บทที่ 4

การวิเคราะห์และออกแบบระบบ

4.1 บทนำ

ในบทนี้จะกล่าวถึงการออกแบบระบบที่ใช้สำหรับตรวจสอบการบุกรุกโดยการตรวจสอบข้อมูลจากล็อกไฟล์ที่ได้บันทึกไว้ โดยจะอธิบายถึงองค์ประกอบต่าง ๆ ของระบบ รวมทั้งแผนผังขั้นตอนการทำงาน หน้าที่ขององค์ประกอบส่วนต่าง ๆ ในระบบ แต่เนื่องจากระบบที่ออกแบบนี้เป็นระบบที่ทำงานบนระบบปฏิบัติการยูนิกซ์ และข้อมูลที่นำมาวิเคราะห์ก็ได้มาจากล็อกไฟล์ในระบบนี้เช่นกัน ดังนั้นก่อนที่จะอธิบายถึงการออกแบบระบบที่พัฒนาขึ้น จะกล่าวถึงการทำงานและรายละเอียดของล็อกไฟล์ต่าง ๆ บนระบบปฏิบัติการยูนิกซ์

4.2 ล็อกไฟล์บนระบบปฏิบัติการยูนิกซ์

ระบบปฏิบัติการยูนิกซ์ส่วนใหญ่สนับสนุนการตรวจสอบการทำงานของระบบสำหรับสนับสนุนความปลอดภัยของระบบคอมพิวเตอร์ในระดับ C2 ตามมาตราฐาน TCSEC ของศูนย์ความปลอดภัยคอมพิวเตอร์ กระทรวงกลาโหม สหรัฐอเมริกา [Department Of Defense Standard, 1985] โดยมีล็อกไฟล์สำหรับเก็บบันทึกว่ามีอะไรเกิดขึ้นในระบบคอมพิวเตอร์ ซึ่งระบบยูนิกซ์ในรุ่นแรก ๆ ใช้ล็อกไฟล์เพื่อบันทึกว่ามีใครเข้าใช้งานระบบ ออกจากระบบ และทำอะไรบ้าง ซึ่งต่อมายูนิกซ์ในรุ่นต่อมา ได้ขยายความสามารถในการเก็บบันทึกกิจกรรมและเหตุการณ์ต่าง ๆ มากขึ้น เช่น เก็บข้อมูลไฟล์ที่มีการถ่ายโอนกันใน เครือข่าย ความพยายามที่จะเป็น superuser และการรับส่งอีเมล์ เป็นต้น

ล็อกไฟล์ส่วนใหญ่จะเก็บในรูปแบบของ text file ที่เป็นการบันทึกข้อมูลทีละบรรทัดโดยโปรแกรมระบบหรือโปรแกรมประยุกต์ต่าง ๆ ในระบบคอมพิวเตอร์ เช่น แต่ละครั้งที่ผู้ใช้ปกติในระบบพยายามที่เปลี่ยนสิทธิเป็น superuser โดยใช้คำสั่ง su (substitute user) โปรแกรม su ก็จะบันทึกข้อมูลลงไปในล็อกไฟล์ที่ชื่อว่า sulog โดยจะบันทึกข้อมูลทั้งหมดไม่ว่าการกระทำนั้นสำเร็จหรือไม่ก็ตาม

ระบบยูนิกซ์แต่ละรุ่นจะมีการเก็บล็อกไฟล์ในไดเรคตอรี่ที่ต่างกัน แต่ส่วนใหญ่ไดเรคตอรี่ที่ใช้เก็บล็อกไฟล์จะเป็นดังนี้

/usr/adm ใช้ในระบบยูนิกซ์รุ่นแรก ๆ

/var/adm	ใช้ในระบบยูนิกซ์รุ่นที่ใหม่กว่าหรือในปัจจุบัน
/var/log	ใช้ในบางระบบของ Solaris, Linux, และ BSD สำหรับเก็บล็อกไฟล์ โดยที่ภายในไดเรคตอรี่ข้างต้น ก็อาจจะมีล็อกไฟล์เหล่านี้อยู่ เช่น
acct หรือ pacct	บันทึกคำสั่งที่สั่งโดยผู้ใช้แต่ละคน
aculog	บันทึกการหมุนโมเด็มออก
lastlog	บันทึกการติดต่อเข้าระบบครั้งสุดท้ายทั้งที่สำเร็จและไม่สำเร็จ
loginlog	บันทึกการพยายามในการติดต่อที่ไม่สำเร็จ
messages	บันทึกข้อความที่แสดงทางจอภาพของระบบ (console) และข้อความอื่น ๆ ที่ได้รับจาก syslog
sulog	บันทึกการเรียกใช้คำสั่ง su
utmp	บันทึกข้อมูลของผู้ใช้ที่กำลังใช้งานระบบอยู่
utmpx	ส่วนขยายเพิ่มเติมจาก utmp
wtmp	บันทึกข้อมูลการเข้าและออกจากระบบของผู้ใช้แต่ละคน รวมถึงการเปิด (startup) และปิด (shutdown) ระบบด้วย
wtmpx	ส่วนขยายเพิ่มเติมจาก wtmp
vold.log	บันทึกข้อผิดพลาดต่าง ๆที่เกิดขึ้นเมื่อมีการใช้สื่อบันทึกข้อมูลภายนอก เช่น แผ่นดิสก์ หรือ CD-ROM เป็นต้น
xferlog	บันทึกข้อมูลการใช้งานของโปรแกรม FTP (File Transfer Protocol)
access_log	บันทึกการทำงานของโปรแกรม Apache ซึ่งเป็นโปรแกรม Web Server

นอกจากล็อกไฟล์ดังกล่าว ยังมีล็อกไฟล์ที่เกิดขึ้นจากโปรแกรมประยุกต์อื่น ๆ อีก เช่น ล็อกไฟล์ที่เกิดจากโปรแกรม sendmail ซึ่งทำหน้าที่เป็น mail server ล็อกไฟล์ที่เกิดจากโปรแกรมประเภทฐานข้อมูลและอื่น ๆ เป็นต้น ซึ่งโปรแกรมเหล่านี้จะสร้างล็อกไฟล์ที่แสดงการทำงานของโปรแกรมและข้อความต่าง ๆ ที่บ่งบอกถึงการทำงานและปัญหาต่าง ๆ ที่เกิดขึ้นในระหว่างการทำงานเพื่ออำนวยความสะดวกในการตรวจสอบ

4.2.1 โปรแกรม syslog

นอกจากล็อกไฟล์ต่าง ๆดังกล่าวข้างต้นแล้ว ในระบบยูนิกซ์ยังมีโปรแกรมที่ทำหน้าที่เป็นศูนย์กลางในการจัดการเก็บล็อกต่าง ๆ ในระบบซึ่งจะเรียกว่า syslog ซึ่งคิดค้นขึ้นโดย Computer Science Research Group (CSRG) ที่ University of California at Berkeley (UC-Berkeley) [Vixie and Avolio, 2001] โดย syslog ได้ถูกออกแบบให้มีความสามารถในการรายงานเหตุการณ์ต่าง ๆ ของระบบ ซึ่งข้อความจากเหตุการณ์ทั้งหมดจะถูกรวบรวมโดย

โปรแกรมที่ทำหน้าที่ในการจัดเก็บข้อมูลลงในล็อกไฟล์ต่างๆ ที่ชื่อว่า syslogd แล้วจึงบันทึกข้อความเหล่านั้นลงไฟล์ที่กำหนด โดยที่ล็อกไฟล์เหล่านั้นอาจจะเก็บอยู่ในเครื่องที่ใช้งานอยู่หรือเครื่องคอมพิวเตอร์อื่นในเครือข่ายก็ได้ ซึ่งไฟล์ที่ใช้ในการกำหนดการทำงาน (configuration file) โดยทั่วไปจะเก็บอยู่ที่ /etc/syslog.conf โดยใช้กำหนดว่าข้อมูลล็อกที่ส่งมาให้ syslogd แต่ละประเภทนั้นให้เก็บไว้ที่ ล็อกไฟล์ใด ซึ่งรายละเอียดการทำงานของ syslogd และไฟล์ syslog.conf มีดังนี้

