

# **System Security Modeling of Feature Selection and Behavior Analysis for Efficient Malware Detection**

**Presentation (#18914) for:**

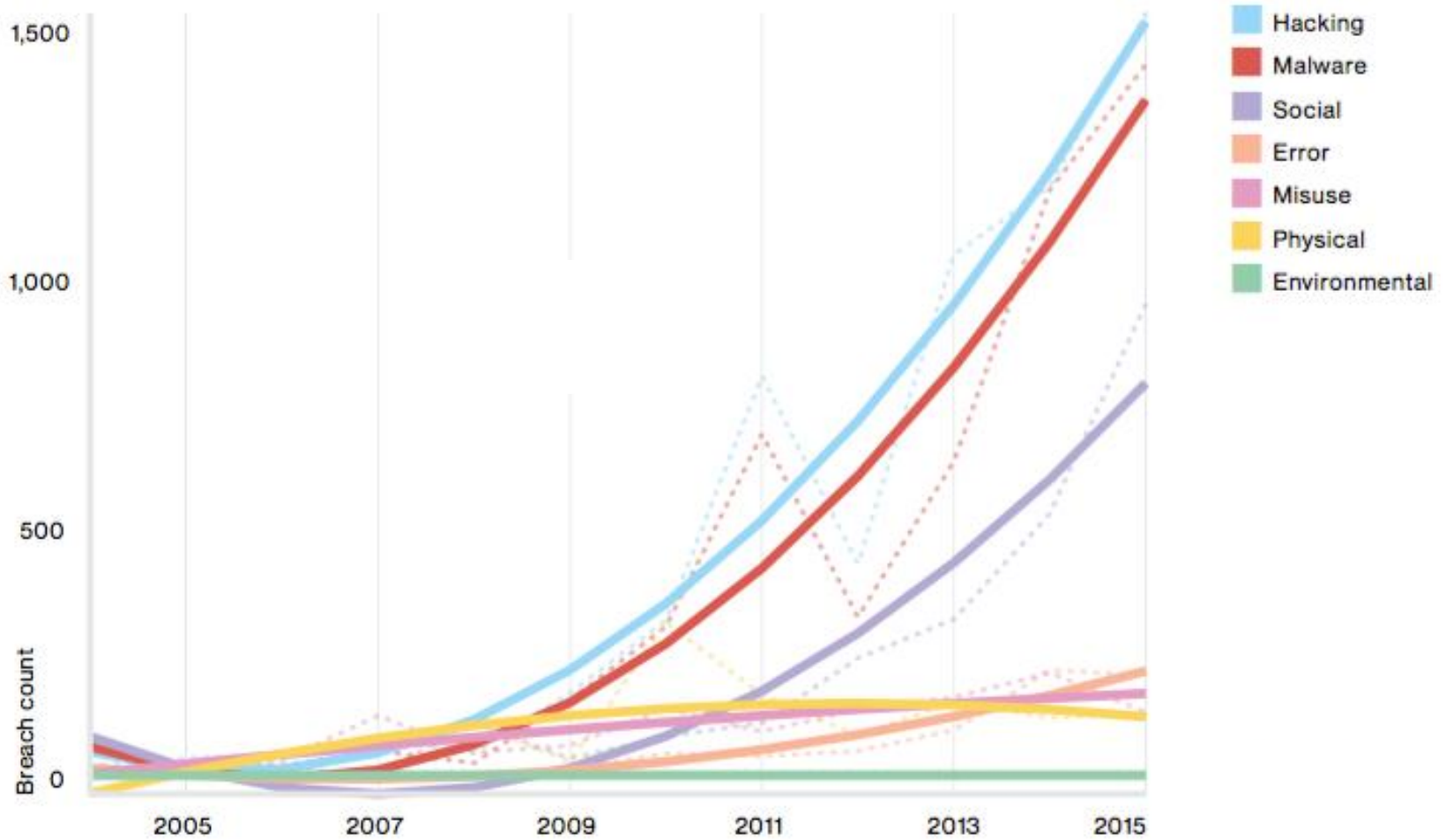
**National Defense Industrial Association**

