Enterprise Linux 實戰講座

RHEL 4 的安全新機制－Security Enhanced Linux

SELinux（Security Enhanced Linux）是 RHEL 4 的安全新機制，但很多 RHEL 4 的使用者卻因其嚴格的安全控管加上不知該如何設定，一開始便停用此機制，著實可惜。本篇文章暫不探討 SELinu 複雜的運作機制，先讓使用者瞭解其基本操作，以便快速利用 SELinux 來自訂系統的安全機制。

簡介:

SELinux（Security Enhanced Linux）這項功能是由國家安全署（National Security Agency）開發，它是建構於 Kernel 之上的安全機制，提供以往只在商業資訊技術作業環境才擁有的高階安全基礎架構。SELinux 實現了以 Policy 為基礎之命令存取控制，進而達成更精密的安全評量，對於任意存取控制機制也有更大的彈性。SELinux 可為每一服務指定權限及其 Policy（Mandatory Access Control），而並非是仰賴於傳統 user/group/other 及 root 帳號的簡單權限控制方式（Discretinoary Access Control）。

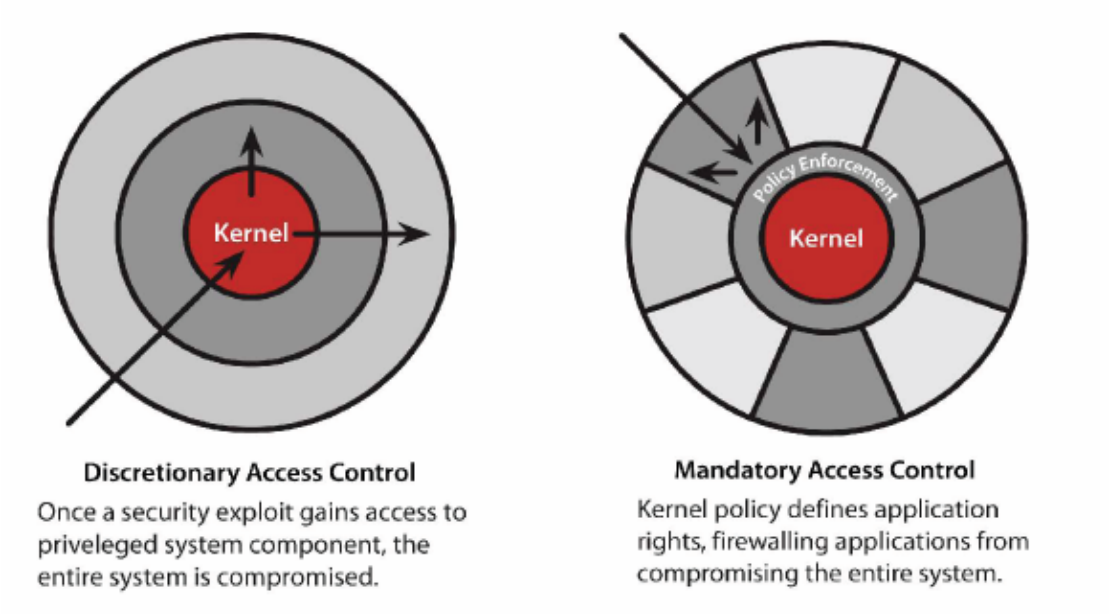

圖 1、SELinux 的架構示意圖

資料來源：Security Features in Red Hat Enterprise Linux 4

DAC（Discretionary access control）機制：由資源的擁有者決定權限，例如傳統 Linux。

MAC（Mandatory Access Control）機制：由安全機制管理者來決定權限，例如 SELinux。

Alex Lin (林彥明) 2005.07.28

RHEL4 將 SELinux 與一系列網際網路服務加以整合，包括有 BIND、Network Time Protocol (NTP)、Apache，使得其優點能夠更輕易地拓展。其要求極度安全環境的組織，可以於更多的應用程式上實作更廣泛的 SELinux 功能，甚至為每一服務制定嚴格的 SELinux 原則。傳統的 Linux 系統若遭駭客侵入 Web Server，可能導致整個伺服器癱瘓；但有了 SELinux 的保護，雖然 Web Server 被入侵，卻可將受害範圍減至最少（圖 2）。

