การพัฒนาคลังข้อมูลเพื่อรองรับการจัดการข้อมูลจาก Network Log File

Data Warehouse Development for Network Log File

ธเนศ ทวีโรจน์สุพล

ประกิต วิบูลย์กาญจน์

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

ภาควิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

Data Warehouse Development for Network Log File

THANET TAWEEROJSUPON

PRAKIT VIBOONKARN

A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIRMENT FOR DEGREE OF BACHELOR OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2010

หัวข้อโครงงานพิเศษ การพัฒนาคลังข้อมูลเพื่อรองรับการจัดการข้อมูลจากเน็ตเวิร์ค Log file

Data Warehouse Development for Network Log File

ชื่อนักศึกษา นายธเนศ ทวีโรจน์สุพล รหัสนักศึกษา 50050146

นายประกิต วิบูลย์กาญจน์ รหัสนักศึกษา 50050165

ปริญญา วิทยาศาสตร์บัณฑิต

สาขาวิชา วิทยาการคอมพิวเตอร์

อาจารย์ที่ปรึกษา ผศ.กฤษฎา บุศรา

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นับโครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ประจำปีการศึกษา 2553

คณะกรรมการสอบ	ลายมือชื่อ
ดร. รุ่งรัตน์ เวียงศรีพนาวัลย์ (ประธานกรรมการ)	
อ. ศังกรศรัณย์ ล่องชูผล (กรรมการ)	
ผศ. กฤษฎา บุศรา (กรรมการและอาจารย์ที่ปรึกษา)	

หัวข้อโครงงานพิเศษ การพัฒนาคลังข้อมูลเพื่อรองรับการจัดการข้อมูลจากเน็ตเวิร์ค Log File
Data Warehouse Development for Network Log File

ชื่อนักศึกษา นายธเนศ ทวีโรจน์สุพล รหัสนักศึกษา 50050146
นายประกิต วิบูลย์กาญจน์ รหัสนักศึกษา 50050165

ปริญญา วิทยาศาสตรบัณฑิต

สาขาวิชา วิทยาการคอมพิวเตอร์

ปีการศึกษา 2553

อาจารย์ที่ปรึกษา ผศ.กฤษฎา บุศรา

บทคัดย่อ

ในปัจจุบัน พรบ. คอมพิวเตอร์ปี 2550 มาตรา 26 เนื่องด้วย พรบ. คอมพิวเตอร์ ปี 2550 มาตรา 26 ผู้ให้บริการต้องเก็บรักษาข้อมูลจราจรทางคอมพิวเตอร์ไว้ไม่น้อยกว่าเก้าสิบวันนับแต่วันที่ข้อมูลนั้นเข้าสู่ระบบคอมพิวเตอร์ โดยมีการจัดเก็บข้อมูลรายละเอียดการทำงานต่างๆ อย่างเช่น มีการทำงานอะไรบ้าง ทำงานอย่างไร ทำงานเมื่อไร และมีการส่งผ่านข้อมูลไปที่ไหนด้วยวิธีการอะไร โดยข้อมูลทุกอย่างจะถูกจัดเก็บไว้ที่ Log File ซึ่งเป็นแฟ้มข้อมูลเฉพาะเครื่องคอมพิวเตอร์แม่ข่ายดังกล่าวที่มีรูปแบบการจัดเก็บข้อมูลที่แตกต่างกัน

จากลักษณะการทำงานดังกล่าวจะมีข้อมูลจำนวนมากที่ต้องถูกนำมาทำงานร่วมกัน ดังนั้นต้องมีการพัฒนาระบบงานที่ใช้หลักการทำงานของคลังข้อมูล(Data Warehouse) ขึ้นมาเพื่อรองรับการจัดเก็บข้อมูลจำนวนมากดังกล่าว โดยจะต้องมีการออกแบบคลังข้อมูลเพื่อรองรับการจัดเก็บข้อมูลและนำหลักการทำงานของ Business Intelligence มาประยุกต์ใช้เพื่อพัฒนาระบบงานในการนำข้อมูลทั้งหมดไปวิเคราะห์เฉพาะเครื่องคอมพิวเตอร์แม่ข่าย หรือนำไปวิเคราะห์การทำงานของเครื่องคอมพิวเตอร์แม่ข่ายทั้งหมดในภาพรวมตามความต้องการของผู้ใช้งาน

การนำหลักการทำงานของ Business Intelligence มาประยุกต์ใช้เพื่อพัฒนาระบบงานในการนำข้อมูลทั้งหมดมาวิเคราะห์นั้น จะมีการออกแบบเพื่อให้สามารถนำ Log File มารวมกันซึ่งจะทำการออกแบบเพื่อให้แสดงผลได้อย่างมีประสิทธิภาพและสามารถทำความเข้าใจได้ง่าย

Title	Data Warehouse Development for Network Log File
Students	Mr.Thanet Taweerojsupon 50050146
	Mr.Prakit Viboonkarn 50050165
Degree	Bachelor of Science
Department	Faculty of Science
Major Program	Computer Science
Academic Year	2010
Advisor	Asst.Prof.Kridsada Budsara

ABSTRACT

At present, due to the Act of Computer Crime B.E.2550 (2007) under section 26, providers must keep logs of computer system's traffic data at least ninety days since the date that the system is accessed. Examples of the logged data are the type of the traffic, the location that the traffic occurs, the location that the traffic is from, the method that the traffic data is transferred into the system, and the way each device such as a firewall, a mail gateway and a server stores the traffic which is done in different formats.

This results in a large amount of data needed to be stored and managed. In this special project, we developed a system by using the concepts in "Data Warehouse" to store and manage the traffic data. We also applied the principles used in "Business Intelligence" to analyze the different types of "Log File". The result of our work is the system which can store the log files effectively. In addition, our system can produce and display traffic reports which are easy to understand.

กิตติกรรมประกาศ

การจัดทำคู่มือการทำปัญหาพิเศษหัวข้อ “การพัฒนาคลังข้อมูลเพื่อรองรับการจัดการข้อมูลจากNetwork Log file” ฉบับนี้สามารถสำเร็จลุล่วงไปได้ด้วยดี ทางคณะผู้จัดทำต้องขอขอบพระคุณบุคคลต่างๆ ที่ได้เสียสละเวลาให้คำแนะนำ และให้ความช่วยเหลือมาโดยตลอด อันได้แก่

1. ผศ. กฤษฎา บุศรา อาจารย์ที่ปรึกษาปัญหาพิเศษ ที่คอยให้ความกรุณาอย่างสูงในการให้คำปรึกษา และแนะนำแนวทางในการจัดทำปัญหาพิเศษฉบับนี้ ให้สามารถสำเร็จลุล่วงไปได้ด้วยดี

2. คุณ กฤษฎิ์ธนิก ศรีธนสาร บุคลากรที่คอยให้คำปรึกษาเกี่ยวกับปัญหาพิเศษฉบับนี้มาโดยตลอด รวมถึงคอยตรวจสอบการทำงานของอุปกรณ์เน็ตเวิร์คให้ทำงานได้อย่างมีประสิทธิภาพ

3. อาจารย์ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ทุกท่าน ที่ได้ให้การอบรมสั่งสอนความรู้ทั้งในภาคทฤษฎีและภาคปฏิบัติให้แก่ทางคณะผู้จัดทำมาโดยตลอด 4 ปี

4. บิดา มารดา ตลอดจนญาติพี่น้อง ซึ่งคอยให้การสนับสนุน ดูแล อบรมสั่งสอน และเป็นกำลังใจให้ทุกๆ เรื่องมาเสมอ

5. เพื่อนๆ ทุกคนที่คอยให้คำแนะนำ และกำลังใจมาโดยตลอด

นายธเนศ ทวีโรจน์สุพล
นายประกิต วิบูลย์กาญจน์

สารบัญ

สารบัญ (ต่อ)

สารบัญภาพ

สารบัญตาราง

บทที่ 1

บทนำ

1. ความสำคัญและที่มาของปัญหา

ในปัจจุบันองค์กรที่ให้บริการพื้นฐานทางด้านเทคโนโลยีสารสนเทศและการสื่อสาร (ICT : Information Communication Technology) ต้องมีการติดตั้งอุปกรณ์ที่ทำงานด้านระบบเครือข่าย (Network System) โดยมีการจัดเก็บข้อมูลรายละเอียดการทำงานต่างๆ อย่างเช่น มีการทำงานอะไรบ้าง ทำงานอย่างไร ทำงานเมื่อไรและมีการส่งผ่านข้อมูลว่าไปที่ไหนด้วยวิธีการอะไร โดยข้อมูลทุกอย่างจะถูกจัดเก็บไว้ที่ Log File ซึ่งเป็นแฟ้มข้อมูลเฉพาะอุปกรณ์ดังกล่าวที่มีรูปแบบการจัดเก็บข้อมูลที่แตกต่างกัน และถ้านำข้อมูลดังกล่าวมาวิเคราะห์และเชื่อมโยงทั้งหมดโดยนำ Log File ของทุกอุปกรณ์มาบูรณาการร่วมกันโดยอาศัยข้อมูล Global Time Data จาก Time Server จะก่อให้เกิดประโยชน์ที่มีความสำคัญมากสำหรับการบริหารจัดการ ICT Infrastructure เพราะข้อมูลดังกล่าวทั้งหมดสามารถนำมาวิเคราะห์ให้เห็นถึงพฤติกรรมของผู้ใช้งานในองค์กร และวิเคราะห์พฤติกรรมของการรุกรานหรือการใช้งานที่ไม่พึงประสงค์ได้

จากลักษณะการทำงานดังกล่าวจะมีข้อมูลจำนวนมากที่ต้องถูกนำมาทำงานร่วมกัน ดังนั้นต้องมีการพัฒนาระบบงานที่ใช้หลักการทำงานของคลังข้อมูล (Data Warehouse) ขึ้นมาเพื่อรองรับการจัดเก็บข้อมูลจำนวนมากดังกล่าว โดยจะต้องมีการออกแบบคลังข้อมูลเพื่อรองรับการจัดเก็บข้อมูลและนำหลักการทำงานของ Business Intelligence มาประยุกต์ใช้เพื่อพัฒนาระบบงานในการนำข้อมูลทั้งหมดไปวิเคราะห์เฉพาะอุปกรณ์หรือนำไปวิเคราะห์การทำงานของอุปกรณ์ทั้งหมดในภาพรวมตามความต้องการของผู้ใช้งาน

2. วัตถุประสงค์

พัฒนาคลังข้อมูล (Data Warehouse) เพื่อรองรับการจัดเก็บข้อมูลขนาดใหญ่ ที่เกิดจากการนำข้อมูลที่ถูกจัดเก็บใน Log File ของแต่ละอุปกรณ์ที่ทำงานด้านระบบเครือข่าย (Network System) มาเก็บรวมกัน หลังจากนั้นจะนำข้อมูลขนาดใหญ่ดังกล่าว มาสร้างความสัมพันธ์เพื่อให้สามารถทำงานร่วมกันได้แล้วนำข้อมูลมาเชื่อมโยงกันทั้งหมดโดยเชื่อมโยงผ่านข้อมูล Global Time Data จาก Time Server

พัฒนาโปรแกรมเพื่อประโยชน์ในการบริหารและจัดการ ICT Infrastructure ตามหลักการทำงานของ Business Intelligence ในการนำข้อมูลทั้งหมดไปวิเคราะห์การทำงานเฉพาะอุปกรณ์ หรือนำไปวิเคราะห์การทำงานร่วมกันของอุปกรณ์ทั้งหมด ทำให้สามารถวิเคราะห์ให้เห็นถึงพฤติกรรมของผู้ใช้งานในองค์กร และพฤติกรรมของการรุกรานหรือการใช้งานที่ไม่พึงประสงค์ได้ตามความต้องการของผู้ใช้งาน

3. ขอบเขตของปัญหา

ศึกษาและวิเคราะห์การทำงานของแต่ละอุปกรณ์ที่ทำงานด้านระบบเครือข่าย (Network System) บน ICT Infrastructure ขององค์กร โดยนำข้อมูลที่ถูกจัดเก็บใน Log File มาสร้างความสัมพันธ์เพื่อให้งานร่วมกันได้โดยเชื่อมโยงผ่านข้อมูล Global Time Data จาก Time Server และพัฒนาโปรแกรมเพื่อบริหารและจัดการ ICT Infrastructure ตามหลักการทำงานของ Business Intelligence เพื่อนำข้อมูลทั้งหมดไปวิเคราะห์การทำงานเฉพาะอุปกรณ์ หรือนำไปวิเคราะห์การทำงานร่วมกันของอุปกรณ์ทั้งหมด ทำให้สามารถวิเคราะห์ให้เห็นถึงพฤติกรรมของผู้ใช้งานในองค์กร และพฤติกรรมของการรุกรานหรือการใช้งานที่ไม่พึงประสงค์ได้ตามความต้องการของผู้ใช้งาน โดยพัฒนาโปรแกรมขึ้นเองและปรับปรุงจากโปรแกรมประเภท Business Intelligence ที่เป็น Open Source เพื่อสร้างรายงานรองรับความต้องการดังกล่าวออกมาในรูปแบบของตาราง และรูปแบบกราฟ และสามารถเรียกใช้งานผ่านเครือข่ายอินเทอร์เน็ต

4. ประโยชน์ที่คาดว่าจะได้รับ

1) ได้ระบบงานการพัฒนาคลังข้อมูลเพื่อรองรับการจัดการข้อมูลจาก Network Log File เพื่อทำหน้าที่จัดเก็บข้อมูล Log File ของอุปกรณ์ที่ทำงานด้านระบบเครือข่าย (Network System) บน ICT Infrastructure เพื่อมาวิเคราะห์เชิงบูรณาการ

2) ได้ระบบงานรองรับการนำข้อมูลปฐมภูมิ (Primary Data) จำนวนมากที่ถูกดึงมาจาก Network Log File ผ่านภาษา XML เข้าสู่คลังข้อมูลเพื่อนำมาวิเคราะห์และประมวลผลให้เป็นข้อมูลสารสนเทศ (Secondary Data หรือ Information)

3) ได้ระบบงานเพื่อนำไปสู่การวิเคราะห์ และสรุปผลของพฤติกรรมการใช้งานของผู้ใช้งาน ตามแนวทาง Business Intelligence และนำไปสู่การประยุกต์ในเชิงพาณิชย์

4) ได้ระบบงานต้นแบบสำหรับรองรับการบริหารจัดการข้อมูล Log File ที่ได้จากอุปกรณ์เครือข่ายที่ติดตั้งเพื่อรองรับการบริการด้าน ICT Infrastructure และนำไปวิเคราะห์เชิงบูรณาการสำหรับองค์กรต่างๆ

5. ขั้นตอนการดำเนินงาน

1. ศึกษาทฤษฎีที่เกี่ยวข้อง

เป็นขั้นตอนในการศึกษาทฤษฎีที่ใช้ในการออกแบบระบบงาน การออกแบบฐานข้อมูล การศึกษาซอฟท์แวร์ที่นำมาใช้ในการพัฒนาระบบงานได้แก่ Linux Operating System และ Apache Web Server และ MySQL DBMS และภาษา PHP(Professional Home Page) ภาษา XML ภาษา Java Script และศึกษาหลักการทำงานของอุปกรณ์ Network รวมถึงรูปแบบการจัดเก็บข้อมูลใน Log File และ Data Warehouse และ Business Intelligence Tools ที่เป็น Open Source เพื่อนำมาใช้ในการวิเคราะห์ข้อมูล

2. ศึกษาปัญหาระบบงานตามความเป็นจริง

เป็นการศึกษาการทำงานของระบบงานที่พัฒนา ศึกษาแบบฟอร์มของเอกสารต่างๆ ศึกษากระบวนการทำงานไม่ว่าจะเป็น Business Process Domain และ Process Flow เพื่อนำมาใช้เป็นข้อมูลประกอบสำหรับการออกแบบและการวิเคราะห์ระบบ เพื่อสามารถพัฒนาระบบงานจริงได้

3. ออกแบบขั้นตอนการทำงานของระบบงาน

เป็นขั้นตอนที่นำเอาทฤษฎีและวิธีการด้านการออกแบบขั้นตอนการทำงานข้างต้น มาวิเคราะห์เพื่อออกแบบระบบงานโดยจะแบ่งออกเป็นส่วนๆ เช่น ส่วนรับข้อมูล ส่วนจัดการข้อมูล ส่วนแสดงผลลัพธ์ ส่วนประมวลผล เป็นต้น เพื่อทำให้ระบบงานสามารถทำงานได้ครอบคลุม ถูกต้องและแม่นยำตามความต้องการของผู้ใช้งานจริง

4 .ออกแบบฐานข้อมูลหรือคลังข้อมูลของระบบงาน

เป็นขั้นตอนที่นำเอาทฤษฎีและวิธีการด้านการออกแบบฐานข้อมูลหรือคลังข้อมูล มาออกแบบเพื่อรองรับการจัดเก็บข้อมูลเพื่อวัตถุประสงค์ไม่ให้ซ้ำซ้อน บริหารได้ง่าย และสามารถเรียกใช้งานได้รวดเร็ว

5. พัฒนาโปรแกรมระบบงาน

เป็นขั้นตอนการเขียนโปรแกรมให้ครอบคลุมตามขั้นตอนของการทำงานที่ได้ออกแบบไว้

6. ทดสอบและติดตั้งระบบงาน

เป็นการทดสอบการใช้งานของโปรแกรมที่ได้พัฒนาขึ้น และบอกถึงความสามารถทั้งหมดที่เป็นไปได้ของโปรแกรมระบบงาน รวมถึงทราบถึงข้อจำกัดและเพื่อขจัดปัญหาที่เกิดขึ้นกับระบบงาน

7. จัดทำเอกสารและสรุปการทำงาน

เป็นขั้นตอนที่สร้างเอกสารประกอบการใช้งานโปรแกรมระบบงาน และเอกสารเพื่อการอ้างอิง

6. อุปกรณ์ที่ใช้ในการทำปัญหาพิเศษ

1. เครื่องคอมพิวเตอร์(Computer) เครื่องแม่ข่ายและลูกข่าย

2. ฮาร์ดดิสก์(Hard disk) และอุปกรณ์ต่อพ่วง

3. ซอฟต์แวร์ที่เกี่ยวข้องกับการพัฒนาระบบ ได้แก่ ระบบปฏิบัติการ Linux ระบบการจัดการฐานข้อมูล MySQL ระบบการบริหารการบริการด้านเว็บ Apache Web Server ตัวแปรภาษาต่างๆที่ใช้ในการพัฒนาระบบงาน

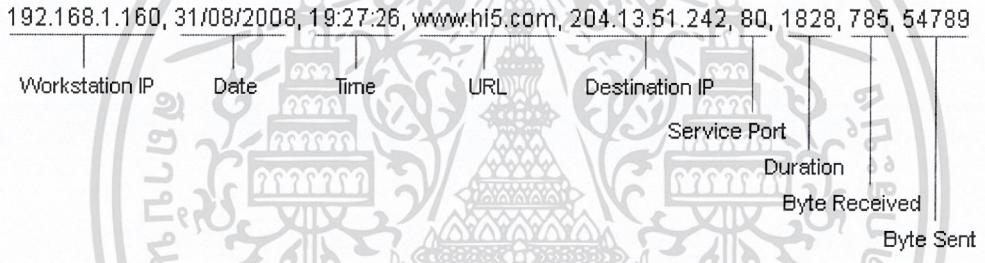
4. อุปกรณ์ Network ที่จัดเก็บข้อมูลในรูปแบบของ Log File

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 Log

ข้อมูลจราจรทางคอมพิวเตอร์ (Log) คือ ข้อมูลเกี่ยวกับการติดต่อสื่อสารของระบบคอมพิวเตอร์ซึ่งแสดงถึง แหล่งกำเนิด ต้นทาง ปลายทาง เส้นทาง วันที่ ปริมาณ ระยะเวลา ชนิดของบริการ หรือข้อมูลอื่นๆ ที่เกี่ยวกับการสื่อสารของระบบคอมพิวเตอร์

ตัวอย่าง:



2.2 Syslog-ng

ผู้ดูแลระบบ *nix ส่วนใหญ่คงคุ้นเคยกับ syslog มาเป็นอย่างดี เพราะ syslog ถือได้ว่าเป็น log daemon ที่ใช้กันมาอย่างยาวนานและกลายเป็นมาตรฐานของการเก็บข้อมูลล็อกของระบบปฏิบัติการ *nix ในหลายๆ ตัว แต่อย่างไรก็ตาม syslog ก็มีข้อเสียบางอย่าง ที่ log daemon ตัวอื่น เช่น syslog-ng, msyslog สามารถแก้ไขข้อบกพร่องดังกล่าวได้ เอกสารฉบับนี้จะแนะนำ syslog-ng ซึ่งเป็น log daemon ตัวใหม่ที่กำลังเป็นที่นิยมกันมากขึ้น และจะกล่าวถึงการสร้าง configuration แบบละเอียดเพื่อให้สามารถนำ syslog-ng ไปใช้งานได้จริง

Syslog-ng (Syslog new generation)

syslog-ng สามารถแก้ไขข้อบกพร่องส่วนใหญ่ของ syslog ได้ โดย

- syslog-ng สามารถทำงานได้ทั้งบน TCP และ UDP
- syslog-ng สามารถทำการกรอง (filter) ข้อมูลได้ด้วย regular expression
- syslog-ng สามารถทำงานในรูปแบบที่อ้างอิง priority/facility ได้ ดังนั้น มันจึงสามารถทำงานแทนที่ syslog ได้
- syslog-ng สนับสนุน log forwarding ซึ่งทำให้สามารถทราบได้ว่า ต้นทางของล็อกถูกส่งมาจากเครื่องใด และผ่านเครื่องใดมาบ้าง

นอกจากนี้ syslog-ng ยังมีรูปแบบของไฟล์ configuration ที่ง่าย แต่มีความยืดหยุ่นสูง สามารถนำไปประยุกต์ใช้ให้ตรงความต้องการได้โดยง่าย

2.3 ไฟร์วอลล์ (Firewall)

ปัจจุบันอินเตอร์เน็ตมีบทบาทสำคัญต่อการดำเนินกิจกรรมต่างๆ เป็นอย่างมาก ไม่ว่าจะเป็นด้านการติดต่อสื่อสาร ธุรกิจ การศึกษา หรือว่าเพื่อความบันเทิง องค์กรต่างๆ ทั้งภาครัฐและเอกชน ต่างก็นำเอาเน็ตเวิร์คของตนเชื่อมต่อเข้ากับอินเตอร์เน็ตเพื่อที่จะได้รับประโยชน์เหล่านี้ แต่เราต้องไม่ลืมว่าการนำเอาเน็ตเวิร์คไปเชื่อมต่อกับอินเตอร์เน็ตนั้น ทำให้ใครก็ได้บนอินเตอร์เน็ตสามารถเข้ามายังเน็ตเวิร์คนั้นๆ ได้ ปัญหาที่ตามมาก็คือความปลอดภัยของระบบเน็ตเวิร์ค เช่น ทำให้เกิดความเสี่ยงต่อการถูกเจาะระบบ และ ขโมยข้อมูล เป็นต้น

