล (Access Control)

ในส่วนของเรื่องแอคเซสคอนโทรลนี้ จะกล่าวถึงวิธีการที่จะใช้ในการควบคุมการเข้าถึงไฟล์ล็อกเพื่อที่จะทำให้ตรงตาม พรบ. คอมพิวเตอร์ โดยจะเกี่ยวกับการจำกัดสิทธิ์หรือในส่วนอื่นๆ ที่จะทำให้ไฟล์ล็อกเกิดการเปลี่ยนแปลงทำให้ไม่น่าเชื่อถือ เช่น ผู้ดูแลแอบแก้ไขไฟล์ล็อกบางอย่างให้ผิดไปจากความเป็นจริง ซึ่งจะกล่าวเป็นหัวข้อดังต่อไปนี้

2.6.1 โอเพ่นเอสเอสแอล (Openssl)

เป็นชุดโปรแกรมโอเพ่นซอร์ส ที่ใช้สำหรับเข้ารหัสข้อมูลหรือใช้ในการทำแฮชฟังก์ชั่นซึ่งมีอัลกอริทึมให้เลือกหลายตัว เช่น aes, base64, bf, des, cast5, rc เป็นต้น และมีอัลกอริทึมของแฮชต่างๆ เช่น md2, md4, md5, mdc2, rmd160, sha, sha1 ซึ่งการทำงานของแต่ละอัลกอริทึมจะแตกต่างกันออกไปแล้วแต่ละรูปแบบ สามารถอธิบายอัลกอริทึมที่นิยมใช้กันมาก ได้ดังนี้

- เออีเอส (aes) หรือ Rijindael อัลกอริทึมนี้ได้รับการพัฒนาโดย Joan Daemen และ Vincent Rijmen ในปี 2000 อัลกอริทึมได้รับการคัดเลือกโดยหน่วยงาน National Institute of Standard and Technology (NIST) ของสหรัฐอเมริกาให้เป็นมาตรฐานในการเข้ารหัสชั้นสูงของประเทศอัลกอริทึมมีความเร็วสูงและมีขนาดกระทัดรัดโดยสามารถ ใช้กุญแจที่มีความยาวขนาด 128, 192 และ 256 บิต
- ดีอีเอส (des) เป็นอัลกอริทึมแบบบล็อก ซึ่งใช้กุญแจที่มีขนาดความยาว 56 บิตและเป็นอัลกอริทึมที่มีความแข็งแกร่ง แต่เนื่องด้วยขนาดความยาวของกุญแจที่มีขนาดเพียง 56 บิต ซึ่งในปัจจุบันถือได้ว่าสั้น เกินไปผู้บุกรุกอาจใช้วิธีการลองผิดลองถูกเพื่อค้นหากุญแจที่ถูกต้องสำหรับการถอดรหัสได้
- โบว์ฟิช (blowfish) เป็นอัลกอริทึมที่มีความรวดเร็วในการทำงาน มีขนาดเล็กกระทัดรัดและใช้การเข้ารหัสแบบบล็อค ผู้พัฒนาคือ Bruce Schneier อัลกอริทึมนี้สามารถใช้กุญแจที่มีขนาดความยาวตั้งแต่ไม่มากนักไปจนถึงขนาด 448 บิต ซึ่งทำให้เกิดความยืดหยุ่นสูงในการเลือกใช้กุญแจรวมทั้งอัลกอริทึมนี้ยังได้รับการออกแบบมาให้ทำงานอย่างเหมาะสมกับหน่วยประมวลผลขนาด 32 หรือ 64 บิต โบว์ฟิชได้เปิดเผยสู่สาธารณะและไม่ได้มีการจดสิทธิบัตรใดๆ นอกจากนั้นยังใช้ในโปรแกรม SSH และอื่นๆ

การใช้งานก็ทำได้โดยพิมพ์ คำสั่งดังนี้

#openssl enc –aes-256-cfb –in ไฟล์ที่จะเข้ารหัส –out ชื่อไฟล์ที่เข้ารหัสแล้ว ซึ่งถ้าหากต้องการจะถอดรหัสกลับ (decryption) ก็ใช้คำสั่งตามรูปแบบนี้

#openssl enc –d –aes-256-cfb –in ไฟล์ที่จะถอดเข้ารหัส

2.6.2 เอสเอสแอล (SSL : Secure Socket Layer)

ทรานสปอร์ต เลเยอร์ ซีเคียวริตี้ (Transport Layer Security : TLS) หรือชื่อเดิม ซีเคียว ซ๊อกเกท เลเยอร์ (Secure Sockets Layer : SSL) เอสเอสแอล เป็นโพรโตคอลจัดการความปลอดภัยในระบบอินเทอร์เน็ตที่ใช้ในการสื่อสารข้อมูล กันระหว่างไคลเอนต์กับเซิร์ฟเวอร์ ปกติแล้วข้อมูลที่ส่งไปหากันจะไม่มีการเข้ารหัสข้อมูลแต่อย่างใด ทำให้การดักจับข้อมูลเป็นไปได้โดยง่าย แต่ถ้าเป็นระบบที่ใช้ เอสเอสแอล ข้อมูลจากไคลเอนต์ที่จะส่งไปที่เซิร์ฟเวอร์จะถูกเข้ารหัสก่อนที่จะส่งไปที่เซิร์ฟเวอร์ ทำให้ข้อมูลที่รับส่งกันมีความปลอดภัยมากยิ่งขึ้น

การเข้ารหัสของ เอสเอสแอล มีได้ 2 แบบ คือ

- การเข้ารหัสแบบ 40 บิต
- การเข้ารหัสแบบ 128 บิต

หลักการของการทำงานของ เอสเอสแอล คือ จะมีการเข้ารหัสข้อมูลที่ทางไคลเอนต์โดยเว็บบราวเซอร์จะเป็นตัวเข้ารหัส ให้ เว็บบราวเซอร์จะเอา พับบริกคีย์ (Public key) จากเซิร์ฟเวอร์มาเข้ารหัสกับ มาสเตอร์ (Master key) ที่บราวเซอร์สร้างขึ้นมา จากนั้นก็ใช้คีย์พวกนี้เข้ารหัสข้อมูลที่จะส่งไปให้เซิร์ฟเวอร์ ข้อมูลที่เข้ารหัสเรียบร้อยแล้วจะส่งไปที่เซิร์ฟเวอร์ ซึ่งเซิร์ฟเวอร์มีหน้าที่ในการถอดรหัสนั้นกลับมาเป็นข้อมูลปกติ

2.7 ระบบควบคุมจัดการ

2.7.1 พีเอชพี (PHP)

พีเอชพี เป็นภาษาจำพวก scripting language คำสั่งต่างๆจะเก็บอยู่ในไฟล์ที่เรียกว่าสคริปต์ (script) และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษาสคริปก็เช่น JavaScript, Perl เป็นต้น ลักษณะของพีเอชพีที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ พีเอชพีได้รับการพัฒนาและออกแบบมา เพื่อใช้งานในการสร้างเอกสารแบบ เอชทีเอ็มเอล (HTML) โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า พีเอชพี เป็นภาษาที่เรียกว่า server-side หรือ HTML-embedded scripting language เป็นเครื่องมือที่สำคัญชนิดหนึ่ง ที่ช่วยให้สามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น

2.7.1.1 รูปแบบของภาษา พีเอชพี

PHP จัดเป็นภาษาสคริปต์ภาษหนึ่งที่ดำเนินการที่ฝั่งเซิพเวอร์ คือ เมื่อโค้ดถูกเรียกใช้โดยบราวเซอร์ โปรแกรม PHP ที่อยู่ในเครื่องที่เป็นเวปเซิพเวอร์จะทำการประมวลผลแล้วสร้างผลลัพธ์

ที่อยู่ในรูปแบบของภาษา HTML ขึ้น แล้วจึงส่งมาให้กับเครื่องไคลเอนท์เพื่อให้บราวเซอร์แสดงผล ลักษณะการเขียนจะแทรกไว้ภายในไฟล์ HTML โดยเปิดด้วยแท็ก <?php หรือ <? หรือ <script language="php"> และปิดด้วย ?> หรือ </script> ดังนี้

```
<html>
    <head>PHPFirstProgram</head>
    <body>
        <?php>HelloPHP!!?>
    </body>
</html>
```

2.7.2 World Wide Web : www

เวอร์วายเว็บ (World Wide Web:WWW) เป็นบริการหนึ่งของอินเตอร์เน็ต ซึ่งมีการพัฒนาขึ้นมาในช่วงปลายปี 1989 โดยทีมงานจาก ห้องปฏิบัติการทางจุลภาคฟิสิกส์แห่งยุโรป (European Particle Physics Labs) หรือที่รู้จักกัน ในนาม CERN (Conseil European pour la Recherche Nucleaire) ประเทศสวิตเซอร์แลนด์ และได้มี การพัฒนาภาษาที่ใช้สนับสนุน การเผยแพร่เอกสารของนักวิจัย หรือเอกสารเว็บ (Web Document) จากเครื่องแม่ข่าย (Server) ไปยังสถานที่ต่างๆ ในระบบ WWW เรียกว่า ภาษา HTML (HyperText Markup Language)

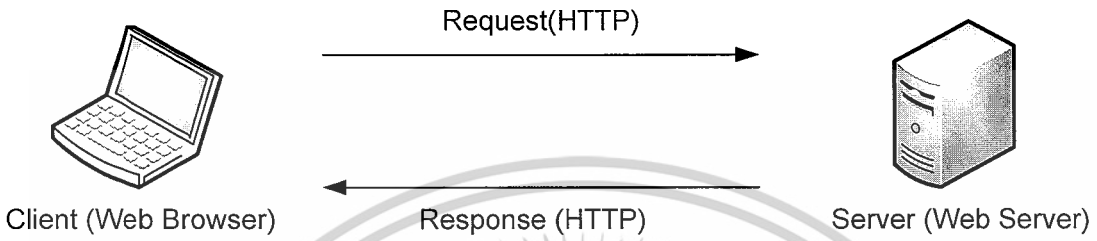
การเผยแพร่ข้อมูลทางอินเทอร์เน็ต ผ่านสื่อประเภทเวปเพจ (Web Page) เป็นที่นิยมกันอย่างสูงในปัจจุบัน ไม่เฉพาะข้อมูลโฆษณาสินค้า ยังรวมไปถึงข้อมูลทางการแพทย์ การเรียน งานวิจัยต่างๆ เพราะเข้าถึงกลุ่มผู้สนใจได้ทั่วโลก ตลอดจนข้อมูลที่นำเสนอออกไป สามารถเผยแพร่ ได้ทั้งข้อมูลตัวอักษร ข้อมูลภาพ ข้อมูลเสียง และภาพเคลื่อนไหว มีลูกเล่นและเทคนิคการนำเสนอ ที่หลากหลาย อันส่งผลให้ระบบ WWW เติบโตเป็นหนึ่ง ในรูปแบบบริการ ที่ได้รับความนิยมสูงสุดของ ระบบอินเทอร์เน็ต

ลักษณะเด่นของการนำเสนอข้อมูลเว็บเพจ คือ สามารถเชื่อมโยงข้อมูลไปยังจุดอื่นๆ บนหน้าเว็บได้ ตลอดจนสามารถเชื่อมโยงไปยังเว็บอื่นๆ ในระบบเครือข่าย อันเป็นที่มาของคำว่าไฮเปอร์เท็ก (HyperText) หรือข้อความที่มีความสามารถมากกว่าข้อความปกตินั่นเอง จึงมีลักษณะคล้ายกับว่าผู้อ่านเอกสารเว็บสามารถโต้ตอบกับเอกสารนั้นๆด้วยตนเอง ตลอดเวลาที่มีการใช้งานนั่นเอง

2.7.2.1 การทำงานของบริการ WWW

จะมีลักษณะเช่นเดียวกับบริการอื่นๆของอินเตอร์เน็ตคืออยู่ในรูปแบบไคลเอ็นต์เซิร์ฟเวอร์ (Client-Server) โดยมีโปรแกรมเว็บไคลเอ็นต์ (Web Client) ทำหน้าที่เป็นผู้ร้องขอบริการ และมีโปรแกรมเว็บเซิร์ฟเวอร์ (Web Server หรือบางครั้งถูกเรียกว่า http Server) ทำหน้าที่เป็นผู้ให้บริการ โปรแกรมเว็บไคลเอ็นต์ก็คือโปรแกรมเว็บบราวเซอร์ (Web browser) ในเครื่องของผู้ใช้นั่นเอง เช่น

Microsoft Internet Explorer, Mozilla Firefox, Opera, Safari, etc สำหรับโปรแกรมเว็บเซิร์ฟเวอร์นั้นจะถูกติดตั้งไว้ในเครื่องของผู้ให้บริการเว็บไซท์ การติดต่อระหว่างโปรแกรมเว็บบราวเซอร์กับโปรแกรมเว็บเซิร์ฟเวอร์จะกระทำผ่านโปรโตคอล เอชทีทีพี (HTTP:Hypertext Transfer Protocol) ดังรูปที่ 2.14



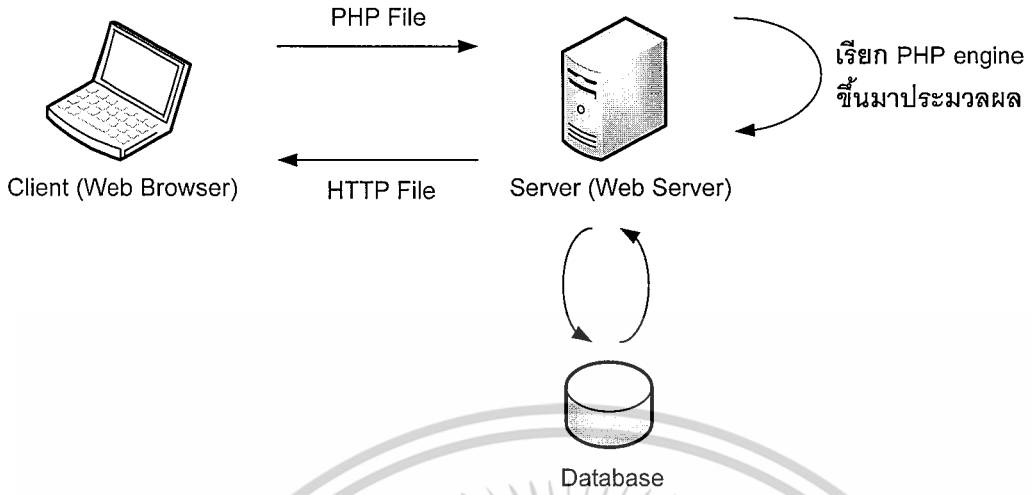
รูปที่ 2.14 แสดงการติดต่อระหว่าง Client และ Server

2.7.3 การทำงานของเว็บเพจที่มีสคริปต์ พีเอชพี

สำหรับเว็บเพจธรรมดาที่โดยปกติแล้วจะมีนามสกุลของไฟล์เป็น เอชทีเอ็ม (htm) หรือ เอช-ทีเอ็มแอล (html) นั้น เมื่อเราใช้เว็บบราวเซอร์เปิดดูเว็บเพจใด เว็บเซิร์ฟเวอร์ก็จะส่งเว็บเพจนั้นกลับมายังบราวเซอร์ จากนั้นบราวเซอร์จะแสดงผลไปตามคำสั่งภาษา HTML (Hypertext Markup Language) ที่อยู่ในไฟล์

เป็นเว็บเพจที่มีลักษณะ static ผู้ใช้จะพบกับเว็บเพจหน้าตาเดิมๆ ทุกครั้งจนกว่าผู้ดูแลเว็บจะทำการปรับปรุงเว็บเพจนั้น นี่คือข้อจำกัดอันมีสาเหตุมาจากภาษา HTML ซึ่งเป็นภาษาที่ใช้อธิบายหน้าตาของเว็บเพจ (HTML จัดเป็นภาษาในกลุ่มที่เรียกว่า page description language) หรือกล่าวอีกนัยหนึ่งคือ HTML สามารถกำหนดให้เว็บเพจมีหน้าตาอย่างที่เราต้องการได้ แต่ไม่ช่วยให้เว็บเพจมีความฉลาดได้

การสร้างเว็บเพจที่มีความฉลาดนั้นสามารถทำได้หลายวิธีด้วยกัน หนึ่งในนั้นคือการฝังสคริปต์หรือชุดคำสั่งที่ทำงานทางฝั่งเซิร์ฟเวอร์ (Server-side script) ไว้ในเว็บเพจ



รูปที่ 2.15 แสดงการทำงานของเว็บเพจที่ฝังสคริปต์ภาษา พีเอชพี

จากรูปที่ 2.15 เป็นการทำงานของเว็บเพจที่ฝังสคริปต์ภาษา PHP ไว้ เมื่อเว็บบราวเซอร์ร้องขอไฟล์ PHP ไฟล์ใด เว็บเซิร์ฟเวอร์จะเรียก PHP engine ขึ้นมาแปล (interpret) และประมวลผลคำสั่งที่อยู่ในไฟล์ PHP นั้น โดยอาจจะมีการดึงข้อมูล HTML (และสคริปต์ที่ทำงานทางฝั่งบราวเซอร์ เช่น client-side JavaScript) จะถูกส่งกลับไปยังบราวเซอร์ บราวเซอร์ก็จะแสดงผลตามคำสั่ง HTML ที่ได้รับมา ซึ่งย่อมไม่มีคำสั่ง PHP ใดๆ หลงเหลืออยู่ เนื่องจากถูกแปลและประมวลผลโดย PHP engine ที่ฝั่งเซิร์ฟเวอร์ไปหมดแล้ว

