Proof carrying code for service oriented systems

L. Morel & N. Stouls — AMAZONES @ CITI

2009-10

Contexte : L'*approche orientée service* est de plus en plus populaire pour le développement d'applications réparties dynamiques. Elle repose sur la possibilité de charger et décharger à chaud (sur une plateforme en cours d'exécution) des composants qui fournissent à leur environnement un ensemble de services. Un défi de cette approche est qu'il faut pouvoir garantir un niveau de sûreté maximal, malgré le peu de connaissance qu'on a *a priori* des services qui doivent s'exécuter.

La vérification automatique de comportement est une méthode qui permet de vérifier mathématiquement la validité d'une ou plusieurs propriétés concernant le fonctionnement d'un programme, ou dans notre cas d'un service. Cette méthode est avant tout statique : une propriété est vérifiée par le fournisseur du programme concerné, avant que celui-ci ne soit communiqué à son utilisateur. Rien ne garantit à celui-ci que le code satisfait réellement sa propriété. La méthode de proof-carrying code repose sur l'idée qu'un programme n'est pas seulement accompagné de la propriété qu'il est sensé satisfaire, mais d'une preuve de cette propriété. Le système utilisateur peut ainsi "re-jouer" la preuve pour s'assurer que le code qu'il s'apprête à charger satisfait bien la propriété supposée.

Sujet: Cette notion de proof-carrying code a été développée pour des applications java sur une plateforme particulière limitée en taille (JavaCard). Le travail proposé vise à étudier l'application de cette technique au domaine des applications orientées-services et à l'appliquer à un environnement basé sur OSGi.