

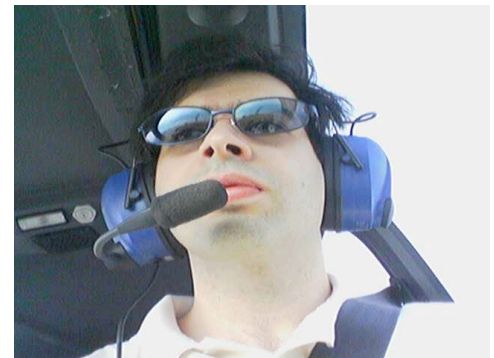
Proteggere i files sensibili da accessi non autorizzati: case study



Giuseppe Paternò
Sikurezza.org
gpaterno@gpaterno.com

Chi sono

- Lavoro come Senior Architect e referente della sicurezza EMEA presso Red Hat
- Nel passato ho ricoperto ruoli analoghi in Sun ed IBM
- Appassionato di sicurezza, sono conosciuto con il soprannome di *Gippa*
- Partecipo alla comunità di *sikurezza.org*
- Maggiori informazioni su:
 - <http://www.gpaterno.com/>
 - <http://www.linkedin.com/in/gpaterno>



Prerequisiti

- La presentazione e' tecnica
- Requisiti dell'audience
 - Prime slides non tecniche, indirizzate per security officer
 - Poi per chi e' tecnico su Linux
 - Si richiede familiarita' con l'ambiente Linux e una base di SE-Linux

La sfida

- La sfida era proteggere dei PDF sensibili
 - Un applicativo J2EE web-based con autenticazione tramite certificati deve accedere ai files PDF e presentarli all'utente finale abilitato
 - I sistemisti e gli operatori non possono accedere in nessun modo ai files PDF e ogni tentativo di accesso verrà loggato
- Il cliente: agenzia governativa

I requisiti (1/2)

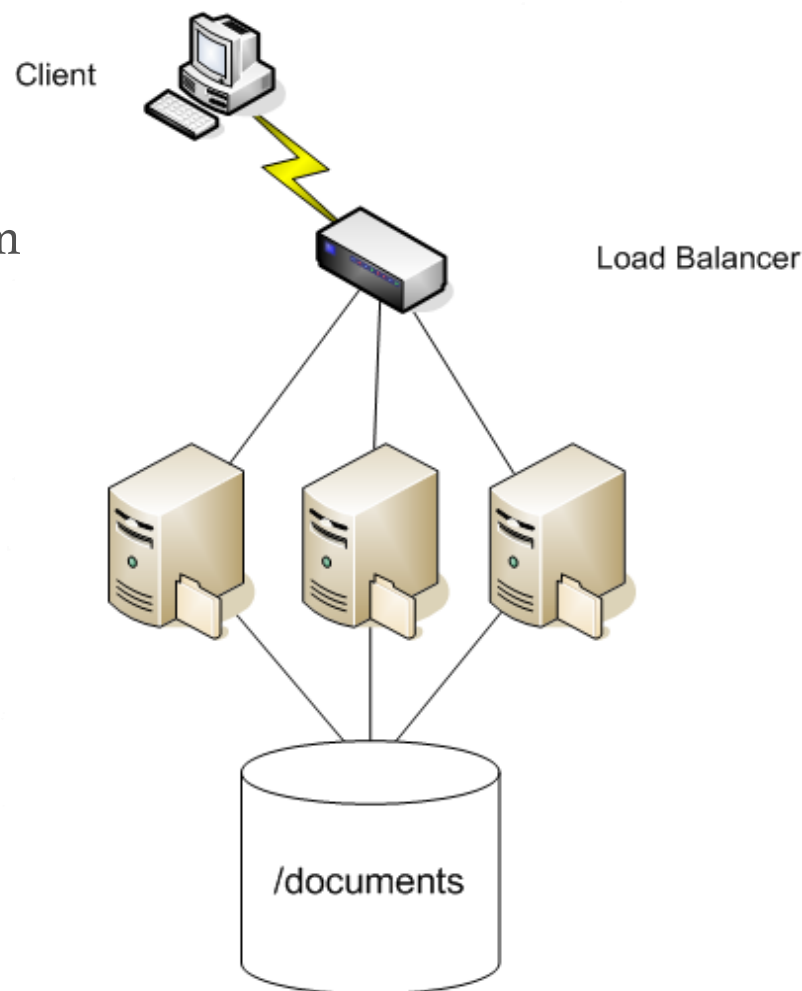
- Univoca identificazione dell'utente tramite LDAP
- Log di audit vanno inviati ad un syslog centralizzato
- Le utenze di gestione non possono diventare root, ma devono eseguire i soltanto alcuni comandi tramite il programma “sudo”
- Le utenze di gestione devono avere diversi privilegi di accesso, da operatore a system administrator autorizzato
- Gli utenti non possono fare “su -” per accedere all'utenza root anche se conoscono la password, tranne quelli esplicitamente autorizzati

