Automatic Yara Rule Generation Using Biclustering

For this paper, there was no presentation but instead a discussion with the professor regarding its contents. The paper deals with generating YARA rules automatically using Biclustering.

- A YARA rule is a framework that is used for pattern matching, extensively in the domain of both machine learning as well as malware detection. In fact, the premium tier of Virus Total allows the users to apply a YARA rule against the whole dataset they have in order to find the malware files that fall under a certain criteria that you are looking for. Conventionally, the YARA rule is created manually by the human analysts who analyze a new malware file and identify the important parts that can be used to identify it. This is an expensive step that takes a lot of effort and does not scale well into large numbers. This is one of the reasons why methods such as the one presented in the paper are important. The advantage of having a rule is that the client does not require a machine learning system in their machine but just a set of rules to match against the files for malware detection. Even though this entire course has been about using machine learning for the purpose of detecting malware, in reality rule based detection is still employed by the AntiVirus companies and YARA rules are the standard way of targeting specific malware files.
- A specific context where techniques like these comes in handy is for zero day attacks. In case of a zero day attack, it will take time for the machine learning based detection systems to have enough data to retrain their models enough to deal with the new malware type. Until then, techniques like YARA rule can be used to stem the flow of attacks. However, YARA rules are inherently linked specifically to one malware file, and hence do not generalize as well as a machine learning based approach. This means that the malicious actors would be able to create variants to bypass the rule based detection. It is still better to have the rules in the client machines to reduce the attacks the client might face, and to force the malicious actors to come up with new variants which could eventually be used by the machine learning models to deal with the concept drift caused by the zero day attack. For these reasons you could observe the antivirus engines making a lot of updates in a short span, which could likely be linked to the company changing their rules to deal with a zero day attack. This would mostly be the company introducing new rules, and then later on taking off the rules that would be causing false positives. In fact, for an Anti Virus companies techniques such as YARA rules can be achieved easily as they already have access to enough benign files to test their YARA rules against to ensure that the false positive rate would be low enough.
- Rules such as these can also be used to hunt through a network to see the infections, once you identify one infection in the network. Using that one malware file that was identified, you could create a YARA rule and use it to see how many other devices are infected.
- The major concern to keep in mind while creating YARA rules is to reduce the probability of it matching with goodware files. This is along the same lines as prioritizing reducing False Positive Rates when it comes to creating machine learning models for malware detection. Security should

not come in the way of usability because if it does, it encourages the users to get rid of the security measures.

- Another interesting fact to note is how the YARA rules created by the machine learning techniques are more complex and are in general longer than the one created by human analysts. This is because a human being would create a rule based on logical reasoning and hence the rule itself would have a level of interpretability to it. On the other hand, a machine learning technique would just use statistics to figure out the parts that could be used to signature a specific malware file. This need not look understandable to a human being at all. However employing machine learning for these tasks still help out immensely to bring down the cost of the expensive process of manually creating YARA rules.
- Another difference between machine learning models and rule based detection is that in a machine learning system, it's normally just the parameters that get updated with more data being used for the training process. On the other hand, in a rule based system the number of rules keep getting longer linearly and hence mgight require larger resources in, say, the client machine for a rule based system depending on the number of rules there are. The same issue would be there in hardware based detection engines. The question of what would take more resources - YARA rules or machine learning models would depend on the number of rules there are in the system. For this reason, most systems try to keep the number of rules to as low as possible. Also, probabilistic comparisons like Bloom Filter can be used to reduce the resource utilization.
- Large Language Models can also be used in the domain of YARA rule generation and it is something that has not been explored as well as it should be. Even though the LLM might not work very well for this task in a zero-shot setting, with a couple of levels of fine tuning it would be possible to have it generate YARA rules efficiently.
- The professor mentioned a research he was doing where a signature could be created by keeping track of a window of the last branch actions (taken/not-taken). Creating this signature would not cost additional resources, as it can be done along with the operating system's regular process and it could be utilized like any other signature systems for identification.
- Also, the YARA rules are not really matched one by one in a linear fashion. Other algorithms and data structures usually come into the mix here, where trees and prefix-based comparisons are utilized with an intermediate representation of the rules to change the running time of the process from linear to sub linear. This is also what tools like grep use in order to speed up their process.
- A whole new strategy that replaces everything in the pipeline and replace it with something in a new direction is very hard for the industry to do. So, it is better if new strategies adapt well into the already existing pipeline. So instead of machine learning completely overhauling the AV rule based pipeline, integrate it into the mix. Machine learning that generates rules is something like this.