**19<sup>th</sup> Annual Systems Engineering Conference**

**October 24-27, Springfield, VA.**

**Joseph W. Mikhail, P.E.**

**George Washington University**



**Figure 1: Growing number of breaches over time**

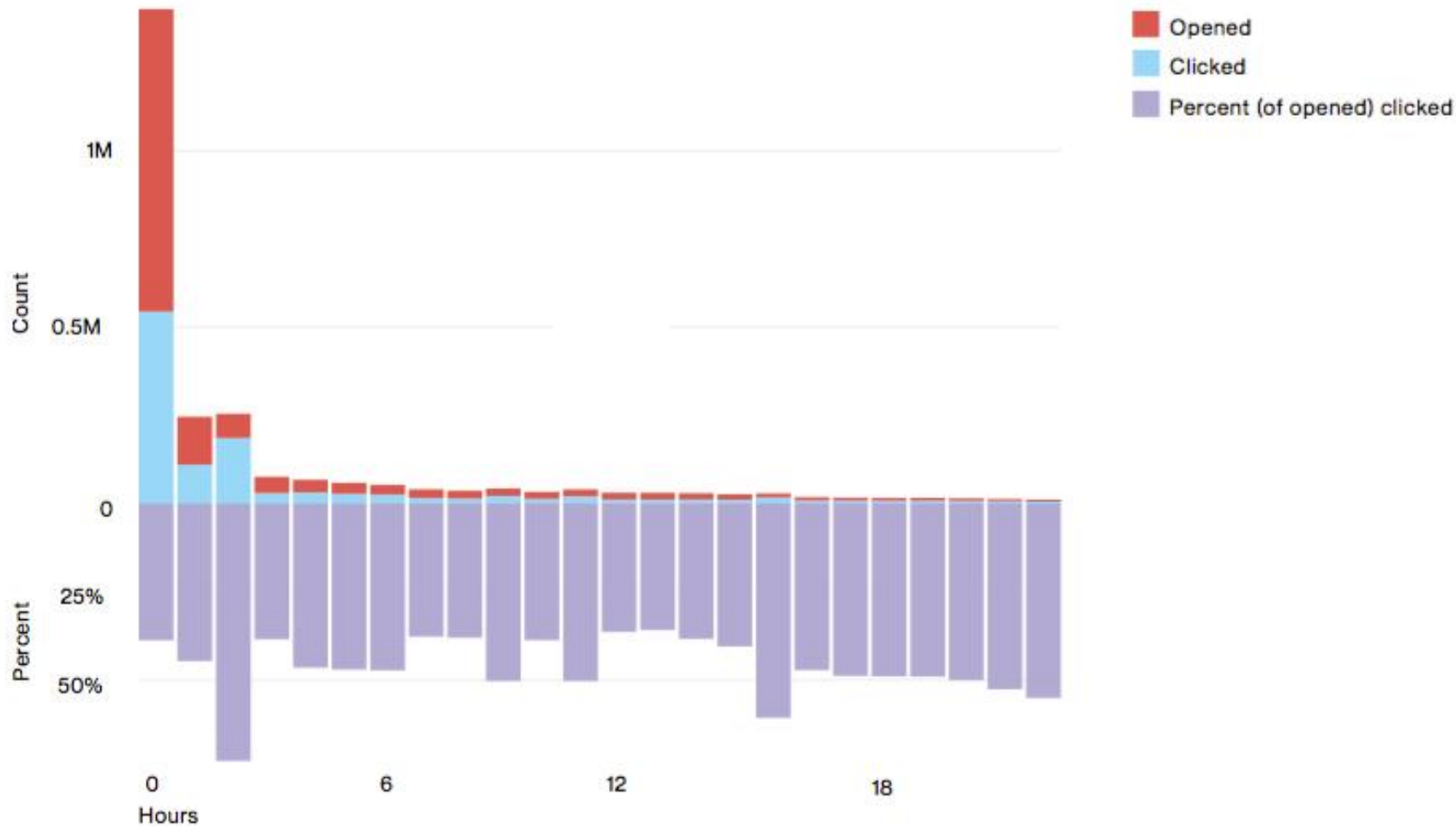


Figure 2: Percentage of phishing emails resulting in “clicks”

\*Source: [www.verizonenterprise.com/resources/reports/rp\\_DBIR\\_2016\\_Report\\_en\\_xg.pdf](http://www.verizonenterprise.com/resources/reports/rp_DBIR_2016_Report_en_xg.pdf)

