Towards Generation of Attack Trees using Machine Learning

Kacper Sowka [1], Madeline Cheah [2], Trang Doan [3], Hoang Nga Nguyen [1], Siraj Shaikh [1]

[1] *Institute for Future Transport and Cities, Coventry University, Coventry, UK*
[2] *HORIBA MIRA, Nuneaton, CV10 0TU, UK*
[3] *Faculty of Engineering, Environment and Computing, Coventry University, Coventry, UK*

PhD start date: 21/09/2020, Expected submission date: 31/04/2024

A comprehensive cybersecurity evaluation of automotive on-board networks has become a crucial antecedent to the commercial distribution of vehicles. However, the means to perform the required testing are limited due to the black-box nature and complexity of automotive systems. To rectify this, several approaches have been put forward to systematise and automate the process of testing vehicular systems, but these still require a significant amount of expert input to build test cases. Accordingly, this work aims to further automate the process by introducing a machine learning based generation scheme for data which can then be used to facilitate the creation of further tests.

1 Introduction

Vehicles are being extended with various features to increase usability and implement various quality of life features. Cars now contain between 50 and 70 electronic control units (ECU) along with a significant amount of code [1]. A large number of interconnected units from different sources means that precise specifications of the system are not available [2].

To assure cybersecurity in such systems, black-box experience-based techniques such as penetration testing are useful. Penetration testing aims to enumerate cybersecurity vulnerabilities by compromising the system under test with the help of a domain expert[2]. Many authors have performed penetration tests on vehicles [1], [3], [4], but this approach comes with caveats regarding safety, cost and time. To surmount these difficulties, simulated penetration testing is an active area of research [5] (though not within the scope of this paper).

1.1 Overview of proposed work

In experience-based testing methods, such as penetration testing, an approach known as "error guessing" is often used. This involves inferring vulnerabilities that may exist in the system-under-test using expert intuition based on experience, then using these inferences to derive test cases [6]. There has been previous work in literature for doing this in a semi-automated fashion [7]. However, to achieve better coverage, full automation is ideal.

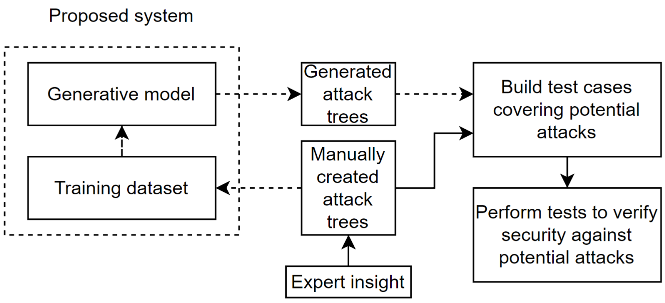
Proposed system
Generative model
Training dataset
Generated attack trees
Manually created attack trees
Expert insight
Build test cases covering potential attacks
Perform tests to verify security against potential attacks

Figure 1

To that end, this work proposes that automated error guessing can be facilitated through the automatic generation of attack trees. Attack trees [8] are a useful formalism for representing potential attacks in a structured and intuitive manner. They have seen use as input to automated test case generation [7], [9] and therefore their automatic generation will further automate the process and extend test coverage.

An overview of this can be seen in Figure 1, where dotted arrows represent new approaches opened up by the proposed system and solid arrows show the existing methodology. The proposed system doesn't aim to replace the existing approach, but to complement it to ease the human burden and expand the coverage of the tests.

2 Work to be done

This work posits that recent advances in the field of labelled graph generation with machine learning [10]–[13] could feasibly be applied towards generating attack trees. Since the attack tree represents an opinion on where vulnerabilities might be found, their generation represents the automation of error guessing, and machine learning methods can simulate experience by teaching the model on a dataset of other attack trees.

But in order to feasibly generate a valid attack tree from a dataset of other attack trees, various obstacles must be overcome:

- Firstly, the training dataset is a significant concern as attack trees are difficult to produce. They usually require one or more domain experts to construct, hence the motivation to automatically generate in the first place, and coupled with the high information barriers involved in producing them there is a scant amount of training data to work with.
- Secondly, ensuring that the learning doesn't simply produce surface level imitations of attack trees. This would require finding a method to facilitate the learning of semantically rich relationships between nodes and furthermore, be able to exploit those relationships to produce novel data such as new paths through which attacks can be executed.
- Finally, validation is a challenge as generated trees must adhere to a given set of requirements. A reliable method of validation must be ascertained to ensure these requirements are being met by the generated attack trees.