ให้สังเกตว่าการทำงานของบราวเซอร์ในกรณีนี้ไม่แตกต่างจากกรณีของเว็บเพจธรรมดา เพราะสิ่งที่บราวเซอร์ต้องกระทำก็คือการร้องขอไฟล์จากเว็บเซิร์ฟเวอร์ จากนั้นก็รอรับผลลัพธ์กลับมาแล้วแสดงผล ความแตกต่างจริงๆ อยู่ที่การทำงานทางฝั่งเซิร์ฟเวอร์ ซึ่งกรณีหลังนี้ เว็บเพจ (ไฟล์ PHP) จะผ่านการประมวลผลก่อน แทนที่จะถูกส่งไปยังบราวเซอร์เลยทันที

การฝังสคริปต์ PHP ไว้ในเว็บเพจ ช่วยให้เราสร้างเว็บเพจแบบ dynamic ได้ ซึ่งหมายถึงเว็บเพจที่มีเนื้อหาสาระหรือหน้าตาเปลี่ยนแปลงไปได้ในแต่ละครั้งที่ผู้ใช้เปิดดู โดยขึ้นอยู่กับเงื่อนไขต่างๆ เช่น ข้อมูลที่ผู้ใช้ส่งมา, ข้อมูลในฐานข้อมูล ฯลฯ เป็นต้น

2.8 ระบบฐานข้อมูล (Database System)

ฐานข้อมูลถือได้ว่าเป็นแอปพลิเคชั่น (Application) หรือโปรแกรมตัวหนึ่งซึ่งทำหน้าที่ในการเก็บข้อมูลต่าง ๆ ด้วยวิธีการและรูปแบบที่เหมาะสม เพื่อให้ผู้ใช้งานสามารถเก็บข้อมูล, ดูแลรักษาข้อมูล และนำข้อมูลมาใช้งานได้ง่ายกว่าการเก็บข้อมูลในรูปแบบไฟล์

ในความเป็นจริงคำว่า "ระบบฐานข้อมูล" มีความหมายแตกต่างกับคำว่า "ฐานข้อมูล" โดยระบบฐานข้อมูล (Database System) จะประกอบไปด้วย 4 ส่วนหลักคือ ฐานข้อมูล (Database), ซอฟต์แวร์จัดการระบบฐานข้อมูล (DBMS), โปรแกรมใช้งานฐานข้อมูล (Application Program) และผู้ใช้งาน (User)

2.8.1 การออกแบบฐานข้อมูลด้วยอี – อาร์โมเดล (Entity Relationship Model)

ในการออกแบบฐานข้อมูลจำเป็นต้องทำการศึกษาถึงคุณสมบัติและความสัมพันธ์ระหว่างข้อมูลที่มีอยู่ในระบบเพื่อให้ได้มาซึ่งโครงสร้างพื้นฐานของฐานข้อมูล โดยทั่วไปมักดำเนินการโดยใช้แบบจำลองข้อมูล

อี-อาร์โมเดลเป็นแบบจำลองข้อมูลที่ได้รับความนิยมมากในการใช้เป็นเครื่องมือสำหรับงานออกแบบฐานข้อมูลด้วย อี–อาร์โมเดลจะเสนอโครงสร้างของฐานข้อมูลในระดับแนวคิดออกมาในรูปแบบของแผนภาพที่มีโครงสร้างง่ายต่อการทำความเข้าใจ ทำให้เห็นภาพรวมของเอนทิตี้ทั้งหมดและความสัมพันธ์ระหว่างเอนทิตี้ในระบบฐานข้อมูล

2.8.2 ขั้นตอนในการออกแบบฐานข้อมูลด้วยอี-อาร์โมเดล

ประกอบด้วยขั้นตอนต่างๆดังนี้คือ

2.8.2.1 การศึกษารายละเอียดและลักษณะหน้าที่งานของระบบ

การศึกษารายละเอียดและลักษณะหน้าที่งานของระบบเป็นการศึกษาและรวบรวมเอารายละเอียดที่เกี่ยวกับลักษณะหน้าที่งานของระบบข้อมูลที่เกี่ยวข้องกับขั้นตอนในการทำงานตลอดจนข้อกำหนดและสมมติฐานต่างๆ ซึ่งทำได้ด้วยการสัมภาษณ์หรือศึกษาจากแบบฟอร์มต่างๆที่มีการใช้งานอยู่ในระบบงานขณะนั้น

2.8.2.2 การกำหนดเอนทิตี้ที่ควรมีในระบบฐานข้อมูล

เนื่องจากฐานข้อมูลหนึ่งๆ อาจประกอบด้วยเอนทิตี้ต่างๆได้จำนวนมาก ดังนั้นในขั้นตอนนี้จึงเป็นการนำรายละเอียดในข้อก่อนหน้านี้มาทำการกำหนดเอนทิตี้ที่จำเป็นต้องมีในระบบฐานข้อมูล โดยคำนึงถึงการเป็นเอนทิตี้ประเภทอ่อนแอตลอดจน Super type หรือ Subtype ด้วย

2.8.2.3 การกำหนดความสัมพันธ์ระหว่างเอนทิตี้

การกำหนดความสัมพันธ์ระหว่างเอนทิตี้ จะเป็นการกำหนดประเภทของความสัมพันธ์ระหว่างเอนทิตตี้ โดยพิจารณาจากข้อกำหนดและสมมติฐานต่างๆ

2.8.2.4 การกำหนดคุณลักษณะของเอนทิตี้

การกำหนดคุณลักษณะของเอนทิตี้เป็นการกำหนดว่า ในแต่ละเอนทิตี้ควรจะประกอบด้วย Property ใดบ้าง Property ใดที่มีคุณสมบัติเป็น Key Property หรือ Composite Property หรือ Derived Property

2.8.2.5 การกำหนดคีย์หลัก (Primary key) ของแต่ละเอนทิตี้

การกำหนดคีย์หลักของแต่ละเอนทิตี้นั้น เป็นการกำหนด Key Property ของแต่ละเอนทิตี้ เพื่อให้แต่ละสมาชิกในเอนทิตี้สามารถมีคุณสมบัติที่เป็นเอกลักษณ์เฉพาะได้

2.8.2.6 การนำสัญลักษณ์ที่ใช้ใน อี-อาร์โมเดล มาอธิบายความสัมพันธ์ระหว่างข้อมูล

การนำสัญลักษณ์ที่ใช้ใน อี–อาร์โมเดล มาอธิบายความสัมพันธ์ระหว่างข้อมูลเป็นการนำรายละเอียดในขั้นตอนต่างๆ มาพิจารณาทบทวนเพื่อเพิ่มหรือลดเอนทิตี้ Property และความสัมพันธ์ต่างๆ จากนั้นจึงนำข้อมูลที่ได้จากขั้นตอนทั้งหมดมาเขียนเป็นแบบจำลอง เพื่ออธิบายความสัมพันธ์ระหว่างข้อมูลด้วยสัญลักษณ์ต่างๆหรือ อี-อาร์ไดอะแกรม ดังนั้นแบบจำลองข้อมูลที่เกิดขึ้นจึงมีความชัดเจน สอดคล้องถูกต้องและเหมาะสมกับองค์ประกอบของงานที่กำลังศึกษาทำให้เป็นที่ยอมรับของทุกฝ่ายที่เกี่ยวข้อง

2.8.3 โครงสร้างของภาษาเอสคิวเอล (SQL)

ภาษาเอสคิวแอล ย่อมาจาก Structured Query language หรือภาษาในการสอบถามข้อมูล เป็นภาษาทางด้านฐานข้อมูลที่สามารถสร้างและปฏิบัติการกับฐานข้อมูลแบบสัมพันธ์ (Relational Database) โดยเฉพาะ และเป็นภาษาที่มีลักษณะคล้ายกับภาษาอังกฤษ ภาษาเอสคิวแอลถูกพัฒนาขึ้นจากแนวคิดของ Relational Calculus หรือ Relational Algebra เป็นหลัก ภาษาเอสคิวแอล เริ่มพัฒนาครั้งแรกโดย Almaden Research Center ของบริษัท IBM โดยมีชื่อเริ่มแรกว่า “ซีเควล” (Sequel) ต่อมาได้เปลี่ยนชื่อเป็น “เอสคิวแอล” (SQL) หลังจากนั้นภาษาเอสคิวแอล ได้ถูกพัฒนาโดยผู้ผลิตซอฟแวร์ด้านระบบจัดการฐานข้อมูลเชิงสัมพันธ์จนเป็นที่นิยมกันอย่างแพร่หลายในปัจจุบัน โดยผู้ผลิตแต่ละรายก็ได้พยายามที่จะพัฒนาระบบจัดการฐานข้อมูลของตนเองให้มีลักษณะเด่นเฉพาะขึ้นมา ทำให้รูปแบบการใช้คำสั่งเอสคิวแอล มีรูปแบบที่แตกต่างกันไปบ้าง เช่น PRACLE ACCESS SQL Base ของ Sybase INGRES หรือ SQL Server ของ Microsoft เป็นต้น ดังนั้นในปี ค.ศ. 1986 American National Standards Institute (ANSI) จึงได้กำหนดมาตรฐานของเอสคิวแอลขึ้น อย่างไรก็ดีโปรแกรมฐานข้อมูลที่ขายในท้องตลาดได้ขยาย เอสคิวแอลออกไปจนเกินข้อกำหนดของ ANSI โดยเพิ่มคุณสมบัติอื่นๆ ที่คิดว่าเป็นประโยชน์เข้าไปอีกแต่โดยหลักทั่วไปแล้วก็ยังปฏิบัติตามมาตรฐานของ ANSI ในการอธิบายคำสั่งต่างๆของภาษาเอสคิวแอล ในหนังสือเล่มนี้จะอธิบายคำสั่งที่เป็นรูปแบบคำสั่งมาตรฐานของภาษาเอสคิวแอล โดยทั่วไป

2.8.4 ประเภทของคำสั่งของภาษาเอสคิวแอล

ภาษาเอสคิวแอล เป็นภาษาที่ใช้งานได้ตั้งแต่ระดับเครื่องคอมพิวเตอร์ส่วนบุคคล (PC) ไปจนถึงระดับเมนเฟรม ประเภทของคำสั่งในภาษาเอสคิวแอล (The subdivision of SQL) แบ่งออกเป็น 3 ประเภท คือ

2.8.4.1 ภาษาสำหรับการนิยามข้อมูล (Data Definition Language: DDL)

ประกอบด้วยคำสั่งที่ใช้ในการกำหนดโครงสร้างข้อมูลว่ามีคอลัมน์อะไร แต่ละคอลัมน์เก็บข้อมูลประเภทใด รวมถึงการเพิ่มคอลัมน์ การกำหนดวิวหรือตารางเสมือนของผู้ใช้ เป็นต้น

2.8.4.2 ภาษาสำหรับการจัดการข้อมูล (Data Manipulation Language: DML)

ประกอบด้วยคำสั่งที่ใช้ในการเรียกใช้ข้อมูล การเปลี่ยนปลงข้อมูล การเพิ่มหรือลบข้อมูล เป็นต้น

2.8.4.3 ภาษาควบคุม (Data Control Language: DCL)

ประกอบด้วยคำสั่งที่ใช้ในการควบคุม การเกิดภาวะพร้อมกัน หรือป้องกันการเกิดเหตุการณ์ที่ผู้ใช้หลายคนเรียกใช้ข้อมูลพร้อมกัน และคำสั่งที่เกี่ยวข้องกับการควบคุมความปลอดภัยของข้อมูลด้วยการกำหนดสิทธิของผู้ใช้ที่แตกต่างกัน เป็นต้น

2.8.5 ลักษณะการใช้งานของภาษาเอสคิวแอล

ภาษาเอสคิวแอล เป็นส่วนประกอบหนึ่งของ DBMS มักพบใน DBMS เชิงสัมพันธ์หลายตัว และเป็นที่นิยมใช้ในปัจจุบัน ภาษาเอสคิวแอลง่ายต่อการเรียนรู้ การใช้งานในภาษาเอสคิวแอล แบ่งเป็น 2 ลักษณะ คือ ภาษาเอสคิวแอล ที่โต้ตอบได้ (interactive SQL) และภาษาเอสคิวแอล ที่ฝังในโปรแกรม (embedded SQL)

2.8.5.1 ภาษาเอสคิวแอล ที่โต้ตอบได้

ใช้เพื่อปฏิบัติงานกับฐานข้อมูลโดยตรง เป็นการใช้คำสั่งภาษาเอสคิวแอล สั่งงานบนจอภาพโดยเรียกดูข้อมูลได้โดยตรงในขณะที่ทำงาน เพื่อให้ได้ผลลัพธ์ที่นำไปใช้งานได้ ตัวอย่างเช่น ต้องการเรียกดูข้อมูลในคอลัมน์ SALENAME และ SALECOM จากตาราง SALESTAR ใช้คำสั่งของภาษา ดังนี้

```
SELSECT SALENAME, SALECOM
FROM SALESATB
```

2.8.5.2 ภาษาเอสคิวแอล ที่ฝังในโปรแกรม

เป็นภาษาเอสคิวแอล ที่ประกอบด้วยคำสั่งต่างๆ ของภาษาเอสคิวแอล ที่ใส่ไว้ในโปรแกรมที่ส่วนมากแล้วเขียนด้วยภาษาอื่น เช่น โคบอล ปาสคาล ภาษาซี ลักษณะของคำสั่งเอสคิวแอล จะแตกต่างจากภาษาอื่นๆ ในแง่ที่ว่าเอสคิวแอล ไม่มีคำสั่งเกี่ยวกับควบคุม (control statement) เหมือนภาษาอื่น เช่น if…then…else, for…do, loop หรือ while ทำให้มีข้อจำกัดในการเขียนชุดคำสั่ง การใช้งานภาษาเอสคิวแอลฝังในโปรแกรมอื่น จะทำให้ภาษาเอสคิวแอลมีความสามารถ และมี

ประสิทธิภาพมากยิ่งขึ้น ผลลัพธ์ของคำสั่งที่เกิดจากภาษาเอสคิวแอล ที่ฝังในโปรแกรมจะถูกส่งผ่านไปให้กับตัวแปรหรือพารามิเตอร์ที่ใช้ โดยโปรแกรมที่ภาษาเอสคิวแอลไปฝังตัวอยู่ก็เช่น

```
While not end-of-file(input)do
Begin
Reading(id-num, salesperson, loc, comm);
EXEC SQL INSERT INTO SALESTAB
VALUES(:id-num,:salesperson,:loc,:comm );
End;
```

ทั้งภาษาเอสคิวแอล ที่โต้ตอบได้และภาษาเอสคิวแอล ที่ฝังในโปรแกรมจะมีลักษณะของคำสั่งที่ใช้งานเหมือนกัน จะต่างกันแต่เพียงภาษาเอสคิวแอล ที่ฝังในโปรแกรมจะมีวิธีการเชื่อมโยงกับภาษาอื่นๆ

2.8.6 การบันทึกข้อมูล, การปรับปรุงข้อมูลและการลบข้อมูล

ในระบบฐานข้อมูล การบันทึกข้อมูล การปรับปรุงข้อมูลและการลบข้อมูลถือเป็นสิ่งสำคัญ ในภาษาเอสคิวแอล มีภาษาสำหรับการจัดการข้อมูล (Data manipulation Language : DML) ซึ่งเป็นภาษาที่ใช้ในการบันทึกข้อมูล การปรับปรุงข้อมูล การลบข้อมูล ภาษาสำหรับการจัดการข้อมูลเป็นส่วนประกอบหนึ่งในภาษาเอสคิวแอล โดยภาษาสำหรับการจัดการข้อมูลใช้สำหรับการจัดการข้อมูลในตารางของฐานข้อมูล ในการใช้คำสั่งที่เป็นภาษาสำหรับนิยามข้อมูลของภาษาเอสคิวแอล เช่น CREATE TABLE จะทำให้ได้โครงสร้างตารางว่างๆ ที่ยังไม่มีข้อมูลใดๆเก็บอยู่ คำสั่งในภาษาเอสคิวแอลสำหรับการจัดการข้อมูลจะเป็นคำสั่งที่ช่วยในการจัดการข้อมูลภายในโครงสร้างตารางที่สร้างขึ้นตัวอย่างของคำสั่งในภาษาสำหรับการจัดการข้อมูลจะเป็นคำสั่งการปรับปรุงข้อมูล ได้แก่ การเพิ่มข้อมูล (INSERT) การปรับปรุง (UPDATE) และ การลบข้อมูล (DELETE) และคำสั่งการเรียกค้นข้อมูลได้แก่คำสั่ง (SELECT)

คำสั่งที่ใช้ในการปรับปรุงข้อมูลของภาษา SQL คือ การเพิ่มข้อมูล (INSERT) การปรับปรุงข้อมูล (UPDATE) และการลบข้อมูล (DELETE) เป็นคำสั่งในภาษาการจัดการข้อมูล เมื่อโครงสร้างหลักของตารางได้ถูกกำหนดขึ้นเรียบร้อยแล้ว ก็จะทำการบันทึกข้อมูลลงในตารางหลักหรืออาจทำการปรับปรุง หรือลบข้อมูลในภายหลัง คำสั่งทั้ง 3 นี้ เมื่อดำเนินการในภาษา SQLจะไม่แสดงผลลัพธ์ออกทางหน้าจอ แต่ผลของคำสั่งจะมีผลต่อข้อมูล ผู้ใช้สามารถดูผลของการใช้คำสั่งในการเพิ่มข้อมูล การปรับปรุงและการลบข้อมูล โดยใช้คำสั่งการเรียกค้นข้อมูล (SELECT)

2.8.6.1 คำสั่งการเพิ่มข้อมูล (INSERT)

คำสั่งการเพิ่มข้อมูลในตารางจะใช้คำสั่ง INSERT จะมีอยู่ 2 รูปแบบคือ การเพิ่มข้อมูลเข้าไปทีละแถวและการเพิ่มข้อมูลโดยการดึงกลุ่มข้อมูลด้วยคำสั่งค้นหาข้อมูล

- คำสั่งการเพิ่มข้อมูลทีละแถวโดยระบุข้อมูลที่จะ INSERT เข้าไปโดยตรง รูปแบบของคำสั่งเป็นดังนี้

  INSERT INTO <Table_name>[(columm 1, columm 2, ….)]