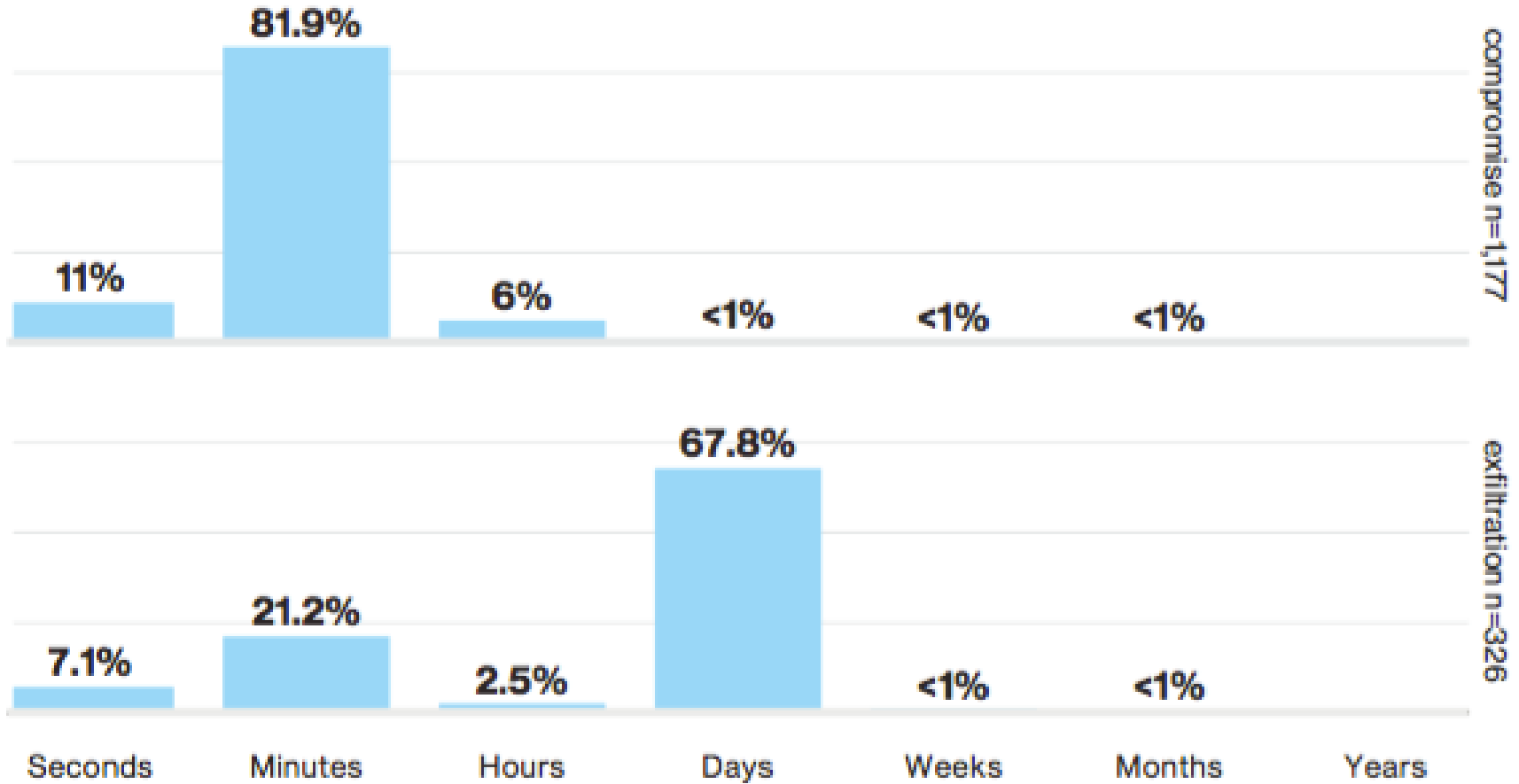


Figure 3: Average compromise and exfiltration times

## Trends and Numbers

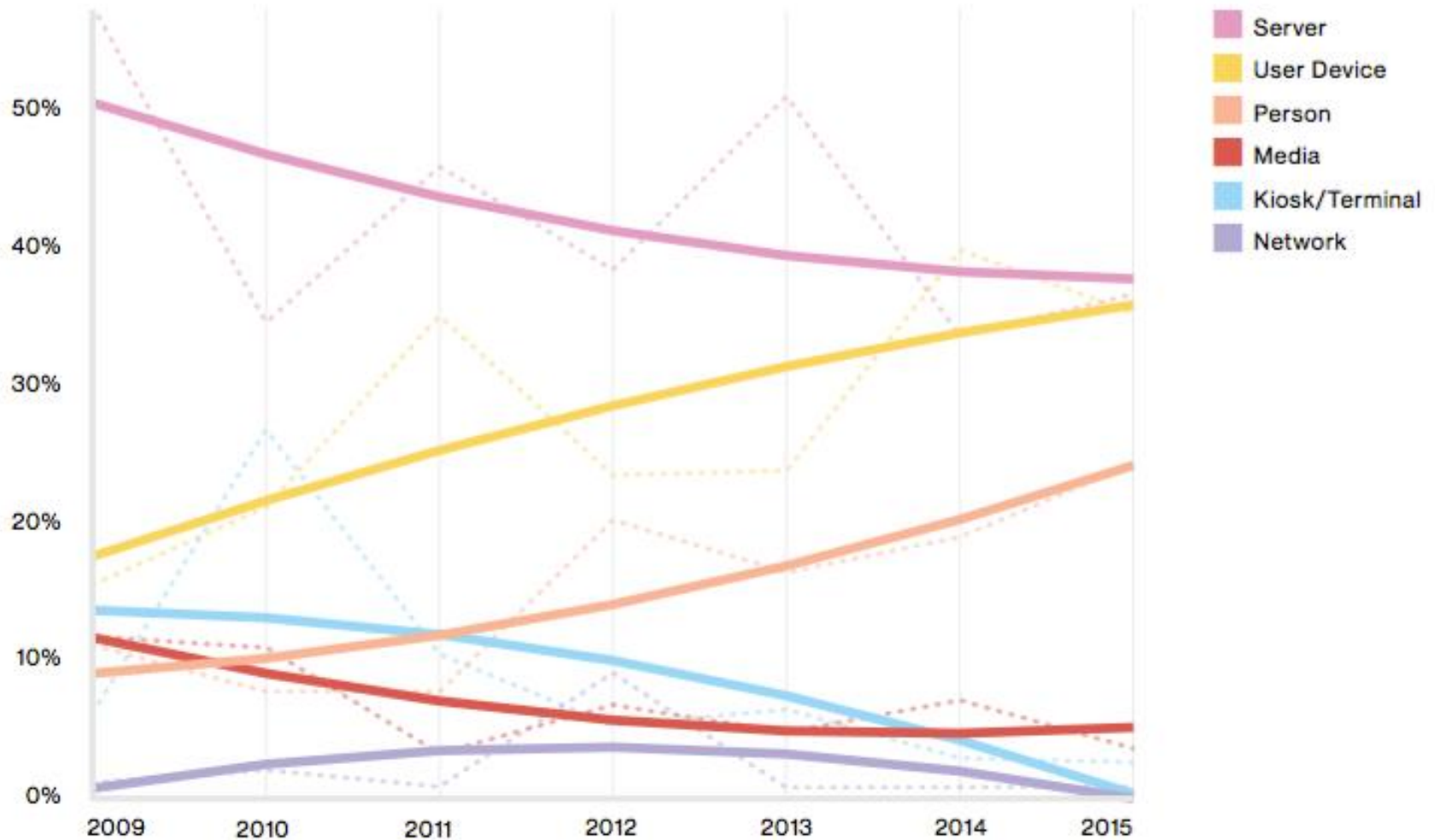


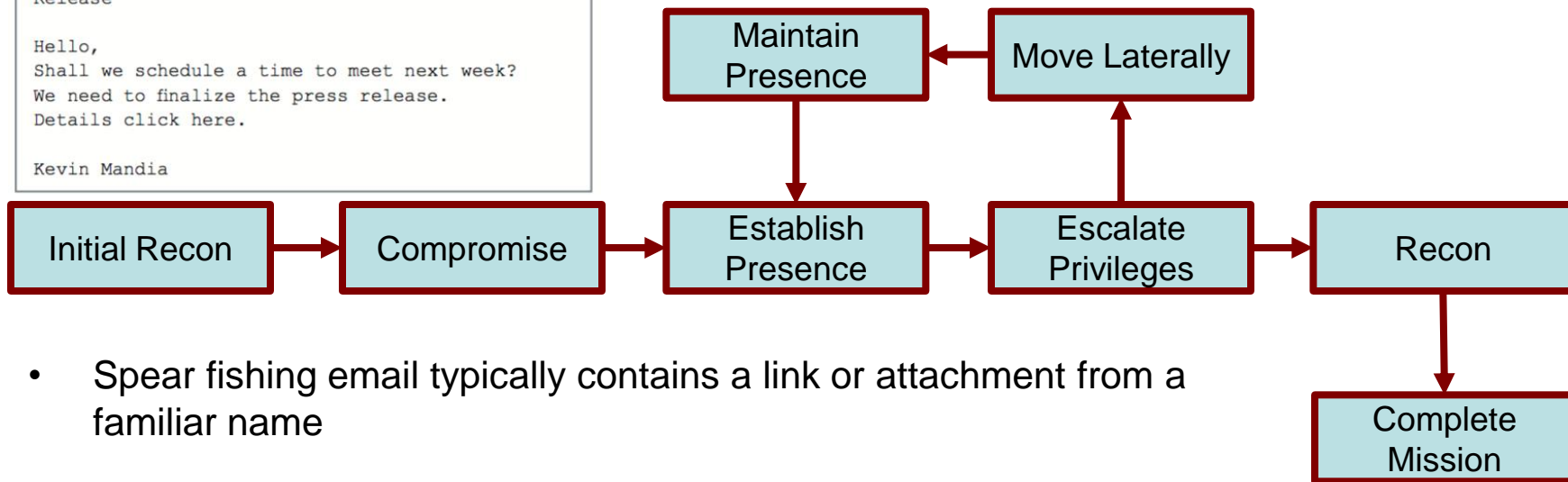
Figure 4: Compromises by Category

# Attack Methodology

```
Date: Wed, 18 Apr 2012 06:31:41 -0700
From: Kevin Mandia <kevin.mandia@rocketmail.com>
Subject: Internal Discussion on the Press
Release
```

```
Hello,
Shall we schedule a time to meet next week?
We need to finalize the press release.
Details click here.
```