The rough timeline for the research proceeds as follows: first, the relevant literature must be examined for guidance on how the above obstacles can be surmounted (see related work section for a glance at progress thus far). This will result in an initial literature review being drafted by March 2021. The first forays into drafting a methodology for attack tree generation will begin in February 2021 with the aim to finish and set a clear direction for further research by June 2021, alongside developing a comprehensive validation strategy by the same deadline.

The latter half of 2021 will be dedicated to further developing the methodology and validation strategy and preparing a practical implementation of the chosen methodology for evaluation. The goal by June 2022 is to evaluate the methodology with at least 1 automotive interface (such as Wi-Fi) being covered, with the results and analysis being published.

It is difficult to clearly ascertain the direction and timeframe of further research as it is heavily contingent on the initial results, but the ultimate goal is to extend the applicability of this method to multiple automotive interfaces by the latter half of 2023, with the final thesis being ready by the expected submission date in April of 2024.

3 Related work

Although there are existing generation schemes for attack trees [14]–[16], these rely on formal definitions of the system-under-test and are not well suited for the error guessing task outlined here. This is because they rely on faithful representations of the system itself, unsuitable to a black box approach in general [2], and to the task of automated testing specifically, since existing test methods rely on attack trees precisely because system information is unavailable [7]. Nevertheless, many of the formal methods introduce useful insight regarding the semantic validity of attack trees and so could offer insight into the validation of generated trees.

Looking towards generative machine learning models, of particular interest are Generative Adversarial Networks [17] and Variational AutoEncoders (VAE) [18] due to their ability to learn rich latent representations of a dataset in an unsupervised fashion. Used in conjunction with generative schemes, approaches for modelling sequences such as Recurrent Neural Networks and Graph Neural Networks have the capacity to learn generative models of graph structured data [13], [19]–[21] and thus can be a potential starting point for the generation of attack trees.

A closely related problem is that of molecule generation, as it considers generation of graph structured data with strict semantic requirements [10], [11]. Several key insights such as the use of reinforcement learning and regularization in order to encourage semantically valid generation are of particular value as attack trees must follow a specific set of requirements, much like molecules, to be considered "valid".

The dataset problem could be mitigated using work done on few-shot multi label classification for graphs [22], based on casting short textual descriptions of the nodes into a vector representation corresponding to classes assigned to the nodes. This can also significantly simplify the issues relating to generation of novel graphs by reducing the dimensionality of the problem from generating rich descriptions to generating correctly classified nodes.

In the context of the existing work and the requirements outlined above, this research will aim to utilise existing advances in labelled graph generation, semi-automated testing and threat modelling in order to facilitate the further automation of testing automotive onboard networks. The ultimate goal is the generation of attack trees which can be used in order to produce test cases in an automated manner, which will help ensure the cybersecurity of automotive on-board system by providing wider coverage of test cases and easing the human burden involved in creating test cases.

Bibliography

[1] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D.

Anderson, H. Shacham, and S. Savage, "Experimental Security Analysis of a Modern Automobile," in *2010 IEEE Symposium on Security and Privacy*, IEEE, 2010, ISBN: 978-1-4244-6894-2. DOI: 10.1109/SP.2010.34.

[2] M. Cheah, S. A. Shaikh, J. Bryans, and H. N. Nguyen, "Combining third party components securely in automotive systems," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9895 LNCS, Springer Verlag, 2016, pp. 262–269, ISBN: 9783319459301. DOI: 10.1007/978-3-319-45931-8_18. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-45931-8{_}18.

[3] S. Checkoway, D. Mccoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno, "Comprehensive Experimental Analyses of Automotive Attack Surfaces," in *Proceedings of the 20th USENIX Conference on Security*, USENIX Association, USENIX Association, 2011. [Online]. Available: https://www.usenix.org/legacy/events/sec11/tech/full{_}papers/Checkoway.pdf.

[4] C. Miller and C. Valasek, "Remote Exploitation of an Unaltered Passenger Vehicle," Black Hat USA, Tech. Rep., 2015.

[5] J. Hoffmann, "Simulated Penetration Testing: From "Dijkstra" to "Turing Test++"," *Proceedings International Conference on Automated Planning and Scheduling, ICAPS*, pp. 364–372, 2015. [Online]. Available: https://fai.cs.uni-saarland.de/hoffmann/papers/icaps15inv.pdf.

[6] ISO, *Iso/iec/ieee international standard - software and systems engineering–software testing–part 4: Test techniques*, 2015. DOI: 10.1109/IEEESTD.2015.7346375. [Online]. Available: https://ieeexplore.ieee.org/document/7346375.