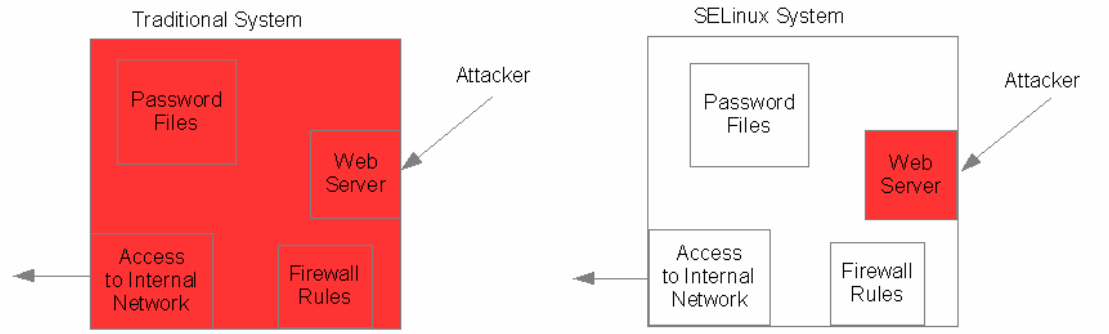

圖 2、傳統 Linux 與 SELinux 安全機制比較圖

初探 SELinux

安裝 RHEL 4 的過程中，會出現如圖 3 的畫面，詢問是否啟用 SELinux，預設安裝的選項會「啟用」SELinux。

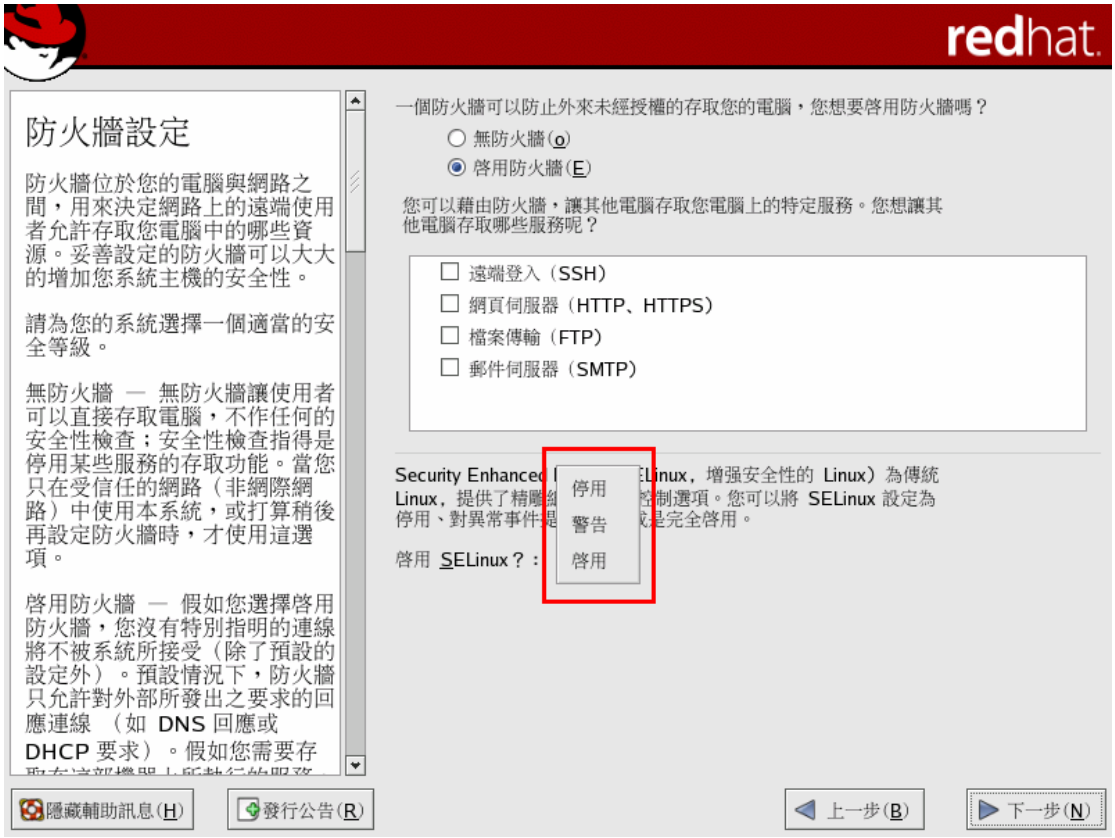

圖 3、安裝時 SELinux 選項

這 3 個選項的差別如下：

- 停用（Disabled）：停用 SELinux 功能
- 警告（Warn／Permissive）：僅顯示警告訊息
- 啟用（Active／Enforcing）預設值：強制執行 SELinux 功能

系統是否啟用 SELinux 的設定，記錄在/etc/sysconfig/selinux，待安裝完成後，若有啟用 SELinux，可檢查/etc/sysconfig/selinux 內容可得知系統預設是否啟用 SELinux。

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#       enforcing - SELinux security policy is enforced.
#       permissive - SELinux prints warnings instead of enforcing.
#       disabled - SELinux is fully disabled.
SELINUX=enforcing
# SELINUXTYPE= type of policy in use. Possible values are:
#       targeted - Only targeted network daemons are protected.
#       strict - Full SELinux protection.
```

SELINUXTYPE=targeted

SELINUX 參數值：有「enforcing、permissive、disabled」分別對應安裝時的選項「啟用、警告、啟用」。

SELINUXTYPE 參數值：有「targeted、strict」兩種
SELINUXTYPE=targeted：保護網路相關服務
SELINUXTYPE=strict：完整的保護功能，包含網路服務、一般指令及應用程式
註：但 RHEL 4 目前尚未支援 strict policy，只提供 targeted policy

