Linux e TCSEC

Una soluzione con SELinux.

Daniele Diodati

Università degli studi di Perugia

21 aprile 2010

Orange Book

Trusted Computer System Evaluation Criteria (TCSEC) è uno standard emanato dall'United States Government Department of Defense che definisce i requisiti basilari per la valutazione della sicurezza di un sistema informatico.
TCSEC, chiamato anche Orange Book, è la colonna centrale della serie di pubblicazioni Rainbow. Sviluppato inizialmente nel 1983 dal National Computer Security Center (NCSC), un ramo della National Security Agency, e aggiornato nel 1985, è stato sostituito dallo standard internazionale Common Criteria pubblicato nel 2005.

Obbiettivi

I criteri sono stati sviluppati con tre obiettivi principali:

- Fornire agli utenti un metro di valutazione sulla sicurezza del trattamento di informazioni sensibili su sistemi informatici;
- Fornire informazioni ai fabbricanti di prodotti commerciali, utili al fine di soddisfare i requisiti di fiducia per le applicazioni sensibili;
- Fornire una base per specificare i requisiti di sicurezza nelle specifiche di acquisizione.

Policy e Marking

Policy politica di sicurezza imposta dal sistema:

- individuazione soggetti e oggetti;
- regole di accesso definite.

Marking labels associate ad ogni oggetto:

- controllo accessi;
- livelli di sensibilità.

Identification e Accountability

Identification identificazione singolo soggetto:

- gestita esclusivamente da ADP;
- mediazione SRA.

Accountability associazione soggetto⇒(azione,oggetto):

- Audit Trail on SRA;
- Minimizzazione audit;
- Protezione log.

Assurance e Continuous Protection

Continuous Protection protezione continua del TCB:

- salvaguardia meccanismi di difesa;
- funzioni di controllo;

Assurance meccanismi HW/SW valutabili in maniera indipendente:

- garanzie di Policy, Marking, Identification e Accountability;
- identificazione TCB;
- documentazione.

Divisione D

Questa divisione contiene una sola classe riservata a quei sistemi che sono stati valutati, ma che non soddisfano i requisiti di una classe superiore di valutazione.

Divisione C Classe 1

TCB soddisfa i requisiti minimi fornendo la separazione di utenti e dati. Esso incorpora una qualche forma di controllo credibile, in grado di far rispettare le limitazioni di accesso su base individuale, vale a dire, apparentemente adeguato per consentire agli utenti di essere in grado di proteggere le informazioni sensibili:

- Discrectionary Access Control;
- Identificazione e Autenticazione (es. username e password);
- Dominio TCB testato e protetto;
- Sistema per test futuri sulla funzionalità;
- Documentazione amministratore, utente, test e design.

Divisione C Classe 2

I sistemi di classe C2 richiedono tutte le proprietà della classe C1, inoltre è richiesto un controllo di accesso a grana più fine, per rendere gli utenti individualmente responsabili delle loro azioni. Questo è svolto attraverso:

- Discrectionary Access Control elevato (single subject - single object);
- Identificazione univoca e Autenticazione obbligatoria;
- Dati audit protetti;
- Object Reuse: Nessuna informazione sensibile, legata ad azioni precedenti di un soggetto svolte con l'ausilio di un oggetto, deve essere reperibile da ciascun soggetto che ottiene l'accesso allo stesso oggetto, reso nuovamente disponibile dal sistema.

Divisione B Classe 1

I sistemi di classe B1 richiedono tutte le proprietà della classe C2. Inoltre:

- Etichettamento soggetti e oggetti sotto il controllo del TCB;
- Mandatory Access Control;
- Canali di comunicazione I/O singolo livello o multilivello;
- Policy *No Read Up No Write Down* (BLP);
- Isolamento spazio singolo processo;
- Dominio TCB testato (eliminazione delle vulnerabilità) e protetto.

