บทที่ 5

การออกแบบและพัฒนาชุดโปรแกรมตรวจสอบความมั่นคง

5.1 การออกแบบชุดโปรแกรมตรวจสอบความมั่นคง

ในบทนี้จะกล่าวถึงโปรแกรมที่ใช้ในการตรวจสอบความมั่นคงของระบบยูนิกซ์ โดยโปรแกรมที่ใช้ในการตรวจสอบความมั่นคงนี้เขียนด้วยภาษาเบอร์นเชลล์

เหตุผลที่ใช้ภาษาเบอร์นเชลล์ในการเขียนโปรแกรมมีดังนี้

1. ระบบยูนิกซ์ทุกเครื่องจะต้องมีเบอร์นเชลล์มาด้วย ซึ่งต่างจาก ซีเชลล์ (C Shell)และ คอร์นเชลล์ (Korn Shell) ซึ่งมากับยูนิกซ์บางรุ่นเท่านั้น ดังนั้นการใช้เบอร์นเชลล์ จะทำให้โปรแกรมมีความเข้ากันได้ (portable) กับยูนิกซ์ทุกรุ่น

2. เราสามารถที่นำโปรแกรมอรรถประโยชน์ (utility program) ซึ่งมีอยู่อย่างมากมายในระบบยูนิกซ์ มาประกอบกับโปรแกรมเบอร์นเชลล์ ทำให้การเขียนโปรแกรมสะดวกและรวดเร็วกว่า การใช้ภาษาโปรแกรมแบบอื่น เช่น ภาษา ซี (C language)

ในการออกแบบชุดของโปรแกรมตรวจสอบความมั่นคงได้คำนึงถึงสิ่งต่อไปนี้

1. มีโปรแกรมหลักที่ทำหน้าที่ค้นหาชื่อเต็ม (absolute pathname) ของคำสั่งต่างๆ โดยเก็บผลลัพธ์ที่ได้ไว้ในแฟ้มหนึ่ง ซึ่งทุกโปรแกรมจะเรียกใช้แฟ้มนี้ ทั้งนี้เพราะว่าระบบยูนิกซ์ต่างรุ่นกัน จะมีการเก็บบางคำสั่งไว้ที่ไดเรคทอรีต่างกัน การที่กำหนดให้มีแฟ้มที่เก็บชื่อคำสั่งเต็มและทุกโปรแกรมเรียกใช้แฟ้มนี้ เพื่อความสะดวกในการนำโปรแกรมไปทำงานยังระบบยูนิกซ์เครื่องอื่น แทนที่จะต้องไปแก้ไขทุกโปรแกรมเพื่อแก้ไขชื่อคำสั่งเต็มของ

บางคำสั่ง โปรแกรมนี้จะทำการแก้ไขให้ โดยที่ผู้ใช้ไม่ต้องแก้ไขด้วยตนเอง

2. คำสั่งหรืออรรถประโยชน์ที่ชุดของโปรแกรมเหล่านี้ใช้ จะเลือกใช้เฉพาะคำสั่งที่มีอยู่ในระบบยูนิกซ์ส่วนใหญ่เท่านั้น จะหลีกเลี่ยงการใช้คำสั่งที่มีเฉพาะในยูนิกซ์รุ่นใหม่ๆ ถึงแม้ว่าคำสั่งใหม่จะมีประสิทธิภาพมากกว่า เพื่อหลีกเลี่ยงปัญหาการเคลื่อนย้าย (portable) ของโปรแกรม เช่นใช้คำสั่ง awk แทนที่จะใช้ nawk นอกจากนี้ในกรณีที่คำสั่งใดเป็นได้ทั้งคำสั่งฝังใน (builtin command) ของเชลล์ และคำสั่งภายนอก (external command) จะเลือกใช้คำสั่งฝังในแทน ทั้งนี้เนื่องจากคำสั่งฝังในของเชลล์ทำงานได้เร็วกว่าคำสั่งภายนอก

3. ทุกโปรแกรมจะมีความอิสระจากกัน ยกเว้นแฟ้มกลางที่เก็บชื่อคำสั่งเต็มเท่านั้น

4. ชุดของโปรแกรมตรวจสอบความมั่นคงนี้จะไม่ผูกติดกับส่วนติดต่อผู้ใช้ ผู้ใช้สามารถที่จะเรียกโปรแกรมใดโปรแกรมหนึ่งทำงานโดยอิสระได้ โดยที่ไม่จำเป็นต้องเรียกโปรแกรมส่วนติดต่อผู้ใช้ ในอนาคตถ้ามีการเปลี่ยนรูปแบบการติดต่อผู้ใช้ ชุดของโปรแกรมเหล่านี้ก็ยังคงสามารถใช้งานได้