Kevin Mandia



- Spear fishing email typically contains a link or attachment from a familiar name
- Victim will inadvertently open a backdoor for adversary
- Adversary will collect information about network and attempt to escalate privileges and move laterally from system to system
- Extremely difficult to detect an adversary that uses legitimate credentials
- When adversary finds target data, they will export the data to C2 servers

- A data set is comprised of features that represent the different characteristics of a sample
- Feature selection is a pre-processing step to classification in which important features are identified and the original feature set is reduced in order to improve classifier performance [Identify Relevance, Remove Redundancy]
- Constructed features can be created from the original feature set

## Typical Malware Features

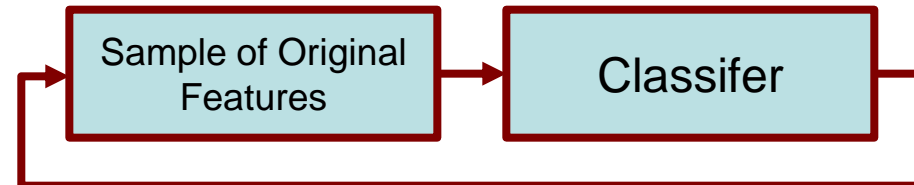
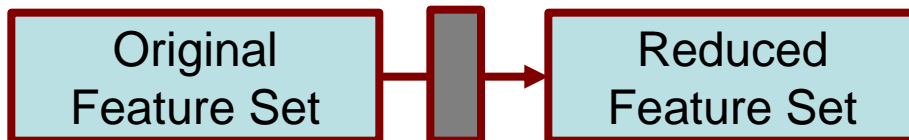
- N-grams, Metadata, Entropy, Opcode counts, Register Values, API calls

### Filter Method

### Wrapper Method

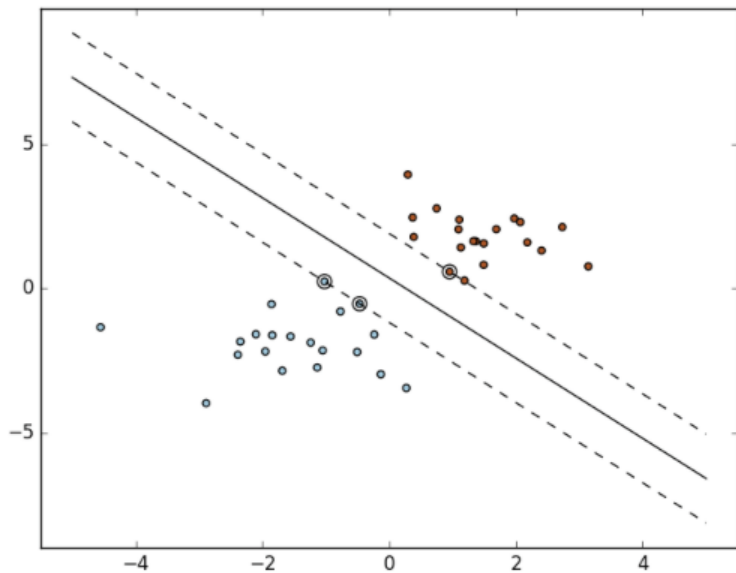
- Based on specific filter metrics

- Based on a feedback loop between classifier and feature selection algorithm



# Classification

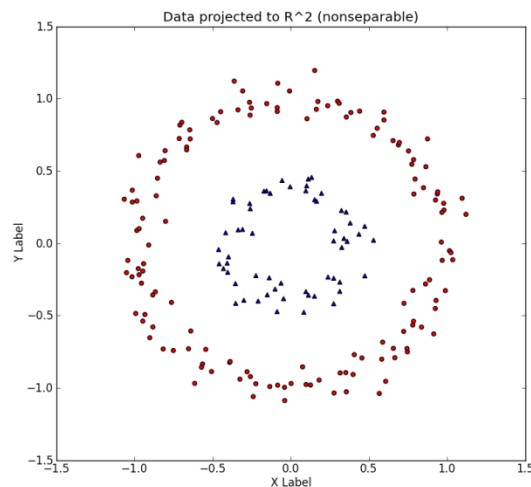
- Classifiers are trained with a training set of data
- Classifier accuracy is validated with cross-fold validation or a test data set
- The “Kernel Trick” can be applied to high dimensional data



