การพัฒนาระบบตรวจจับผู้บุกรุก

Intrusion Detection System Development

โดย

นายบารมี อุดมนิธิกุล

รหัส 43067017

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 2 ปีการศึกษา 2544

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	การพัฒนาระบบตรวจจับผู้บุกรุก
นักศึกษา	นายบารมี อุดมนิธิกุล
อาจารย์ที่ปรึกษา	ผศ.ดร. อาริต ธรรมโน
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2544

บทคัดย่อ

ปัจจุบันมีการนำระบบสารสนเทศเข้ามาช่วยการทำงานในทุกด้าน ประกอบกับความเจริญก้าวหน้าของเครือข่ายอินเตอร์เน็ต ไม่ว่าจะเป็นบริการพาณิชย์อิเล็คทรอนิกส์, เวิลด์ไวด์เว็บ,จดหมายอิเล็คทรอนิกส์,ฯลฯ ทำให้ทุกองค์กรสามารถติดต่อและทำธุรกรรมถึงกันได้ทั่วโลก แต่อุปสรรคสำคัญคือ ความไม่แน่นอนของระบบรักษาความปลอดภัยในการทำธุรกิจ

การศึกษานี้จะทำการพัฒนาระบบตรวจจับผู้บุกรุก เพื่อช่วยงานของผู้ดูแลระบบ โดยจะคอยเฝ้าดู,ตรวจสอบระบบ และแจ้งเตือนได้ตลอดเวลา เพื่อให้ความเสี่ยงในการใช้งานน้อยที่สุด และประโยชน์ในการพัฒนาระบบรักษาความปลอดภัยในอนาคต

Title	Intrusion Detection System Development
Student	Mr. Baramee Udomnitikul
Advisor	Asst.Prof.Dr. Arit Thammano
Level of Study	Master of Science in Information Technology
Major	Information Science
Acedemic Year	2001

Abstract

The present year, Information Technology is implemented for helping any process in almost all of modern organization. In addition ,the growth of Internet using application such as E-commerce ,World Wide Web ,Electronic mail ,etc. make every organization easily communicates with other business parts around the world. But the largest barrier is in the uncertainty of the security system.

This project will develops Intrusion Detection System for assisting system administrator's jobs. The system will all time monitors ,check the critrical events and inform the system administrator ,to manage the risks effectively for businesses and security system development as a whole will benefit in future.

สารบัญ

หน้า

สารบัญตาราง

หน้า

สารบัญภาพ

หน้า

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

ระบบรักษาความปลอดภัยนั้นเป็นเรื่องที่สำคัญมากสำหรับงานด้านต่างๆในการประกอบธุรกิจ ซึ่งในการประกอบธุรกิจทั่วไปนั้นเราจะควบคุมกันที่ตัวบุคคลและวัตถุจริงๆ แต่สำหรับในระบบคอมพิวเตอร์นั้นเราจำเป็นต้องพึ่งวิธีการทางอิเล็คทรอนิกส์เพื่อดูแลความปลอดภัยของระบบ

วิธีการของระบบรักษาความปลอดภัยมีมากมายหลายวิธี ซึ่งจะใช้ในสถานะที่แตกต่างกันออกไป ซึ่งส่วนหนึ่งที่สำคัญเป็นอย่างมากคือ การป้องกันไม่ให้ผู้ใดเข้ามาในระบบเพื่อมาทำการใดๆที่ไม่ได้รับอนุญาติ หรือเข้ามาดูข้อมูลที่เป็นความลับขององค์กรได้

เนื่องจากว่าในการทำงานจริง ผู้ดูแลระบบไม่สามารถคอยเฝ้าดู หรือตรวจสอบระบบได้ตลอดเวลา จึงต้องมีระบบที่จะคอยเฝ้าดูและตรวจสอบระบบคอมพิวเตอร์ของเราแทน และทำงานตลอดเวลา ซึ่งเรียกว่า ระบบตรวจจับผู้บุกรุก

1.2 วัตถุประสงค์

เพื่อสร้างระบบตรวจจับผู้บุกรุกที่คอยเฝ้าดู หรือตรวจสอบระบบได้ตลอดเวลา เป็นการช่วยการทำงานของผู้ดูแลระบบ และระบบตรวจจับผู้บุกรุกที่ทำการสร้างขึ้นนี้จะพยายามออกแบบให้ใช้งานได้ง่ายที่สุด เพราะคำนึงถึงความสะดวกในการใช้งาน ซึ่งปกติผู้ใช้จะไม่ชอบการเปลี่ยนแปลงการเปลี่ยนจากระบบปฏิบัติการที่ใช้อยู่ประจำอย่างเช่น Windows มาใช้ Unix หรือ Linux ก็ต้องใช้เวลาในการศึกษาก่อนการใช้งานมากอยู่แล้ว จึงไม่ต้องการที่จะเพิ่มเวลาศึกษาหรือเพิ่มงานให้ผู้ใช้มากขึ้นอีกในการนำระบบตรวจจับผู้บุกรุกไปใช้

1.3 ขอบเขตของโครงการ

ระบบที่สร้างขึ้นนี้จะสามารถป้องกันการบุกรุกระบบและตรวจจับการบุกรุกได้ในรูปแบบที่ทำการศึกษาเท่านั้น ซึ่งเป็นรูปแบบหลักที่ผู้บุกรุกใช้ในการบุกรุกระบบ ส่วนรูปแบบใหม่ๆที่มีการคิดค้นขึ้นมาทุกวันในปัจจุบัน บางอย่างอาจไม่สามารถป้องกันได้ถ้าอยู่นอกเหนือจากขอบเขตที่ได้ศึกษาไว้

1.4 วิธีการดำเนินงาน

โครงการพัฒนาระบบตรวจจับผู้บุกรุกนี้ จะทำการทดลองโดยใช้ระบบปฏิบัติการ Linux ซึ่งเป็นระบบปฏิบัติการที่แจกจ่ายให้ใช้ฟรี และมีการใช้ทรัพยากรของระบบไม่มากนัก

ออกแบบระบบตรวจจับผู้บุกรุก โดยพิจารณาจากพฤติกรรมที่ผู้บุกรุกมักจะใช้ในการบุกรุกระบบ

ใช้ภาษาเพิร์ล (Perl) ในการตรวจสอบล็อกไฟล์ (log file) และไฟล์ที่สำคัญของระบบ เพื่อตรวจจับการบุกรุกระบบ และการเปลี่ยนแปลง แก้ไขไฟล์ต่างๆ โดยไม่ได้รับอนุญาต ซึ่งภาษาเพิร์ล (Perl) เป็นภาษาที่สามารถทำงานได้ดีกับไฟล์ข้อมูลชนิดตัวอักษร (text file) จึงเหมาะที่จะนำมาใช้ในโครงการนี้

ทดลองการทำงานของระบบที่ได้สร้างขึ้น และแก้ไขข้อผิดพลาดที่อาจเกิดขึ้น

สรุปผลการทำงานของระบบที่ได้สร้างขึ้นพร้อมแนวทางในการนำไปพัฒนาต่อในอนาคต

บทที่ 2

ทฤษฎีและหลักการ

ในส่วนนี้จะบอกถึงทฤษฎีและหลักการ ที่จำเป็นต้องใช้ในการออกแบบระบบตรวจจับผู้บุกรุก ซึ่งแต่ละส่วนจะทำงานสัมพันธ์กัน และมีรายละเอียดต่างๆกัน ดังนี้

2.1 ระบบปฏิบัติการลีนุกซ์ (Linux)

เนื่องจากลีนุกซ์เป็นระบบปฏิบัติการที่แจกจ่ายให้ใช้ฟรี ซึ่งสามารถนำมาใช้ได้โดยไม่ต้องกังวลเรื่องลิขสิทธิ์ ทั้งยังมีคุณลักษณะของระบบยูนิกซ์ (Unix) ที่เป็นระบบหลายงานหรือหลายผู้ใช้อย่างแท้จริง และเป็นระบบที่ใช้ทรัพยากรของระบบไม่มากนัก จึงเหมาะที่จะนำมาใช้ในโครงงานนี้เป็นอย่างยิ่ง

ลีนุกซ์ได้เตรียมเครื่องมือในการพัฒนาโปรแกรมไว้ให้อย่างมากมาย เช่น คอมไพเลอร์ของเพิร์ล (Perl) ,C/C++ ,สมอลล์ท็อค(smalltalk) เป็นต้น

มีการเก็บล็อกไฟล์ (Log file) ที่จะบอกถึงรายละเอียดของเหตุการณ์ต่างๆที่เกิดขึ้นกับระบบ ซึ่งเป็นสิ่งสำคัญที่จะนำไปวิเคราะห์ความปลอดภัยของระบบและทำการตรวจจับการบุกรุกที่เกิดขึ้น

ล็อกไฟล์ของลีนุกซ์ (Linux's Log file system) แบ่งเป็น 2 ประเภทใหญ่ๆ คือ

1. ล็อกไฟล์ที่เกี่ยวกับผู้ใช้งาน เช่น

lastlog	บันทึกการล็อกอินครั้งสุดท้าย
acct	บันทึกคำสั่งที่ถูกรันโดยผู้ใช้งานในระบบทุกคน
utmp	บันทึกเวลาที่ผู้ใช้งานเริ่มล็อกออน (logon) เข้าสู่ระบบ
wtmp	บันทึกเวลาของผู้ใช้งานที่เคย login และ logout จากระบบ
xferlog	บันทึกเมื่อมีการ FTP เข้ามาในระบบ

โดยตำแหน่งที่เก็บ logfile นี้จะเปลี่ยนไปตามระบบปฏิบัติการที่ใช้ ดังนี้

/usr/adm	ในระบบปฏิบัติการ UNIX รุ่นแรกๆ
/var/adm	ในระบบปฏิบัติการ UNIX รุ่นใหม่
/var/log	ในระบบปฏิบัติการ LINUX,free BSD,BSD และ Solaris

2. ล็อกไฟล์ที่เกี่ยวกับระบบ (system) และแอพพลิเคชัน (application) เช่น

etc/syslog.conf	ไฟล์สำหรับตั้งค่าการทำงานของ syslogd
var/log/messages	บันทึกไฟล์ที่ได้ทำการตั้งค่าใน /etc/syslog.conf ซึ่งส่วนใหญ่บันทึกความผิดพลาดที่เกิดจากผู้ใช้งาน เป็นล็อกไฟล์ที่เกิดจาก syslogd โดยจะเก็บทุกเมสเซสที่มีไพออริตี้เป็น info ขึ้นไป ยกเว้นจากแฟกซิลิตี้ mail และ authpriv
/var/log/cron	เป็นล็อกไฟล์ที่เกิดจากล็อกของระบบ cron ที่มาจากเดมอนชื่อ crond
/var/log/secure	เป็นล็อกไฟล์ที่เกิดจาก syslogd ที่มีแฟกซิลิตี้เป็น authpriv

รายละเอียดของการจัดเก็บล็อกไฟล์ที่มีประโยชน์ในการตรวจจับผู้บุกรุก

1. Syslogd

Syslogd เป็นกลไกการจัดเก็บล็อกกิ้งเมสเซส (Logging messages) จากเคอร์เนลและแอปพลิเคชั่นที่รันอยู่บนระบบปฏิบัติการลีนุกซ์ ในตอนแรกสร้างไว้สำหรับโปรแกรม sendmail บันทึกโพรเซสของอีเมล์ทั้งการรับและการส่ง แต่ในปัจจุบันได้มีการเพิ่มฟังก์ชันให้มากขึ้น ซึ่งโดยปกติแล้วเดมอนตัวนี้จะถูกติดตั้งเป็นดีฟอลล์ของระบบอยู่แล้ว และจะต้องมีการคอนฟิก (config) หรือตั้งค่าว่าเมสเซสใดจะถูกเก็บลงไฟล์หรือส่งต่อไปยังโฮสต์อื่น ซึ่งสามารถตั้งค่าได้ในไฟล์ /etc/syslog.conf

การคอนฟิกของ Syslogd ตามดีฟอลล์นั้นจะไม่ได้เก็บเมสเซสจากแอพพลิเคชั่นโปรแกรมในทุกไพออริตี้ (priority)

วิธีการใช้งานและการคอนฟิก Syslogd

เมสเซสทุกๆเมสเซสที่ถูกส่งผ่านมายัง Syslogd จะถูกกำหนดระดับความสำคัญ ประกอบด้วย

- แฟกซิลิตี้ (facility) เป็นส่วนที่บอกว่าเมสเซสนั้นมาจากส่วนไหน เช่น จากระบบเมล์ เป็นต้น
- ไพออริตี้ (priority) เป็นส่วนที่บอกระดับความรุนแรงของเมสเซสนั้น เช่น เป็นการเตือน (warning) เป็นต้น

การคอนฟิก Syslogd นั้นจะใช้ทั้งแฟกซิลิตี้และไพออริตี้ควบคู่กันในการแบ่งแยกแมสเซสต่างๆ ซึ่งมีรายละเอียดดังนี้

แฟกซิลิตี้	คำอธิบาย
user	เป็นแฟกซิลิตี้พื้นฐานสำหรับโปรแกรมใดๆ
auth	ถูกใช้โดยระบบออโธไรเซชัน (authorization) หรือ โปรแกรมที่มีการถามชื่อผู้ใช้และรหัสผ่าน เช่น login,su,getty,ftpd เป็นต้น
authpriv	ระบบออโธไรเซชันที่เกี่ยวข้องกับสิทธิพิเศษ
kern	เป็นแมสเซสที่ถูกสร้างจากเคอร์เนล
mail	ใช้ในระบบเมล์
daemon	ซิสเต็มหรือเน็ตเวิร์กเดมอน เช่น in.ftpd
lpr	ระบบการพิมพ์
news	ใช้สำหรับระบบข่าว
uucp	ใช้สำหรับระบบยูยูซีพี
cron	ใช้สำหรับระบบครอน (cron) และเอที (at)
local0..7	ใช้สำหรับงานโลคอล
mark	ใช้สำหรับการทำไทม์แสตมป์ (timestamp) ซึ่งส่งแมสเซสทุกๆ 20 นาที
syslog	ใช้สำหรับ syslog
*	หมายถึงทุกแฟกซิลิตี้ ยกเว้น mask

ตารางที่ 2-1 แสดงรายละเอียดของแฟกซิลิตี้

ไพออริตี้	คำอธิบาย
none	ไม่ต้องทำการส่งแมสเซสที่มาจากแฟกซิลิตี้ที่กำหนด
debug	ใช้สำหรับการดีบัก (debug)
info	เป็นแมสเซสที่ให้ข้อมูล (information)
notice	แสดงสภาพที่ต้องจับตาดูเป็นพิเศษ
warning	คำเตือนต่างๆ
warn	เหมือนกับ warning (แต่เลิกใช้ไปแล้ว)
err	ความผิดพลาดต่างๆ
error	เหมือนกับ err (แต่เลิกใช้ไปแล้ว)
crit	สภาพที่มีปัญหารุนแรง อย่างเช่น ปัญหาเกี่ยวกับฮาร์ดแวร์
alert	สภาพที่ต้องการการดูแลในทันที
emerg	สภาวะเร่งด่วนทุกชนิด เช่น ระบบเกิดความผิดพลาด

ตารางที่ 2-2 แสดงรายละเอียดของไพออริตี้โดยเรียงลำดับความสำคัญจากน้อยไปหามาก

แอ็กชันจะเป็นตัวกำหนดว่าให้ทำอย่างไร เมื่อมีแมสเซสที่ตรงกับแฟกซิลิตี้ และไพออริตี้ที่กำหนด โดยมี 4 รูปแบบ คือ