5. เนื่องจากบางโปรแกรมใช้เวลาทำงานค่อนข้างนาน ดังนั้นจะมีข้อความแสดงให้ผู้ใช้ทราบ โดยที่ผู้ใช้สามารถที่จะสั่งให้โปรแกรมทำงานเบื้องหลัง (background) แล้วให้เก็บผลลัพธ์ไว้ในแฟ้มที่ต้องการ หรือให้ทำงานแบบเชิงโต้ตอบในขณะนั้นได้

6. เนื่องจากบางโปรแกรมต้องการสิทธิพิเศษของผู้จัดการระบบ (root privi-ledge) ดังนั้นบางโปรแกรมจะมีข้อความเตือนผู้ใช้ว่าไม่สามารถทำงานได้ เฉพาะผู้จัดการระบบเท่านั้นที่เรียกใช้ได้

ชุดโปรแกรมตรวจสอบความมั่นคงจำแนกออกเป็นประเภทต่างๆดังนี้

1. ชุดโปรแกรมตรวจสอบความมั่นคงสำหรับผู้ใช้
2. ชุดโปรแกรมตรวจสอบความมั่นคงสำหรับแฟ้มและไดเรคทอรีระบบ
3. ชุดโปรแกรมตรวจสอบความมั่นคงทางด้านเครือข่าย
4. ชุดโปรแกรมตรวจสอบความมั่นคงทางด้านอื่นๆ

5.2 ชุดโปรแกรมตรวจสอบความมั่นคงสำหรับผู้ใช้

5.2.1 chkpath.sh

หน้าที่ : ตรวจสอบว่าผู้ใช้คนใดมีการกำหนดการค้นหาคำสั่ง (search path) ที่เป็นอันตรายหรือไม่

ในกรณีที่เชลล์เป็นเบอร์นเชลล์หรือคอร์นเชลล์ การกำหนดการค้นหาคำสั่งที่ถูกต้องควรเป็นดังนี้

```
PATH=/bin:/usr/bin:/etc:.
```

ในกรณีที่เป็น ซีเชลล์ การกำหนดการค้นหาคำสั่งที่ถูกต้องควรเป็นดังนี้

```
set path = ( /bin /usr/bin /etc . )
```

แต่ถ้าผู้ใช้กำหนดลักษณะการค้นหาคำสั่งเป็นดังนี้

```
PATH=.:/bin:/etc:/usr/bin  หรือ
set path = ( . /bin /etc /usr/bin )
```

เชลล์จะทำการค้นหาคำสั่งโดยเริ่มจาก ไดเรคทอรีที่อยู่ปัจจุบัน (current directory) ก่อนแล้วจึงตามด้วยไดเรคทอรีที่กำหนดไว้ การกำหนดการค้นหาคำสั่งโดยวิธีนี้อาจทำให้ผู้ใช้เรียกโปรแกรมโทรจัน (trojan horse) (Garfenkel, Spafford,1991) โดยไม่ตั้งใจ ซึ่งอาจเป็นอันตรายต่อแฟ้มของผู้ใช้ได้

5.2.2 chkuser.sh

หน้าที่ : 1. ตรวจสอบบิตอนุญาตในไดเรคทอรีบ้าน (home directory) ของผู้ใช้
2. ตรวจสอบบิตอนุญาตของโปรแกรมเริ่มต้นที่สำคัญ (startup program) ของผู้ใช้
3. ตรวจสอบว่าผู้ใช้คนไหนมีแฟ้ม .rhosts

โดยทั่วไปแล้วไดเรคทอรีบ้านของผู้ใช้แต่ละคน ไม่ควรจะให้ผู้ใช้คนอื่นสามารถเขียนลงไปได้ ซึ่งบิตอนุญาตที่เหมาะสมควรจะเป็น 750 หรือ 755

ในไดเรคทอรีบ้านของผู้ใช้แต่ละคน จะมีโปรแกรมเริ่มต้น (startup file) สำหรับโปรแกรมแต่ละประเภท ดังนั้นผู้ใช้คนอื่นไม่ควรที่จะเขียนหรือลบข้อความลงในแฟ้มเหล่านี้

