Modern Security Analytics
Finding a Needle in the Hayblower

Martin Rehak, Principal Engineer

July 1, 2014, AIMS 2014, Brno
Stream Security Analytics

- What is malware and why do we need to fight it?
- Step-by-step walkthrough through a security incident
- Security domain considerations, evolutions and particularities
- Stream Analytics: understanding big data in flight
- Ensemble of anomaly detectors and collective classification
- False Positives Analysis
The Advanced Malware Attack Lifecycle

**PLAN**
Attacker determines possible entry points, formulates a plan of attack

**EXPLOIT / ATTACK**
Attacker exploits vulnerabilities and delivers its weapon

**INFECT / SPREAD**
Malware moves laterally through the internal network in search of additional resources and data

**STEAL / DISRUPT**
Attacker takes action on its objectives and exfiltrates data or disrupts systems
Attack Kill Chain: Post Breach

1. Command and Control
   - Switches
   - Routers
   - Firewall
   - IPS
   - N-AV
   - Web Sec
   - Email Sec

2. Reconnaissance

3. Propagation

4. Data Theft
Technology Escalation in Advanced Cyber Threats

- Improved Delivery Mechanism
- Targeted execution, limiting sandboxing, detection and analysis options
- Advanced Command & Control
- Obfuscation of content, behavior and mission
- Hiding between the random malware infections
(Trivial) Incident Walkthrough
High-Level View of the System
Cloud Web Security
The Security Perimeter in the Cloud

- Collective Security Intelligence
- The Distributed Perimeter
- Cloud Connected Network

- Telemetry Data
- Threat Research
- Advanced Analytics

3M+ Cloud Web Security Users
6 GB Web Traffic Examined, Protected Every Hour
75M Unique Hits Every Hour
10M Blocks Enforced Every Hour

Mobile
Router
Firewall

810
Cloud Web Security Users
810
6 GB
810
75M
810
10M
Cisco Cloud Delivered Security Capability

Cloud

- Web Filtering
- Web Reputation
- Malware Signature
- File Rep / Sandbox
- File Retrospection
- Threat Analytics
- Active Reporting

Roaming Users

Headquarters

Branch Office
Three Worlds of Security Analytics
Pattern Matching

- Finds known malware or other attacks by matching the content of communication or files against patterns of known attacks.

Threat Intelligence

- Analyses known malicious behavior to infer general characteristics of attacks and uses the model to discover new attacks by their associations with known attacks.

Anomaly Detection

- Analyses the normal behavior of the network to build a predictive model and detects any patterns deviating from the normal behavior as potentially malicious.
Pattern Matching
- Lowest False Alerts
- Very specific and verifiable convictions
- Proven and traditional industry standard...
  - ... that needs to be complemented by other techniques in order to cope with advanced attacks.

Threat Intelligence
- Medium false alerts
- Convictions are not direct, but still understandable by humans
- Allows the detection of malware by exploiting the operational and technical imperfections of the attackers...
  - ... not committed by the advanced organizations.

Anomaly Detection
- Higher false alerts
- Convictions are not specific and are difficult to explain
- Allows the detection of broadest scope of malware, including the advanced attacks...
  - ... provided that we can separate the malware from random statistical fluctuations
Pattern Matching

- Complexity: Proportional to the number of known and described unique samples/traits that need to be described

Threat Intelligence

- Complexity: Proportional to the number of known botnets, malicious infrastructures,

Anomaly Detection

- Complexity: proportional to the number of users and organizations protected
Anomaly Detection – A Machine Learning Problem

Anomaly detection requires us to build a predictive model of

1. global internet traffic
2. each customer’s network, and
3. each host or user in each customer’s network

while respecting strict privacy and economic constraints.
“Traditional” Big Data Workflow

1. Receive the data
2. Perform an ETL (Extract-Transform-Load) process
   - Check correctness, formatting, deduplication
   - Extract Features
   - Schedule for writing into DB/Storage
3. Store the transformed data in data store
4. Run the analytics process on the data to find value
5. Display the results
Cost Breakdown