**Figure 5: Support Vector Machine\***

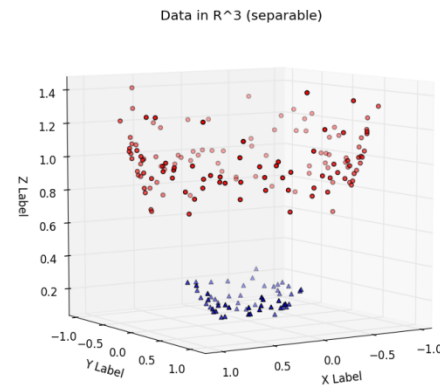
Malware Accuracy Measures:

- True Positive
- True Negative
- False Positive
- False Negative



**Figure 6: Kernel Trick Visualized\*\***

- Kernel = Mapping of inputs to feature space
- Multiple Kernel Learning approaches involve combining kernels



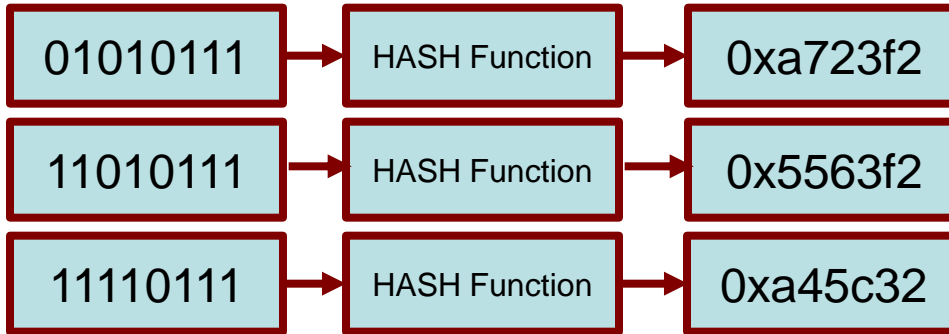
\*Source: <http://scikit-learn.org/stable/modules/svm.html>

\*\*Source: [http://www.eric-kim.net/eric-kim-net/posts/1/kernel\\_trick.html](http://www.eric-kim.net/eric-kim-net/posts/1/kernel_trick.html)



# Malware Detection Approaches: Signature/Anomaly

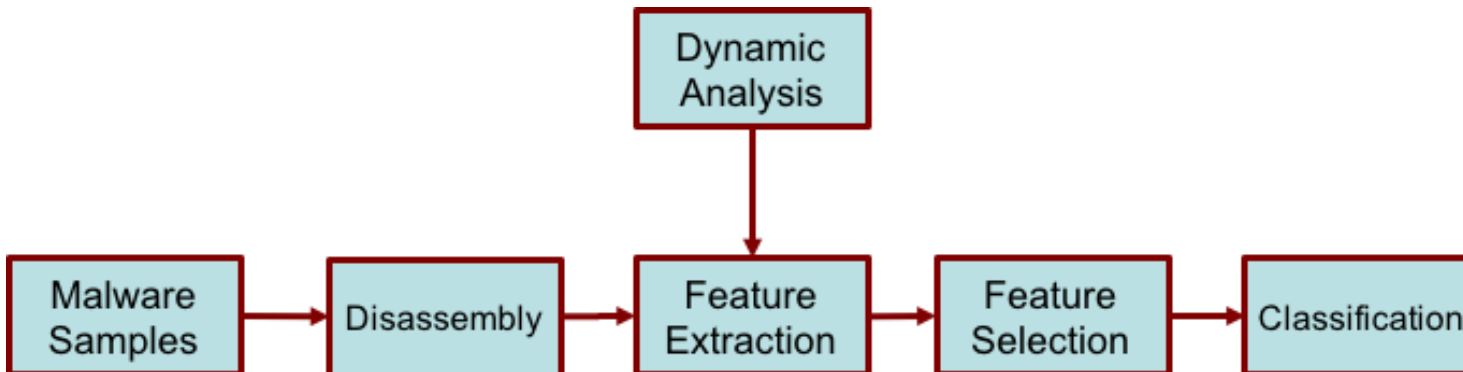
- **Signature based detection methods are well suited for known threats**



Code Obfuscation can easily overcome signature based methods!