จากปัญหาดังกล่าวทำให้เราต้องมีวิธีการในการรักษาความปลอดภัย สิ่งที่สามารถช่วยลดความเสี่ยงนี้ได้ก็คือ ไฟร์วอลล์ โดยไฟร์วอลล์นั้นจะทำหน้าที่ป้องกันอันตรายต่างๆ จากภายนอกที่จะเข้ามายังเน็ตเวิร์กของเรา

รู้จักกับไฟร์วอลล์

ในความหมายทางด้านการก่อสร้างแล้ว ไฟร์วอลล์ จะหมายถึง กำแพงที่เอาไว้ป้องกันไฟไม่ให้ลุกลามไปยังส่วนอื่นๆ ส่วนทางด้านคอมพิวเตอร์นั้นก็จะมีความหมายคล้ายๆ กันก็คือ เป็นระบบที่เอาไว้ป้องกันอันตรายจากอินเตอร์เน็ตหรือเน็ตเวิร์กภายนอกนั่นเอง

ไฟร์วอลล์เป็นคอมโพเน็นต์หรือกลุ่มของคอมโพเน็นต์ที่ทำหน้าที่ในการควบคุมการเข้าถึงระหว่างเน็ตเวิร์กภายนอกหรือเน็ตเวิร์กที่เราคิดว่าไม่ปลอดภัย กับเน็ตเวิร์กภายในหรือเน็ตเวิร์กที่เราต้องการจะป้องกัน โดยที่คอมโพเน็นต์นั้นอาจจะเป็นเราเตอร์ คอมพิวเตอร์ หรือเน็ตเวิร์ก ประกอบกันก็ได้ ขึ้นอยู่กับวิธีการหรือ Firewall Architecture ที่ใช้

รูปที่ 2.1 ไฟร์วอลล์กั้นระหว่างอินเตอร์เน็ตกับเน็ตเวิร์กภายใน

การควบคุมการเข้าถึงของไฟร์วอลล์นั้น สามารถทำได้ในหลายระดับและหลายรูปแบบขึ้นอยู่ชนิดหรือเทคโนโลยีของไฟร์วอลล์ที่นำมาใช้ เช่น เราสามารถกำหนดได้ว่าจะให้มีการเข้ามาใช้เซอร์วิสอะไรได้บ้าง จากที่ไหน เป็นต้น

สิ่งที่ไฟร์วอลล์ช่วยได้

ไฟร์วอลล์สามารถช่วยเพิ่มความปลอดภัยให้กับระบบได้โดย

- บังคับใช้นโยบายด้านความปลอดภัย โดยการกำหนดกฎให้กับไฟร์วอลล์ว่าจะอนุญาตหรือไม่ให้ใช้เซอร์วิสชนิดใด
- ทำให้การพิจารณาดูแลและการตัดสินใจด้านความปลอดภัยของระบบเป็นไปได้ง่ายขึ้น เนื่องจากการติดต่อทุกชนิดกับเน็ตเวิร์กภายนอกจะต้องผ่านไฟร์วอลล์ การดูแลที่จุดนี้เป็นการดูแลความปลอดภัยในระดับของเน็ตเวิร์ก (Network-based Security)
- บันทึกข้อมูล กิจกรรมต่างๆ ที่ผ่านเข้าออกเน็ตเวิร์กได้อย่างมีประสิทธิภาพ
- ป้องกันเน็ตเวิร์กบางส่วนจากการเข้าถึงของเน็ตเวิร์กภายนอก เช่นถ้าหากเรามีบางส่วนที่ต้องการให้ภายนอกเข้ามาใช้เซอร์วิส (เช่นถ้ามีเว็บเซิร์ฟเวอร์) แต่ส่วนที่เหลือไม่ต้องการให้ภายนอกเข้ามากรณีเช่นนี้เราสามารถใช้ไฟร์วอลล์ช่วยได้
- ไฟร์วอลล์บางชนิด [1] สามารถป้องกันไวรัสได้ โดยจะทำการตรวจไฟล์ที่โอนย้ายผ่านทางโปรโตคอล HTTP, FTP และ SMTP

อะไรที่ไฟร์วอลล์ช่วยไม่ได้

ถึงแม้ว่าไฟร์วอลล์จะสามารถช่วยเพิ่มความปลอดภัยให้กับเน็ตเวิร์กได้มากโดยการตรวจดูข้อมูลที่ผ่านเข้าออก แต่อย่าลืมว่าสิ่งเหล่านี้ไม่สามารถป้องกันได้จากการใช้ไฟร์วอลล์

- อันตรายที่เกิดจากเน็ตเวิร์กภายใน ไม่สามารถป้องกันได้เนื่องจากอยู่ภายในเน็ตเวิร์กเองไม่ได้ผ่านไฟร์วอลล์เข้ามา
- อันตรายจากภายนอกที่ไม่ได้ผ่านเข้ามาทางไฟร์วอลล์ เช่นการ Dial-up เข้ามายังเน็ตเวิร์กภายในโดยตรงโดยไม่ได้ผ่านไฟร์วอลล์
- อันตรายจากวิธีใหม่ๆ ที่เกิดขึ้น ทุกวันนี้มีการพบช่องโหว่ใหม่ๆ เกิดขึ้นทุกวัน เราไม่สามารถไว้ใจไฟร์วอลล์โดยการติดตั้งเพียงครั้งเดียวแล้วก็หวังให้มันปลอดภัยตลอดไป เราต้องมีการดูแลรักษาอย่างต่อเนื่องสม่ำเสมอ
- ไวรัส ถึงแม้จะมีไฟร์วอลล์บางชนิดที่สามารถป้องกันไวรัสได้ แต่ก็ยังไม่มีไฟร์วอลล์ชนิดใดที่สามารถตรวจสอบไวรัสได้ในทุกๆ โปรโตคอล

ชนิดของไฟร์วอลล์

ชนิดของไฟร์วอลล์แบ่งตามเทคโนโลยีที่ใช้ในการตรวจสอบและควบคุม แบ่งได้เป็น

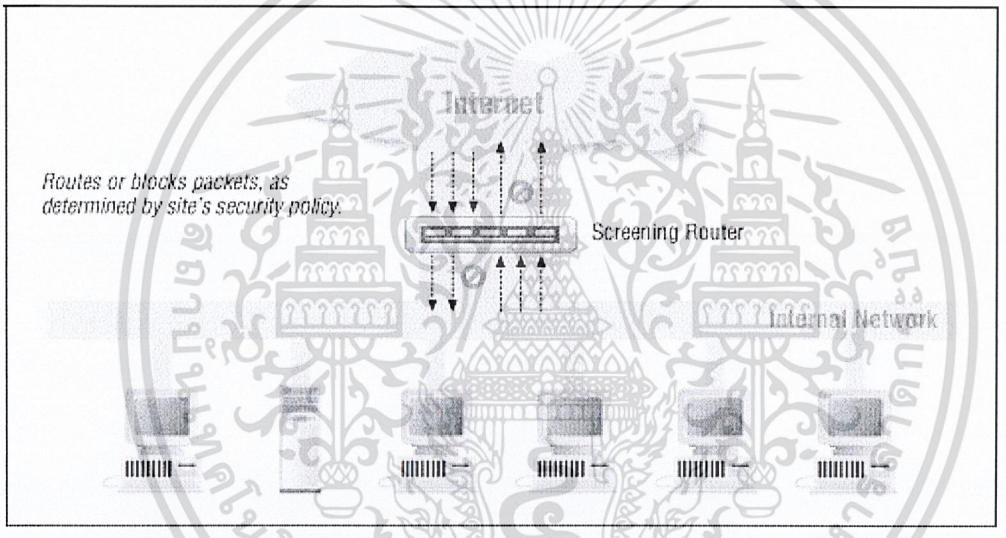
- Packet Filtering
- Proxy Service

- Stateful Inspection

Packet Filtering

Packet Filter คือเราเตอร์ที่ทำการหาเส้นทางและส่งต่อ (route) อย่างมีเงื่อนไข โดยจะพิจารณาจากข้อมูลส่วนที่อยู่ในเฮดเดอร์ (header) ของแพ็กเก็ตที่ผ่านเข้ามา เทียบกับกฎ (rules) ที่กำหนดไว้และตัดสินว่าควรจะทิ้ง (drop) แพ็กเก็ตนั้นไปหรือว่าจะยอม (accept) ให้แพ็กเก็ตนั้นผ่านไปได้

รูปที่ 2.2 ใช้ Screening Router ทำหน้าที่ Packet Filtering



ในการพิจารณาเฮดเดอร์ Packet Filter จะตรวจสอบในระดับของอินเตอร์เน็ตเลเยอร์ (Internet Layer) และทรานสปอร์ตเลเยอร์ (Transport Layer) ในอินเตอร์เน็ตโมเดล ซึ่งในอินเตอร์เน็ตเลเยอร์จะมีแอตทริบิวต์ที่สำคัญต่อ Packet Filtering ดังนี้

- ไอพีต้นทาง
- ไอพีปลายทาง
- ชนิดของโปรโตคอล (TCP UDP และ ICMP)

และในระดับของทรานสปอร์ตเลเยอร์ มีแอตทริบิวต์ที่สำคัญคือ

- พอร์ตต้นทาง
- พอร์ตปลายทาง
- แฟล็ก (Flag ซึ่งจะมีเฉพาะในเฮดเดอร์ของแพ็กเก็ต TCP)

- ชนิดของ ICMP message (ในแพ็กเก็ต ICMP)

ซึ่งพอร์ตของทรานสปอร์ตเลเยอร์ คือทั้ง TCP และ UDP นั้นจะเป็นสิ่งที่บอกถึงแอพพลิเคชันที่แพ็กเก็ตนั้นต้องการติดต่อด้วยเช่น พอร์ต 80 หมายถึง HTTP, พอร์ต 21 หมายถึง FTP เป็นต้น ดังนั้นเมื่อ Packet Filter พิจารณาเฮดเดอร์ จึงทำให้สามารถควบคุมแพ็กเก็ตที่มาจากที่ต่างๆ และมีลักษณะต่างๆ (ดูได้จากแฟล็กของแพ็กเก็ต หรือ ชนิดของ ICMP ในแพ็กเก็ต ICMP) ได้ เช่น ห้ามแพ็กเก็ตทุกชนิดจาก crack.cracker.net เข้ามายังเน็ตเวิร์ก 203.154.207.0/24 , ห้ามแพ็กเก็ตที่มีไอพีต้นทางอยู่ในเน็ตเวิร์ก 203.154.207.0/24 ผ่านเราเตอร์เข้ามา (ในกรณีนี้เพื่อเป็นการป้องกัน ip spoofing) เป็นต้น

Packet Filtering สามารถอิมพลีเมนต์ได้จาก 2 แพล็ตฟอร์ม คือ

- เราเตอร์ที่มีความสามารถในการทำ Packet Filtering (ซึ่งมีในเราเตอร์ส่วนใหญ่อยู่แล้ว)
- คอมพิวเตอร์ที่ทำหน้าที่เป็นเราเตอร์

ซึ่งจะมีข้อได้เปรียบเสียเปรียบกันดังนี้

	ข้อดี	**ข้อเสีย**
เราเตอร์	ประสิทธิภาพสูง มีจำนวนอินเตอร์เฟสมาก	เพิ่มเติมฟังก์ชันการทำงานได้ยาก, อาจต้องการหน่วยความจำมาก
คอมพิวเตอร์ที่ทำหน้าที่เป็นเราเตอร์	เพิ่มฟังก์ชันการทำงานได้ไม่จำกัด	ประสิทธิภาพปานกลาง,จำนวนอินเตอร์เฟสน้อย,อาจมีความเสี่ยงจากระบบปฏิบัติการที่ใช้

ตารางที่ 2.1 เปรียบเทียบข้อดีข้อเสียในการเลือกอุปกรณ์มาทำหน้าที่ Packet Filtering

ข้อดี-ข้อเสียของ Packet Filtering

ข้อดี

- ไม่ขึ้นกับแอพพลิเคชัน
- มีความเร็วสูง

- รองรับการขยายตัวได้ดี

ข้อเสีย

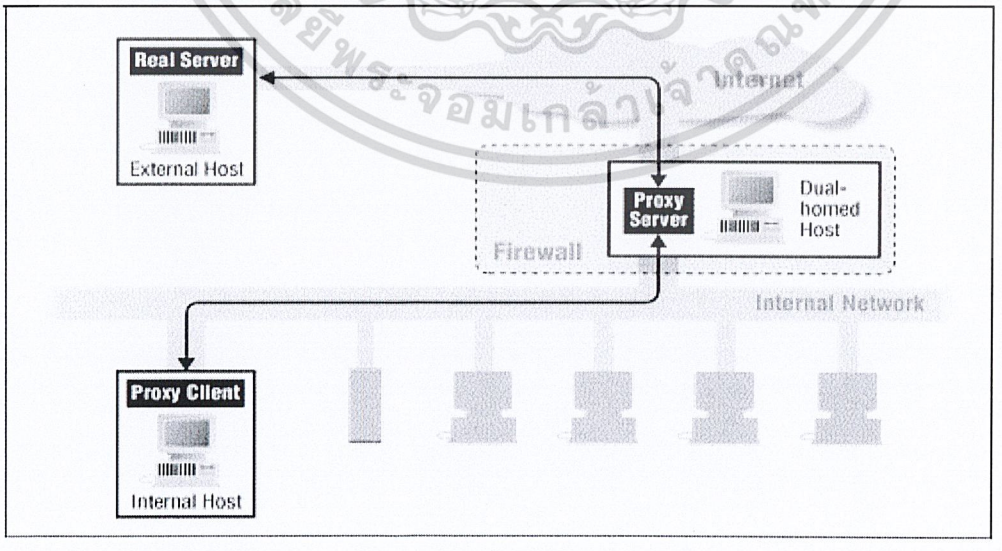
- บางโปรโตคอลไม่เหมาะสมกับการใช้ Packet Filtering เช่น FTP, ICQ

Proxy

Proxy หรือ Application Gateway เป็นแอพพลิเคชันโปรแกรมที่ทำงานอยู่บนไฟร์วอลล์ที่ตั้งอยู่ระหว่างเน็ตเวิร์ก 2 เน็ตเวิร์ก ทำหน้าที่เพิ่มความปลอดภัยของระบบเน็ตเวิร์กโดยการควบคุมการเชื่อมต่อระหว่างเน็ตเวิร์กภายในและภายนอก Proxy จะช่วยเพิ่มความปลอดภัยได้มากเนื่องจากมีการตรวจสอบข้อมูลถึงในระดับของแอพพลิเคชันเลเยอร์ (Application Layer)

เมื่อไคลเอนต์ต้องการใช้เซอร์วิสภายนอก ไคลเอนต์จะทำการติดต่อไปยัง Proxy ก่อน ไคลเอนต์จะเจรจา (negotiate) กับ Proxy เพื่อให้ Proxy ติดต่อไปยังเครื่องปลายทางให้ เมื่อ Proxy ติดต่อไปยังเครื่องปลายทางให้แล้วจะมีการเชื่อมต่อ (connection) 2 การเชื่อมต่อ คือ ไคลเอนต์กับ Proxy และ Proxy กับเครื่องปลายทาง โดยที่ Proxy จะทำหน้าที่รับข้อมูลและส่งต่อข้อมูลให้ใน 2 ทิศทาง ทั้งนี้ Proxy จะทำหน้าที่ในการตัดสินใจว่าจะให้มีการเชื่อมต่อกันหรือไม่ จะส่งต่อแพ็กเก็ตให้หรือไม่

รูปที่ 2.3 ใช้ Dual-homed Host เป็น Proxy Server



ข้อดี-ข้อเสียของ Proxy

ข้อดี

- มีความปลอดภัยสูง
- รู้จักข้อมูลในระดับแอพพลิเคชัน

ข้อเสีย

- ประสิทธิภาพต่ำ
- แต่ละบริการมักจะต้องการโปรเซสของตนเอง
- สามารถขยายตัวได้ยาก

Stateful Inspection Technology

โดยปกติแล้ว Packet Filtering แบบธรรมดา (ที่เป็น Stateless แบบที่มีอยู่ในเราเตอร์ทั่วไป) จะควบคุมการเข้าออกของแพ็กเก็ตโดยพิจารณาข้อมูลจากเฮดเดอร์ของแต่ละแพ็กเก็ต นำมาเทียบกับกฎที่มีอยู่ ซึ่งกฎที่มีอยู่ก็จะเป็นกฎที่สร้างจากข้อมูลส่วนที่อยู่ในเฮดเดอร์เท่านั้น ดังนั้น Packet Filtering แบบธรรมดาจึงไม่สามารถทราบได้ว่า แพ็กเก็ตนี้อยู่ส่วนใดของการเชื่อมต่อ เป็นแพ็กเก็ตที่เข้ามาติดต่อใหม่หรือเปล่า หรือว่าเป็นแพ็กเก็ตที่เป็นส่วนของการเชื่อมต่อที่เกิดขึ้นแล้ว เป็นต้น

Stateful Inspection เป็นเทคโนโลยีที่เพิ่มเข้าไปใน Packet Filtering โดยในการพิจารณาว่าจะยอมให้แพ็กเก็ตผ่านไปนั้น แทนที่จะดูข้อมูลจากเฮดเดอร์เพียงอย่างเดียว Stateful Inspection จะนำเอาส่วนข้อมูลของแพ็กเก็ต (message content) และข้อมูลที่ได้จากแพ็กเก็ตก่อนหน้านี้ที่ได้ทำการบันทึกเอาไว้ นำมาพิจารณาด้วย จึงทำให้สามารถระบุได้ว่าแพ็กเก็ตใดเป็นแพ็กเก็ตที่ติดต่อเข้ามาใหม่ หรือว่าเป็นส่วนหนึ่งของการเชื่อมต่อที่มีอยู่แล้ว

Firewall Architecture

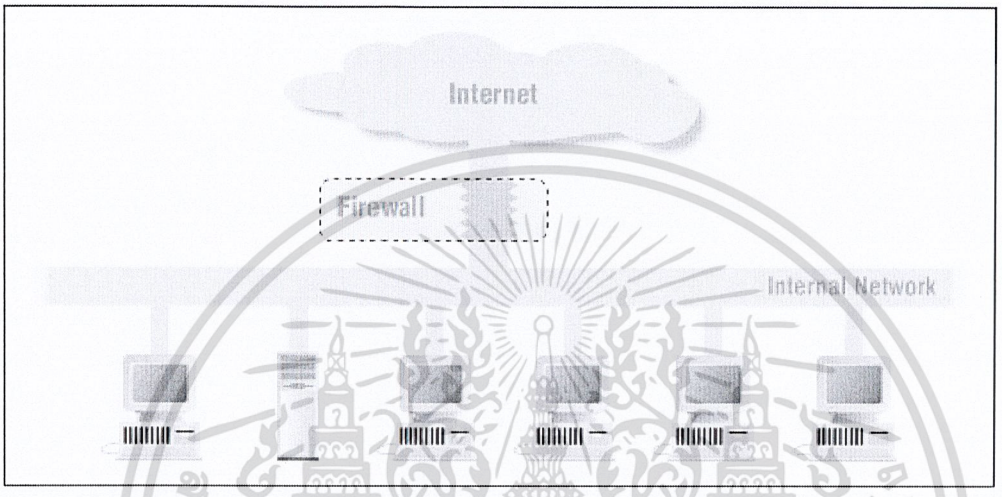
ในส่วนของ Firewall Architecture นั้น จะพูดถึงการจัดวางไฟร์วอลล์คอมโพเน็นต์ในแบบต่างๆ เพื่อทำให้เกิดเป็นระบบไฟร์วอลล์ขึ้น

Single Box Architecture

Single Box Architecture เป็น Architecture แบบง่ายๆ ที่มีคอมโพเน็นต์ทำหน้าที่เป็นไฟร์วอลล์เพียงอันเดียวตั้งอยู่ระหว่างเน็ตเวิร์กภายในกับเน็ตเวิร์กภายนอก ข้อดีของวิธีนี้ก็คือการที่มีเพียงจุดเดียวที่หน้าที่ไฟร์วอลล์ทั้งหมด ควบคุมการเข้าออกของข้อมูล ทำให้ดูแลได้ง่าย เป็นจุดสนใจในการดูแล

ความปลอดภัยเน็ตเวิร์ก ในทางกลับกันข้อเสียของวิธีนี้ก็คือ การที่มีเพียงจุดเดียวนี้ ทำให้มีความเสี่ยงสูง หากมีการคอนฟิกูเรชันผิดพลาดหรือมีช่องโหว่เพียงเล็กน้อย การผิดพลาดเพียงจุดเดียวอาจทำให้ระบบถูกเจาะได้

รูปที่ 2.4 Firewall Architecture แบบชั้นเดียว



คอมโพเน็นต์ที่ใช้ใน Architecture นี้อาจเป็น Screening Router , Dual-Homed Host หรือ Multi-purposed Firewall Box ก็ได้

1) Screening Router

เราสามารถใช้เราเตอร์ทำ Packet Filtering ได้ วิธีนี้จะทำให้ประหยัดค่าใช้จ่ายเนื่องจากส่วนใหญ่จะใช้เราเตอร์ต่อกับเน็ตเวิร์กภายนอกอยู่แล้ว แต่วิธีนี้อาจไม่ยืดหยุ่นมากนักในการคอนฟิกกูเรชัน

Architecture แบบนี้เหมาะสำหรับ

- เน็ตเวิร์กที่มีการป้องกันความปลอดภัยในระดับของโฮสต์ (Host security) เป็นอย่างดีแล้ว
- มีการใช้โปรโตคอลไม่มาก และโปรโตคอลที่ใช้ก็เป็นโปรโตคอลที่ไม่ซับซ้อน
- ต้องการไฟร์วอลล์ที่มีความเร็วสูง

2) Dual-Homed Host

เราสามารถใช้ Dual-Homed Host (คอมพิวเตอร์ที่มีเน็ตเวิร์กอินเตอร์เฟสอย่างน้อย 2 อัน) ใช้การบริการเป็น Proxy ให้กับเครื่องภายในเน็ตเวิร์ก

Architecture แบบนี้เหมาะสำหรับ

- เน็ตเวิร์กที่มีการใช้งานอินเตอร์เน็ตค่อนค่างน้อย
- เน็ตเวิร์กที่ไม่ได้มีข้อมูลสำคัญๆ

3) Multi-purposed Firewall Box

มีผลิตภัณฑ์หลายชนิดที่ผลิตออกมาเป็นกล่องๆ เดียว ซึ่งทำหน้าที่ได้หลายอย่าง ทั้ง Packet Filtering, Proxy แต่ก็อย่าลืมว่านี่คือ Architecture แบบชั้นเดียว ซึ่งถ้าพลาดแล้วก็จะเสียหายทั้งเน็ตเวิร์กได้

