Enhancing Deep Learning DGA Detection Models Using Separate Character Embedding

Vikash Yadav
Data Scientist, Royal Bank of Canada
Content

Problem
• Malware Life Cycle
• Domain Generation Algorithm

Existing Solution
• Traditional Approaches
• Machine Learning Approaches

RNN model
• Recurrent Neural Network based detecting technique
• Unified Architecture

Enhancement
• Separate Embedding Model
• Improvement in detecting unknown DGA
Problem
- Malware Life Cycle
- Domain Generation Algorithm

Existing Solution
- Traditional Approaches
- Machine Learning Approaches

RNN Model
- Recurrent Neural Network based detecting technique
- Unified Architecture

Enhancement
- Separate Embedding Model
- Improvement in detecting unknown DGA
Life Cycle of a Malware

Initial Compromise

Successful connection to Command & Control

Receive instruction to carry out malicious activities
Life Cycle of a Malware

- **Initial Compromise**
- **Successful connection to Command & Control**
- **Receive instruction to carry out malicious activities**
Domain Generation Algorithm (DGA)

- DGA uses a seed value and/or time-dependent element to avoid command and control domains or IPs being seized or sinkhole.

```python
def generate_domain(year, month, day):
    """Generates a domain name for the given date."""
    domain = ""
    for i in range(16):
        year = ((year ^ 8 * year) >> 11) ^ ((year & 0xFFFFFFFF0) << 17)
        month = ((month ^ 4 * month) >> 25) ^ 16 * (month & 0xFFFFFFFF8)
        day = ((day ^ (day << 13)) >> 19) ^ ((day & 0xFFFFFFFF) << 12)
        domain += chr(((year ^ month ^ day) % 25) + 97)
    return domain
```

Credit: Wikipedia
Problem

• Malware Life Cycle
• Domain Generation Algorithm

Existing Solution

• Traditional Approaches
• Machine Learning Approaches

RNN model

• Recurrent Neural Network based detecting technique
• Unified Architecture

Enhancement

• Separate Embedding Model
• Improvement in detecting unknown DGAs
Stopping DGA Malware – traditional approach

• Reverse engineer the binary to identify the DGA

• Blacklist the domain name and IP address of C2 server

• Sinkhole the C2 communication by registering the domain in advance
Stopping DGA Malware – traditional approach

• Reactive

• Time consuming

• Not scalable
Stopping DGA Malware – ML based approach

• NXDOMAIN DNS request based detection

• ML approach using handcrafted features

• RNN based detection
NXDOMAIN DNS request based detection

• DGA generates a large number of domains of which only a select few are registered to host a C2 server

• A client making requests to a large number of NXDomains is potentially hosting a DGA malware

Credit: Detecting DGA domains with recurrent neural networks and side information
ML approach using handcrafted features

- Entropy, Length of the domain etc.
- Number of vowels vs consonants in the domain
- Periodicity of the request
- Popularity of the domain
- Total byte sent and received
RNN based detection

- No explicit feature engineering required
- Proactive
- Easy to build and deploy
- Easy to retrain outdated models
- Highly scalable
- Highly accurate
Problem
• Malware Life Cycle
• Domain Generation Algorithm

Existing Solution
• Traditional Approaches
• Machine Learning Approaches

RNN model
• Recurrent Neural Network based detecting technique
• Unified Architecture

Enhancement
• Separate Embedding Model
• Improvement in detecting unknown DGAs
Unified RNN Model Architecture
Dataset for RNN model

Benign Domains
- Alexa top million domains
- Cisco top million domains
- ~1.8 million unique domains
  - google.com
  - youtube.com
  - facebook.com
  - baidu.com
  - wikipedia.org
  - yahoo.com

DGA Domains
- ~1.1 million unique domains
  - ydqtkptuwsa.org
  - bnnkqzwzmy.biz
  - glrmwqh.net
  - ibymtpyd.info
  - bxyozfikd.ws
  - nvjwoofansjbh.ru
Character Embedding