- **syslogd**

syslogd เป็นโปรแกรมที่ทำหน้าที่ในการอ่านและส่งต่อข้อมูลล็อกที่เกิดจากโปรแกรมระบบและโปรแกรมประยุกต์บนระบบยูนิกซ์ไปยังล็อกไฟล์ต่างๆ โดยที่ผู้ดูแลระบบสามารถกำหนดค่าการทำงานของ syslogd ได้จากไฟล์ syslog.conf เช่น ต้องการให้ syslogd เก็บข้อมูลล็อกที่ไฟล์ใด หรือให้ส่งข้อมูลล็อกไปเก็บไว้ยังเครื่องใดในเครือข่าย

ข้อมูลล็อกที่ควบคุมโดย syslogd นั้น จะถูกกำหนดให้มีค่า facility และ priority โดยค่าของ facility จะใช้บ่งบอกหรืออธิบายถึงแหล่งที่มาของข้อมูลล็อกนั้นๆ เช่น ข้อมูลล็อกที่ส่งมาจากระบบเมล์ ก็จะมี facility เป็น mail ส่วนค่า priority จะแสดงถึงระดับความสำคัญของเหตุการณ์ที่เกิดในแต่ละ facility ทั้งนี้ข้อมูลล็อกทุกข้อความจำเป็นต้องมี facility และ priority เสมอ สำหรับรายละเอียดของ facility และ priority ต่างๆ ของ syslog แสดงได้ดังตาราง 4.1 และ 4.2 ตามลำดับ

ตาราง 4.1 แสดงรายละเอียด facility ต่างๆ ของ syslog

ชื่อ facility	ชื่อระบบ/โปรแกรม
kern	Kernel
user	Regular user processes
mail	Mail system
daemon	System daemons
auth	Authorization system
syslog	Internal syslogd messages
lpr	Line printer system
news	Usenet News system
uucp[1]	unix-to-unix copy program (uucp)
sys9-sys14	Reserved for system (defined only on Cisco router)

[1] อาจจะไม่ค่อยมีการใช้งานในปัจจุบัน แต่มีไว้ในตารางนี้เพื่อความสมบูรณ์ของเนื้อหา

ตาราง 4.1 (ต่อ)

ชื่อ facility	ชื่อระบบ/โปรแกรม
cron	Cron daemon
local0 – local7	Reserved for local use

ตาราง 4.2 แสดง priority ต่างๆ ของ syslog (เรียงค่า priority จากมากไปหาน้อย)

ชื่อ priority	ระดับของเหตุการณ์
emerg	panic situations
alert	urgent, less than panic, situations
crit	critical conditions
err	error conditions
warning	warning (non-error) messages
notice	unusual conditions
info	informational messages
debug	debugging messages

ข้อมูลในตาราง 4.1 ประกอบด้วยชื่อ facility และชื่อระบบ/โปรแกรมประยุกต์ต่างๆ ที่จะส่งข้อมูลโดยใช้ค่า facility เหล่านั้น ส่วนตารางที่ 4.2 ประกอบด้วย ชื่อ priority และระดับของเหตุการณ์ที่ใช้สำหรับการกำหนดในการบันทึกข้อมูลลงในล็อกไฟล์

สำหรับการทำงานของ syslog นั้น จะเริ่มต้นจากโปรแกรม syslogd อ่านข้อกำหนดการทำงานจากไฟล์ syslog.conf เพื่อที่จะดูว่าเหตุการณ์แบบใดบ้างที่จะบันทึกลงล็อกไฟล์และ ล็อกไฟล์เหล่านั้นเก็บอยู่ที่ใด จากนั้นโปรแกรม syslogd ก็คอยดักจับข้อความจากสองแหล่งดังนี้

1. /dev/log หรือ /var/run/log
   เป็น socket ใช้สำหรับรับข้อมูลล็อกที่สร้างโดยโปรแกรมต่างๆ ที่กำลังทำงานอยู่บนระบบ
2. UDP port 514
   เป็น Internet domain socket ใช้สำหรับเป็นช่องทางสื่อสารสำหรับการรับข้อมูลล็อกที่ส่งมาจากเครื่องอื่นๆ ในเครือข่าย

- **syslog.conf**

การทำงานของ syslogd นั้น จะขึ้นอยู่กับการกำหนดค่าต่างๆในไฟล์ syslog.conf เป็นหลัก การแก้ไขใดๆ ที่เกิดขึ้นกับไฟล์นี้นั้นจะยังไม่มีผลต่อการทำงานของ syslogd ในทันที จะต้องทำการสั่งให้โปรแกรม syslogd เริ่มต้นการทำงานใหม่เสียก่อน รูปแบบการกำหนดค่าในไฟล์ syslog.conf มีดังนี้

Facility.priority	*Action*
Facility1, facility2. priority	*Action*
Facility1. Priority1; facility2. priority2	*Action*
**. priority*	*Action*
**. priority;badfacility.none*	*Action*

หมายเหตุ : ระหว่างส่วนของ facility.priority กับ action ใช้ Tab 2 ตัวในการคั่น ไม่ใช่ช่องว่าง (blank space)

ความหมายของรูปแบบในไฟล์นี้คือ เมื่อมีข้อมูลล็อกที่มี facility และ priority (ตามที่ได้กล่าวไว้ในตารางที่ 4.1 และ 4.2) ที่มากกว่าหรือเท่ากับค่าที่ระบุไว้ ก็จะกระทำ action ตามที่กำหนดไว้ และถ้ากำหนดค่า priority เป็น debug ซึ่งเป็นค่า priority ต่ำสุด ก็จะคลอบคุลมทุก priority ของ facility นั้นๆ ทั้งนี้สามารถใช้เครื่องหมาย * แทนทุกๆ ค่าใน facility หรือ priority นั้นๆ ได้ เช่น

mail. /var/log/mail*

ซึ่งหมายถึงกำหนดให้ syslogd เก็บข้อมูลล็อกของ mail ทุก priority ในไฟล์ /var/log/mail

แต่ถ้าค่า priority เป็น none หมายความว่าไม่ให้สนใจ facility ที่ประกาศค่า priority เป็น none

**.emerg;mail.none /var/log/emer.log*

คือให้เก็บข้อมูลล็อกที่มี priority เป็น emerg สำหรับทุก facility ยกเว้น mail.emerg สำหรับ action นั้นสามารถเลือกใช้ได้ดังนี้คือ

- filename : เก็บข้อมูลล็อกนั้นลงในไฟล์ที่กำหนด
- @hostname : ส่งข้อมูลล็อกไปยัง syslogd บน host ที่กำหนด
- @IP_address : ส่งข้อมูลล็อกไปยัง host ที่มี IP address ตามที่กำหนด
- user1, user2 : ส่งข้อมูลล็อกไปยังหน้าจอของผู้ใช้ที่กำหนด ถ้าผู้ใช้เหล่านั้นอยู่ในระบบ
- * : ส่งข้อมูลล็อกไปยังทุกๆ user ที่อยู่ในระบบ
- /dev/console เพื่อส่งข้อมูลล็อกไปยัง console device หรือ device อื่นๆ ตามที่ต้องการ

โดยปกติจะเก็บข้อมูลล็อกไว้ในไฟล์ซึ่งอยู่ในไดเรคตอรี่ /var/log หรือ /var/adm

สิ่งที่ควรระวังเป็นอย่างยิ่งสำหรับการเก็บรวบรวมล็อกจากหลาย ๆ เครื่องก็คือในกรณีที่เวลาของแต่ละเครื่องไม่ตรงกันนั้นอาจจะก่อให้เกิดความยุ่งยากในการวิเคราะห์ล็อก เพราะในการบันทึกกิจกรรมต่าง ๆ ต้องมีการบันทึกเวลาด้วยและเวลาที่บันทึกลงในล็อกไฟล์นั้นเป็นเวลาของเครื่องที่ทำหน้าที่บันทึกข้อมูลล็อก ไม่ได้ใช้เวลาของเครื่องที่ส่งข้อมูลล็อกมาให้ ดังนั้นจึงควรตั้งเวลาของทุก ๆ เครื่องที่มีการส่งและบันทึกข้อมูลล็อกให้ถูกต้องและตรงกัน