Screened Host Architecture

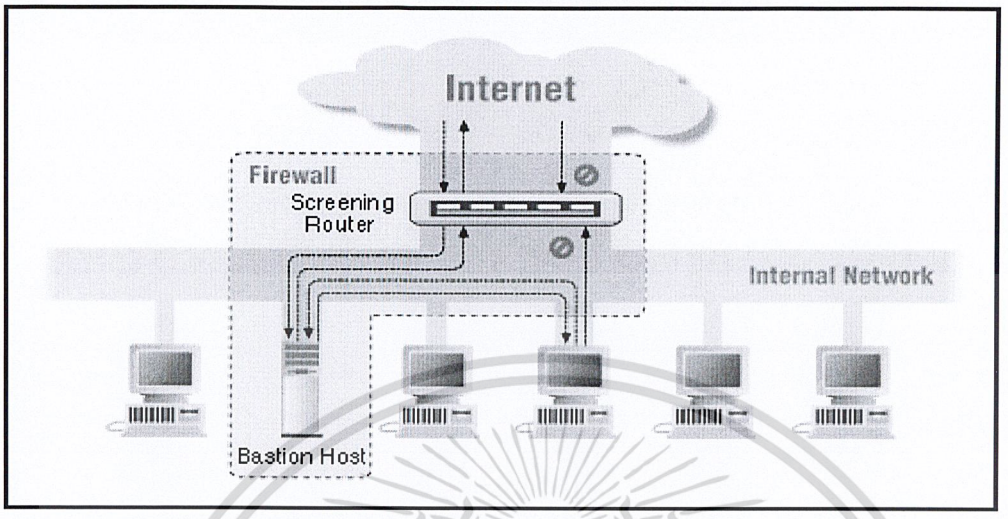
Screened Host Architecture จะมีโฮสต์ซึ่งให้บริการ Proxy เหมือนกับใน Single Box Architecture ที่เป็น Dual-homed Host แต่จะต่างกันตรงที่ว่า โฮสต์นั้นจะอยู่ภายในเน็ตเวิร์ก ไม่ต่ออยู่กับเน็ตเวิร์กภายนอกอื่นๆ (ดังนั้นก็ไม่จำเป็นที่จะต้องใช้ Dual Homed Host) และจะมี เราเตอร์ที่ทำหน้าที่ Packet Filtering ช่วยบังคับให้เครื่องภายในเน็ตเวิร์กต้องติดต่อเซอร์วิสผ่าน Proxy โดยไม่ยอมให้ติดต่อใช้เซอร์วิสจากภายนอกโดยตรง และก็ให้ภายนอกเข้าถึงได้เฉพาะ Bastion host (คือ โฮสต์ที่มีความเสี่ยงสูงต่อการถูกโจมตี มักจะเป็นโฮสต์ที่เปิดให้บริการกับอินเตอร์เน็ต ดังนั้นโฮสต์นี้ต้องมีการดูแลเป็นพิเศษ) เท่านั้น

จากรูปที่ 5 ใน Architecture แบบนี้จะประกอบไปด้วยเราเตอร์ทำหน้าที่ Packet Filtering และภายในเน็ตเวิร์กจะมี Bastion Host ให้บริการ Proxy อยู่ โดยที่เราเตอร์นั้นอาจจะถูกเซ็ตดังนี้

- อาจจะอนุญาตให้เครื่องภายในใช้เซอร์วิสบางอย่างได้โดยตรง
- ส่วนเซอร์วิสอื่นๆ จะไม่ยอมให้เครื่องภายในติดต่อผ่านออกไปโดยตรง ยกเว้น Bastion Host เท่านั้นที่สามารถติดต่อกับเน็ตเวิร์กภายนอกได้ทั้งนี้เพื่อเป็นการบังคับให้ใช้บริการ Proxy ผ่านทาง Bastion Host เท่านั้น

หรืออาจจะเซ็ตให้เซอร์วิสส่วนใหญ่ผ่านเราเตอร์ออกไปได้โดยตรงแล้ว ให้บางส่วนต้องใช้เซอร์วิสผ่าน Proxy ก็แล้วแต่นโยบายและความเหมาะสมขององค์กร

รูปที่ 2.5 Screened Host Architecture



วิธีนี้ถึงแม้ว่าจะมีทั้ง Proxy และเราเตอร์ทำหน้าที่ Packet Filtering แต่ก็ยังคงอันตรายอยู่ เพราะว่าเราเตอร์ต้องยอมให้ภายนอกสามารถติดต่อกับ Bastion Host ได้อยู่แล้ว หากแฮกเกอร์สามารถเจาะเข้ามายัง Bastion Host ได้ก็เสร็จ

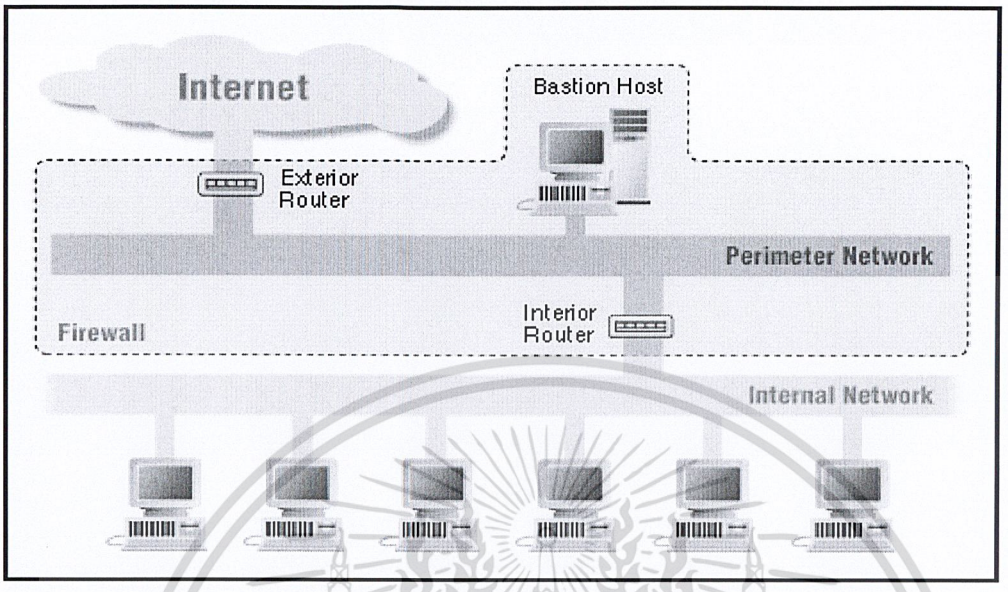
Architecture นี้เหมาะสำหรับ

- เน็ตเวิร์กที่มีการติดต่อกับเน็ตเวิร์กภายนอกน้อย
- เน็ตเวิร์กที่มีการป้องกันความปลอดภัยในระดับของโฮสต์เป็นอย่างดีแล้ว

Multi Layer Architecture

ในสถาปัตยกรรมแบบหลายชั้น ไฟร์วอลล์จะเกิดขึ้นจากคอมโพเน็นต์หลายๆส่วนทำหน้าที่ประกอบกันขึ้นเป็นระบบ วิธีการนี้สามารถเพิ่มความปลอดภัยได้มาก เนื่องจากเป็นการลดความเสี่ยงต่อความผิดพลาดที่อาจเกิดขึ้น ถ้าหากมีไฟร์วอลล์เพียงจุดเดียวแล้วมีเกิดความผิดพลาดเกิดขึ้นระบบทั้งหมดก็จะเป็นอันตราย แต่ถ้ามีการป้องกันหลายชั้น หากในชั้นแรกถูกเจาะ ก็อาจจะมีความเสียหายเพียงบางส่วน ส่วนที่เหลือระบบก็ยังคงมีชั้นอื่นๆ ในการป้องกันอันตราย และยังลดความเสี่ยงได้โดยการที่แต่ละชั้นนั้นมีการใช้เทคโนโลยีที่แตกต่างกัน เพื่อให้เกิดความหลากหลาย เป็นการหลีกเลี่ยงการโจมตีหรือช่องโหว่ที่อาจมีในเทคโนโลยีชนิดใดชนิดหนึ่ง โดยทั่วไปแล้วสถาปัตยกรรมแบบหลายชั้นจะเป็นการต่อกันเป็นซีรี่โดยมี Perimeter Network (หรือบางทีเรียกว่า DMZ Network) อยู่ตรงกลาง เรียกว่า Screened Subnet Architecture

รูปที่ 2.6 Screened Subnet Architecture



Screened Subnet Architecture

Screened Subnet Architecture เป็นสถาปัตยกรรมที่มีการเพิ่ม Perimeter Network เข้าไปกั้นระหว่างอินเตอร์เน็ตกับเน็ตเวิร์กภายในไม่ให้เชื่อมต่อกันโดยตรง ทำให้เน็ตเวิร์กภายในมีความปลอดภัยมากขึ้น

ในรูปที่ 6 แสดง Screened Subnet Architecture อย่างง่าย ประกอบไปด้วย เราเตอร์ 2 ตัว ตัวนึงอยู่ระหว่างอินเตอร์เน็ตกับ Perimeter Network ส่วนอีกตัวหนึ่งอยู่ระหว่าง Perimeter Network กับเน็ตเวิร์กภายใน ถ้าหากแฮกเกอร์จะเจาะเน็ตเวิร์กภายในต้องผ่านเราเตอร์เข้ามาถึง 2 ตัวด้วยกัน ถึงแม้ว่าจะเจาะชั้นแรกเข้ามายัง Bastion host ได้ แต่ก็ยังต้องผ่านเราเตอร์ตัวในอีก ถึงจะเข้ามายังเน็ตเวิร์กภายในได้

คอมโพเน็นต์ของ Screened Subnet Architecture ในรูปที่ 6

- Perimeter Network เป็นเน็ตเวิร์กที่เพิ่มเข้ามาเพื่อความปลอดภัย อยู่ระหว่างเน็ตเวิร์กภายนอกกับเน็ตเวิร์กภายใน ประโยชน์ของ Perimeter Network ที่เห็นได้ชัดก็คือ การแบ่งเน็ตเวิร์กออกเป็นส่วนๆ ทำให้การไหลของข้อมูลถูกแบ่งออกเป็นส่วนๆตามเน็ตเวิร์กด้วย เนื่องจากโดยทั่วไปแล้ว เน็ตเวิร์กที่เป็นแลนนั้น จะเป็นแบบ Ethernet ซึ่งจะมีการส่งข้อมูลแบบ Broadcast ดังนั้นถ้ามีใครคอบดักจับข้อมูลอยู่ในเน็ตเวิร์กนั้น ก็จะได้พาสเวิร์ดข้อมูลต่างๆ ไปหมด ดังนั้นหากไฟร์วอลล์เรามีชั้นเดียวและแฮกเกอร์สามารถเข้ามาได้ โดน

ดักจับข้อมูลก็เสร็จหมด แต่ถ้าเรามี Perimeter Network ถึงจะดักจับข้อมูลได้แต่ก็จะได้เพียงที่อยู่บน Perimeter Network เท่านั้น

- Bastion Host ตั้งอยู่บน Perimeter Network ทำหน้าที่ให้บริการ Proxy กับเน็ตเวิร์กภายในและให้บริการต่างๆ กับผู้ใช้บนอินเตอร์เน็ต Bastion Host นั้นจะมีความเสี่ยงต่อการโจมตีสูง จึงต้องมีการดูแลความปลอดภัยเป็นพิเศษ
- Interior Router ตั้งอยู่ระหว่าง Perimeter Network กับเน็ตเวิร์กภายใน ทำหน้าที่ Packet Filtering ป้องกันเน็ตเวิร์กภายในจาก Perimeter Network ในการเซ็ต configuration ระหว่าง เน็ตเวิร์กภายในกับ Perimeter Network ควรกำหนดอย่างรอบคอบ อนุญาตเฉพาะเซอร์วิสที่จำเป็นเท่านั้นอย่างเช่น DNS, SMTP
- Exterior Router ตั้งอยู่ระหว่างเน็ตเวิร์กภายนอกกับ Perimeter Network เนื่องจาก Exterior Router นี้เป็นจุดที่ต่ออยู่กับเน็ตเวิร์กภายนอก จึงมีหน้าที่ที่สำคัญอย่างหนึ่งคือ การป้องกันแพ็กเก็ตที่มีการ Forged IP Address เข้ามา โดยอ้างว่ามาจากเน็ตเวิร์กภายในทั้งๆ ที่จริงๆ แล้วมาจากเน็ตเวิร์กภายนอก

2.4 XML

XML ย่อมาจาก Extensible Markup Language เป็นภาษาแบบ markup คล้ายๆ กับ HTML แต่มีจุดประสงค์ในการใช้งานที่แตกต่างกัน โดย XML นั้นถูกออกแบบมาเพื่อใช้เป็นรูปแบบในการเก็บข้อมูล หรือรูปแบบของข้อมูลที่จะต้องมีการรับส่งกันผ่านเครือข่าย โดยโปรแกรมเมอร์สามารถกำหนดชื่อ tag ต่างๆ ได้เอง ส่วน HTML นั้นเป็นภาษาที่ใช้ในการแสดงผลข้อมูล การกำหนดว่าข้อมูลต่างๆ จะแสดงอยู่ในส่วนใดของเว็บเพจ ในบางครั้งเราอาจจะนำ XML และ HTML มาใช้ร่วมกัน เช่น การนำข้อมูลที่บันทึกไว้ในรูปแบบ XML มาแสดงเป็นส่วนหนึ่งในเว็บเพจที่เขียนด้วย HTML ปัจจุบันได้มีการใช้งาน XML กันอย่างแพร่หลาย เช่น การใช้ XML เป็นรูปแบบของข้อมูลในระบบ Web Services, การเขียน configuration file ของโปรแกรมต่างๆ ในรูปของ XML, การใช้งาน XML ใน AJAX (Asynchronous JavaScript and XML) รวมไปถึงการใช้ XML เป็นรูปแบบกลางในการแปลงข้อมูลระหว่างฐานข้อมูลต่างชนิดกัน เป็นต้น

ประโยชน์ของ XML

สำหรับประโยชน์ของ XML นั้น เป็นด้านความยืดหยุ่นในการใช้งานสำหรับแอพพลิเคชันที่อิงกับ Web Base ที่ใช้ง่ายในการค้นหาข้อมูล มีความยืดหยุ่นในการพัฒนาเว็บ สามารถผสมผสานข้อมูลจากหลายแหล่ง จากแอพพลิเคชันที่ต่างกัน สามารถแสดงข้อมูลแบบต่างๆ และสามารถ

Update ข้อมูลให้ทันสมัยเสมอ และคาดว่าจะเป็นมาตรฐานใหม่ของระบบเปิด ซึ่งนับเป็น format ใหม่สำหรับการส่งข้อมูลบนเว็บที่มากด้วยข้อมูลหลายแบบ แต่ส่งผ่านด้วยเทคโนโลยีที่บีบอัดข้อมูลที่ให้ความเร็วได้รับการสนับสนุนจากผลิตภัณฑ์ค่ายไมโครซอฟท์ แต่อย่างไรก็ตามในปัจจุบันนี้ XML นั้นสามารถใช้ได้กับทุกค่ายทุกผลิตภัณฑ์

ประโยชน์หลักๆนั้นสามารถยกตัวอย่างได้ดังนี้

- Self-describe data: ใช้สำหรับสร้างข้อมูลที่สามารถอธิบายเนื้อหาของมันเองได้

- Data exchange: ใช้สำหรับการแลกเปลี่ยนข้อมูล

- Messaging format: กำหนดรูปแบบข้อความในการสื่อสาร ระหว่างแอปพลิเคชั่นหรือโปรแกรม เช่นโพรโตคอล SOAP เป็นต้น

- ใช้สำหรับการเข้าถึงระบบข้อมูลขนาดใหญ่ อาทิเช่น ใช้กับระบบเครือข่ายในองค์กร หรืออินเตอร์เน็ตเพื่อดูข้อมูลหรือเรียกใช้ข้อมูลที่ให้การแสดงผลทางหน้าจอที่รวดเร็วง่ายในการจัดการ

- XML จะเกิดความสะดวกในระบบพาณิชย์อิเล็กทรอนิกส์ ในระบบการค้าอิเล็กทรอนิกส์ในอนาคตข้างหน้านี้ ข้อมูลจะถูกเก็บอยู่ในรูปแบบของเอกสาร XML ทั้งหมด XML ทำให้ผู้ค้า ผู้ใช้และผู้พัฒนาเทคโนโลยี มีความอิสระในการเพิ่มศักยภาพในผลิตภัณฑ์ ของตัวเองไม่จำเป็นต้องกังวลรูปแบบกาสื่อสาร ที่จะต้องออกแบบมาเฉพาะ ทำให้ลูกค้าต้องยึดติดกับผลิตภัณฑ์รายใดรายหนึ่งเมื่อข้อมูลอยู่ในเอกสาร XML แล้วผลิตภัณฑ์ต่างๆก็จะมีรูปแบบข้อมูลที่สามารถแลกเปลี่ยนกันได้โดยที่ไม่ต้องกังวลกับเทคโนโลยี

ที่ไม่สอดคล้อง กันในปัจจุบันได้

- XML ลดค่าใช้จ่าย แน่นอน ผลที่ตามมากับ เทคโนโลยีที่แลกเปลี่ยนข้อมูลกันได้ ทำให้ค่าใช้จ่ายที่จะต้องสูญเสียไป ในปัจจุบันสำหรับการสื่อสารข้อมูล ที่มีรูปแบบที่หลากหลายถูก ขจัดออกไป นั้นเป็นผลดีสำหรับยุคการสื่อสารด้วย XML

- XML สนับสนุนการทำงานกับ UNICODE และผสมได้หลากหลายภาษา

การพัฒนา XML Processor ทำให้สามารถดึงเอกสาร XML มาใช้งานได้ง่าย และใช้ร่วมกับโปรแกรมประยุกต์อื่นได้ง่าย เช่น โปรแกรม DB2, Oracle, SAP เป็นต้น

- XML ช่วยทำให้เกิดการรับส่งข้อมูลแบบ Electronic Data Interchange โดยทำให้แนวทางการเชื่อมโยงและสร้างความเป็นเอกสาร หรือมาตรฐานระหว่างองค์กร

- XML มีสภาพช่วยในการขนส่งข้อมูลไปยังปลายทางเพื่อให้แปลความหมายและใช้งานได้อย่าง เต็มประสิทธิภาพมีการสร้างการประยุกต์ และนำเสนอผลลัพธ์ไปใช้งานจาก XML ได้มาก

การประยุกต์การดำเนินกิจกรรมบนเครือข่ายมีมาก เช่น eBusiness, EDI, eCommerce, การจัดการ Supply chain, Demand chain management การดำเนินการแบบ intranet และ web base application

2.5 ฐานข้อมูลคลังข้อมูล (Data Warehouse)

สามารถให้คำจำกัดความของฐานข้อมูลคลังข้อมูล (Data Warehouse) ด้วยนิยาม 4 ข้อคือ ฐานข้อมูลคลังข้อมูล (Data Warehouse) เป็นฐานข้อมูลที่มีการรวบรวม (Integrated) การเก็บข้อมูลแยกตามเนื้อหา (Subject-Oriented) ข้อมูลที่ทำการเก็บนั้นมีความสัมพันธ์กับช่วงระยะเวลา (Time-Variant) และข้อมูลที่เก็บเข้าไปในคลังข้อมูลจะไม่สูญสลาย (Nonvolatile) ซึ่งมีไว้เพื่อสนับสนุนการตัดสินใจ โดยคุณสมบัติของแต่ละข้อมีดังต่อไปนี้

1) ฐานข้อมูลที่มีการรวบรวม (Integrated) หมายความว่า ฐานข้อมูลคลังข้อมูล (Data Warehouse) นั้นเป็นฐานข้อมูลที่เป็นศูนย์กลาง ซึ่งนำข้อมูลทั้งหมดที่ได้มาจากทั้งองค์กรมารวมไว้ด้วยกัน การที่จะรวบรวมข้อมูลต่าง ๆ ให้เป็นหนึ่งเดียวนั้นหมายถึงว่าจะต้องมีการจัดการเป็นอย่างดี เพื่อที่จะกำหนดและหามาตรฐานที่เหมาะสมให้กับทุกส่วนของข้อมูล การรวบรวมข้อมูลจึงต้องใช้เวลา เมื่อทำสำเร็จก็จะสามารถให้ภาพรวมของทั้งองค์กรได้ การรวบรวมข้อมูลนั้นเป็นการยกระดับการตัดสินใจ และช่วยให้ผู้บริหารสามารถเข้าใจขั้นตอนของธุรกิจได้ดีมากยิ่งขึ้น

2) การเก็บข้อมูลแยกตามเนื้อหา (Subject-Oriented) หมายความว่า การทำฐานข้อมูลคลังข้อมูล (Data Warehouse) ข้อมูลจะถูกเตรียมทำให้สมบูรณ์มากที่สุดเท่าที่จะเป็นไปได้ เพื่อเตรียมการสำหรับการตอบคำถามของปัญหาต่าง ๆ ดังนั้นฐานข้อมูลคลังข้อมูล (Data Warehouse) จึงประกอบไปด้วยข้อมูลที่ได้รับการจัดการและสรุปรวมตามหัวข้อหลักต่าง ๆ ซึ่งในแต่ละหัวข้อหลัก ๆ ของฐานข้อมูลคลังข้อมูล (Data Warehouse) ก็ยังประกอบไปด้วยหัวข้อเฉพาะที่สนใจอีกด้วย