Input → Embedding Layer → RNN LSTM Layer → Output

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>39</td>
<td>47</td>
<td>47</td>
<td>39</td>
<td>44</td>
<td>37</td>
<td>12</td>
<td>35</td>
<td>47</td>
<td>45</td>
<td></td>
</tr>
</tbody>
</table>

embedding_layer

<table>
<thead>
<tr>
<th></th>
<th>-0.57</th>
<th>-0.01</th>
<th>0.51</th>
<th>2.95</th>
<th>1.4</th>
<th>1.03</th>
<th>1.24</th>
<th>0.04</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.09</td>
<td>0.95</td>
<td>0.29</td>
<td>0.61</td>
<td>-0.72</td>
<td>1.67</td>
<td>1.38</td>
<td>-0.96</td>
<td></td>
</tr>
<tr>
<td>-1.09</td>
<td>0.95</td>
<td>0.29</td>
<td>0.61</td>
<td>-0.72</td>
<td>1.67</td>
<td>1.38</td>
<td>-0.96</td>
<td></td>
</tr>
<tr>
<td>-0.57</td>
<td>-0.01</td>
<td>0.51</td>
<td>2.95</td>
<td>1.4</td>
<td>1.03</td>
<td>1.24</td>
<td>0.04</td>
<td></td>
</tr>
<tr>
<td>0.48</td>
<td>-0.82</td>
<td>0.49</td>
<td>-1.06</td>
<td>0.8</td>
<td>0.83</td>
<td>-2.97</td>
<td>-0.35</td>
<td></td>
</tr>
<tr>
<td>-0.51</td>
<td>-0.16</td>
<td>-0.9</td>
<td>-1.65</td>
<td>-0.2</td>
<td>0.6</td>
<td>0.34</td>
<td>0.94</td>
<td></td>
</tr>
<tr>
<td>0.27</td>
<td>0.4</td>
<td>-2</td>
<td>0.12</td>
<td>2.09</td>
<td>-2.12</td>
<td>2.38</td>
<td>-1.4</td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>-0.3</td>
<td>2.02</td>
<td>1.68</td>
<td>-0.98</td>
<td>0.63</td>
<td>0.19</td>
<td>1.36</td>
<td></td>
</tr>
<tr>
<td>-1.09</td>
<td>0.95</td>
<td>0.29</td>
<td>0.61</td>
<td>-0.72</td>
<td>1.67</td>
<td>1.38</td>
<td>-0.96</td>
<td></td>
</tr>
<tr>
<td>-2.02</td>
<td>-1.69</td>
<td>-0.46</td>
<td>0.26</td>
<td>1.97</td>
<td>0.58</td>
<td>-0.16</td>
<td>-0.45</td>
<td></td>
</tr>
</tbody>
</table>
Recurrent Neural Network
I grew up in France ... I speak fluent ___?
I grew up in **France** ... I speak fluent **French**?
Long Short-Term Memory
Unified RNN Model Architecture

Input Domain → 1D integer sequence of fixed length → Embedding Layer, Output Dimension=8 → Bidirectional LSTM Layer (256) → Output Sigmoid
Result

Test Accuracy for known DGA types

<table>
<thead>
<tr>
<th>Label</th>
<th>Record Count</th>
<th>Unified Model Accuracy</th>
<th>F score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benign</td>
<td>750153</td>
<td>0.9946</td>
<td>0.9874</td>
</tr>
<tr>
<td>Malicious</td>
<td>415976</td>
<td>0.9845</td>
<td></td>
</tr>
</tbody>
</table>

Very high accuracy for detecting known DGA types
Detection Accuracy for unknown DGA types –