什麼是 Policy，其實讀者可以先把 Policy 想成當使用者要執行程式（例如啟動 WWW Server），或 Process 要執行動作時，系統就會依照 Policy 所制定內容來檢查使用者或 Process 相對應的權限，如果全部權限都符合的話，系統就會允許這個操作的執行。

SELinux 檢查方式是獨立於傳統的使用者權限，在 SELinux 中，你必須要同時符合傳統的使用者權限和 SELinux 權限才能順利執行操作。

SELinux 最麻煩的就是需要一個好的 Policy 才可以讓 SELinux 發揮效果。如果制定的太寬鬆會使 SELinux 毫無用武之地，而太嚴格又會讓系統管理者凡事礙手礙腳。NSA 把制定 Security Policy 的工作由套件發行者來做，而 RedHat,、Novell SUSE,、Fedora…等等也都制定了一套基本的 Policy。

而 targeted policy 即是 RHEL 4 已定義好的 policy，這個 targeted policy 的用途為保護下列的網路服務：

- dhcpd
- httpd
- mysqld
- named
- nscd
- ntpd
- portmap
- postgres
- snmpd
- squid
- syslogd

註：targeted policy 內容置於 /etc/selinux/targeted/

看了上述的文字，讀者可能會疑問什麼是 targeted policy，對 SELinux 也還是沒什麼概念，其實各位現在只要知道 SELinux 保護上述的網路服務，加強其安全性；不過通常安全性愈高，也表示受到限制加多，更加不方便。例如原本 root 可任意重新啟動這些服務，但現今可能因為受到 SELinux 控制而無法順利執行。下面筆者用個簡單的實例演練讓讀者體會啟用 SELinux 會對系統造成什麼影響，並介紹基本 SELinux 指令。

實例演練一：SELinux 對 httpd daemon 的影響

步驟 1、啟用 SELinux

若是讀者安裝時未啟用 SELinux，請直接修改 SELinux 設定檔 /etc/sysconfig/selinux，把 SELINUX 這個參數設成 enforcing，然後重新開機。讀者會發現開機時出現奇怪的錯誤訊息，例如圖 4 可看到是關於 syslogd 及 portmap 的訊息；其實這些訊息是因為啟用 SELinux 所產生的，因為 targeted policy 有針對這兩個 daemon 個安全控管，所以才會出現這些奇怪的訊息。

註：這個實例演練必須是安裝時停用 SELinux，之後再啟用 SELinux 才會看到如圖 4 的錯誤訊息。

