Overview

- Introduction
- Monitoring System Security by New Mexico Tech
- Spector Pro for Windows by SpectorSoft
- Conclusion: How Neural methods can be used for analyzing results of the recorded data

Introduction

- Information system security is a growing concern since computer systems worldwide are becoming increasingly vulnerable due to the rapid increase in connectivity and accessibility that has resulted in more-frequent intrusions, misuses, and attacks.
- In order to determine system security risk, system monitoring should be used to screen users activities (log and usage files).
- The neural networks can be trained with minimized, abstracted user logs for realtime detection of attacks and misuses.

Monitoring System Security

- This system was designed at New Mexico Institute of Technology
- Uses neural networks for misuse and anomaly detection of information systems that include both Unix/Linux processors and web servers.
- Provides real-time monitoring (the neural network based tool runs in real-time or near real-time).
- Cost-effective approach (highly effective in detecting true attacks and minimizing false alarms after proper "tuning").

Anomaly Detection vs. Misuse Detection

- Anomaly detection looks for differences in normal usage behavior of everything from standard running programs, to user behavior.
  - Highly difficult to customize system to system, or even user to user.
  - Discovers new problems (attack methods).
  - Irregular behavior may be counted as attack.
- Misuse detection looks primarily for recognized patterns of attack.
  - Simpler to process and locate.
  - Tends to fail when new attack method are discovered and implemented.

Sources

- http://www.computer.org/students/looking/2002fall/2.pdf
- http://www.spectorsoft.com/
Serial vs. Association Patterns

- In a serial pattern time comes into play, as the features it looks for must be done in a somewhat sequential order.
  - For example: if w and if then x, and if then y, then z. (IF w then x then y THEN z) is not equal to (IF y then x then w THEN z).
- In an Association pattern it is no particular order, but simply is a collection of incidents.
  - If w and x and y, then z is the same as if w and y and x, then z.

Information Structure

- For the purpose of training neural networks, the data must be constructed in pairs of input/output.
- The input data is taken from several system log files for each user to be processed by a set of rules to form the input for a back-propagation neural network.
- The output is a single value between 1 and 5 that indicates the likelihood that an intrusion has occurred.

Input Information (1)

Linux/Unix command (c1-c3) and operating system (c4)

<table>
<thead>
<tr>
<th>c1</th>
<th>Weight</th>
<th>Each command in the system is assigned by a weight ranging 1-10, where 10 is the highest suspicious command. (n<strong>n</strong> is higher than n filename)</th>
</tr>
</thead>
<tbody>
<tr>
<td>c2</td>
<td>Average Weight</td>
<td>The average weight of the commands (shows the level of the users intensity to committing an intrusion)</td>
</tr>
<tr>
<td>c3</td>
<td>Highest Weight</td>
<td>The weight of the most dangerous command the user has done (used to check if user has violated their access privileges)</td>
</tr>
<tr>
<td>c4</td>
<td>Authentication Failures</td>
<td>Measures login permission information</td>
</tr>
</tbody>
</table>

Input Information (2)

HTTP (h1-h3)

<table>
<thead>
<tr>
<th>h1</th>
<th>Page Accesses</th>
<th>The number of times a user (IP address) has accessed the system. A measure of intensity.</th>
</tr>
</thead>
<tbody>
<tr>
<td>h2</td>
<td>Page Failures</td>
<td>The number of times a page was not found, access in a restricted location was attempted, etc.</td>
</tr>
<tr>
<td>h3</td>
<td>Activities Average Weight</td>
<td>The averaged weight of the activity of the user, and meant to be a preliminary indication of the user's intentions to their current actions. Can also be used to differentiate between a user who accidentally does something that can be considered an attack.</td>
</tr>
</tbody>
</table>

Input Information (3)

Fingerprints (or Attack Scenario Patterns)

<table>
<thead>
<tr>
<th>f1</th>
<th>Number of patterns located</th>
<th>A simple counter value that indicates how many rules were triggered.</th>
<th>A large amount of time by the same user, a pattern has been found.</th>
</tr>
</thead>
<tbody>
<tr>
<td>f2</td>
<td>Average fingerprint value</td>
<td>An averaged weight of the fingerprints that the rules have located. A measure of intensity.</td>
<td></td>
</tr>
<tr>
<td>f3</td>
<td>Highest fingerprint value</td>
<td>The highest value within the rule patterns. A measure of attack likelihood.</td>
<td></td>
</tr>
</tbody>
</table>
Output Information

Suspiciousness

<table>
<thead>
<tr>
<th>s1</th>
<th>Suspiciousness value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A single value that assigns the user to one of the suspicious levels:</td>
</tr>
<tr>
<td></td>
<td>LOW – typical user;</td>
</tr>
<tr>
<td></td>
<td>MEDIUM – review of user activity is needed;</td>
</tr>
<tr>
<td></td>
<td>HIGH – user activity is suspicious and must be reviewed.</td>
</tr>
<tr>
<td></td>
<td>ALERT1 – immediate system alert</td>
</tr>
<tr>
<td></td>
<td>ALERT2 – problematic user such as novice user</td>
</tr>
</tbody>
</table>

NOTE: The single output of the neural network is the Suspiciousness Score.

Spector Pro

- Spector Pro is one of the existed applications for System Monitoring for Windows.
- Spector Pro is used for recording every detail of PC and Internet activity for home, office or school.

Spector Pro Features

- Email recording
- Chat and Instant Message recording
- Web Site recording
- Keystroke recording
- VCR-like snapshot recording
- Keyword Detection

All tools record simultaneously, secretly saving the data to a hidden location on the PC.

Additional Features

- Scheduled Recording
- Record by User ID
- Keep Recorded Information Hidden
- Stealth Mode
- Hot Key Access
- Password Protection
- Using Spector in a Network

Conclusion

- Collected information can be used in neural network model to automate settings for users
- The network has to be trained with enough examples of the monitoring process
- Results can be used to modify user’s restriction policy (or to create user’s restrictions dynamically based on his/her activities)
- Further step: create a set of rules for each security level