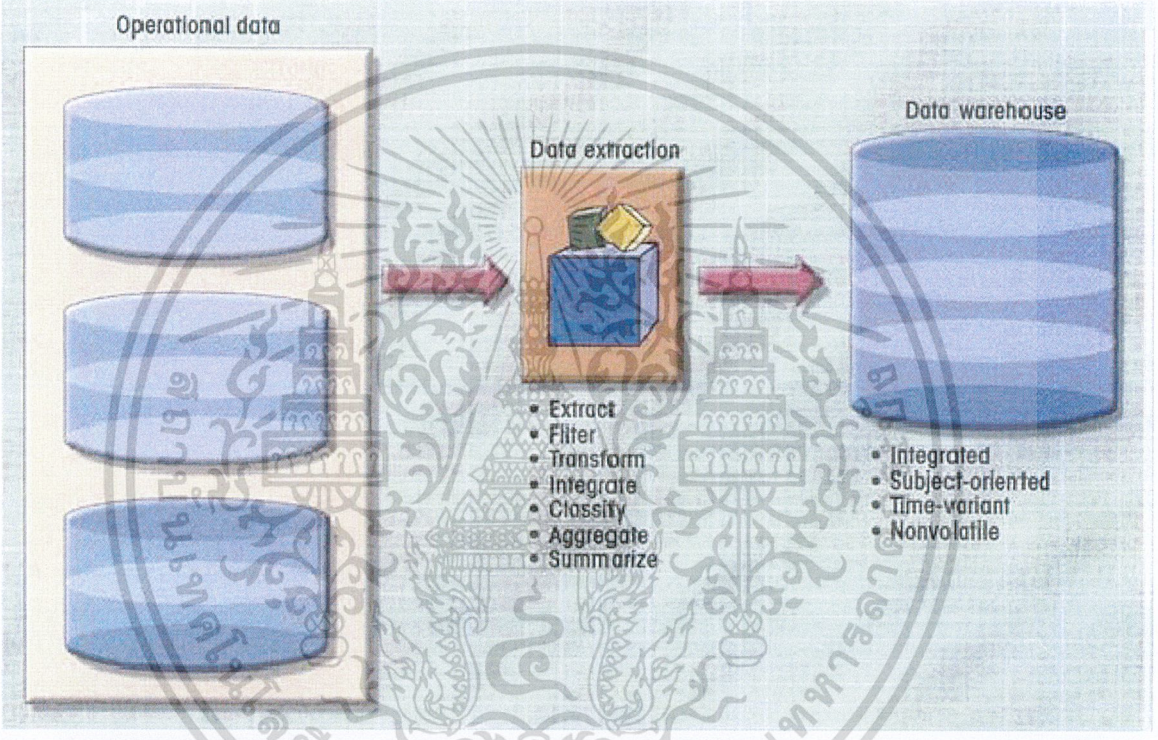
3) ข้อมูลที่ทำการเก็บนั้นมีความสัมพันธ์กับช่วงระยะเวลา (Time-Variant) หมายความว่า ฐานข้อมูลคลังข้อมูล (Data Warehouse) จะแสดงการเคลื่อนที่ของข้อมูลต่อเวลา เมื่อข้อมูลเก็บมาถึงระยะเวลาที่กำหนดก็จะทำการบรรจุข้อมูลลงไปในฐานข้อมูลคลังข้อมูล (Data Warehouse)

4) ข้อมูลที่เก็บเข้าไปในคลังข้อมูลจะไม่สูญสลาย (Nonvolatile) หมายความว่า เมื่อข้อมูลถูกบรรจุลงในฐานข้อมูลคลังข้อมูล (Data Warehouse) แล้วก็จะคงอยู่ในนั้น ไม่มีการนำข้อมูลออก เนื่องจากคลังข้อมูลนั้นจะแสดงข้อมูลเกี่ยวกับประวัติทั้งหมดขององค์กร สาเหตุจากการที่ข้อมูลในคลังข้อมูล (Data Warehouse) นั้นไม่มีการลบข้อมูลทิ้ง แต่มีการเพิ่มข้อมูลขึ้นเรื่อย ๆ จึงทำให้คลังข้อมูล (Data Warehouse) นั้นมีขนาดใหญ่ขึ้น ดังนั้น DBMS จึงต้องสามารถที่จะจัดการกับข้อมูลจำนวนมาก ๆ ได้

ตารางที่ 2.2 การเปรียบเทียบลักษณะของฐานข้อมูลคลังข้อมูล (Data Warehouse) และ ฐานข้อมูลการทำงานปกติ (Operational Database)

ลักษณะ	ข้อมูลในฐานข้อมูลการทำงานปกติ	ข้อมูลในฐานข้อมูลคลังข้อมูล
Integrated	ข้อมูลที่เหมือนกันอาจมีการนำเสนอและมีความหมายที่แตกต่างกัน เช่น เบอร์โทรศัพท์อาจเก็บเป็น #-####-#### หรือ #########, เงื่อนไขอาจจะเก็บเป็น T/F, 0/1 หรือ Y/N	สร้างมุมมองที่สอดคล้องกันของส่วนประกอบของข้อมูลด้วยการจำกัดความร่วมกันและนำเสนอทั้งองค์กร
Subject-oriented	ข้อมูลจะถูกเก็บตามกระบวนการทำงานของระบบ	ข้อมูลจะถูกเก็บตามหัวข้อที่สนใจ ที่จะนำมาช่วยตัดสินใจ
Time-variant	ข้อมูลจะถูกบันทึกตามการประมวลผลรายการ (Transaction) ปัจจุบันที่เกิดขึ้น	ข้อมูลจะถูกบันทึกเป็นประวัติเพื่อทำให้ง่ายสำหรับการวิเคราะห์และเปรียบเทียบข้อมูลในระยะเวลาที่แตกต่างกัน

Nonvolatile	การเปลี่ยนแปลงข้อมูลเกิดขึ้นบ่อยครั้ง	ข้อมูลไม่มีการเปลี่ยนแปลง แต่จะมีการเพิ่มข้อมูลเข้าไปตามช่วงระยะเวลา



รูปที่ 2.7 การสร้างฐานข้อมูลคลังข้อมูล

ข้อมูลในฐานข้อมูลคลังข้อมูล (Data Warehouse) จะต้องเป็นข้อมูลที่รวมเป็นอันหนึ่งอันเดียวกันเพื่อให้เกิดความสม่ำเสมอกับทุก ๆ องค์ประกอบ คำว่า Data Integration มีความหมายว่าในทุก ๆ ส่วนของระบบ ส่วนประกอบของข้อมูล และลักษณะต่าง ๆ ของข้อมูลนั้นจะต้องถูกอธิบายไปในทางเดียวกันทั้งระบบ ถึงแม้ว่าสิ่งนี้จะเป็นความต้องการในทางตรรกะ เนื่องจากมีหลากหลายวิธีในการวัดผลการทำงานยกตัวอย่างเช่นการความแตกต่างในการวัดผลการทำงานเกี่ยวกับด้านการขายภายในองค์กร และความหลากหลายนี้ก็เกิดขึ้นกับส่วนประกอบอื่น ๆ ที่อยู่ภายในองค์กรด้วย ซึ่งตัวอย่างที่ยกมานี้เป็นเพียงตัวอย่างของปัญหาเพียงเล็กน้อยที่จะต้องเผชิญในการรวบรวมข้อมูลในการทำมาสร้างเป็นคลังข้อมูล (Data Warehouse) นอกจากนี้การใช้คำจำกัดความที่แตกต่างกันใน

การอธิบายถึงข้อมูลตัวเดียวกันก็เป็นปัญหา ยกตัวอย่าง ในแผนกต่าง ๆ อาจมีวิธีการคำนวณ และการวัดค่าที่แตกต่างกันออกไป ดังตัวอย่าง เกี่ยวกับสถานะของการสั่งซื้อ ในแผนกหนึ่งอาจแสดงเป็น “เปิด” “รับ” “ยกเลิก” หรือ “ปิด” แต่ในแผนกอื่นอาจใช้แทนสถานะดังกล่าวว่า “1”, “2”, “3” หรือ “4” เป็นต้น และอีกตัวอย่างเกี่ยวกับการบอกสถานะนักศึกษาในแผนกการบัญชีอาจใช้ “freshman” , “sophomore” , “junior” หรือ “senior” แต่ในแผนกการลงทะเบียนอาจใช้ “FR” , “SO” , “JR” หรือ “SP” ดังนั้นเพื่อที่จะหลีกเลี่ยงการเกิดปัญหาวุ่นวายเกี่ยวกับการกำหนดรูปแบบข้อมูลคลังข้อมูล (Data Warehouse) จึงจำเป็นต้องปฏิบัติตามรูปแบบที่เหมือนกันเพื่อเป็นข้อตกลงร่วมกันทั้งระบบ

ข้อควรจำเกี่ยวกับการทำฐานข้อมูลการทำงานปกติ (Operational Database) จะเป็นการทำงานในรูปแบบที่มุ่งความสนใจไปยังกระบวนการที่มาเปลี่ยนแปลงข้อมูล ดังนั้นนักออกแบบที่มาทำการออกแบบระบบการทำใบแจ้งราคาสินค้า (Invoice) จะมุ่งประเด็นไปที่การออกแบบในเรื่องการนอร์มอลไลซ์ (Normalized) โครงสร้างของข้อมูล เพื่อสนับสนุนการทำงานของระบบโดยการเก็บข้อมูลของใบแจ้งราคาสินค้า (invoice) เป็น 2 ตาราง ได้แก่ ตาราง INVOICE และ ตาราง INVLINE อีกนัยหนึ่งเนื่องจากฐานข้อมูล คลังข้อมูล (Data Warehouse) มีคุณสมบัติคือ Subject-oriented (Subject กล่าวถึงการนำส่วนประกอบของข้อมูลในฐานข้อมูลการทำงานปกติ (Operational Database) นำมาพิจารณาเกี่ยวกับการวิเคราะห์และทำการรวบรวมทำให้ได้ข้อมูลที่เราสนใจ) นักออกแบบฐานข้อมูลคลังข้อมูล (Data Warehouse) จะมุ่งความสนใจไปที่ตัวข้อมูลมากกว่ากระบวนการที่มาปรับเปลี่ยนข้อมูล (นอกจากนี้ข้อมูลในฐานข้อมูลคลังข้อมูล (Data Warehouse) นั้นไม่ใช่ข้อมูลที่มีการเปลี่ยนแปลงแบบ real-time) ดังนั้นฐานข้อมูลคลังข้อมูล (Data Warehouse) จะไม่เก็บข้อมูลการทำใบแจ้งราคาสินค้า (Invoice) แต่จะทำการเก็บข้อมูลเกี่ยวกับสินค้าและลูกค้าแทน เนื่องจากกิจกรรมการสนับสนุนการตัดสินใจนั้นต้องการผลสรุปเกี่ยวกับการขายสินค้าและลูกค้าเพียงเท่านั้น

ข้อมูลในฐานข้อมูลคลังข้อมูล (Data Warehouse) เป็นข้อมูลที่ถูกประกอบขึ้นมาจากการรวบรวมข้อมูลที่เป็นประวัติที่ผ่านมาขององค์กรตามตัวแปรก็คือเวลา ดังนั้นส่วนประกอบที่เกี่ยวกับเวลาจึงมีความสำคัญ ในการสร้างฐานข้อมูลคลังข้อมูล (Data Warehouse) จึงต้องมี time ID เพื่อเป็นการบอกเวลาที่เหมาะสมในการทำการเก็บรวบรวมข้อมูล และเมื่อข้อมูลถูกเก็บลงในฐานข้อมูลคลังข้อมูล (Data Warehouse) ตัว time ID จะถูกกำหนดให้กับตัวข้อมูลนั้นและไม่สามารถทำการเปลี่ยนแปลงได้

โดยสรุปแล้วฐานข้อมูลคลังข้อมูล (Data Warehouse) นั้นจะสร้างขึ้นเพื่อเป็นฐานข้อมูลที่ไว้ใช้อ่านเพียงอย่างเดียว สำหรับเก็บข้อมูลที่จะนำมาใช้เพื่อการวิเคราะห์ และการถาม (Query) ข้อมูล โดยทั่วไปแล้วข้อมูลจะถูกคัดลอกออกมาจากแหล่งข้อมูลต่าง ๆ จากนั้นก็ถูกเปลี่ยนสภาพและทำให้ข้อมูลนั้นอยู่ในรูปแบบเดียวกันก่อนที่จะถูกนำไป

1. ความเหมาะสมในการนำคลังข้อมูลเข้ามาใช้

การทำคลังข้อมูลเหมาะสมกับองค์กรที่ มีข้อมูลถูกเก็บอยู่ในระบบที่แตกต่างกัน มีการใช้วิธีการในการจัดการกับข้อมูล (Information-Base Approach) มีลูกค้าจำนวนมาก มีข้อมูลเดียวกันที่ถูกนำไปใช้แสดงแตกต่างไปใน แต่ละระบบ ข้อมูลถูกเก็บด้วยวิธีการและมีรูปแบบที่ยากต่อการนำมาใช้ ระบบการปฏิบัติงานที่มีอยู่ยังไม่มีการเก็บ ข้อมูลเก่าๆ อย่างรวดเร็ว มีข้อมูลที่ต้องการเก็บอยู่ในหลายๆ ระบบการปฏิบัติงาน และมีประสิทธิภาพในการสอบถามข้อมูลยังไม่ดีพอ โดยการทำคลังข้อมูลมีคุณประโยชน์และข้อดี ดังต่อไปนี้

2. ข้อดีของการทำคลังข้อมูล

1) สนับสนุนการวิเคราะห์ และการตัดสินใจทางธุรกิจโดยการสร้างฐานข้อมูลรวมที่มี รูปแบบตรงกัน แบ่งตามเนื้อหาที่สนใจ และมีการเก็บข้อมูลเก่าๆ ไว้ใช้ในการวิเคราะห์ได้

2) มีการรวบรวมข้อมูลจากหลายๆ ระบบที่มีรูปแบบไม่เหมือนกันมาไว้ในฐานข้อมูลเดียวกัน และมีการแปลงข้อมูลให้เป็นสารสนเทศที่มีความหมาย

3) ทำให้ผู้จัดการสามารถทำการวิเคราะห์ข้อมูลได้อย่างรวดเร็วและถูกต้อง

4) ลดค่าใช้จ่าย ประหยัดเวลา และเพิ่มผลผลิตในการดำเนินการ

5) แยกการทำงานในส่วนของฐานข้อมูล ซึ่งทำให้ระบบการประมวลผลรายการช้าออกจากการประมวลผลแบบเร่งด่วน ทำให้สามารถเข้าถึงข้อมูลที่ต้องการได้ง่ายและรวดเร็วขึ้น

6) มีความสามารถสรุปข้อมูลในระดับสูง

7) ปรับปรุงความรู้ในด้านธุรกิจ

8) ปรับปรุงการเลือกกลุ่มเป้าหมายในตลาดได้ดีขึ้นทำให้ได้เปรียบคู่แข่ง เพิ่มความพอใจในการบริการให้กับลูกค้าได้

2.6 Business Intelligence: BI (OLAP : Online Analytical Processing)

BI : Business Intelligence

ธุรกิจอัจฉริยะ (BI) คือ กระบวนการสำหรับการเพิ่มความสามารถในการแข่งขันของธุรกิจ โดยอาศัยข้อมูลที่อยู่มาใช้ในการตัดสินใจ โดยการนำเอาข้อมูลสารสนเทศที่มีอยู่มาใช้ก่อให้เกิดประโยชน์สูงสุด เพื่อช่วยให้เกิดการตัดสินใจที่ถูกต้องและแม่นยำ เพิ่มความได้เปรียบในการแข่งขันของธุรกิจโดยการใช้ข้อมูลที่มีอยู่ได้อย่างอัจฉริยะ ธุรกิจอัจฉริยะ คือ การเข้าถึงการวิเคราะห์ และการค้นพบโอกาสใหม่ๆ โดยใช้เทคโนโลยีเป็นส่วนประกอบที่ทำให้ประสบผลสำเร็จ

Online Analytical Processing (OLAP)

ความจำเป็นสำหรับการสนับสนุนการตัดสินใจที่เพิ่มมากขึ้น ทำให้เกิดเครื่องมือรุ่นใหม่ที่เรียกว่า Online Analytical Processing (OLAP) ซึ่งสามารถสร้างการวิเคราะห์ข้อมูลที่มีความก้าวหน้า และมีส่วนช่วยในการสนับสนุนการตัดสินใจ สนับสนุนโครงสร้างของธุรกิจ และกิจกรรมสำหรับการค้นคว้า ระบบ OLAP แบ่งออกเป็น 3 ส่วน ได้แก่

- ใช้เทคนิคการวิเคราะห์ข้อมูลแบบ Multidimensional
- จัดหาเครื่องมือที่สนับสนุนฐานข้อมูล
- จัดหา End-User Interface ที่ง่ายต่อการใช้งาน

1) ใช้เทคนิคการวิเคราะห์ข้อมูลแบบ Multidimensional

สิ่งนี้เป็นลักษณะที่เด่นชัดที่สุดของ OLAP การวิเคราะห์ข้อมูลที่เป็น Multidimensional หมายถึง กระบวนการของข้อมูลที่ถูกมองว่าเป็นส่วนของโครงสร้างแบบ Multidimensional ความน่าสนใจในเกณฑ์ของ Multidimensional ในการวิเคราะห์ข้อมูลเกิดจากข้อเท็จจริงที่ว่า ผู้ตัดสินใจมักมองข้อมูลจากทรรศนะทางธุรกิจ (Business Perspective) ให้มีแนวโน้มที่จะเชื่อมโยงกับข้อมูลทางธุรกิจด้านอื่นๆ

เพื่อให้มองภาพได้ง่ายขึ้น ต้องดูว่านักวิเคราะห์ข้อมูลทางธุรกิจต้องการที่จะตรวจสอบมุมมองในแง่ของการขายให้เป็นไปในรูแบบใด ในกรณีนี้ พวกเขาอาจจะมีความสนใจในมุมมอง

ของการขายสินค้า โดยมองว่าการขายมีความสัมพันธ์อย่างไรกับตัวแปรทางธุรกิจอื่นๆ เช่น ลูกค้า และเวลา เป็นต้น

การมองโดย End User เกี่ยวกับข้อมูลการขายจะถูกแสดงให้เห็นอย่างใกล้ชิด โดยมุมมอง Multidimensional ได้ชัดเจนมากกว่ามุมมองที่เป็นของตารางที่แยกออกจากกัน นอกจากนี้มุมมองแบบ Dimensional ยังช่วยให้ End User สามารถรวบรวมข้อมูล (Aggregate Data) ที่ระดับต่างๆ ได้ เช่น ยอดรวมการขายที่แสดงโดยลูกค้า และโดยวัน ประการสุดท้ายมุมมอง Dimensional ของข้อมูลช่วยให้นักวิเคราะห์ข้อมูลทางธุรกิจสะดวกในการสับเปลี่ยนทรรศนะทางธุรกิจ จากการขายที่แสดงจากลูกค้าเป็นการขายจากแผนก เขต และอื่นๆ ได้อย่างสะดวก เป็นต้น

เทคนิคการวิเคราะห์ข้อมูลแบบ Multidimensional อาจจะเพิ่มเติมได้จากฟังก์ชันดังต่อไปนี้

ฟังก์ชันการแสดงข้อมูล : กราฟฟิกส์ 3 มิติ , ตาราง Pivot , Crosstab , การหมุนข้อมูล (Data Rotation) , ลูกบาศก์ 3 Dimension (Three Dimensional Cube) เป็นต้น เครื่องมือที่ใช้แสดงข้อมูลเหล่านี้จะเข้ากันได้กับเดสก์ทอป Spreadsheets, แพ็คเกจที่เป็นสถิติ (Statistical Package) และแพ็คเกจการสอบถาม (Query) และการทำรายงาน

ฟังก์ชันการรวบรวมข้อมูล : (Data Aggregation) และการจำแนกข้อมูล (Data Classification) ซึ่งจะทำให้นักวิเคราะห์ทางธุรกิจสามารถสร้างลำดับชั้นของข้อมูลได้หลายระดับชั้น การ Slice และ Dice ข้อมูล และ การ Drill Down การ Row Up ข้าม Dimension ของเวลาได้

ฟังก์ชันการคำนวณ : จากตัวแปรต่างๆ ทางธุรกิจ (ส่วนแบ่งตลาด, การเปรียบเทียบตามช่วงเวลา, จำนวนเผื่อเหลือเผื่อขาดในการขาย, จำนวนเผื่อเหลือเผื่อขาดของสินค้า, เปอร์เซ็นต์ในการเปลี่ยนแปลง และ อื่นๆ) อัตราส่วนทางด้านการเงินและการบัญชี (กำไร, ส่วนที่สิ้นเปลือง, ต้นทุนที่ต้องเสีย, ความคุ้มค่า เป็นต้น), ฟังก์ชันทางสถิติและการคำนวณเป็นต้น ฟังก์ชันเหล่านี้จะถูกจัดให้โดยอัตโนมัติและ End User ไม่จำเป็นต้องกำหนดองค์ประกอบเหล่านี้ใหม่ในแต่ละครั้งที่เข้าถึง

ฟังก์ชันรูปแบบของข้อมูล : สำหรับให้การสนับสนุนคำถามประเภท “What-If” , การประเมินความเปลี่ยนแปลง (Variable Assessment), ตัวแปรที่สนับสนุนผลลัพธ์, โปรแกรมเชิงเส้น (Linear Programming) และเครื่องมืออื่นๆ

เนื่องจากฟังก์ชันการวิเคราะห์และการแสดงข้อมูลมักจะมีอยู่ในแพคเกจเดสก์ทอป Spreadsheet ดังนั้นผู้ผลิต OLAP ส่วนใหญ่จึงมักจะเชื่อมโยงระบบบอย่างใกล้ชิดกับเดสก์ทอป Spreadsheet เช่น Microsoft Excel และ Lotus 1-2-3 การใช้ลักษณะซึ่งหาได้ง่ายใน Graphical End User Interface เช่น Window ทำให้ทางเลือกของเมนูใน OLAP กลายเป็นอีกทางเลือกหนึ่งใน Lotus หรือตัวเมนูบาร์ของ Excel การเชื่อมโยงที่กลมกลืนกันนี้กลายเป็นอีกหนึ่งข้อได้เปรียบสำหรับระบบของ OLAP และสำหรับผู้ผลิต Spreadsheet เนื่องจาก End User สามารถเข้าถึงเทคนิคการวิเคราะห์ข้อมูลชั้นสูงได้ โดยการใช้โปรแกรม และ Interface ที่คุ้นเคยได้ ดังนั้นจึงเป็นการลดต้นทุนในการฝึกอบรมและพัฒนาได้อย่างมาก

2) จัดหาเครื่องมือที่สนับสนุนฐานข้อมูล

เพื่อทำให้การสนับสนุนการตัดสินใจเป็นไปอย่างมีประสิทธิภาพ เครื่องมือ OLAP จึงต้องมีรูปแบบในการเข้าถึงข้อมูล ซึ่งประกอบไปด้วย

- สามารถเข้าถึง DBMS, Flat File และแหล่งข้อมูลทั้งภายใน และภายนอกได้หลากหลายชนิด

- เข้าถึงข้อมูลที่ทำการรวมเก็บไว้ในคลังข้อมูล (Data Warehouse) ได้ดีเท่ากับการเข้าถึงข้อมูลในฐานข้อมูลการทำงานปกติ (Operational Database)

- มีลักษณะเด่นในการทำ Data Navigation เช่น การ Drill-Down และ Roll-Up

- เวลาในการตอบสนองการสอบถาม (Query) รวดเร็วสม่ำเสมอ

- มีความสามารถในการจัดวางเค้าโครงของการร้องขอจาก End-User ที่ชัดเจน และส่งคำร้องขอนั้นๆ ไปยังแหล่งข้อมูลที่เหมาะสม โดยเลือกใช้ภาษาที่ใช้เข้าถึงข้อมูลที่มีเหมาะสมด้วย (ส่วนมากจะเป็นภาษา SQL) ต้องมีการปรับคำสั่ง (Code) ในการสอบถาม (Query) ให้เหมาะสมเพื่อให้สามารถจับคู่ให้ถูกกับแหล่งข้อมูล โดยไม่สนใจว่าแหล่งข้อมูลจะเป็นฐานข้อมูลทำงานปกติ (Operational Database) หรือฐานข้อมูลคลังข้อมูล (Data Warehouse)