- **Anomaly methods (Static/Dynamic Analysis) are required to prevent zero-day attacks**

## Standard Anomaly-based Malware Classification System



# Dynamic System Analysis Modeling of Malware\*

- Paper discusses an approach for the detection and characterization of malware based on the properties of kernel data structures
- Kernel object mapping system identifies dynamic kernel objects at runtime using virtualization software
- An object map of memory allocation events is created
- This approach takes into account “data hiding” which is a malware technique to hide kernel objects by removing memory references
- A profile/signature is created that describes the malware’s unique data access behavior

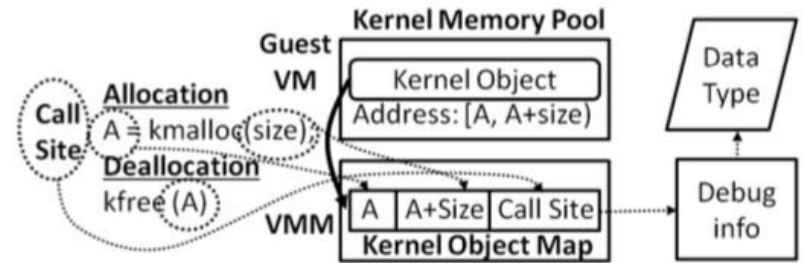


Figure 7: Memory Allocation Structure

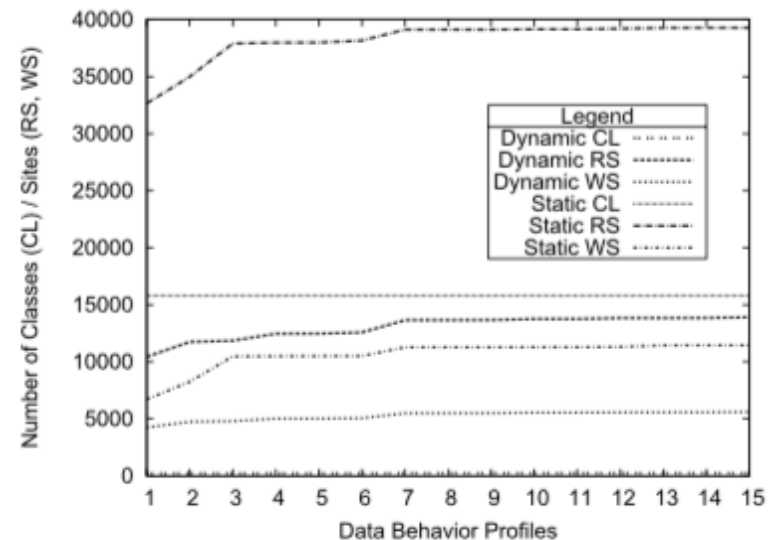
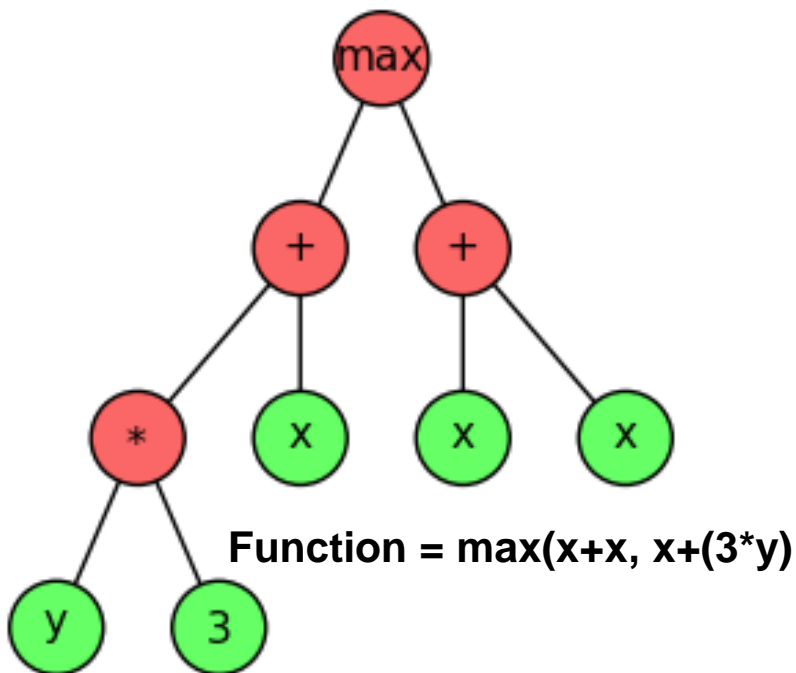


Figure 8: Unique Signature

\*Source: Rhee, Junghwan (01/2014). Data-Centric OS Kernel Malware Characterization. *IEEE transactions on information forensics and security.*, 9 (1), p. 72 - 87. (ISSN: 1556-6013)



**Figure 9: Example of Genetic Programming Function\***

- Genetic Programming (GP) models are evolutionary algorithms to produce the “most fit” individual
- For classification applications, fitness is typically representative of classifier accuracy
- Model Parameters: Population Size, Generations, Probability of crossover/mutation, max depth
- A population of individuals is initially created
- Individual fitness is calculated
- Individuals in each generation undergo crossover and mutation
- The “fittest individuals” survive