<table>
<thead>
<tr>
<th>Label</th>
<th>Record Count</th>
<th>Unified Model Accuracy</th>
<th>Sample</th>
</tr>
</thead>
<tbody>
<tr>
<td>chinad</td>
<td>1000</td>
<td>0.996000</td>
<td>qowhi81jvoid4j0m.biz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>29cqdf6obnq462yv.com</td>
</tr>
<tr>
<td>ramnit</td>
<td>15080</td>
<td>0.718899</td>
<td>jrkaxdlkvhgsiyknhw.com</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>mtsoexdphaqliva.com</td>
</tr>
<tr>
<td>shifu</td>
<td>2554</td>
<td>0.438919</td>
<td>urkaelt.info</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>rsymdhk.info</td>
</tr>
</tbody>
</table>
Limitation

• Accuracy suffers for unknown DGA type

• Possible overfitting to training data

• Embedding representation is specific to the training data and is not representative of English language
- Malware Life Cycle
- Domain Generation Algorithm

- Traditional Approaches
- Machine Learning Approaches

- Recurrent Neural Network based detecting technique
- Unified Architecture

- Separate Embedding Model
- Improvement in detecting unknown DGAs
Training Separate Character Embedding Model

• Learn embedding representation to capture the contextual information of the English language by training on articles from popular US newspapers

• Use this general representation to transform domain names

• The error is calculated based on the model’s ability to predict the next character in the sequence
Learning Character Embedding

- Input
- 1D integer sequence of fixed length
- Embedding Layer, Output Dimension=8
- Bidirectional LSTM Layer (256)
- Bidirectional LSTM Layer (256)
- Output Sigmoid
Separate Character Embedding based RNN Model Architecture

1D integer sequence of fixed length → Pre-trained Embedding Layer, Output Dimension=8 → Bidirectional LSTM Layer (256) → Output Sigmoid
# Result

## Test Accuracy for known DGA types

<table>
<thead>
<tr>
<th>Label</th>
<th>Record Count</th>
<th>Unified Model Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benign</td>
<td>750153</td>
<td>0.9946</td>
</tr>
<tr>
<td>Malicious</td>
<td>415976</td>
<td>0.9845</td>
</tr>
</tbody>
</table>
## Result

### Test Accuracy for known DGA types

<table>
<thead>
<tr>
<th>Label</th>
<th>Record Count</th>
<th>Unified Model Accuracy</th>
<th>Separate Embedding Model Accuracy</th>
<th>F Score Embedding Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benign</td>
<td>750153</td>
<td>0.9946</td>
<td>0.9922</td>
<td>0.9875</td>
</tr>
<tr>
<td>Malicious</td>
<td>415976</td>
<td>0.9845</td>
<td>0.9889</td>
<td></td>
</tr>
</tbody>
</table>
Result

Detection Accuracy for unknown DGA types –

<table>
<thead>
<tr>
<th>Label</th>
<th>Record Count</th>
<th>Unified Model Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>chinad</td>
<td>1000</td>
<td>0.996000</td>
</tr>
<tr>
<td>ramnit</td>
<td>15080</td>
<td>0.718899</td>
</tr>
<tr>
<td>shifu</td>
<td>2554</td>
<td>0.438919</td>
</tr>
</tbody>
</table>
### Detection Accuracy for unknown DGA types –

<table>
<thead>
<tr>
<th>Label</th>
<th>Record Count</th>
<th>Unified Model Accuracy</th>
<th>Separate Embedding Model Accuracy</th>
<th>% Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>chinad</td>
<td>1000</td>
<td>0.996000</td>
<td>0.998000</td>
<td>0.2</td>
</tr>
<tr>
<td>ramanit</td>
<td>15080</td>
<td>0.718899</td>
<td>0.768833</td>
<td>5.0</td>
</tr>
<tr>
<td>shifu</td>
<td>2554</td>
<td>0.438919</td>
<td>0.831245</td>
<td>39.23</td>
</tr>
</tbody>
</table>
Wrapping Up

- LSTM based RNNs are highly effective in detecting DGA
- Our proposed changes can improve detection accuracy for unknown DGA malware
- RNN based detection is proactive rather than reactive