แฟ้ม .rhosts ทำให้ผู้ใช้ที่อยู่บนเครื่องที่มีกำหนดไว้ในแฟ้ม .rhosts สามารถเข้าไปเป็นผู้ใช้เจ้าของแฟ้ม .rhosts ได้โดยไม่ต้องใส่รหัสผ่าน ซึ่งอาจไม่ใช่คนคนเดียวกัน ดังนั้นจึงไม่ควรมีแฟ้ม .rhosts ในไดเรคทอรีของผู้ใช้ เพื่อหลีกเลี่ยงปัญหาความมั่นคง

5.2.3 trojan.sh

หน้าที่ : ตรวจสอบหาโปรแกรมโทรจัน (trojan program) ที่อาจอยู่ในระบบ

โปรแกรมโทรจัน คือโปรแกรมที่ดูเหมือนว่าทำงานอย่างหนึ่ง แต่ทำงานอีกอย่างหนึ่งอย่างซ่อนเร้น โปรแกรมโทรจันมักจะมีชื่อเหมือนกับคำสั่งของระบบทั่วไป แต่จะซ่อนอยู่ในไดเรคทอรีอื่น

ถ้าผู้ใช้กำหนดการค้นหาคำสั่งไม่ถูกต้อง เช่นใส่ไดเรคทอรีปัจจุบันไว้เป็นไดเรคทอรีแรกสุด เวลาเรียกใช้คำสั่งของระบบ แทนที่จะเป็นคำสั่งของระบบจริง ก็จะเป็นโปรแกรมโทรจันซึ่งซ่อนอยู่ในไดเรคทอรีที่ผู้ใช้อยู่ขณะนั้น

5.3 ชุดโปรแกรมตรวจสอบความมั่นคงของระบบ

5.3.1 passwd.sh

หน้าที่ : 1. ตรวจสอบว่าผู้ใช้คนใดที่ไม่มีรหัสผ่าน
2. ตรวจสอบโครงสร้างภายในแฟ้ม /etc/passwd , /etc/shadow , /etc/group และ /etc/inittab

แฟ้ม /etc/passwd และ แฟ้ม /etc/shadow เป็นแฟ้มที่เก็บรายชื่อของผู้ใช้ทุกคนในระบบ เป็นแฟ้มที่มีความสำคัญมากในระบบยูนิกซ์ ดังนั้นถ้าแฟ้มทั้งสองนี้มีโครงสร้างภายในที่ไม่ถูกต้อง จะมีผลทำให้ผู้ใช้ไม่สามารถเข้าไปในระบบได้

ผู้ใช้ทุกคนควรจะมีรหัสผ่าน ทั้งนี้เพื่อไม่ให้บุคคลภายนอกสามารถเข้ามาแอบใช้ได้ ซึ่งระบบยูนิกซ์บางรุ่นจะเก็บรหัสที่ผ่านการเข้ารหัสแล้วในแฟ้ม /etc/shadow ซึ่งแฟ้มนี้ไม่ควรให้ผู้ใช้ทั่วไปสามารถดูได้

แฟ้ม /etc/group เป็นแฟ้มที่เก็บรายชื่อกลุ่มที่มีอยู่ในระบบ และกำหนดว่าผู้ใช้คนใดอยู่กลุ่มใด

แฟ้ม /etc/inittab เป็นแฟ้มกำหนดการทำงานระดับต่างๆของระบบยูนิกซ์ แฟ้มทั้งสองนี้มีโครงสร้างที่แน่นอน ดังนั้นจึงจำเป็นที่ต้องตรวจสอบว่าโครงสร้างภายในแฟ้มถูกต้องหรือไม่

5.3.2 checkdir.sh

หน้าที่ : ตรวจสอบว่าผู้ใช้ทั่วไปสามารถที่จะเขียนลงในไดเรคทอรีของระบบอันได้แก่ /bin , /etc , /usr , /lib , /var และ /sbin ได้หรือไม่

โดยทั่วไปบิตอนุญาตของไดเรคทอรีของระบบจะเป็น 755 หรือ 775 ยกเว้นบางไดเรคทอรีเท่านั้น เช่น /tmp , /usr/spool/uucppublic ซึ่งทุกคนสามารถที่จะนำแฟ้มไปเก็บที่ไดเรคทอรีนั้นได้

คำสั่งที่นำมาใช้ในการตรวจสอบคือคำสั่ง find ดังตัวอย่างข้างล่าง

```
find /bin -type d -perm -2 -exec ls -ld {} \;
```

