UNIVERSITE BRETAGNE LOIRE / MATHSTIC

CentraleSupélec

THESE DE DOCTORAT DE

CENTRALESUPELEC RENNES
COMUE UNIVERSITE BRETAGNE LOIRE

ECOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Informatique*

Par

Thomas Letan

Specifying and Verifying Hardware-based Security Enforcement Mechanisms

Thèse présentée et soutenue à Paris, le 25 octobre 2018
Unité de recherche : CIDRE
Thèse N° : 2018-06-TH

Rapporteurs avant soutenance :

Gilles Barthe	Professor, IMDEA Software Institute
Laurence Pierre	Professeur, Université Grenoble Alpes

Composition du Jury :

Emmanuelle Encrenaz	Maître de conférences, Sorbonne Université
Gilles Barthe	Professor, IMDEA Software Institute
Laurence Pierre	Professeur, Université Grenoble Alpes
Pierre Chifflier	Chef le laboratoire, ANSSI
Guilaume Hiet	Maître de conférences, CentraleSupélec Rennes

Directeur de thèse

Ludovic Mé	Professeur, Inria Rennes

Invité

Alastair Reid	Researcher, ARM Ltd.

Titre : Spécifier et vérifier des stratégies d'application de politiques de sécurité s'appuyant sur des mécanismes matériels

Mots clés : Sécurité ▪ Vérification matérielle ▪ Méthodes formelles ▪ Coq

Résumé : Dans ces travaux de thèse, nous nous intéressons à une classe de stratégies d'application de politiques de sécurité que nous appelons HSE, pour *Hardware-based Security Enforcement*. Dans ce contexte, un ou plusieurs composants logiciels de confiance contraignent l'exécution du reste de la pile logicielle avec le concours de la plate-forme matérielle sous-jacente afin d'assurer le respect d'une politique de sécurité donnée.
Pour qu'un mécanisme HSE contraigne effectivement l'exécution de logiciels arbitraires, il est nécessaire que la plate-forme matérielle et les composants logiciels de confiance l'implémentent correctement. Ces dernières années, plusieurs vulnérabilités ont mis à défaut des implémentations de mécanismes HSE. Nous concentrons ici nos efforts sur celles qui sont le résultat d'erreurs dans les spécifications matérielles et non dans une implémentation donnée.
Plus précisément, nous nous intéressons aux cas particulier de l'usage légitime, par un attaquant, d'une fonctionnalité d'un composant matériel pour contourner les protections offertes par un second.
Notre but est d'explorer des approches basées sur l'usage de méthodes formelles pour spécifier et vérifier des mécanismes HSE. La spécification de mécanismes HSE peut servir de point de départ pour la vérification des spécifications matérielles concernées, dans l'espoir de prévenir des attaques profitant de la composition d'un grand nombre de composants matériels. Elles peuvent ensuite être fournies aux développeurs logiciels, sous la forme d'une liste de prérequis que leurs produits doivent respecter s'ils désirent l'application d'une politique de sécurité clairement identifiée.

Title : Specifying and Verifying Hardware-based Security Enforcement Mechanisms

Keywords: Security ▪ Hardware Verification ▪ Formal Methods ▪ Coq

Abstract: In this thesis, we consider a class of security enforcement mechanisms we called Hardware-based Security Enforcement (HSE). In such mechanisms, some trusted software components rely on the underlying hardware architecture to constrain the execution of untrusted software components with respect to targeted security policies. For instance, an operating system which configures page tables to isolate userland applications implements a HSE mechanism.
For a HSE mechanism to correctly enforce a targeted security policy, it requires both hardware and trusted software components to play their parts. During the past decades, several vulnerability disclosures have defeated HSE mechanisms. We focus on the vulnerabilities that are the result of errors at the specification level, rather than implementation errors. In some critical vulnerabilities, the attacker makes a legitimate use of one hardware component to circumvent the HSE mechanism provided by another one. For instance, cache poisoning attacks leverage inconsistencies between cache and DRAM's access control mechanisms. We call this class of attacks, where an attacker leverages inconsistencies in hardware specifications, compositional attacks.
Our goal is to explore approaches to specify and verify HSE mechanisms using formal methods that would benefit both hardware designers and software developers. Firstly, a formal specification of HSE mechanisms can be leveraged as a foundation for a systematic approach to verify hardware specifications, in the hope of uncovering potential compositional attacks ahead of time. Secondly, it provides unambiguous specifications to software developers, in the form of a list of requirements.