1. บันทึกลงไฟล์ หรือดีไวซ์ (device) ซึ่งประกอบด้วยชื่อไฟล์หรือชื่อดีไวซ์ และต้องนำหน้าด้วยเครื่องหมาย / (slash) เช่น /var/log/messages หรือ /dev/console
2. ส่งข้อความให้ผู้ใช้ ซึ่งประกอบด้วย username เช่น alice ซึ่งสามารถส่งข้อความให้ผู้ใช้หลายๆคนพร้อมกันได้ โดยใช้เครื่องหมายคอมม่า เช่น alice,bob
3. ส่งข้อความถึงผู้ใช้งานทุกคน โดยใช้เครื่องหมายดอกจัน (*)
4. ส่งข้อความไปยังโฮสต์อื่น โดยใส่ชื่อโฮสต์ และต้องใช้เครื่องหมาย @ นำหน้าชื่อโฮสต์นั้น เช่น @backup.kmitl.ac.th

ตัวอย่างของแอ็กชันแสดงไว้ในตารางที่ 2-3

แอ็กชัน	คำอธิบาย
/dev/console	ส่งข้อความออกทางอุปกรณ์
/var/log/messages	เขียนแมสเซจลงในไฟล์ที่ชื่อ /var/log/messages
@loghost	ส่งแมสเซจต่อไปยังโฮสต์อื่น
alice,user1	ส่งแมสเซจไปยังผู้ใช้ชื่อ alice และ user1
*	ส่งแมสเซจไปยังผู้ใช้ที่กำลังล็อกออนอยู่ทุกคน

ตารางที่ 2-3 ตัวอย่างของแอ็กชัน

นอกจากนี้ยังมีการใช้สัญลักษณ์พิเศษช่วยในการคอนฟิก เพื่อให้การคอนฟิกค่าต่างๆมีความสะดวกและง่ายยิ่งขึ้น แสดงในตารางที่ 2-4

สัญลักษณ์	คำอธิบาย
,	เครื่องหมายคอมม่า ใช้สำหรับกำหนดหลายแฟกซิลิตี้ที่มีไพออริตี้เดียวกัน เช่น mail,auth.info มีความหมายเหมือนกับ mail.info และ auth.info
;	เครื่องหมายเซมิโคลอน เป็นการกำหนดการเลือกแฟ่กซิลิตี้ และไพออริตี้หลายอันเข้ากับแอ็กชันเดียว เช่น mail.info;kern.crit /dev/console
=	เครื่องหมายเท่ากับ หมายถึง เฉพาะไพออริตี้นั้นๆเท่านั้น
!	เครื่องหมายอัศเจรีย์ หมายถึง ทุกไพออริตี้ที่มีความสำคัญต่ำกว่า
#	เครื่องหมาย hash ใช้เป็นคอมเมนท์ (comment)
|	เครื่องหมายเส้นตรงแนวตั้ง หรือไปป์ (pipe) ใช้ในส่วนแอ็กชัน เพื่อเชื่อม 2 คำสั่งเข้าด้วยกัน
@	เครื่องหมาย at ใช้ในส่วนแอ็กชันเพื่อส่งต่อแมสเซจไปยัง loghost อื่น
*	เครื่องหมายดอกจัน หมายถึง ทุกไพออริตี้หรือทุกแฟกซิลิตี้

ตารางที่ 2-4 แสดงสัญลักษณ์พิเศษที่ใช้ใน syslog.conf

2. Utmp และ Wtmp

เมื่อผู้ใช้งานมีการล็อกอินเข้ามาในระบบ และล็อกเอาต์ออกจากระบบ จะมีการบันทึกในไฟล์ /usr/adm/utmp และ /usr/adm/wtmp

ไฟล์ /usr/adm/utmp ใช้บันทึกผู้ที่กำลังล็อกออนในระบบอยู่ ข้อมูลในไฟล์นี้สามารถแสดงได้โดยใช้คำสั่ง finger ,w ,who ,users ,whodo

โดยคำสั่ง w จะแสดงข้อมูลของผู้ใช้งานแต่ละคน ประกอบด้วย login name ,terminal being used ,remote host ,login time ,total cpu time ,user time ,system time และ active process

ไฟล์ /usr/adm/wtmp บันทึกทุกครั้งที่มีผู้ใช้งานล็อกอินเข้าสู่ระบบ และล็อกเอาต์ออกจากระบบ ข้อมูลในไฟล์นี้จะมีขนาดใหญ่ขึ้นเรื่อยๆ เนื่องจากจะบันทึกทุกๆครั้งเมื่อมีการล็อกอินและล็อกเอาต์ ไม่เหมือนกับไฟล์ utmp คือเมื่อผู้ใช้ได้ล็อกเอาต์ออกไป ข้อมูลในไฟล์ก็จะถูกลบออกไปด้วย ข้อมูลในไฟล์ wtmp นี้สามารถแสดงได้โดยใช้คำสั่ง last

โดยคำสั่ง last จะแสดง user name ,terminal ,remote host ,เวลาที่เริ่มต้นใช้งาน และเวลาที่ออกจากระบบ หากเป็นข้อความ still logged in แสดงว่าผู้ใช้ยังล็อกอินอยู่ในระบบ และสุดท้ายแสดงเวลาทั้งหมดที่เข้ามาใช้งานระบบ

ไฟล์ wtmp นี้จะมีขนาดใหญ่ขึ้นเรื่อยๆ ดังนั้นเพื่อป้องกันไม่ให้ไฟล์มีขนาดใหญ่เกินไป ควรสำรองข้อมูลของไฟล์นี้ไว้ โดยอาจทำเป็นต่อวันหรือต่อสัปดาห์ ทำให้เมื่อต้องการจะตรวจสอบไฟล์นี้จะทำให้ง่ายขึ้น และข้อมูลไม่มากเกินไป

ผู้ดูแลระบบควรใช้คำสั่ง last เพื่อตรวจสอบระบบอย่างสม่ำเสมอ เพื่อดูว่ามีผู้ล็อกอินเข้ามาในระบบเป็นใคร และมาจากไหนบ้าง และถ้าสังเกตุเห็นถึงสิ่งผิดปกติที่เกิดขึ้น เช่น มีการล็อกอินเข้ามาใช้งานจากโฮสต์ (host) อื่นที่ไม่รู้จักบ่อยครั้ง ,มีการล็อกอินมาจากหลายๆที่ในเวลาเดียวกัน นั่นก็ควรที่จะสงสัยได้ว่ามีการบุกรุกเข้ามาใช้งานในระบบโดยไม่ได้รับอนุญาติเกิดขึ้นแล้ว

ครอนและครอนแท็ป (Cron and Crontab)

ครอน (Cron)เป็นระบบจัดการเอ็กซิคิวต์ (execute) คำสั่งตามเวลาที่กำหนดได้ล่วงหน้า โดยตัวครอนเดมอน (Crond) ถูกรันจากไฟล์ /etc/rc หรือ /etc/rc.local ซึ่งตัวเดมอนนี้เมื่อเริ่มต้นทำงานจะอ่านค่าเริ่มต้นจากไฟล์ /var/spool/cron ซึ่งเป็นส่วนที่ผู้ใช้แต่ละคนสามารถสร้างตารางเวลาของแต่ละคนได้ และชื่อไฟล์จะมีชื่อเดียวกันกับ ล็อกอินเนม (login name) ของผู้ใช้ที่สร้าง

ครอนแท็ป (Crontab) เป็นการกำหนดตารางเวลาล่วงหน้าของระบบ ซึ่งจะอยู่ในไฟล์ /etc/crontab โดยผู้ที่จัดการกับไฟล์นี้ได้ต้องเป็น root เท่านั้น

ไฟล์ครอนแท็ป มีข้อกำหนดในการเขียน คือ บรรทัดที่ว่าง และบรรทัดที่ขึ้นต้นด้วย # ถือว่าเป็นคอมเมนต์ ในไฟล์ crontab สามารถกำหนดค่าตัวแปรในรูปแบบ ดังนี้

name = value

และในส่วนของการกำหนดตารางการทำงาน จะมีรูปแบบ ดังนี้

MINUTE HOUR DAY OF MONTH MONTH DAY OF WEEK [USER] COMMAND

โดย	MINUTE	คือ นาที	ค่าที่เป็นไปได้ คือ 0-59
	HOUR	คือ ชั่วโมง	ค่าที่เป็นไปได้ คือ 0-23
	DAY OF MONTH	คือ วันที่	ค่าที่เป็นไปได้ คือ 0-31
	MONTH	คือ เดือน	ค่าที่เป็นไปได้ คือ 0-12 (หรือเป็นชื่อเดือนก็ได้)
	DAY OF WEEK	คือ วัน	ค่าที่เป็นไปได้ คือ 0-7 (0 หรือ 7 เป็นวันอาทิตย์ หรือเป็นชื่อวันก็ได้)

ค่าของฟิลด์ ถ้าหากเป็น * ก็จะหมายถึง ทุกค่าที่เป็นไปได้

สามารถกำหนดเป็นช่วงได้ โดยใช้เครื่องหมาย – ตัวอย่าง เช่น hour เป็น 8-10 ก็จะหมายถึงเวลา 8,9,10 และเราสามารถกำนดค่าเป็นลิสท์ได้เช่น 1,3,5,7 หรือ 1-3, 9-11 ก็ได้

สามารถกำหนดเป็นสเต็ปได้ เช่น hour กำหนดเป็น 0-23/2 ก็จะหมายถึง 0,2,4,6,8,10,12,14,16,18,20,22 หรือหมายถึง ทุกๆ 2 ชั่วโมง นั่นเอง หรืออาจเขียนอีกแบบนึงได้ */2 และเครื่องหมาย % มีความหมายเท่ากับ เครื่องหมายขึ้นบรรทัดใหม่

ฟิลด์ month และ day of week สามารถใช้ชื่อภาษาอังกฤษสามตัวแรกแทนได้ เช่น วันอาทิตย์ ก็แทนด้วย sun หรือ เดือนธันวาคม ก็จะแทนด้วย dec เป็นต้น โดยไม่ต้องสนใจว่าจะเป็นตัวพิมพ์เล็กหรือตัวพิมพ์ใหญ่

ตัวอย่างเช่น

5	4	*	*	sun	หมายถึง คำสั่งจะรันทุกวันอาทิตย์เวลา 4:05 น.
30	4	1,15	*	5	หมายถึง คำสั่งจะรันที่เวลา 4:30 ทุกวันที่ 1 และ 15 ของเดือน รวมถึงทุกวันศุกร์ด้วย
23	0-23/2	*	*	*	หมายถึง คำสั่งจะรันที่เวลา 23 นาทีหลังเที่ยงคืน แล้วรันทุกๆสองชั่วโมง ทุกวัน

2.2 ภาษาเพิร์ล (Perl)

ภาษาเพิร์ล (Perl) ย่อมาจากคำว่า Practical Extraction and Report Language ถูกพัฒนาขึ้นเพื่อใช้งานกับระบบปฏิบัติการยูนิกซ์ (Unix) ซึ่งในขณะนั้นการพัฒนาแอพพลิเคชันบนระบบปฏิบัติการยูนิกซ์เป็นเรื่องที่ยุ่งยากและต้องมีความรู้ความเข้าใจในภาษาโปรแกรมหลายภาษา เพราะในขณะนั้นภาษาโปรแกรมแต่ละภาษาจะทำงานในเรื่องใดเรื่องหนึ่งโดยเฉพาะ การจะทำงานออกมาสักงานหนึ่งจะต้องใช้ภาษาโปรแกรมแทบทุกภาษาที่มี

ภาษาเพิร์ล (Perl) ถูกสร้างขึ้นมาเพื่อให้ทำงานทุกส่วนเสร็จสมบูรณ์ในตัวเองโดยไม่ต้องไปเรียกใช้งานภาษาอื่นอีก และรวบรวมเอาส่วนดีของภาษาต่างๆเข้ามาไว้ด้วยกัน ทำให้การสร้างแอพพลิเคชันบนระบบปฏิบัติการยูนิกซ์หรือลีนุกซ์เป็นไปได้อย่างมีประสิทธิภาพมากขึ้น และที่สำคัญภาษาเพิร์ล (Perl)ถูกออกแบบมาให้ใช้งานง่าย โครงสร้างของภาษาไม่ซับซ้อนโดยมีโครงสร้างของภาษาคล้ายกับภาษา C มาก

ในปัจจุบันภาษาเพิร์ลสามารถใช้งานได้กับระบบปฏิบัติการยูนิกซ์,ลีนุกซ์,MS/DOS, Macintosh,OS/2,Amiga และอื่นๆ ความสามารถที่สำคัญของภาษาเพิร์ลคือการจัดการกับไฟล์ text ได้เป็นอย่างดี ทำให้ภาษาเพิร์ลกลายมาเป็นภาษาสำคัญที่มีการใช้งานกันบนเครือข่ายอินเตอร์เน็ต และยังมีข้อดีอื่นๆอีกมากมาย

คุณสมบัติเด่นของ ภาษาเพิร์ล (Perl) คือ การทำการตรวจสอบรูปแบบของข้อความ (pattern matching) โดยใช้เรกกูล่าร์เอ็กซ์เพรสชั่น (regular expression) ซึ่งเป็นเทมเพลท (template) ในการทำการตรวจสอบข้อความนั้น โดยภายหลังจากการตรวจสอบข้อความเราอาจจะแทนที่ข้อความนั้นด้วยข้อความอื่นๆได้หากต้องการ

การใช้งานภาษาเพิร์ลไม่ต้องเสียค่าใช้จ่ายใดๆทั้งสิ้น เพราะตัวแปรภาษาเพิร์ลแจกจ่ายให้ใช้ได้ฟรีภายใต้ลิขสิทธิ์ GNU คือหมายความว่าถ้าเราได้ติดตั้งและใช้งานตัวแปรภาษาเพิร์ลในเครื่องของเราแล้ว เราสามารถไป download โค้ดฟังก์ชัน หรือโมดูลต่างๆที่มีการเขียนเอาไว้แล้วมาใช้งานได้โดยไม่เสียค่าใช้จ่ายใดๆทั้งสิ้น

ภาษาเพิร์ลเป็นภาษาสคริปต์ คือ โค้ดคำสั่งที่เราเขียนขึ้นมาจะอยู่ในรูปไฟล์ text และจะรันในรูปแบบของไฟล์ text โดยใช้อินเตอร์พรีเตอร์ (interpreter) ไม่ต้องแปลโค้ดคำสั่งเป็นไบนารีไฟล์ (.exe) ซึ่งจะทำให้กระบวนการแปลความหมายคำสั่งของภาษาเพิร์ลเร็วกว่าภาษาที่ใช้คอมไพเลอร์ (compiler) เป็นตัวแปลภาษา โดยการทำงานของตัวอินเตอร์พรีเตอร์ (interpreter) นั้นจะทำงานแบบบรรทัดต่อบรรทัด คือ อ่านโค้ดคำสั่งมาหนึ่งบรรทัดแล้วก็ทำงานให้ผลออกมาเลย

โดยพื้นฐานของโครงการนี้เป็นระบบที่ทำงานกับล็อกไฟล์ (Log file) ซึ่งต้องนำล็อกไฟล์มาทำการตรวจสอบและวิเคราะห์ตามรูปแบบที่กำหนด และทำการแสดงผลในรูปแบบที่ต้องการ ทำให้ภาษาเพิร์ล (Perl) เป็นภาษาที่เหมาะสมที่จะนำมาใช้ในโครงการนี้

บทที่ 3

การออกแบบและหลักการทำงานของระบบตรวจจับผู้บุกรุก

3.1 โครงสร้างของระบบ

ระบบตรวจจับผู้บุกรุก ประกอบด้วยส่วนสำคัญ 2 ส่วน คือ

1. ส่วนบันทึกเหตุการณ์ที่เกิดขึ้นในระบบ (Logging)
2. ส่วนตรวจจับผู้บุกรุก (Intrusion Detection)