[7] M. Cheah, H. N. Nguyen, J. Bryans, and S. A. Shaikh, "Formalising systematic security evaluations using attack trees for automotive applications," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10741 LNCS, Springer Verlag, 2018, pp. 113–129, ISBN: 9783319935232. DOI: 10.1007/978-3-319-93524-9_7. [Online]. Available: https://doi.org/10.1007/978-3-319-93524-9{_}7.

[8] S. Mauw and M. Oostdijk, "Foundations of Attack Trees," *Lecture Notes in Computer Science*, vol. 3935, pp. 186–198, 2006.

[9] M. Cheah, S. A. Shaikh, O. Haas, and A. Ruddle, "Towards a systematic security evaluation of the automotive Bluetooth interface," *Vehicular Communications*, vol. 9, pp. 8–18, 2017, ISSN: 22142096. DOI: 10.1016/j.vehcom.2017.02.008. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2214209616301474.

[10] T. Ma, J. Chen, and C. Xiao, "Constrained Generation of Semantically Valid Graphs via Regularizing Variational Autoencoders," 2018. arXiv: 1809.02630. [Online]. Available: https://arxiv.org/abs/1809.02630.

[11] N. de Cao and T. Kipf, *MolGAN: An implicit generative model for small molecular graphs*, 2018. arXiv: 1805.11973.

[12] D. Bacciu, A. Micheli, and M. Podda, "Edge-based sequential graph generation with recurrent neural networks," *Neurocomputing*, vol. 416, 2020, ISSN: 09252312. DOI: 10.1016/j.neucom.2019.11.112.

[13] S. Fan and B. Huang, "Conditional Labeled Graph Generation with GANs," in *Iclr*, 2019, pp. 1–26. [Online]. Available: https://rlgm.github.io/papers/22.pdf.

[14] O. Gadyatskaya, R. Jhawar, S. Mauw, R. Trujillo-Rasua, and T. A. C. Willemse, "Refinement-Aware Generation of Attack Trees," in *Security and Trust Management*, 1st ed., vol. 10547, Springer International Publishing, 2017. DOI: 10.1007/978-3-319-68063-7_11.

[15] K. Karray, J.-L. Danger, S. Guilley, and M. Abdelaziz Elaabid, "Attack Tree Construction and Its Application to the Connected Vehicle," in *Cyber-Physical Systems Security*, Springer International Publishing, 2018. DOI: 10.1007/978-3-319-98935-8_9.

[16] J. Bryans, L. S. Liew, H. N. Nguyen, G. Sabaliauskaite, S. Shaikh, and F. Zhou, "A Template-Based Method for the Generation of Attack Trees," in *Information Security Theory and Practice*, Laurent Maryline, and T. Giannetsos, Eds., Cham: Springer International Publishing, 2020, pp. 155–165, ISBN: 978-3-030-41702-4. DOI: 10.1007/978-3-030-41702-4_10. [Online]. Available: https://link.springer.com/chapter/10.1007{\%}2F978-3-030-41702-4{_}10.

[17] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, *Generative Adversarial Networks*, 2014. arXiv: 1406.2661. [Online]. Available: https://arxiv.org/abs/1406.2661.

[18] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," International Conference on Learning Representations, ICLR, Dec. 2014. [Online]. Available: `https://arxiv.org/abs/1312.6114v10`.

[19] J. You, R. Ying, X. Ren, W. L. Hamilton, and J. Leskovec, "GraphRNN: Generating realistic graphs with deep auto-regressive models," in *35th International Conference on Machine Learning, ICML 2018*, vol. 13, 2018, pp. 9072–9081, ISBN: 9781510867963. arXiv: `1802.08773`.

[20] M. Simonovsky and N. Komodakis, *GraphVAE: Towards Generation of Small Graphs Using Variational Autoencoders*, 2017. arXiv: `arXiv:1802.03480v1`.

[21] H. Wang, J. Wang, J. Wang, M. Zhao, W. Zhang, F. Zhang, X. Xie, and M. Guo, "Graphgan: Graph representation learning with generative adversarial nets," in *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 2018, pp. 2508–2515, ISBN: 9781577358008. arXiv: `1711.08267`. [Online]. Available: `https://arxiv.org/abs/1711.08267`.

[22] A. Rios and R. Kavuluru, "Few-shot and zero-shot multi-label learning for structured label spaces," Association for Computational Linguistics, 2018, pp. 3132–3142. DOI: `10.18653/v1/D18-1352`. [Online]. Available: `http://aclweb.org/anthology/D18-1352`.