Outside the Closed World: On Using Machine Learning For Network Intrusion Detection

Robin Sommer, Vern Paxson

IEEE Symposium on Security and Privacy 2010 (S&P '10)

Agenda

- **Introduction**
- **Challenges of using machine learning**
- **Recommendations**
- **Conclusions**

Introduction

- Network intrusion detection systems (NIDS)
  - detects malicious activities such as DoS attacks, port scans, etc. by monitoring network traffic
- There are two types of NIDS:
  - misuse-detection
  - anomaly-detection
- In real world, misuse-detection type is used almost exclusively
- The paper aims to answer why this is the case, when machine learning is generally successful in many other applications

Challenges of Using Machine Learning (1)

- Main claim: "intrusion detection domain exhibits particular characteristics that make effective deployment fundamentally harder than other domains"
  - A. Outlier detection
  - B. High cost of errors
  - C. Semantic gap
  - D. Diversity of network traffic
  - E. Difficulty with evaluation

Challenges of Using Machine Learning (2)

A. Outlier Detection
   - Machine learning is better at finding similarities than finding an outlier
   - Outlier detection can be thought of as a classification problem. However, there is no training data for the "outlier" class
   - As a result, we need *perfect* model of normality

Challenges of Using Machine Learning (3)

B. High Cost of Errors
   - In intrusion detection, the cost of any misclassification is high

- Compared to other domains:
   - Product recommendation systems can tolerate errors
   - OCR technology: spelling and grammar checkers are commonly employed to clean up result. Users are expected to proofread, which is much easier than verifying an NIDS alert manually
   - Spam detection has a highly unbalanced cost model: false positives (ham declared as spam) are much more expensive than false negatives

Challenges of Using Machine Learning (4)

C. Semantic Gap
  - How to transfer the results into actionable reports for the network operator
  - The key question is, “What is the difference between *abnormal activity* and *attacks*?”
  - Need to incorporate local security policies

Challenges of Using Machine Learning (5)

D. Diversity of Network Traffic
  - Even basic characteristics such as bandwidth, duration of connections, and application mix can exhibit immense variability. Large bursts of activity are common
  - One solution is aggregation (volume per hour vs. volume per second)

Challenges of Using Machine Learning (6)

E. Difficulties with Evaluation

- Evaluation turns out to be more difficult than building the detector itself
- Difficulties of data
  - Lack of public datasets; the two publicly available datasets (DARPA/Lincoln Labs packet traces and KDD Cup dataset) are old and no longer suitable for evaluation
  - The reason: inspection of network traffic can reveal sensitive information about the organization
  - Two alternatives: simulation and anonymization
    - Simulations are not realistic enough
    - It is difficult to completely remove all sensitive information, and the artifacts removed by anonymization are often needed by NIDS
  - Researchers need to make their own datasets
    - Need access to large network

Challenges of Using Machine Learning (6)

E. Difficulties with Evaluation (continued)

- Semantic gap
  - The system needs to support the operator in understanding the activity to quickly assess the impact
- Adversarial setting
  - Attackers will try to evade NIDS
  - OCR users won't try to conceal characters
  - Customers have no incentive to mislead company's recommendation system

Recommendations (1)

- Understand the threat model
  - What kind of environment does the system target?
  - What do missed attacks cost?
  - What skills and resources will attackers have?
  - What concern does evasion pose?
- Keep the scope narrow
  - What specifically are the attacks to be detected?
  - Assess what tools are appropriate for the task
    - A common pitfall is *starting* with the premise to use machine learning , and then looking for a problem to solve
  - Identify the appropriate feature set

Recommendations (2)

- Reduce the costs (misclassifications)
  - Reduce the system's scope
  - Aggregating features over suitable time intervals
  - Post-process results with additional information
- Evaluation
  - One alternative is to bring the experiment to the data, i.e., researchers send their analysis programs to data providers who run them and return the output
  - Need multiple datasets, from multiple sources
  - Understand the results by manual examination of misclassifications
  - Also inspect the true positives and negatives

Recommendations (3)

- Gain insights to the problem space
  - Rather than treating machine learning as a classifier, we could examine how it uses the features to understand more about the difference between benign and malicious activity, which could be used as a basis for a *non-machine-learning* detector
    - In spam classification, by examining the learned Bayesian classifier, we discover that certain parts of the message (e.g. subject lines, Received headers, MIME tags) provide much more detection power

Conclusions

- The imbalance between deployments of machine learning-based NIDS stems from the specifics of the problem domain
  - outlier detection
  - very high cost of errors
  - semantic gap
  - high variability of benign traffic
  - adversarial setting