3.2 ส่วนบันทึกเหตุการณ์ที่เกิดขึ้นในระบบ (Logging)

เพื่อให้ผู้ดูแลระบบทราบเมื่อมีเหตุการณ์ต่างๆเกิดขึ้นในระบบ โดยจะบันทึกสิ่งต่างๆได้ในไฟล์ เรียกว่า " log files" หรือ ล็อกไฟล์ ซึ่งรายชื่อและตำแหน่งที่เก็บล็อกไฟล์ได้กล่าวไว้แล้วข้างต้น

สิ่งที่ได้จากล็อกไฟล์สามารถแบ่งเป็นส่วนหลักๆ ได้ ดังนี้

- ความผิดพลาด (Error) ของระบบ ว่ามีความผิดพลาดอะไรเกิดขึ้นบ้าง เช่น ความผิดพลาดของโปรแกรมที่กำลังทำงานอยู่,ความผิดพลาดในการอ่านข้อมูล เพื่อที่ผู้ดูแลระบบจะได้สามารถทำการหาสาเหตุและทำการแก้ไขข้อบกพร่องที่เกิดขึ้นได้
- ประสิทธิภาพ (Performance) ของระบบ เช่น อัตราการใช้งานของระบบ, มีการส่งผ่านข้อมูลมากแค่ไหน, ระบบปัจจุบันสามารถรองรับการใช้งานได้เพียงพอหรือไม่ ช่วยในผู้ดูแลระบบสามารถตัดสินใจในการปรับปรุง หรือ อัพเกรด (upgrade) เครื่องให้บริการเพื่อที่จะสามารถรองรับการใช้งานได้อย่างเหมาะสม และมีประสิทธิภาพ
- ความปลอดภัย (Security) ของระบบ เช่น ผู้ใช้งานแต่ละคนทำอะไรกับระบบบ้าง, มีใครที่ไม่ได้รับอนุญาตเข้ามาใช้งานระบบบ้าง และถ้ามีก็สามารถตรวจสอบได้ว่ามาจากที่ไหน เพื่อให้ผู้แลระบบจะได้ปรับปรุงและพัฒนาระบบในการป้องกันการบุกรุกระบบต่อไป

ระบบตรวจจับผู้บุกรุกที่สร้างขึ้นจะอาศัยการจัดเก็บล็อกไฟล์ของ syslogd ที่เป็นระบบจัดเก็บล็อกไฟล์ที่ระบบส่วนใหญ่จะมีอยู่แล้ว เพราะต้องการให้ทุกระบบสามารถนำระบบตรวจจับผู้บุกรุกที่สร้างขึ้นนี้ไปใช้ได้โดยไม่ต้องติดตั้งส่วนอื่นๆเพิ่มอีก โดย syslogd จะจัดเก็บล็อกไฟล์ ใน /var/log/messages

3.3 ส่วนตรวจจับผู้บุกรุก (Intrusion Detection)

เป็นส่วนที่นำข้อมูลในล็อกไฟล์มาวิเคราะห์ตรวจจับพฤติกรรมการบุกรุกที่เกิดขึ้นตามรูปแบบที่กำหนดไว้ ซึ่งถ้ามีเหตุการณ์ที่น่าสงสัยเกิดขึ้นกับระบบ ส่วนตรวจจับผู้บุกรุกจะทำการแจ้งเตือนไปยังผู้ดูแลระบบทันที และจะทำการส่งรายงานประจำวันไปให้ผู้ดูแลทุกวัน เพื่อให้ทราบว่าตลอดทั้งวันนั้นมีเหตุการณ์อะไรเกิดขึ้นกับระบบบ้าง

สามารถแบ่งได้เป็น 2 ส่วนหลักๆ คือ

3.3.1 ส่วนตรวจจับการบุกรุกที่ทำงานตามเวลาที่เราต้องการและสร้างรายงานประจำวัน (daily report)

เป็นส่วนที่ทำงานตามเวลาที่เราต้องการ ซึ่งเราสามารถสั่งให้โปรแกรมตรวจจับการบุกรุก (ids) ทำงานได้ทุกเวลาที่เราต้องการโดยไม่มีผลกระทบกับส่วนที่ทำงานตลอดเวลา (ids.d) และเป็นส่วนที่สร้างรายงานประจำวันบอกว่าตลอดวันนั้นมีเหตุการณ์อะไรเกิดขึ้นกับระบบของเราบ้าง โดยจะทำการเปรียบเทียบพฤติกรรมผู้ใช้ที่ได้บันทึกในล็อกไฟล์กับรูปแบบพฤติกรรมการบุกรุกที่รู้จัก (Misuse Detection) ซึ่งประกอบด้วยพฤติกรรมต่างๆดังนี้ คือ

- มีการเข้าใช้งานในชื่อผู้ดูแลระบบ (Root login)

  เป็นการเข้าใช้งานในสิทธิของผู้ดูแลระบบอย่างถูกต้อง ข้อมูลที่ได้มีประโยชน์ในการที่จะวิเคราะห์การบุกรุกระบบ ซึ่งถ้ามีการเข้าใช้งานในชื่อผู้ดูแลระบบมากเกินกว่าที่ผู้ดูแลระบบเข้าใช้งานจริงก็อาจหมายความได้ว่ามีการบุกรุกระบบเกิดขึ้น

- การพยายามเข้าใช้ระบบโดยใช้ชื่อผู้ใช้ที่ไม่มีอยู่จริง (bad username login)

  เป็นการบันทึกเหตุการณ์ที่ผู้ที่ไม่ได้เป็นผู้ใช้ระบบ (user) หรือผู้บุกรุกพยายามเข้ามาในระบบโดยการเดาชื่อผู้ใช้และรหัสผ่าน

- การพยายามเข้าใช้ระบบโดยใส่รหัสผ่านไม่ถูกต้อง (bad password login)

ในกรณีที่ผู้บุกรุกรู้ชื่อ (user name) ผู้ใช้แต่ไม่รู้รหัสผ่าน(password) จึงทำการคาดเดารหัสผ่านหลายๆครั้งเพื่อที่จะเข้ามาในระบบ ซึ่งลักษณะของรหัสผ่านที่มักจะถูกคาดเดาได้ง่าย เช่น

1. ตัวอักษรที่เรียงกัน เช่น abc ,123
2. รหัสผ่านเหมือนชื่อผู้ใช้ เช่น ผู้ใช้ชื่อ smith ก็ตั้งรหัสผ่านเป็น smith ด้วยซึ่งลักษณะการตั้งรหัสผ่านแบบนี้มีชื่อเรียกเฉพาะว่าjoe account
3. เรียงอักษรชื่อผู้ใช้ใหม่หรือสลับกัน เช่น ผู้ใช้ชื่อ adisak ก็ใช้รหัสผ่านเป็น kasida เป็นต้น
4. คำต่างๆที่มีอยู่ในพจนานุกรม ที่สามารถถูกคาดเดาได้ง่ายโดย library ต่างๆ ที่ทำงานร่วมกับโปรแกรม Hack
5. ชื่อต่างๆที่เกี่ยวพันกับผู้ใช้ เช่น ชื่อเพื่อน ,ชื่อสัตว์เลี้ยง ,สิ่งของ ,วันเกิด เป็นต้น ซึ่งถ้าผู้บุกรุกรู้จักกับผู้ใช้อาจทำให้สามารถคาดเดาได้ง่าย

- พยายามเข้าใช้งานในชื่อของผู้ดูแลระบบแต่ไม่สำเร็จ (Failed root login) ซึ่งจะแสดงรวมกับส่วนวิเคราะห์พฤติกรรมการพยายามเข้าใช้ระบบโดยใส่รหัสผ่านไม่ถูกต้อง (bad password) โดยจะมีชื่อผู้ login=root หรือผู้ดูแลระบบ

เป็นการพยายามคาดเดารหัสผ่านของ root โดยรหัสผ่านที่มักจะถูกคาดเดาได้กล่าวไว้แล้วข้างต้น

- พยายามเปลี่ยนสิทธิเป็นผู้ดูแลระบบแต่ไม่สำเร็จ (su command failure)

ผู้บุกรุกเข้ามาในระบบโดยเข้ามาในรูปของผู้ใช้ระบบ และพยามยามเปลี่ยนสิทธิของตัวเองเป็นผู้ดูแลระบบ โดยใช้คำสั่ง su หลายๆครั้งแต่ไม่สำเร็จ

- มีการเปลี่ยนสิทธิเป็นรูตได้สำเร็จ (su command success)

เพื่อใช้ในการตรวจสอบว่าใครบ้างที่ทำการเปลี่ยนสิทธิตัวเองเป็นผู้ดูแลระบบ และผู้นั้นได้รับอนุญาตอย่างถูกต้องหรือไม่

- มีการสร้างชื่อผู้ใช้งานระบบใหม่ (User account create)

เมื่อผู้บุกรุกเข้ามาในระบบได้แล้ว มักสร้างประตูหลัง (Backdoor) ไว้ในระบบ เพื่อเป็นช่องทางในการเข้ามาในระบบครั้งต่อไป ซึ่งก็คือการสร้างชื่อผู้ใช้งานระบบขึ้นมาใหม่นั่นเอง หรือ ผู้บุกรุกที่ได้สิทธิเป็นรูตแล้วมักสร้างผู้ใช้ใหม่ในกลุ่มของรูตขึ้นในระบบ เพื่อที่จะสามารถใช้บริการบางอย่าง หรือเปลี่ยนแปลง

ไฟล์สำคัญที่สามารถทำได้โดยผู้ใช้ในกลุ่มรูตเท่านั้น (ขึ้นอยู่กับเพอร์มิสชันที่ตั้งไว้) ดังนั้นผู้บุกรุกไม่จำเป็นต้องล็อกอินเข้ามาในระบบโดยใช้ชื่อรูตอีกต่อไป เพื่อลดความเสี่ยงที่จะถูกตรวจพบ เช่น กรณีที่มีผู้ใช้ระบบเป็นจำนวนมาก ผู้ดูแลระบบไม่สามารถตรวจสอบได้ทั่วถึง ซึ่งต่างกับการล็อกอินโดยใช้ชื่อรูตโดยตรง ผู้ดูแลระบบสามารถสังเกตุได้ง่าย เพราะล็อกไฟล์จะเก็บเป็นตัวอักษรพิมพ์ใหญ่ และจะพบความผิดปกติได้ทันทีถ้ามีการล็อกอินในช่วงที่ผู้เป็นรูตไม่อยู่

- มีการลบชื่อผู้ใช้งานระบบ (User account delete)

เมื่อผู้บุกรุกเข้ามาในระบบได้แล้ว และต้องการลบร่องรอยของตน เช่น มีการสร้างชื่อผู้ใช้ใหม่ไว้ในระบบ ผู้บุกรุกมักลบชื่อผู้ใช้ที่ตนไม่ต้องการใช้งานแล้ว เพื่อไม่ให้ผู้ดูแลระบบสามารถติดตามได้ นอกจากนี้การกระทำเช่นนี้อาจเนื่องมาจากการที่ผู้บุกรุกต้องการสร้างความเสียหายให้แก่ระบบหรือทำให้ผู้ใช้คนใดคนหนึ่งไม่สามารถเข้าใช้ระบบได้ก็เป็นได้

- มีการเปลี่ยนแปลงในไฟล์ passwd (file passwd change)

เป็นการบันทึกการเปลี่ยนแปลงแก้ไขไฟล์ passwd เพราะอาจจะเกิดขึ้นจากผู้บุกรุกที่ไม่ได้รับอนุญาต ซึ่งจุดมุ่งหมายอาจจะเพื่อเพิ่มสิทธิให้กับผู้ใช้คนใดคนหนึ่ง เพราะบางครั้งการสร้างผู้ใช้ใหม่ในกลุ่มรูตจะทำให้ผู้ดูแลระบบสามารถสังเกตุเห็นความผิดปกติได้ง่าย เนื่องจากในไฟล์ passwd สามารถเห็นความแตกต่างได้ชัดเจนระหว่างผู้ใช้ธรรมดาซึ่งตามปกติมีเลขกลุ่มเป็น 500-1000 กับผู้ใช้ในกลุ่มรูตที่มีเลขกลุ่มเป็น 0 ผู้บุกรุกอาจทำการสร้างผู้ใช้ปกติในระบบขึ้นมาก่อนแล้วจึงเพิ่มสิทธิของผู้ใช้คนนั้นโดยการแก้ไขในไฟล์ passwd หรืออีกกรณีหนึ่งคือ ผู้บุกรุกเห็นว่าการเพิ่มผู้ใช้ใหม่เข้าไปในระบบอาจทำให้ถูกตรวจพบได้โดยง่าย เลยอาจไปขโมยแอ็กเคาท์ของผู้ใช้ที่มีอยู่แล้วมา และทำการเพิ่มสิทธิของผู้ใช้คนนั้นให้อยู่ในกลุ่มรูตต่อไป

- ล็อกของระบบเปลี่ยนแปลง (log file change)

ระบบตรวจจับผู้บุกรุกจะตรวจสอบข้อมูลในไฟล์ messages และ ids-log0 หากพบว่าข้อมูลจากล็อกไฟล์ทั้งสองไม่ตรงกัน ระบบตรวจจับผู้บุกรุกจะมีการแจ้งเตือนทันทีว่ามีการเปลี่ยนแปลงแก้ไขล็อกไฟล์โดยไม่ได้รับอนุญาต

- ไฟล์ครอนแท็ปของระบบมีการเปลี่ยนแปลง (file crontab change)

ระบบตรวจจับผู้บุกรุกจะตรวจสอบข้อมูลในไฟล์ crontab และ ids-crontab หากพบว่าข้อมูลในไฟล์ทั้งสองไม่ตรงกัน ระบบตรวจจับผู้บุกรุกจะมีการแจ้งเตือนทันทีว่ามีการเปลี่ยนแปลงแก้ไขไฟล์ crontabโดยไม่ได้รับอนุญาต

