How Machine Learning is Changing the Game in Breach Prevention

Max Adams
Regional Sales Manager, West

Mike Myers
Systems Engineer
Our Heritage

Spun out of Northrop Grumman

Industry-Leading Machine Learning and AI Research

Arming Threat Analysts and Incident Response Teams
The network security paradox

Companies can never know about every possible threat to their enterprise infrastructures so they must change their thinking about how to defend their networks.
Advanced threats continue to elude even the best defenses

72% OF ENTERPRISES say it is impossible to get ahead of threats with traditional systems
By The Numbers: Not enough people or money to keep up
Humans are physically unable to analyze new threats at scale

Sources: Verizon DBR 2015; Symantec ISTR 20; Trend Micro Targeted Attacks 2015
**Network Security Timeline**

- **'87-93**
  - **Explosion of Anti Virus Companies**
    - McAfee ('87), Sophos ('88), Panda & Trend Micro ('90), Symantec ('91), Checkpoint ('93)
  - **Intrusion Detection Systems**
    - 1995 NetRanger IDS
    - 1997 ISS IDS
    - 1998 SNORT

- **'94**
  - **Application Layer Firewall**
    - Trusted Info Systems
  - **FireWall 1988**
    - Packet Filters developed by DEC; Stateful Packet Filter developed by Bell Labs in 1990

- **'96**
  - **First Paper for a ‘Sandbox’**
    - UC Berkeley

- **'98**
  - **IPS 2001**
    - Kaspersky 1997
    - Foundstone 1999
    - ArcSight 2000
    - Clam AV 2001
    - SourceFire 2001

- **'04**
  - **FireEye is Founded**
    - Kevin Mandia founds Mandiant

- **'05**
  - **Gartner Analysts Create the term ‘SIEM’**
    - NetWitness ‘06, Invincea ‘06, ArcSight IPO ‘07

- **'05**
  - **Next Gen Firewall 2012**
  - **CISCO Palo Alto Juniper**
Machine Learning is the growing trend in network defense

Thousands of academic papers published as early as 1995 on various ML techniques and processes related to malware... ramp up in popularity beginning in the late 2000s

- 3 presentations at BlackHat 2013
- 1 presentation at DEFCON 2013
- 18 presentations at RSA 2014/15

Commercial companies with ML-based solutions currently in the market

Commercial companies with published ML-based malware detection research but no ML-based malware detection product
Enterprises need response at digital speed

- Security Event Alert
  - Look Across Entirety of Network

- Event Triage
  - Automated

- Event Analysis & Annotation
  - Leverage Analytics to ID Relationships and Links in Data

- Case Creation
  - Develop a Threat Case File at Digital Speed

- Final Case Escalation

High Fidelity Detection using Machine Learning

Data Collection Centralized

minutes not days or months
Breach Timeline

- **T-1 sec**: Malware Delivery
- **T=0**: Breach
- **T+1 sec**: Initial C2 Secondary Payloads Heartbeat
- **T+1 hour**: Reconnaissance, Lateral Movement, and Staging
- **T+1 day**: Exfil
- **T+1 month**: Exfil
- **T>1 month**: Exit Network

**IoC Arrives**

**Anomaly Detection**

**Lucky**

**Unlucky**

Hunt Success!
Finding What the Perimeter Missed
Retrospective Analysis

- Indicators of Compromises (IoCs) are used to search the log repository
- IoCs may arrive in feeds days to months after threat actors are actively using them
Finding What the Perimeter Missed
Analytics/Anomaly Detection

- Establish a baseline of normal user and network behavior over time
- Once baselined, monitor for statistically significant changes in asset, user or network behavior
- Typically require multiple suspicious occurrences before alerting an analyst
- Require sophisticated analysts to understand how to interpret alerts or visually identify anomalies
Supervised vs. Unsupervised Learning

Supervised (label the items):

Only one right answer
- Model has been trained to make correct predictions
- We have test set with known labels
- Can quantify classifier performance

Unsupervised (sort the items):

Many reasonable answers
- More difficult to measure classifier performance
- Often requires training model on site to learn “normal”
Machine Learning Use Cases
**Different analytics, different use cases**

**Supervised Machine Learning**  
Goal is to **predict the right answer** based on exposure to training data.  
USE CASE: Zero Day/Polymorphic Threat Detection

**Unsupervised Machine Learning**  
Goal is to **find unusual patterns/behaviors** within unlabeled data sets by clustering like data and identifying outliers.  
USE CASE: Unusual communication between hosts

**Signature/Rules/Pattern Engine**  
Goal is to **match an explicit** string within a given data set.  
USE CASE: Known virus, IOC detection
Machine Learning in Cyber Security

Unsupervised (no labels)
- User behavior analytics
- Insider Threat Detection
- Malware Family Identification
- C2 Detection

Supervised (with labels)
- Network Anomaly Detection
- Network Traffic Profiling
- Spam Filtering
- Malware Detection

Incremental (Learn continuously)

Batch (Learn only once or in discrete steps)

The problem you are trying to solve dictates the data, features and machine learning algorithms used.
ML-Based Malware Detection Features: Static

Files Are A Bunch of Bytes

An n-gram, n=4

Static approach uses just the binary content and context (metadata) as features
Does not execute or open the software or file
Does not require disassembly but may use it