```
Setting network parameters:                                [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:                                [  OK  ]
Starting system logger: syslogd: error while loading shared libraries: /lib/ld-l
inux.so.2: cannot apply additional memory protection after relocation: Permissio
n denied
                                                           [FAILED]
Starting kernel logger:                                    [  OK  ]
Starting irqbalance:                                       [  OK  ]
Starting portmap: audit(1119482531.354:0): avc:  denied  { read } for  pid=1722
exe=/sbin/portmap name=libnsl-2.3.4.so dev=dm-0 ino=67194 scontext=user_u:system
_r:portmap_t tcontext=system_u:object_r:file_t tclass=file
audit(1119482531.355:0): avc:  denied  { read } for  pid=1722 exe=/sbin/portmap
name=libnsl-2.3.4.so dev=dm-0 ino=67194 scontext=user_u:system_r:portmap_t tcont
ext=system_u:object_r:file_t tclass=file
portmap: error while loading shared libraries: libnsl.so.1: cannot open shared o
bject file: No such file or directory
                                                           [FAILED]
Starting NFS statd:                                        [  OK  ]
Starting NFS4 idmapd:                                      [  OK  ]
Mounting other filesystems:                                [  OK  ]
Starting automount:No Mountpoints Defined                  [  OK  ]
Starting smartd:                                           [FAILED]
Starting acpi daemon:                                      [  OK  ]
Starting cups: _
```

圖 4：啟用 SELinux 的開機錯誤訊息

步驟 2、啟動 httpd daemon

待開機完成後，請用 root 身份啟動 httpd daemon

```
[root@server1 ~]# service httpd restart
Stopping httpd:                                            [FAILED]
Starting httpd: /usr/sbin/httpd: error while loading shared libraries: libpcre.so.0:
cannot open shared object file: No such file or directory
                                                           [FAILED]
```

不會吧，怎麼出現一堆奇怪的訊息，竟然用 root 啟動 httpd 也出現錯誤。讀者想到原因了嗎？沒錯，還是 SELinux 作怪！SELinux 啟用的環境下，root 不再似以前可任意作為。

步驟 3、利用 sestatus 查看 SELinux 狀態

讀者可利用 sestatus 獲知現在 **SELinux 的狀態**

sestatus 語法：

```
Usage: sestatus [OPTION]
   -v    Verbose check of process and file contexts.
Without options, show SELinux status.
```

```
[root@server1 ~]# sestatus
SELinux status:                 enabled ← SELinux 啟用的狀態
SELinuxfs mount:                /selinux ← selinuxfs 檔案系統掛載點
Current mode:                   enforcing ←目前 SELinux 的啟用模式
Mode from config file:          enforcing ←目前 SELinux 啟用模式的設定檔
Policy version:                 18 ← SELinux Policy 之版本
Policy from config file:targeted ← SELinux 啟用模式的設定檔的名稱

Policy booleans: ←SELinux Policy 的布林值（布林值就是 0 和 1），active
即代表布林值為 1；inactive 即代表布林值為 0。
allow_ypbind                    inactive
dhcpd_disable_trans             active
httpd_disable_trans             inactive
httpd_enable_cgi                active
httpd_enable_homedirs           active
httpd_ssi_exec                  active
httpd_tty_comm                  inactive
httpd_unified                   active
```

```
mysqld_disable_trans            inactive
named_disable_trans             inactive
named_write_master_zones        inactive
nscd_disable_trans              inactive
ntpd_disable_trans              inactive
portmap_disable_trans           inactive
postgresql_disable_trans        active
snmpd_disable_trans             inactive
squid_disable_trans             inactive
syslogd_disable_trans           inactive
winbind_disable_trans           active
ypbind_disable_trans            inactive
```

步驟 4、停用 SELinux，重新啟動 httpd

修改 SELinux 設定檔/etc/sysconfig/selinux，把 SELINUX 這個參數設成 disable，然後重新開機。