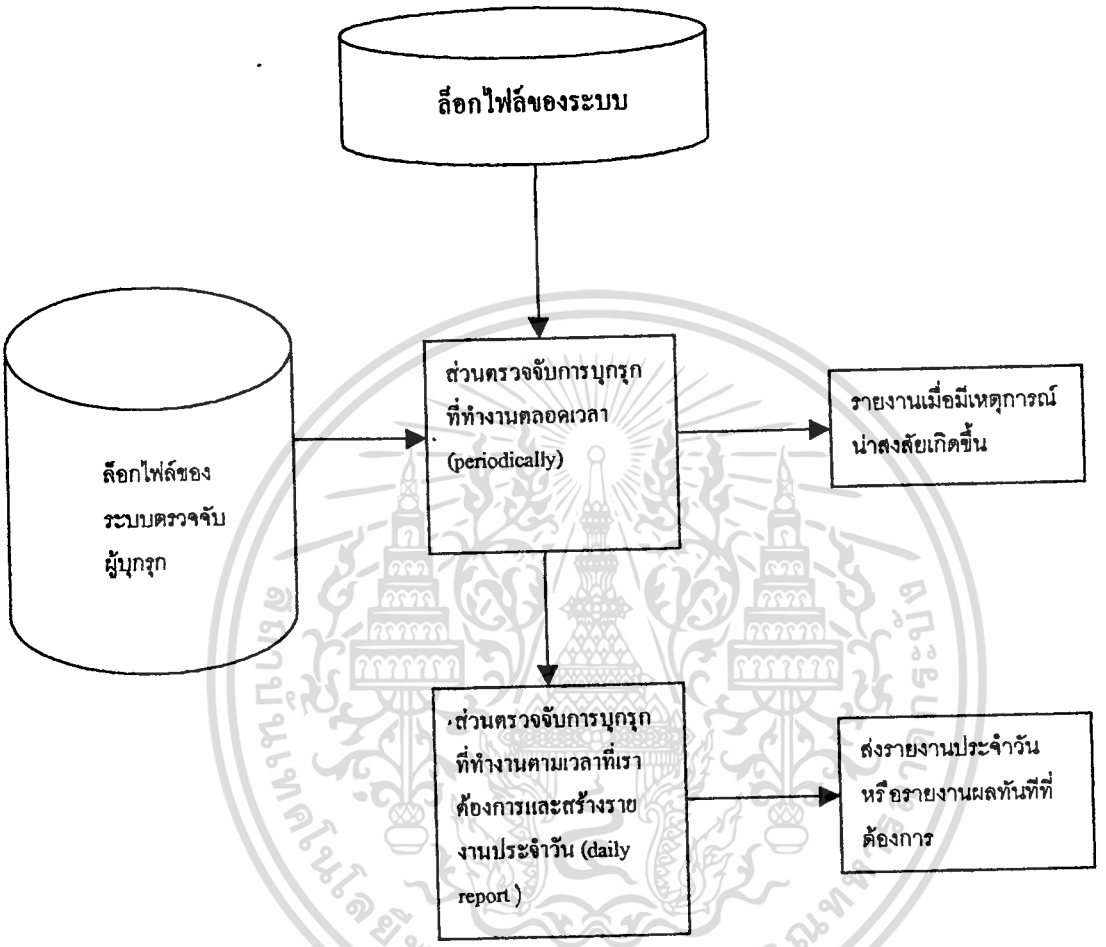
- ไฟล์ syslog.conf ของระบบมีการเปลี่ยนแปลง (file syslog.conf change)

ไฟล์ syslog.conf เป็นส่วนสำคัญในการกำหนดการทำงานของระบบจัดเก็บล็อกไฟล์ syslogd หากมีการเปลี่ยนแปลงอาจทำให้ ไม่มีการจัดเก็บล็อกไฟล์ของเหตุการณ์บางอย่างที่เกิดขึ้นกับระบบ

ระบบตรวจจับผู้บุกรุกจะตรวจสอบข้อมูลในไฟล์ syslog.conf และ ids-syslog.conf หากพบว่าข้อมูลในไฟล์ทั้งสองไม่ตรงกัน ระบบตรวจจับผู้บุกรุกจะมีการแจ้งเตือนทันทีว่ามีการเปลี่ยนแปลงแก้ไขไฟล์ syslog.confโดยไม่ได้รับอนุญาต

3.3.2 ส่วนตรวจจับการบุกรุกที่ทำงานตลอดเวลา (periodically)

เป็นส่วนที่ทำงานตลอดเวลาเป็นระยะๆ (ids.d) ตามระยะเวลาที่เรากำหนดในไฟล์ครอนแท็ป เมื่อระบบตรวจจับผู้บุกรุกตรวจพบความน่าสงสัยถึงระดับที่ได้กำหนดไว้จะทำการส่งอีเมล์ไปเตือนให้ผู้ดูแลระบบได้รับทราบ ซึ่งรูปแบบของพฤติกรรมที่ตรวจจับก็จะเหมือนกับส่วนตรวจจับการบุกรุกที่ทำงานตามเวลาที่เราต้องการและสร้างรายงานประจำวัน (ids)



รูปที่ 3.1 ระบบตรวจจับผู้บุกรุกระบบคอมพิวเตอร์

3.4 การเริ่มใช้งานระบบตรวจจับผู้บุกรุก

ระบบตรวจจับผู้บุกรุกที่สร้างขึ้นนี้อยู่บนวัตถุประสงค์ที่จะให้ใช้งานได้ง่ายที่สุด เพราะคำนึงถึงความสะดวกในการใช้งาน และไม่ต้องการที่จะเพิ่มเวลาศึกษาหรือเพิ่มงานให้ผู้ใช้มากขึ้นอีกในการนำระบบตรวจจับผู้บุกรุกไปใช้

ระบบตรวจจับผู้บุกรุกหรือโปรแกรมที่สร้างขึ้นนี้จึงออกแบบในอยู่ในไฟล์เดียวกันทั้งหมด และเมื่อเริ่มต้นการใช้งานหรือสั่งให้โปรแกรมทำงานครั้งแรก โปรแกรมจะทำการตรวจสอบระบบของเราว่ามีการติดตั้งระบบตรวจจับผู้บุกรุกนี้หรือยัง ถ้ายังก็จะทำการติดตั้งไฟล์ที่ใช้ในการทำงาน, สร้างไดเร็คทอรีสำหรับเก็บไฟล์ต่างๆที่สำคัญในการทำงาน และปรับตั้งไฟล์ที่ใช้ในการกำหนดการจัดเก็บล็อกแมสเซส (syslog.conf) รวมถึงการตั้งเวลาการทำงานของระบบตรวจจับผู้บุกรุกนี้ให้เอง (crontab) ทำให้เราไม่ต้องเสียเวลาในการติดตั้งระบบมากนัก เพียงแค่รู้ว่าไฟล์ไหนเป็นไฟล์ที่สั่งให้ระบบทำงานก็พอ หลังจากนั้นระบบตรวจจับผู้บุกรุกก็จะทำการส่งรายงานประจำวันให้เราทุกวัน ว่าระบบของเรามีเหตุการณ์อะไรเกิดขึ้นบ้าง และเมื่อมีเหตุการณ์น่าสงสัยเกิดขึ้นระบบตรวจจับผู้บุกรุกก็จะทำการส่งอีเมล์ไปให้ผู้ดูแลระบบทันที หรือถ้าเราต้องการรายงานเหตุการณ์ที่เกิดขึ้นในระบบเมื่อใดก็สามารถสั่งให้โปรแกรมทำการรายงานให้เราได้ทุกเวลาที่เราต้องการ

หลังจากที่สั่งให้ระบบตรวจจับผู้บุกรุกหรือโปรแกรมเริ่มการทำงาน โปรแกรมจะทำงานดังนี้ คือ

1. ตรวจสอบว่ามีการติดตั้งระบบตรวจจับผู้บุกรุกนี้หรือยัง
2. ทำการสร้างไดเร็คทอรี ids ขึ้นในไดเร็คทอรีของผู้ดูแลระบบเพื่อเก็บไฟล์ที่จำเป็นในการใช้ตรวจสอบการบุกรุก ซึ่งไฟล์ต่างๆที่จะสร้างขึ้นต่อไปจะถูกเก็บไว้ในไดเร็คทอรีนี้
3. ทำการสร้างไฟล์ ids-log0 เพื่อสำหรับเก็บล็อกไฟล์ไว้ใช้ในการตรวจสอบการเปลี่ยนแปลงล็อกไฟล์ของระบบโดยไม่ได้รับอนุญาต
4. ทำการสร้างไฟล์ ids-log(1-7) เพื่อสำหรับเก็บล็อกไฟล์ไว้ใช้ในการตรวจสอบย้อนหลังได้เป็นเวลา 1 สัปดาห์
5. ทำการปรับตั้งค่าเวลาการทำงานของระบบตรวจจับผู้บุกรุกในไฟล์ crontab โดยเพิ่มข้อความดังนี้

#ids setup

* 0-23/1 * * * root /root/ids/ids.d เพื่อเรียกให้ ids.d ทำงานทุกๆ 1 ชั่วโมง

55 23 * * * root /root/ids เพื่อเรียกให้ ids ทำงานที่เวลา 23.55 น. ของทุกวัน

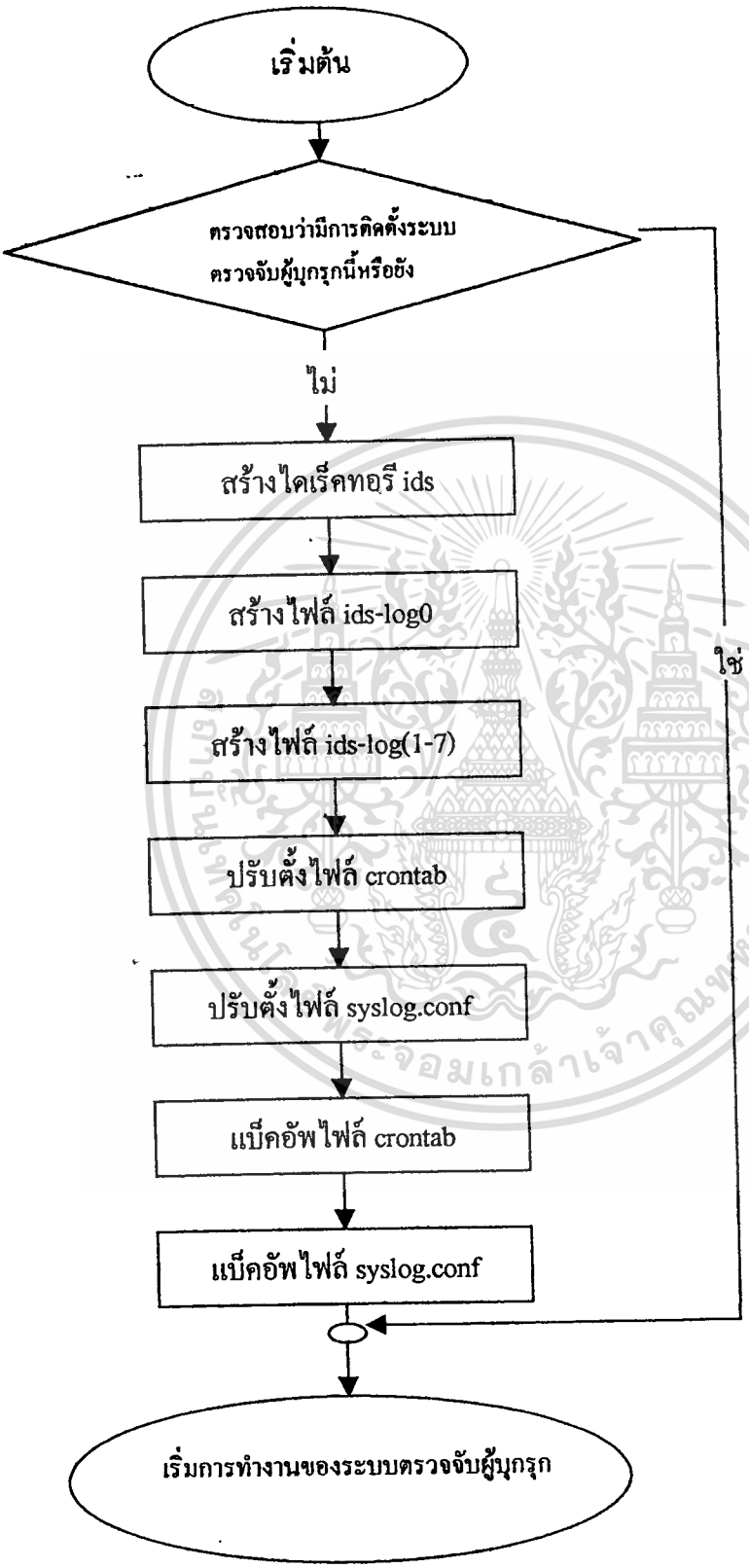
6. ทำการปรับตั้งค่าการทำงานของ syslogd ในไฟล์ syslog.conf โดยเพิ่มข้อความดังนี้

#ids setup

*.info;mail.none;authpriv.none;cron.none /root/ids/ids-log0

เพื่อบอกให้ syslogd ทำการสร้างล็อกไฟล์อีกชุดหนึ่ง ไปเก็บไว้ในล็อกไฟล์ของระบบตรวจจับผู้บุกรุก

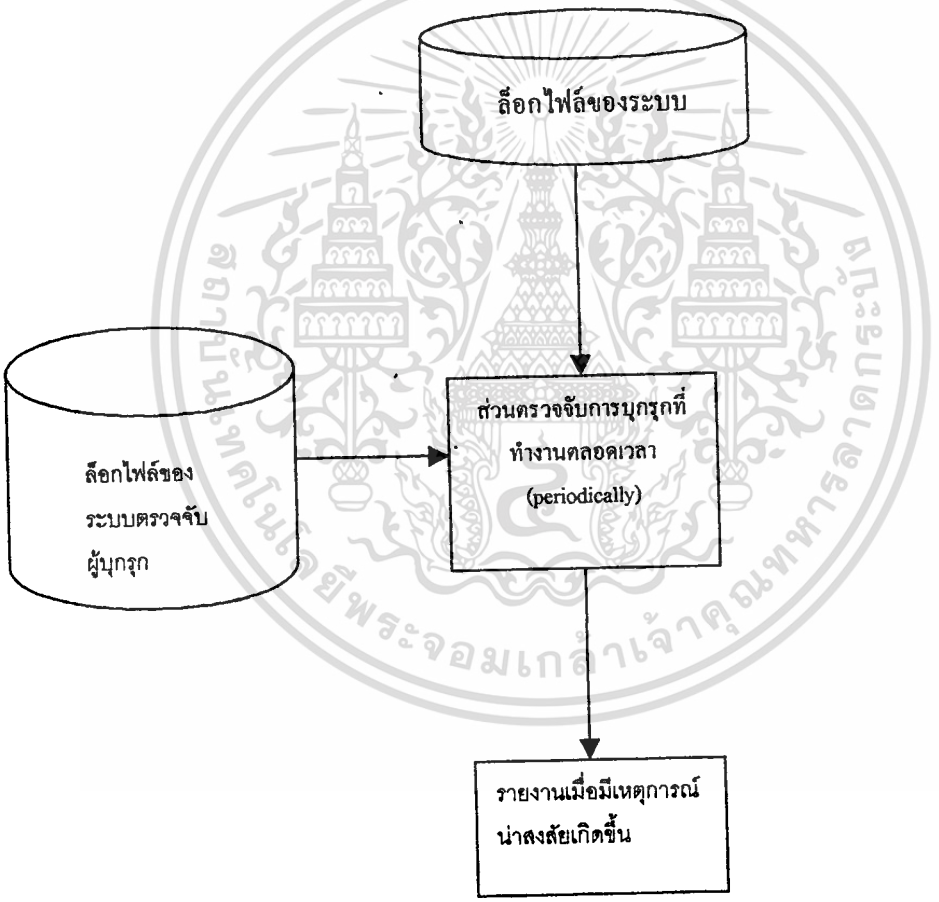
7. ทำการแบ็คอัพไฟล์ crontab ไว้ในไฟล์ ids-crontab เพื่อใช้ในการตรวจสอบการเปลี่ยนแปลงไฟล์ crontab ของระบบโดยไม่ได้รับอนุญาต
8. ทำการแบ็คอัพไฟล์ syslog.conf ไว้ในไฟล์ ids-syslog.conf เพื่อใช้ในการตรวจสอบการเปลี่ยนแปลงไฟล์ syslog.conf ของระบบโดยไม่ได้รับอนุญาต
9. เริ่มการทำงานของระบบตรวจจับผู้บุกรุก



