Memory Heat Map: Anomaly Detection in Real-Time Embedded Systems Using Memory Behavior

Man-Ki Yoon, Sibin Mohan, Jaesik Choi, Lui Sha

University of Illinois at Urbana-Champaign
Information Trust Institute
Ulsan National Institute of Science and Technology

DAC 2015
Anomaly Detection Using Memory Behavior

• Kernel Memory Behavior
  – Good indicator of system-wide behavior
    • Every application has to use kernel services
  – Simple hardware design
    • Known location, no need for address translation and paging
  – Monitoring kernel instructions

• Key Approaches
  – Memory Heat Map (MHM)
  – Image recognition technique
  – Monitoring HW module (Memometer)
Overall Process

Real-Time Applications

Time
Overall Process

Monitoring interval

Real-Time Applications

Memory Heat Map of Kernel .text
Overall Process

Real-Time Applications

Memory Heat Map of Kernel .text

Monitoring interval

Linux Kernel .text Segment (0xC0008000, 0xC02E7AA4)

Memory Region Size: 3,013,284 Bytes
Granularity: 2,048 Bytes
# cells: 1,472
Overall Process

Monitoring interval

Real-Time Applications

Memory Heat Map of Kernel .text

Reduced MHM

Classification

Normal/Abnormal

Normal/Abnormal

Normal/Abnormal

Normal/Abnormal
Memometer

Instrumenting Security: Monitor Memory Accesses in Memory Monitors

- Monitored Core
- Secure Core
- Address Filtering and Target Cell Calculation
- Double Buffering for Uninterrupted Monitoring

Controller

MHM Memory 0

MHM Memory 1

Address Filtering and Target Cell Calculation

Double Buffering for Uninterrupted Monitoring

Configuration by SecureCore
Learning Memory Heat Maps

- **Goal**
  - Find patterns from normal MHMs
  - Given a new observation, analyze the statistical similarity to the patterns

- **Idea**
  - MHM = Image
  - Normal memory behavior can be grouped into a finite number of similar image groups
  - Use an image recognition technique and clustering

→ Is this normal or not?
MHM Learning and Anomaly Detection

**Dimensionality reduction using Eigenface**
- Image recognition technique
- Based on PCA (Principal Component Analysis)

**Pattern learning using clustering**
- E.g. Gaussian Mixture Model
- Identify representative MHM patterns
Anomaly Detection on Prototype

Prototype implementation
- ARM Cortex-A9 on Simics
- Linux 3.4
- 10ms interval, 2KB cell size
- Embedded benchmarks

Tested anomalies
- Unknown application launch
- Application kill
- Shellcode execution
- Kernel rootkit

Example
Analysis Latency

<table>
<thead>
<tr>
<th>Given</th>
<th>Dimensionality reduction</th>
<th>Similarity calculation</th>
<th>Decision</th>
</tr>
</thead>
<tbody>
<tr>
<td>MHM Size (# cells)</td>
<td>1472</td>
<td>368</td>
<td>1472</td>
</tr>
<tr>
<td># Eigenmemories</td>
<td>9</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td># Clusters</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Avg. time</td>
<td>358 us</td>
<td>100 us</td>
<td>216 us</td>
</tr>
</tbody>
</table>

Normal? Abnormal?
Summary

• Anomaly detection in real-time embedded systems
  – Kernel memory behavior monitoring
    – **Memory Heat Map (MHM)**
      • Dimensionality reduction and clustering
  – Memometer
    • Uninterrupted MHM monitoring on multicore processor

• Future work
  – RTOS
  – Deep Learning