- สนับสนุนสำหรับฐานข้อมูลขนาดใหญ่ เนื่องจากฐานข้อมูลคลังข้อมูล (Data Warehouse) ขยายตัวได้ง่ายและรวดเร็วจนอาจเป็น Gigabytes หรือแม้แต่ Terabytes

ในการให้ Interface กลมกลืนกัน เครื่องมือ OLAP จะวางเค้าโครง Data Dictionary จากฐานข้อมูลคลังข้อมูล (Data Data warehouse) และจากฐานข้อมูลทำงานปกติ (Operational Database) จากนั้น Metadata เหล่านี้จะถูกใช้ต่อเพื่อแปลงการร้องขอจาก End-User ให้เป็นคำสั่ง (Code) ในการสอบถาม (Query) ที่เหมาะสม (หรือปรับให้เหมาะสม) ซึ่งจากนั้นก็จะถูกนำไปยังแหล่งข้อมูลที่เหมาะสมต่อไป

3) จัดหา End-User Interface ที่ง่ายต่อการใช้งาน

OLAP จะมีประโยชนมากขึ้นหากสามารถเข้าถึงได้ง่าย และผู้จำหน่าย (Vendor) ก็ได้เรียนรู้จุดนี้ และติดตั้งเครื่องมือการดึงข้อมูล (Data Extraction) ต่างๆ ที่มีความซับซ้อนและเครื่องมือวิเคราะห์ให้มี Interface ที่เป็นกราฟฟิกส์ช่วยให้เข้าใช้งานได้ง่าย โดย Interface จำนวนมากถูกยืมมาจากเครื่องมือการวิเคราะห์ข้อมูลรุ่นก่อนหน้านี้ ซึ่งเป็นที่คุ้นเคยของผู้ใช้อยู่แล้ว ทำให้ OLAP นั้นง่ายต่อการยอมรับและใช้งานได้รวดเร็ว

2.7 การ Extract Transform load (ETL)

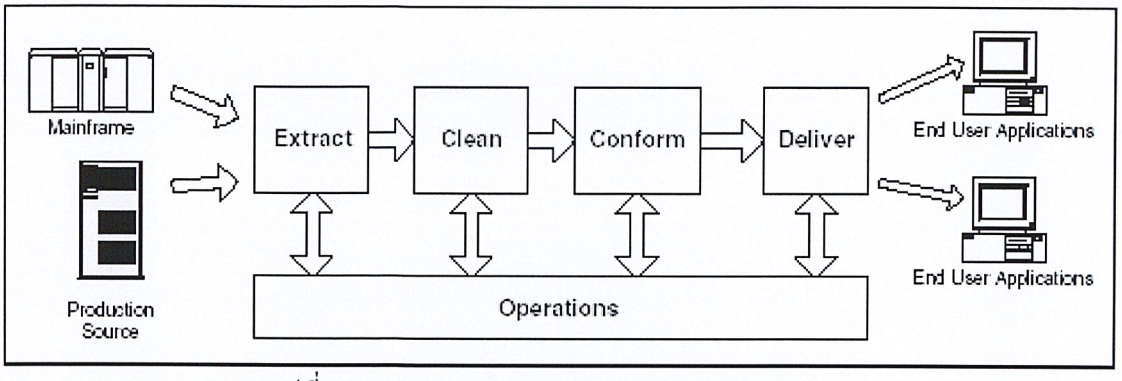
ETL (Extract-Transform-Load) คือกระบวนการหนึ่งในระบบ Data Warehouse โดยระบบที่ออกแบบเอาไว้จะดึงข้อมูลออกมาจากหลายๆ ที่, นำกระบวนการตรวจสอบคุณภาพของข้อมูลมาประยุกต์ใช้, มีการเชื่อมโยงและปรับข้อมูลให้เป็นไปในรูปแบบเดียวกันเพื่อให้ ข้อมูลจากหลายๆ แหล่งสามารถใช้งานร่วมกันได้ และท้ายที่สุดทำการส่งมอบ (Delivery) ข้อมูลเหล่านั้นในรูปแบบที่ง่ายต่อการใช้งาน เพื่อใช้ในการตัดสินใจขององค์กร

โดยมีกระบวนการหลักๆ ที่เกี่ยวข้องกับกระบวนการต่างๆ ดังต่อไปนี้

• Extract - กระบวน การดึงข้อมูลจากแหล่งของข้อมูลภายนอก

• Transforming – แปลง ข้อมูลเพื่อให้ได้ตรงตามกับความต้องการ ซึ่งเป็นกระบวนการที่ต้องใช้วิธีการเชิงคุณภาพ

• Loading – นำ ข้อมูลเข้าสู่ระบบปลายทางที่ต้องการ ซึ่งโดยทั่วไปจะหมายถึงระบบ Data Warehouse หรือ ฐานข้อมูลอื่นใด



รูปที่ 2.8 แสดงกระบวนการทำงานของระบบ ETL

บทที่ 3
วิธีการดำเนินงาน

3.1 รายละเอียดของระบบงาน

ในปัจจุบันองค์กรที่ให้บริการพื้นฐานทางด้านเทคโนโลยีสารสนเทศและการสื่อสาร (ICT : Information Communication Technology) ต้องมีการติดตั้งอุปกรณ์ที่ทำงานด้านระบบเครือข่าย (Network System) โดยมีการจัดเก็บข้อมูลรายละเอียดการทำงานต่างๆ อย่างเช่น มีการทำงานอะไรบ้าง ทำงานอย่างไร ทำงานเมื่อไรและมีการส่งผ่านข้อมูลว่าไปที่ไหนด้วยวิธีการอะไร โดยข้อมูลทุกอย่างจะถูกจัดเก็บไว้ที่ Log File ซึ่งเป็นแฟ้มข้อมูลเฉพาะอุปกรณ์ดังกล่าวที่มีรูปแบบการจัดเก็บข้อมูลที่แตกต่างกัน และถ้านำข้อมูลดังกล่าวมาวิเคราะห์และเชื่อมโยงทั้งหมดโดยนำ Log File ของทุกอุปกรณ์มาบูรณาการร่วมกันโดยอาศัยข้อมูล Global Time Data จาก Time Server จะก่อให้เกิดประโยชน์ที่มีความสำคัญมากสำหรับการบริหารจัดการ ICT Infrastructure เพราะข้อมูลดังกล่าวทั้งหมดสามารถนำมาวิเคราะห์ให้เห็นถึงพฤติกรรมของผู้ใช้งานในองค์กร และวิเคราะห์พฤติกรรมของการรุกรานหรือการใช้งานที่ไม่พึงประสงค์ได้

จากลักษณะการทำงานดังกล่าวจะมีข้อมูลจำนวนมากที่ต้องถูกนำมาทำงานร่วมกัน ดังนั้นต้องมีการพัฒนาระบบงานที่ใช้หลักการทำงานของคลังข้อมูล (Data Warehouse) ขึ้นมาเพื่อรองรับการจัดเก็บข้อมูลจำนวนมากดังกล่าว โดยจะต้องมีการออกแบบคลังข้อมูลเพื่อรองรับการจัดเก็บข้อมูลและนำหลักการทำงานของ Business Intelligence มาประยุกต์ใช้เพื่อพัฒนาระบบงานในการนำข้อมูลทั้งหมดไปวิเคราะห์เฉพาะอุปกรณ์หรือนำไปวิเคราะห์การทำงานของอุปกรณ์ทั้งหมดในภาพรวมตามความต้องการของผู้ใช้งาน

3.2 รายงานความต้องการของผู้บริหาร

1. รายงานแสดงพฤติกรรมการรุกรานหรือการใช้งานที่ไม่พึงประสงค์
   1.1 รายงานแสดงประเภทการรุกรานแต่ละประเภทจำแนกตามช่วงเวลา
   1.2 รายงานแสดงจำนวนการรุกรานทั้งหมดจำแนกตามช่วงเวลา
2. รายงานแสดง Ip Address ต้นทางที่โจมตีจำนวนมากที่สุดในแต่ละช่วงเวลา

3.3 การออกแบบระบบ

3.3.1 Data Warehouse Bus

3.3.1.1 รายงานแสดงพฤติกรรมการรุกรานหรือการใช้งานที่ไม่พึงประสงค์

	Time				ประเภทการโจมตี	อุปกรณ์
	วัน	สัปดาห์	เดือน	ปี		
จำนวน	✓	✓	✓	✓	✓	✓

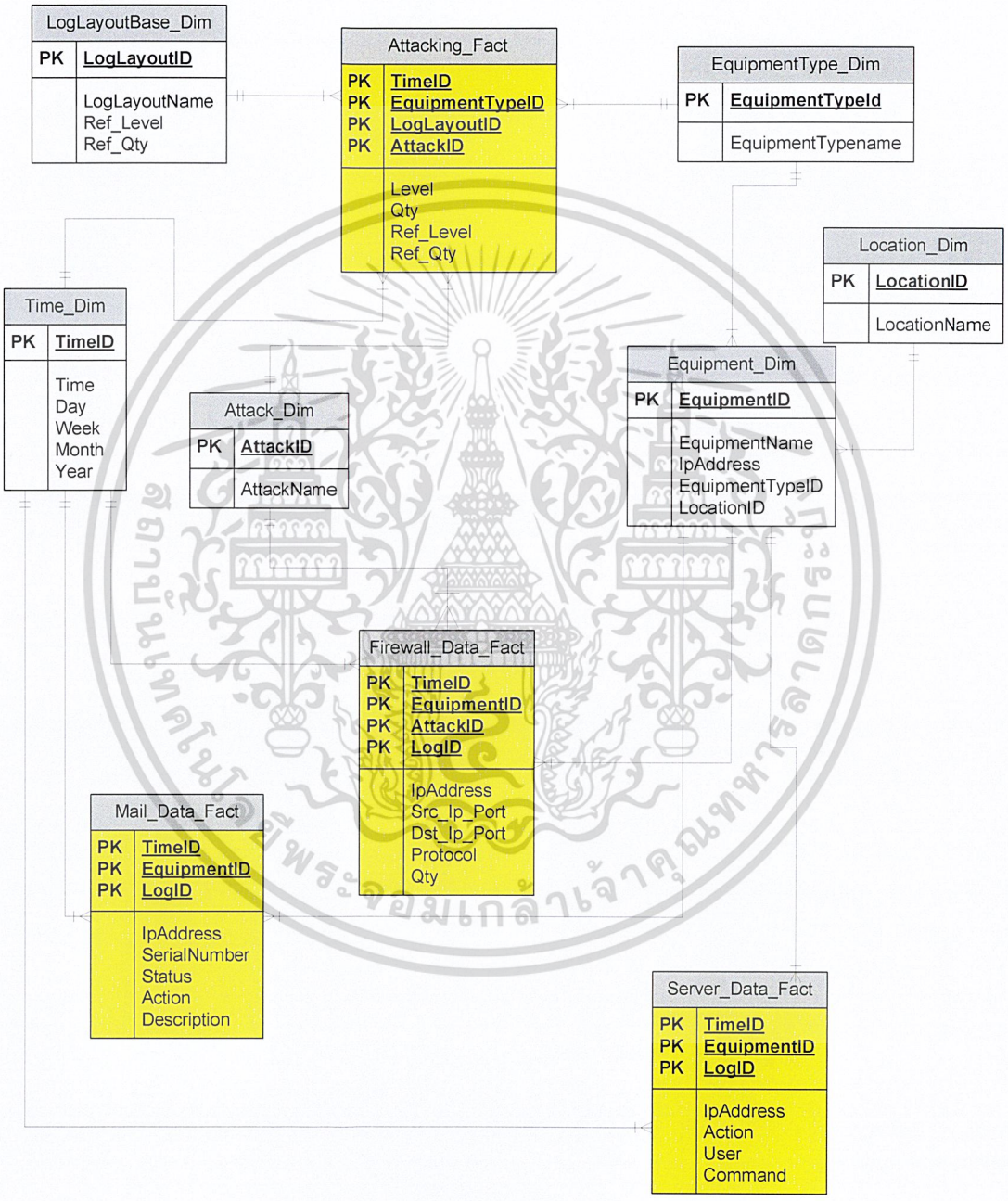
ตารางที่ 3.1 Data Warehouse Bus ของรายงานแสดงพฤติกรรมการรุกราน

3.3.1.2 รายงานแสดง Ip Address ที่โจมตีเข้ามาจำนวนมากที่สุด

	Time				การโจมตี	อุปกรณ์	Ip Address
	วัน	สัปดาห์	เดือน	ปี			
จำนวน	✓	✓	✓	✓	✓		✓

ตารางที่ 3.2 Data Warehouse Bus ของรายงานแสดง Ip Address

3.3.2 Star Schema



รูปที่ 3.1 โครงสร้างของ Star Schema

3.3.3 โครงสร้างตารางใน Star Schema

ตารางที่ 3.3 ตารางแสดงตารางทั้งหมดที่เกี่ยวข้องกับ Network logfile

ลำดับที่	ชื่อตาราง	ความหมาย
1	Equipment_Dim	อุปกรณ์เน็ตเวิร์ก
2	EquipmentType_Dim	ประเภทอุปกรณ์เน็ตเวิร์ก
3	Location_Dim	สถานที่ตั้งอุปกรณ์เน็ตเวิร์ก
4	Attack_Dim	การโจมตี
5	LogLayoutBase_Dim	มาตรฐานของล็อกไฟล์
6	Time_Dim	เวลา
7	Attacking_Fact	การเปรียบเทียบล็อกไฟล์
8	Firewall_Data_Fact	ข้อมูลล็อกไฟล์ประเภท firewall
9	Mail_Data_Fact	ข้อมูลล็อกไฟล์ประเภท mail gateway
10	Server_Data_Fact	ข้อมูลล็อกไฟล์ประเภท syslog-server

ตารางที่ 3.4 ตารางอุปกรณ์เน็ตเวิร์ก (Equipment_Dim)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	EquipmentID	INT(3)	รหัสอุปกรณ์เน็ตเวิร์ก	P.K
2	EquipmentName	VARCHAR(50)	ชื่ออุปกรณ์เน็ตเวิร์ก	N.N
3	IpAddress	VARCHAR(15)	Ip Address	N.N
4	EquipmentTypeID	INT(2)	รหัสประเภทอุปกรณ์เน็ตเวิร์ก	F.K
5	LocationID	INT(2)	รหัสสถานที่ตั้งอุปกรณ์เน็ตเวิร์ก	F.K

ตารางที่ 3.5 ตารางประเภทอุปกรณ์เน็ตเวิร์ก (EquipmentType_Dim)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	EquipmentTypeID	INT(2)	รหัสประเภทอุปกรณ์เน็ตเวิร์ก	P.K
2	EquipmentTypeName	VARCHAR(50)	ชื่อประเภทอุปกรณ์เน็ตเวิร์ก	N.N

ตารางที่ 3.6 ตารางสถานที่ตั้งอุปกรณ์เน็ตเวิร์ก (Location_Dim)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	LocationID	INT(2)	รหัสสถานที่ตั้งอุปกรณ์เน็ตเวิร์ก	P.K
2	LocationName	VARCHAR(50)	ชื่อสถานที่ตั้งอุปกรณ์เน็ตเวิร์ก	N.N

ตารางที่ 3.7 ตารางการโจมตี (Attack_Dim)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	AttackID	INT(2)	รหัสการโจมตี	P.K
2	AttackName	VARCHAR(50)	ชื่อการโจมตี	N.N

ตารางที่ 3.8 ตารางมาตรฐานล็อกไฟล์ (LogLayoutBase_Dim)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	LogLayoutID	INT(15)	รหัสค่ามาตรฐานของล็อกไฟล์	P.K
2	LogLayoutName	VARCHAR(50)	ชื่อล็อกไฟล์	N.N
3	Ref_Level	INT(3)	ระดับอ้างอิง	N.N
4	Ref_Qty	INT(7)	จำนวน	N.N

ตารางที่ 3.9 ตารางเวลา(Time_Dim)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	TimeID	INT(5)	รหัสเวลา	P.K
2	Time	VARCHAR(20)	เวลา	N.N
3	Day	VARCHAR(20)	วัน	N.N
4	Week	VARCHAR(20)	สัปดาห์	N.N
5	Month	VARCHAR(20)	เดือน	N.N
6	Year	VARCHAR(20)	ปี	N.N

ตารางที่ 3.10 ตารางการเปรียบเทียบล็อกไฟล์ (Attacking_Fact)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	TimeID	INT(5)	รหัสเวลา	P.K.
2	EquipmentTypeID	INT(2)	รหัสประเภทอุปกรณ์เน็ตเวิร์ก	P.K
3	LogLayoutID	INT(15)	รหัสค่ามาตรฐานของล็อกไฟล์	P.K
4	AttackID	INT(2)	รหัสการโจมตี	P.K
5	Level	INT(3)	ระดับ	N.N
6	Qty	INT(7)	จำนวน	N.N
7	Ref_Level	INT(3)	ระดับอ้างอิง	N.N
8	Ref_Qty	INT(7)	จำนวนอ้างอิง	N.N

ตารางที่ 3.11 ตารางข้อมูลล็อกไฟล์ประเภท firewall (Firewall_Data_Fact)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	TimeID	INT(5)	รหัสเวลา	P.K
2	EquipmentID	INT(2)	รหัสอุปกรณ์เน็ตเวิร์ก	P.K
3	AttackID	INT(2)	รหัสการโจมตี	P.K
4	LogID	INT(20)	รหัสล็อกไฟล์	P.K
5	IpAddress	VARCHAR(15)	Ip Address	N.N
6	Src_Ip_Port	VARCHAR(30)	Ip และ Port ต้นทาง	N.N
7	Dst_Ip_Port	VARCHAR(30)	Ip และ Port ปลายทาง	N.N
8	Protocol	VARCHAR(20)	โปรโตคอล	N.N
9	Qty	INT(7)	จำนวนการโจมตี	

ตารางที่ 3.12 ตารางข้อมูลล็อกไฟล์ประเภท mail gateway (Mail_Data_Fact)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	TimeID	INT(5)	รหัสเวลา	P.K
2	EquipmentID	INT(2)	รหัสอุปกรณ์เน็ตเวิร์ก	P.K
3	LogID	INT(20)	รหัสล็อกไฟล์	P.K
4	IpAddress	VARCHAR(15)	Ip Address	N.N
5	SerialNumber	VARCHAR(30)	Serial Number	N.N
6	Status	VARCHAR(1)	สถานะ	N.N
7	Action	VARCHAR(5)	การกระทำ	N.N
8	Description	VARCHAR(100)	ข้อมูลอธิบายการกระทำ	N.N

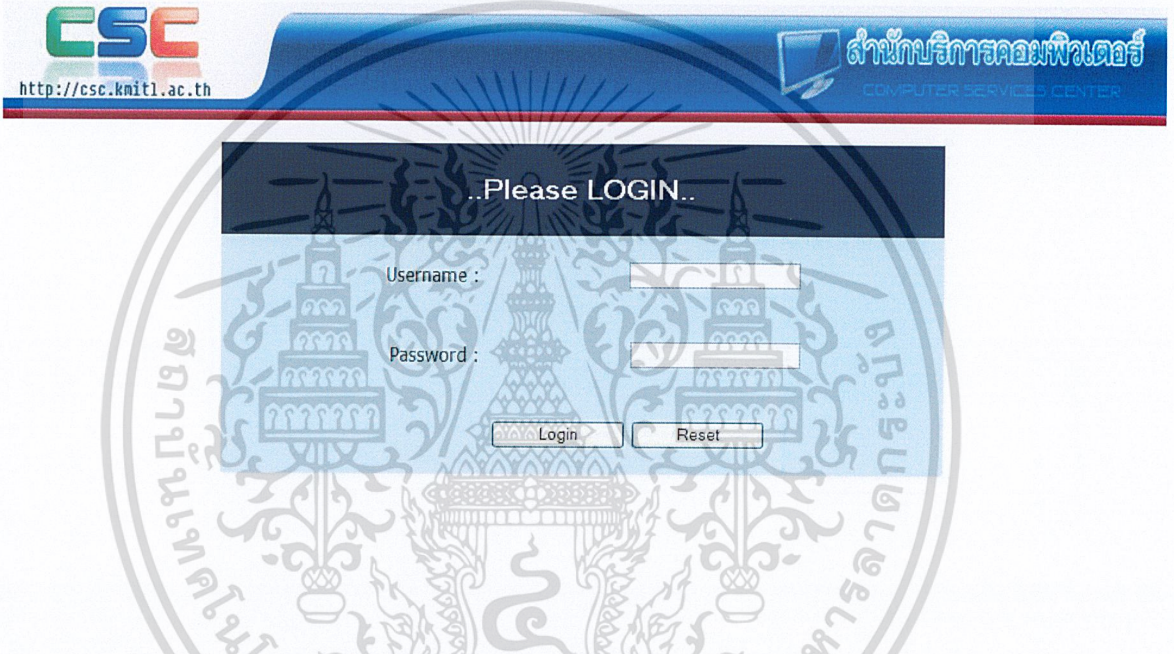
ตารางที่ 3.13 ตารางข้อมูลล็อกไฟล์ประเภท syslog-server (Server_Data_Fact)

ลำดับที่	ชื่อ column	ชนิดข้อมูล	ความหมาย	Key
1	TimeID	INT(5)	รหัสเวลา	P.K
2	EquipmentID	INT(2)	รหัสอุปกรณ์เน็ตเวิร์ก	P.K
3	LogID	INT(20)	รหัสล็อกไฟล์	P.K
4	IpAddress	VARCHAR(15)	Ip Address	N.N
5	Action	VARCHAR(30)	การกระทำ	N.N
6	User	VARCHAR(20)	ผู้ใช้งาน	N.N
7	Command	VARCHAR(100)	คำสั่ง	N.N

บทที่ 4

ผลการดำเนินงาน

4.1 หน้าจอและวิธีการใช้งาน



รูปที่ 4.1 หน้าจอในการล็อกอินเข้าสู่ระบบ

ในหน้านี้จะเป็นหน้าจอในการล็อกอินเข้าสู่ระบบเพื่อเข้าไปใช้งานระบบ และเมื่อทำการ Log out จากระบบจะกลับมาสู่หน้าจอนี้ โดย User password ที่ใช้จะเป็นดังนี้

Username : admin และ Password : admin

รูปที่ 4.2 หน้าจอหลักในการใช้งานระบบ

เป็นหน้าจอเมนูหลักของระบบให้ได้ใช้จะมีอยู่ 4 เมนูหลัก ซึ่งประกอบไปด้วยเมนู ดังนี้

