Sertif : Simulation pour l'Evaluation de la RobusTesse des applications embarquées contre l'Injection de Fautes (ANR-DGA ASTRID 2014 (durée 30 mois))

Marie-Laure Potet*[1], Thanh-Ha Le*[2], and Jessy Clédière*[3]

[1]Laboratoire Vérimag – Grenoble INP – 2 avenue de Vignate 38610 Gières, France
[2]Morpho-Safran – Morpho – France
[3]CEA-LETI – Commissariat à l'Énergie Atomique et aux Énergies Alternatives (CEA) - Grenoble – France

Résumé

ANR-DGA ASTRID 2014 (durée 30 mois).
Partenaires : Vérimag (porteur), CEA-LETI, Safran-Morpho
Site web : http://sertif-projet.forge.imag.fr/fr/

L'objectif du projet SERTIF est de rationaliser et automatiser autant que possible le processus d'analyse de robustesse d'un composant hautement sécurisé à l'injection de fautes, de l'analyse de code à la réalisation physique des attaques, avec comme objectif le passage au multi-fautes qui fait apparaitre une limite aux pratiques actuelles. Pour ce faire le projet SERTIF s'intéressera aux défis suivants : 1) caractérisation de modèles de fautes correspondant aux attaques physiques et mise en oeuvre de ces modèles dans l'analyse de code, 2) définition de critères de robustesse permettant de quantifier une campagne d'analyse de vulnérabilité à l'injection de faute vis-à-vis d'objectifs de sécurité, 3) aide au développement d'applications sécurisées par analyse des contre-mesures présentes dans le code. Les techniques utilisées seront la simulation bas niveau, la mutation de code et plus généralement la combinaison d'analyse statique et dynamique sur du code bas niveau pour maîtriser la combinatoire et qualifier/quantifier les résultats de l'analyse de vulnérabilité à l'injection de fautes.

Les résultats du projet sont des propositions d'ordre méthodologique sur l'analyse de robustesse et l'interaction attaques physiques et analyse de code; les outils développés par les partenaires et la constitution d'un benchmark public d'applications représentatives du domaine, et " durcies " contre l'injection de fautes. Ce benchmark permettra de rendre public et de comparer des résultats d'analyse.

Mots-Clés: injection de fautes, cartes à puce, analyse de code

*Intervenant