PREDICTING NETWORK ATTACK PATTERNS IN SDN USING MACHINE LEARNING APPROACH

SAURAV NANDA,
FAHEEM ZAFARI, CASIMER DECUSATIS,
ERIC WEDAA AND BAIJIAN YANG
Agenda

- Motivation
- Introduction
- Problem Statement
- Methodology
- Experimentation
- Results
- References
Motivation

• LongTail Project
  • Analyzes SSH brute force attacks
  • Statistically quantifies:
    - IP addresses
    - Accounts
    - Passwords
    - Account/password pairs
    - Analyzing attack patterns

http://longtail.it.marist.edu
Data on this page last updated on Tue Nov 8 00:27:05 EST 2016

Number of ssh login attempts today 29503
Number of usernames seen today 61
Number of unique usernames seen today 61
Number of passwords seen today 8121
Number of unique passwords seen today 6
Number of IP addresses seen today 43
Number of unique IP addresses seen today 7
## Sample Data from LongTail

<table>
<thead>
<tr>
<th>IP</th>
<th>Lifetime In Days</th>
<th>Botnet</th>
<th>First Date Seen</th>
<th>Last Date Seen</th>
<th>## of Attack Patterns Recorded</th>
</tr>
</thead>
<tbody>
<tr>
<td>49.236.204.180</td>
<td>640.23</td>
<td>pink_roses</td>
<td>2015/01/12 13:26:34</td>
<td>2016/10/13 19:53:54</td>
<td>272</td>
</tr>
<tr>
<td>122.160.154.221</td>
<td>639.52</td>
<td>big_botnet</td>
<td>2015/02/02 16:46:25</td>
<td>2016/11/03 06:20:11</td>
<td>11</td>
</tr>
<tr>
<td>59.51.24.186</td>
<td>620.73</td>
<td>pink_roses</td>
<td>2015/02/06 16:47:43</td>
<td>2016/10/19 11:26:04</td>
<td>124</td>
</tr>
<tr>
<td>222.186.56.42</td>
<td>618.37</td>
<td>15-07-01-botnet-20</td>
<td>2015/02/07 07:45:01</td>
<td>2016/10/17 17:37:26</td>
<td>12</td>
</tr>
</tbody>
</table>

[http://longtail.it.marist.edu/honey/current_attackers_lifespan.shtml](http://longtail.it.marist.edu/honey/current_attackers_lifespan.shtml)
Log Analysis of IP Attacks

100.38.47.218
1 lines, dict-e53664bda267cedce5900c80d1902ef2.txt To: edub Attack #: 1 started on 2016/03/24 13:49:33
1 lines, dict-3d520cba13d3e60f92b8d6874428e82f.txt To: edu_c Attack #: 2 started on 2016/04/07 08:14:54

100.38.74.99
1 lines, dict-02a719d9d242acd4fcd8cc6da9f6cfbd.txt To: shepherd Attack #: 1 started on 2016/07/18 12:31:34

101.0.44.181
1 lines, dict-d64b8ef614272f5c703f4ae0cf1c51d7.txt To: syrtest Attack #: 1 started on 2015/08/25 00:24:02

101.0.44.231
1 lines, dict-ad6234f04947b500af48eba5d7f4a6fd.txt To: kippo2Jul Attack #: 1 started on 2015/07/24 11:44:45

101.0.44.236
1 lines, dict-864992c0102f84225736a7291a3791eb.txt To: edu_c Attack #: 1 started on 2015/07/31 16:08:51

http://longtail.it.marist.edu/honey/ip_attacks.shtml#58.206.126.29
Problem Statement

- Abdou et al. recorded \( \sim 17M \) login attempts from 112 different countries and over 6K distinct IP addresses.

- Our Objective
  - To use machine learning algorithms on historical network attack data set to predict the host which will be attacked.
  - To block particular subnet as a whole rather than blocking individual IP addresses.
Methodology

Machine Learning algorithms

- **C4.5**
  - To generate a decision tree based on information entropy.

- **BayesNet**
  - Probabilistic graphical model using a directed acyclic graph (DAG) showing a set of random variables and their conditional dependencies.

- **Naive-Bayes**
  - Probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions between the features.