I requisiti (2/2)

- Nessun utente, tranne *root* e l'utente applicativo *appserv* possono accedere alla directory “/documents/” e relativi documenti/subdirectories
- L'utente *root* ha diritto di accedere alla directory, ma le operazioni di lettura devono essere sottoposte ad audit
- L'utente applicativo non deve essere sottoposto ad audit per la salvaguardia delle performance dell'applicazione.
- Con l'utente applicativo *appserv* verrà eseguito l'application server e verranno eseguiti alcuni batch che hanno diritto all'accesso alla directory

L'architettura

- Cluster in bilanciamento a 3 nodi
- Filesystem condiviso tramite GFS
 - I files PDF risiedono su questo filesystem
- Application Server Jboss
 - Frontend con Apache per l'autenticazione con smartcard
- Linux come piattaforma OS
 - Red Hat Enterprise Linux
 - Red Hat Cluster Suite e GFS
 - SE-Linux per fornire un accesso mandatorio



I Livelli di accesso

Classe di utente	Permessi
operatore	Può collegarsi alla macchina e visualizzare I log
appmanager	I permessi di operatore + riavviare i servizi
admin	I permessi di appmanager + possibilità di fare “su –”. Non è in grado di accedere ai PDF
Utente applicativo (<i>appserv</i>)	Utente con cui viene avviato l’application server, ha il diritto di accesso ai PDF.
root	Utente di amministratore unix, con accesso diretto dalla sola console.

SE-Linux: introduzione

- SE-Linux è un sistema di sicurezza ad accesso mandatorio (MAC)
- Sviluppato dalla National Security Agency (USA) e presente nel kernel upstream di Linux
- Per default, **qualsiasi cosa non esplicitamente permessa è negata**
- Le regole si chiamano “policy”
- Due tipi di policy predefinite:
 - “targeted”: confinano solamente i daemon (tipicamente quelli di init.d), lasciando non confinati gli altri processi
 - “stricted”: tutto il sistema viene confinato, inclusi gli utenti

SE-Linux: controlli di accesso

- SELinux ha tre tipi di controllo di accesso:
 - Type Enforcement (TE): Type Enforcement è il meccanismo primario di controllo di accessi usato nella policy targeted
 - Role-Based Access Control (RBAC): basato sulle utenze SELinux (che non necessariamente coincidono con quelle di sistema), ma non usata nella policy targeted, serve ad identificare ruoli di un utente
 - Multi Level Security (MLS) e Multi-Category Security (MCS): usata raramente, serve per attribuire delle categorie o dei livelli ai files o alle risorse.

La scelta implementativa

- Il problema era trovare una “formula” che coniugasse la sicurezza con la gestibilita'
 - La gestione è in mano ad operatori con conoscenze di base
 - Il mio obiettivo era non “sconvolgere” le abitudini degli operatori
- Ho deciso di usare:
 - SE-Linux in modalità targeted
 - Multi-Category Security, assegnando una categoria speciale ai files PDF
 - Creazione di un modulo di SE-Linux ad-hoc
 - Configurazione appropriata dei tools di sistema

Creazione della categoria

- Creazione della categoria per “etichettare” i files PDF
- Implementata nel file */etc/selinux/targeted/setrans.conf*
 - *s0:c3=TopSecret*
- Tutti i files devono avere questa label per poter essere protetti, tramite *chcat* o *restorecon*
- Abilitazione dell'utente appserv per l'accesso ai PDF
 - *chcat -l +TopSecret appserv*
- Le informazioni vengono conservate negli “extended attributes” (xattr) del filesystem
 - GFS è un filesystem clustered che supporta xattr