- Home
- Generete XML
- Report &Graph
- About ME
- Contact US
- Log Out

4.1.1) Home

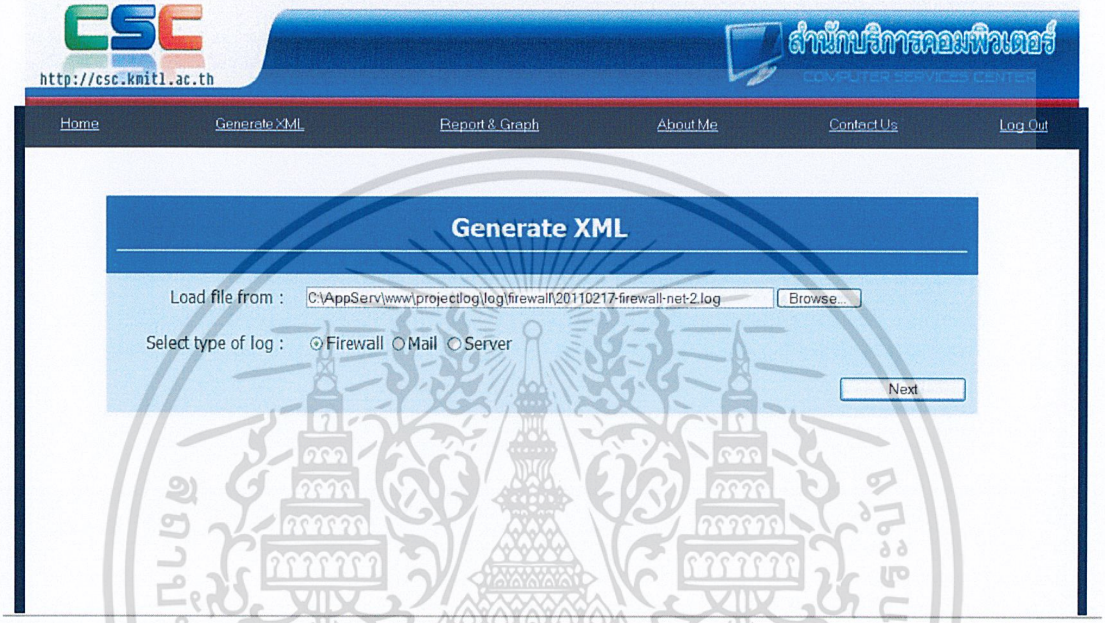
เป็นเมนูที่ใช้เพื่อกลับไปหน้าจอหลักของระบบ

4.1.2) Generate XML

เป็นเมนูที่ใช้ในการแปลงข้อมูลต่างที่รับเข้ามา ให้อยู่ในรูปของ XML ซึ่งจะมีวิธีการแปลงอยู่ 2 วิธีดังนี้

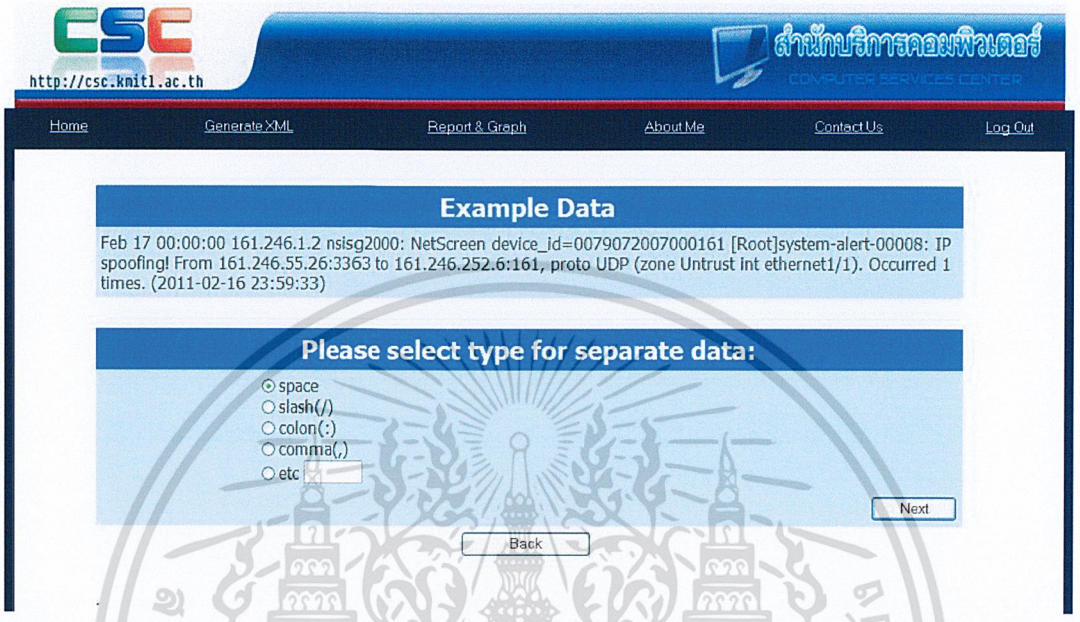
4.1.2.1) ตั้งค่าเอง

ในหน้าแรกจะเป็นหน้าจอที่ใช้ในการเลือกไฟล์ล็อกที่ต้องการทำเป็นไฟล์ xml พร้อมระบุด้วยว่าล็อกนั้นเป็นประเภทไหน แล้วกด Next ดังรูปที่ 4.3



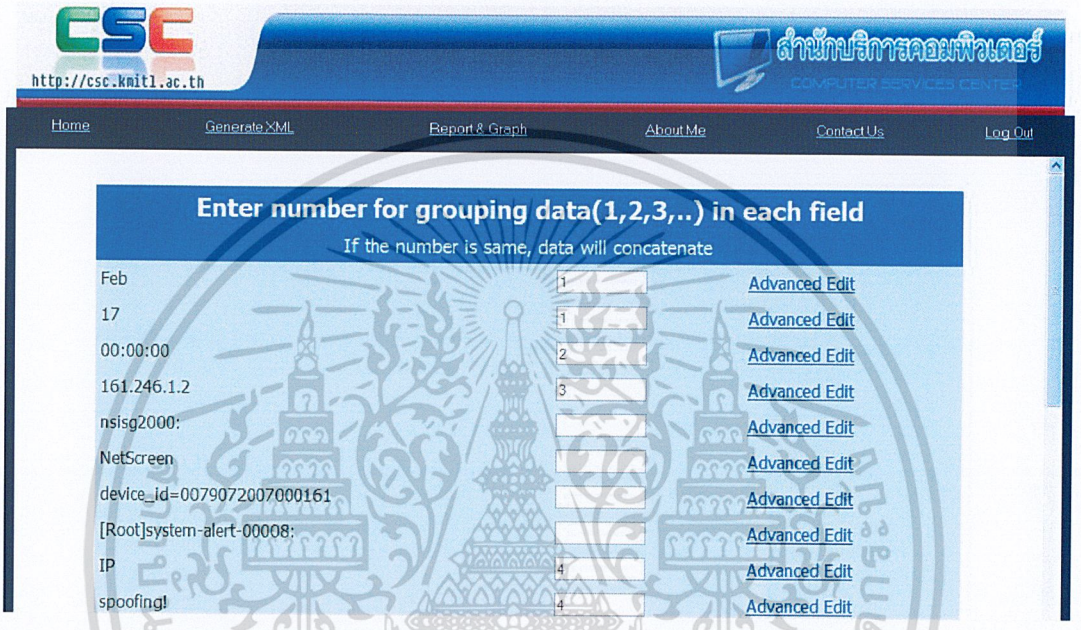
รูปที่ 4.3 หน้าจอ Generate XML(1)

เมื่อกด Next แล้ว ต่อมาจะเป็นหน้าจอแสดงตัวอย่างข้อมูลล็อกในไฟล์ที่ได้เลือกไว้ในหน้าแรกขึ้นมาและให้เลือกสัญลักษณ์หรือเครื่องหมายที่ใช้ในการตัด ดังรูปที่ 4.4



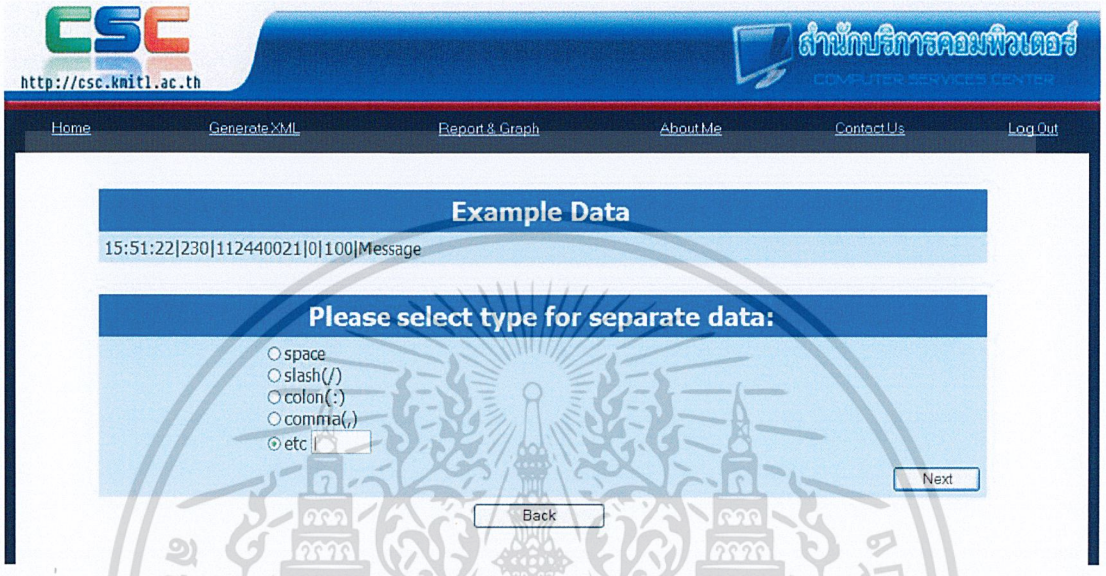
รูปที่ 4.4 หน้าจอ Generate XML(2)

เมื่อเราได้มีการเลือกสัญลักษณ์หรือเครื่องหมายที่ใช้ในการตัดแล้วเราก็จะต้องมีการแบ่งกลุ่มข้อมูลที่เราตัดออกมาให้อยู่ในรูปแบบที่เราต้องการ โดยในการแบ่งกลุ่มนั้นเราจะใช้ตัวเลขแทนการจัดกลุ่ม ตัวไหนที่เราต้องการให้อยู่ด้วยกันเราจะให้อยู่ในเลขเดียวกัน แต่ตัวไหนที่อยู่คนละกลุ่มก็ให้เขียนแตกต่างกัน ส่วนตัวไหนที่เราไม่ต้องการนั้นเราก็จะไม่ใส่ค่าลงไป ดังรูปที่ 4.5



รูปที่ 4.5 หน้าจอ Generate XML(3)

ในกรณีที่ข้อมูลที่ตัดออกมานั้นยังไม่ตรงตามความต้องการเราสามารถตัดอีกทีได้ โดยกดปุ่ม Advance Edit ซึ่งเมื่อกดปุ่มนี้จะขึ้นหน้าจอที่กำหนดว่าจะใช้สัญลักษณ์หรือเครื่องหมายใดตัด ดังรูปที่ 4.6



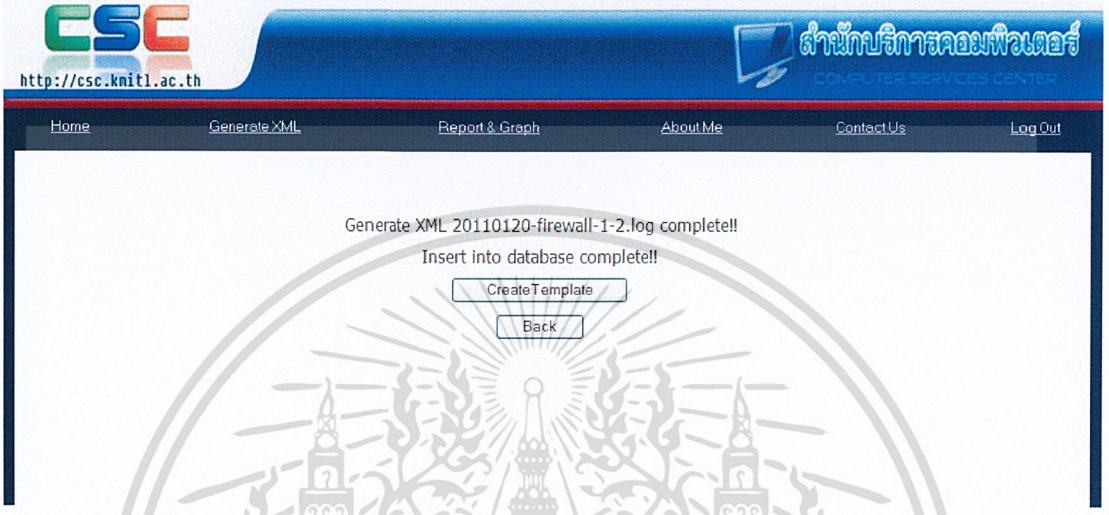
รูปที่ 4.6 หน้าจอ Generate XML(4)

ต่อจากนั้นก็จะเป็นการแบ่งกลุ่มข้อมูลซึ่งจะได้ออกมาเป็นดังรูปที่ 4.5

ต่อมาเมื่อเราตัดข้อมูลได้ตามที่เราต้องการแล้วนั้นเราต้องมีการกำหนดชื่อข้อมูลว่าข้อมูลที่เข้ามานั้นเป็นข้อมูลอะไร พร้อมทั้งใส่ชื่อ Database และชื่อ table เพื่อนำข้อมูลเหล่านี้ลงฐานข้อมูล ดังรูปที่ 4.7

รูปที่ 4.7 หน้าจอ Generate XML(5)

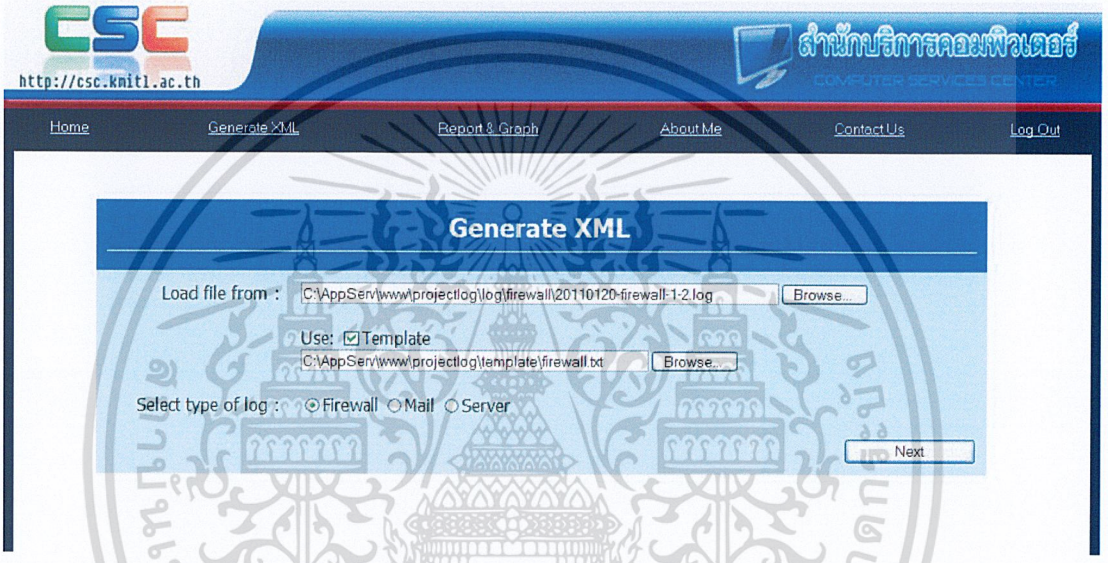
เมื่อทำการใส่ข้อมูลทั้งหมดแล้ว ระบบจะแสดงข้อความให้ทราบว่าได้ทำการสร้างไฟล์ xml และนำข้อมูลเหล่านี้ลงฐานข้อมูลเรียบร้อยแล้ว ซึ่งตรงนี้เราจะสามารถสร้าง Template ของเราเองได้โดย Click ไปที่ Create Template ดังรูปที่ 4.8



รูปที่ 4.8 หน้าจอ Generate XML(6)

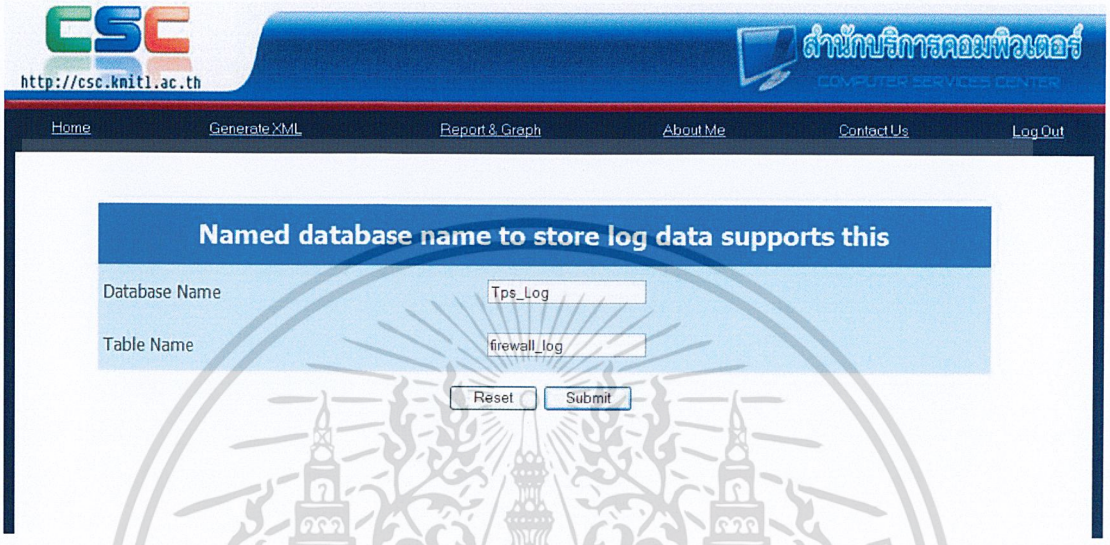
4.1.2.2) ใช้ Template

ในหน้าแรกจะเป็นหน้าจอที่ใช้ในการเลือกไฟล์ล็อกที่ต้องการทำเป็นไฟล์ xml พร้อมระบุด้วยว่าล็อกนั้นเป็นประเภทไหน แล้ว Tick ตรงช่อง Use Template ละ เลือก Template ที่เราต้องการขึ้นมา



รูปที่ 4.9 หน้าจอ Generate XML(7)

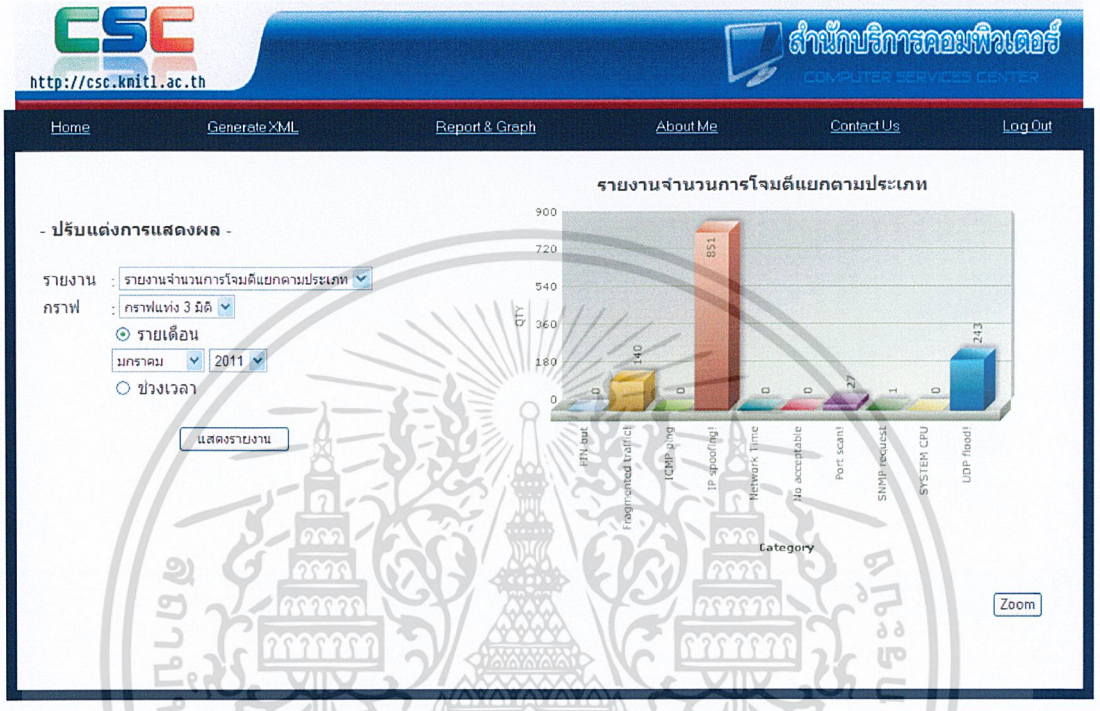
เมื่อกด Next แล้วระบบจะ Gen ข้อมูลทั้งหมด แล้วให้เราเลือกว่าเราต้องการจะนำข้อมูลที่ได้มานั้นใส่ไว้ใน Database และ table อะไร ดังรูป 4.10 แล้วเมื่อข้อมูลเข้าหมดแล้ว จะขึ้นหน้าใจ ดังรูป 4.8



รูปที่ 4.10

4.1.3) Report &Graph

เป็นเมนูที่แสดงรายงานด้านต่าง ๆ ของระบบ ซึ่งมีวิธีการใช้งาน ดังนี้



รูปที่ 4.11 หน้าจอแสดงรายงานและกราฟ

ในหน้าแรกจะเป็นหน้าจอที่เกี่ยวกับการเลือกรายงานและรูปแบบการแสดงรายงานต่างๆ ซึ่งการเลือกหรือปรับแต่งต่างๆนั้น จะอยู่ทางด้านซ้ายมือของหน้าจอ ส่วนทางด้านขวามือจะเป็นการแสดงรายงานต่างๆตามที่ได้เลือกไว้ ดังรูปที่ 4.11

