ARAKIS – An Early Warning and Attack Identification System

Piotr Kijewski
Piotr.Kijewski@cert.pl

16th Annual FIRST Conference
June 13-18, Budapest, Hungary

Presentation outline

- Trends in large scale malicious code attacks
- ARAKIS project goals
- Malicious code: target selection and propagation methods
- ARAKIS general architecture
- ARAKIS data sources
- Current stage of development
- Future work

Trends

- Self-propagating malicious code and automated attack increase
- Targeted automated attacks (examples: Bugbear.B, MyDoom)
- Shortening time window between vulnerability announcement, exploit release, worm appearance (example: Witty)
- Possible zero-day exploits
- Classic IDS/IPS systems base their knowledge on rules of known attacks – provide historic view
Trends: blended threats

ARAKIS Project Goals

- Attempt to address the identified trends
- Practical detection, analysis, response to large scale malicious code attacks and other automated attacks against Polish networks
- Unique view of attacks on Polish networks, possibly those part of the Polish national critical infrastructure

- Enable automated detection of novel threats
- Automate the process of attack analysis, allowing for the identification and description of novel attacks
- Develop a method of automated signature generation for use in firewalls, IDS/IPS
- Develop an automated process for the comparison of trends across multiple administrative domains
- Improve network situational awareness
- Serve as an aid in general incident handling
- Provide network attack statistics
A Taxonomy of Computer Worms, by Nicholas Weaver, Vern Paxson, Stuart Staniford, Robert Cunningham

Target selection:
- Scanning
- Pre-generated target lists
- Externally generated target lists
- Internal target lists
- Passive

Propagation method:
- Self-carried
- Second channel
- Embedded

Different target selection and propagation methods require different algorithms and different data sources

Early warning component:
- Detection of a new widespread threat (anomaly detection)
- Issuing an alert
- Provide supporting data to other components

Attack analysis component:
- Collecting attack data and preparing for analysis
- Identifying if the threat and attack is novel
- Describing the full attack scenario

Architecture

Different target selection and propagation methods require different algorithms and different data sources

Early warning component:
- Detection of a new widespread threat (anomaly detection)
- Issuing an alert
- Provide supporting data to other components

Attack analysis component:
- Collecting attack data and preparing for analysis
- Identifying if the threat and attack is novel
- Describing the full attack scenario
Architecture

- Detector generation:
  - Based on data from other components, identifies the most relevant attack characteristics
  - Generates attack signature

- Knowledge base:
  - Labeled data about identified threats, attacks, their characteristics, proposed signatures
  - Used to support other modules (identifying novel attacks)

Data sources

<table>
<thead>
<tr>
<th>Firewalls</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• see &quot;bad&quot; packets</td>
<td>• &quot;bad&quot; packet definition incomplete</td>
</tr>
<tr>
<td></td>
<td>• good anomaly detectors</td>
<td>• poorer at detecting non-scanning/probing attacks</td>
</tr>
<tr>
<td></td>
<td>• observe production networks</td>
<td>• do not supply much packet information</td>
</tr>
<tr>
<td></td>
<td>• present in some form on many connections to the Internet</td>
<td>• cannot describe attacks</td>
</tr>
<tr>
<td></td>
<td>• allow for monitoring of a large disjoint address space</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Honeypots</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• see mostly malicious traffic, less false positives</td>
<td>• not production networks</td>
</tr>
<tr>
<td></td>
<td>• simplified detection process</td>
<td>• require additional address space</td>
</tr>
<tr>
<td></td>
<td>• can be used to observe full attack process</td>
<td>• may be recognized as fake</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• may attract attacks of no special interest</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• risk</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Netflow</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• WAN viewpoint</td>
<td>• huge amount of data</td>
</tr>
<tr>
<td></td>
<td>• large scale attack detection (scans, DDoS)</td>
<td>• not much information on packet headers, none about payloads</td>
</tr>
<tr>
<td></td>
<td>• can be used to observe full attack process on real nets</td>
<td></td>
</tr>
</tbody>
</table>
Current stage of development

- Firewalls as early warning data source
- Monitoring over 2000 dst IPs
- Trend analysis on port activity
- Tables, graphs, maps published on web page
- System for mass IP to country, IP to AS, IP to PL states/cities lookups
- Honeyd deployed across 4 /26 networks, with 2 different /8 prefixes, across 2 ASes
- Libpcap based sniffer collects and organizes data for future on-line and off-line analysis
- C/Perl/PHP

Future work

- Add netflow as early warning data source
- Develop and implement algorithms for threat and attack analysis
- Explore data mining and machine learning techniques
- Automate generation of snort signatures
- Knowledge base organization
- Formats for threat description
- Communication framework

Questions?

Piotr Kijewski
Piotr.Kijewski@cert.pl