- **Decision Table**
  - Compact way to model complex rule sets and respective actions using methods such as: flowcharts, switch-case and if-then-else.
Architecture

Historical Attack Data

Machine Learning Algorithms

Obtained Trained Model

Real Time Network Data

Classification / Prediction

Set Rules

SDN Controller
Dataset

- We are using a public dataset from “LongTail”
  - Open source project that records SSH brute force attacks 32 honeypots.
  - Dataset 1 - List of attacks including china = 278,598
  - Dataset 2 - List of attacks without china = 187,488
  - Dataset 3 – List of attacks with only from china = 91,110

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Size</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>278,598 (With Chinese attack data)</td>
<td>&lt;attacker IP&gt; &lt;attacked host&gt; &lt;number of attempts in an attack&gt; &lt;timestamp&gt;</td>
</tr>
<tr>
<td>2</td>
<td>187,488 (Without Chinese attack data)</td>
<td>&lt;attacker IP&gt; &lt;attacked host&gt; &lt;number of attempts in an attack&gt; &lt;timestamp&gt;</td>
</tr>
<tr>
<td>2</td>
<td>91,110 (Only Chinese attack data)</td>
<td>&lt;attacker IP&gt; &lt;attacked host&gt; &lt;number of attempts in an attack&gt; &lt;timestamp&gt;</td>
</tr>
</tbody>
</table>
Experiment Results

- **Results and discussion.**
  - Weka – Java based ML tool.
  - The datasets were split in 30/70, 40/60, 50/50, 60/40, and 70/30 ratio for training and testing purposes.
  - Prediction accuracy of different ML algorithms, for different data sets, training/testing split ratio and the threshold $\alpha$.
  - $\alpha$ is the minimum probability required to consider any host as vulnerable.
  - An average prediction accuracy of 91.68% was achieved with Bayesian Network (254,834 out of 278,598 attacks).
  - Highest accuracy of 99.99% is obtained with Decision Table for dataset 1 when alpha = 0.
Average Prediction Accuracy with Alpha

\[
\text{Accuracy} = \frac{\text{Number of correctly predicted attacks}}{\text{Total number of attacks}} \times 100
\]

<table>
<thead>
<tr>
<th>$\alpha$ (%)</th>
<th>Avg. Prediction Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>97.06</td>
</tr>
<tr>
<td>1</td>
<td>95.78</td>
</tr>
<tr>
<td>5</td>
<td>85.74</td>
</tr>
<tr>
<td>10</td>
<td>75.59</td>
</tr>
</tbody>
</table>

![Bar chart showing average prediction accuracy for different algorithms with varying Alpha values.](chart.png)
Overall Accuracy and Percentage Split

Average Prediction Accuracy

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Avg. Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>C4.5</td>
<td>86.19</td>
</tr>
<tr>
<td>Naive-Bayes</td>
<td>87.78</td>
</tr>
<tr>
<td>BayesNet</td>
<td>91.68</td>
</tr>
<tr>
<td>Decision Table</td>
<td>88.52</td>
</tr>
</tbody>
</table>

Prediction Accuracy based on training/testing split

<table>
<thead>
<tr>
<th>Split Ratio</th>
<th>10</th>
<th>5</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>30/70</td>
<td>74.28</td>
<td>84.52</td>
<td>95.42</td>
<td>96.80</td>
</tr>
<tr>
<td>40/60</td>
<td>75.32</td>
<td>85.36</td>
<td>95.66</td>
<td>96.97</td>
</tr>
<tr>
<td>50/50</td>
<td>75.47</td>
<td>85.86</td>
<td>95.77</td>
<td>97.04</td>
</tr>
<tr>
<td>60/40</td>
<td>76.20</td>
<td>86.25</td>
<td>95.97</td>
<td>97.21</td>
</tr>
<tr>
<td>70/30</td>
<td>76.68</td>
<td>86.74</td>
<td>96.08</td>
<td>97.27</td>
</tr>
</tbody>
</table>
Effect of Dataset on Average Prediction Accuracy

<table>
<thead>
<tr>
<th>Dataset</th>
<th>10</th>
<th>5</th>
<th>1</th>
<th>0</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>74.26</td>
<td>85.13</td>
<td>96.06</td>
<td>97.32</td>
<td>88.19</td>
</tr>
<tr>
<td>2</td>
<td>75.63</td>
<td>86.14</td>
<td>96.53</td>
<td>97.52</td>
<td>88.96</td>
</tr>
<tr>
<td>3</td>
<td>76.88</td>
<td>85.96</td>
<td>94.74</td>
<td>96.33</td>
<td>88.47</td>
</tr>
</tbody>
</table>

Average Time Taken by Machine Learning Algorithms for prediction/classification

- C4.5: 85.95 seconds
- Naive Bayes: 57.93 seconds
- BayesNet: 57.84 seconds
- Decision Table: 205.37 seconds
Conclusion

- Machine Learning approach can help in defining security rules for SDN controller.
- A small probability of attack, obtained through ML approach, has significant effect on the SDN security.
- Achieved an average prediction accuracy of 91.68% with Bayesian Network (total 278,598 attacks).
- Blocking the subnet, rather than the individual IPs.
- The decline in accuracy in response to increasing $\alpha$ proves even small probability of attack cannot be ignored.

GitHub: [https://github.com/wedaa/LongTail-Log-Analysis](https://github.com/wedaa/LongTail-Log-Analysis)
Thank You!

Questions