Malware Detection by Eating a Whole EXE

Jeff Malavasi
Dept. of Computing Security
Rochester Institute of Technology
Email: jm3378@rit.edu

I. Review

The authors of the paper investigate the drawbacks of current automated malicious software analysis that focuses on signature based detection. They propose a new technique that uses machine learning to analyze the raw byte sequence of a binary. While these systems have been successful at labeling byte sequences and images, the diverse nature of malware requires a novel architecture to be accurate. This work expands on the current use of n-gram string sequences to classify binaries, which can easily be evaded by malware programmers.

The researchers used two different datasets to train and test their model. The first was comprised of an even split of malicious and benign samples, whereas the second set had double the amount of malicious samples. However, it was found that the second dataset resulted in a biased model, so the researchers omitted it from further testing [1]. During training, the researchers favored features that were able to scale as the byte sequence grew, as they found that smaller subsets resulted in less accurate results. Additionally, their architecture needed to be resistant to the variability in portable executables. For example, most binaries can be rearranged without significantly changing their function. Due to this, they selected a convolution network that analyzed the binary in blocks of 500 bytes. Although it may seem intuitive that testing the entire file at once would be the most accurate, the researchers found that this technique was unable to generalize new binaries due to the fact that many pieces of malware contain a large volume of junk code [1].

Ultimately the researchers selected a MalConv algorithm that uses DeCov regularization to improve the accuracy of the model [1]. Through manual analysis, they determined that the model utilized features from many areas of the binary such as the PE-Header, the executable code, and UPX1 which indicates whether or not the binary is packed. Finally, the researchers attempted to use batch normalization to further improve their model. While batch normalization historically has complimented MalConv models, the researchers observed an inverse effect. This is likely due to the fact that byte sequences are multi-modal, meaning that the same byte value can have multiple purposes [1].

In this paper the authors propose an accurate malware detection system based on the analysis of byte sequences. The researchers tested a large variety of different architectures, which helped them to hone in on the best model. However, the authors were unable to train a model on larger sequences of bytes due to the high amount of memory requirements needed to process the convolutions. When using smaller samples, it makes it easier for a bad actor to use polymorphism to evade detection.

Many researchers are focused on creating malware detection systems that utilize machine learning in order to combat the increased sophistication of evasion techniques. By creating a model that can predict whether or not a new file is benign or dangerous, defenders can get ahead of zero day attacks.

Nonetheless, the proposed model is not entirely infallible. The authors acknowledge the need to rebuild the architecture in order to reduce memory consumption, which would allow for the analysis of larger chunks of bytes. Additionally, they failed to use batch processing to normalize the data. This is necessary to reduce noise in the dataset, which will in turn improve accuracy.

As malware continues to gain sophistication, novel techniques are required to classify binaries. The researchers successful create a neural network that can classify binaries based on byte sequence analysis. Further research will be needed to improve the accuracy of these models.

References

[1] E. Raff, J. Barker, J. Sylvester, R. Brandon, B. Catanzaro, and C. Nicholas, “Malware detection by eating a whole exe,” in *arXiv*, 2017.