สำหรับการทดสอบการทำงานของ syslog ว่าทำงานหรือไม่นั้น สามารถทำได้โดยการใช้ คำสั่ง logger เช่น
เพิ่มบรรทัดด้านล่างนี้ในไฟล์ syslog.conf

local5.warning */var/log/test.log*

จากนั้นให้ restart syslogd ใหม่ แล้วจึงใช้คำสั่ง

#logger -p local5.warning "test"

ซึ่งถ้าเป็นไปตามปกติแล้ว ในไฟล์ /var/log/test.log ก็จะมีคำว่า test ปรากฏอยู่ด้วย

สำหรับข้อความล็อกต่าง ๆ ที่เก็บในล็อกไฟล์นั้นจะเก็บในลักษณะของไฟล์ข้อความที่มีรูปแบบของข้อความประกอบด้วย 4 ส่วน ดังตัวอย่าง

```
Feb 19 15:11:00 ratree imapd[19437]: imap service init from 192.168.100.2
Feb 19 15:11:01 ratree ipop3d[19443]: connect from 192.168.37.81
|<------(1)------ ->|--(2)--|<------(3)------>|<---------------------(4)---------------------->|
```

โดยส่วนที่

(1) คือ วันเวลาที่บันทึกข้อมูลล็อก(ไม่บันทึกข้อมูล ปี) ในที่นี้ คือ Feb 19 15:11:01
(2) คือ ชื่อหรือหมายเลข IP address ของระบบ เครื่องคอมพิวเตอร์ อุปกรณ์ ที่เป็นแหล่งข้อมูลของข้อความ คือ ratree
(3) คือ ชื่อและหมายเลข process ที่ทำงานบนระบบ เครื่องคอมพิวเตอร์ อุปกรณ์ ที่ส่งข้อความ เช่น ipop3d[19443]
(4) คือ ข้อความที่ส่งมายัง syslog เช่น connect from 192.168.37.81

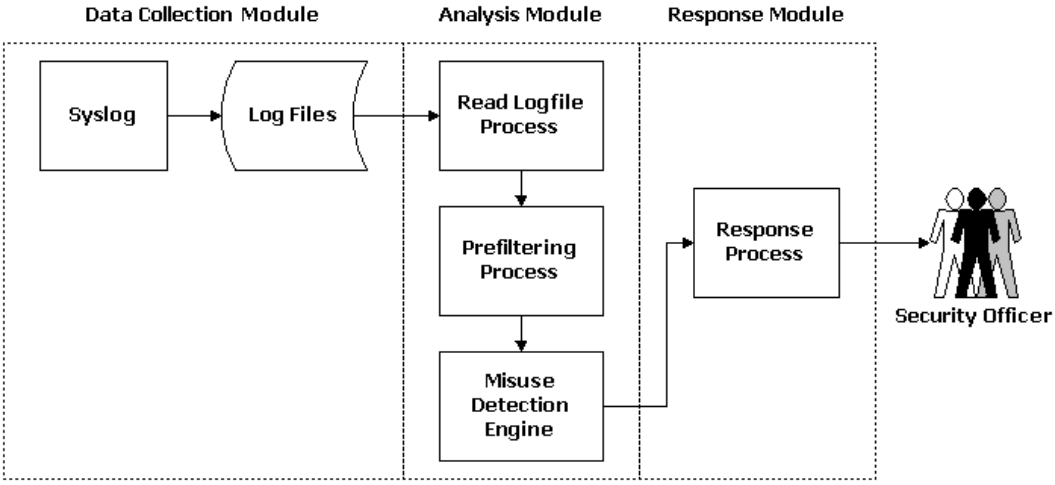
ถึงแม้ว่า syslog จะเป็นโปรโตคอลมาตรฐานสำหรับการจัดการข้อความล็อกในระบบที่มีพื้นฐานจากระบบยูนิกซ์ (สำหรับรายละเอียดของโปรโตคอล syslog อ่านได้จาก RFC 3164) แต่อย่างไรก็ตามโปรโตคอล syslog นั้นมีปัญหาที่สำคัญหลายประการ คือ จำกัดขนาดของข้อความที่ส่งไว้เพียง 1024 ไบต์ ไม่มีกระบวนการตรวจสอบความบูรณภาพ (Integrity) และการยืนยันตัวตน (Authenticity) โดยเฉพาะอย่างยิ่งในกรณีที่มีการส่งข้อมูลล็อกไปยังเครื่องที่ทำหน้าที่เก็บข้อมูลล็อกโดยเฉพาะในเครือข่ายนั้น syslog จะทำงานโดยใช้ UDP port 514 ซึ่งเป็นโปรโตคอลที่ไม่มีการรับประกันการส่งข้อมูลและยังสามารถปลอมแปลงข้อมูลที่ส่งได้โดยง่าย ดังนั้นในการปฏิบัติจริงผู้ดูแลระบบควรติดตั้งไฟร์วอลล์เพื่อป้องกันไม่ให้เครื่องจากภายนอกส่งข้อมูลล็อกมายังเครื่องที่ทำหน้าที่เก็บข้อมูลล็อกดังกล่าว นอกจากนี้ในการส่งข้อมูลโดยใช้ syslog นั้นจะส่งข้อมูลโดยที่ไม่มีการเข้ารหัสข้อมูลก่อนส่ง ถ้าหากมีการดักจับข้อมูลระหว่างทางที่ส่ง ผู้บุกรุกก็จะสามารถดูข้อมูลล็อกเหล่านั้นได้ทันที

ดังนั้นจึงได้มีความพยายามในการปรับปรุง syslog โดยการเพิ่มความสามารถในเรื่องการตรวจสอบความถูกต้อง การยืนยันตัวตน และความลับของข้อมูลในระหว่างที่ส่งเพื่อทำให้ข้อมูล ล็อกมีความน่าเชื่อถือและปลอดภัยยิ่งขึ้น และเมื่อนำข้อมูลเหล่านั้นมาใช้ประโยชน์ เช่น วิเคราะห์เพื่อตรวจจับการบุกรุก ก็จะทำให้ผลลัพธ์ที่ได้ก็จะน่าเชื่อถือและถูกต้องด้วยเช่นกัน ตัวอย่างการปรับปรุงนี้ ได้แก่

- Modular Syslog [Sanchez, 2001] ได้ปรับปรุงในเรื่องของการตรวจสอบความถูกต้องของข้อมูล โดยใช้อัลกอริทึม PEO-1 และ L-PEO เพิ่มความสามารถในการส่งข้อมูลล็อกไปเก็บลงในฐานข้อมูล MySQL หรือ PostgreSQL และการส่งข้อมูลและกรองข้อมูลล็อกไปยังไฟล์ต่างโดยใช้ regular expression
- Nsyslogd [Reed, 1999] พัฒนาโดย Darren Reed ซึ่งสนับสนุนการส่งข้อมูลล็อกโดยใช้ TCP แทน UDP และเพิ่มความปลอดภัยของข้อมูลระหว่างการส่งโดยใช้การเข้ารหัสข้อมูลด้วย SSL (Secure Socket Layer)
- syslog-ng (syslog new generation) [Scheidler, 2000] พัฒนาโดยบริษัท BalaBit IT มีความสามารถในการส่งข้อมูลล็อกผ่าน UDP หรือ TCP ก็ได้ สามารถส่งข้อมูลล็อกไปยังฐานข้อมูล สามารถแยกเก็บข้อมูลล็อกหรือกรองข้อมูลโดยใช้ regular expression สามารถทำงานในรูปแบบที่อ้างอิง priority/facility ได้ ดังนั้น syslog-ng จึงสามารถทำงานแทนที่ syslog ได้ นอกจากนี้ ยังสนับสนุน log forwarding ซึ่งทำให้สามารถทราบได้ว่า ต้นทางของล็อกถูกส่งมาจากเครื่องใดและมีการส่งผ่านเครื่องใดมาบ้าง