```
[root@server1 ~]# vi /etc/sysconfig/selinux
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#       enforcing - SELinux security policy is enforced.
#       permissive - SELinux prints warnings instead of enforcing.
#       disabled - SELinux is fully disabled.
SELINUX= disabled
# SELINUXTYPE= type of policy in use. Possible values are:
#       targeted - Only targeted network daemons are protected.
#       strict - Full SELinux protection.
SELINUXTYPE=targeted
```

```
[root@server1 ~]#reboot
```

```
[root@server1 ~]# sestatus
SELinux status:                 disabled
```

```
[root@server1 ~]# service httpd restart
Stopping httpd:                                            [FAILED]
Starting httpd:                                            [  OK  ]
```

root 之所以無法順利啟動 httpd 的原因是 Policy 中「**httpd_disable_trans inactive**」規則所致。但初接觸 RHEL 4 的使用者一定通常不知道這是 SELinux targeted policy 中「httpd_disable_trans inactive」所造成的。遍尋網路，發現原來是 SELinux 作怪，便將 SELinux 停用，一切便順利正常，從此再也不敢啟用 SELinux，如此作法未免有點因噎廢食。下面筆者便介紹 RHEL 4 的 targeted policy 中個別的安全規則的作用及如何針對某條安全規則 active/inactive。

探索 targeted policy

targeted policy 內容置於 /etc/selinux/targeted/，此目錄包含下列幾個子目錄及檔案：

```
[root@server1 ~]# ls -l /etc/selinux/targeted/
-rwx------       1 root root    432 Jun 22 22:22 booleans
drwxr-xr-x      4 root root 4096 Mar 23 04:11 contexts
drwxr-xr-x      2 root root 4096 Mar 23 04:11 policy
```

booleans：存放targeted policy中每個限制的布林值
contexts/：儲存這個targeted policy的security contexts。
policy/：存放二進位型態的policy檔。

查看 booleans 檔的內容，讀者會發現和用 sestatus 指令輸結果中的 Policy booleans 區段項目一樣，只是這個檔案內每個限制規則的值為 0/1，而 sestatus 的表示法為 active/inactive。

```
 [root@server1 /etc/selinux/targeted]# cat booleans
allow_ypbind=1
dhcpd_disable_trans=0
httpd_disable_trans=0
httpd_enable_cgi=1
httpd_enable_homedirs=1
httpd_ssi_exec=1
httpd_tty_comm=0
httpd_unified=1
mysqld_disable_trans=0
named_disable_trans=0
named_write_master_zones=0
nscd_disable_trans=0
ntpd_disable_trans=0
```

```
portmap_disable_trans=0
postgresql_disable_trans=0
snmpd_disable_trans=0
squid_disable_trans=0
syslogd_disable_trans=0
winbind_disable_trans=0
ypbind_disable_trans=0
```

查看及修改/etc/selinux/targeted/booleans

SELinux 提供 getsebool 可查看 targeted policy 中某條限制規則是否為作用中；setsebool 可將限制規則改為作用中或非作用中。

getsebool 語法：

```
Usage:   getsebool -a or getsebool boolean…
```

-a：顯示所有 SELinux 布林值

範例：查詢 allow_ypbind、dhcpd_disable_trans 限制規則是否為作用中

```
[root@server1 ~]# getsebool allow_ypbind dhcpd_disable_trans
allow_ypbind --> active
dhcpd_disable_trans --> inactive
```

由 getsebool 得到 allow_ypbind 規則狀態是作用中（active）而 dhcpd_disable_trans 規則狀態是非作用中（inactive）。這時查看 /etc/selinux/targeted 目錄下 booleans 檔案這個項目的值。

```
[root@server1 /etc/selinux/targeted]# head -2 booleans
allow_ypbind=1
dhcpd_disable_trans=0
```

由上述指令可得知：
/etc/selinux/targeted/ booleans 檔案中某個項目的值若為 1，即代表 active。
/etc/selinux/targeted /booleans 檔案中某個項目的值若為 0，即代表 inactive。

若要修改/etc/selinux/targeted/ booleans 檔案某個項目（限制規則）的布林值（0/1；inactive/active）可利用 setsebool 指令。

setsebool 語法：