Divisione B Classe 2

I sistemi di classe B2 richiedono tutte le proprietà della classe B1. Inoltre:

- Etichettamento esteso a tutti i soggetti e oggetti ADP;
- Comunication Channel sicuro tra TCB e utenti per login e autenticazione;
- Stima massima banda CC;
- TCB suddivisa in moduli ben definiti e isolati;
- DTLS (Distribuited Top Level Specification): descrizione completa TCB con eccezioni, messaggi di errore e conseguenze;
- CMS (Configuration Management System): mapping consistente sulla documentazione e il codice associato al TCB;
- Sistema relativamente resistente ad attacchi.

Divisione B Classe 3

I sistemi di classe B3 richiedono tutte le proprietà della classe B2. Inoltre:

- Canale sicuro tra TCB e utenti per qualsiasi comunicazione;
- TCB capace di evidenziare un eccesso di violazioni, notificandolo all'amministratore e in caso estremo di effettuare una procedura (la meno distruttiva) per far cessare le violazioni;
- TCB strutturata con meccanismi di protezione semplici, escludendo tutte le componenti non essenziali alla gestione della sicurezza del sistema;
- Trusted recovery: procedura di failback senza compromissione della protezione;
- Sistema fortemente resistente ad attacchi.

Divisione A Classe 1

I sistemi di classe A richiedono tutte le proprietà della classe B2. Inoltre:

- Modello formale della policy chiaramente identificato e documentato, compresa una dimostrazione della sua consistenza;
- FTLS (Formal Top-Level Specification): definizoni astratte delle funzioni svolte dal TCB e dei meccanismi HW e/o FW che sono usati per supprtare domini di esecuzione distinti;
- Legame Policy-FTLS consistente, dimostazioni formali o informali;
- Legame FTLS-TCB consistente, dimostrazione informale;
- Tecniche di analisi formale devono essere utilizzate per identificare e analizzare Covert Channels. L'esistenza di CC residui nel sistema deve essere giustificata.

Classificazione

Linux, come tutti i moderni sistemi operativi di più ampia diffusione è classificato a livello C2, i limiti più significativi sono:

- scarso controllo sull'utilizzo appropriato della memoria e dei meccanismi di comunicazione interprocesso, a vantaggio dell'efficienza;
- complessità del software e conseguente maggiore difficoltà di verifica della correttezza;
- inefficace limitazione dei privilegi di accesso alle risorse, secondo il modello DAC, a vantaggio della semplicità di configurazione;
- assenza di livelli di privilegio intermedi tra l'utente comune e l'amministratore.

Improve

Il primo passo per avvicinare la classe B è certamente quello di abilitare il Mandatory Access Control, così facendo si avrà:

- controllo granulare su tutto il sistema;
- livelli di privilegio intermedi tra utente e amministratore;
- limitazione spazio esecutivo malware;
- policy di sicurezza del sistema.