Pro: Fast, especially if no disassembly is used
Con: Difficult to associate features to function
Dynamic approaches use software or system activity as features
Requires a virtual execution environment (sandbox) or host-based agent to conduct feature extraction
Some activities are immediate others can take hours to appear, depend on user inputs, or vary with the runtime environment

**Pro:** Features are functions...easier to understand the classifier

**Con:** Slow...requires **sandbox spin up** and **fixed observation window**, malware authors can elude detection by using the same analysis methods used by the good guys
Network and User Behavior Anomaly Detection Features

Network flow and host log data as features
Does not work out-of-the-box (models are not trained prior to deployment).
Requires training time on-site.
Alerts are typically comprised of multiple anomalous events

**Pro:** Not file based, useful for both malware and insider threat detection

**Con:** Anomalous does not always mean malicious and there are a lot of statistically anomalous benign activities
Pipelining ML Solutions

• Use of ML in security is not mutually exclusive
• Provide defense in depth by staging solution with different ML technology
• Example:
  • Use static analysis as a first stage filter
  • Followed by dynamic execution engine
  • Followed by anomaly detection – do logs indicate anomalous behavior some time post installation?
Supervised Learning Workflow

1. Library of Known Malicious and Benign Files
2. Feature Extraction
3. Prioritized Features
4. Train Classifier
5. Test
6. Classifier
7. Unknown File(s)
8. Classify Unknown Software
9. Predicted Classification (Benign/Malicious)

Classification (Customer Site)
Developing Supervised Learning for Cybersecurity

- **Feature selection** is the “secret sauce” in supervised machine learning
  - Determines how well the algorithm can separate samples into classes
  - **Data collection** (training data) also an important element (enough data + right data)
- Algorithms tend to be standard
  - Particularly in cybersecurity, not a lot of development of specialized learning algorithms
  - Algorithms are well supported by ML libraries
  - Algorithms may be hybridized and tested against sets of training data – trial and error
- Most time is spent on **feature engineering** and **data collection**
- Other challenges are around operationalizing ML –
  - How do you get real-time data? Where do you place your sensor?
  - Do you learn continuously (online) or use a batch learning model?
Trading Off Different Types of Errors in ML

- Effectiveness of a ML model depends on your goals
  - Low false positive?
    - Make your analysts happy
  - Low false negative?
    - You’re trying to detect malware!
- Most models will take a balanced approach between false + and false – rate
- Some systems make this a tunable parameter
- With enough samples this is possible to measure statistically

Visualization 1: False Positives vs. False Negatives

In order to reduce the false negative rate, we may want to reduce the threshold to 0.1 or less
Learning Process

- The internet (Malware and benign software) → Collect Samples
- Repository Files Classifiers Results → Store Classifiers and results
- Select Files → Select and Extract Features
- File lists e694dcd302a36 59e0b3a8cb931 978ca43f2a832
- Classifiers → Build Classifiers
- Deploy Classifiers
- In-Situ
- Classify Samples → Hector Score: .82
Detecting the Threat Vector using Supervised Machine Learning

Trained

Trained beforehand so model is ready to work right out of the box.

Highly Accurate

Quantitatively measure model performance

Tailored

Customize your defense to your environment reducing false positives, improving detection, and giving you a unique model unavailable to the adversary.

Classifier Test Statistics

<table>
<thead>
<tr>
<th>Overall Performance Score</th>
<th>Existing Classifier</th>
<th>Candidate Classifier</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.969</td>
<td>0.992</td>
</tr>
</tbody>
</table>

Hector recommends accepting the candidate classifier.

Change in Threshold-Dependent Metrics

<table>
<thead>
<tr>
<th>Sample Sets</th>
<th>Change in False Positives</th>
<th>Change in False Negatives</th>
<th>Change in Performance Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Test Data</td>
<td>0.4%</td>
<td>0.1%</td>
<td>0.004</td>
</tr>
<tr>
<td>Local Traffic Holdout Data</td>
<td>0.7%</td>
<td>0.2%</td>
<td>0.050</td>
</tr>
<tr>
<td>Combined Data</td>
<td>0.5%</td>
<td>0.1%</td>
<td>0.023</td>
</tr>
</tbody>
</table>
Takeaways – how machine learning is changing the game …

<table>
<thead>
<tr>
<th>Detection of Unknown Threats</th>
<th>Improved Resource Efficiency</th>
<th>Tailored to Customer Environment</th>
<th>Earlier Detection Reduces Risk</th>
<th>Maximize Existing Security Investments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improved detection of zero-day, targeted, and polymorphous threats that evade detection by traditional solutions</td>
<td>Helps skilled threat analysts prioritize their workloads and focus on actionable threats</td>
<td>Machine learning can custom tune each implementation to the customers’ specific environment, which improves accuracy, and makes each instance unique</td>
<td>Proactive vs. reactive detection means that the risk can be contained and mitigated before it’s had the chance to embed and cause significant damage</td>
<td>Some machine learning based solutions can be integrated with existing layers such as threat intel feeds, SIEM, and post analysis tools to improve overall effectiveness and maximize ROI</td>
</tr>
</tbody>
</table>
THANK YOU

Max Adams
max.adams@bluvector.io

Mike Myers
mike.myers@bluvector.io

www.bluvector.io