  VALUE(<value1, value2, ….>);

  ซึ่งอธิบายคำสั่งได้ ดังนี้

  INSERT INTO เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล

  Table_name ชื่อตารางที่จะเพิ่มข้อมูล

  Column 1, Column 2,… คอลัมน์ที่ต้องการเพิ่มข้อมูล

  Value1, value2, ค่าข้อมูลของแต่ละคอลัมน์ที่ต้องการเพิ่ม

- คำสั่งการเพิ่มข้อมูลโดยการดึงกลุ่มข้อมูลด้วยคำสั่งค้นหาข้อมูล ในภาษาเอสคิวแอล สามารถใช้คำสั่ง INSERT ในการนำค่าหรือหาค่าจากตารางหนึ่งแล้วไปใส่ไว้ในอีกตารางหนึ่งได้ โดยได้ค่านั้นมาจากการสอบถามข้อมูล รูปแบบเป็นดังนี้

  INSERT INTO <Table_name>[(column 1, column 2,…)]

  SELECT statement;

  ซึ่งอธิบายคำสั่งได้ ดังนี้

  INSERT INTO เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการเพิ่มข้อมูล

  Table_name ชื่อตารางที่จะเพิ่มข้อมูล

  SELECT statement เป็นคำสั่ง SELECT ที่ต้องการข้อมูลอีกตารางหนึ่ง

2.8.6.2 คำสั่งปรับปรุงแถวข้อมูล (UPDATE)

หลังจากที่ป้อนข้อมูลเข้าไปเก็บไว้ในตารางแล้ว ในกรณีที่ต้องการจะปรับปรุงแก้ไขข้อมูลสามารถทำได้ด้วยภาษาเอสคิวแอล การปรับปรุงแถวข้อมูล เป็นการปรับปรุงหรือแก้ไขค่าคอลัมน์ ซึ่งในคำสั่งปรับปรุงข้อมูลอาจมีมากกว่า 1 คอลัมน์ในแถวทุกแถวที่มีเงื่อนไขสอดคล้องกับระบุไว้หลังคำว่า WHERE

รูปแบบของคำสั่งปรับปรุงแถวข้อมูลดังนี้

UPDATE <table name>SET<column 1>[,column 2,…]=

<expression |subquery >

[WHERE<condition>];

ซึ่งอธิบายคำสั่งได้ ดังนี้

UPDATE เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการปรับปรุงข้อมูล

Table_name ชื่อตารางที่ต้องการปรับปรุง

SET<column>ชื่อคอลัมน์ที่ต้องการปรับปรุง

Expression ค่าข้อมูลที่ต้องการปรับปรุง

WHERE<condition> เงื่อนไขในการปรับปรุง

2.8.6.3 คำสั่งการลบข้อมูลทั้งแถว (DELETE)

คำสั่งในการลบแถวข้อมูลเป็นคำสั่งที่ใช้ในการลบแถวข้อมูลทุกแถวที่มีเงื่อนไขสอดคล้องกับที่ระบุไว้หลัง WHERE คำสั่งการลบข้อมูลมีรูปแบบทั่วไปดังนี้

DELETE FROM <Table_name>

[WHERE<condition>];

ซึ่งอธิบายคำสั่งได้ ดังนี้

DELETE FROM เป็นคำสั่งที่ต้องมีทุกครั้งที่ต้องการลบข้อมูล

Table_name ชื่อตารางที่ต้องการลบข้อมูล

WHERE<condition> เงื่อนไขในการลบข้อมูล

2.8.7 การเรียกค้นข้อมูล (SELECT)

การจัดทำฐานข้อมูลในรูปแบบตารางนั้นเกิดจากการที่ ข้อมูลได้ออกแบบมาเพื่อลดความซ้ำซ้อน (Normalization) ดังนั้นข้อมูลที่มีรายละเอียดของข้อมูลมากอาจจะถูกเก็บไว้ในหลาย ๆ ตารางแยกออกมาต่างหาก เช่น ตารางข้อมูลที่เป็นตารางหลัก (master table) และ ตารางข้อมูลที่เป็นตารางเชิงรายการ (transaction table) และตารางข้อมูลที่เป็นตารางอยู่ (address table) เป็นต้น การแยกออกเป็นตารางย่อย ๆ นี้ นอกจากลดความซ้ำซ้อนแล้ว ยังช่วยในการประหยัดเนื้อที่และยังเพิ่มประสิทธิภาพของฐานข้อมูล

2.8.7.1 การเรียกค้นข้อมูลจากตารางหลายตารางในภาษาเอสคิวแอล

เป็นการกำหนดความสัมพันธ์ระหว่างตารางทั้งหลาย โดยที่จะสามารถเอาข้อมูลในตารางกี่ตารางก็ได้ให้มาสัมพันธ์กันดังนั้นจึงสามารถเชื่อมต่อข้อมูลที่แตกต่างกันได้โดยการใช้คำสั่ง WHERE คำสั่ง WHERE เป็นคำสั่งในการกำหนดเงื่อนไขในการเรียกดูข้อมูลใช้คู่กับคำสั่ง SELECT และ FROM

SELECT * FROM TABLE1, TABLE2

2.8.7.2 การเรียกดูข้อมูลแบบซ้อนกัน (subqueries)

เป็นการสร้างคำสั่ง SELECT ซ้อนกันการเรียกดูข้อมูลแบบซ้อนกัน มีจุดประสงค์ก็เพื่อลดภาระในการเชื่อมตารางที่ต้องการใช้หน่วยความจำเป็นเป็นจำนวนมาก คำสั่งย่อยนี้สามารถสร้างหลังคำสั่ง WHERE มีรูปแบบดังนี้

SELECT [*] <column 1, column 2,…>

FROM<table_name>

[WHERE<column list = <Select Statement>]

ซึ่งอธิบายคำสั่งได้ ดังนี้

SELECT คำสั่งที่ต้องมีทุกครั้งที่ต้องการเรียกค้นข้อมูล

Column 1, column 2,… คอลัมน์ที่ต้องการเรียกค้น

FROM การกำหนดว่าให้เรียกดูข้อมูล ได้จากตารางใดบ้าง

Table_name ชื่อตารางที่ต้องการเรียกค้นข้อมูล

WHERE<condition>ส่วนของคำสั่งที่บอกเงื่อนไขที่จะใช้ในการค้นหาข้อมูล

Select Statement ส่วนของคำสั่งที่เรียกค้นข้อมูลตามเงื่อนไข

การทำงานของคำสั่งย่อยที่ใช้ในการระบุเงื่อนไขหรือเรียกข้อมูลจะทำจากคำถามย่อยด้าน ในสุดผลที่ได้จะเป็นค่ากลับมาให้กับค่าที่อยู่หน้าเครื่องหมาย (=) เพื่อเรียกค้นข้อมูลตามที่ต้องการ

บทที่ 3
การออกแบบและพัฒนา

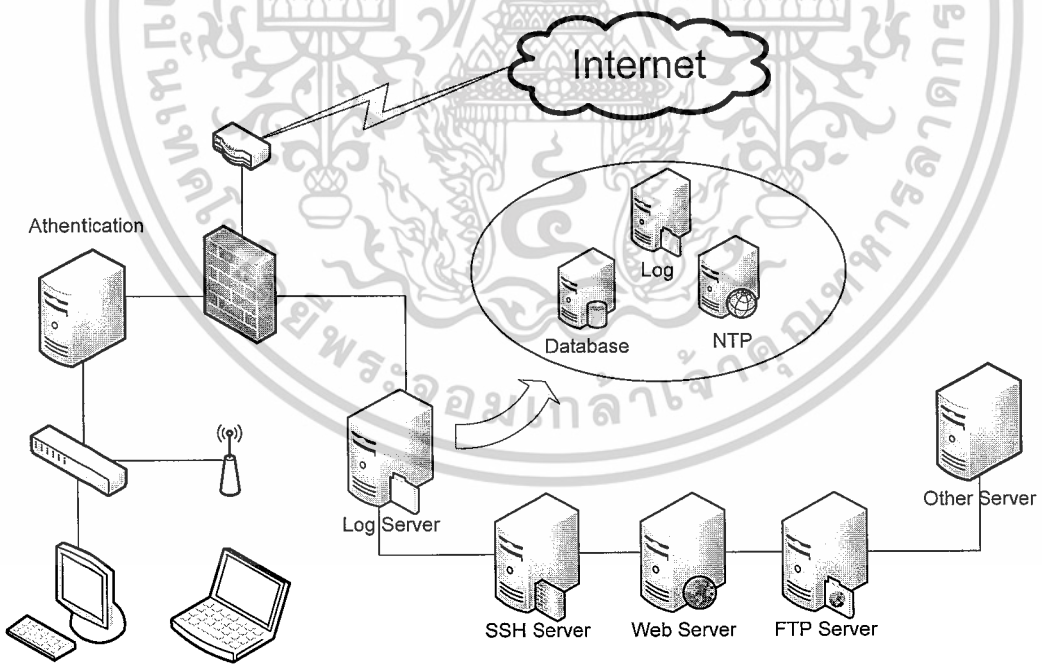
3.1 บทนำ

ในส่วนของการออกแบบและพัฒนานั้นจะต้องมีการพิจารณาถึงองค์ประกอบต่างๆที่จะนำมารวมและสร้างเป็นระบบขึ้นเพื่อให้ทำงานได้ตามเป้าหมายที่วางไว้ โดยในส่วนต่างๆนั้นได้มีการพิจารณาอย่างเหมาะสม

นอกจากนั้นสิ่งสำคัญที่จะต้องนำมาพิจารณาคือ ความปลอดภัยของระบบ ทั้งนี้ระบบที่สร้างขึ้นจะต้องมีความปลอดภัยในตัวเองระดับหนึ่ง ดังนั้นการออกแบบจึงต้องมีความรัดกุมและรอบคอบมากที่สุด โดยรายละเอียดของการออกแบบในแต่ละส่วนนั้นจะอธิบายในหัวข้อถัดไปซึ่งแยกเป็นการออกแบบในส่วนของฮาร์ดแวร์ และการออกแบบในส่วนของซอฟต์แวร์

3.2 การออกแบบฮาร์ดแวร์

โครงสร้างของระบบที่ออกแบบจะมีดังรูป



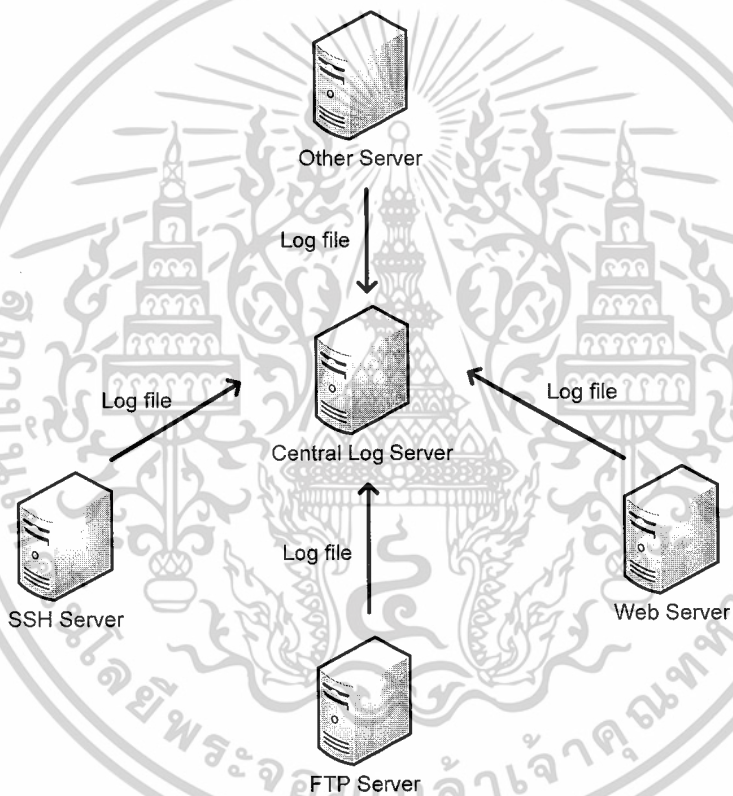
รูปที่ 3.1 การออกแบบฮาร์ดแวร์

3.3 การออกแบบซอฟต์แวร์

ในการออกแบบซอฟต์แวร์จะแบ่งออกเป็น 2 ส่วนหลักคือ

3.3.1 ส่วนรวบรวมไฟล์ล็อกกลาง (Central Log)

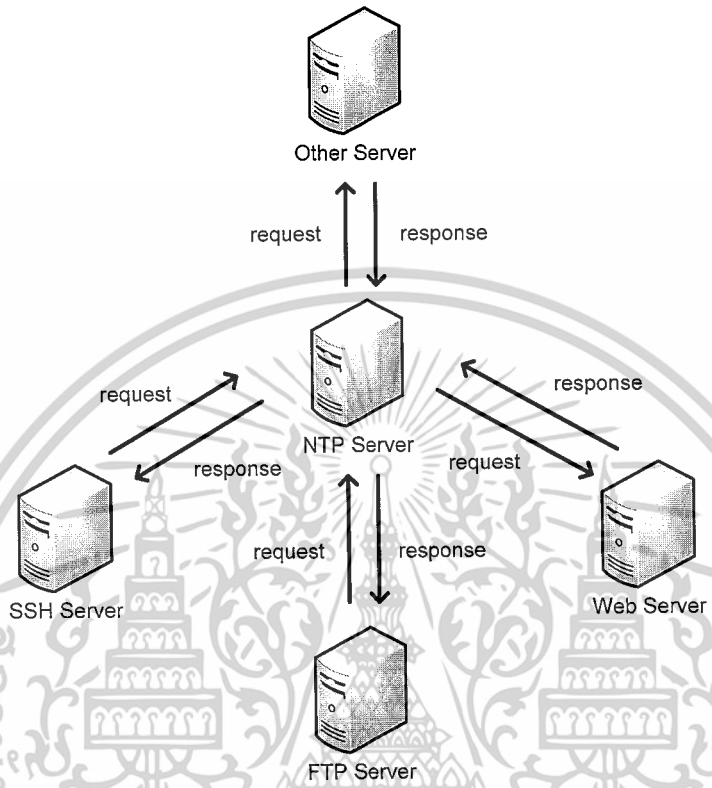
- ระบบจัดการไฟล์ล็อก เป็นส่วนที่ออกแบบเพื่อรับไฟล์ล็อกจากเครื่องเซิพเวอร์ เครื่องไคล-เอ็นท์ หรืออุปกรณ์อื่นๆในระบบ โดยมี Syslog-ng เป็นตัวหลักที่ใช้ในการจัดการ ซึ่งSyslog-ng นั้นเป็นกลไกที่ใช้ในการเก็บข้อมูลล็อกของ kernel และ application บนระบบยูนิกซ์และลินุกซ์ โดยจะทำการบันทึกไฟล์ล็อกลงตามที่ที่ได้ระบุไว้ในไฟล์คอนฟิก เช่น ให้บันทึกที่เครื่อง Log Server ที่ไฟล์ หรืออื่นๆ



รูปที่ 3.2 แสดงการรับไฟล์ล็อกจากเครื่องไคลเอนท์

- ระบบการซิงโครไนท์เวลา เป็นส่วนที่ออกแบบเพื่อให้เวลาในระบบนั้นมีความเที่ยงตรงและเวลาตรงกัน เนื่องจาก พรบ.ว่าด้วยการกระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 กำหนดไว้ว่าอุปกรณ์ทุกชนิดต้องตั้งเวลาให้ตรงกับสากลโดยไม่เกิน 10 มิลลิวินาที ส่วนนี้จึงใช้ เอ็นทีพี ซึ่งเป็นโปรโตคอลที่เกี่ยวกับเวลาเพื่อให้เครื่องเซิพเวอร์ เครื่องไคลเอนท์ หรืออุปกรณ์อื่นๆในระบบไว้ใช้อ้างอิงเวลาให้ตรงกับเครื่อง Log Server เนื่องจากถ้าให้เครื่องเซิพเวอร์ เครื่องไคลเอนท์