รูปที่ 3-2 แผนภาพแสดงการทำงานของส่วนติดตั้งระบบตรวจจับผู้บุกรุก

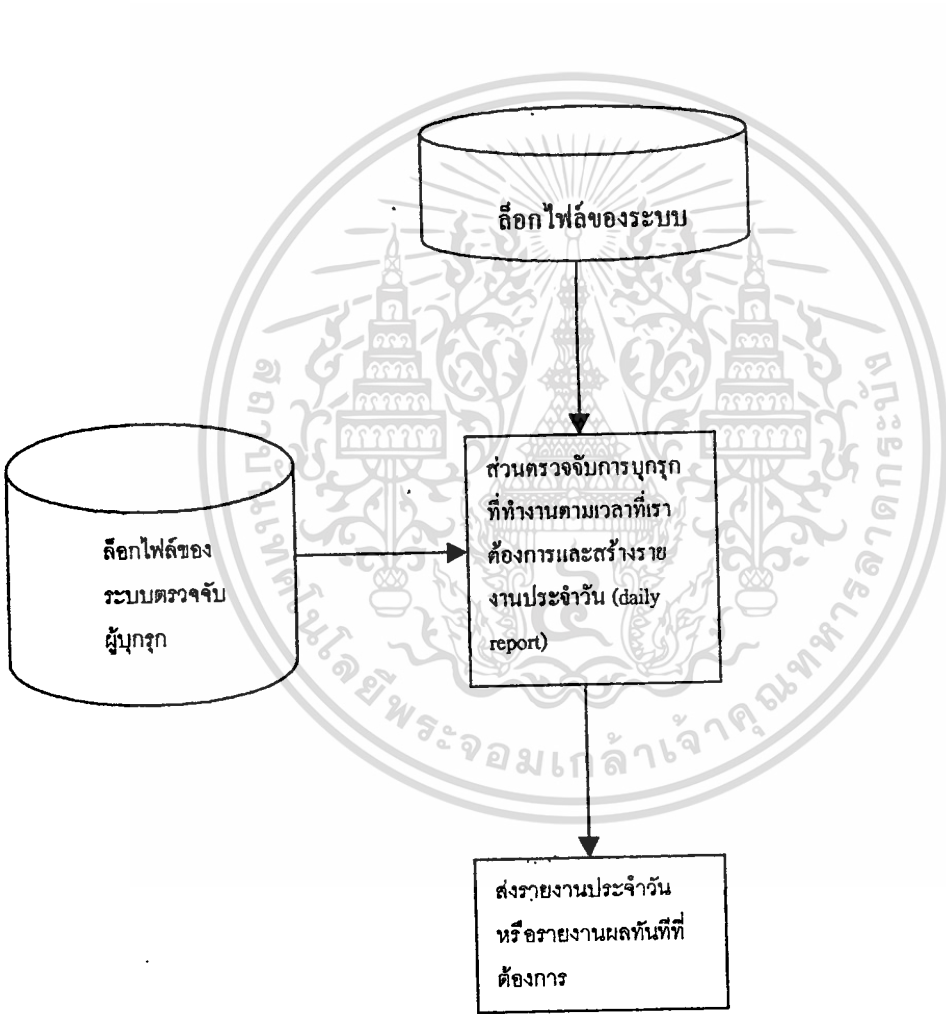
3.5 การทำงานของระบบตรวจจับผู้บุกรุก

1. ระบบตรวจจับผู้บุกรุกจะทำการตรวจสอบระบบของเราเป็นระยะๆ ว่ามีเหตุการณ์อะไรเกิดขึ้นบ้างจากล็อคไฟล์ของ syslogd และถ้ามีเหตุการณ์น่าสงสัยเกิดขึ้นถึงระดับที่กำหนดไว้ ระบบตรวจจับผู้บุกรุกก็จะทำการส่งอีเมล์ไปให้ผู้ดูแลระบบทันที



รูปที่ 3-3 แสดงการทำงานของส่วนตรวจจับการบุกรุกที่ทำงานตลอดเวลา (periodically)

2. ระบบตรวจจับผู้บุกรุกจะทำการส่งอีเมล์รายงานประจำวันให้ผู้ดูแลระบบทุกวัน ว่ามีเหตุการณ์อะไรเกิดขึ้นบ้างตลอดทั้งวันนั้น หรือถ้าเราต้องการรายงานเหตุการณ์ที่เกิดขึ้นในระบบเมื่อใดก็สามารถสั่งให้โปรแกรมทำการรายงานได้ทุกเวลาที่เราต้องการ



รูปที่ 3-4 แสดงการทำงานของส่วนตรวจจับการบุกรุกที่ทำงานตามเวลาที่เราต้องการและสร้างรายงานประจำวัน (daily report)

3. เมื่อระบบตรวจจับผู้บุกรุกทำการส่งอีเมล์รายงานประจำวันให้ผู้ดูแลระบบแล้ว จะทำการบันทึกล็อกไฟล์ของระบบลงในล็อกไฟล์ของระบบตรวจจับผู้บุกรุก ซึ่งสามารถตรวจสอบย้อนหลังได้ 7 วัน หรือ 1 สัปดาห์ (ids-log[1-7]) และจะทำการลบล็อกไฟล์ของระบบออกทั้งหมด เพื่อป้องกันผู้บุกรุกใช้ข้อมูลในล็อกไฟล์เป็นประโยชน์ในการบุกรุกระบบ
4. ถ้ามีการเปลี่ยนแปลงไฟล์ที่สำคัญของระบบ โดยตัวผู้ดูแลระบบเอง ต้องทำการลบ directory ids ออก แล้วทำการ run โปรแกรมเพื่อให้ทำการติดตั้งระบบตรวจจับผู้บุกรุกใหม่ มิฉะนั้นระบบตรวจจับผู้บุกรุกจะรายงานว่ามีความเปลี่ยนแปลงเกิดขึ้นกับไฟล์นั้นๆ

บทที่ 4

การทดสอบการทำงานของระบบตรวจจับผู้บุกรุก

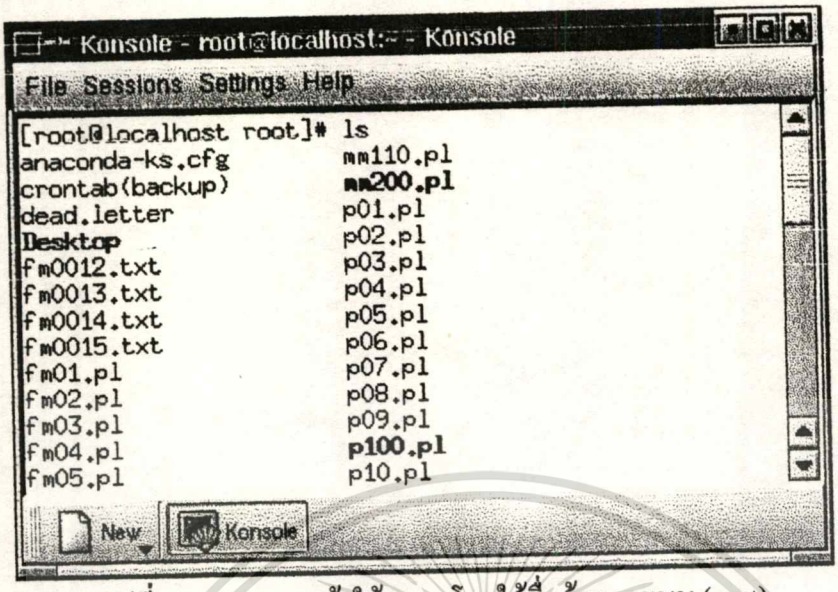
หลังจากที่ได้ทำการติดตั้งระบบตรวจจับผู้บุกรุกเรียบร้อยแล้ว ต่อไปจะทำการทดสอบการทำงานตรวจจับผู้บุกรุกตามรูปแบบที่กำหนดไว้ ซึ่งจะทำการทดลองทีละพฤติกรรมที่กำหนดให้มีการตรวจจับ จากนั้นจะทำการสั่งให้ระบบตรวจจับผู้บุกรุกทำงานโดยการ run โปรแกรม ids ในสถานะของผู้ดูแลระบบ และจะแสดงตัวอย่างรายงานประจำวันที่จะส่งไปให้ผู้ดูแลระบบทุกวันอีกครั้งหนึ่ง ซึ่งรายงานประจำวันนี้จะรวมผลการตรวจจับทั้งหมด เพื่อให้เห็นภาพการทำงานของระบบ

4.1 ผลการทดสอบ

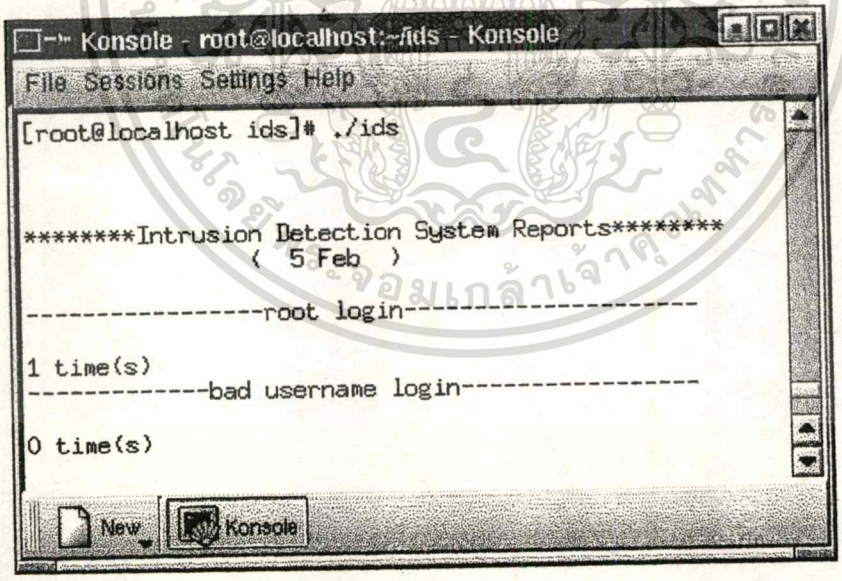
การทดสอบการตรวจจับพฤติกรรมการบุกรุกระบบจะเรียงลำดับตามรูปแบบของพฤติกรรมที่ออกแบบให้มีการตรวจจับ คือ

- root login
- bad username login
- bad password login
- su command failure
- su command success
- user change
- file passwd change
- log file change
- file crontab change
- file syslog.conf change

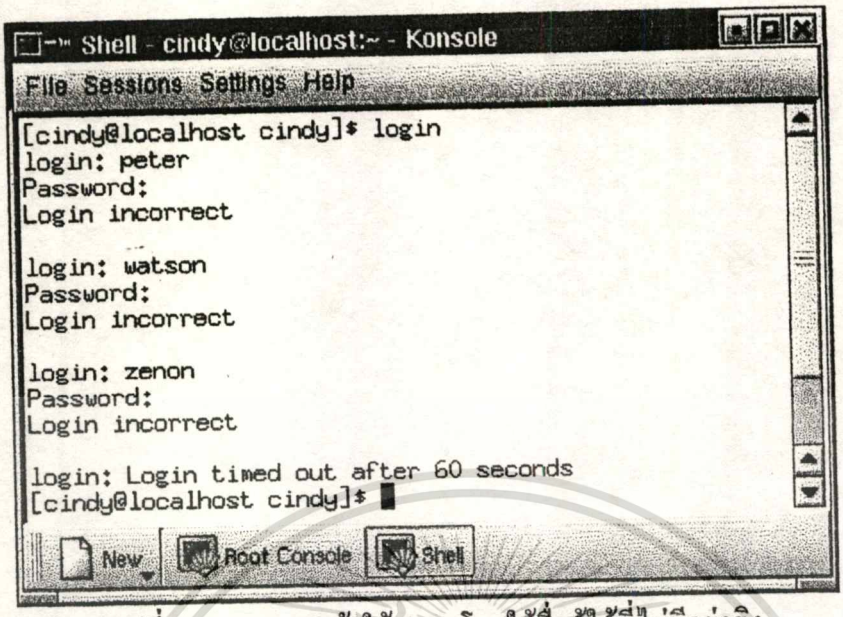
การทดสอบและผลการทดสอบตรวจจับพฤติกรรมการบุกรุกจะแสดงเป็นภาพ ดังนี้



รูปที่ 4-1 แสดงการเข้าใช้ระบบโดยใช้ชื่อผู้ดูแลระบบ (root)

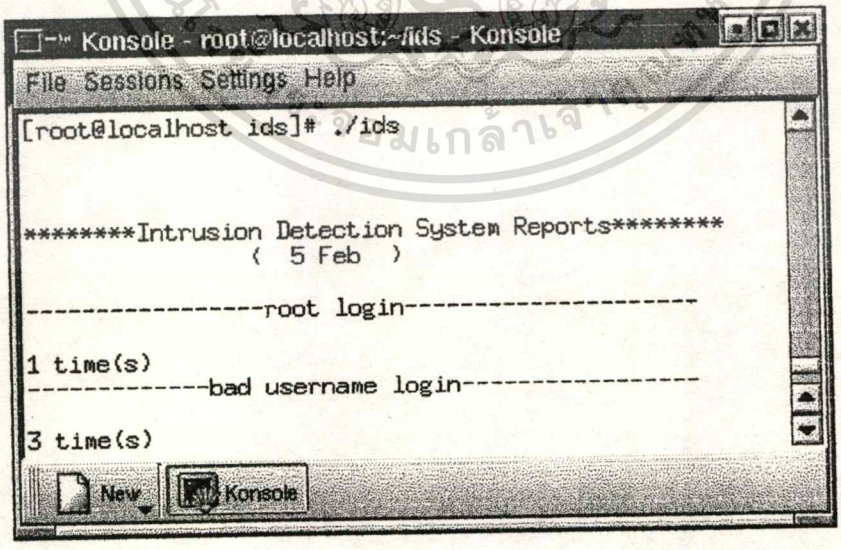


รูปที่ 4-2 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนการเข้าใช้ระบบ โดยใช้ชื่อผู้ดูแลระบบ (root)



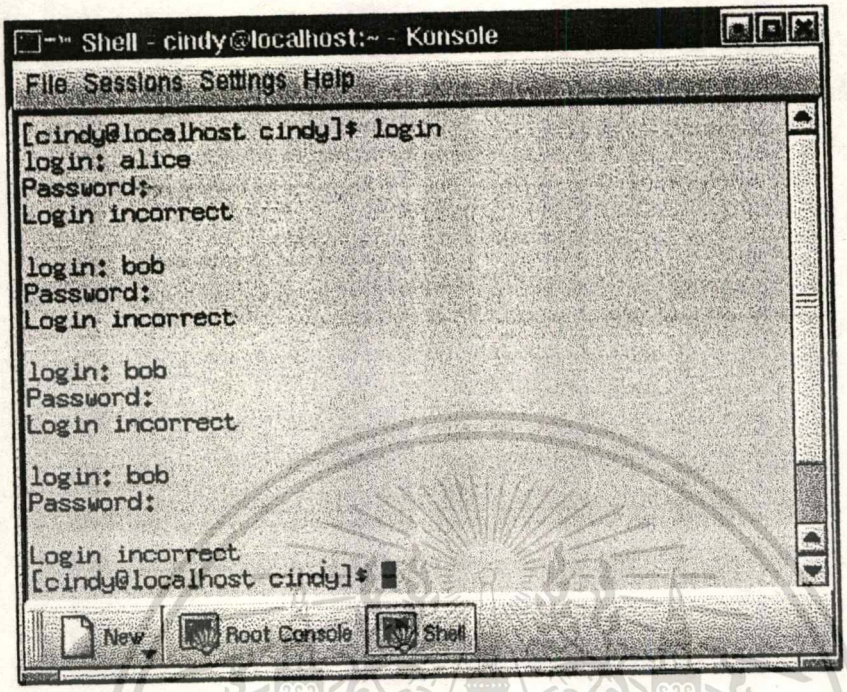
รูปที่ 4-3 แสดงการเข้าใช้ระบบโดยใช้ชื่อผู้ใช้ที่ไม่มีอยู่จริง

(bad username login)