โดยสามารถปรับแต่งหน้าจอตามที่ต้องการได้ตามเมนู ดังนี้

- รายงาน

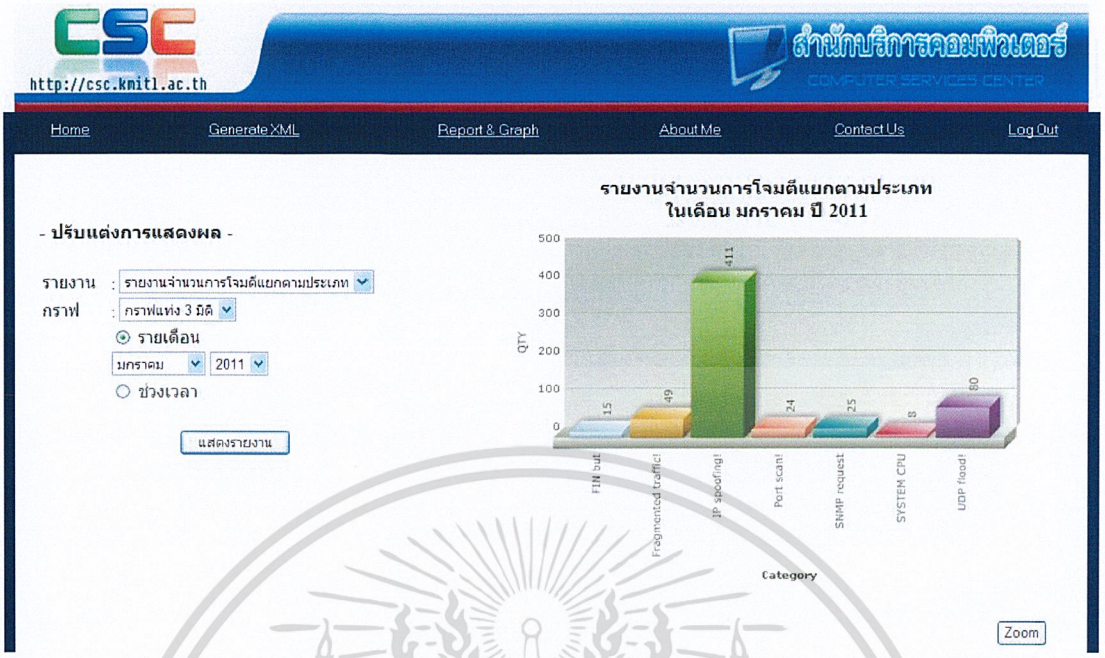
  จะเป็นการเลือกรายงานตามที่ต้องการ

- กราฟ

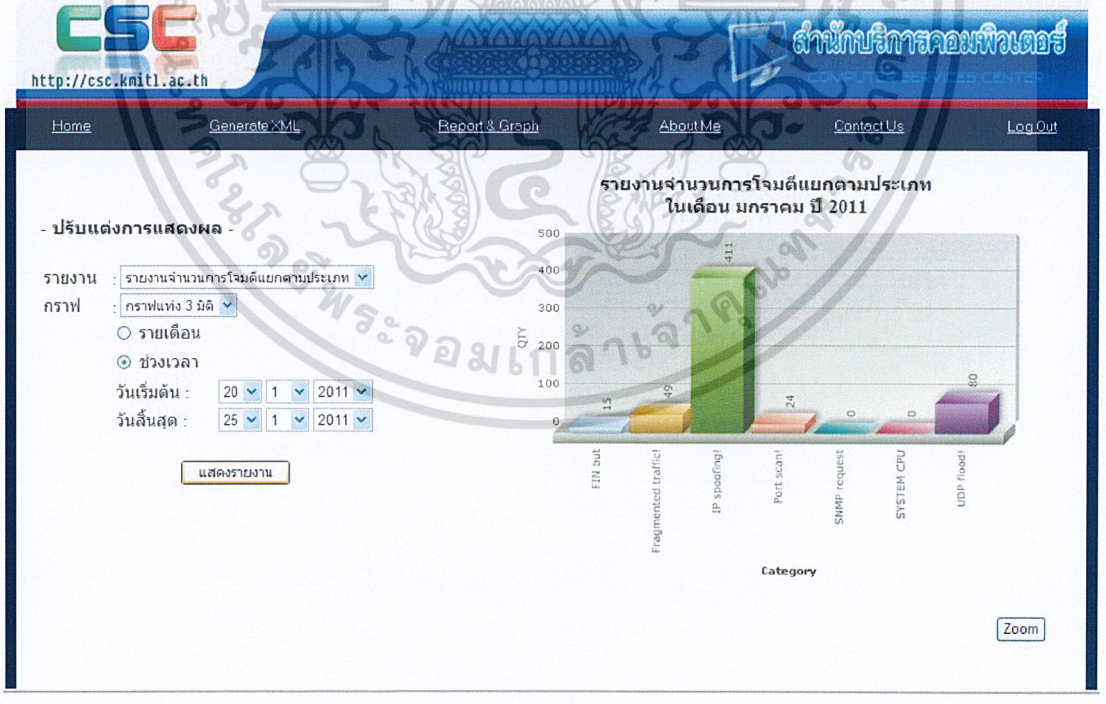
  จะเป็นการเลือกกราฟแสดงผลตามที่ต้องการ

- เวลา

  เลือกเวลาตามที่ต้องการโดยแบ่งออกเป็น รายเดือนและช่วงเวลา

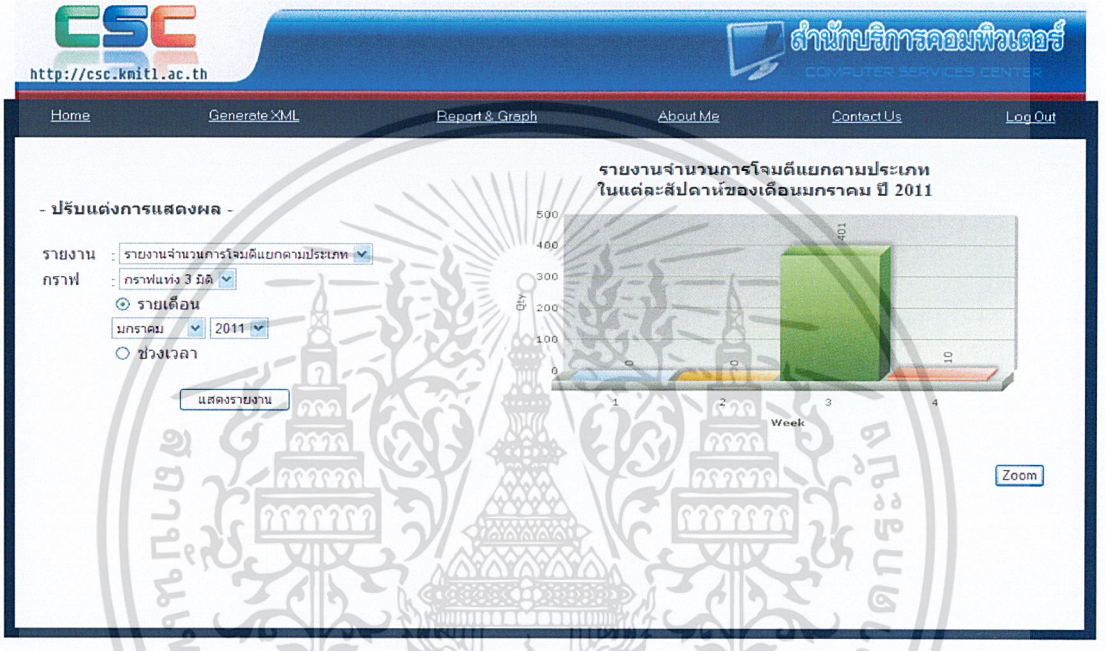


รูปที่ 4.12 แสดงรายงานตามเวลาแบบรายเดือน

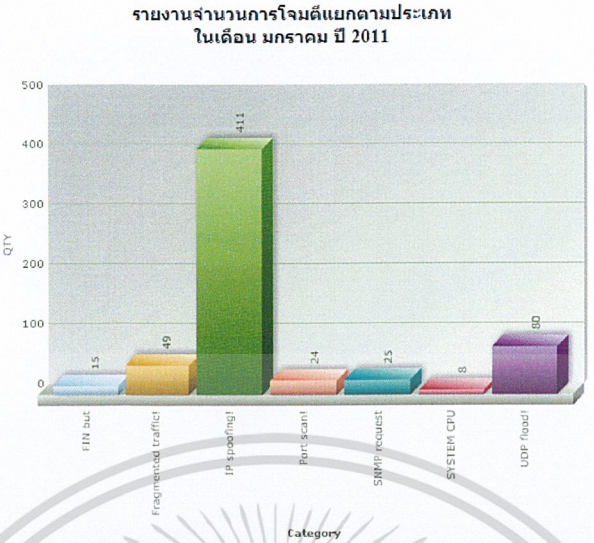


รูปที่ 4.13 แสดงรายงานตามเวลาแบบช่วงเวลา

และสามารถดูกราฟขนาดใหญ่ได้โดยกดปุ่ม Zoom หากต้องการดูข้อมูลย่อยของกราฟแต่ละแท่ง ก็สามารถเลือกดูได้โดยการคลิกเลือกที่แท่งกราฟที่ต้องการจะดูข้อมูล ดังรูปที่ 4.12 และ รูปที่ 4.13



รูปที่ 4.14 หน้าจอแสดงรายงานและกราฟ (Drill-Down)



ชื่อ	จำนวน
Fragmented traffic!	49
IP spoofing!	411
Port scan!	24
SNMP request	25
SYSTEM CPU	8
UDP flood!	80

รูปที่ 4.15 หน้าจอแสดงรายงานและกราฟ (Scale)

4.1.4) About Me

เป็นเมนูที่แสดงถึงข้อมูลผู้จัดทำ และรายละเอียดต่างๆของผู้จัดทำ

4.1.5) Contact US

เป็นเมนูที่ใช้ในการติดต่อกับผู้สร้างระบบ

4.1.6) Log Out

เป็นเมนูที่ใช้ในการ Log Out จากระบบ

บทที่ 5

สรุปผลการพัฒนาโปรแกรมและข้อเสนอแนะ

5.1 ผลการวิจัยและพัฒนา

ในการทำการศึกษาการวิจัย และการพัฒนาระบบการการพัฒนาคลังข้อมูลเพื่อรองรับการจัดการข้อมูลจาก Network Log File สามารถสรุปได้ดังนี้

5.1.1 สรุปผลการทำงานของโปรแกรม

เมื่อได้ทำการศึกษา รวบรวมข้อมูลต่างๆ เพื่อใช้เป็นแนวทางในการทำการออกแบบคลังข้อมูล (Data Warehouse) ได้มีการทำการศึกษาความต้องการของผู้บริหารในเรื่องของรายงานต่างๆที่ต้องการตามแต่ละช่วงเวลา รวมถึงซักถามเกี่ยวกับปัญหาต่างๆ ที่เกิดขึ้นเพื่อเป็นแนวทางในการนำมาออกแบบคลังข้อมูล (Data Warehouse) ให้มีความสอดคล้องในการทำการแก้ไขปัญหาต่างๆ ที่เกิดขึ้นเพื่อให้คลังข้อมูล (Data Warehouse) ที่ได้ออกแบบมานั้นมีความครอบคลุมถึงลักษณะความต้องการในการใช้งานของผู้ใช้งานให้ได้มากที่สุด นอกจากนี้ยังได้มีการพูดคุยและทำการซักถามปัญหาของข้อมูลในระบบ TPS (ข้อมูล Log file) ที่มีความสงสัยในด้านต่างๆ เช่น ความหมายของ Log ที่มีลักษณะแปลกไป รวมไปถึงข้อมูล Log ที่ไม่สามารถทำความเข้าใจได้ เป็นต้น ซึ่งจากการซักถามข้อมูลต่างๆ เบื้องต้นนั้น ทำให้สามารถเก็บข้อมูลเหล่านั้นมาทำการออกแบบ XML แล้วปรับเปลี่ยนรูปแบบข้อมูลที่เข้ามาให้อยู่ในรูปแบบของ XML ตามที่ได้ออกแบบไว้ได้ เพื่อที่จะสามารถส่งข้อมูลในรูปแบบนี้ไปยังฐานข้อมูลต่างๆ หรือเครื่องคอมพิวเตอร์อื่นๆ ได้อย่างมีประสิทธิภาพและรวดเร็วต่อการใช้งาน โดยการปรับเปลี่ยนข้อมูลเป็น XML จะอยู่ในรูปแบบ dynamic ซึ่งเป็นรูปแบบที่ผู้ใช้งานสามารถทำการเลือกข้อมูลได้ตามที่ต้องการ โดยข้อมูลเหล่านั้นต้องเป็นไปตามโครงสร้างของ XML ที่ได้ทำการออกแบบไว้แล้วเบื้องต้น และจากข้อมูลที่ศึกษามานั้นยังสามารถมาทำการออกแบบคลังข้อมูล (Data Warehouse) เพื่อให้รองรับการใช้งานตามความต้องการของผู้ใช้และผู้บริหารให้ได้มากที่สุด โดยระบบที่ออกแบบนั้นจะออกแบบไปตามรายงานที่ผู้บริหารมักต้องการที่จะเรียกดูตามแต่ละช่วงเวลา เพื่อเป็นข้อมูลในการพิจารณาในการดำเนินงานในด้านต่างๆ ต่อไปของผู้บริหาร ซึ่งได้ทำการแสดงผล

ลัพธ์ของข้อมูลดังกล่าวออกมาในรูปแบบของกราฟและแผนภาพต่างๆ เพื่อให้เกิดความน่าสนใจในการดูข้อมูลที่มีจำนวนมาก อีกทั้งยังเพิ่มความสวยงามของระบบงานอีกด้วย และเมื่อทราบถึงความต้องการของผู้บริหารที่ชัดเจนแล้ว ก็ได้นำข้อมูลต่างๆ เหล่านั้นมาทำการออกแบบคลังข้อมูล (Data Warehouse) และทำขั้นตอนของการ ETL(Extract Transform Load) ข้อมูลจากฐานข้อมูล SQL Server ลงสู่ฐานข้อมูล MySQL ตามโครงสร้างของคลังข้อมูล (Data Warehouse) ที่ได้ทำการออกแบบไว้ และหลังจากนั้นจึงนำมาพัฒนาให้อยู่ในรูปแบบของเว็บแอพพลิเคชั่น(Web Application) โดยงานจะครอบคลุมในส่วนของการสร้างเมนูในการเรียกใช้งานของผู้ใช้งานซึ่งในการออกรายงานของผู้ใช้งานสามารถอออกรายงานได้แบบ static report ซึ่ง static report นั้นจะเป็นรายงานที่มีความถี่ค่อนข้างมากซึ่งเกี่ยวข้องกับลูกค้าที่ผู้บริหารมักต้องการเรียกดูข้อมูลในแต่ละช่วงเวลา ซึ่งจะต้องเป็นไปตามโครงสร้างของคลังข้อมูล(Data Warehouse) ที่ได้ทำการออกแบบไว้แล้วเบื้องต้น ซึ่งจะทำให้โปรแกรมนั้นมีความยืดหยุ่นในการใช้งานมากขึ้น

5.1.2 การวิเคราะห์และออกแบบรายงาน

ในการทำการวิเคราะห์และออกแบบรายงานเพื่อพัฒนาระบบนั้นได้ใช้ Star Schema เป็นเครื่องมือในการจำลองความสัมพันธ์ของข้อมูล เพื่อให้เกิดความครอบคลุมในเรื่องความต้องการข้อมูลที่มีอยู่ในปัจจุบัน รวมไปถึงยังได้ทำการออกแบบฐานข้อมูลเพื่อรองรับข้อมูลที่จะมีการเพิ่มขึ้นในอนาคต

5.1.3 การวิเคราะห์หาความสัมพันธ์ของข้อมูลและออกแบบฐานข้อมูล

ในการทำการวิเคราะห์ออกแบบฐานข้อมูลนั้น จะใช้ Star Schema ซึ่งได้ทำการแสดงไว้แล้วในบทที่ 3

5.1.4 การพัฒนาโปรแกรมคอมพิวเตอร์

การพัฒนาโปรแกรมนั้น ได้ทำการพัฒนามาจากภาษา PHP (Personal Home Page), Java script, CSSและ XML โดยได้ทำการพัฒนาโปรแกรมบน Text Editor ที่ชื่อ Netbeans ภายใต้ระบบปฏิบัติการ Microsoft Window XP โดยมี Appserv และ MySQL ทำหน้าที่เป็น Web Server และ Database Server ตามลำดับ

5.1.5 การติดตั้งการใช้งาน

การพัฒนาโปรแกรมนั้นเพื่อช่วยให้ผู้ใช้งานสามารถทำงานได้สะดวกรวดเร็วมากขึ้น เนื่องจากโปรแกรมถูกออกแบบมาให้สามารถใช้งานได้ง่ายและสะดวก อีกทั้งช่วยให้การจัดเก็บข้อมูลเป็นระเบียบเรียบร้อย และง่ายในการเรียกดูข้อมูลรายงานต่างๆ ตามความต้องการในแต่ละช่วงเวลา

5.2 สรุปประสิทธิภาพของโปรแกรม

ผลการประมวลผลที่สำคัญคือ

1) สามารถออกรายงานที่เป็น static report ได้ ซึ่งเป็นรายงานที่ผู้บริหารมักจะเรียกดูเป็นประจำ โดยสามารถดูข้อมูลได้ในรูปแบบของกราฟได้
2) สามารถทำการ drill-down และ row-up ในการทำการดูข้อมูลในรายงานต่างๆ ได้
3) สามารถเรียกดูข้อมูลได้ทันทีจากคอมพิวเตอร์

5.3 ข้อเสนอแนะ

ในการพัฒนาระบบนั้นเป็นการพัฒนาระบบในส่วนของคลังข้อมูล(Data Warehouse)ที่ต้องมีการพัฒนาให้มีความยืดหยุ่นมากยิ่งขึ้น และยังมีส่วนของการออกรายงานให้เป็นไปตามความต้องการของผู้บริหารให้ได้มากที่สุด ซึ่งไม่สามารถที่จะทำการวิเคราะห์ข้อมูลที่เกิดจากความสัมพันธ์ต่างๆ ได้โดยอัตโนมัติ ซึ่งเมื่อผู้บริหารได้ทำการดูข้อมูลต่างๆ เหล่านี้ไปแล้วก็ต้องทำการนำข้อมูลเหล่านี้ไปวิเคราะห์ต่อเพื่อให้ได้ข้อสรุปของแนวโน้มทางด้านการตลาดและการวางแผนงานต่างๆ ต่อไป ดังนั้นหากสามารถทำให้ระบบงานนี้สามารถมีข้อเสนอแนะและวิเคราะห์ข้อมูลแนวโน้มต่างๆ ในเบื้องต้นได้ ก็จะเป็นเครื่องมืออีกอย่างหนึ่งที่จะช่วยในการคิดและวิเคราะห์ข้อมูลต่างๆของผู้บริหาร ซึ่งจะทำให้การตัดสินใจในด้านต่างๆ มีประสิทธิภาพ และเกิดความคุ้มทุนมากที่สุดได้ โดยอาจจะต้องทำการเน้นการพัฒนาไปในด้านของการวิเคราะห์ข้อมูลในทาง Data Mining เพิ่มเข้ามา

เอกสารอ้างอิง

กฤษฎา บุศรา. 2551. เอกสารประกอบการสอนวิชาระบบฐานข้อมูล. กรุงเทพฯ: สถาบันพระจอมเกล้าเจ้า คุณทหารลาดกระบัง

กฤษฎา บุศรา. 2551. เอกสารประกอบการสอนวิชาความรู้พื้นฐานเกี่ยวกับการโปรแกรม SQL และ PL/SQL. กรุงเทพฯ: สถาบันพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

วิสันต์ ตั้งวงษ์เจริญ. 2552. เอกสารประกอบคำสอนวิชา การเขียนโปรแกรมเว็บ. กรุงเทพฯ: สถาบันพระจอมเกล้าเจ้า คุณทหารลาดกระบัง

Xml(ออนไลน์). (2553). สืบค้นจาก: www.w3schools.com [21 พฤศจิกายน 2553]

ภาคผนวก

1. Syslog-ng

1.1. การติดตั้ง Syslog-ng

syslog-ng ได้ถูกติดตั้งไว้แล้วใน Debian แต่ในระบบปฏิบัติการอื่นนั้น ผู้ดูแลระบบจะต้องติดตั้งเองโดยการคอมไพล์จาก source ทั้งนี้จะต้องติดตั้ง libol ก่อนจึงจะสามารถติดตั้ง syslog-ng ได้

ผู้ดูแลระบบสามารถดาวน์โหลด libol และ syslog-ng จาก http://www.balabit.com/downloads/

หลังจากนั้นให้ขยายไฟล์ออกมา และทำการติดตั้งดังคำสั่งด้านล่างนี้

```
# cd libol-x.x
# ./configure; make; make install

# cd syslog-ng-x.x
# ./configure --sysconfdir=/etc; make; make install
```

คำสั่งด้านบนจะทำการติดตั้ง syslog-ng ไปไว้ที่ตำแหน่งโดยดีฟอลต์ (default location) คือ /usr/local หากต้องการติดตั้ง syslog-ng ไปยัง path อื่นให้ใช้คำสั่ง ./configure --prefix=/your/dir/

หลังจากการติดตั้งแล้ว ผู้ดูแลระบบจำเป็นต้องดำเนินการบางอย่างเพื่อให้ syslog-ng ทำงานได้ตามปกติ ดังนี้

- สร้างไดเรกทอรี /etc/syslog-ng
- สร้างไฟล์ configuration ของ syslog-ng (หรือคัดลอกมาจากไดเรกทอรี contrib/ และ doc/) ไว้ที่ /etc/syslog-ng/syslog-ng.conf
- สร้าง startup script ของ syslog-ng ไว้ที่ /etc/init.d/syslog-ng รวมทั้งสร้าง symbolic link จาก run level ต่างๆ เช่น /etc/rc2.d, /etc/rc3.d, /etc/rc5.d) ผู้ดูแลระบบสามารถคัดลอกตัวอย่าง startup script ของระบบปฏิบัติการที่ต้องการได้จากไดเรกทอรี contrib/

1.2 การใช้งาน syslog-ng

ผู้ดูแลระบบควรรัน syslog-ng ภายหลังการสร้างไฟล์ configuration เสร็จสิ้นแล้วเท่านั้น โดย syslog-ng มีออปชันในการรันค่อนข้างง่าย ดังตารางที่ 1

ตารางที่ 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)

1.3 Configuring Syslog-ng

Configuration ของ syslog-ng มีความยุ่งยากมากกว่าของ syslog แต่ก็ให้ประโยชน์ในแง่ของความยืดหยุ่นที่ได้และความสามารถที่มีมากกว่า หลังจากที่ทำความเข้าใจ configuration แล้ว ผู้ดูแลระบบสามารถสร้างไฟล์ configuration ง่ายๆ ขึ้นมาได้ด้วยตัวเอง และสามารถปรับปรุงให้เหมาะสมกับระบบของตนต่อไป

โดยปกติแล้ว syslog-ng จะอ่านข้อมูล configuration จากไฟล์ /etc/syslog-ng/syslog-ng.conf

ตัวอย่างที่ 1 แสดง configuration อย่างง่ายของ syslog-ng

```
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); };
```

จากตัวอย่างจะเห็นได้ว่า ส่วนประกอบหลักของ configuration ประกอบไปด้วย 5 statement หลัก คือ options{}, source{}, destination{}, filter{}, log{} ซึ่งแต่ละ statement จะคั่นด้วยเครื่องหมาย semicolon(;)