หรืออุปกรณ์อื่นๆในระบบเครือข่ายไปอ้างอิงเวลาจากภายนอกอาจเกิดปัญหาเครือข่ายได้ เช่น เครื่อง NTP Server นอกเครือข่ายล่ม หรือมีปัญหาทางด้านมีเดีย



รูปที่ 3.3 แสดงการร้องขอเวลาจากเครื่องไคลเอนท์และเครื่องทำการตอบกลับ

- ระบบตรวจสอบความถูกต้องของข้อมูล เป็นส่วนที่ออกแบบเพื่อเอาไว้เช็คการเปลี่ยนแปลงของไฟล์ล็อกที่ได้ทำการบันทึกลงในเครื่อง Log Server แล้ว เพื่อป้องกันการแก้ไข ปลอมแปลง หรือเปลี่ยนแปลงข้อมูลที่อยู่ในไฟล์ล็อกที่ได้ทำการบันทึกไว้แล้ว โดยในส่วนนี้จะใช้ Tripwire เป็นตัวหลักในการตรวจสอบการเปลี่ยนแปลงไฟล์ล็อกของเครื่อง Log Server
- ระบบควบคุมการเข้าถึง เป็นวิธีการที่ใช้ในการควบคุม จำกัด การเข้าถึงระบบหรือไฟล์ต่างๆที่สำคัญ ที่ต้องให้มีการกำหนดสิทธิ์ว่าบุคคลใดสามารถเข้าถึงระบบหรือไฟล์ใดๆ ได้บ้าง เช่น การเข้ารหัสไฟล์ เป็นต้น ซึ่งจะทำให้ข้อมูลที่ได้ไม่สามารถเข้าใจในเนื้อหาได้ง่ายๆ

3.3.2 ส่วนแสดงผล

- ระบบแสดงผล ออกแบบให้ควบคุมจัดการผ่านเว็ปเบสแอพพลิเคชั่น (Web Base Application) เพื่อให้ง่ายต่อการเรียกดู เข้าถึง โดยใช้ พีเอชพีไฟว์ (PHP5) เป็นตัวหลักในการเขียน Web Application เนื่องจากเป็นภาษาที่ทำงานได้รวดเร็ว รูปแบบภาษาเข้าใจได้ไม่ยาก
- ระบบยืนยันตัวบุคคล ออกแบบให้มีการตรวจสอบ user และ password เพิ่มใช้สำหรับเข้าไปยัง Web Application
- ระบบควบคุมหน้าเวป ออกแบบให้ควบคุมเกี่ยวกับการเข้าถึงหน้าเวป เพื่อป้องกันการเข้าไปยังหน้าเวปที่ไม่มีสิทธิ์เข้าดู
- ระบบฐานข้อมูล เป็นส่วนที่ใช้เก็บและจัดการไฟล์ล็อกที่รับมาจากเครื่องเซิพเวอร์ เครื่องไคลเอ็นท์ หรืออุปกรณ์อื่นๆและนำไปเชื่อมต่อกับ Web Application เพื่อให้ Web Application เรียกข้อมูลไปประมวลผล โดยมีภาษา เอสคิวแอล (SQL) เป็นตัวช่วยในการเชื่อมต่อและส่ง query เข้าไป

บทที่ 4

การทดลองและผลการทดลอง

4.1 บทนำ

ในการทดลองจะเป็นการทดสอบระบบที่ได้ทำการสร้างขึ้นมาโดยจะแยกการทดสอบออกเป็น 4 ส่วน ดังนี้

4.1.1 ทดสอบระบบรับไฟล์ล็อกจากระบบอื่นๆ

4.1.2 ทดสอบการซิ้งค์เวลาระหว่างเอ็นทีพี เซิพเวอร์ และ เอ็นทีพี ไคลแอนในระบบ

4.1.3 ทดสอบการเข้ารหัสไฟล์ล็อก

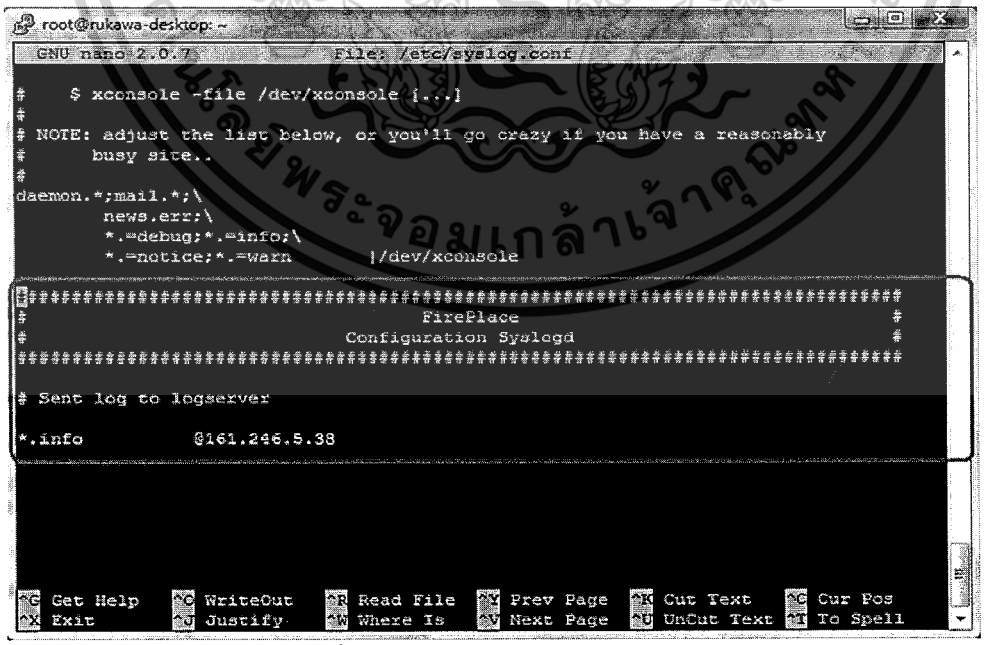
4.1.4 ทดลองการกรองข้อมูลในไฟล์ล็อกเพื่อนำเข้าดาต้าเบส (Data Base)

4.1.5 ทดสอบการเรียกค้นและสรุปข้อมูลไฟล์ล็อกบนเวบแอพพลิเคชั่น (Web Application)

4.2 การทดสอบที่ 1 ทดสอบระบบรับไฟล์ล็อกจากระบบอื่นๆ

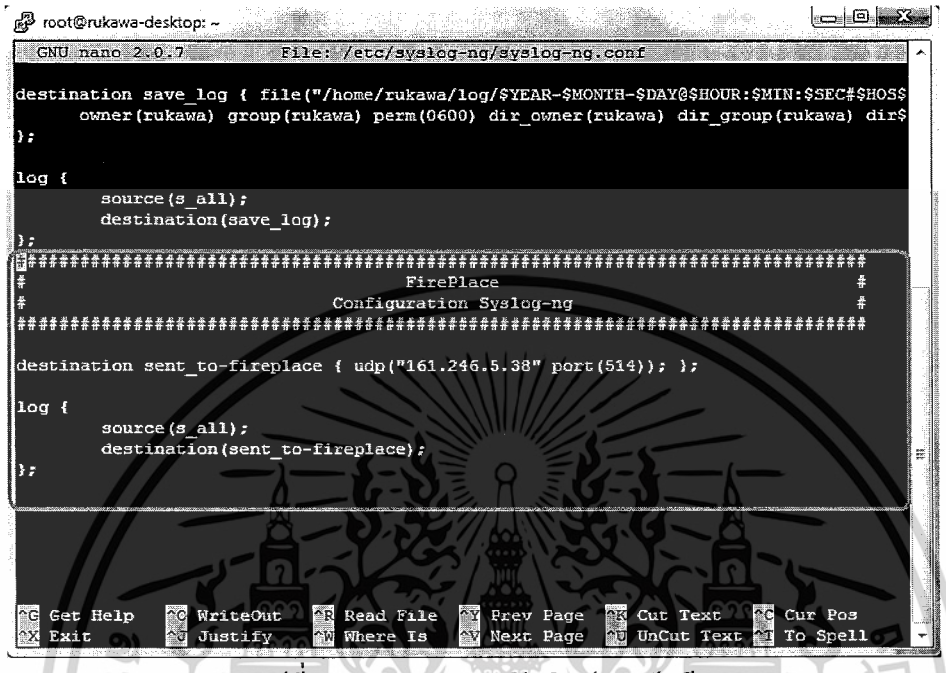
4.2.1 วิธีทดสอบ

4.2.1.1 ทำการคอนฟิกค่า syslog-ng.conf ให้เครื่องไคลแอนหรือเครื่องในระบบส่งไฟล์-ล็อกมายังเซ็ลทรัล ล็อก เซิพเวอร์ โดยจะมีการคอนฟิกซิทล็อกดีจากรูปที่ 4.1 สำหรับเครื่องที่ใช้ซิทล็อกดี ในการส่งไฟล์ล็อก



รูปที่ 4.1 แสดงการคอนฟิกซิทล็อกดี

และคอนฟิกซิทล็อก-เอ็นจีตามรูปที่ 4.2 สำหรับเครื่องที่ใช้ ซิทล็อก-เอ็นจี ในการส่งไฟล์ล็อก



รูปที่ 4.2 แสดงการคอนฟิกซิทล็อก-เอ็นจี

4.2.1.2 ทำการคอนฟิกค่า syslog-ng.conf สำหรับเซ็ลทรัล ล็อก เซิพเวอร์ เพื่อให้รับค่าไฟล์ล็อกที่เครื่องภายในระบบส่งมา ในการทดลองนี้เราจะใช้ ซิทล็อก-เอ็นจี ซึ่งมีการคอนฟิกตามรูปที่ 4.3

รูปที่ 4.3 แสดงการคอนฟิกซิทล็อก-เอ็นจี สำหรับเซ็ลทรัล ล็อก เซิพเวอร์

หลังจากที่คอนฟิกเสร็จไม่ว่าจะเป็น ซิทล็อกดี หรือ ซิทล็อก-เอ็นจี ให้ทำการรีสตาร์ทเซอร์วิส โดยถ้าหากใช้ ซิทล็อกดี ก็ทำการรีสตาร์ทเซอร์วิสโดยใช้คำสั่ง

#/etc/init.d/sysklogd restart

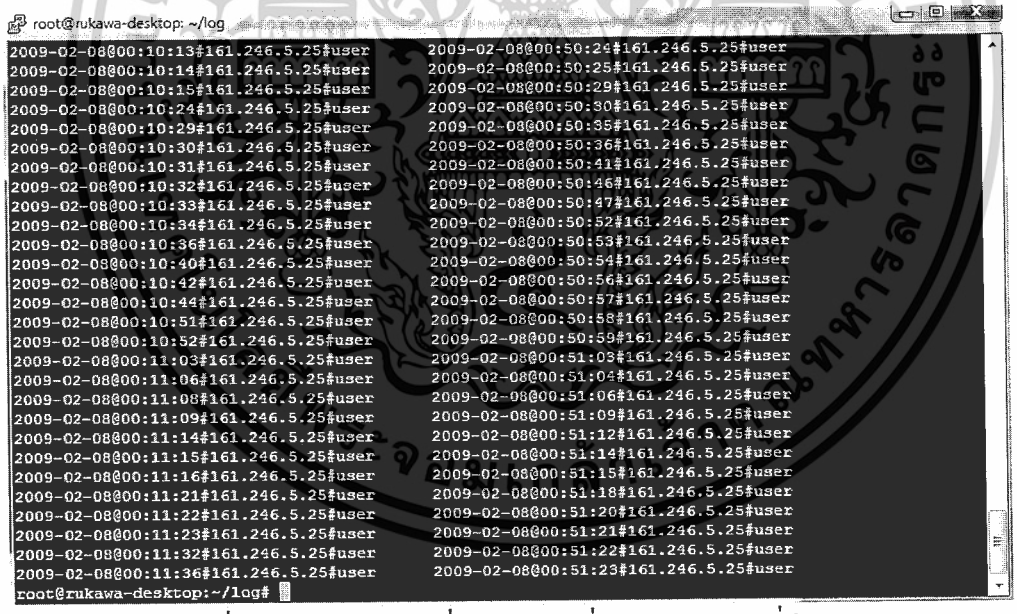
หรือถ้าหากใช้ ซิทล็อก-เอ็นจี ก็ให้ใช้คำสั่ง

#/etc/init.d/syslog-ng restart

เมื่อทำการรีสตาร์ทเซอร์วิสแล้ว ถ้าไม่มีอะไรผิดหรือคอนฟิกถูกระบบก็พร้อมที่จะทำงาน แต่ถ้าหากคอนฟิกผิดหรือคอนฟิกไม่ถูกรูปแบบ ระบบก็จะไม่เริ่มการทำงาน ก็ให้กลับไปแก้ไขในส่วนที่คอนฟิกผิดหรืออื่นๆให้ถูกต้อง

4.2.2 ผลการทดสอบ

หลังจากที่ได้ทำการรีสตาร์ท ซิทล็อก-เอ็นจีแล้ว ซึ่งไม่มีปัญหาในการคอนฟิกผิดรูปแบบ หรืออื่นๆ ตัวซิทล็อกก็จะเริ่มทำงานซึ่งหากเครื่องภายในระบบมีการใช้งานที่ทำให้เกิดเหตุการณ์ล็อกเกิดขึ้น ก็จะส่งไฟล์ล็อกที่เกิดมายังเซ็ลทรัล ล็อก เซิพเวอร์ ซึ่งส่วนของเซ็ลทรัล ล็อก เซิพเวอร์ก็จะทำการจัดเก็บและกรองเอาเฉพาะเหตุการณ์ที่ต้องการไปเก็บไว้ในพื้นที่ที่ได้กำหนดไว้ ซึ่งไฟล์ล็อกที่ได้จะแสดงในรูปที่ 4.4



รูปที่ 4.4 แสดงไฟล์ล็อกที่รับมาจากเครื่องภายในระบบที่ส่งมา

4.3 การทดสอบที่ 2 ทดสอบการซิ้งค์เวลาระหว่าง เอ็นทีพี เซิพเวอร์ และ เอ็นทีพี ไคล เอนท์

4.3.1 วิธีทดสอบ

4.3.1.1 ทำการคอนฟิกเอ็นทีพี ให้กับเซ็ลทรัล ล็อก เซิพเวอร์

4.3.1.2 ทำการคอนฟิกเอ็นทีพี ไคลแอนให้ซิ้งค์เวลาจากเซ็ลทรัล ล็อก เซิพเวอร์

4.3.2 ผลการทดสอบ

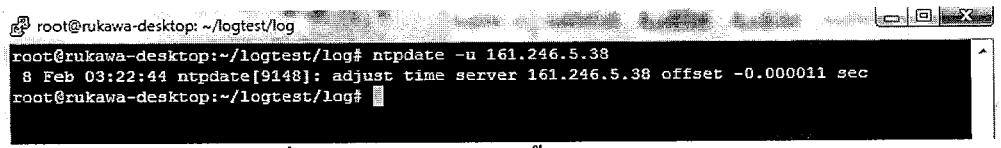
4.3.2.1 ทดลองทำการคอนฟิกเอ็นทีพี ให้กลับเซ็ลทรัล ล็อก เซิพเวอร์ เพื่อที่จะให้เครื่องภายในระบบมาขอซิ้งค์เวลาจากเซ็ลทรัล ล็อก เซิพเวอร์ ซึ่งมีการคอนฟิกดังรูปที่ 4.5

รูปที่ 4.5 แสดงการคอนฟิกเอ็นทีพีให้กลับเซ็ลทรัล ล็อก เซิพเวอร์

หลังจากทำการคอนฟิกเสร็จก็ให้ทำการรีสตาร์ทเอ็นทีพี เพื่อให้ระบบเริ่มทำงาน โดยใช้คำสั่งดังนี้

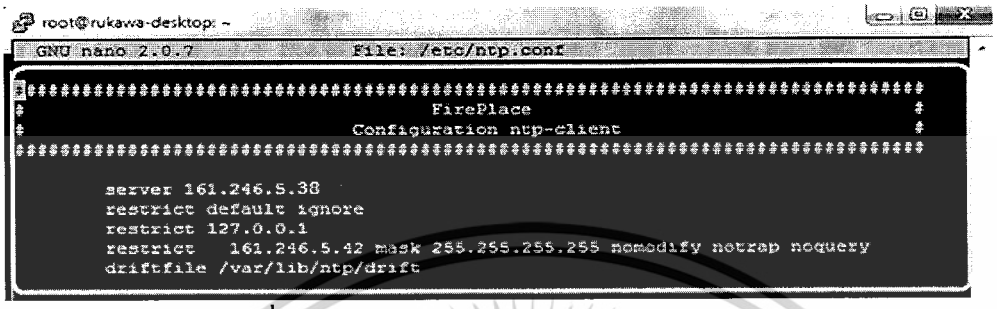
#/etc/init.d/ntp restart

ผลที่ได้จากการคอนฟิกเอ็นทีพีที่เซ็ลทรัล ล็อก เซิพเวอร์



รูปที่ 4.6 แสดงผลของการซิ้งค์เวลาจากสเตตัม 0

4.3.2.2 ทดลองทำการคอนฟิกเอ็นทีพี ให้กลับไคลแอนหรือเครื่องภายในระบบเพื่อที่จะไปซิ้งค์เวลาจากเซ็ลทรัล ล็อก เซิพเวอร์ ซึ่งจะทำให้ในส่วนของเวลาจะตรงตามที่ พรบ. ได้ระบุเอาไว้ ซึ่งจะแสดงดังรูปที่ 4.7