4.3 องค์ประกอบของระบบตรวจจับการบุกรุก

ระบบตรวจจับการบุกรุกที่ออกแบบและพัฒนาขึ้นนี้เป็นระบบที่ใช้ข้อมูลบันทึกการทำงานที่เก็บอยู่ล็อกในไฟล์มาวิเคราะห์ เพื่อค้นหาเหตุการณ์ความผิดปกติหรือการบุกรุกที่เกิดขึ้น ดังนั้นจึงถือได้ว่าระบบดังกล่าวเป็นระบบตรวจจับการบุกรุกประเภท Host-Based Intrusion Detection System (HIDS) ซึ่งภาพรวมของระบบนี้จะประกอบส่วนสำคัญ 3 ส่วนหลัก ดังภาพประกอบที่ 4.1



ภาพประกอบ 4.1 องค์ประกอบของระบบตรวจจับการบุกรุกที่พัฒนา

1. Data Collection Module ทำหน้าที่ในการเก็บรวบรวมข้อมูลล็อกต่าง ๆ ที่เกิดขึ้นในระบบแล้วบันทึกลงในล็อกไฟล์ต่าง ๆที่กำหนดไว้ โดยในที่นี้จะอาศัยการทำงานของโปรแกรม syslog ซึ่งเป็นโปรแกรมที่มีอยู่แล้วในระบบปฏิบัติการยูนิกซ์
2. Analysis Module เป็นโปรแกรมที่พัฒนาขึ้นซึ่งทำหน้าที่คอยอ่านข้อมูลล็อกที่มีการบันทึกเพิ่มเติมจากล็อกไฟล์แล้วนำมาวิเคราะห์เพื่อตรวจสอบหาเหตุการณ์ความผิดปกติหรือเหตุการณ์การบุกรุก โดยสามารถแบ่งการทำงานออกได้เป็น 3 ส่วนย่อย คือ Read Logfile Process, Prefiltering Process และ Misuse Detection Engine
3. Response Module ประกอบด้วยโปรแกรมส่วนที่เรียกว่า Response Process ซึ่งทำหน้าที่ในการแจ้งเตือนไปผู้ดูแลระบบเมื่อผลของการวิเคราะห์ข้อมูลพบว่ามีเหตุการณ์การบุกรุกเกิดขึ้น

ซึ่งรายละเอียดการทำงานของแต่ละส่วนจะกล่าวในหัวข้อที่ 4.3.1, 4.3.2 และ 4.3.3 ตามลำดับ

4.3.1 Data Collection Module

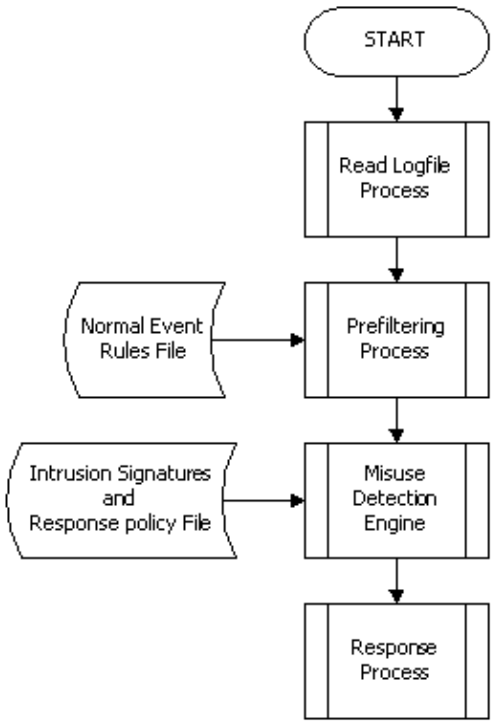
Data Collection Module เป็นส่วนที่ทำหน้าที่ในการเก็บรวบรวมข้อมูลบันทึกการทำงานต่างๆ ลงในล็อกไฟล์เพื่อที่จะใช้สำหรับการวิเคราะห์ข้อมูล ซึ่งการทำวิทยานิพนธ์นี้เป็นการศึกษาและวิเคราะห์ข้อมูลล็อกไฟล์ต่างๆ ที่มีอยู่และใช้งานบนระบบปฏิบัติการยูนิกซ์ ดังนั้นในการเก็บรวบรวมข้อมูลจึงใช้กลไกของโปรแกรม syslog ในการทำงาน โดยมีการกำหนดค่าการทำงานของ syslog ผ่านไฟล์ syslog.conf สำหรับการเก็บข้อมูลล็อกต่างๆ นั้นได้กำหนดให้เก็บข้อมูลล็อกทั้งหมดที่ต้องการให้บันทึกรวมอยู่ในไฟล์เดียวกัน เพราะทำให้ง่ายและสะดวกสำหรับการวิเคราะห์ข้อมูลในส่วนของหัวข้อที่ 4.3.2

4.3.2 Analysis Module

Analysis Module เป็นส่วนที่ทำหน้าที่คอยอ่านข้อมูลล็อกเฉพาะส่วนที่มีการบันทึกเพิ่มเติมจากล็อกไฟล์แล้วนำมาวิเคราะห์เพื่อตรวจสอบหาเหตุการณ์ความผิดปกติหรือเหตุการณ์การบุกรุก โดยในการออกแบบการทำงานของส่วนนี้จะประกอบด้วยการทำงาน 3 ส่วนย่อย คือ

1. Read Logfile Process มีหน้าที่คอยอ่านข้อมูลจากล็อกไฟล์โดยอ่านข้อมูลเฉพาะส่วนที่มีการบันทึกเพิ่มเติม แล้วส่งข้อมูลที่อ่านได้ไปวิเคราะห์ที่ Prefiltering Process ต่อไป
2. Prefiltering Process ทำหน้าที่กรองเอาข้อมูลล็อกของเหตุการณ์ปกติที่ไม่เกี่ยวข้องกับการรักษาความปลอดภัยออกไป โดยการกรองจะอาศัยกฎที่เก็บอยู่ในไฟล์กฎของเหตุการณ์ปกติ (Normal Event Rule) ข้อมูลล็อกที่ผ่านการกรองแล้วจะถือว่าเป็นล็อกของเหตุการณ์ที่ผิดปกติ ซึ่งก็จะถูกส่งไปวิเคราะห์ที่ Misuse Detection Engine ต่อไป รายละเอียดของกฎที่ใช้กรองจะได้กล่าวในหัวข้อ 5.8.1
3. Misuse Detection Engine เป็นส่วนที่ทำหน้าที่ในการบ่งชี้ว่าเหตุการณ์ความผิดปกติที่ส่งมานั้นเป็นการบุกรุกรูปแบบใด โดยใช้วิธีการเปรียบเทียบรูปแบบล็อกของเหตุการณ์ที่เกิดขึ้นกับรูปแบบร่องรอยของการบุกรุกแบบต่างๆ ที่กำหนดไว้เป็นกฎซึ่งเก็บอยู่ในไฟล์ Intrusion Signature and Response Policy และหากพบว่ามีรูปแบบที่ตรงกันก็จะทำการแจ้งเตือนตามวิธีการที่ระบุไว้ในกฎข้อดังกล่าว แต่หากรูปแบบล็อกที่ส่งมาไม่ตรงกับกฎข้อใดเลย ก็จะบันทึกข้อมูลล็อกนั้นลงในไฟล์ที่ใช้สำหรับเก็บล็อกเหตุการณ์ความผิดปกติที่ไม่สามารถระบุได้ (Unknown Message) เพื่อที่จะให้ผู้ดูแลระบบสามารถทำการตรวจสอบล็อกดังกล่าวได้ในภายหลัง

องค์ประกอบและขั้นตอนการทำงานของส่วน Analysis Module แสดงเป็นแผนภาพการทำงานได้ดังภาพประกอบ 4.2