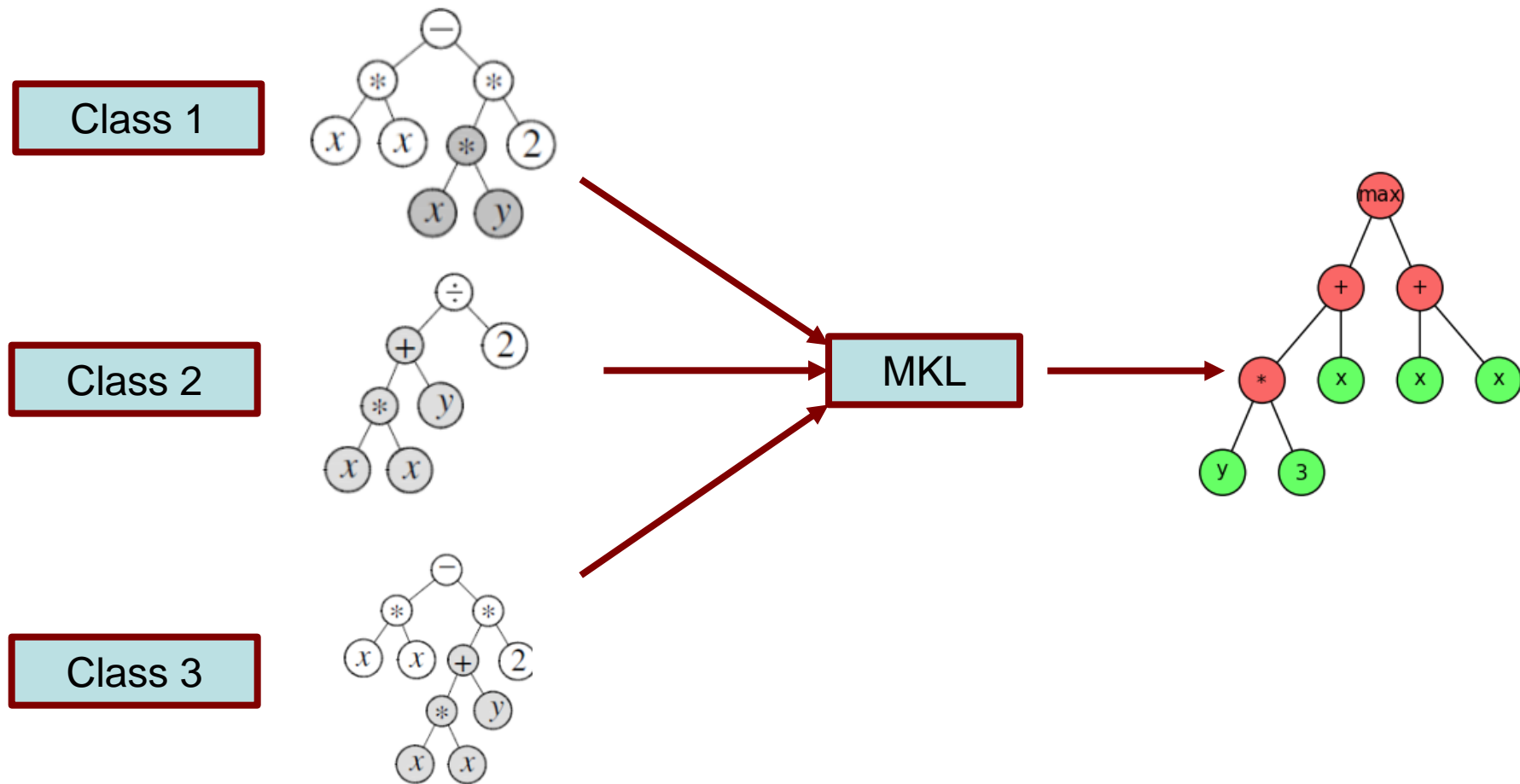
## Genetic Programming for Malware Classification

- Prior research has successfully applied GP to malware classification and feature selection
- My Ph.D. research is focused on improving existing GP methods for malware classification
  - Preliminary results are promising

\*Source: <http://deap.readthedocs.io/en/master/tutorials/advanced/gp.html>

# Genetic Programming for Multi-Class Classification

- X-sized populations of discriminant functions for each class are developed
- Second phase is based on exploring the best combination of individuals with an emphasis on reducing conflicting situations between classes



- Increasing threat space = More focus on feature reduction and feature importance to build accurate classifiers
- Feature construction appears to be a promising method to capture multiple data characteristics into a single element
- A combination of system behavior analysis and source code disassembly are needed to build complete models
- New methods needed to account for the human factor in system security
- Manual malware analysis is time consuming
- Future work: Models needed to account for increasing system complexity and interaction

Joe Mikhail

George Washington University, Ph.D. Candidate

[joemik@gwu.edu](mailto:joemik@gwu.edu)

703-855-4528