รูปที่ 4.7 แสดงการคอนฟิกเอ็นทีพีให้กลับไคลเอนท์

หลังจากทำการคอนฟิกเสร็จก็ให้ทำการรีสตาร์ทเอ็นทีพี เพื่อให้ระบบเริ่มทำงาน โดยใช้คำสั่งดังนี้

#/etc/init.d/ntp restart

4.4 การทดสอบที่ 3 ทดสอบการเข้ารหัสไฟล์ล็อก

4.4.1 วิธีทดสอบ

4.4.1.1 ทดลองการแฮซ (hash) ข้อมูลด้วยเอ็มดี ไฟร์ (md5)

4.4.1.2 ทดลองการเอนคลิบชั่น ข้อมูลด้วยโอเพ่นแอสแอสแอล (Openssl)

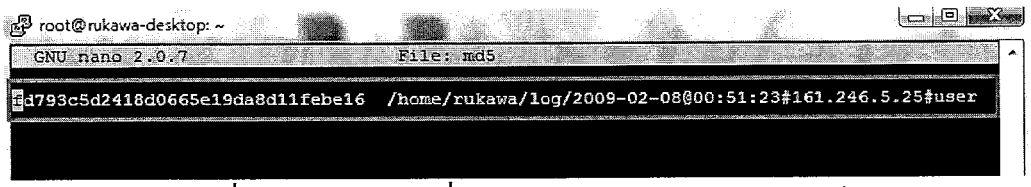
4.4.1.3 ทดลองการนำข้อมูลที่เอนคลิบชั่นแล้วมารวมกับค่าแฮซ และนำข้อมูลที่ได้มาเอน-คลิบชั่นเพื่อให้เข้าถึงเนื้อข้อมูลได้ยากยิ่งขึ้น

4.4.2 ผลการทดสอบ

4.4.2.1 ทดลองการแฮซข้อมูลด้วยเอ็มดี ไฟร์ เพื่อที่จะให้ได้ค่าแฮซ ไว้สำหรับตรวจเช็คว่าข้อมูลที่ต้องการตรวจสอบนั้นได้ถูกทำการเปลี่ยนแปลงหรือแก้ไขข้อมูลหรือไม่ ซึ่งการทดลองนี้เราจะใช้คำสั่งดังนี้ ในการแฮซ

#md5sum 2009-02-08@00:51:23#161.246.5.25#user

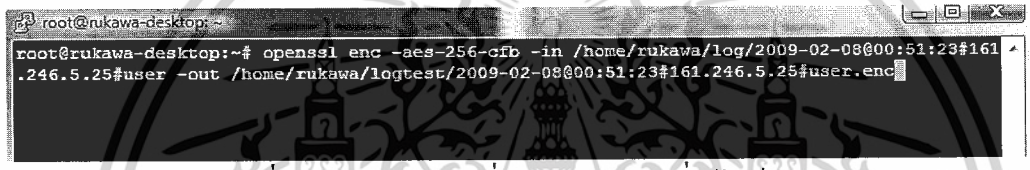
หลังจากที่ใช้คำสั่งนี้แล้วผลลัพท์ที่ได้จะแสดงในรูปที่ 4.8



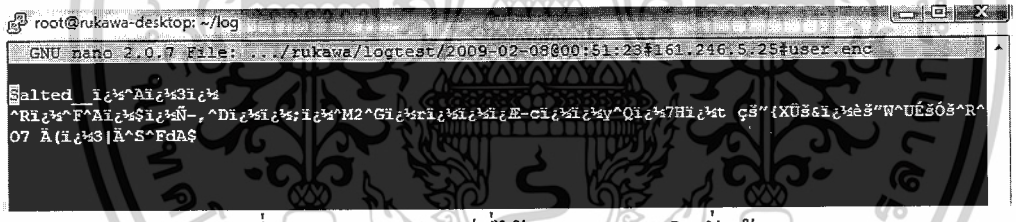
รูปที่ 4.8 แสดงผลลัพท์ที่ได้จากการแฮซข้อมูลด้วย เอ็มดี ไฟร์

ซึ่งค่าที่ได้จากการแฮซนี้จะนำไปใช้ในการตรวจตอบความถูกต้องของข้อมูลตาม พรบ. ในเรื่องของความถูกต้องของข้อมูล

4.4.2.2 ทดลองนำเอาข้อมูลมาเอนคลิบชั่น เพื่อที่จะใช้เป็นตัวป้องกันการเข้าถึงเนื้อข้อมูลภายใน และทำให้ยากต่อการแก้ไข ซึ่งเราได้ทดลองโดยการใช้โปรแกรมโอเพ่นเอสเอสแอล และผลที่ได้จากการทดลองจะแสดงดังรูปที่ 4.9

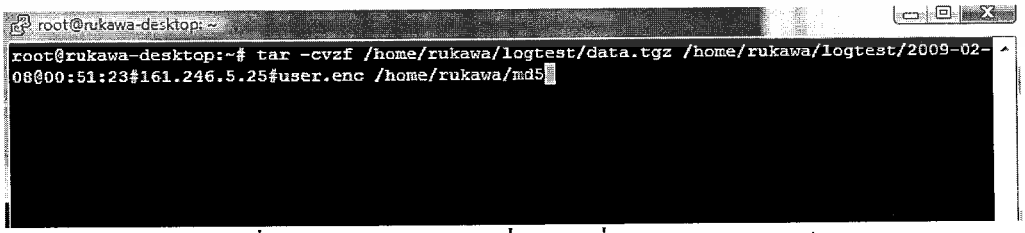


รูปที่ 4.9 แสดงการใช้คำสั่งในการเอนคลิบชั่น ไฟล์ล็อก



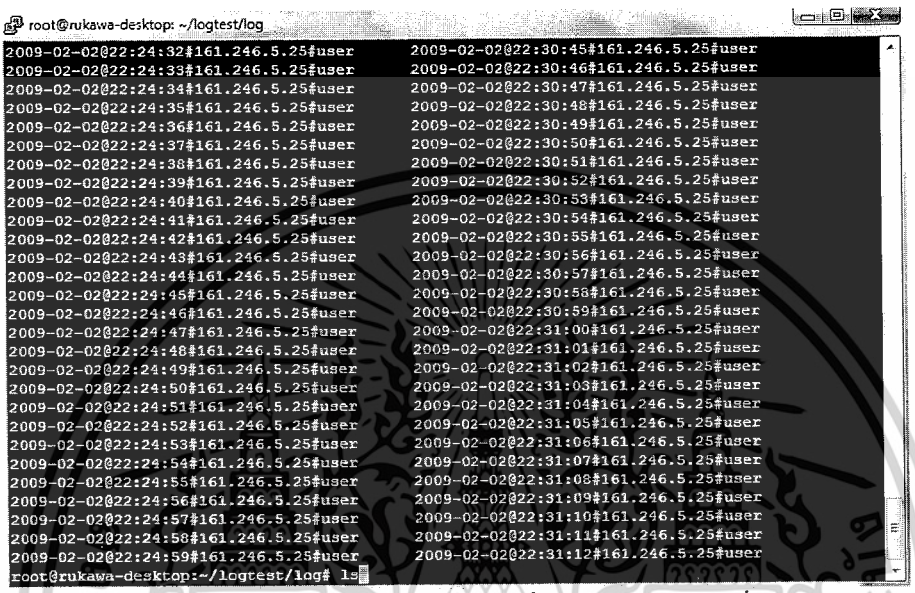
รูปที่ 4.10 แสดงผลลัพท์ที่ได้จากการเอนคลิบชั่นข้อมูล

4.4.2.3 ทดลองการนำเอาผลลัพท์ที่ได้จากการทดลองที่ 4.4.2.1 และ 4.4.2.2 มารวมกันโดยใช้โปรแกรม ทาร์ (tar) ในการรวมไฟล์สองไฟล์นี้เข้าด้วยกันเพื่อที่จะใช้เป็นตัวเปรียบเทียบข้อมูลกับค่าแฮซ ว่าตรงกันหรือไม่ ซึ่งใช้รูปแบบคำสั่งทาร์ ในการรวมไฟล์ดังรูปที่ 4.11



รูปที่ 4.11 แสดงรูปแบบคำสั่งทาร์ เพื่อใช้ในการรวมไฟล์

หลังจากที่ทำการรวมไฟล์เสร็จแล้วตามการทดลองที่ 4.4.2.3 เราก็จะนำผลลัพท์ที่ได้นั้นมาทำการเข้ารหัสหรือเอนคลิบชั่นอีกครั้งเพื่อใช้ป้องกันการเข้าถึงเนื้อข้อมูล และการแก้ไขให้ยากขึ้น ซึ่งผลที่ได้จะแสดงในรูปที่ 4.12 ซึ่งเป็นการสั่งงานสคลิบไฟล์ที่เราได้จากการสร้างของโปรแกรมของเราขึ้นมา ซึ่งข้อมูลที่ได้จะถูกเอนคลิบไว้ทุกๆ ไฟล์ล็อกที่เข้ามา



รูปที่ 4.12 แสดงผลการเอนคลิบไฟล์ล็อกที่ได้จากการทดลองที่ 4.4.2.3

และเมื่อเราทำการทดลองตรวจสอบดูผลในการเอนคลิบไฟล์ที่ได้จากการทดลองที่ 4.4.2.3 ก็จะแสดงได้ดังรูปที่ 4.13

รูปที่ 4.13 แสดงผลการตรวจสอบเนื้อข้อมูลที่ได้เอ็นคลิบแล้ว

4.5 การทดสอบที่ 4 ทดสอบการกรองข้อมูลในไฟล์ล็อกเพื่อนำเข้าดาต้าเบส

4.5.1 วิธีทดสอบ

4.5.1.1 ทดลองการกรองข้อมูลในไฟล์ล็อกเพื่อนำไปอัพเดทฐานข้อมูล

4.5.1.2 ทดลองการกรองข้อมูลเพื่อนำไปใช้ในการเข้ารหัส

4.5.2 ผลการทดสอบ

4.5.2.1 ทดลองการกรองข้อมูลในไฟล์ล็อกเพื่อที่จะแยกเอาแต่ส่วนที่ต้องการออกมาอย่าง เช่น หากเกิดเหตุการณ์หนึ่งๆจะมีข้อมูลที่เกิดซ้ำกันหรือเวลาตรงกัน ซึ่งเราไม่ต้องการก็เลยทำการสร้างไฟล์โปรแกรมขึ้นมาเพื่อใช้ให้กรองข้อมูลไฟล์ล็อกที่ได้รับมาในส่วนที่ต้องการหรือส่วนที่ข้อมูลไม่ซ้ำกัน แล้วจึงนำไปอัพเดทที่ฐานข้อมูล โดยโปรแกรมที่ได้จะแสดงในรูปที่ 4.14

รูปที่ 4.14 แสดงผลโปรแกรมที่ใช้สำหรับกรองไฟล์ล็อกเพื่ออัพเดทฐานข้อมูล

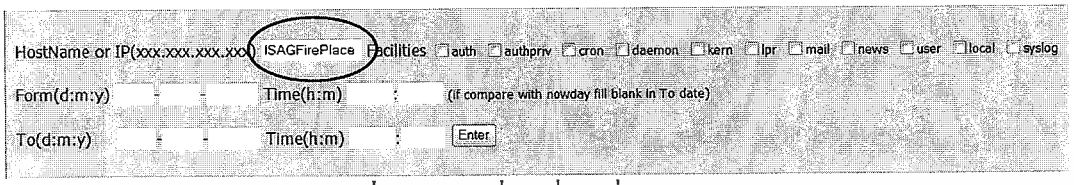
4.5.2.2 ทดลองการกรองข้อมูลเพื่อนำไปเข้ารหัส ซึ่งเราจะทำการกรองเพื่อเอาเฉพาะชื่อของไฟล์ล็อกแต่ละไฟล์มาใช้ในการตั้งชื่อไฟล์ล็อกที่ถูกเข้ารหัสแล้ว เพื่อความสะดวกในการเรียกดูไฟล์ล็อกนั้นๆ ซึ่งจะทำให้ง่ายต่อการจัดการดูแลและไม่ซ้ำซ้อนกันอีกด้วย โดยเราจะทำการสร้างโปรแกรมขึ้นมาเพื่อใช้ในการกรองและเอาผลลัพท์ที่ได้นั้นไปใช้ในการตั้งชื่อไฟล์ล็อกที่เข้ารหัสแล้วต่อไป โดยโปรแกรมที่ได้จะแสดงในรูปที่ 4.15

รูปที่ 4.15 แสดงผลโปรแกรมที่ใช้กรองเพื่อนำไปใช้ในการตั้งชื่อไฟล์ล็อก

4.6 การทดสอบที่ 5 ทดสอบการเรียกค้นและสรุปข้อมูลบนเวบแอบพลิเคชั่น

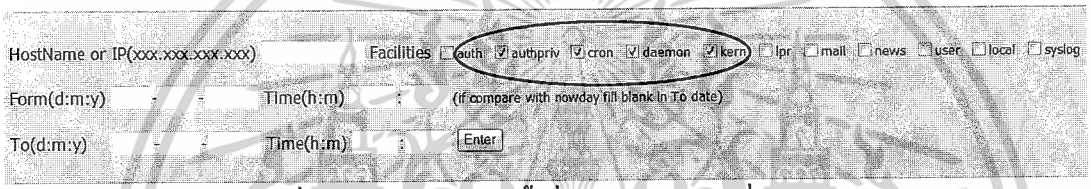
4.6.1 วิธีทดสอบ

4.6.1.1 ทดลองใส่ชื่อเครื่องในระบบที่ต้องการเรียกดูข้อมูล



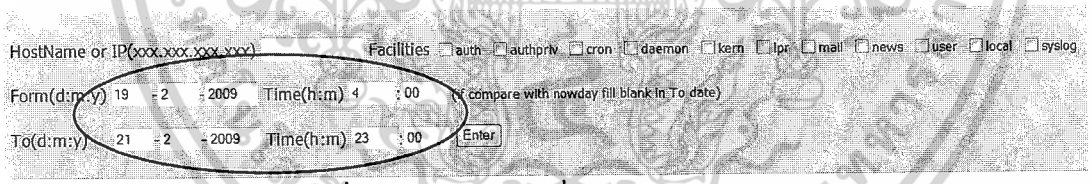
รูปที่ 4.16 ระบุชื่อเครื่องเพื่อใช้ดูข้อมูล

4.6.1.2 ทดลองเลือกแฟคซิลิตี้ (Facilities) ที่ต้องการเรียกดู



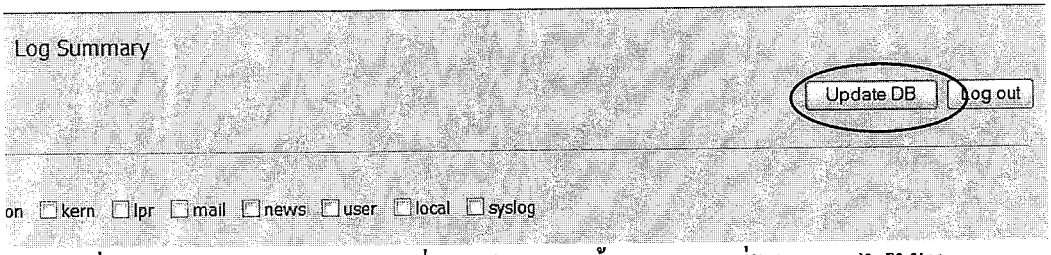
รูปที่ 4.17 เลือกแฟคซิลิตี้เพื่อใช้ดูเหตุการณ์ที่เกิดล็อก

4.6.1.3 ทดลองใส่วันเวลาที่ต้องการเรียกดู



รูปที่ 4.18 ระบุวันเวลาที่ต้องการดูข้อมูลล็อก

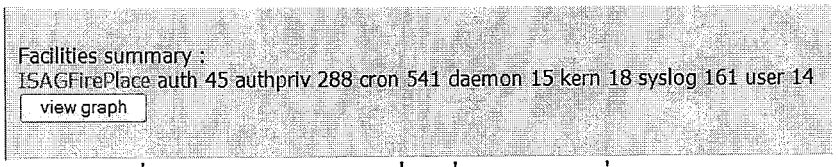
4.6.1.4 ทดลองอัพเดท (Update) ดาต้าเบส ในกรณีมีไฟล์ล็อกเกิดขึ้นใหม่ภายในเวลาที่โปรแกรมยังไม่อัพเดท