จะเห็นได้ว่ารูปแบบ configuration ของ syslog-ng.conf จะคล้ายคลึงกับรูปแบบของภาษาซี (C) ซึ่งทุกๆ statement จะต้องลงท้ายด้วยเครื่องหมาย semicolon ส่วน whitespace หรือช่องว่างนั้นไม่มีผลใดๆ ใน configuration จะใช้งานเพียงเพื่อให้สามารถอ่านได้ง่ายเท่านั้น

Global options

เป็นออปชันที่ถูกประกาศใช้งานภายใน options {} statement ซึ่งบางออปชันนั้นนอกจากสามารถใช้งานได้ใน option {} เองแล้วยังสามารถใช้งานใน statement อื่น เช่น source {}, destination {}, filter {}, log {} ได้อีกด้วย

ตารางที่ 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() นั้น สนใจเฉพาะค่า hostname ของเครื่องที่ส่งล็อกมาเท่านั้น ไม่เกี่ยวข้องกับ hostname ที่ระบุใน message body แต่อย่างใด

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 นั้นถูก resolve ให้เป็น joe-chong แต่ข้อมูลไอพีอื่นที่อยู่ใน message body คือ 10.0.0.222 นั้น ไม่ได้ถูก resolve ไปด้วย ดังนั้นจึงสรุปได้ว่าออปชัน use_dns(yes) นั้นจะทำการ resolve เฉพาะ hostname ที่อยู่ในส่วนต้นบรรทัดของ message เท่านั้น

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

หากมีการระบุค่าออปชันบางตัวที่ซ้ำกันใน options() section และ section อื่นๆ ค่าที่ระบุใน section อื่นๆ จะถูกนำไปใช้แทนที่ค่าใน options() section

keep_hostname() เป็นออปชันที่ใช้งานค่อนข้างมาก ซึ่งจะตั้งค่าดีฟอลต์เป็น no ซึ่งหมายถึง syslog-ng จะไม่ใช้ค่า hostname ที่ส่งมา มันจะทำการ resolve หา hostname จาก source IP address ของแพ็กเกตที่ส่งล็อกเข้ามา เพื่อป้องกันการปลอม hostname จากเครื่องที่ส่งล็อกเข้ามา ซึ่งจะแตกต่างจาก syslog ซึ่งใช้ค่า hostname ตามที่ได้รับมาจาก log message

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

ตัวอย่างที่ 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 DNS จากนั้นล็อกจึงจะถูกส่งต่อไปยัง host3 ต่อไป

ตัวอย่างที่ 2 แสดงตัวอย่างล็อกที่ถูกส่งต่อผ่านโฮสต์

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

สิ่งที่น่าสนใจจากตัวอย่างที่ 2 คือ

- เมื่อ host2 บันทึกข้อมูลล็อก ตัว syslog-ng ได้ตรวจสอบข้อมูลจาก DNS แล้วพบว่า จริงๆ แล้ว host1 นั้นมี DNS name เป็น linux แต่ syslog-ng เองก็ยังไม่

มั่นใจ จึงเพิ่ม hostname "linux" ต่อท้าย hostname "host1" (host1 อาจจะเป็นชื่อที่ปลอมมา)

- timestamp ที่ระบุในล็อกทั้งสามชุดมีเวลาที่ตรงกัน ซึ่งหมายถึง เวลาที่เห็นนั้นถูกสร้างขึ้นจากเครื่องที่ให้กำเนิดล็อกแล้วจึงส่งล็อกต่อไปเรื่อยๆ ผ่านโฮสต์ต่างๆ ซึ่งโฮสต์เหล่านั้นไม่ได้ตั้งค่า use_time_recd() ให้เป็น yes โฮสต์ต่างๆ จึงไม่ได้แก้ไขข้อมูล timestamp จึงมีผลให้เวลาทั้งสามจุดตรงกันหมด
- จากข้อมูลล็อกที่ host1 จะพบคำว่า s_loc อยู่ ซึ่งค่าดังกล่าวเป็นค่า source{} ของ syslog-ng ที่อยู่บน host1

ตัวอย่างที่ 3 แสดง configuration ของ syslog-ng บนเครื่อง 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); };
```

Sources

จากตัวอย่างที่ 3 มีการประกาศค่า source{} หนึ่งครั้ง โดยข้อมูลภายใน source{} ซึ่งก็คือ source driver ทำหน้าที่ระบุถึงแหล่งที่มาของข้อมูลล็อก ทั้งนี้ใน syslog-ng.conf หนึ่งๆ สามารถประกาศ source{} ได้ไม่จำกัดครั้ง ซึ่งภายใน source{} แต่ละตัวนั้นสามารถบรรจุ driver ได้ไม่จำกัดเช่นกัน

รูปแบบการประกาศ source{}

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

โดย sourcelabel หมายถึง string ที่ใช้เพื่ออ้างอิงกลุ่มของ source driver เพื่อให้สามารถนำไปใช้งานต่อได้อย่างสะดวก เช่น

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

จากบรรทัดด้านบน s_loc เป็นชื่อที่ถูกใช้เพื่ออ้างอิงถึงข้อมูลล็อกที่ถูกดึงมาจาก /dev/log และข้อมูลล็อกจาก syslog-ng เอง

syslog-ng มีความหยืดหยุ่นอย่างมากในการใช้งาน source driver ซึ่งสามารถรับข้อมูลล็อกได้จาก Unix socket เช่น /dev/log หรือล็อกจาก syslog-ng เอง รวมทั้งล็อกที่ส่งมาจากเครื่องอื่นผ่านทาง TCP, UDP protocol และยังสามารถรับล็อกจากไฟล์พิเศษเช่น ไฟล์ใน /proc ได้อีกด้วย ตารางที่ 3

ตารางที่ 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 ที่อยู่ในโหมด connection-oriented stream เช่น /dev/log (maximum concurrent connections default = 100)
unix-dgram("filename" [options])	ล็อกที่รับมาจาก Unix socket ที่อยู่ในโหมด 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)

internal()

syslog-ng เองจะส่งข้อมูลล็อก เช่น startup message, errors หรือล็อกอื่นๆ ไปยัง internal() ดังนั้นหากต้องการรับล็อกของตัวโปรแกรม syslog-ng จะต้องระบุ internal() ไว้ใน source{} ด้วย

file()

file() ใช้เพื่อระบุชื่อไฟล์ที่ต้องการให้ syslog-ng ไปดึงข้อมูลล็อกมา เช่น ไฟล์ /proc/kmsg ซึ่งเป็นไฟล์ข้อมูลล็อกของเคอร์เนลหากต้องการให้ syslog-ng ดึงข้อมูลล็อกจาก text file ปกติ เช่น ล็อกของ httpd นั้น จะต้องสร้างสคริปต์ขึ้นมาเพิ่มเติมเพื่อทำหน้าที่ pipe ผลลัพธ์ของคำสั่ง tail -f [filename] ไปยัง logger (ดูรายละเอียดเพิ่มเติมเกี่ยวกับการใช้งาน logger ได้จากคำสั่ง # man logger)

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(); };

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

tcp(),udp()

ทั้ง tcp() และ udp() จะรับข้อมูลล็อกจาก remote host ผ่านทาง TCP protocol (connection-oriented) และ UDP protocol (connectionless) โดยทั้งคู่สามารถตั้งให้รอรับข้อมูลล็อกผ่านทาง IP address และ port ที่ระบุได้ โดยดีฟอลต์แล้ว syslog-ng จะรอรับการเชื่อมต่อที่ 0.0.0.0:514 ซึ่งหมายถึง "รอรับการเชื่อมต่อที่ทุก network interface, port 514"

การระบุ IP address มีประโยชน์สำหรับโฮสต์ที่มี network interface มากกว่าหนึ่ง และต้องการเปิดพอร์ตรอรับล็อกจากบาง interface เท่านั้น ดังตัวอย่างที่ 4

ตัวอย่างที่ 4 ตัวอย่างการระบุ ip, port ใน source{}

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

จากตัวอย่างที่ 4 ซึ่งกำหนดให้ s_tcpmessages รับข้อมูลล็อกทุกอันที่ส่งมายัง network interface ที่มีไอพีเป็น 192.178.190.190 TCP port 10514 ส่วน s_udpmessages นั้นรอรับข้อมูลล็อกทุกอันผ่านทาง UDP port 514 ในทุกๆ local network interface

ip(), port(), max_connections()

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

ตัวอย่างที่ 5 ตัวอย่างการใช้งาน max-connections()

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

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

Destinations

Syslog-ng สามารถเก็บข้อมูลล็อกในรูปแบบเดียวกันกับที่ syslog เก็บได้ ไม่ว่าจะเป็น ASCII file, name pipe, remote host (ผ่านทาง UDP) และแสดงผลออกทาง TTY นอกจากนี้ syslog-ng ยังสามารถส่งข้อมูลล็อกไปยัง Unix socket, remote host (ผ่าน TCP) และส่งต่อไปยัง standard input ของโปรแกรมอื่น

ตารางที่ 5 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

syslog-ng สามารถเก็บข้อมูลลงไฟล์ได้และมีความสามารถมากกว่า syslog ตรงที่มีการใช้งานมาโคร มาโครช่วยให้สามารถตั้งชื่อไฟล์ที่ใช้เก็บข้อมูลล็อกได้อย่างน่าดี เช่น ตั้งชื่อไฟล์ตามปีเดือนวัน หรือตั้งชื่อไฟล์ตาม facility, priority

ตัวอย่างที่ 6 ตัวอย่างการใช้งาน

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

จากตัวอย่าง configuration ด้านบน เมื่อ syslog-ng ต้องการเขียนข้อมูลล็อกลงไฟล์ มันจะสร้างไฟล์ชื่อ /var/log/messages.Tues, /var/log/messages.Wed ซึ่งขึ้นกับวันที่เก็บข้อมูลล็อกดังกล่าว

ตารางที่ 6 Macros supported in file() destinations

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

syslog-ng จะสร้างไฟล์ขึ้นมาใหม่ หากไฟล์ที่ระบุใน file() ไม่มีอยู่จริง นอกจากนี้ syslog-ng ยังสามารถกำหนดออปชันบางตัวในระดับทั่วไป (general rule) คือให้มีผลกับ configuration ทั้งไฟล์ได้ ขณะเดียวกันก็สามารถกำหนดออปชันในระดับ per-log-file ได้ ซึ่งการกำหนดออปชันชนิดหลังนี้จะเป็นการ overridden ออปชันในระดับ general rule

ตัวอย่างที่ 7 การควบคุม file()

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

จากตัวอย่างที่ 7 เป็นการระบุออปชัน dir_owner(), dir_group(), dir_perm() ใน destination{} ซึ่งค่าที่ระบุนี้จะมีผลแทนที่ค่าที่ระบุใน options{} โดยอัตโนมัติ นอกจากนี้ยังสามารถระบุออปชัน owner(), group(), perm() ได้เช่นเดียวกันกับออปชันด้านบน

โดยปกติ syslog-ng จะสร้างไฟล์ล็อกที่ไม่มีอยู่ในระบบโดยอัตโนมัติ เว้นเสียแต่ว่าไฟล์ที่ระบุดังกล่าวจะอยู่ใน path ที่ไม่มีอยู่จริงและออปชัน create_dirs() ถูกตั้งค่าเป็น no

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

โดยดีฟอลต์แล้ว ค่าล็อกถูกตั้งค่าเป็นศูนย์ ซึ่งหมายถึงให้บันทึกข้อมูลล็อกทุกอันในทันที โดยปกติค่า sync() ต่ำๆ จะเหมาะสำหรับระบบที่ข้อมูลล็อกไม่เยอะมาก ส่วนระบบที่มีข้อมูลล็อกจำนวนมากควรใช้ค่า sync() สูง ซึ่งค่าระหว่าง 100 ถึง 1000 นั้นถือว่ามีค่าสูงพอสมควร ซึ่งผู้ดูแลระบบจะต้องทดสอบเพื่อหาค่าที่เหมาะสมกับระบบของตนต่อไป

อย่างไรก็ตามหากระบบที่ติดตั้ง syslog-ng ได้ติดตั้งโปรแกรมจำพวก log monitoring tool เช่น Swatch แล้วไม่ควรตั้งค่า sync() ไว้สูงมากนัก เพราะอาจจะทำให้ไม่สามารถแจ้งเตือนผู้ดูแลระบบได้ในกรณีที่ไฟล์ล็อกโดนลบ

Filters

filter หรือการกรองข้อมูลเป็นส่วนที่มีความสำคัญส่วนหนึ่ง นอกเหนือจากการกรองข้อมูลโดยใช้ facility, priority แล้ว syslog-ng ยังสามารถตรวจสอบชื่อโปรแกรมที่ส่งข้อมูลล็อกมา ชื่อเครื่องที่ทำหน้าที่ส่งต่อล็อกมา และยังสามารถกรองข้อมูลล็อกตาม regular expression ที่ตั้งไว้อีกด้วย

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

ตารางที่ 7 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

จากตัวอย่างที่ 8 แสดง syslog-ng.conf ในระบบปฏิบัติการลินุกซ์เดเบียน 2.2 (Debian 2.2)

ตัวอย่างที่ 8 ตัวอย่างการใช้งาน filter{}

```
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); };
```

บรรทัดแรกในตัวอย่างที่ 8 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

ยกเว้นข้อมูล ล็อกที่มี facility เป็น auth, authpriv, cron, daemon, mail, news
บรรทัดสุดท้าย filter f_cother กรองข้อมูลล็อกที่มี priority เป็น debug, info, notice และ warn **หรือ** ข้อมูล ล็อกที่มี facility เป็น daemin และ mail

Log statements

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

ตัวอย่างที่ 9 ตัวอย่าง syslog-ng.conf

```
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); };
```

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

จากตัวอย่างที่ 9 อาจจะเกิดข้อสังสัยว่า ล็อกบางส่วนที่ไม่ได้ถูกจัดเก็บโดย log{} statement ทั้งสามตัวนั้นจะถูกจัดเก็บไว้ที่ใด syslog-ng มีค่า filter(DEFAULT) ซึ่งสามารถใช้ระบุในตอนท้ายเพื่อสั่งให้ syslog-ng บันทึกข้อมูลล็อกที่ไม่ได้ถูกจัดเก็บโดย log{} ก่อนหน้านี้ได้ ดังตัวอย่างที่ 10

ตัวอย่างที่ 10 ตัวอย่าง syslog-ng.conf

```
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); };
```

Advanced Configurations

ตัวอย่างที่ 11 แสดงการใช้ syslog-ng เพื่อคอยเฝ้าดูข้อมูลล็อกที่ต้องการ (log monitoring)

ตัวอย่างที่ 11

```
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); };
```

ตัวอย่างที่ 12 เป็นตัวอย่าง script ที่ใช้สำหรับส่งอี-เมล์

ตัวอย่างที่ 12

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

จุดที่น่าสนใจในตัวอย่างที่ 11 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

ข้อควรระวังในการใช้งานดังตัวอย่างที่ 11 คือ การใช้ program() นั้นเป็นการเรียกใช้งานโปรแกรมที่ระบุ โดยโปรแกรมนั้นจะยังคงรันอยู่จนกว่า syslog-ng จะหยุดการทำงานหรือเริ่มการทำงานใหม่ ดังนั้นผู้ดูแลระบบควรไตร่ตรองก่อนการใช้งานออปชันดังกล่าว เช่น หากรัน bash process ก็จะทำให้เกิดการสิ้นเปลืองงานทรัพยากร นอกจากนี้หากรันโปรแกรมในฐานะ root ก็จะเป็นการเพิ่มความเสี่ยงให้กับระบบอีกด้วย นอกจากนี้การใช้ระบบเตือนภัยผ่านทางอี-เมล์ดังตัวอย่างที่ 11 ยังก่อให้เกิดความเสี่ยงที่ทำให้ระบบถูกโจมตีแบบ Denial of Service ได้ เช่น ทำให้ mailbox ของผู้ดูแลระบบเต็ม

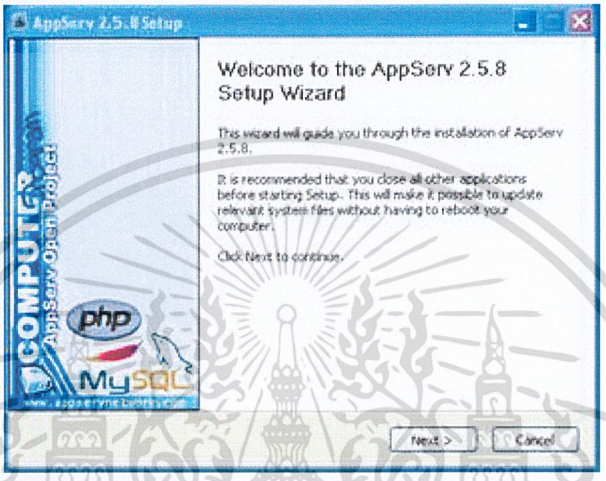
สรุป

syslog-ng เป็นโปรแกรมที่มีความหยืดหยุ่นในการทำงาน เหมาะสำหรับการนำมาใช้งานเป็นเป็น log server เป็นอย่างยิ่ง เพราะสามารถเก็บข้อมูลล็อกแยกตามเครื่องที่ส่งล็อกมาได้ นอกจากนี้ยังสามารถทำงานร่วมกับโปรแกรม sqlsyslogd เพื่อนำข้อมูลล็อกทั้งหมดบันทึกลงในฐานข้อมูลได้ ซึ่งจะนำเสนอรายละเอียดในโอกาสต่อไป

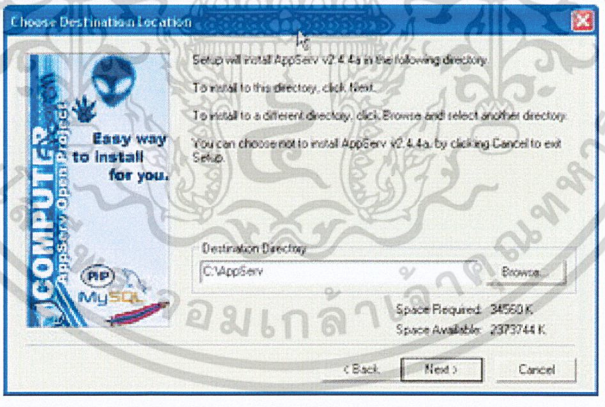
2. การติดตั้ง Appserv (PHP Apache Mysql)

1. ดาวน์โหลดโปรแกรม Appserv

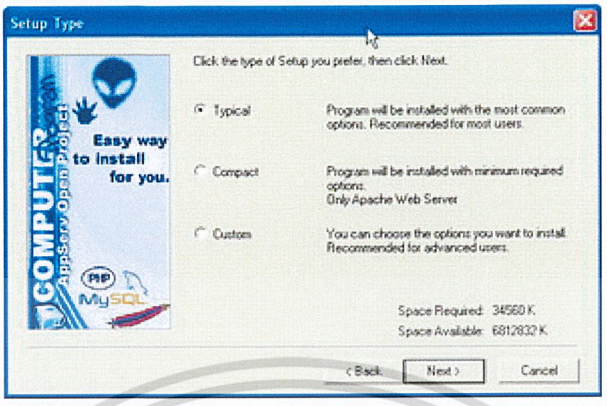
2. Double Click ที่โปรแกรม เพื่อทำการติดตั้ง จะปรากฏหน้าจอ



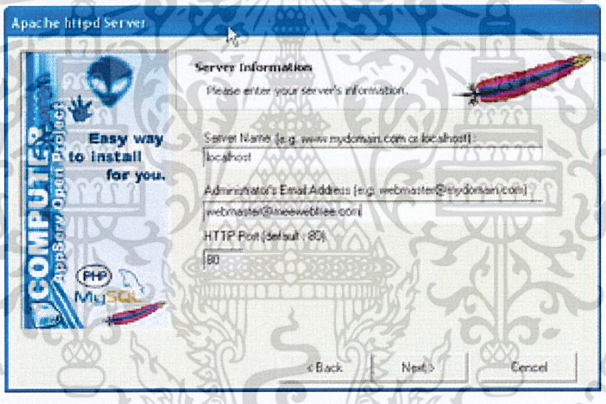
3. เลือก Drive ที่ต้องการติดตั้ง



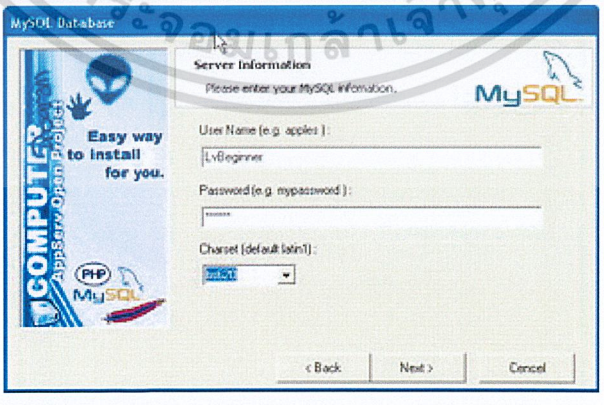
4. เลือก Typical



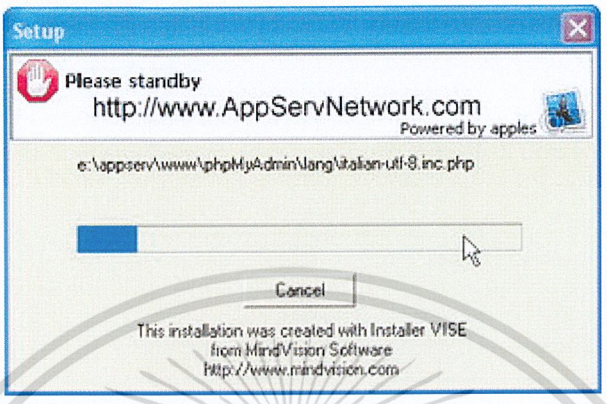
5. กรอกข้อมูล Server Name แนะนำว่า Default เป็น localhost



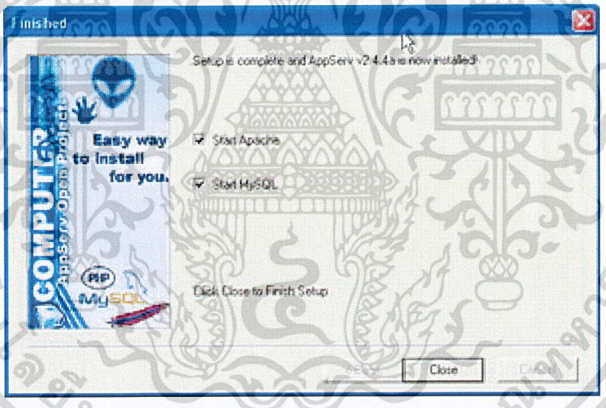
6. กำหนด Username และ Password ในการเข้าใช้งาน Mysql



7. โปรแกรมดำเนินการติดตั้ง



8. ดำเนินการเสร็จแล้วจะปรากฏดังหน้าจอ



9. เมื่อทำการติดตั้งแล้ว สามารถตรวจสอบได้คือ พิมพ์ URL http://localhost จะปรากฏหน้าจอดังนี้