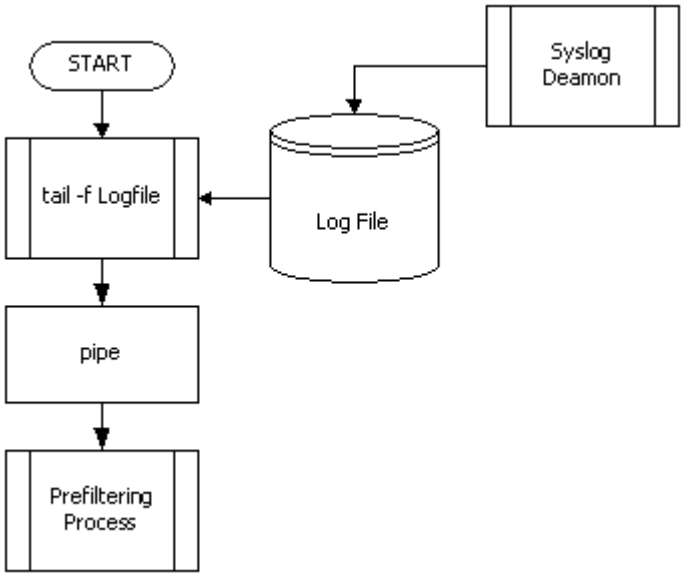
ภาพประกอบ 4.2 องค์ประกอบและขั้นตอนการทำงานของส่วน Analysis Module

การทำงานของโปรแกรมส่วน Analysis Module จะทำในลักษณะเกือบทันทีทันใด (Near Real Time) สำหรับรายละเอียดขั้นตอนการทำงานในส่วน Read Logfile Process, Prefiltering Process และ Misuse Detection Engine จะกล่าวถึงในหัวข้อ 4.3.2.1 4.3.2.2 และ 4.3.2.3 ตามลำดับ

4.3.2.1 Read Logfile Process

โปรแกรมส่วนที่ทำหน้าที่คอยอ่านข้อมูลล็อกจากล็อกไฟล์นั้นจะอาศัยการทำงานของโปรแกรม tail ซึ่งเป็นโปรแกรมมาตราฐานที่มีใช้งานอยู่บนระบบปฏิบัติการยูนิกซ์ โดยโปรแกรมนี้จะที่ทำหน้าที่อ่านข้อมูลจากล็อกไฟล์ที่ต้องการวิเคราะห์ แล้วส่งผลลัพธ์ที่ได้ต่อไปยังส่วน Prefiltring Process โดยใช้ pipe สำหรับการใช้โปรแกรม tail นั้นจะระบุตัวเลือกการทำงาน

เป็น –f ซึ่งหมายความว่าให้อ่านเฉพาะข้อมูลส่วนที่มีการบันทึกเพิ่มเติมเท่านั้น โดยขั้นตอนการทำงานของส่วนนี้ แสดงได้ดังภาพประกอบ 4.3



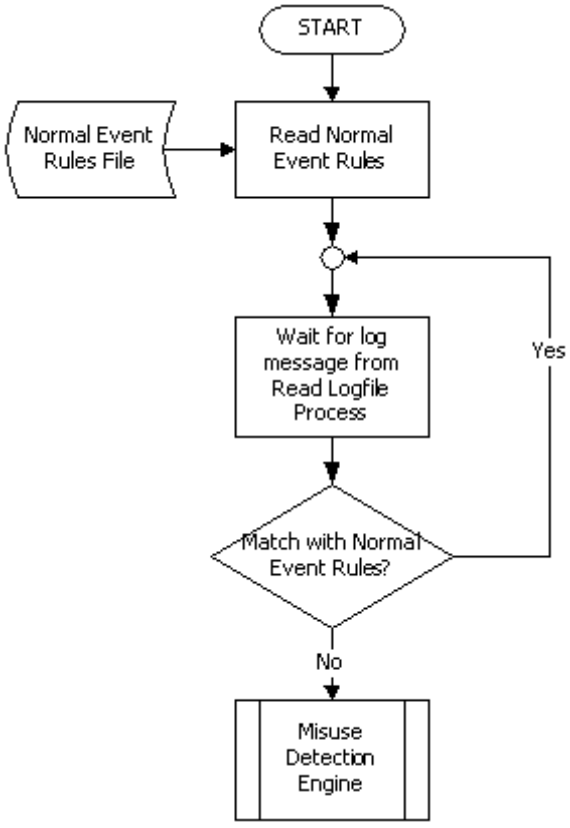
ภาพประกอบ 4.3 ขั้นตอนการทำงานของ Read Logfile Process

4.3.2.2 Prefiltering Process

ในขณะที่ระบบคอมพิวเตอร์กำลังการทำงานอยู่นั้น โปรแกรม syslogd จะคอยเก็บรวบรวมข้อมูลล็อกการทำงานของโปรแกรมและเหตุการณ์ต่าง ๆ ที่เกิดขึ้นในระบบ พร้อมกับส่งข้อมูลเหล่านี้ไปบันทึกลงในล็อกไฟล์ ตามที่กำหนดไว้ในไฟล์ /etc/syslog.conf ข้อมูลล็อกเหล่านี้จะเป็นล็อกของทุกเหตุการณ์ที่เกิดขึ้นนั้นในระบบ โดยอาจจะเป็นของเหตุการณ์การทำงานปกติ ข้อผิดพลาดต่าง ๆ และรวมทั้งเหตุการณ์การบุกรุกด้วย ทำให้ข้อมูลล็อกที่บันทึกลงในล็อกไฟล์มีปริมาณมาก และจะส่งผลให้การวิเคราะห์ข้อมูลเพื่อตรวจจับการบุกรุกนั้นทำได้ยากขึ้น ดังนั้นในการออกแบบโปรแกรมจึงได้ออกแบบให้มีส่วน Prefiltering Process เพื่อทำหน้าที่ในการกรองข้อมูลล็อกที่เป็นของเหตุการณ์ปกติหรือเหตุการณ์ที่ไม่เกี่ยวข้องออกไปก่อน ทั้งนี้เพื่อให้มีปริมาณข้อมูลที่จะต้องวิเคราะห์เพื่อตรวจจับการบุกรุกมีปริมาณที่น้อยลง โดยการออกแบบและพัฒนาโปรแกรมในส่วนนี้ใช้วิธีการที่เรียกว่า artificial ignorance ที่นำเสนอโดย Marcus Ranum [Rannum, 1997] โปรแกรมส่วนนี้จะมีไฟล์ที่เก็บกฎต่าง ๆ ที่ใช้ในการกรองเอาข้อมูลล็อกของเหตุการณ์ปกติหรือไม่มีประโยชน์ออกไป กฎเหล่านี้จะเรียกว่า Normal Event Rule ซึ่งใช้สำหรับการกำหนดรูปแบบล็อกของเหตุการณ์การทำงานปกติ โดยที่ผู้ใช้สามารถเขียนกฎเหล่านี้ในรูปแบบของ Regular Expression ที่อธิบายถึงรูปแบบข้อมูลล็อกของเหตุการณ์ปกติ

ข้อมูลที่สำคัญของกระบวนการนี้คือกฎของเหตุการณ์ปกติ การสร้างกฎเหล่านี้ต้องใช้ข้อมูล ล็อกของการทำงานในสภาวะปกติของระบบ ซึ่งการเก็บรวบรวมข้อมูลล็อกดังกล่าวจึงทำใน เครือข่ายที่จัดทำขึ้นสำหรับการทดสอบและสั่งให้โปรแกรมต่าง ๆ ที่ต้องการทำงานตามปกติ แล้วจึงเก็บรวบรวมข้อมูลล็อกที่ได้มาจำแนกตามเหตุการณ์ และนำมาเขียนเป็นกฎ โดยกฎแต่ละข้อใช้อธิบายถึงรูปแบบข้อมูลล็อคที่เกิดขึ้นของเหตุการณ์หนึ่งเหตุการณ์