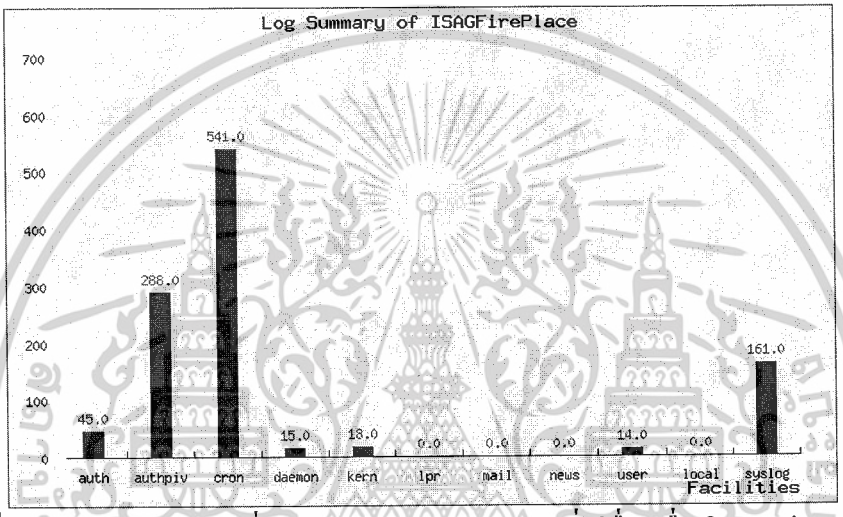
รูปที่ 4.19 อัพเดทดาต้าเบสกรณีที่มีไฟล์ล็อกเกิดขึ้นใหม่ขณะที่โปรแกรมไม่ได้อัพเดท

4.6.2 ผลการทดสอบ

4.6.2.1 การใส่ชื่อเครื่องในระบบที่ต้องการเรียกดูข้อมูล จะทำให้ทราบถึงแฟคซิลิตี้ที่มีในระบบและสรุปยอดของแฟคซิลิตี้ต่างๆ พร้อมทั้งกราฟ

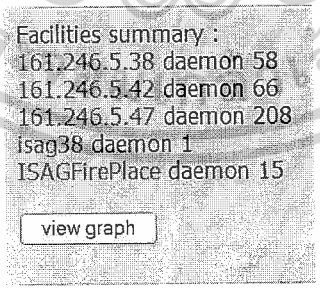


รูปที่ 4.20 แสดงผลการระบุชื่อเครื่องในระบบเพื่อเรียกดูล็อก

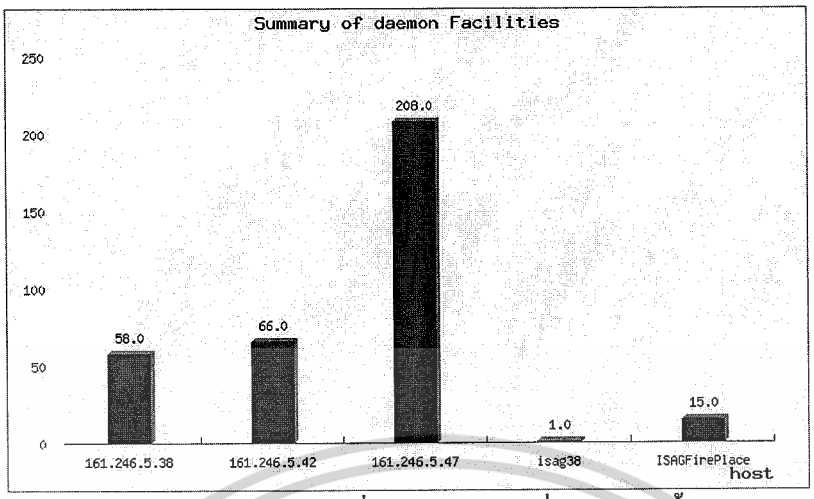


รูปที่ 4.21 แสดงข้อมูลล็อกที่เรียกดูในรูปแบบกราฟ กรณีที่ใส่ชื่อเครื่องในการเรียกดูล็อก

4.6.2.2 การใส่แฟคซิลิตี้ที่ต้องการเรียกดู จะทำให้ทราบถึงเครื่องที่มีในระบบและยอดของแฟคซิลิตี้ในเครื่องนั่นๆพร้อมทั้งแสดงเป็นรูปแบบกราฟ



รูปที่ 4.22 แสดงผลของการระบุเป็นแฟคซิลิตี้เพื่อดูข้อมูล



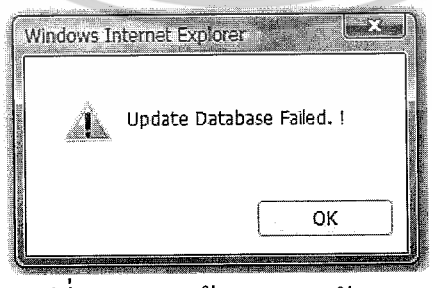
รูปที่ 4.23 แสดงข้อมูลล็อกในรูปแบบที่เป็นกราฟ กรณีที่ใส่แฟคซิลิตี้ในการเรียกดูล็อก

4.6.2.3 การใส่วันเวลาที่ต้องการเรียกดู จะทำให้ทราบถึงแฟคซิลิตี้ของเครื่องทั้งหมดที่มีในระบบพร้อมทั้งสรุปยอดของแฟคซิลิตี้นั้น

```
Facilities summary :
161.246.5.38 auth 4 authpriv 46 cron 7 daemon 43 kern 14 syslog 30 user 4
161.246.5.42 auth 9 authpriv 231 cron 83 daemon 16 kern 37 syslog 153 user 2
161.246.5.47 auth 39 authpriv 132 cron 82 daemon 158 kern 5 syslog 21 user 1
isag38 authpriv 5 cron 3 daemon 1 syslog 4
isag42 authpriv 1
isag47 authpriv 1 cron 1
ISAGFirePlace auth 41 authpriv 261 cron 486 daemon 13 kern 16 syslog 74 user 12
```

รูปที่ 4.24 แสดงผลของการระบุวันเวลาเพื่อใช้เรียกดูข้อมูล

4.6.2.4 การอัพเดทดาต้าเบส ถ้าอัพเดทสำเร็จจะรีไดเร็ค (redirect) กลับไปที่หน้ารอรับอินพุท (input) แต่ถ้าอัพเดทไม่สำร็จ จะแสดงข้อความ อัพเดท ดาต้าเบส เฟล (Update Database Failed. !) แล้วจึงค่อยกลับไปที่หน้ารอรับอินพุท



รูปที่ 4.25 แสดงข้อความการอัพเดท

บทที่ 5
บทวิจารณ์และสรุป

5.1 วิเคราะห์และสรุปผลการทดลอง

วิเคราะห์ผลการทดลอง

จากการทดลองที่ได้ทดสอบมาเป็นที่น่าพอใจ เนื่องจากระบบที่เตรียมไว้นั้นสามารถทำการเก็บข้อมูลหลักฐานได้อย่างครบถ้วน และสามารถแสดงผลออกมาถูกต้องตามที่ออกแบบไว้ มีหลักฐานเพื่อเอาผิดผู้บุกรุกซึ่งเป็นหลักฐานที่เป็นไปตาม พรบ.คอมพิวเตอร์ 2550 อีกทั้งยังสามารถทราบถึงพฤติกรรมของผู้บุกรุกได้ในรูปแบบต่างๆ ที่ผู้บุกรุกใช้โจมตี

5.2 ปัญหาและอุปสรรค

ในช่วงการพัฒนาชุดโปรแกรมรวบรวมและจัดการไฟล์ล็อกเพื่อการรักษาความปลอดภัยนั้น ได้ประสบปัญหาต่างๆ หลายประการ ซึ่งได้รวบรวมมาและสรุปเป็นข้อๆ ดังนี้

5.2.1 เนื่องจากโครงงานนี้มีส่วนเนื้อหาซึ่งเกี่ยวข้องกับ พรบ.คอมพิวเตอร์ 2550 ทำให้ต้องศึกษาเนื้อหาใน พรบ. อย่างละเอียดที่เป็นตัวบทกฎหมาย ซึ่งต้องใช้เวลาในการพิจารณาเป็นพิเศษเพราะต้องเป็นไปตามกฎหมาย

5.2.2 ในการพัฒนาโปรแกรมนั้นจะมีการทำงานสอดคล้องกันระหว่างหลายภาษา ทำให้รูป-แบบของตัวแปรไม่ตรงกัน เช่น ตัวแปรเวลาที่ได้มาจากสคลิบการกรองนั้น ไม่อยู่ในรูปแบบของฐานข้อมูล ทำให้จะต้องทำการแก้ไขก่อนที่จะอัพเดทฐานข้อมูล

5.2.3 ในเรื่องของการทำแอคเซสคอนโทรล (Access Control) สามารถที่จะทำได้ยากในเรื่องของการป้องกันการแก้ไขไฟล์ล็อก ของผู้ใช้ที่มีสิทธิเป็นรูท (root) เนื่องจากสิทธิที่รูทได้ในระบบลีนุกซ์นั้นเป็นผู้ที่มีสิทธิสูงสุดในการแก้ไขหรือจัดการเกี่ยวกับระบบได้ทั้งหมด ในส่วนของเรื่องนี้จึงได้แค่พยายามให้รูทสามารถเข้าถึงเนื้อข้อมูลล็อกโดยใช้เวลานานหรือยากที่สุดเท่าที่จะทำได้

5.2.4 ในการเขียนเวบแอพพลิเคชั่นนั้นจะพัฒนาด้วยภาษาพีเอชพี ซึ่งจะต้องมีการติดต่อกับไฟล์ในระบบปฏิบัติการลีนุกซ์ ซึ่งเป็นคำสั่งที่นอกเหนือจากการพัฒนาเวบแอพพลิเคชั่นทั่วๆไป ทำให้เมื่อพัฒนาต้องหาแหล่งอ้างอิงหรือแหล่งความรู้ต่างๆเพิ่มเติม

5.2.5 โครงงานนี้จะต้องมีการทำงานสอดประสานกับโครงงานชุดโปรแกรมรักษาความปลอด-ภัยสำหรับคอมพิวเตอร์เซิพเวอร์ และชุดโปรแกรมรักษาความปลอดภัยสำหรับคอมพิวเตอร์ไคล-เอนท์ ทำให้ต้องคิดรูปแบบในการรับ-ส่งไฟล์ล็อกและยังต้องออกแบบโครงสร้างระบบเครือข่ายให้มีความปลอดภัย

5.3 แนวทางการพัฒนาต่อ

5.3.1 พัฒนาระบบให้สามารถรองรับเครื่องคอมพิวเตอร์ให้ได้มากขึ้นในพื้นฐานของตัวระบบที่เหมือนเดิม ซึ่งสามารถทำได้หากมีการปรับปรุงโค๊ดของระบบและมีฮาร์ดแวร์ที่มีประสิทธิภาพมากพอในการพัฒนา

5.3.2 ทำการปรังปรุงโค๊ดของระบบที่ได้เขียนให้มีประสิทธิภาพและรัดกุมมากขึ้น รวมทั้งเพิ่มการดักเออเรอร์ (error) หรือเอ็กซ์เซ็บชั่น (exception) เข้าไปด้วย

5.3.3 ทำการปรับปรุงในส่วนของการรับไฟล์ล็อกของระบบไม่ให้สูญหาย และสามารถรองรับได้ขณะที่มีไฟล์ล็อกส่งเข้ามามากๆ

5.3.4 ทำการปรับปรุงเวบแอพพลิเคชั่น ให้มีความสมบูรณ์มากยิ่งขึ้น เช่น การคอนฟิกระบบผ่านเวบแอพพลิเคชั่น

5.3.5 พัฒนาในส่วนของการคอนฟิกระบบให้เป็นจียูไอ (GUI) เพื่อให้มีความสะดวกในการใช้งานและทำให้สวยงามมากยิ่งขึ้น

5.3.6 ทำการปรับปรุงระบบให้มีความปลอดภัยมากยิ่งขึ้น

บรรณานุกรม

กระทรวงเทคโนโลยีสารสนเทศและการสื่อสาร.2550. **พระราชบัญญัติ ว่าด้วยการกระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550.** กรุงเทพฯ

ทวีชัย สนธิลัคนากุล และ ธนพัฒน์ เหลืองรุ่งเรือง. 2549. **"วิศวกรรมซอฟต์แวร์เพื่อการรักษาความ ปลอดภัย."** ปริญญานิพนธ์วิศวกรรมศาสตร์ สาขาวิชาวิศวกรรม คอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

ธีรยุทธ ดำรงตระกูลเจริญ และ ปิติพล พลพบูล. **"ชุดโปรแกรมทดสอบความปลอดภัยระบบคอมพิวเตอร์."** ปริญญานิพนธ์วิศวกรรมศาสตร์ สาขาวิชาวิศวกรรม คอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

ภูวดล ด่านระหาญ. 2544. **ทำความรู้จักกับ syslogd**. [OnLine].Avaliable:http://thaicert.nectec.or.th/ paper/unix_linux/linux_syslog.php

ภูวดล ด่านระหาญ. 2546. **Syalog-ng (Syslog new generation)**. [Online].Avaliable: http://thaicert.nectec.or.th/paper/unix_linux/syslog-ng.php

Henry F. Korth, Abraham Silberschatz and S. Sudarshan. 2006. **Database System Concepts.** 5th. Singapore:Mc Graw Hill.

Douglas E. Comer. 2006. **Internetworking with TCP/IP Principles, Protocols, and Architecture Volume1.** 5th. London:Pearson Education.

ภาคผนวก ก

การเตรียมการทดลอง

1. ติดตั้งโปรแกรม

เราจะใช้โปรแกรม วีเอม แวร์ (VMware) ในการจำลองเซนทรัลไลซ์ ล็อก เซิพเวอร์ และใช้ระบบปฏิบัติการ ลินุกซ์ อูบุนตุ (ubuntu) 8.04 เป็นระบบปฏิบัติการหลัก ซึ่งเปิดให้ดาวน์โหลดฟรีได้ที่ http://www.ubuntu.com/getubuntu/download หลังจากที่ติดตั้งระบบปฏิบัติการเสร็จเรียบร้อยแล้วจึงติดตั้งโปรแกรมที่อื่นๆต่อไป ซึ่งโปรแกรมที่จะติดตั้งมีดังนี้

1.1 syslog-ng เป็นล็อก เดมอน ที่จะใช้เป็นตัวรวบรวมและจัดเก็บ ล็อกไฟล์จากเครื่องไคลแอน

ขั้นที่ 1 ถอนการติดตั้ง syslogd ตัวเดิมออกไปก่อน

#aptitude purge sysklogd

ขั้นที่ 2 ติดตั้งโปรแกรม syslog-ng

#apt-get insatall syslog-ng

ขั้นที่ 3 ปรับแต่งคอนฟิกต่างๆตามรูป

```
options {
        chain_hostnames(0);
        time_reopen(10);
        time_reap(360);
        log_fifo_size(2048);
        create_dirs(yes);
        use_dns(no);
        stats_freq(0);
        bad_hostname("^gconfd$");
};
source s_remote { internal(); udp(); };
destination d_log3 {
        file("/home/logadmin/log/$YEAR-$MONTH-$DAY@$HOUR:$MIN:$SEC#$HOST\
        #$FACILITY" owner(logadmin) group(logadmin) perm(0600)\
        dir_owner(logadmin)dir_group(logadmin) dir_perm(600));
};
log { source(s_remote); destination(d_log3); };
```

รูปที่ ก.1 การคอนฟิก ล็อก เซิพเวอร์

ขั้นที่ 4 เริ่มเซอร์วิสใหม่อีกครั้ง

#/etc/init.d/syslog-ng restart

1.2 เอนทีพี เป็นโปรแกรมสำหรับใช้เทียบ ล็อก เซิพเวอร์ กับ กับ เอนทีพีเซิพเวอร์ สะเตตัม 0

ขั้นที่ 1 ติดตั้งโปรแกรมเทียบเวลา เอนทีพี

#apt-get install ntp

ขั้นที่ 2 ปรับแต่งคอนฟิก เอนทีพี เซิพเวอร์ ดังรูป

```
restrict default kod nomodify notrap noquery nopeer
restrict 161.246.5.0 mask 255.255.255.0 nomodify notrap
server time.navy.mi.th
server 203.185.69.60
server clock.nectec.or.th
server clock2.nectec.or.th
server 127.127.1.0 # local clock
fudge 127.127.1.0 stratum 2
driftfile /var/lib/ntp/drift
keys /etc/ntp/keys
```

รูปที่ ก.2 การคอนฟิก เอนทีพีเซิพเวอร์

ขั้นที่ 3 เริ่มเซอร์วิสใหม่อีกครั้ง

#/etc/init.d/ntp restart

1.3 โอเพ่นเอสเอสแอล (openssl) โปรแกรมนี้เป็นโรแกรมสำหรับเข้ารหัสล็อกไฟล์ ซึ่งโดยปกติแล้ว ระบบปฏิบัติการ อูบุนตุ เวอร์ชันนี้จะติดตั้งโปรแกรมมาให้อยู่แล้ว

