

Réunion du 1^{er} mars 2024

Compte-rendu de réunion de TER

Présents : Pierre DAVID, Léon GALL.

Durée : 9h30 à 10h10.

La réunion du vendredi 1^{er} mars a porté sur les points suivants :

- Retour sur le travail réalisé ;
- Travail à réaliser.

I Retour sur le travail réalisé

Lecture de l'article [4] présentant Nighthawk

- Permet de vérifier l'intégrité du noyau et des structures de l'hyperviseur ainsi que de monitorer l'état des applications (processus, interruptions, cache, ...)
- Avantages :
 - + Pas besoin d'hardware supplémentaire
 - + Haut privilège
 - + Petit TCB
 - + Peu de surcoût
 - + Transparent
- Limites :
 - Performance hautement dépendante de l'IME.
- Utilisation de DMA et HECI pour accéder à la mémoire de l'hôte
- Remappage mémoire pour modifier le code de l'IME.
- Cet outil a pour but de vérifier l'état du système, et non des périphériques.

Lecture de l'article [2] présentant les performances de SPDM

- L'environnement est émulé.

M. DAVID a souligné que puisque les opérations de cryptographie s'exécutent donc sur l'ordinateur hôte, les performances sont meilleures que si elles s'exécutaient sur un périphérique lambda. Il faudrait pouvoir tester sur un réel périphérique ou bien étudier les différentes performances des processeurs spécialisés en cryptographie, qui pourraient être intégrés dans des périphériques.
- Cf. résumé pour les différents résultats en détail
- SPDM a un impact non négligeable sur la vitesse en lecture/écriture séquentielle pour un disque. Lorsque les accès sont aléatoires, l'impact diminue puisque le goulot d'étranglement passe de la cryptographie aux opérations du disque.
- Analyse du code : j'ai trouvé les certificats utilisés, mais pas les clés pré-partagées, malgré de sérieuses recherches.

Lecture de [1] présentant succinctement une implémentation de SPDM par NVIDIA

- NVIDIA voulait implémenter un sous-ensemble des fonctions de SPDM, avec du code prouvé. C'est pour ça qu'ils ont choisi de ne pas utiliser *libspdm*, mais *AdaCore RecordFlux*.
- SPDM est implémenté dans l'adaptateur réseau pour RDMA *ConnectX-7*. Étant un produit haut de gamme, cela donne une idée de la cible de SPDM.

Début de lecture de l'article [3] présentant BitVisor, un hyperviseur chargé d'assurer la sécurité des périphériques

II Travail à réaliser

Lors de la précédente réunion, le sujet avait été élargi. Le travail va maintenant consister à lire de la documentation concernant les points d'élargissement.

De plus, je vais continuer à chercher dans le code des informations concernant le stockage et l'accès aux clés pré-partagées. En particulier, dans la fonction *send_receive_psk_exchange* de la librairie *libspdm*.

III Prochaine réunion

La prochaine entrevue aura lieu le mercredi 13 mars 2024 à partir de 15h30.

Références

- [1] NVIDIA : Using RecordFlux and SPARK to Implement SPDm for Secure Computing. *AdaCore Technical Paper*, February 2023.
- [2] Renan C. A. Alves, Bruno C. Albertini, and Marcos A. Simplicio Jr au2. Benchmarking the Security Protocol and Data Model (SPDM) for component authentication, 2023. _eprint : 2307.06456.
- [3] Takahiro Shinagawa, Hideki Eiraku, Kouichi Tanimoto, Kazumasa Omote, Shoichi Hasegawa, Takashi Horie, Manabu Hirano, Kenichi Kourai, Yoshihiro Oyama, Eiji Kawai, Kenji Kono, Shigeru Chiba, Yasushi Shinjo, and Kazuhiko Kato. BitVisor : a thin hypervisor for enforcing i/o device security. In *Proceedings of the 2009 ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments*, VEE '09, pages 121–130, New York, NY, USA, 2009. Association for Computing Machinery. event-place : Washington, DC, USA.
- [4] Lei Zhou, Fengwei Zhang, Jidong Xiao, Kevin Leach, Westley Weimer, Xuhua Ding, and Guojun Wang. A Coprocessor-Based Introspection Framework Via Intel Management Engine. *IEEE Transactions on Dependable and Secure Computing*, 18(4) :1920–1932, 2021.