สำหรับการทำงานของโปรแกรมส่วนนี้เริ่มต้นจากการอ่านข้อมูลกฎการกรองต่าง ๆ จากไฟล์กฎของเหตุการณ์ปกติ (Normal Event Rules) มาเก็บไว้ในหน่วยความจำ หลังจากนั้นก็จะคอยรับข้อมูลล็อกที่ส่งมาจาก Read Logfile Process แล้วนำข้อมูลล็อกที่ได้มาทำการวิเคราะห์ โดยการเปรียบเทียบกับกฎการกรองที่อยู่เก็บในหน่วยความจำว่าตรงกันหรือไม่ ถ้าหากไม่ตรงกับกฎข้อใดเลย จะส่งข้อมูลล็อกนั้นไปให้ส่วน Misuse Detection Engine เพื่อตรวจจับการบุกรุกต่อไป แต่หากรูปแบบตรงกับกฎข้อใดข้อหนึ่งก็หมายความว่าเป็นล็อกของเหตุการณ์ปกติ โปรแกรมจะวนกลับไปคอยรับข้อมูลล็อกของรายการถัดไปมาทำวิเคราะห์ต่อ โดยขั้นตอนการทำงานของส่วน Prefiltering Process แสดงได้ดังภาพประกอบ 4.4



ภาพประกอบ 4.4 ขั้นตอนการทำงานของ Prefiltering Process

4.3.2.3 Misuse Detection Engine

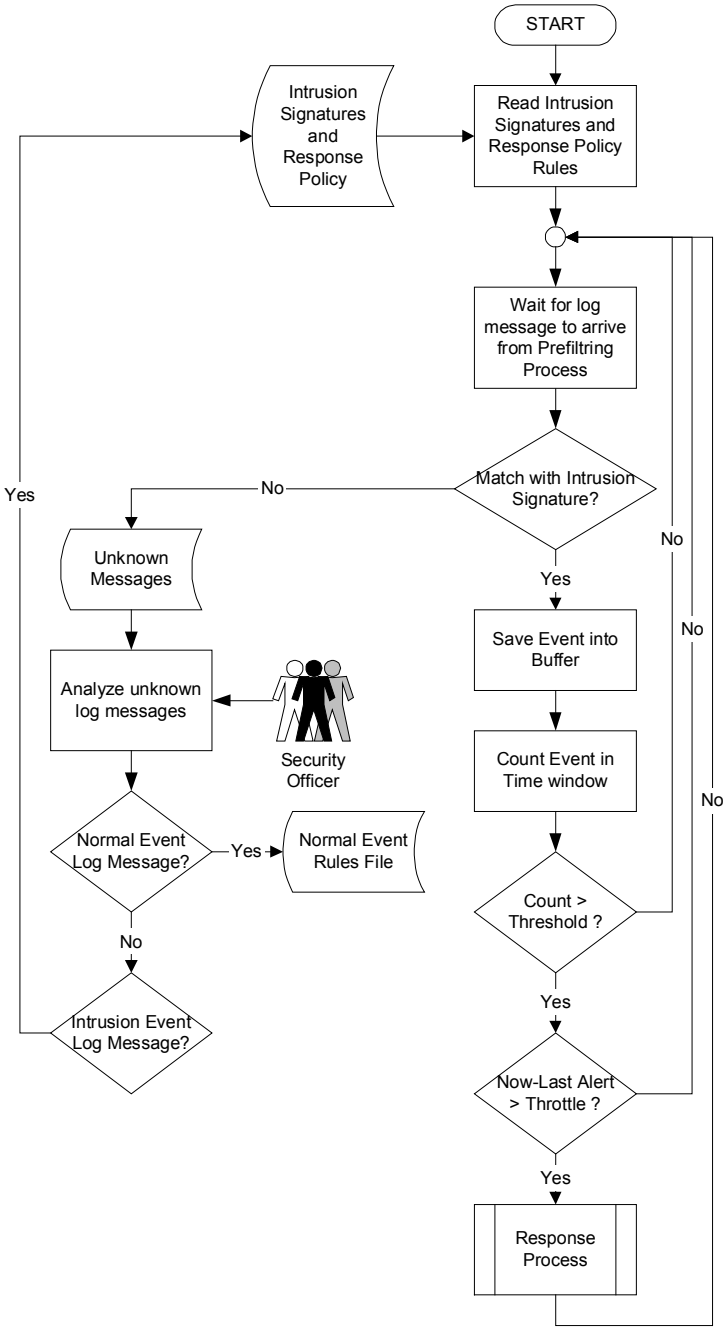
เมื่อข้อมูลล็อกผ่านส่วนของการกรองโดยแยกเหตุการณ์ปกติออกไปแล้ว ข้อมูลที่เหลือและถูกส่งมายังส่วน Misuse Detection Engine จึงเป็นล็อกของเหตุการณ์ที่น่าสงสัยว่าเป็นความผิดปกติหรือการบุกรุก ดังนั้นการทำงานของส่วนนี้จึงเป็นการค้นหาว่ามีเหตุการณ์การบุกรุกหรือความผิดปกติใดเกิดขึ้นในระบบ ซึ่งการค้นหานี้จะใช้วิธีการเปรียบเทียบรูปแบบข้อมูลล็อกของเหตุการณ์นั้นว่าตรงกับกฎของการตรวจจับการบุกรุกข้อใด

สำหรับการสร้างกฎที่ใช้สำหรับการตรวจจับนั้น ทำโดยการเก็บรวบรวมข้อมูลล็อกที่เกิดขึ้นเมื่อมีการบุกรุก ซึ่งทั้งหมดนี้ได้ทำในเครือข่ายที่จัดทำขึ้นสำหรับการทดสอบเท่านั้น ในการทดสอบจะทำการบุกรุกด้วยวิธีการต่าง ๆ แล้วเก็บรวบรวมข้อมูลล็อกที่เป็นร่องรอยของการบุกรุกมาจำแนกตามเหตุการณ์และนำมาเขียนเป็นกฎ ซึ่งกฎในการตรวจจับการบุกรุกจะประกอบด้วยข้อมูลต่าง ๆ ดังนี้

1. รูปแบบร่องรอยของเหตุการณ์การบุกรุกหรือเหตุการณ์ความผิดปกติที่ต้องการตรวจจับ โดยจะเขียนในรูปแบบของ Regular Expression
2. คำอธิบายรายละเอียดของเหตุการณ์
3. คำอธิบาย ข้อแนะนำ หรือวิธีการแก้ไขปัญหาที่เกิดขึ้น
4. วิธีการในการแจ้งเตือน เมื่อตรวจสอบพบเหตุการณ์ตามรูปแบบที่กำหนดไว้ในข้อ 1 สามารถระบุได้หลายวิธีการ ใช้เครื่องหมายเซมิโคลอน(;) คั่นแต่ละวิธีการ
5. ช่วงเวลาที่หน่วงสำหรับการแจ้งเตือน (throttle) เพื่อป้องกันการแจ้งเตือนที่มากเกินไป
6. ค่าความถี่ของเหตุการณ์สำหรับการแจ้งเตือน (threshold)
7. ช่วงเวลาที่ใช้ในการนับจำนวนเหตุการณ์ (time window) เพื่อหาความถี่ในข้อ 6

ข้อมูลในแต่ละส่วนของกฎจะคั่นด้วยเครื่องหมายจุดคู่ 2 ตัว กฎเหล่านี้จะเก็บอยู่ในไฟล์ที่เรียกว่า Intrusion Signature and Response Policy ซึ่งโปรแกรมจะอ่านข้อมูลกฎทั้งหมดจากไฟล์ดังกล่าวมาเก็บไว้ในหน่วยความจำเมื่อเริ่มต้นการทำงาน แล้วจึงทำการเปรียบเทียบรูปแบบข้อมูลล็อกที่ส่งมาจากส่วน Prefiltering กับรูปแบบร่องรอยการบุกรุกของกฎแต่ละข้อโดยใช้วิธีการ Pattern Matching ซึ่งหากพบว่าข้อมูลล็อกนั้นตรงกับกฎข้อใด ก็จะเก็บข้อมูลเหตุการณ์นั้นไว้ในหน่วยความจำโดยจัดกลุ่มเหตุการณ์ตามข้อมูลคำอธิบายรายละเอียดของเหตุการณ์ แล้วนับจำนวนเหตุการณ์ของกลุ่มนั้น ๆ ภายในช่วงเวลาที่กำหนดไว้ในกฎว่ามีค่าเป็นเท่าใด ทั้งนี้เพื่อใช้สำหรับการตรวจจับความผิดปกติที่มีรูปแบบเป็นเหตุการณ์ปกติแต่มีความถี่ของเหตุการณ์ในช่วงเวลาหนึ่ง ๆ มากเกินไปกว่าสภาวะการทำงานปกติของระบบ ดังนั้นถ้าหากค่าที่นับได้มากกว่า