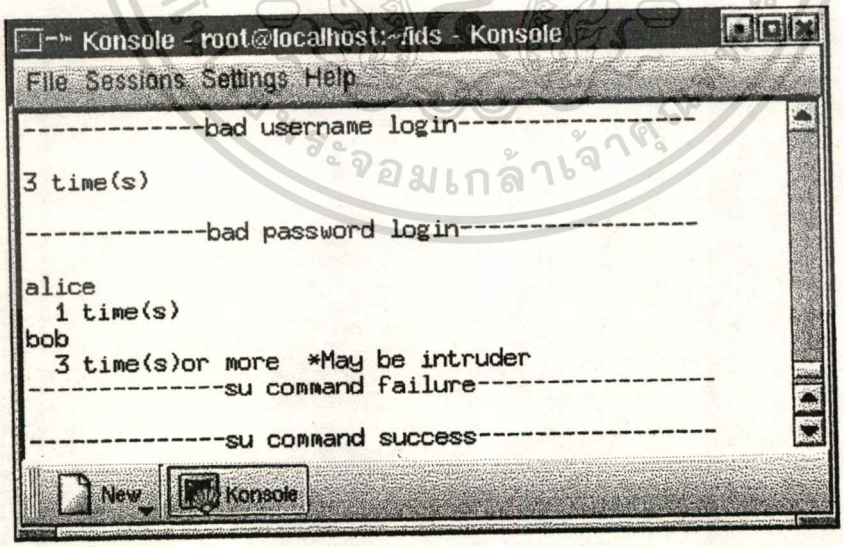


รูปที่ 4-4 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนการเข้าใช้ระบบ

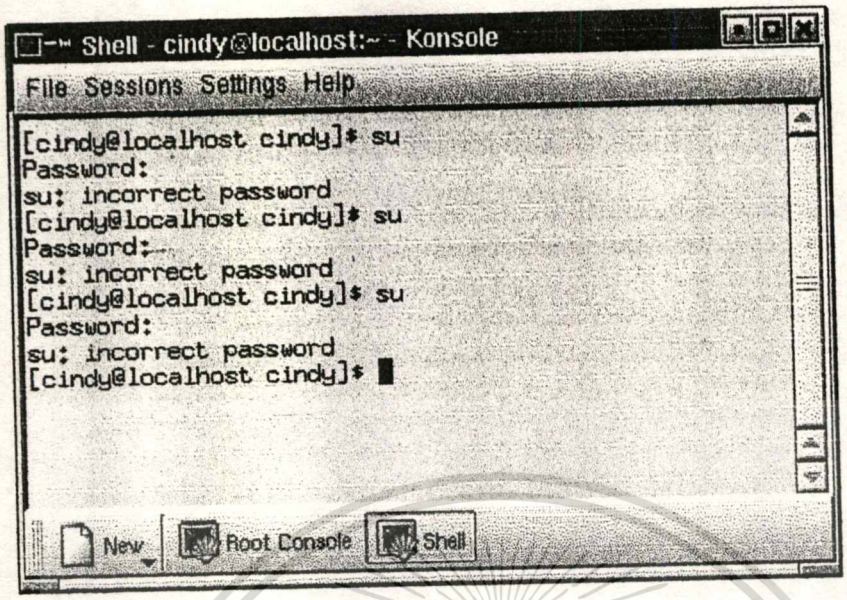
โดยใช้ชื่อผู้ใช้ที่ไม่มีอยู่จริง (bad username login)



รูปที่ 4-5 แสดงการเข้าใช้ระบบโดยใช้ชื่อผู้ใช้ที่มีอยู่จริง แต่ใส่รหัสผ่านไม่ถูกต้อง (bad password login)

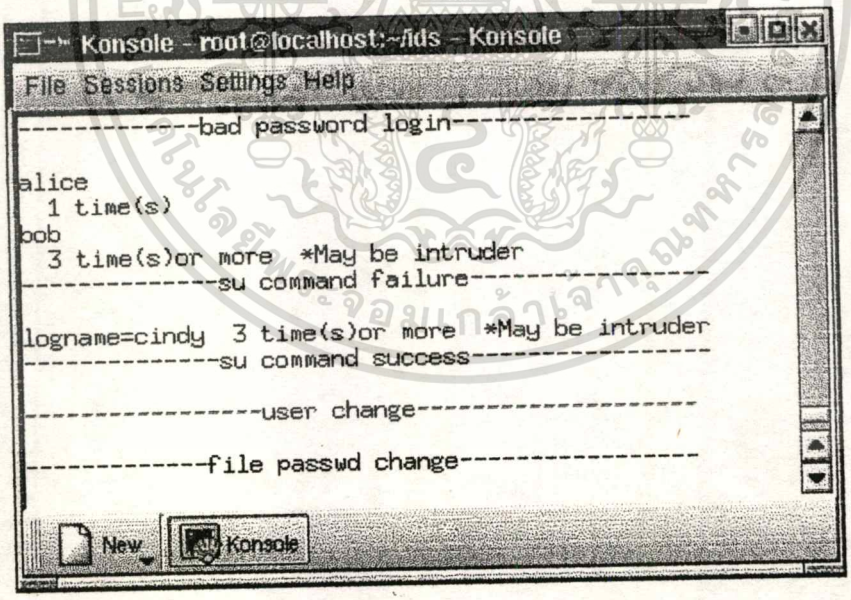


รูปที่ 4-6 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนการเข้าใช้ระบบโดยใช้ชื่อผู้ใช้ที่มีอยู่จริง แต่ใส่รหัสผ่านไม่ถูกต้อง (bad password login)



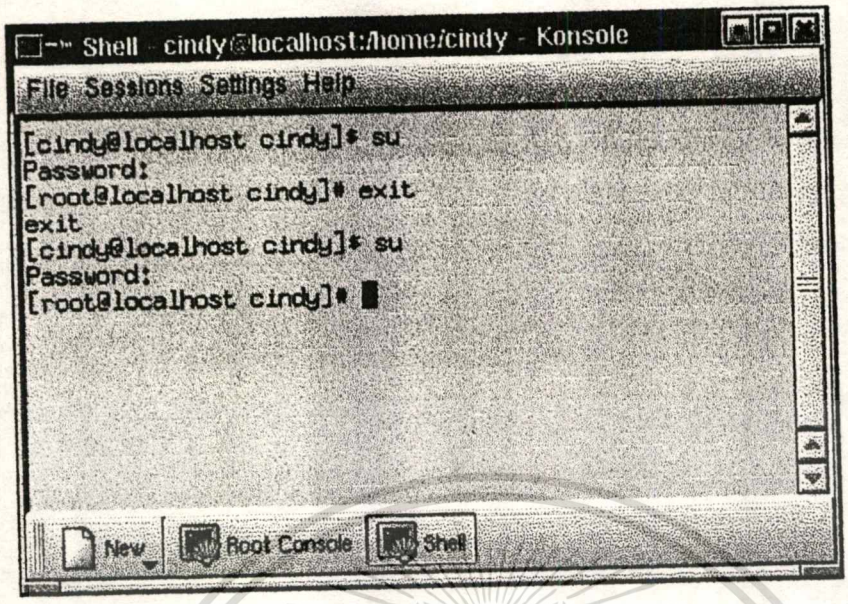
รูปที่ 4-7 แสดงการพยายามเปลี่ยนสิทธิเป็นผู้ดูแลระบบแต่ไม่สำเร็จ

(su command failure)



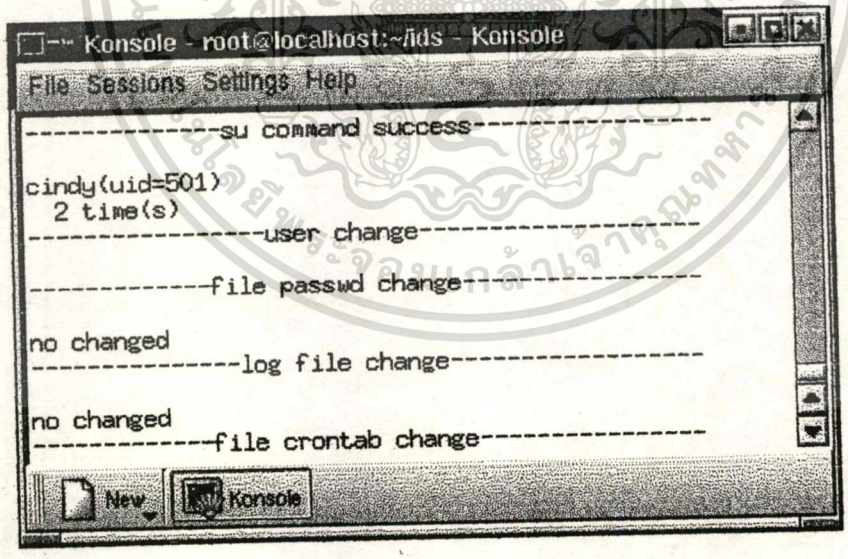
รูปที่ 4-8 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนของ
การพยายามเปลี่ยนสิทธิเป็นผู้ดูแลระบบแต่ไม่สำเร็จ

(su command failure)



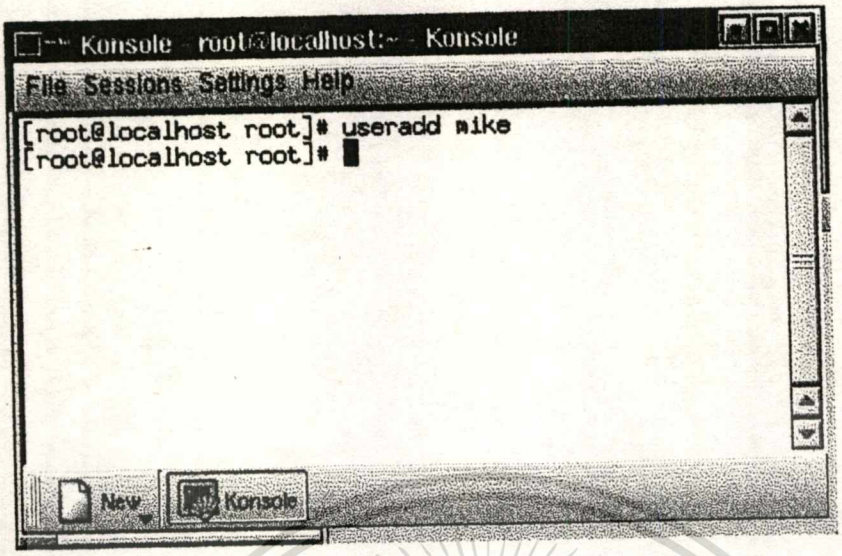
รูปที่ 4-9 แสดงการเปลี่ยนสิทธิเป็นผู้ดูแลระบบสำเร็จ

(su command success)



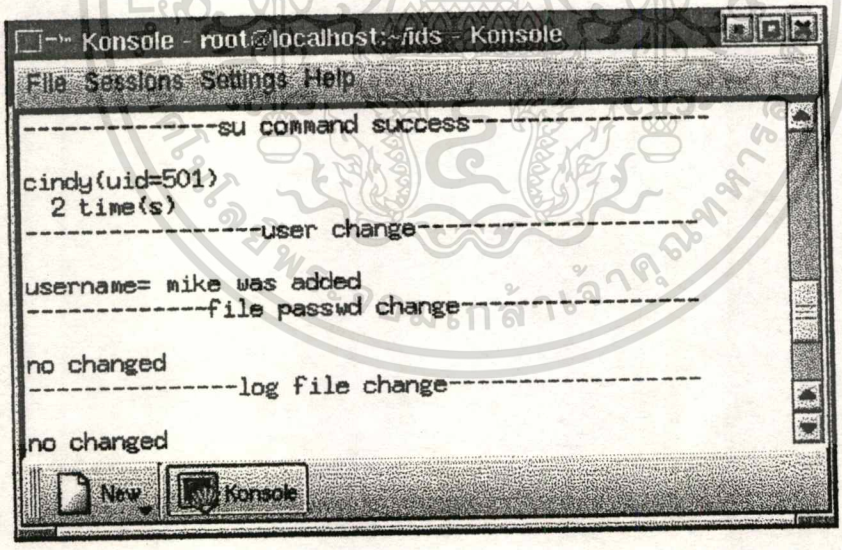
รูปที่ 4-10 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนของ

การเปลี่ยนสิทธิเป็นผู้ดูแลระบบสำเร็จ (su command success)



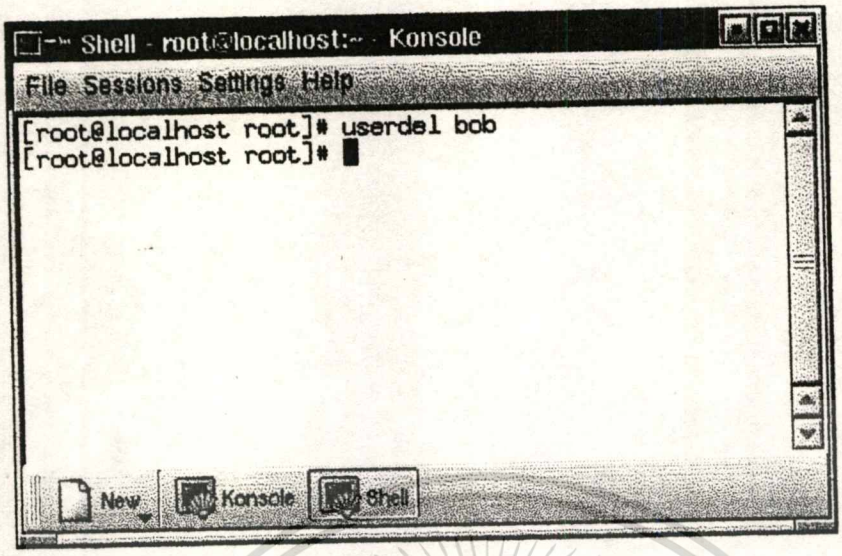
รูปที่ 4-11 แสดงการเปลี่ยนแปลงผู้ใช้ระบบ โดยการเพิ่มผู้ใช้ระบบ

(user change [add])



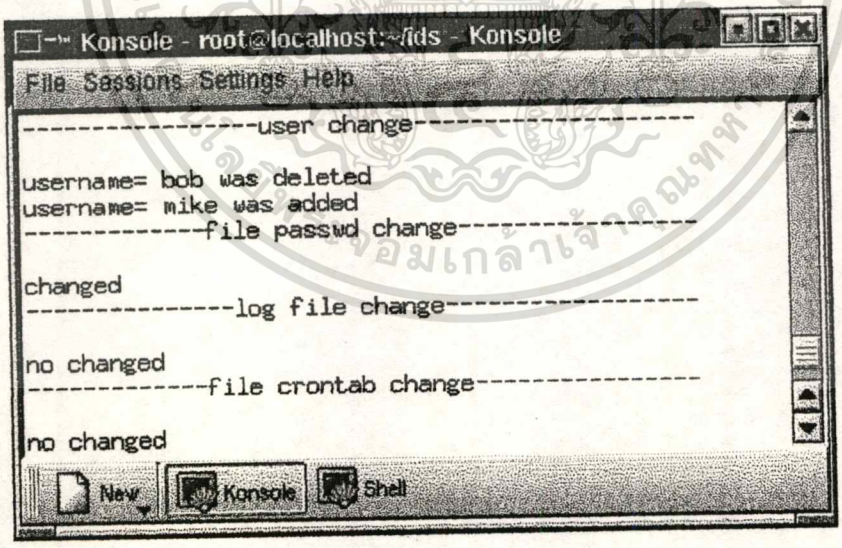
รูปที่ 4-12 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนของ
การเปลี่ยนแปลงผู้ใช้ระบบ โดยการเพิ่มผู้ใช้ระบบ

(user change [add])