```
Usage:   setsebool [ -P ] boolean value | bool1=val1 bool2=val2...
```

[-P]：若沒有附加-P 選項，會立即修改記憶體中該項目（限制規則）的布林值；但不會更新 booleans 檔案。當系統重新開機時會根據/etc/selinux/targeted/booleans 檔的布林值決定是否啟用此限制規則。所以如果需要永久生效，記得要加上-P 選項。

範例：修改 httpd_disable_trans 的布林值

```
[root@server1 ~]# getsebool httpd_disable_trans ←查詢現值
httpd_disable_trans --> inactive
[root@server1 ~]# grep httpd_disable_trans /etc/selinux/targeted/booleans
httpd_disable_trans=0
```

```
[root@server1 ~]# setsebool httpd_disable_trans=1
[root@server1 ~]# getsebool httpd_disable_trans
httpd_disable_trans --> active
[root@server1 ~]# grep httpd_disable_trans /etc/selinux/targeted/booleans
httpd_disable_trans=0 ← 並未改變 booleans 檔案的內容
```

```
[root@server1 ~]# setsebool –P httpd_disable_trans=1
[root@server1 ~]# getsebool httpd_disable_trans
httpd_disable_trans --> active
[root@server1 ~]# grep httpd_disable_trans /etc/selinux/targeted/booleans
httpd_disable_trans=1 ← 改變 booleans 檔案的內容
```

除了使用 getsebool 及 setsebool 指令來查看及修改 /etc/selinux/targeted/booleans 各個項目（安全規則）的布林值外，RHEL 4 還提供了圖形管理工具「system-config-securitylevel」可以來查詢及修改其布林值。

圖 5：system-config-securitylevel 畫面

「system-config-securitylevel」工具除了可設定防火牆外，也可用來修改 SELinux 的政策。system-config-securitylevel 中的「修改 SELinux 的政策」框架中的項目正是對應/etc/selinux/targeted/booleans 每個項目。

system-config-securityleve 與/etc/selinux/targeted/booleans

普羅大眾可能還是熟悉圖形介面，接著筆者便用「system-config-securitylevel」工具來對應/etc/selinux/targeted/booleans 檔案中的項目。

- HTTPD Serivce

圖 6：HTTPD Service 畫面

表 1：HTTPD Service SELinux 政策與 booleans 對照表

HTTPD Service 類別	預設值	booleans 對應項目
Allow HTTPD cig support	☑	httpd_enable_cgi=1
Allow HTTPD to read home directories	☑	httpd_enable_homedirs=1
Allow HTTPD to run SSI executables is the same domain as system CGI scripts	☑	httpd_ssi_exec=1
Disable SELinux protection for httpd daemon	☐	httpd_disable_trans=0