คำสั่งนี้เป็นการแสดงไดเรคทอรีที่อยู่ภายใต้ /bin ที่มีบิตอนุญาตเป็น 777 และแสดงผลลัพธ์ออกมา โดยใช้คำสั่ง ls -ld ซึ่งเป็นการแสดงรายละเอียดของ

ไดเรคทอรีนั้น

5.3.3 <u>checkfile.sh</u>

<u>หน้าที่</u> : ตรวจสอบแฟ้มของระบบว่าผู้ใช้สามารถเขียนได้หรือไม่

โดยทั่วไปบิตอนุญาตของแฟ้มระบบจะเป็น 755 หรือ 644 แต่ไม่ควรเป็น 777 หรือ 666 ซึ่งทำให้ผู้ใช้ทุกคนสามารถที่จะลบข้อความหรือดัดแปลงหรือลบแฟ้มข้อมูลนั้นได้

คำสั่งที่นำมาใช้ในการตรวจสอบ คือคำสั่ง find ดังตัวอย่างข้างล่าง

```
find /bin -type f -perm -2 -exec ls -l {} \;
```

คำสั่งนี้ให้แสดงแฟ้มภายใต้ไดเรคทอรี /bin ที่มีบิตอนุญาตเป็น 777 หรือ 666 แล้วให้แสดงผลออกมาเป็น คำสั่ง ls -l ของแฟ้มนั้น

5.3.4 <u>device.sh</u>

<u>หน้าที่</u> : 1. ตรวจสอบบิตอนุญาตของแฟ้มอุปกรณ์ที่สำคัญ (device file)
2. ตรวจสอบแฟ้มอุปกรณ์ที่อยู่นอกเหนือ ไดเรคทอรี /dev

โดยทั่วไปแฟ้มอุปกรณ์ที่สำคัญเช่น /dev/mem , /dev/kmem หรือ /dev/swap ไม่ควรให้ผู้ใช้ทั่วไปสามารถเขียนทับลงไปได้ นอกจากนี้ แฟ้มอุปกรณ์ทั้งหลายควรจะอยู่ในไดเรคทอรี /dev เท่านั้น ไม่ควรจะมีแฟ้มอุปกรณ์ในไดเรคทอรีอื่นๆ

i17365899

5.4 ชุดโปรแกรมตรวจสอบความมั่นคงทางด้านเครือข่าย

5.4.1 ftp.sh

หน้าที่ : ตรวจสอบบิตอนุญาตของแฟ้มและไดเรคทอรีในการทำ anonymous ftp

anonymous ftp คือ การที่ผู้ใช้ที่อยู่บนเครื่องอื่นสามารถทำการโอนแฟ้มไปยังเครื่องของตนได้จากไดเรคทอรีที่กำหนดไว้โดยที่ไม่ต้องมี account ของตนบนเครื่องนั้น เนื่องจากการทำ anonymous ftp สามารถทำให้ผู้ใช้ทุกคนที่อยู่บนเครือข่ายสามารถเข้ามาโอนแฟ้มไปยังเครื่องของตนได้ ดังนั้นจึงจำเป็นที่จะต้องมีการกำหนดบิตอนุญาตของแฟ้มและไดเรคทอรีให้เหมาะสม เพื่อป้องกันปัญหาทางด้านความมั่นคง

5.4.2 tcpchk.sh

หน้าที่ : 1. ตรวจสอบว่าสามารถโอนแฟ้มโดยใช้คำสั่ง tftp (trivial file transfer program) ได้หรือไม่
2. ตรวจสอบว่าในแฟ้ม alias มี decode alias หรือไม่
3. ตรวจสอบว่าโปรแกรม sendmail มีช่องโหว่ทางความมั่นคง หรือไม่

โปรแกรม tftp เป็นโปรแกรมที่ใช้ในการโอนแฟ้มระหว่างเครื่อง แต่เนื่องจากเป็นโปรแกรมขนาดเล็ก จึงไม่ได้คำนึงถึงความมั่นคง ดังนั้นโปรแกรม tftp จึงไม่ควรที่จะสามารถโอนแฟ้มใดๆ จากไดเรคทอรีใดๆได้

แฟ้ม alias เป็นแฟ้มที่กำหนดนามแฝง (alias) ของกลุ่มผู้ใช้ ในกรณีที่แฟ้ม alias มี decode alias อยู่ทำให้สามารถส่งจดหมายอิเล็กทรอนิกส์ (e-mail) ไปยังโปรแกรม decode ซึ่งจะทำให้เกิดปัญหาทางด้านความมั่นคงในระบบ