หรือเท่ากับค่าความถี่ดังกล่าวก็จะคำนวณหาช่วงเวลาการแจ้งเตือน (เวลาปัจจุบัน-เวลาที่แจ้งเตือนครั้งสุดท้าย) แล้วนำผลลัพธ์ที่ได้ตรวจสอบกับค่าช่วงเวลาที่หน่วงสำหรับการแจ้งเตือน หากมีค่ามากกว่าก็จะทำการแจ้งเตือนตามวิธีการที่ได้กำหนดไว้ในกฎซึ่งจะกระทำโดยส่วนของ Response Process และกลับไปคอยรับข้อมูลล็อกรายการต่อไป แต่ในกรณีที่ผลการเปรียบเทียบข้อมูลล็อกกับรูปแบบร่องรอยการบุกรุกแล้วไม่ตรงกับกฎข้อใดเลย ทั้งนี้อาจจะเนื่องจากข้อมูลล็อกนั้นเป็นของเหตุการณ์การบุกรุกรูปแบบใหม่ที่ยังไม่รู้จักหรืออาจจะเป็นของเหตุการณ์ปกติก็ได้ ดังนั้นการออกแบบระบบจึงกำหนดให้ข้อมูลล็อกที่ไม่ตรงกับกฎข้อใดเลย ถูกบันทึกลงในไฟล์ที่กำหนดไว้ ซึ่งเรียกไฟล์ที่เก็บล็อกเหล่านี้ว่า Unknown Message File ซึ่งผู้ดูแลระบบต้องคอยตรวจสอบข้อมูล ล็อกในไฟล์ดังกล่าวในภายหลังว่าเป็นของเหตุการณ์ใด โดยหากพบว่าเป็นล็อกเหตุการณ์ปกติ ต้องไปเพิ่มรูปแบบข้อมูลของล็อกนั้นลงในไฟล์ที่ใช้เก็บกฎของเหตุการณ์ปกติ (Normal Event Rules) แต่หากพบว่าเป็นล็อกของเหตุการณ์การบุกรุก ก็ต้องไปเพิ่มกฎการตรวจจับสำหรับเหตุการณ์นั้นลงในไฟล์ Intrusion Signature and Response Policy เพื่อในภายหลังจะได้สามารถตรวจจับเหตุการณ์การบุกรุกรูปแบบใหม่ได้อย่างทันท่วงที ซึ่งขั้นตอนการทำงานทั้งหมดเหล่านี้ สามารถแสดงเป็นแผนภาพได้ดังภาพประกอบ 4.5



ภาพประกอบ 4.5 ขั้นตอนการทำงานของ Misuse Detection Engine

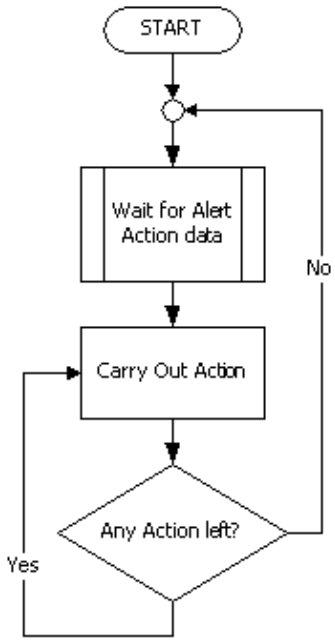
4.3.3 Response Module

Response Module ประกอบด้วย Response Process ซึ่งเป็นส่วนที่ทำหน้าที่แจ้งเตือนไปยังผู้รับผิดชอบเมื่อส่วน Misuse Detection Engine ตรวจสอบพบการบุกรุกเกิดขึ้นในระบบ ซึ่งวิธีการแจ้งเตือนสามารถกำหนดได้ว่าจะใช้วิธีการใดกับแต่ละเหตุการณ์ โดยระบุไว้ใน

ส่วนหนึ่งของกฎที่ใช้สำหรับการตรวจจับการบุกรุกตามที่ได้กล่าวไว้ในหัวข้อที่ 4.2.2.3 สำหรับวิธีการแจ้งเตือนมีดังนี้

1. email เป็นการส่งอีเมล์ไปยังผู้ดูแลระบบ โดยใช้โปรแกรม sendmail
2. write บันทึกข้อมูลล็อกลงในไฟล์ที่ต้องการ
3. exec สั่งให้โปรแกรมที่ต้องการทำงาน

การทำงานของโปรแกรมส่วนนี้จะคอยรับข้อมูลวิธีการแจ้งเตือนที่ส่งมาจาก Misuse Detection Engine และทำการแจ้งเตือนตามวิธีการต่าง ๆที่ระบุมาจนครบ แล้วก็จะกลับไปคอยรับข้อมูลวิธีการแจ้งเตือนรายการถัดไป โดยสามารถขั้นตอนการทำงานในลักษณะแผนภาพการทำงานดังภาพประกอบที่ 4.6



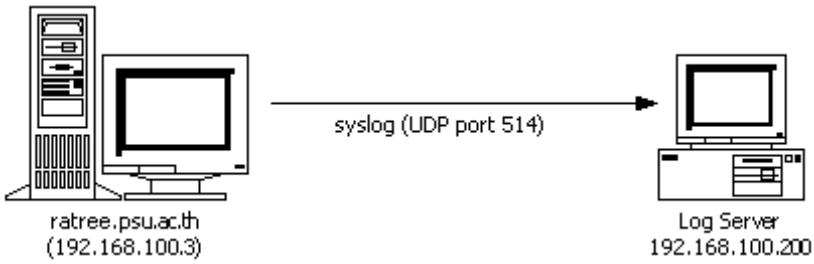
ภาพประกอบ 4.6 ขั้นตอนการทำงานของ Response Process

จากการออกแบบระบบดังกล่าวจะเห็นว่าส่วน Collection Module จะใช้โปรแกรม syslog ซึ่งมีอยู่แล้วบนระบบปฏิบัติการยูนิกซ์แต่ส่วน Analysis Module และ Response Module จะพัฒนาโปรแกรมขึ้นมาทำงาน โดยจะพัฒนาโปรแกรมให้ทำงานในลักษณะ daemon process

4.4 การเก็บรวบรวมข้อมูลล็อก

เนื่องจากระบบตรวจจับการบุกรุกที่ออกแบบนี้ใช้ข้อมูลจากล็อกไฟล์มาวิเคราะห์เพื่อตรวจจับการบุกรุก ซึ่งปัญหาที่สำคัญของล็อกไฟล์ก็คือปริมาณข้อมูลที่มีจำนวนมาก ก่อให้เกิดปัญหาที่จะตามมาก็คือพื้นที่ที่จะต้องใช้เก็บล็อก นอกจากนี้ปริมาณข้อมูลที่มีจำนวนมากยังส่งผลถึงประสิทธิภาพและเวลาที่ใช้ในการวิเคราะห์ข้อมูลด้วยเช่นกัน ดังนั้นในการจัดทำวิทยานิพนธ์07' จึงกำหนดวัตถุประสงค์หนึ่งในการจัดทำวิทยานิพนธ์นี้คือศึกษาวิเคราะห์และนำเสนอระดับของการบันทึกเหตุการณ์ของ syslog เพื่อให้ได้ข้อมูลที่มีประโยชน์กับการวิเคราะห์เพื่อตรวจจับการบุกรุกสูงสุดและใช้เนื้อที่ในการเก็บเหมาะสมที่สุด