Unify HTTPD handling of all content files	☑	httpd_unified=1

- Name Serivce

圖 7：Name Service 畫面

表 2：Name Service SELinux 政策與 booleans 對照表

HTTPD Service 類別	預設值	booleans 對應項目
Allow named to overwrite master zone files	☐	named_write_master_zones=0
Disable SELinux protection for named daemon	☐	named_disable_trans=0
Disables SELinux protection for nscd daemon	☐	nscd_disable_trans=0

- NIS

圖 8：NIS Service 畫面

表 3：NIS SELinux 政策與 booleans 對照表

NIS 類別	預設值	booleans 對應項目
Allow deamons to run with NIS	☑	httpd_enable_cgi=1
Disable SELinux protection for ypbind daemon	☐	ypbind_disable_trans=0

- SELinux Service Protection

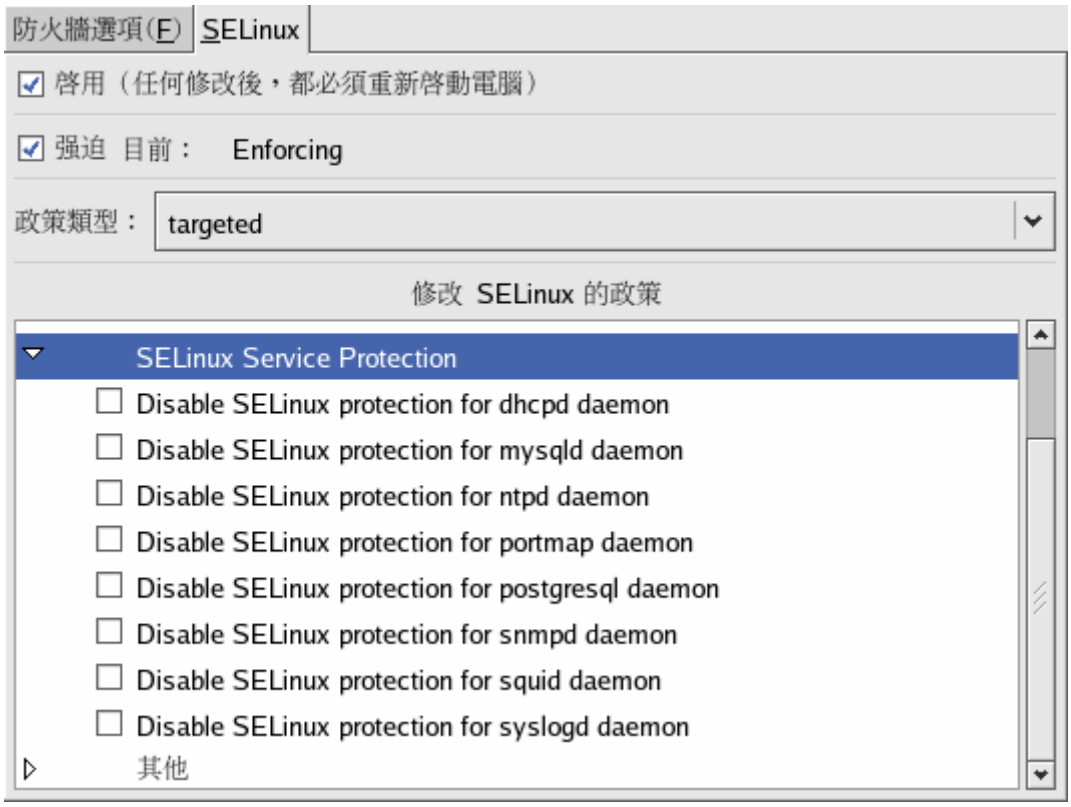

圖 9：SELinux Service Protection 畫面

表 4：SELinux Service Protection 政策與 booleans 對照表

NIS 類別	預設值	booleans 對應項目
Disable SELinux protection for dhcpd daemon	☐	dhcpd_disable_trans=0
Disable SELinux protection for mysqld daemon	☐	mysqld_disable_trans=0
Disable SELinux protection for ntpd daemon	☐	ntpd_disable_trans=0
Disable SELinux protection for portmap daemon	☐	portmap_disable_trans=0
Disable SELinux protection for postgresql daemon	☐	postgresql_disable_trans=0
Disable SELinux protection for snmpd daemon	☐	snmpd_disable_trans=0
Disable SELinux protection for squid daemon	☐	squid_disable_trans=0
Disable SELinux protection for syslogd daemon	☐	syslogd_disable_trans=0

■ 其他

圖 10：其他畫面

表 5：SELinux Service Protection 政策與 booleans 對照表

NIS 類別	預設值	booleans 對應項目
httpd_tty_comm	☐	httpd_tty_comm=0
Winbind_disable_trans	☐	winbind_disable_trans=0

實例演練二：停用 SELinux 對 httpd daemon 的保護

在【實例演練一】中，發現啟用 SELinux 後造成 httpd 無法順利啟動，其主要原因是預設情形，SELinux 會保護 httpd daemon，不讓任何人隨意重新啟動（httpd_disable_trans=0/inactive）。【演練一】把整個 SELinux 停用以解決此問題，本演練嘗試將 httpd_disable_trans 改為 1，再重新啟動 httpd deamon，觀察是否可成功執行。

步驟 1、查看 httpd_disable_trans 是否啟用

利用 getsebool 指令或「system-config-securitylevel」工具檢查。