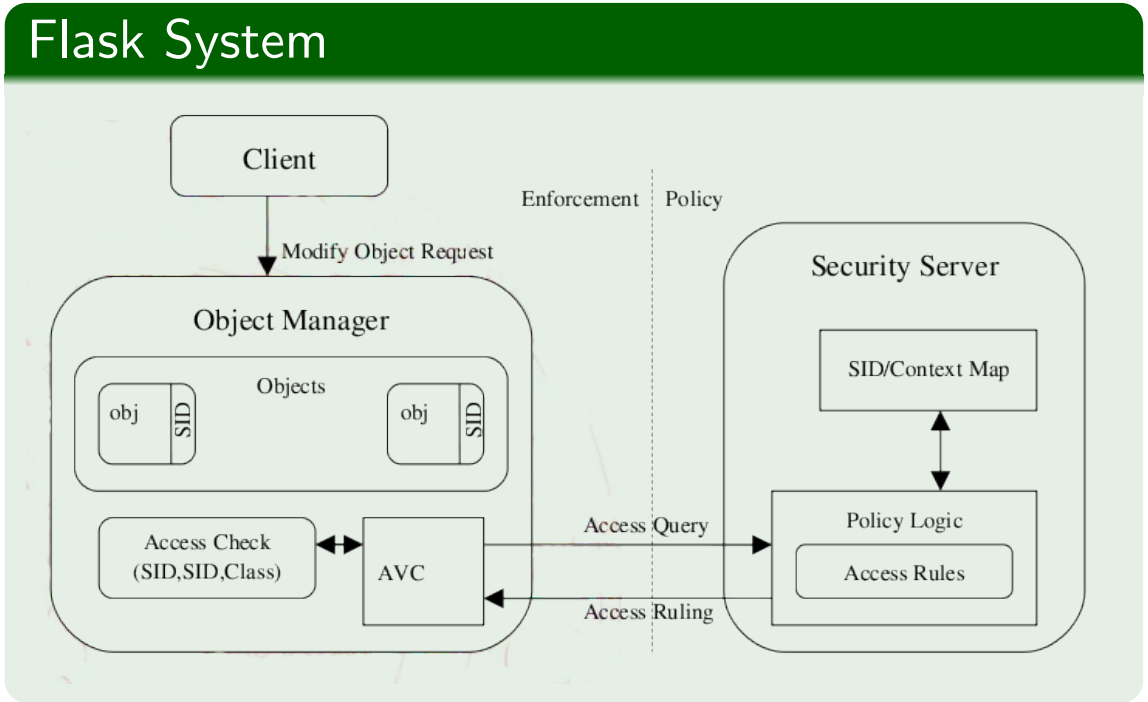
NSA Security-enhanced Linux Team

NSA Security-enhanced Linux è un insieme di patches al Linux kernel e utility per gestire MAC su linux. Esso fornisce un meccanismo per far rispettare la separazione delle informazioni basato su requisiti di riservatezza e integrità, limitando così minacce di manomissione, di elusione dei meccanismi di sicurezza delle applicazioni e possibili danni causati da applicazioni maligne. Esso comprende un insieme di file di configurazione che costituiscono un modello di policy per il sistema.

Architettura FLASK

Quando un soggetto cerca di accedere ad un oggetto, la parte del kernel responsabile all'applicazione della policy controlla un Access Vector Cache (AVC), dove i permessi degli oggetti e dei soggetti vengono conservati. Se non è possibile prendere una decisione in base ai dati contenuti nella AVC, la richiesta viene inoltrata al server di sicurezza, il quale, estrae i dati da una matrice (Context Map). Se l'accesso è rifiutato, viene indicato nel file /var/log/messages.

Flask System



Classificazione

Ogni soggetto/oggetto è classificato per mezzo di

- un numero intero chiamato SID (Security Identifier);
- una terna chiamata Security Context, composta da:
  1. User identity: l'account SELinux associato al soggetto/oggetto, gli account SELinux sono diversi da quelli di sistema, più account di sistema possono essere mappati su uno stesso account SELinux;
  2. Role: il ruolo correntemente utilizzato dall'utente, in ogni istante un utente può utilizzare un solo ruolo;
  3. Type: è l'attributo fondamentale utilizzato da SELinux per prendere le decisioni, si parla di type per gli oggetti e di domain per i soggetti

Le due viste sono messe in relazione 1:1 tramite la Context Map.

Decisioni

Il security server è chiamato a prendere due tipi di decisione:

- decisioni di accesso: determinare se un soggetto può o no svolgere una determinata operazione su di un oggetto;
- decisioni di transizione: determinare quale tipo assegnare ad un oggetto o soggetto all'atto della creazione.

SELinux assume una closed world policy, per cui ogni richiesta di autorizzazione è negata dal security server se non esiste una regola che la consente esplicitamente.

Decisioni di accesso

Ad ogni classe di oggetti è associato un Access Vector, che contiene una regola per ogni azione definita per la classe. Il security server, interrogato sulla possibilità di effettuare una determinata azione su di un oggetto da parte di un soggetto, restituisce le regole, calcolate in base al dominio del soggetto e al tipo e classe dell'oggetto.

Access Vector

File security class										
Append	Create	Execute	Get attribute	I/O control	Link	Lock	Read	Rename	Unlink	Write
?	?	?	?	?	?	?	?	?	?	?

	File security class										
	Append	Create	Execute	Get attribute	I/O control	Link	Lock	Read	Rename	Unlink	Write
Allow	X	X	–	–	–	–	–	–	–	–	–
Auditallow	–	–	–	–	–	–	–	–	–	–	–
Dontaudit	–	–	–	–	–	–	–	–	–	–	–

Decisioni di transizione

Riguardano l'assegnamento di un contesto di sicurezza ai nuovi soggetti ed oggetti:

- Un nuovo processo eredita il contesto del processo padre, ma è possibile una transizione ad un altro dominio:
  - decisa dal processo padre;
  - richiesta dallo stesso processo. [1]
- I nuovi file ereditano il contesto della directory che li contiene, è possibile una transizione ad un altro tipo.