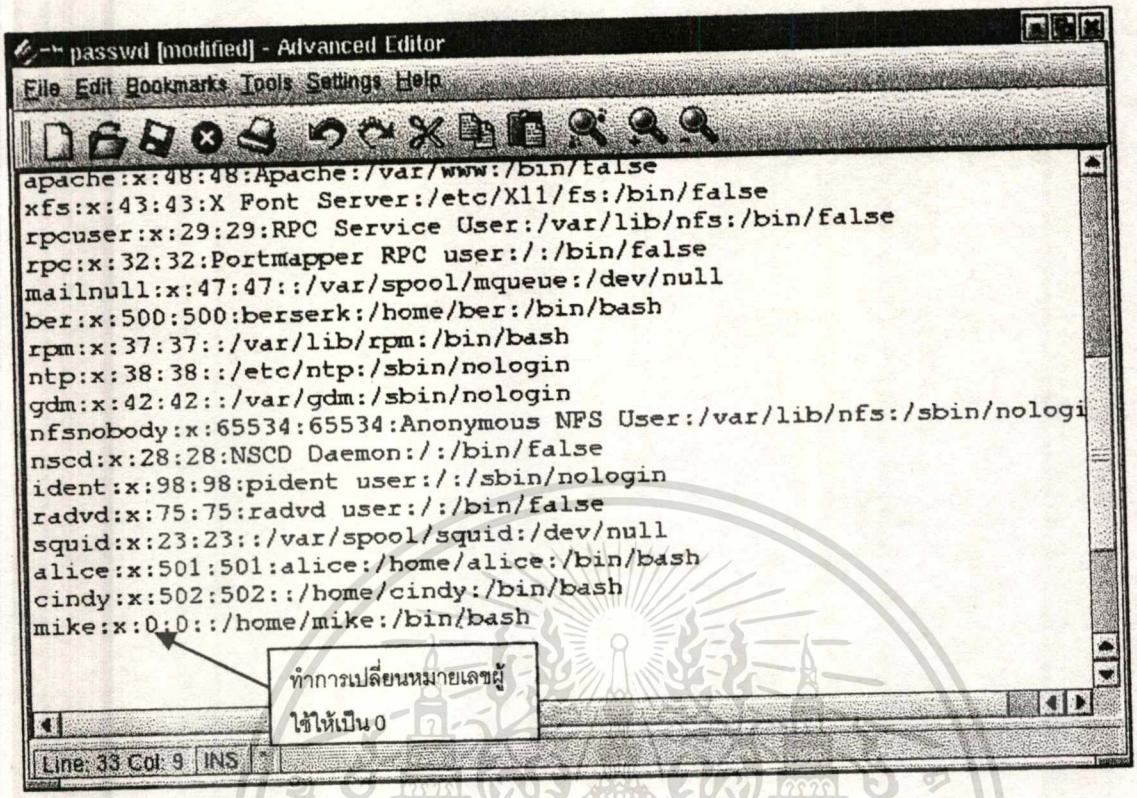
รูปที่ 4-13 แสดงการเปลี่ยนแปลงผู้ใช้ระบบ โดยการลบผู้ใช้ระบบ

(user change [delete])

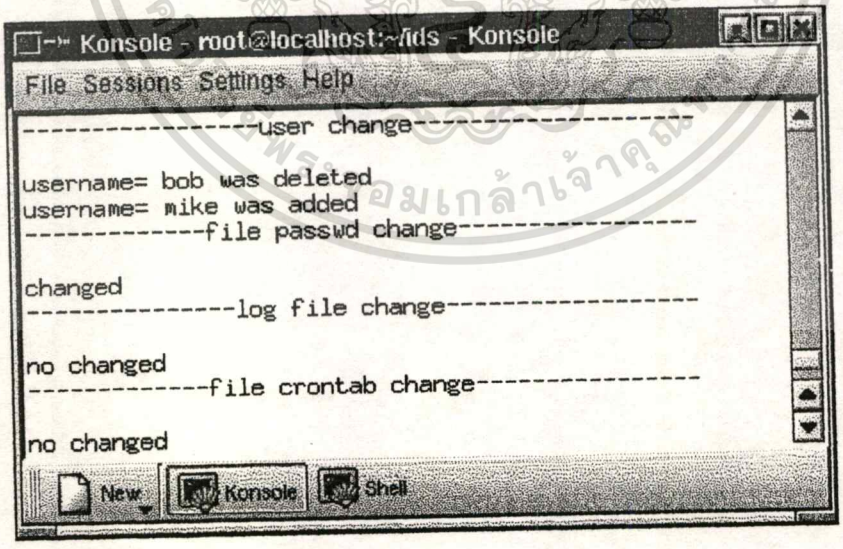


รูปที่ 4-14 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนของ
การเปลี่ยนแปลงผู้ใช้ระบบ โดยการลบผู้ใช้ระบบ

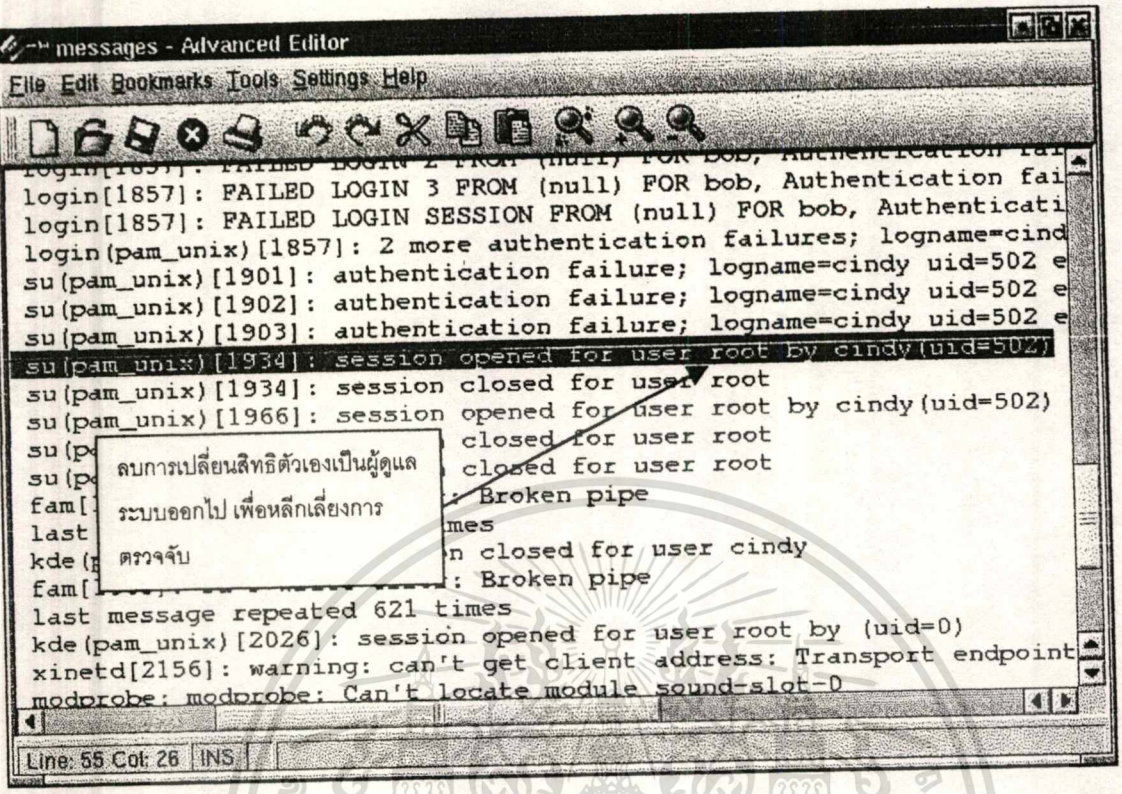
(user change [delete])



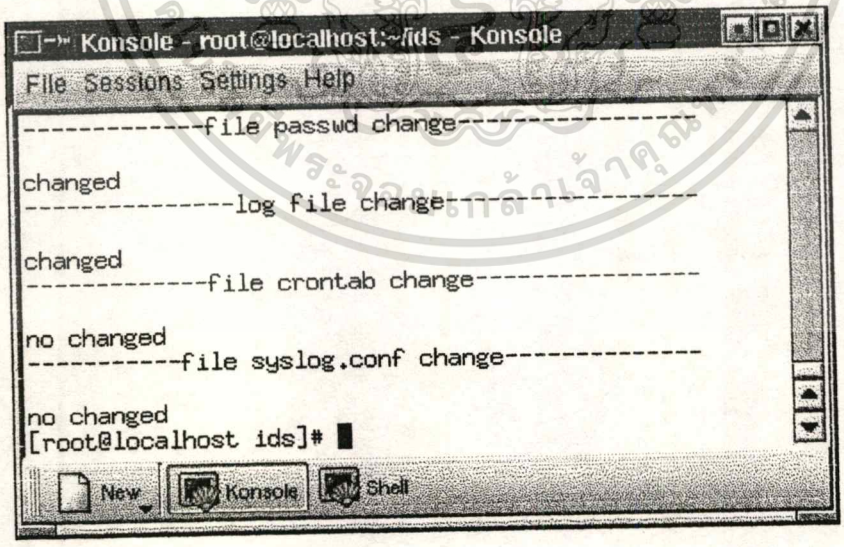
รูปที่ 4-15 แสดงการเปลี่ยนแปลงไฟล์ passwd โดยไม่ได้รับอนุญาต (file passwd change)



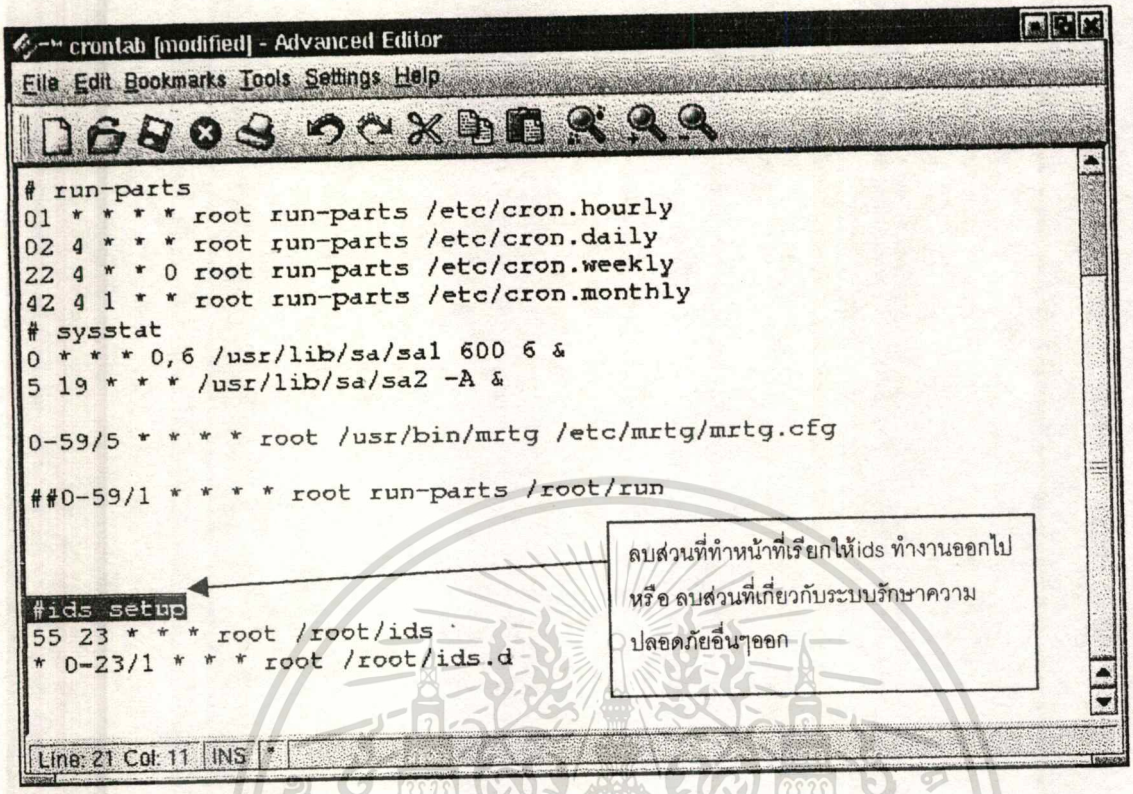
รูปที่ 4-16 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนของการเปลี่ยนแปลงไฟล์ passwd โดยไม่ได้รับอนุญาต (file passwd change)



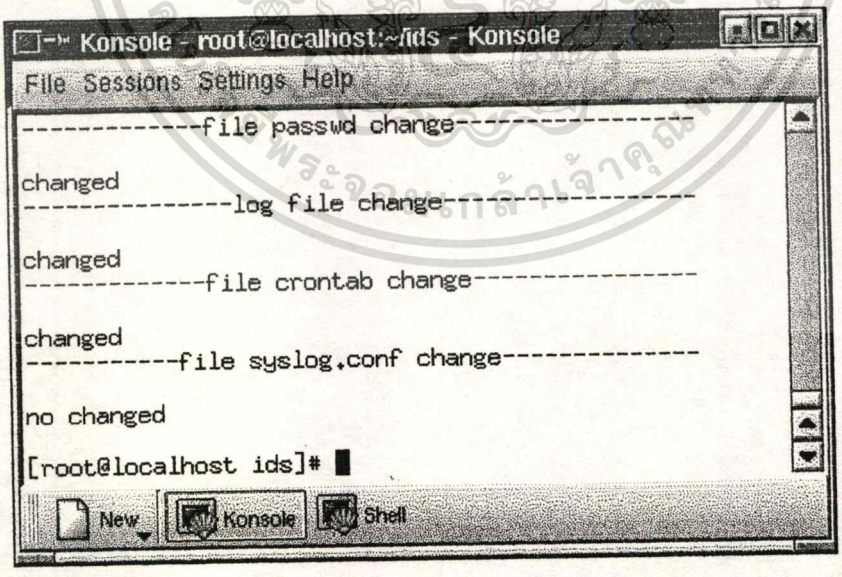
รูปที่ 4-17 แสดงการเปลี่ยนแปลงล็อกไฟล์ โดยไม่ได้รับอนุญาต (log file change)



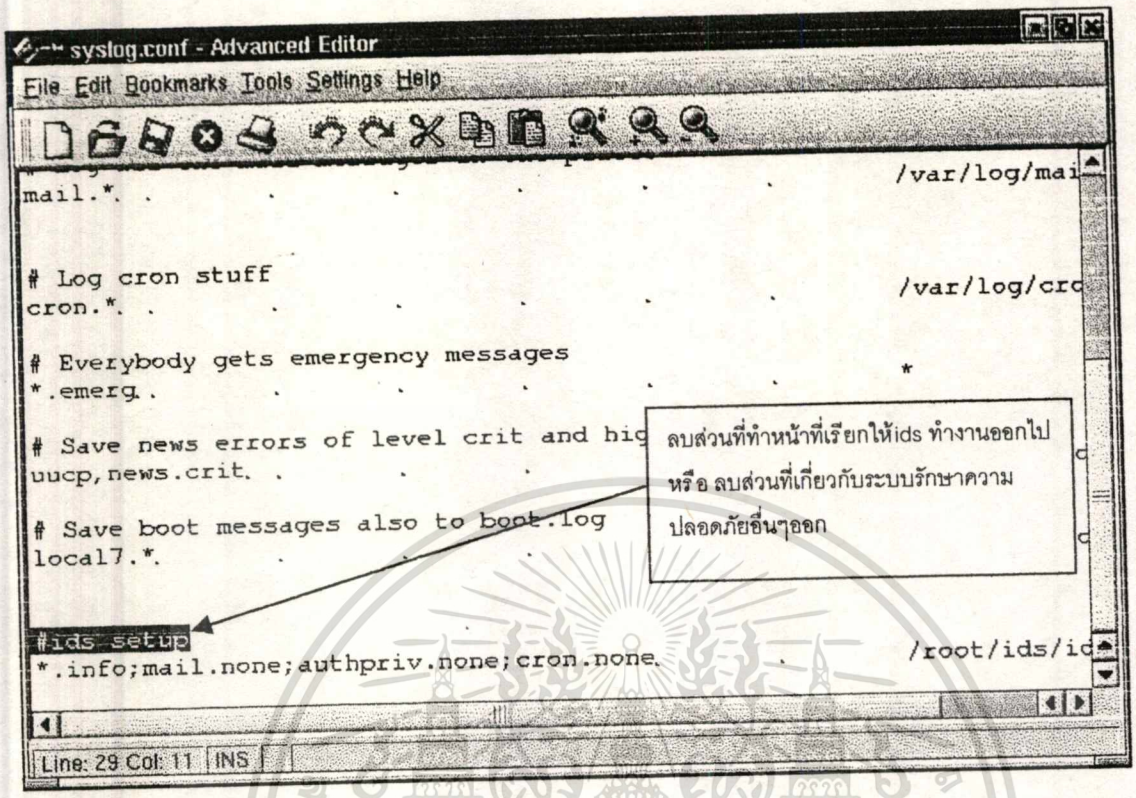
รูปที่ 4-18 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนของ
การเปลี่ยนแปลงล็อกไฟล์ โดยไม่ได้รับอนุญาต (log file change)



