**Project Proposal – Team 2**

Esteban Guillen, Thomas LaBruyere, William Collishaw

**Topic: Machine Learning Techniques for Classifying Malware**

**Research Statement**

The explosion of malware in recent years has created an enormous burden on the cyber security community and threatens to disrupt our way of life. It is estimated that over 100,000 new malware threats are detected every day. In an attempt to avoid detection by anti-virus software the author of malware will design their malicious code to be polymorphic and metamorphic which allows it to change as it propagates. Traditional anti-virus software is signature-based which requires the structure of the malicious code to be know in advance. Malware that can change as it propagates across the internet becomes nearly impossible for the best in class anti-virus systems to detect. Cyber security professionals need better tools to detect and classify previously unseen malicious code.

**Research Procedure**

Machine learning techniques have emerged as a promising method for classifying malware into families (similar behavior). Classification analysis techniques can be grouped into two main categories, static and dynamic. For this project we will perform static analysis using machine learning methods to classifying malware. We will implement the machine learning approaches (Association Rules, Support Vector Machines, Decision Trees, Random Forests, Naïve Bayes, Clustering, etc.) outlined by leading cyber security professional and researchers. In addition, we hope to identify new features (not identified in the research literature) of malware code which can be used by machine learning algorithms to classify malicious code by family.

Microsoft published a large malware data set in 2015 and has made it available for academic research. The large (500+ GB) data set consists of of over 20,000 malware samples. Each sample has an associated ASM file (Figure 1) and hex (hexadecimal representation of the binary content) file (Figure 2). The samples are also labeled by malware family. From the ASM and hex files we will extract a set of features (byte-sequence n-grams, library calls, control flow graph, opcode frequency distribution, etc.) that will be used to train our machine learning algorithms. Half of the data set samples will be used for training our models and the other half will be used for testing (evaluation). The evaluation criterial for our models will be based on the accuracy of classifying the testing samples into their appropriate families. Open source machine learning libraries will be use and where appropriate we will implement our own algorithms.

**Conclusion**

Effective malware classification techniques are needed to help combat the growing threat that malicious code poses. This project will evaluate the leading machine learning techniques for static malware classification and we hope to find new approaches for extracting features (to train on). This work can lead to the creation of powerful new tool that can be used to detect malware before it can do harm to a network.

Table . System requirements

|  |  |
| --- | --- |
| Operating system | Windows, Linux, or Mac |
| CPU | Core i7 2.4 GHz |
| Memory | 8 GB RAM |
| Hard drive | 1 TB |
| Software | Python, scikit-learn, numpy, pandas |
| Cloud (if local resources are inadequate) | AWS (EC2 and EBS) |