[1] Caratteristica fondamentale che distingue SELinux dalle altre soluzioni è la presenza di un'API che consente di scrivere programmi in grado di interagire con il security server, anzichè solamente subire una policy.

Modalità

Un sistema con SELinux può essere avviato in tre modi:

- Disabled: nessun controllo degli accessi, nessun log, perdita dei SID associati ai file in caso di modifica;
- Permissive:il controllo degli accessi genera unicamente un log delle decisioni di sicurezza, ma non le mette in atto;
- Enforcing: piena funzionalità del sistema di controllo degli accessi.

Installazione su Debian 5.0

L'installazione di SELinux in un sistema Debian Lenny può essere effettuata direttamente dai repository ufficiali con i pacchetti `selinux-policy-default` e `selinux-basics`. Successivamente deve essere lanciato il comando selinux-activate per attivare SELinux.

Bug Fixes

- In `/etc/default/rcS` settare `FSCKFIX=yes` e aggiungere `EDITMOTD=no`;
- In `/etc/init.d/bootmisc.sh` commentare le due righe della sezione Update motd, successivamente lanciare il comando `rm /var/run/motd` e sostituire il link simbolico `/etc/motd` con un file statico con un messaggio message of the day;
- Se `devpts` è montato (`mount | grep devpts`) lanciare il comando `rm -f /dev/[tp]ty[abcdepqrstuvwxyz][0-9a-f]` (SELinux non supporta pseudo-tty old style);
- in `/etc/pam.d/login` modificare la stringa relativa a selinux in `session required pam_selinux.so multiple`;

Bug Fixes

- In `/etc/udev/udev.conf` aggiungere la riga `no_static_dev=1`. Lanciare `update-initramfs -k all -u`;
- In `/etc/cron.daily/standard` commentare le sezioni relative al backup dei file `/etc/shadow` ed `/etc/gshadow` (`cron` non deve poter leggere questi file);
- In `/etc/cron.daily/mlocate` aggiungere all'inizio del file la riga `exit 0` (information leak);
- Lanciare `fixfiles relabel` (N alla prima scelta).

Un pò di pratica

```
$ ls -Z file
```

```
-rwxrw-r– user group unconfined_u:object_r:user_home_t:s0 file
```

Nell'esempio, SELinux prevede un SELinux-user (unconfined_u), un ruolo (objet_r) ovvero l'insieme dei permessi garantiti, un tipo o dominio (user_home_t) ed un livello (s0). Con il DAC, l'accesso è controllato solo in base allo user Linux e all'ID group; le regole SELinux vengono controllate dopo le regole DAC: se queste ultime negano l'accesso, le regole SELinux non vengono usate.

Un pò di pratica

...live session!

Conclusione

Con questa configurazione abbiamo raggiunto il livello B1 di Assurance definito nei TCSEC.

FINE

Grazie per l'attenzione...

FINE

e preparate i seminari!!!

Bibliografia

http://computer-legacy.blogspot.com/.

http://en.wikipedia.org/.

http://web.mit.edu/rhel-doc/4/rh-docs/rhel-rg-it-4/ch-selinux.html.

http://wiki.debian.org/selinux/setup#basics.

http://www.boran.com/security/tcsec.html.

http://www.gentoo.org/proj/it/hardened/selinux/selinux-handbook.xml.

M. Prandini.
Implementazione di modelli di sicurezza evoluti nel sistema operativo linux.
edenti.deis.unibo.it, 2006.

Ray Spencer, Stephen Smalley, Peter Loscocco, Mike Hibler, David Andersen, and Jay Lepreau.
The flask security architecture: system support for diverse security policies.
In *SSYM'99: Proceedings of the 8th conference on USENIX Security Symposium*, pages 11–11, Berkeley, CA, USA, 1999. USENIX Association.