Creazione della categoria

- I permessi di accesso: l'utente di default non può accedere a nessuna categoria, mentre root può vedere tutte le categorie.
- All'utente *appserv* è stato concesso l'autorizzazione ai files *TopSecret*

```
# semanage login -l
```

Login Name	SELinux User	MLS/MCS Range
__default__	user_u	s0
appserv	user_u	-TopSecret
root	root	SystemLow-SystemHigh

Il modulo SE-Linux

- Creazione di due files:
 - **docsecret.te**
 - contiene le policy e la definizione dei tipi
 - Il tipo *docsecret_t* protegge l'accesso da altri processi confinati che necessitano di un *allow* esplicita
 - **docsecret.fc**
 - Contiene i contesti da applicare ai files
 - Consente di applicare automaticamente la categoria di *TopSecret* ai files nella directory */documents*
- I files vengono compilati e caricati in memoria come modulo di SE-Linux (**docsecret.pp**)

Accesso admin: le restrizioni

- Sudo è stato configurato per permettere al gruppo *appmanager* di eseguire i comandi di start/stop dell'application server
- Il comando “su” è stato ristretto al solo gruppo *admin*
- Accesso SSH limitati ai gruppi
 - operatore, appmanager, admin
 - Nessun collegamento direttamente come root
 - In ascolto soltanto sulla rete di amministrazione ad accesso controllato
- L'utente *root* può solo collegarsi in console
- La password di root è in possesso del responsabile dell'erogazione dei servizi, responsabile anche di eventuali fughe di informazioni

Accesso admin: la protezione

Cosa succede se eseguiamo comandi con “sudo” o dopo che abbiamo fatto “su -”:

```
# id
```

```
uid=0 (root) gid=0 (root) groups=0 (root),1 (bin),  
2 (daemon),3 (sys),4 (adm),6 (disk),10 (wheel)  
context=user_u:system_r:unconfined_t
```

Da notare: nonostante l'utente sia diventato root, ha mantenuto la label SE-Linux, proteggendo la confidenzialità dei PDF:

```
# cat /documents/mydoc.pdf
```

```
cat: /documents/mydoc.pdf: Permission denied
```

Audit

- Requisito: ogni tentativo di accesso alla cartella deve essere loggato
 - Ad esclusione dell'utente applicativo *appserv* per performance
 - Inviato ad un syslog esterno per evitare che un root cancelli i files di log
- SE-Linux logga tramite il processo di audit in `/var/log/audit/audit.log`
- Configurato il subsystem di audit `/etc/audit/audit.rules`
 - `-a exit,always -S open -S truncate -F dir=/documents -F uid!=300`
 - Configurato il plugin di syslog per l'invio ad un log server centralizzato

ACL

- Sono state inserite delle ACL a livello di filesystem
 - Le ACL fanno parte della filosofia DAC
 - Il DAC è complementare al MAC, non viene mai escluso
 - Fornisce un'ulteriore protezione in caso qualcuno da console disabili temporaneamente SE-Linux per l'amministrazione (setenforce 0)
- Le ACL
 - # chmod 0750 appserv:appserv /documents/
 - # setfacl -m appserv:rwx /documents/
 - # setfacl -m root:rwx /documents/
 - # getfacl --access /documents/ | setfacl -d -M- /documents/

DEMO

Conclusione

- SE-Linux è sicuramente un sottosistema interessante, tuttavia molto difficile da configurare e gestire. Bisogna trovare il giusto equilibrio:
 - Magari inutile da utilizzare in un printer server interno
 - Utile in situazioni di “frontiera”, con servizi esposti su internet/intranet/extranet
 - Non sempre “certificato” con alcune applicazioni commerciali

Conclusione

- E' bene sempre pensare in un'ottica “Defense-in-depth”
 - Utilizzare sempre più di un “livello” di protezione
 - la sicurezza è più importante quanto è più vicino al dato da proteggere.
 - Nel nostro “case study” è importante criptare i PDF per proteggersi da buchi applicativi
- Potete scaricare questa presentazione e il whitepaper **“Proteggere i files sensibili con SE-Linux”** dal mio sito:
 - <http://www.gpaterno.com/pubblicazioni/>



Q & A



Grazie!

Giuseppe Paternò
gpaterno@gpaterno.com
<http://www.gpaterno.com/>

Sikurezza.org
<http://www.sikurezza.org/>