5.4.3 tcpfile.sh

หน้าที่ : ตรวจสอบโครงสร้างภายในแฟ้มที่เกี่ยวข้องกับโปรแกรม TCP/IP อันได้แก่ แฟ้ม /etc/hosts, /etc/hosts.equiv , /etc/ftpusers, /etc/service และ /etc/inetd.conf

แฟ้ม /etc/hosts จะเก็บชื่อเครื่องและ IP address ที่อยู่บนเครือข่าย แฟ้ม /etc/hosts.equiv เก็บชื่อเครื่องซึ่งผู้ใช้ที่มีชื่อเหมือนกันสามารถเข้าไปใช้ได้โดยไม่ต้องใส่รหัสผ่าน ซึ่งเครื่องที่เก็บในแฟ้ม /etc/hosts.equiv เรียกว่า trusted host ในกรณีที่แฟ้ม /etc/hosts.equiv มีคำว่า "+" หมายถึงว่ายอมให้ทุกเครื่องเป็น trusted host ซึ่งเป็นสิ่งที่ไม่เหมาะสม ดังนั้นจึงต้องมีการตรวจสอบ trusted host ในแฟ้มนี้

แฟ้ม /etc/ftpusers เก็บชื่อผู้ใช้ที่ไม่สามารถใช้คำสั่ง ftp (file transfer program) ในกรณีที่ระบบมีแฟ้ม /etc/ftpusers ควรจะมีชื่อผู้ใช้ root ในแฟ้มนี้ด้วย

แฟ้ม /etc/services เป็นแฟ้มที่เก็บชื่อบริการ (service) ของโปรแกรม TCP/IP ที่อยู่บนเครื่องนั้น ส่วนแฟ้ม /etc/inetd.conf เป็นแฟ้มที่เก็บชื่อบริการซึ่งโปรแกรม inetd จะอ่านจากแฟ้มนี้ ทั้งสองแฟ้มนี้มีโครงสร้างที่แน่นอน ซึ่งหากโครงสร้างภายในของแฟ้มทั้งสองนี้ไม่ถูกต้อง จะมีผลทำให้โปรแกรม TCP/IP ทำงานผิดพลาดได้

5.4.4 uucp.sh

หน้าที่ : ตรวจสอบบิตอนุญาตของโปรแกรมและไดเรคทอรีที่เกี่ยวข้องกับ UUCP (UNIX TO UNIX COPY PROGRAM)

โปรแกรม UUCP เป็นโปรแกรมที่ใช้ในการติดต่อระหว่างระบบยูนิกซ์สองเครื่องขึ้นไป ซึ่งมีโปรแกรมและไดเรคทอรีที่เกี่ยวข้องจำนวนมาก ถ้าหากมีการกำหนดบิตอนุญาตไม่ถูกต้อง จะทำให้เกิดจุดหละหลวมทางด้านความมั่นคงได้

5.5 ชุดโปรแกรมตรวจสอบความมั่นคงด้านอื่นๆ

5.5.1 chkmail.sh

หน้าที่ : ตรวจสอบบิตอนุญาตของแฟ้มภายใต้ไดเรคทอรีเมล์ (Mail directory) และตรวจสอบว่าเจ้าของแฟ้มและชื่อแฟ้มเหมือนกันหรือไม่

โดยทั่วไปในไดเรคทอรีเมล์จะประกอบด้วยแฟ้มที่มีชื่อเป็นผู้ใช้ในระบบและเจ้าของแฟ้มควรจะเป็นชื่อเดียวกันกับชื่อของแฟ้ม ทั้งนี้หากชื่อแฟ้มและชื่อเจ้าของแฟ้มไม่ตรงกันจะทำให้บุคคลที่มีชื่อเป็นเจ้าของแฟ้ม สามารถอ่านแฟ้มของบุคคลอื่นได้

นอกจากนี้บิตอนุญาตของแฟ้มเมล์ควรจะเป็น 600 ทั้งนี้เพื่อไม่ให้บุคคลอื่นที่ไม่ใช่เจ้าของเมล์สามารถอ่านเมล์ของผู้อื่นได้

5.5.2 chkmisc.sh

หน้าที่ : ตรวจสอบว่าคำสั่งต่อไปนี้มีจุดหละหลวมทางความมั่นคงหรือไม่

1. คำสั่ง write
2. คำสั่ง find
3. คำสั่ง xargs
4. โปรแกรม preserve
5. โปรแกรม shell
6. ตัวแปร IFS ในโปรแกรม shell

