KIDS – Keyed Intrusion Detection System

SAŠA MRDOVIĆ, BRANISLAVA DRAŽENOVIC
UNIVERSITY OF SARAJEVO
FACULTY OF ELECTRICAL ENGINEERING
BOSNIA AND HERZEGOVINA
Introduction

Related work

Proposed detection method
  - Key introduction

Testing

Conclusion
Introduction

- Intrusion detection systems
  - Evolution and improvement 😊
- => Attack improvement
  - => undetected 😞
- Anomaly based NIDS
  - Detection method known
    - packet elements used to build model of normal
  - Mimicry attack
    - mimics normal packets – in used elements
Related Work

- **PAYL [Wang04,05]**
  - Model - single payload bytes frequencies

- **Anagram [Wang06]**
  - Model - fixed length payload byte sequences (n-grams)
  - Simple (fast) anomaly score calculation
    - new n-gram/all n-gram in packet payload

- **Language model ...[Rieck07]**
  - Payload divided into words
    - byte sequence between delimiters
  - Comparable accuracy to n-grams
  - Smaller computational load
Proposed Detection Method

- Words based
- Word transitions, also
- Resistant to some attacks in training data
- Prevents mimicry
  - Introduce key
    - Kerckhoffs’ principle [Kerckhoffs, 1883]
    - Shannon’s maxim [Shannon, 1949]
    - Open design principle [Saltzer and Schroeder, 1975]
Set of Delimiters - Key

- Set of “normal” words depends on selection of delimiters
- Selected set of delimiters determines model of normal packet payload
- The same model creation method and different delimiters set => different model
- Set of delimiters – Key
  - Method – public
  - Set of delimiters - secret
Learning

- Normal, attack free, payloads - partitioned into words.
- Model of normal packet:
  - Word frequency distribution
  - Word transition frequency distribution
- Training phase
  - Appearance of any word is counted and stored
  - Appearance of any pair of words is counted and stored.
Detection

- **Word based score**
  - \( k \) – number of words in payload
  - \( n(w_i) \) - number of appearances of the word \( w_i \) in learned model
  - Tolerant to some attacks in training data
  
  \[
  S_w = \frac{1}{k} \sum_{i=1}^{k} \frac{1}{n(w_i)}
  \]

- **Transition based score**
  - \( m \) – number of word transitions in a payload
  - \( n(t_i) \) - number of times transition \( t_i \) occurred during training

  \[
  S_t = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n(t_i)}
  \]

- **Total score**
  \[
  S = S_w \times S_t
  \]
Testing

- Used HTTP traffic and attacks
- Real university department traffic
  - Cleaned using Snort and manual inspection
- Metasploit for attacks
Initial Set of Delimiters

- From [Rieck07]
  
  CR LF TAB SPACE , . : / \ & ? = ( ) [ ] " ; < >

- Number of learned words
  - Levels after 96 hours of traffic
  - Around 33000

- Number of transitions
  - 33000 x 33000 matrix
  - Too much
  - Some words are very rare
  - Use only words that appear more than 10 times
  - 80 times smaller matrix
## Test attacks (part)

<table>
<thead>
<tr>
<th>No.</th>
<th>Vulnerability / payload</th>
<th>CVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Apache Chunked-Encoding / meterpreter-reverse_tcp</td>
<td>2002-0393</td>
</tr>
<tr>
<td>2</td>
<td>Apache Chunked-Encoding / shell-reverse_http</td>
<td>2002-0394</td>
</tr>
<tr>
<td>3</td>
<td>Apache mod_jk overflow / adduser</td>
<td>2007-0775</td>
</tr>
<tr>
<td>4</td>
<td>Apache mod_rewrite / shell-bind_tcp</td>
<td>2006-3748</td>
</tr>
<tr>
<td>5</td>
<td>Apache mod_rewrite / vncinjext-reverse_tcp</td>
<td>2006-3749</td>
</tr>
<tr>
<td>6</td>
<td>IIS 5.0 IDQ Path Overflow / shell-reverse_http</td>
<td>2001-0501</td>
</tr>
<tr>
<td>7</td>
<td>IIS 5.0 IDQ Path Overflow / shell-reverse_tcp</td>
<td>2001-0502</td>
</tr>
<tr>
<td>8</td>
<td>IIS ISAPI w3who.dll / exec</td>
<td>2004-1135</td>
</tr>
<tr>
<td>9</td>
<td>IIS ISAPI w3who.dll / shell-reverse_tcp</td>
<td>2004-1136</td>
</tr>
<tr>
<td>10</td>
<td>Oracle 9i XDB HTTP PASS / shell-reverse_tcp</td>
<td>2003-0728</td>
</tr>
<tr>
<td>11</td>
<td>Xitami If_Mod_Since / shell-reverse_tcp</td>
<td>2007-5068</td>
</tr>
</tbody>
</table>

Attacks with related vulnerability and used payload
ROC curve

![ROC curve graph]

The graph shows the relationship between the true positive rate and the false positive rate. The x-axis represents the false positive rate, ranging from 0 to 0.005, while the y-axis represents the true positive rate, ranging from 0 to 1. The data points form a curve that indicates the performance of a binary classification system.
Arbitrary Set of Delimiters

- Different sets of delimiters
- Different number of delimiters in set
  - 15, 20, 25, 30
- 30 different sets of each size
- Total of 120
- Random choice of delimiters
  - Function “rand” to generate number 0 - 255
Results

- **Number of learned words**
  - Levels after 96 hours of traffic (again)
  - 40000 – 50000 (20 to 50% increase)

- **Number of transitions**
  - Again, some words are very rare
  - Use only words that appear more than 10 times
  - Matrix of manageable size
ROC curves
Conclusion

- Implementation of open design principle
- Now HTTP – others should work too
  - Protocol independent
- Key selection should be further tested
- Keyed IDS is the main idea
  - There might be better implementations
Questions

sasa.mrdovic@etf.unsa.ba