```
[root@server1 ~]# getsebool httpd_disable_trans ←查詢現值
httpd_disable_trans --> inactive
```

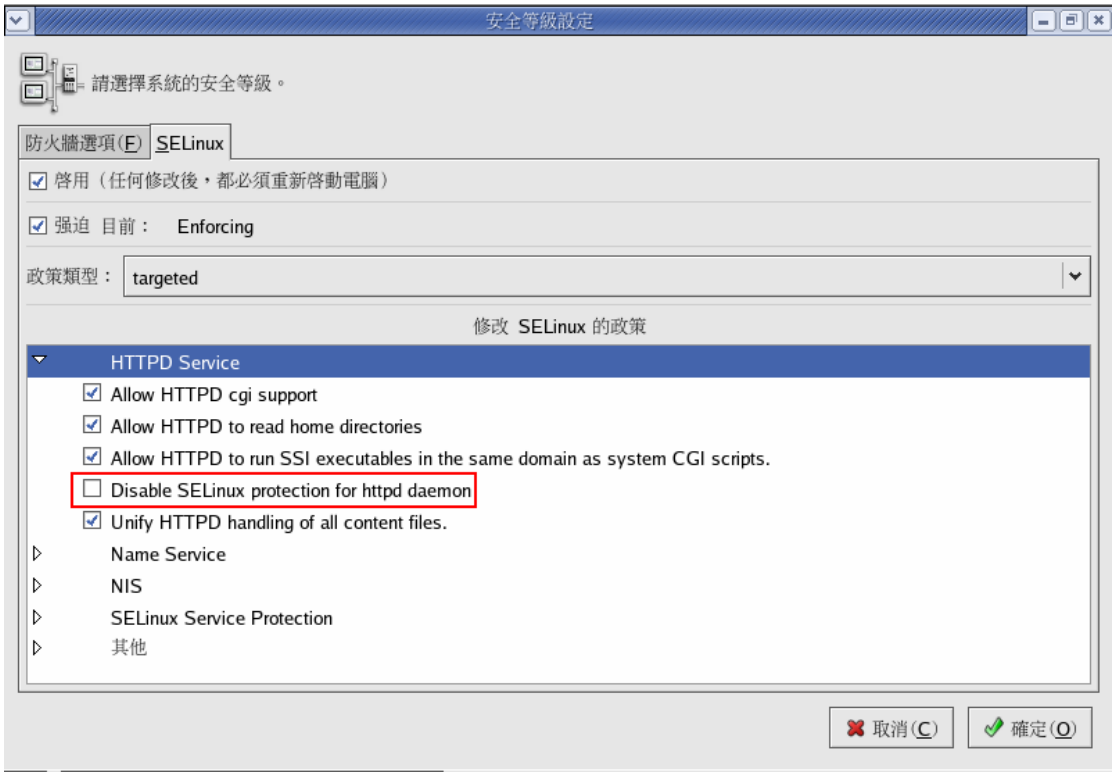

圖 12：「Disable SELinux protection for httpd daemon」畫面

此時，是無法重新啟動 httpd daemon

```
[root@server1 ~]# service httpd restart
Stopping httpd:                                            [FAILED]
Starting httpd: /usr/sbin/httpd: error while loading shared libraries: libpcre.so.0:
cannot open shared object file: No such file or directory
                                                           [FAILED]
```

步驟 2、啟用 httpd_disable_trans

利用 setsebool 改變 httpd_disable_trans 的布林值為 1（啟用）

```
[root@server1 ~]# setsebool –P httpd_disable_trans=1
[root@server1 ~]# getsebool httpd_disable_trans
httpd_disable_trans --> active
```

此指令執行過後，再啟動「system-config-securitylevel」工具會發現「Disable SELinux protection for httpd daemon」項目狀態變為「已勾選」。

圖 12：已啟用「Disable SELinux protection for httpd daemon」畫面

步驟 3、重新啟動 httpd daemon

```
[root@server1 ~]# service httpd restart
Stopping httpd:                                            [FAILED]
Starting httpd:                                            [  OK  ]
```

後記：SELinux 是蠻複雜的機制，除了使用 RHEL 4 預設的 targeted policy 之外，亦可自定 policy，這期文章筆者並未提到 SELinux 的詳細原理，先用圖形工具「system-config-securitylevel」讓讀者先瞭解 SELinux 的基本操作，之後的文章再細究其運作機制與原理。