ในการวิเคราะห์หาข้อสรุปดังกล่าวข้างต้นผู้ทำการวิจัยได้กำหนดการเก็บรวบรวมข้อมูลล็อกภายใต้สภาวะแวดล้อมการใช้งานคอมพิวเตอร์ของศูนย์คอมพิวเตอร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่ โดยเก็บรวบรวมและวิเคราะห์ล็อกที่เกิดขึ้นบนเครื่องคอมพิวเตอร์ที่ชื่อ ratree.psu.ac.th เนื่องจากเครื่องดังกล่าวใช้ระบบปฏิบัติการยูนิกซ์และให้บริการต่างๆ ในภาพรวมสำหรับบุคลากรของมหาวิทยาลัย โดยในการเก็บรวบรวมข้อมูลล็อกใช้โปรแกรม syslog สำหรับบันทึกล็อกของเหตุการณ์ต่างๆ โดยวิธีการในการเก็บข้อมูลนั้นกำหนดให้เครื่องคอมพิวเตอร์ ratree.psu.ac.th ทำการส่งข้อมูลล็อกทั้งหมดที่เกิดขึ้นในระบบไปยังเครื่องซึ่งทำหน้าที่เป็น Log Server ดังภาพประกอบที่ 4.7 โดยเครื่องดังกล่าวจะคอยรับข้อมูลล็อกทั้งหมดที่ส่งมาและบันทึกลงในล็อกไฟล์ต่างๆ ตามค่าที่กำหนดไว้ในไฟล์ syslog.conf ที่อยู่บนเครื่องนั้น



ภาพประกอบ 4.7 แผนภาพการเก็บรวบรวมข้อมูลล็อกจากเครื่อง ratree.psu.ac.th

สำหรับการป้องกันไม่ให้เครื่อง Log Server รับข้อมูลล็อกจากเครื่องอื่นๆ ที่อาจจะส่งมาเครื่องนี้ได้ โดยให้รับเฉพาะข้อมูลล็อกที่ส่งมาจากเครื่อง ratree.psu.ac.th เท่านั้น จึงทำการติดตั้งโปรแกรม iptables เพื่อทำหน้าที่เป็นไฟร์วอลล์คอยสกัดกั้นข้อมูลที่ไม่ต้องการและใช้คำสั่งดังต่อไปนี้

รับข้อมูลทั้งหมดที่มาเครื่อง Log Server

 # iptables --append INPUT --source 127.0.0.0/8 --in-interface lo --jump ACCEPT

รับข้อมูล log messages ที่มาจากเครื่อง ratree.psu.ac.th

 # iptables --append INPUT --source ratree.psu.ac.th --protocol udp --destination-port syslog -j ACCEPT

ปฏิเสธข้อมูลอื่น ๆที่ส่งมายังพอร์ต syslog

 # iptables --append INPUT --protocol udp --destination-port syslog -j DROP

บันทึกการเพิ่มกฎของ iptables

 # /etc/init.d/iptables save

เนื่องจากวัตถุประสงค์ของเก็บข้อมูลล็อคครั้งนี้เพื่อนำข้อมูลที่ได้ไปวิเคราะห์ว่าเป็นข้อมูลล็อกที่มาจากโปรแกรมใดและเกิดขึ้นในเหตุการณ์ระดับใด รวมถึงพื้นที่ในการจัดเก็บเพื่อหาจุดสมดุลในการเก็บล็อกที่เหมาะสมและสามารถนำข้อมูลไปใช้ในการตรวจจับการบุกรุกได้ ดังนั้นในการเก็บรวบรวมข้อมูลล็อกของเครื่อง ratree.psu.ac.th จึงได้เก็บรวบรวมในระหว่างวันที่ 1 ถึง 31 สิงหาคม 2546 เพราะว่าช่วงเวลาดังกล่าวเป็นช่วงเวลาของการเรียนการสอนตามปกติของมหาวิทยาลัย โดยการเก็บกำหนดให้เครื่อง ratree.psu.ac.th ส่งต่อข้อมูลล็อกไปยังเครื่องหมายเลข IP 192.168.100.200 ที่ติดตั้งระบบปฏิบัติการ Red Hat Linux 7.2 ซึ่งในที่นี้เรียกว่าเครื่อง Log Server โดยการแก้ไขไฟล์ /etc/syslog.conf ของเครื่อง ratree.psu.ac.th ดังนี้

เพิ่มข้อมูล

**.debug @192.168.100.200*

ความหมายคือข้อมูลล็อกทั้งหมดใน priority ระดับ debug ของเครื่อง ratree.psu.ac.th จะถูกส่งไปยังเครื่องหมายเลข IP 192.168.100.200 และกำหนดให้เครื่อง Log Server ซึ่งมีหมายเลข IP 192.168.100.200 สามารถรับข้อมูลล็อกที่ส่งมาได้โดยการเปลี่ยนค่าในไฟล์ /etc/sysconfig/syslog ของเครื่องดังกล่าว

จาก SYSLOGD_OPTIONS="-m 0"

เป็น SYSLOGD_OPTIONS="-m 0 -r"

ซึ่งตัวเลือกการทำงาน syslog ที่เพิ่มเติมคือ -r หมายถึง syslog สามารถรับข้อมูลล็อกจากเครื่องอื่น ๆ จากระยะไกลที่ส่งมาได้

ภายหลังจากเก็บรวบรวมล็อกต่าง ๆแล้ว ก็จะนำข้อมูลเหล่านั้นมาวิเคราะห์ว่าเป็นข้อมูลล็อกที่มาจากโปรแกรมใดบ้าง เป็นล็อกที่เกิดขึ้นในเหตุการณ์ระดับใด การใช้พื้นที่ในการ

จัดเก็บทั้งหมดและแต่ละวันเป็นเท่าใด โดยแสดงข้อมูลในรูปแบบของตารางและกราฟเปรียบเทียบให้เห็นถึง สัดส่วนการใช้พื้นที่ของล็อกเหล่านั้น เพื่อที่ผู้ดูแลระบบสามารถคำนวณปริมาณพื้นที่ที่จะต้องใช้และวางแผนการใช้พื้นที่หรือลดปริมาณล็อกของระบบในอนาคตได้ ซึ่งผลจากการศึกษาและวิเคราะห์นี้ จะได้กล่าวในหัวข้อ 6.5

4.5 สรุป

สำหรับบทนี้ได้กล่าวถึงรายละเอียดการทำงานและการกำหนดค่าในการทำงานของ syslog รวมทั้งวิธีการในการเก็บรวบรวมข้อมูลล็อกที่ศูนย์คอมพิวเตอร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่ และการออกแบบระบบตรวจจับการบุกรุกโดยพัฒนาเป็นระบบตรวจจับการบุกรุกประเภท Host-Based Intrusion Detection System ซึ่งภาพรวมของระบบนี้จะประกอบส่วนสำคัญ 3 ส่วนหลัก คือ ส่วน Data Collection Module ทำหน้าที่ในการเก็บรวบรวมข้อมูลล็อกต่าง ๆ ที่เกิดขึ้นในระบบแล้วบันทึกลงในล็อกไฟล์ต่าง ๆที่กำหนดไว้ โดยใช้ syslog ส่วน Analysis Module เป็นโปรแกรมที่ทำหน้าที่คอยอ่านข้อมูลล็อกที่มีการบันทึกเพิ่มเติมจากล็อกไฟล์แล้วนำมาวิเคราะห์เพื่อตรวจสอบหาเหตุการณ์การบุกรุก และส่วน Response Module ทำหน้าที่ในการแจ้งเตือนไปผู้ดูแลระบบเมื่อผลของการวิเคราะห์ข้อมูลพบว่ามีเหตุการณ์การบุกรุกเกิดขึ้น ซึ่งในบทต่อไปจะกล่าวถึงรายละเอียดการพัฒนาระบบตามที่ได้ออกแบบไว้และผลการทำงานของระบบ