Détection de logiciels malveillants dans les applications embarquées.

Directeur : Akka ZEMMARI (zemmari@labri.fr)
Co-Directeur : Mohamed MOSBAH (mosbah@labri.fr)

Contexte

L'informatique embarquée est de nos jours présente partout. Des téléphones, des tablettes, des objets de tous les jours sont de plus en plus équipés de systèmes dits intelligents et interconnectés. De nouvelles thématiques de recherche telles que l'internet des objets (Internet of Things) ont eu un essor fulgurant car ils correspondent à des usages et font émerger de nouveaux défis.

L'utilisation de ces objets (ou applications) est devenue quotidienne, de fait leur popularité et leur diffusion en font des cibles privilégiées pour des personnes mal intentionnées. Le développement de logiciel malveillant pouvant nuire à ces équipements et par conséquent à leurs utilisateurs (risque de fuite de données sensibles, de perte de contrôle sur ces équipements, etc) connaît actuellement une croissance exponentielle.

Détecter de tels logiciels, et donc éviter les dégâts occasionnés, requiert la mise au point de nouvelles techniques d'identification et de catégorisation de ces logiciels. Une de ces pistes actuellement étudiée porte sur l'exploitation du volume de données collectées. Actuellement les techniques les plus performantes se basent sur l'apprentissage et la classification automatique (Machine learning) des données. Dans [AZ15], nous avons proposé une solution basée sur ces techniques pour la détection des logiciels android malveillants. Dans cette thèse, nous nous intéresserons plus particulièrement aux techniques d'apprentissage (Deap Learning), techniques qui s'avèrent particulièrement pertinentes dans le domaine de l'image et de la vidéo [WY13, ZZY12, Y09].

Objectifs de la thèse

L'apprentissage profond (Deep Learning) est une branche récente de l'apprentissage machine [Y09]. Elle a la particularité de traiter les informations en couches successives. Chaque couche fournissant un niveau d'abstraction conceptuelle plus avancée que la couche précédente.

Initialement, le deap learning s'appuie sur des couches réseaux constituées de neurones artificiels, cependant de nouvelles méthodes apparaissent comme les méthodes probabilistes, les « Deep Belief Networks » (DBN) et les « Deep Boltzmann Machines » (BDM) [SS14,SH12]. Tout en préservant l'idée de réseau profond en couches successives, elles substituent aux neurones artificiels des classifieurs stochastiques, inspirés des variantes du modèle de la machine de Boltzmann.

Le but de cette thèse est d'étudier la pertinence de ces modèles d'apprentissage profond dans le cadre de l'analyse statique des « logiciels malveillants » mais aussi dans le cadre dynamique (pendant l'exécution).

Dans le cadre de l'analyse statique, les techniques d'ingénierie inversée (reverse engineering) peuvent être utilisées pour extraire des vecteurs de caractéristiques (« features ») du logiciel dont la dimension est très importante. Une des caractéristiques que nous voulons investiguer dans cette thèse est celle des « chemins typiques » [CDF11]. En effet, le graphe d'exécution du logiciel peut conduire à une explosion combinatoire de l'ensemble des chemins d'exécutions possibles. Ce qui rend l'analyse exhaustive des exécutions non envisageables. L'idée des chemins typiques est d'utiliser une mesure de l'entropie afin de ne conserver que les « chemins importants ».

En se basant sur des approches bayésienne et sur le calcul de l'entropie, il semble possible d'extraire seulement les caractéristiques signifiantes, permettant ainsi une réduction en dimension conservant l'information discriminante [MV15, VV15]. Suite à cette phase de réduction, nous entrainerons avec ces

caractéristiques des modèles d'apprentissage profond tels que les DBF et les FBD et nous en évaluerons la pertinence en termes de classification.

Nous nous intéresserons ensuite à l'analyse dynamique des logiciels embarqués. En plus des caractéristiques collectés dans l'analyse statique nous pouvons extraire d'autres caractéristiques basées sur l'exécution telles que : la consommation d'énergie, les accès à internet, les fichiers ouverts.

La réalisation du projet devra être accompagnée de la mise en place d'un corpus (dataset) suffisamment important pour être utilisé dans la phase d'apprentissage et servir de base de test. Nous attacherons une importance particulière à la phase d'expérimentation et de validation des outils méthodologiques développés.

Bibliographies

[BLSZ16] S. Bhandari, V. Laxmi, M. S. Gaur and A. Zemmari. Intersection Automata based Model for Android Application Collusion. In the 30th IEEE International Conference on Advanced Information Networking and Applications (AINA 2016). Crans-Montana, Switzerland, March 23-25, 2016.

[AZ15] M. Anikeev, S. Bhandari, R. Gupta, M.S. Gaur, V. Laxmi, A. Zemmari. DRACO: DRoid Analyst COmbo, An Android Malware Analysis Framework. In 8th International Conference on Security of Information and Networks (SIN 2015). Sochi, Russia, September 8-10, 2015.

[BMYZ15] W. Ben Jaballah, M. Mosbah, H. Youssef , A. Zemmari. Lightweight Secure Group Communications for Resource Constrained Devices. International Journal of Space-Based and Situated Computing. 5(4):187-200 (2015).

[JCMP13] W. Ben Jaballah, M. Conti, M. Mosbah, C. Palazzi, Fast and Secure Multihop Broadcast Solutions for Intervehicular Communication. IEEE Transactions on Intelligent Transportation Systems, vol.PP, no.99, pp.1,18, doi: 10.1109 / TITS.2013.2277890

[JCMPC13] W. Ben Jaballah, M. Conti, M. Mosbah, C. E. Palazzi. Secure Verification of Location Claims on a Vehicular Safety Application. ICCCN 2013: 1-7, 1EEE (2013)

[MV15] M.J. Meenu, P. Vinod. Hartley's test Ranked Opcodes for Android Malware Analysis.
In Proc. of 8th ACM International Conference on Se- curity of Information and Networks (SIN'15), Sochi, Russia, pp. 304–311, September, 2015.

[CDF11] C. Cui, Z. Dang, T. R. Fischer. Typical Paths of a Graph. Fundam. Inform. 110(1-4): 95-109 (2011)

[VV15] V. Varsha, P. Vinod. Hetrogenous Feature Space for Android Mal- ware Detection. In Proc. of 8th IEEE International Conference on Con- temporary Computing (IC3), August, 20-22, 2015.

[AV14] A. M. Aswini, P. Vinod. Android Malware Analysis Using Ensemble Features.
SPACE 2014: 303-318

[Y09] B. Yoshua. Learning deep architectures for AI. Foundations and Trends in Machine Learning, Volume 2, 2009.

[SS14] N. Srivastava, R. Salakhutdinov. Multimodal learning with deep Boltzmann machines.
Journal of Machine Learning Research 15(1): 2949-2980 (2014)

[SH12] R. Salakhutdinov, G. E. Hinton. An Efficient Learning Procedure for Deep Boltzmann Machines.
Neural Computation 24(8): 1967-2006 (2012)

[WY13] N. Wang, D. Yeung. Learning a Deep Compact Image Representation for Visual Tracking.
NIPS 2013: 809-817

[ZZY12] W. Y. Zou, A. Y. Ng, S. Zhu, K. Yu. Deep Learning of Invariant Features via Simulated Fixations in Video. NIPS 2012: 3212-3220