#apt-get install openssl

1.4 เอมดีไฟว์ซัม (md5sum) โปรแกรมนี้เป็นโปรแกรมที่ใช้ทำการหาค่าแฮชฟังก์ชัน ซึ่งจะใช้ในการเทียบความถูกต้องของข้อมูลและโดนปกติแล้ว ระบบปฏิบัติการ อูบุนตุ เวอร์ชันนี้จะติดตั้งโปรแกรมมาให้อยู่แล้ว

#apt-get install md5sum

1.5 ทาร์ (tar) โปรแกรมนี้เป็นโปรแกรมที่ใช้ในการบีบอัดไฟล์ และรวมไฟล์หลายๆไฟล์ให้เป็นไฟล์เดียว ซึ่งจะใช้ในการรวมล็อกไฟล์และค่าแฮชที่ได้จากล็อกไฟล์นั้นเข้าด้วยกัน โดยปกติแล้ว ระบบปฏิบัติการ อูบุนตุ เวอร์ชันนี้จะติดตั้งโปรแกรมมาให้อยู่แล้ว

#apt-get install tar

1.6 เอสเอ็ชซี (shc) โปรแกรมนี้เป็นโปรแกรมที่ใช้ในการเปลี่ยนรูปแบบไฟล์จากไฟล์ เชลล์สคริป ให้เป็นไบนารีไฟล์

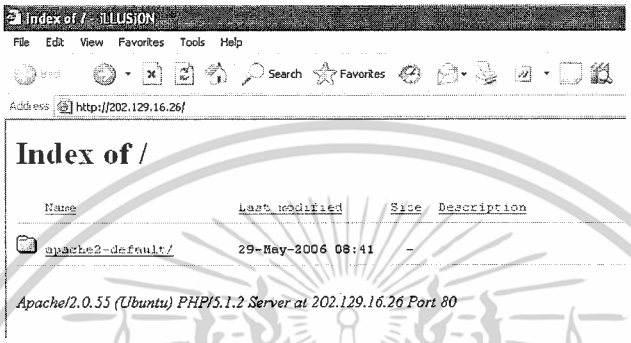
#apt-get install shc

1.7 อาร์พาเช 2 (apache2) เป็นโปรแกรมที่ทำหน้าที่เป็นเวปเซิพเวอร์

#apt-get install apache2

#/etc/init.d/apache2 restart

สามารถตรวจสอบการทำงานได้โดยทดลองเรียกใช้งานเวปเซิพเวอร์ ดังรูป



รูปที่ ก.3 ตัวอย่างเวบเซิพเวอร์

1.8 พีเอชพี5 (php5) เป็นโปรแกรมสำหรับอ่านภาษาพีเอชพี

#apt-get install php5

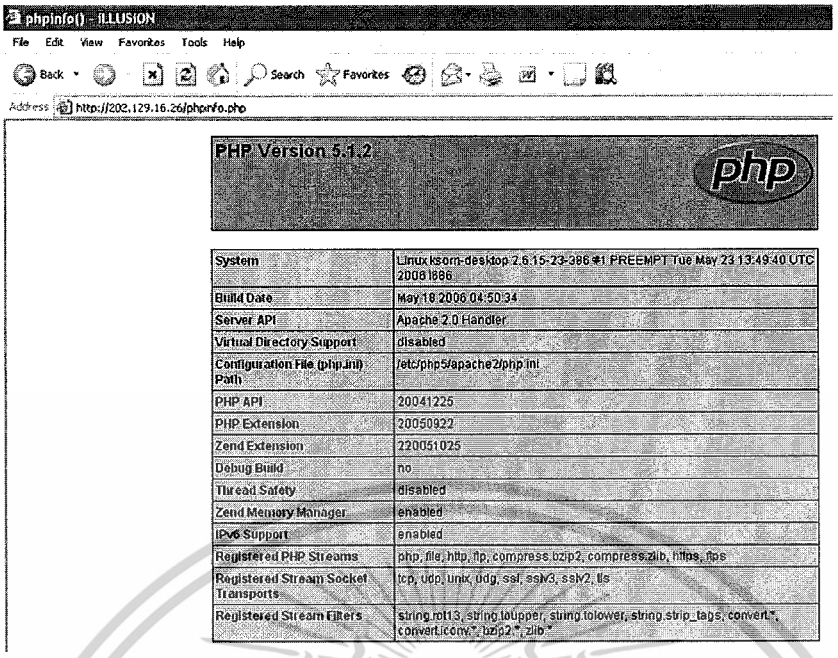
ทดสอบการทำงานด้วยการสร้างไฟล์ที่เป็นพีเอวพีไฟล์หนึ่งไว้ในตำแหน่งข้อมูลของเวปเซิพเวอร์ ซึ่งอยู่ที่ /var/www ด้วยคำสั่ง:

#sudo nano /var/www/phpinfo.php

โดยให้มีข้อความต่อไปนี้ในไฟล์

```
<?
phpinfo().php
?>
```

และเมื่อลองเรียกใช้งานไฟล์ดังกล่าวดูจะได้ดังรูป



System	Linux ksorn-desktop 2.6.15-23-386 #1 PREEMPT Tue May 23 13:49:40 UTC 2006 i686
Build Date	May 18 2006 04:50:34
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2/php.ini
PHP API	20041225
PHP Extension	20050922
Zend Extension	220051025
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	php, file, http, ftp, compress.bzip2, compress.zlib, https, ftps
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, sslv2, tls
Registered Stream Filters	string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, convert.iconv.*, bzip2.*, zlib.*

รูปที่ ก.4 แสดงข้อมูลของพีเอชพี

1.9 มายเอสคิวแอล5 (mysql5) เป็นโปรแกรมสำหรับการจัดการฐานข้อมูลสามารถจัดการได้หลายผู้ใช้งาน หลายฐานข้อมูล

#sudo apt-get install mysql-server-5.0

เนื่องจากค่าเริ่มต้น (default) ยังไม่ได้มีการกำหนดรหัสผ่านให้กับผู้ใช้ของมายเอสคิว-แอล ที่เป็นผู้ดูแลระบบ (root) ดังนั้นเราสามารถกำหนดรหัสผ่านให้กับผู้ดูแลระบบ ด้วยคำสั่ง

#sudo mysqladmin -u root password <password>

และเมื่อทดลองเรียกใช้งานมายเอสคิวแอลจะได้ดังรูป

```
ksorn@ksorn-desktop:/var/www$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 34 to server version: 5.0.21-Debian_3ubuntu1-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

รูปที่ ก.5 ทดลองเรียกใช้งาน มายเอสคิวแอล

1.10 พีเอชพีมายแอดมิน (phpMyAdmin) ให้ดาวน์โหลดโปรแกรมมาจากอินเตอร์เน็ตที่เวป ไซต์ http://www.phpmyadmin.net โดยในตัวอย่างนี้ได้โหลดไฟล์ phpMyAdmin-2.8.0.tar.gz มาแล้วทำไปวางไว้ในตำแหน่งข้อมูลของเวปเซิพเวอร์ /var/www/ จากนั้นก็ให้แตกไฟล์ด้วยคำสั่ง

 # sudo tar xvfz phpMyAdmin-2.8.1.tar.gz

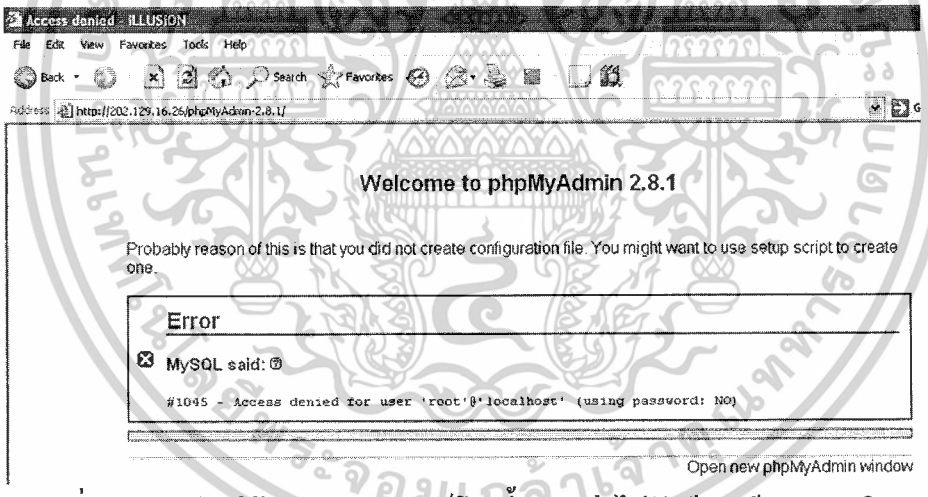
การแตกไฟล์ด้วยคำสั่งข้างบนจะมีการสร้างโฟลเดอร์ใหม่ เป็นชื่อเดียวกับไฟล์ที่เราแตกออก แล้วลองเรียกใช้งาน เวปเซิพเวอร์ โดยชี้ไปยังโฟลเดอร์ดังกล่าว ก็จะเจอความผิดพลาดดังรูป

รูปที่ ก.6 ข้อผิดพลาดที่เกิดจากการเรียกใช้เวบเซิพเวอร์

1.11 พีเอชพี5-มายเอสคิวแอล (php5-mysql) จากข้อผิดพลาดด้านบนแก้ไขโดย

#sudo apt-get install php5-mysql

จากนั้นเมื่อลองเรียกใช้งานเวปเซิพเวอร์โดยชี้ตำแหน่งไปยังโฟลเดอร์ของพีเอชพีมายแอดมิน (http://202.129.16.26/phpmyadmin-2.8.1/) ก็ได้จะได้ดังรูป



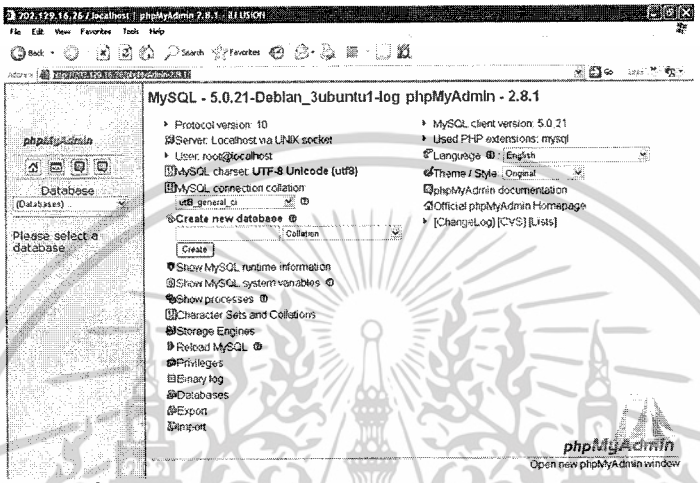
รูปที่ ก.7 การเรียกใช้งานเวบเซิพเวอร์โดยชี้ตำแหน่งไปยังพีเอชพีมายแอดมิน

ตอนนี้มายเอสคิวแอลพร้อมที่จะใช้งานได้แล้ว แต่ที่ฟ้องข้อผิดพลาดดังรูปข้างบนก็เพราะเรายังไม่ได้ปรับแต่ง พีเอชพีมายแอดมิน ให้ถูกต้อง ซึ่งวิธีการปรับแต่งอาจจะแตกต่างกันไป แต่ในที่นี้จะแสดงการปรับแต่งแบบง่าย ๆ ด้วยการกำหนดให้ พีเอชพีมายแอดมิน ติดต่อกับมายเอสคิวแอล เซิพเวอร์ ผ่าน ผู้ใช้ที่เป็นผู้ดูแลระบบ (root) ซึ่งค่าเริ่มต้นของ พีเอชพีมายแอดมิน ได้ถูกปรับแต่งให้ผู้ใช้เป็นผู้ดูแลระบบอยู่แล้วเพียงแต่ยังไม่มีการกำหนดค่าของรหัสผ่านให้ตรงกับของ มายเอสคิวแอล เซิพเวอร์ นั่นเอง นั่นคือค่าเริ่มต้นของรหัสผ่านจะเป็นค่าว่าง

การปรับแต่ง พีเอชพีมายแอดมิน ก็ให้เปิด ไฟล์คอนฟิก ขึ้นมาซึ่งเป็นไฟล์ที่อยู่ในโฟลเดอร์ libraries ชื่อว่า config.default.php (แต่ละเวอร์ชันอาจจะแตกต่างกัน) ดังนี้

#sudo nano libraries/config.default.php

แล้วให้ป้อนค่ารหัสผ่านของผู้ดูแลระบบที่บรรทัด $cfg['Servers'][$i]['password']= "" และเมื่อบันทึกไฟล์ดังกล่าวแล้วลองเรียกใช้งานใหม่จะได้ดังรูป



รูปที่ ก.8 ผลจากการปรับแต่งไฟล์ config.default.php

2. สร้างบัญชีผู้ใช้ (user account) ขึ้นมาใหม่ 1 ชื่อ เพื่อให้เป็นผู้ตรวจสอบล็อกไฟล์

```
root@ISAGFirePlace:~# adduser logadmin
Adding user `logadmin ' ...
Adding new group `logadmin ' (1003) ...
Adding new user `logadmin ' (1003) with group `logadmin ' ...
Creating home directory `/home/logadmin ' ...
Copying files from `/etc/skel' ...
Enter new UNIX password: <password>
Retype new UNIX password: <password>
passwd: password updated successfully
Changing the user information for logadmin
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
root@ISAGFirePlace:~#
```

รูปที่ ก.9 การสร้างบัญชีผู้ใช้สำหรับผู้ดูแลล็อกไฟล์

3. สร้างโปรแกรมกรองข้อมูลเพื่อบันทึกลงในเดต้าเบสและโปรแกรมเข้ารหัสไฟล์

3.1 สร้างไฟล์เชลล์สคริปมา 1 ไฟล์

#nano program.sh

3.2 เขียนโปรแกรม เชลล์สคริป

```
#!/bin/bash
while true
do
        for p in /home/rukawa/log/*
        do
        ######### filter data to database ###########
                a=$p\#
                echo p=$p
                echo a=$a
                sleep 2
                x=`sed 's/  / /g' $p | cut -d ' ' -f 5 | sed 's/:$/ /g' | uniq`
                echo x=$x
                sleep 2

                for i in $x
                do
                        tmp=$a
                        echo tmp a=$tmp
                        tmp=$tmp$i
                        echo tmp i=$tmp
                        echo $tmp >> /home/rukawa/logtest/tmp.log
                        echo tmp=$tmp
                done
         sed 's/\//|/' /home/rukawa/logtest/tmp.log |
         sed 's/\//|/' |
         sed 's/\//|/' |
         sed 's/\//|/' |
         cut -d '|' -f 5 |
         sed 's/@/ /g' |
         sed 's/--/MARK/g' >> /home/rukawa/logtest/tmp.log
         echo OK!
         sleep 2
         rm /home/rukawa/logtest/tmp.log

         ######### Encryption log-file ###########

                ###echo /home/rukawa/log/* to filter
                echo $p > /home/rukawa/logtest/log/filter

                ###filter fil 5 save to /home/rukawa/logtest/log/filter
                cut -d '/' -f 5 /home/rukawa/logtest/log/filter > /home/rukawa/logtest/log/filter1

                ###sed value in /home/rukawa/logtest/log/filter get name file
                tmp=`sed '1q' /home/rukawa/logtest/log/filter1`

                ###md5 file (Integrity)
                md5sum $p > /home/rukawa/logtest/log/md5msg

                ###Encrypt file (data)
                openssl enc -aes-256-cfb -in $p -out /home/rukawa/logtest/log/$tmp.enc -k rukawa
                tar -cvzf /home/rukawa/logtest/log/$tmp.tgz /home/rukawa/logtest/log/$tmp.enc /home/rukawa/logtest/log/md5m
                openssl enc -aes-256-cfb -in /home/rukawa/logtest/log/$tmp.tgz -out /home/rukawa/logtest/log/$tmp -k 123456
                rm /home/rukawa/logtest/log/fil*
                rm /home/rukawa/logtest/log/md*
                rm /home/rukawa/logtest/log/*.tgz
                rm /home/rukawa/logtest/log/*.enc
        done
done
```

รูปที่ ก.10 แสดงโปรแกรมเข้ารหัสไฟล์

สคริปที่เขียนมานี้ จะแยกออกเป็น 2 ส่วน คือส่วนแรกใช้สำหรับ กรองล็อกไฟล์เพื่อใช้ในการสอดแทรก (insert) ลงในเดตาเบส ส่วนที่สอง คือสคริปสำหรับเข้ารหัสไฟล์ล็อก

3.3 ทำการเปลี่ยนไฟล์ สคริป ให้เป็น ไบนารีไฟล์เนื่องจากสคริปไฟล์ที่เราเขียนขึ้นมานั้น ได้มีการ กำหนดรหัสผ่านสำหรับการเข้ารหัสไฟล์ล็อก จึงทำให้ผู้ที่เข้ามาดูสคริปนี้รู้สหัสผ่านไฟล์ล็อกและสามารถถอดรหัสไฟล์ล็อกนี้ได้

#shc –f program.sh

3.4 จะได้ไฟล์มาเพิ่มอีก 2 ไฟล์