1. Receive the data
2. Perform an ETL (Extract-Transform-Load) process
   - Check correctness, formatting, deduplication
   - Extract Features
   - Schedule for writing into DB/Storage
3. Store the transformed data in data store
4. Run the analytics process on the data
5. Display the results
Slow and Expensive

Low Sensitivity (Recall)

Context Size

Small Context

Affordable Effectiveness

Large Context

Slow and Expensive
Stream Analytics

1. Receive the data
2. Perform an ETL (Extract-Transform-Load) process
   - Check correctness, formatting, deduplication
   - Extract Features
   - Schedule for writing into DB/Storage
3. Run fast and effective analytics process on the data
4. Store the transformed and filtered data in data store
5. Run (second) analytics process
6. Display the results
Low Sensitivity (Recall)

Small Context

Affordable Effectiveness

Context Size

Large Context

Slow and Expensive
Inside View
Model Decomposition – Hierarchical and Distributed

- Anomaly Detectors
- Trust Models
- Activity Classifiers
- Long-Term Behaviour Models

- Normal Flows
- Trusted Flows
- Non-Security Anomalies
- Fluctuations

5 billions requests per day
± 1% is anomalous
± 0.5% is security-relevant
10 M events per day
1K-50K Incidents per day

NetFlow
Web
Identity
3rd Party Feeds
DNS
...

Anomaly Detection
Trust Models
Trust Score
Local Correlation
Correlation Across Multiple Networks
Global & Local Correlation
Investigative Output
High Precision Output

Near real time processing

© 2013-2014 Cisco and/or its affiliates. All rights reserved.
Cognitive Threat Analytics
Layered Detection Engine

Detection
Agent 1
Agent 2
Agent 3
Agent N

Filtering
Trust Modeling Layer

Classification / Layer 1
WPAD
Generated Domain
Data Exfiltration

Classification / Layer 2
Malware 1
Malware 2
Malware 2

Data
Incidents

Individual Detectors
Correlation & Memory
Unsupervised Learning
Supervised Learning

Anomalous
Malicious
Malware
Examples of AD output (HTTP, real and synthetic malware)

Typical anomaly detection algorithm does not quite work

Problems:
False negatives
False positives

AUC around 0.7–0.8 not an exception
More AD output samples
Anomaly Detection Approaches

Statistics and Empirical Distribution, Information Theory:

Principal Components Analysis:

Graph Theory:
Anomaly Detectors

Individual AD models can detect malicious behavior as an outlier

- Making them better would compromise their generality

Base-Rate Fallacy as a fundamental limitation

- Not enough attacks in the traffic

Precision and Recall are not sufficient for direct use

Can we make the Precision better?
<table>
<thead>
<tr>
<th>Unstructured</th>
<th>Structured</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Short-term artifacts</td>
<td>• Long-term behaviors</td>
</tr>
<tr>
<td>• Uniformly distributed over users</td>
<td>• Confined to subset of hosts/users</td>
</tr>
<tr>
<td>• Proportional to traffic volumes</td>
<td>• No direct relationship to background</td>
</tr>
<tr>
<td>• Frequently associated with novelty, rather than anomaly</td>
<td>• Structured and explainable</td>
</tr>
</tbody>
</table>
High-Level False Positive Breakdown

- Anomaly
  - Unstructured
    - Noise
  - Structured
    - Structured Anomaly
    - Dual Behaviour
    - Malicious Behaviour
Unstructured Anomalies - Noise

- Unstructured, short-term anomalies
- Evenly distributed over hosts with the distribution proportional to the background traffic
  - Triggered by widespread, uniformly distributed behaviors (such as web browsing or SW updates).
  - Small proportion of high-volume behavior will be anomalous.
- Same anomaly instance generated by one or small number of hosts only, but noise categories and types can spawn over many or almost all hosts.
  - One specific favicon anomaly will be there for one user, but every user will have a favicon anomaly with some probability
- Examples:
  - **Novelty**: Users visiting unusual discussion forums or unusual documents (how to repair a dishwasher X)
  - Random failures: Redirects, dead links, incorrect retries
  - **Fragments**: Fragments of legitimate browsing behavior (ads from less known sites invoked by js)
  - **Multimedia**: Flash games & other multimedia fragments

