

Progetto e implementazione in FACPL di un monitor a runtime per il supporto al controllo continuativo degli accessi

Candidato:	Filippo Mameli	<code>filippo.mameli@stud.unifi.it</code>
Relatore:	Rosario Pugliese	<code>rosario.pugliese@unifi.it</code>
Correlatore:	Andrea Margheri	<code>andrea.margheri@unifi.it</code>

Riassunto

La tesi propone lo sviluppo di un'estensione, basata su $UCON_{ABC}$, del linguaggio Formal Access Control Policy Language (FACPL) in modo tale da migliorare la gestione delle richieste di accesso a runtime.

Nella prima parte sono state descritte le strutture su cui si è basato lo sviluppo. Sono stati introdotti alcuni modelli di Access Control, il modello $UCON_{ABC}$ e si è analizzato il linguaggio FACPL.

FACPL è stato creato come alternativa al linguaggio eXtensible Access Control Markup Language (XACML) per sviluppare sistemi basati sul modello per il controllo degli accessi Policy Based Access Control (PBAC). Questo è supportato da una libreria Java e ha l'obiettivo di rendere semplice la scrittura di politiche e di definire un framework costruito sopra basi formali solide.

L'obiettivo della tesi è creare un'estensione in grado di modificare il processo valutativo di FACPL. Lo sviluppo è stato diviso in due parti. In primis la struttura del linguaggio è stata modificata di modo che si possa creare delle politiche che si basino sul comportamento passato. In seguito il linguaggio è stato ulteriormente esteso per migliorare la gestione dell'utilizzo continuativo di risorse. In questa tesi si espone il secondo sviluppo, per la prima parte dell'estensione si rimanda alla tesi del mio collega Federico Schipani, con cui ho lavorato per l'implementazione dell'intera nuova struttura.

Il problema principale da risolvere è la staticità del processo di valutazione del linguaggio. Tutte le richieste di accesso ricevono una risposta attraverso un sistema di verifica che esegue sempre tutti i controlli. Non modificando mai i passi della valutazione, vari controlli per le richieste potrebbero risultare ridondanti e in alcuni casi inutili. Le richieste che non modificano i dati o lo stato del sistema, come ad esempio le letture su file, passano per lo stesso processo di valutazione di tutte le altre richieste e il sistema dei controlli risulta eccessivamente complesso per l'autorizzazione di azioni semplici. Per questo motivo è stato ideato un monitor in grado di scegliere a runtime il processo di verifica delle richieste.

La descrizione dell'estensione continua mostrando le modifiche e le strutture aggiunte alla libreria Java che supporta FACPL. È stato mostrato il plugin Eclipse con cui si può trasformare un file scritto in FACPL nei relativi file Java e sono stati presentati due esempi utilizzando la nuova estensione. Nell'ultima parte sono stati mostrati i miglioramenti nell'uso delle risorse computative attraverso una serie di benchmark.