```
program.sh
program.sh.x
program.sh.x.c
```

รูปที่ ก.11 ไฟล์ที่ได้

ไฟล์ที่ได้เพิ่มมา 2 ไฟล์ ไฟล์แรก program.sh.x จะเป็นไบนารีไฟล์ ซึ่งเราจะนำไปใช้ ส่วนไฟล์ program.sh.x.c จะแปลงซอร์ดโค๊ด (source code) จาก เชล์สคริป เป็น ซอร์ดโค๊ดภาษาซี ซึ่งเราจะลบไฟล์ program.sh และ program.sh.x.c ทิ้งไป

#rm program.sh program.sh

ภาคผนวก ข

คู่มือการใช้งานโปรแกรม

1. รายละเอียดของโปรแกรม

ตัวโปรแกรมเขียนด้วยภาษาเชลสคลิบ ซึ่งจะเป็นตัวที่ใช้ในการคอนฟิกค่าหรือปรับแต่งค่าต่างให้กับเครื่องไคลเอนท์และเซิพเวอร์ นอกจากนั้นยังสามารถที่จะใช้สร้างไฟล์ที่ใช้ในการเข้ารหัสเพื่อเอาไว้สำหรับเข้ารหัสไฟล์ล็อกหรือไฟล์ที่ต้องการ และอีกส่วนก็จะเป็นการคอนฟิกเครื่องที่ต้องการจะทำให้เป็นเซ็นทรัล ล็อก เซิพเวอร์ อีกด้วย

2. การใช้งานโปรแกรม

ในการใช้งานโปรแกรมก้อให้ทำการก๊อบปี้ตัวโปรแกรมลงไดว์ที่เราต้องการ จากนั้นให้ทำการสั่งรันโปรแกรม โดยตัวโปรแกรมมีชื่อว่า install_log-1.0.3 จากนั้นจะปรากฏหน้าต่างของตัวโปรแกรมดังนี้

รูปที่ ข.1 แสดงหน้าตาของตัวโปรแกรม

เมนูที่ 1 เป็นการคอนฟิกซิทล็อกดี ถ้าหากเครื่องไคลเอนท์หรือเครื่องเซิพเวอร์นั้นต้องการที่จะใช้ซิทล็อกดี เป็นตัวส่งล็อกหรือจัดการเกี่ยวกับล็อก

เมนูที่ 2 เป็นการคอนฟิกซิทล็อก-เอ็นจี ถ้าหากเครื่องไคลเอนท์หรือเครื่องเซิพเวอรนั้นต้องการที่จะใช้ซิทล็อก-เอ็นจี เป็นตัวส่งล็อกหรือจัดการเกี่ยวกับล็อก

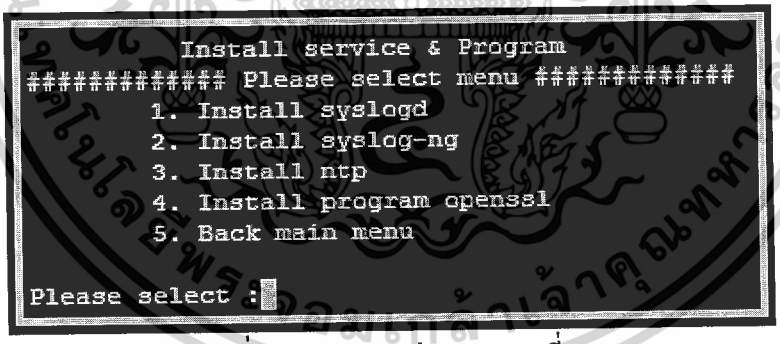
เมนูที่ 3 เป็นการคอนฟิกเอ็นทีพีให้กับเครื่องไคลเอนท์ เพื่อซิ้งค์เวลาให้ตรงกับเครื่องเซ็นทรัลล็อก เซิพเวอร์

เมนูที่ 4 เป็นการคอนฟิกเอ็นทีพีให้กับเครื่องเซิพเวอร์ เพื่อซิ้งค์เวลาให้กับเครื่องเซิพเวอร์ซึ่งจะใช้เป็นตัวอ้างอิงให้กับเครื่องภายในระบบมาซิ้งค์เวลาจากเซิพเวอร์อีกที

เมนูที่ 5 เป็นส่วนที่จะใช้ในเข้ารหัสไฟล์หรือถอดรหัสไฟล์ ในส่วนของการเข้ารหัสไฟล์นี้จะเป็นการสร้างไฟล์โปรแกรมขึ้นมา เพื่อที่จะนำไฟล์นี้ไปใช้ในการเข้ารหัส ซึ่งในเมนูนี้จะมีเมนูย่อยเพื่อให้เลือกว่าต้องการที่จะทำอะไรระหว่างเข้ารหัสกับถอดรหัส ดังรูปที่ ข.2

รูปที่ ข.2 แสดงเมนูย่อยของเมนูที่ 5

เมนูที่ 6 เป็นส่วนที่ใช้ในการติดตั้งเซอร์วิซหรือโปรแกรมต่างๆ ที่ต้องใช้ในการสร้างระบบเซ็นทรัล ล็อก เซิพเวอร์ ซึ่งจะประกอบไปด้วย ซิทล็อกดี, ซิทล็อก-เอ็นจี, เอ็นทีพี และในส่วนของตัวโปรแกรมที่ใช้ก็คือ โอเพ่นเอสเอสแอล, เอสเอชซี ซึ่งในเมนูนี้จะแบ่งย่อยออกไปเพื่อใช้ในการเลือกว่าต้องการที่จะติดตั้งอะไร ดังจะเห็นในรูปที่ ข.3



รูปที่ ข.3 แสดงเมนูย่อยของเมนูที่ 6

เมนูที่ 7 เป็นส่วนที่ใช้ในการคอนฟิกซิทล็อก-เอ็นจี ในตัวเซิพเวอร์ที่ต้องการจะทำเป็นเซ็นทรัล ล็อก เซิพเวอร์

เมนูที่ 8 ใช้ในการออกจากโปรแกรม

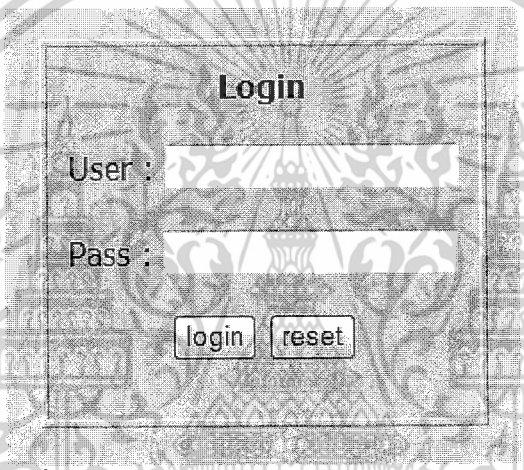
ภาคผนวก ค

คู่มือการใช้งานเวปแอพพลิเคชัน

1. การเข้าใช้งานเวบแอพพลิเคชั่น

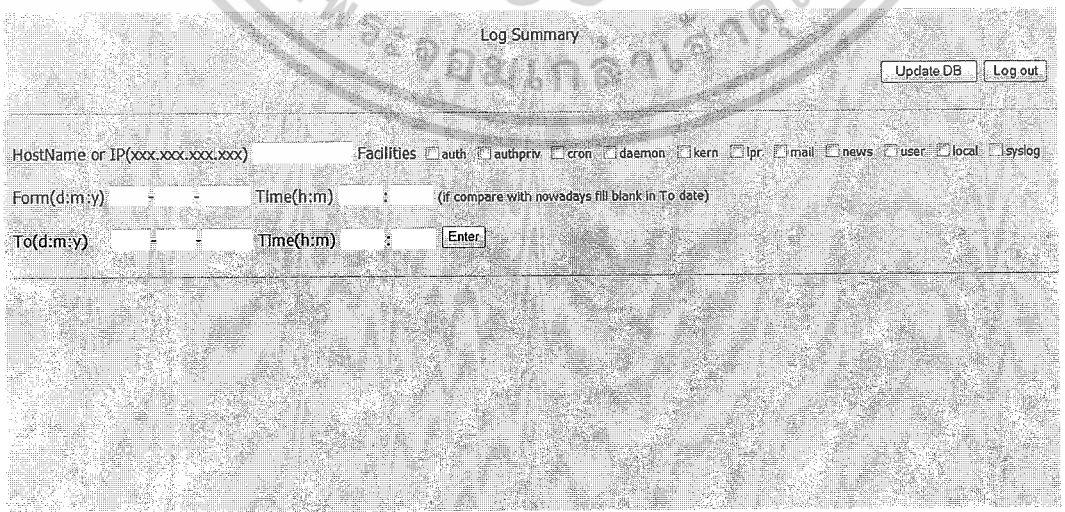
เข้าสู่เวบไซด์เพื่อทำการเรียกดูข้อมูลของระบบ โดยเข้าทาง http://161.246.5.38 โดยก่อนจะเข้าใช้งานต้องทำการล็อกอินเข้าสู่ระบบก่อนโดยใช้

- username : admin
- password : admin



รูปที่ ค.1 แสดงหน้าต่างในการล็อกอินเข้าใช้งาน

2. หน้าพื้นฐานของเวบแอพพลิเคชั่น

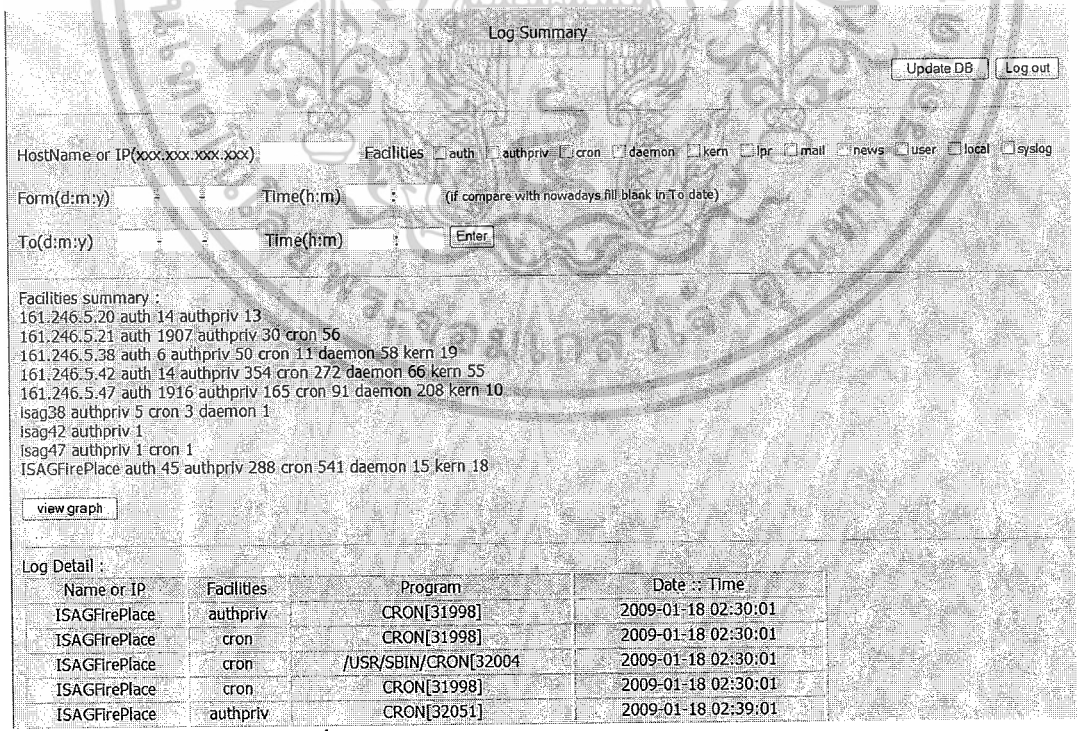


รูปที่ ค.2 แสดงหน้าพื้นฐานของเวบแอพพลิเคชั่น

3. ฟังก์ชั่นการใช้งาน

1. การเรียกดูข้อมูลไฟล์ล็อก
   - Hostname or IP เป็นช่องที่สำหรับใส่ชื่อเครื่องหรือไอพีของเครื่องที่ต้องการจะเรียกดูข้อมูลไฟล์ล็อก
   - Facilities เป็นช่องให้เลือกว่าจะเลือกดู แฟคซิลิตี้ ใด
   - From เป็นช่องให้ใส่เวลาว่าต้องการดูตั้งแต่วันไหน
   - To เป็นช่องให้ใส่เวลาว่าต้องการดูถึงวันไหน
   - Time เป็นเวลาหรือช่วงเวลาที่ต้องการเรียกดู
2. การอัพเดทดาต้าเบสไฟล์ล็อก
   - เป็นปุ่มให้กดอัพเดทดาต้าเบสไฟล์ล็อกในกรณีที่ยังไม่ทำการออโต้อัพเดท
3. การล็อกเอาท์
   - เป็นปุ่มสำหรับกดเพื่อออกจากเวบแอพพลิเคชั่น
4. การดูกราฟ
   - เป็นปุ่มสำหรับกดดูยอดสรุปของข้อมูลที่ได้จากการเรียกดูไฟล์ล็อก

4. หน้าแสดงผลการเรียกดูข้อมูลไฟล์ล็อก



รูปที่ ค.3 แสดงผลของการเรียกดูข้อมูลไฟล์ล็อก

1. Facilities summary

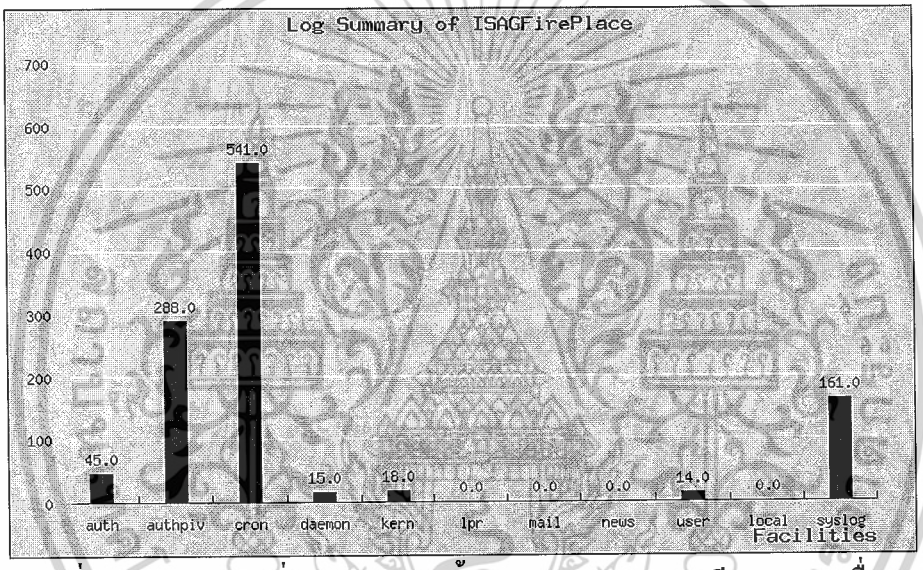
เป็นการสรุปจำนวนครั้งของ Facilities ที่เกิดขึ้นในเครื่อง

2. Log Detail

- Name or IP เป็นชื่อเครื่องหรือไอพีของเครื่อง
- Facilities เป็นชื่อ Facilities ของเมสเซส
- Program เป็นชื่อโปรแกรมที่ได้แจ้งเตือนเมสเซส
- Date :: Time เป็นเวลาที่เกิดเมสเซส

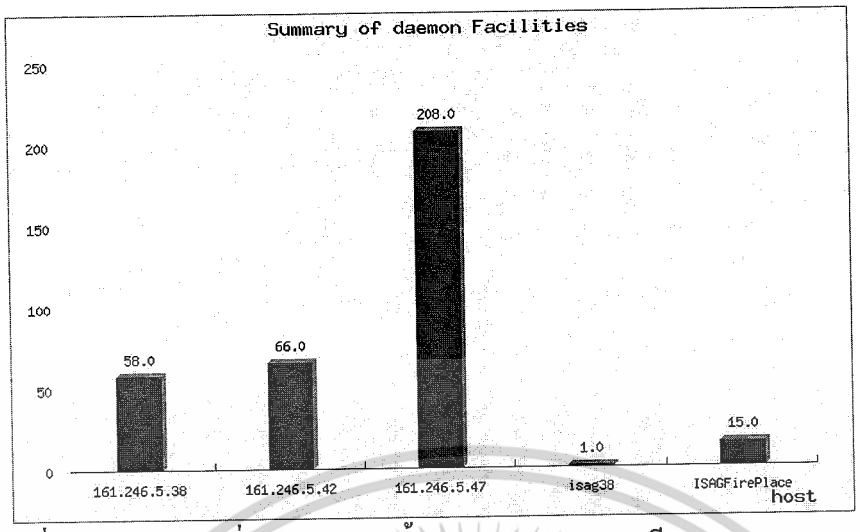
3. View graph

- แบบเรียกดูตามเครื่อง จะแสดงจำนวนครั้งของ Facilities ที่ได้แจ้งเตือนเมสเซสออกมา



รูปที่ ค.4 แสดงกราฟที่แสดงจำนวนครั้งของ Facilities แบบเรียกดูตามเครื่อง

- แบบเรียกดูตาม Facilities จะแสดงเครื่องต่างๆ ว่ามี Facilities ที่เลือกนั้นแจ้งเตือนเมสเซสออกมา



รูปที่ ค.5 แสดงกราฟที่แสดงจำนวนครั้งของ Facilities แบบเรียกดูตาม Facilities