Structured Anomaly

- Understandable outlier behavior with internal structure of each instance
- Long-term behavior
- Specific to one host or a small subset
  - The type of the anomaly will be restricted to a small subset of hosts
  - Triggered by a specific application (software update/software usage), context (wpad) or by combination of both factors (Skype running in some context)
- Additional evidence allows its (ex-post) separation from malicious behavior
- Examples:
  - Software updates
  - Skype/P2P over HTTP(S)
  - Configuration scripts, remote log creation, regular calls of unusual APIs
(Slides removed)
Dual Anomalies

• Major characteristics are identical with Structural Anomalies
  • Understandable behavior with internal structure of each instance
  • Long-term behavior
  • Specific to one host or a small subset

• Events, where the maliciousness can only be decided on the level of intent, as the identical events would appear in malicious and legitimate scenarios alike.
  • Features necessary for separation are not accessible (judging the HTTPS by the endpoint) or the behaviors are inherently inseparable.
  • Direct effect is the same in case of attack or legitimate activity. Intent can be dramatically different.

• Example:
  • Large file download/upload
  • Long-lasting HTTPS connection to a single host outside the company network
Malicious Behavior

• Major characteristics are close to Structural Anomalies or Dual Behaviors
  • Understandable behavior with internal structure of each instance
  • Long-term behavior
  • Specific to one host or a small subset

• Our experience, correlated evidence or the behavior itself allow us to conclusively confirm the behavior as malicious upon examination.

• Prioritization within the malware category is not trivial!
(Slides removed)
Unstructured Noise

- Caused by inherent unpredictability & randomness of the modeled system, computational limitations imposed on anomaly detectors and predictive precision of the anomaly detection algorithms.
- Can be reduced, but never fully eliminated.

Structured Anomaly

- Caused by the limited capability of anomaly detectors that are unable to model whole classes of behavior. Capability can be either a modeling problem (such as wrong timescale), or a feature selection problem.
- Separable, but not separated...

Dual Behavior

- The accessible features do not allow separation between the legitimate and malicious behavior of this class without a qualitatively different model.
- Opens exciting new research areas:
  - Intent modeling
  - Strategic correlation
  - Plan recognition
  - ...
Assumptions for Ensemble Approach

**Model Diversity:** We assume that the methods and features used by the same model ensure that the fact that one model has singled out a legitimate object as a false positive is not correlated with other models doing so.

**Model Alignment:** We assume that malicious behavior detection is correlated between the models that we combine.
And should the assumptions hold...

Averaging the outputs of several AD’s should allow you to dramatically increase the precision of the system without the negative impact on recall

But the assumptions hold only partially...
Limitations

**Question:** Actually, as system designers, we ought to be able to design the detectors with enough diversity by careful detection of features and AD methods – right?

**Short answer:** Sort of. Assuming the attackers are nice/stupid enough, or the AD methods are almost flawless in terms of generality.

**Better answer:**

- **Diversity** assumptions are relatively easy to satisfy and verify, as they rely on the background traffic characteristics. They also tend to carry over between the contexts (networks) reasonably well.

- **General Alignment** is far more difficult to guarantee due to the diversity of malicious behaviors. In reality, we don’t average all the detectors, but select subsets of 4–10 detectors to optimize the recall.
• M Reháč, E Staab, V Fusenig, M Pěchouček, M Grill, J Stiborek, K Bartoš, Thomas Engel. Runtime monitoring and dynamic reconfiguration for intrusion detection systems, RAID 2008 - Recent Advances in Intrusion Detection
• J Stiborek, M Grill, M Rehak, K Bartos, J Jusko. Game Theoretical Adaptation Model for Intrusion Detection System, PAAMS 2012
(Slides removed)
User-Level Perception of the System

- **Precision alone** defines the user experience & operations
  - Government – more recall sensitive in top security areas
  - Multinational corporation – precision determines the real usability
  - Smaller enterprises and companies – even higher precision requirements

- Recall is (almost) always argued on the “compared to the state without the system X deployed” basis

- Improving recall is our professional duty, but the business rationale and user perception is less clear than in case of precision
  - But there is a catch…precision and recall are directly related through the **user bottleneck**