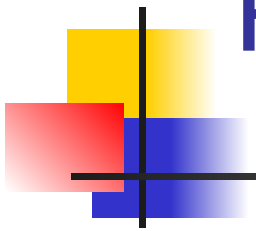
Generation of Malware Samples Using Deep Autoencoders

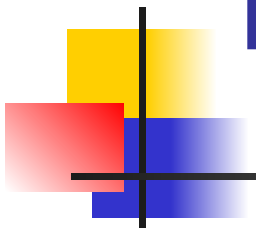
Project Advisor: Fabio Di Troia
Team Members: Aaron Choi
Albert Giang
Sajit Jumani
David Luong

Project Description



- Generate malware opcode sequences with a deep variational autoencoder (VAE) across multiple malware families as input.
  - If time permits, may add Hidden Markov Models (HMMs), or other Generative Adversarial Networks (GANs) like EBGAN, LSGAN, or LSTM-GAN
- Evaluate the classification metrics (P, R, F1) of four machine learning (ML) classifiers that differentiate between generated (i.e. fake) and real malicious code.
- Determine if the deep autoencoder produced more realistic synthetic opcode malware compared to Wasserstein Generative Adversarial Networks with Gradient Penalty (WGAN-GP)

Project Deliverables



- Project code that uses:
  - VAE and GAN libraries to generate malware opcodes
  - ML classifiers to differentiate between fake and real samples
- Results of ML classifiers
  - Support Vector Machines (SVM)
  - k-Nearest Neighbor (k-NN)
  - Random Forest
  - Naive Bayes

Project Dependencies and Concerns

- Dependencies
  - GPU access for training
- Concerns
  - VAE generates easily distinguishable malware
  - Determining convergence criteria during training

Draft Architecture