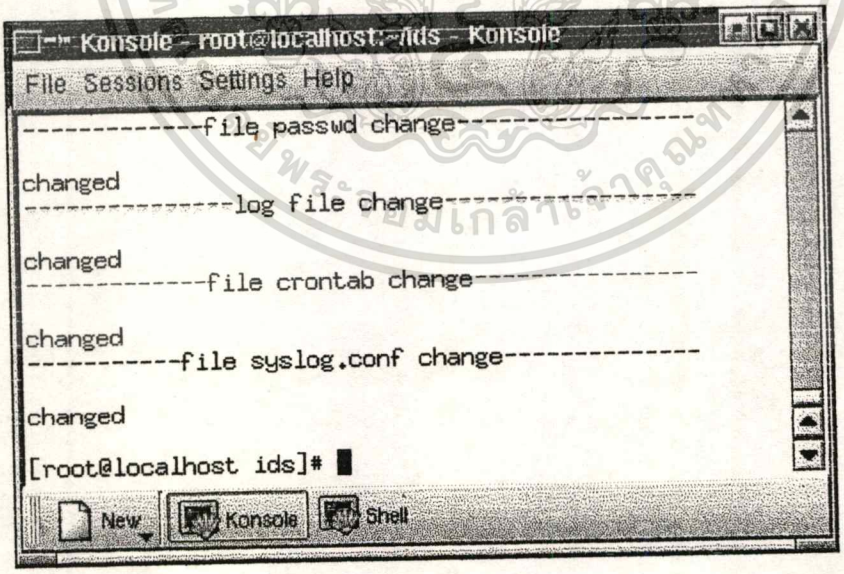
รูปที่ 4-19 แสดงการเปลี่ยนแปลงไฟล์ crontab โดยไม่ได้รับอนุญาต (file crontab change)



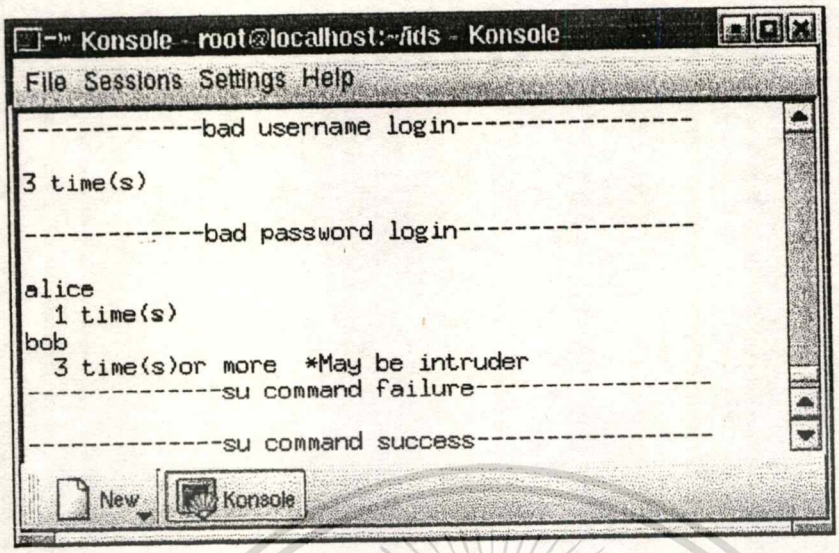
รูปที่ 4-20 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนของ
การเปลี่ยนแปลงไฟล์ crontab โดยไม่ได้รับอนุญาต (file crontab change)



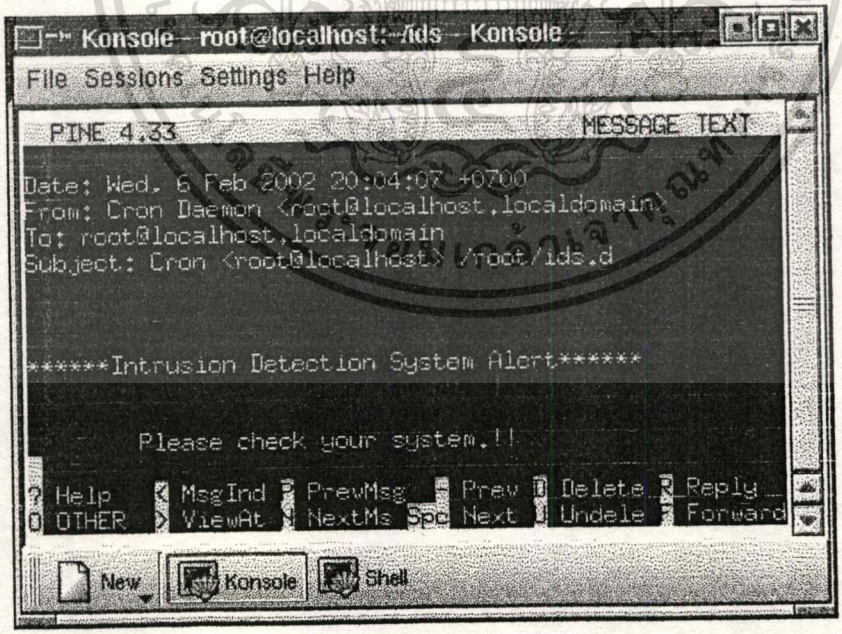
รูปที่ 4-21 แสดงการเปลี่ยนแปลงไฟล์ syslog.conf โดยไม่ได้รับอนุญาต (file syslog.conf change)



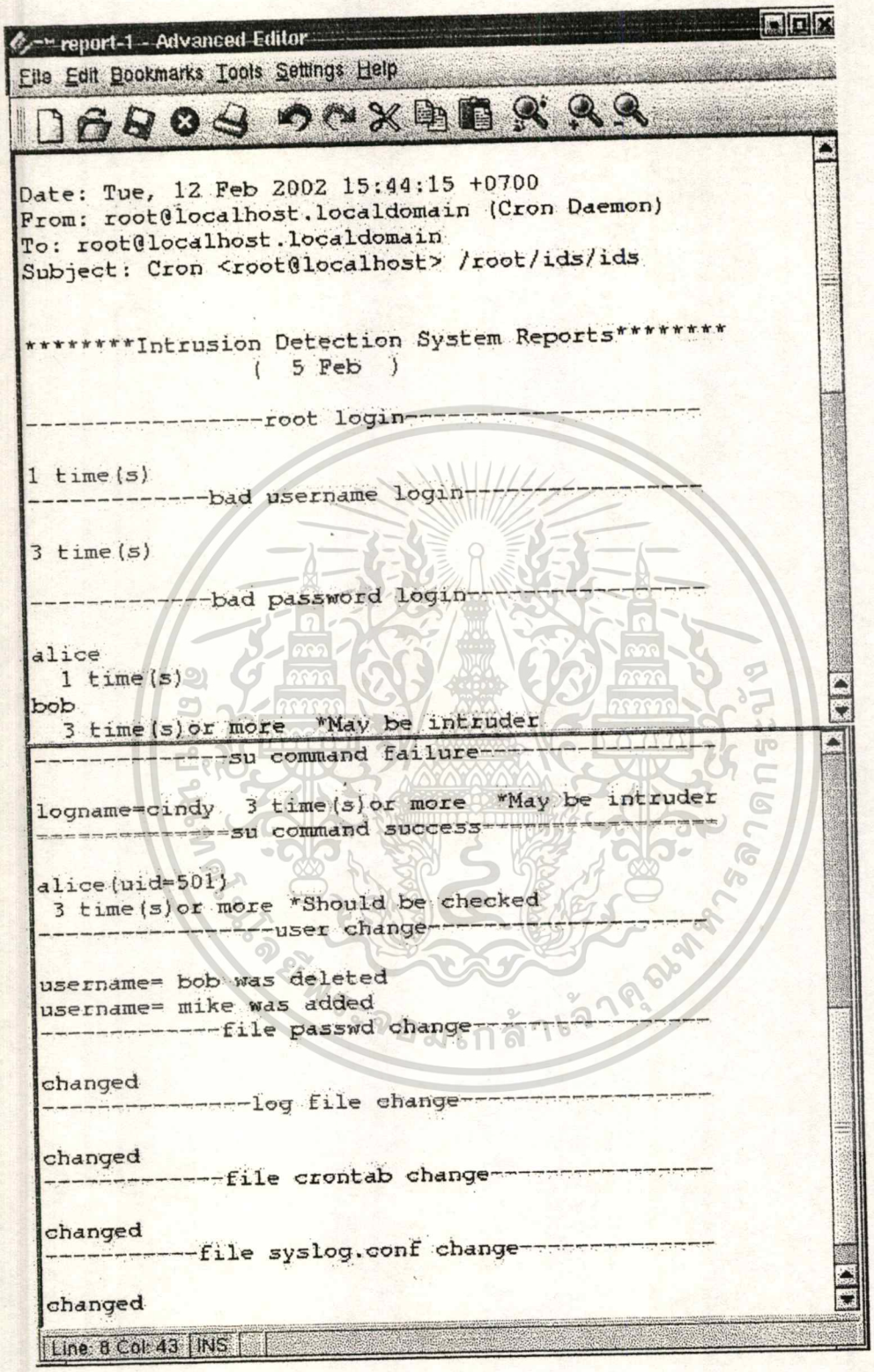
รูปที่ 4-22 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนของ
การเปลี่ยนแปลงไฟล์ syslog.conf โดยไม่ได้รับอนุญาต
(file syslog.conf change)



รูปที่ 4-23 แสดงเมื่อมีเหตุการณ์ที่น่าสงสัยเกิดขึ้น และอาจเป็นผู้บุกรุกระบบ



รูปที่ 4-24 แสดงผลการรายงานของระบบตรวจจับผู้บุกรุกในส่วนที่ทำงานตลอดเวลา (ids.d) โดยจะส่งอีเมล์แจ้งเตือนไปยังผู้ดูแลระบบทันที



รูปที่ 4-25 ตัวอย่างรายงานประจำวันของระบบตรวจจับผู้บุกรุก ที่จะส่งเมล์ให้ผู้ดูแลระบบทุกวัน

บทที่ 5

สรุปและแนวทางในการพัฒนาระบบตรวจจับผู้บุกรุกในอนาคต

5.1 สรุป

การเริ่มใช้งานระบบตรวจจับผู้บุกรุกที่สร้างขึ้นสามารถทำได้ง่าย เพราะออกแบบคำนึงถึงความสะดวกในการใช้งาน และไม่ต้องการที่จะเพิ่มเวลาศึกษาหรือเพิ่มงานให้ผู้ใช้มากขึ้นอีกในการนำระบบตรวจจับผู้บุกรุกไปใช้ ซึ่งเป็นจุดมุ่งหมายหนึ่งในการสร้างระบบตรวจจับผู้บุกรุกนี้ขึ้น เพราะระบบตรวจจับผู้บุกรุกอื่นๆในปัจจุบัน บางครั้งใช้งานยากและต้องใช้เวลาศึกษานานในการนำไปใช้

ระบบตรวจจับผู้บุกรุกที่สร้างขึ้นนี้มีความสามารถในการตรวจจับการบุกรุกตามรูปแบบที่กำหนดไว้ ซึ่งบางรูปแบบระบบอื่นๆไม่สามารถตรวจจับได้

ระบบตรวจจับผู้บุกรุกที่สร้างขึ้นจะอาศัยการจัดเก็บล็อกไฟล์ของ syslogd ที่เป็นระบบจัดเก็บล็อกไฟล์ที่ระบบส่วนใหญ่จะมีอยู่แล้ว เพราะต้องการให้ทุกระบบสามารถนำระบบตรวจจับผู้บุกรุกที่สร้างขึ้นนี้ไปใช้ได้โดยไม่ต้องติดตั้งส่วนอื่นๆเพิ่มอีก ซึ่งจะให้ความปลอดภัยได้ในระดับหนึ่งเท่านั้น

การนำระบบตรวจจับผู้บุกรุกไปใช้ควรใช้ร่วมกับระบบตรวจจับผู้บุกรุกอื่นๆหลายๆระบบ ถ้าต้องการให้ระบบคอมพิวเตอร์ที่ใช้งานอยู่ปลอดภัยที่สุด เพราะไม่มีระบบตรวจจับผู้บุกรุกใดที่สามารถตรวจจับการบุกรุกได้ทุกรูปแบบ และผู้บุกรุกก็คิดวิธีการบุกรุกใหม่ๆขึ้นมาทุกวัน

5.2 แนวทางในการพัฒนาระบบตรวจจับผู้บุกรุกในอนาคต

- เพิ่มรูปแบบการบุกรุกระบบที่สามารถตรวจจับได้ให้มากขึ้น
- ศึกษารูปแบบการบุกรุกใหม่ๆที่ผู้บุกรุกใช้แล้วนำมาพัฒนาระบบตรวจจับผู้บุกรุกให้สามารถตรวจจับรูปแบบการบุกรุกนั้นๆได้
- พัฒนาให้สามารถป้องกันระบบหรือยกเลิกการเชื่อมต่อกับระบบของผู้ที่ถูกสงสัยว่าจะเป็นผู้บุกรุกได้โดยอัตโนมัติ

บรรณานุกรม

Garfinkel, S and E, Spafford. 1997. *Web Security and Commerce.*:O'Reilly & Associates.

Karen, A Forcht. 1994. *Computer Security Management.*:boyd & fraser publishing company.

Mark , F Komarinki. 1996. *Linux Companion The Essential Guide For Users And System Administrators.*:Prentice Hall.

Michael, Beck and Harald, Bohme. 1996. *LINUX Kernel Internals.*:Addison-Wesley.

Randal ,L Schwartz. 1998. *Learning Perl .*:O'Reilly & Associates.

Scott, Mann and Ellen, L Mitchell. 2000. *Linux System Security.*:Prentice-Hall.

Simson, Garfinkel and Gene, Spafford. 1996. *Practical Unix & Internet Security.*:O'Reilly & Associates.

ประวัติผู้เขียน

ชื่อ-นามสกุล	นายบารมี อุดมนิธิกุล
ประวัติการศึกษา	วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมการก่อสร้าง สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ตำแหน่งการทำงาน	วิศวกรที่ปรึกษาโครงการ
สถานที่ทำงาน	บริษัทออกแบบและควบคุมงานก่อสร้างเอกชน
ระยะเวลาที่ทำงาน	ปี 2541 - ปัจจุบัน