ในการเขียนเชลล์โปรแกรม ซึ่งมีความจำเป็นต้องใช้คำสั่งของระบบยูนิกซ์เข้ามาช่วย ในบางกรณีถ้ามีการกำหนดบิตอนุญาตไม่ถูกต้อง อาจทำให้เกิดจุดหละหลวมทางความมั่นคงได้

5.5.3 config.sh

หน้าที่ : ดูคุณลักษณะทั่วไปของเครื่อง โดยดูจากแฟ้ม /etc/default/login และ แฟ้ม /etc/default/su

ในแฟ้ม /etc/default/login จะกำหนดตัวแปรต่างๆ เช่น ขนาดของแฟ้มที่ใหญ่ที่สุดที่สามารถสร้างได้ หรือกำหนดว่าผู้ใช้ทุกคนจำเป็นต้องมีรหัสผ่านหรือไม่

ในแฟ้ม /etc/default/su จะกำหนดว่า เมื่อใดก็ตามที่มีการใช้คำสั่ง su จะมีการบันทึกการใช้คำสั่งนี้ที่ใด ซึ่งโดยทั่วไปจะเป็นแฟ้ม /usr/adm/sulog

5.5.4 suid1.sh

หน้าที่ : ตรวจสอบแฟ้มที่มีบิตกำหนด SUID และ SGID ภายใต้ไดเรคทอรี ที่ต้องการ

เนื่องจากแฟ้มที่มีบิตกำหนด SUID และ SGID สามารถทำให้ผู้ใช้คนอื่นที่เรียกแฟ้มเหล่านี้มีสิทธิเทียบเท่าเจ้าของแฟ้ม โดยเฉพาะอย่างยิ่งแฟ้มที่มีบิตกำหนด SUID เป็น root ซึ่งทำให้ผู้ที่เรียกโปรแกรมสามารถเป็น root ได้ ดังนั้นจึงจำเป็นที่ต้องมีการตรวจสอบแฟ้มเหล่านี้

การค้นหาแฟ้มที่มีบิตกำหนด SUID และ SGID ใช้คำสั่งดังนี้

```
echo "Please enter directory name "
read DIR
find $DIR \( -perm -4000 -o -perm -2000 \) -print
```

-perm -4000 เป็นคำสั่งเลือก (option) ของคำสั่ง find ที่ใช้ค้นหาแฟ้มข้อมูลที่มีบิตกำหนด SUID ส่วน -perm -2000 เป็นคำสั่งเลือกที่ใช้ค้นหาแฟ้มที่มีบิตกำหนด SGID

5.5.5 suid2.sh

หน้าที่ : ตรวจสอบแฟ้มที่มีบิตกำหนด SUID และ SGID ทั้งระบบที่แตกต่างจาก แฟ้มที่เก็บรายชื่อไว้แล้ว

เนื่องจากแฟ้มที่บิตกำหนด SUID และ SGID ที่ทำอย่างไม่ถูกต้อง สามารถทำให้ผู้ใช้หรือผู้บุกรุก สามารถมีสิทธิเทียบเท่าเจ้าของแฟ้ม ดังนั้นจึงจำเป็นที่จะต้องมีแฟ้มที่เก็บรายชื่อแฟ้มที่มีบิตกำหนด SUID และ SGID ทั้งระบบไว้แล้วสำหรับตรวจสอบหาแฟ้มที่ผิดปกติ

5.5.6 sulog.sh

หน้าที่ : ตรวจสอบหาผู้ที่ไม่ได้เป็นผู้จัดการระบบที่สามารถใช้คำสั่ง su เป็น root ได้

คำสั่ง su เป็นคำสั่งที่ทำให้ผู้ใช้สามารถเป็นผู้ใช้ที่กำหนดใน อาร์กิวเมนต์ (argument) ตามหลังคำสั่งsu ในกรณีที่ใช้คำสั่ง su โดยไม่มีอาร์กิวเมนต์ตามหลัง หมายถึงว่าต้องการเป็น root ในกรณีนี้ระบบจะมีการบันทึกผู้ที่ใช้คำสั่ง su ไว้ในแฟ้ม /usr/adm /sulog

โปรแกรมนี้จะถามชื่อของผู้ที่มีสิทธิใช้คำสั่ง su ได้ จากนั้นจะเข้าตรวจสอบ และแสดงรายชื่อของผู้ที่ใช้คำสั่ง su เป็น root โดยที่ไม่ได้อยู่ในกลุ่มของผู้มีสิทธิ