Log analysis using reinforcement learning for intrusion detection

- Advisors:
  - Olivier Gesny (ogesny@silicom.fr)
  - Samuel Hangouet (samuel.hangouet@intradef.gouv.fr)
  - Frédéric Majorczyk (frederic.majorczyk@irisa.fr)

- keywords: intrusion detection, log analysis, machine learning, reinforcement learning

Subject

One of the main activities inside security operation centers (SOC) is the analysis of logs from applications, operating systems and networks. Two different contexts can lead to a log analysis process: in the case of a security incident and in the hunting activity of a SOC.

First, after a security incident, it is necessary to retrieve the maximum amount of logs on the compromised system and next to analyze them to understand the steps of the attack, which computers are compromised, how they were compromised, which data was exfiltrated, etc. In this case, this analysis has always a starting point that lead to the report of the incident (for example: the abnormal behaviour of a computer).

The hunting activity inside SOCs consists in looking for elements that were missed by traditional detection tools or rules. It is generally done by a log analysis which is different from the previous case: there is no starting point for the analysis, there may not be malicious events during the considered period, logs from the information system are retrieved on a daily basis and thus, the analysts have access to an history of the activity on the system.

Numerous research work [7, 1] proposed the application of machine learning for the detection of anomalous events or entities (machine, user, domaine name, etc.). This gives the analyst a starting point to explore the logs of the information system. Less research work proposed an help to the analysis of events [3]. This internship has the objective to contribute to one of those two questions: detection of anomalous events or entities, or help to the analysis of events.

Among the different machine learning techniques, reinforcement learning (sometimes combined with deep learning) allowed to obtain good results in domains such as games [2, 6]. The use of reinforcement learning in computer security has not been fully explored [4]. The model of reinforcement learning [5, 2] is based on an agent that observes his environment, chooses and does an action, and gets a reward (that can be negative). This model can be applied to the log analysis problem. The agent observes some characteristics of the logs, sends a request and obtains a reward depending on the results of the request.

During this internship, we would like to study reinforcement techniques and their application to log analysis for intrusion detection and develop a PoC on a simple use case.

References

[1] D. Gonçalves, J. Bota, and M. Correia. Big data analytics for detecting host misbehavior in large logs. In *Trustcom/BigDataSE/ISPA, 2015 IEEE*, volume 1, pages 238–245. IEEE, 2015.

[2] S. Kelly and M. I. Heywood. Emergent tangled graph representations for atari game playing agents. In *European Conference on Genetic Programming*, pages 64–79. Springer, 2017.

[3] K. Pei, Z. Gu, B. Saltaformaggio, S. Ma, F. Wang, Z. Zhang, L. Si, X. Zhang, and D. Xu. Hercule: Attack story reconstruction via community discovery on correlated log graph. In *Proceedings of the 32Nd Annual Conference on Computer Security Applications*, pages 583–595. ACM, 2016.

[4] N. Sengupta, J. Sen, J. Sil, and M. Saha. Designing of on line intrusion detection system using rough set theory and q-learning algorithm. *Neurocomputing*, 111:161–168, 2013.

[5] R. S. Sutton and A. G. Barto. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

[6] H. Van Hasselt, A. Guez, and D. Silver. Deep reinforcement learning with double q-learning. In *AAAI*, volume 2, page 5. Phoenix, AZ, 2016.

[7] T.-F. Yen, A. Oprea, K. Onarlioglu, T. Leetham, W. Robertson, A. Juels, and E. Kirda. Beehive: Large-scale log analysis for detecting suspicious

activity in enterprise networks. In *Proceedings of the 29th Annual Computer Security Applications Conference*, pages 199–208. ACM, 2013.