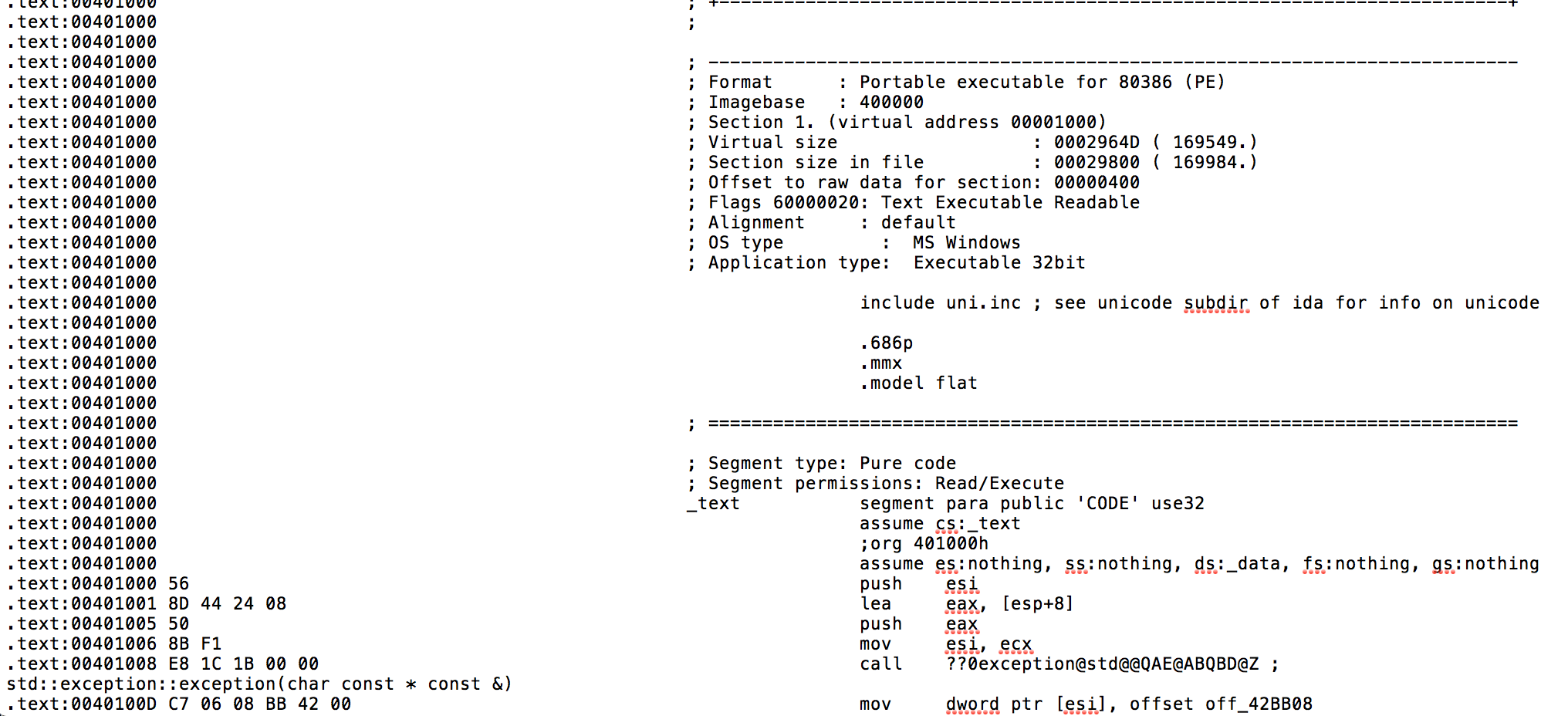


Figure . ASM file for malware

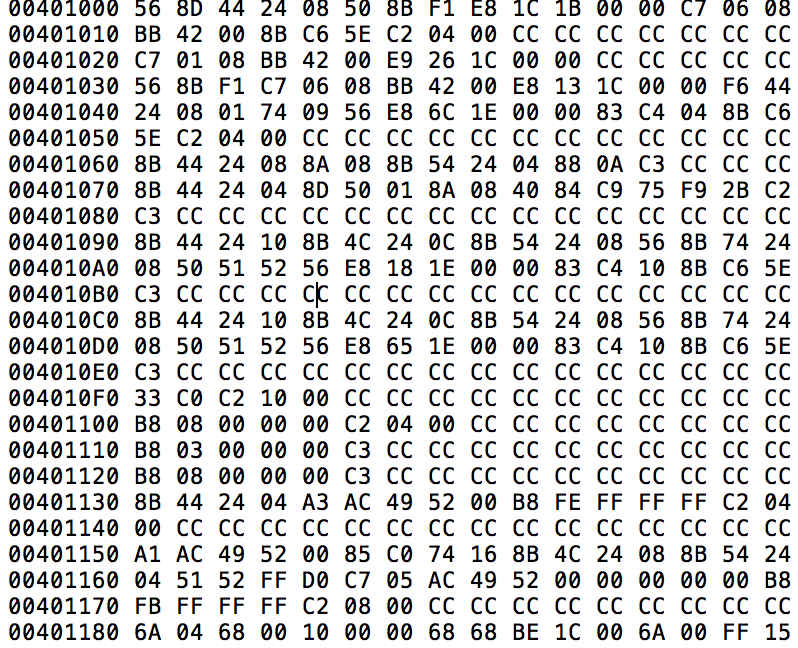


Figure . Hexadecimal representation of malware binary content