The Applications of Deep Learning on Traffic Identification

Zhanyi Wang
whoami

• Zhanyi Wang, Signal & Information Processing, Ph.D.
• Data mining researcher @ Qihoo 360
• Machine Learning - rich experience/Cyber Security – beginner
• Colleagues
  • Zhuo Zhang, Bo Liu, Chuanming Huang
• Focus on "Data-driven Security“
  • Statistical Analysis
  • Deep Learning
  • Pattern Recognition
  • Anomaly Detection
Outline

• Traditional Methods of Traffic Identification
• Neural Networks and Deep Learning
• Applications
  • Protocol Classification
  • Automatic Feature Learning
  • Application Identification
  • Unknown Protocol Identification
• Conclusions and Future Work
Traditional Methods of Traffic Identification

• An accurate mapping of traffic to protocols or applications is important for network management, anomaly detection

• Base on special or predefined ports
  • Standard HTTP port is 80, default port of SSL is 443
  • Weakness: doesn’t work when ports are new or changed

• Signature-based traffic identification
  • Static, dynamic and distinguishable features
  • Weakness: very time-consuming and labor-intensive
Why we choose deep learning

- Base on statistical features and machine learning
  - Identification process: automatic
  - Difficulty: how to choose appropriate features

- Is there any ways not to depend on experts?
- Is unsupervised feature learning possible?
- Answer: Deep Learning in artificial intelligence
The power of deep learning techniques

- Image

- Speech

- NLP
Neural Networks

- Neural Networks
- Basic unit
  - neuron
- Structure:
  - Input layer
  - Hidden layers
  - Output layer
- Each pair of neighboring layers is connected
- neurons in the same layer are not connected
Auto-Encoder

- Auto-Encoder
- a specific type of neural network
- Only one hidden layer
- Output layer is identical with input layer!
Auto-Encoder in image recognition

- handwritten digits experiment
Stacked Auto-Encoder

- Stacked Auto-Encoder (SAE)
- Consisting of multiple layers of AE
- SAE is a neural network essentially

- Use greedy layer-wise training
- Use fine-tuning
Image VS Payload

• Do they look alike?

TCP flow Payloads

255 210 21 53 ...
255 52 3 0 ...
52 6 0 85 ...
...
...

474554206874...727665720020...732048545450...33a31353a323...

732048545450...33a31353a323...

115 32 72 84 84 80...

range of values: [0,255]
Both 256 numbers!
Implementation of protocol identification

- Data: collected in our intranet
- Experimental environment
  - Scheme 1 - CPU: E5-2630 * 2 + GPU: AMD S9150 * 4
  - Scheme 2 - only use CPU cluster: 2~10 servers
- Training time: less than 3 hours in Scheme 1
Parallel computing based on multi-GPU

- Large amount of data
  - Hundreds of millions original data per day
- Too many parameters
  - More than 5 millions
- Very long training time
  - Several days if just use CPU
- Solution
  - OpenCL
  - Multi-GPU
  - Multi-machine

*OpenCL is a framework for writing programs that execute across heterogeneous platforms consisting of CPUs, GPUs, DSPs, FPGAs and other processors.
Data association & transformation

- up_payload1: 474554206874......
- down_payload1: 727665720020......
- up_payload2: 732048545450......
- down_payload2: 33a31353a323......

Concatenate together:

474554206874......727665720020......732048545450......33a31353a323......

Length = 1024

HEX to DEC:

71 69 84 32 104 116......114 118 101 114 0 32......115 32 72 84 84 80......51 163 19 83 163 35......
The process of protocol identification

0.85, 0.6, 0.02, 0.00, 0.01, 0.00 ...... How to predict?

Deep Learning Model

71 69 84 32 104 116......114 118 101 114 0 32......115 32 72 84 84 80......51 163 19 83 163 35......

116 199 225 220 82 116......14 211 51 17 37 110......139 18 253 58 80 172......172 26 91 146 1 23......

180 39 27 205 22 76......226 123 177 230 163 14......77 76 150 167 3 237......183 9 78 44 30 162......

......
The process of protocol identification

Logistic Regression
More than one outputs

| 0.85, 0.6, 0.02, 0.00, 0.01, 0.00 ...... |
| MySQL, SSH, FTP_CONTROL, HTTP_Proxy, SMB, SMTP ...... |
| >0.5? |
| Predictions: 1. MySQL 2. SSH |

Softmax Regression
Just one output

| 0.91, 0.01, 0.02, 0.00, 0.01, 0.00 ...... |
| MySQL, SSH, FTP_CONTROL, HTTP_Proxy, SMB, SMTP ...... |
| Maximum? |
| Prediction: MySQL |
## Protocol Classification

- Overall Precision: >99%  Average Precision: 97.9%

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Precision</th>
<th>Protocol</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMB</td>
<td>1.0000</td>
<td>RSYNC</td>
<td>0.9987</td>
</tr>
<tr>
<td>DCE_RPC</td>
<td>1.0000</td>
<td>Redis</td>
<td>0.9985</td>
</tr>
<tr>
<td>NetBIOS</td>
<td>1.0000</td>
<td>FTP_CONTROL</td>
<td>0.997</td>
</tr>
<tr>
<td>TDS</td>
<td>1.0000</td>
<td>HTTP_Connect</td>
<td>0.9967</td>
</tr>
<tr>
<td>SSH</td>
<td>0.9996</td>
<td>SMTP</td>
<td>0.9949</td>
</tr>
<tr>
<td>Kerberos</td>
<td>0.9996</td>
<td>Whois-DAS</td>
<td>0.9943</td>
</tr>
<tr>
<td>LDAP</td>
<td>0.9996</td>
<td>IMAPS</td>
<td>0.9814</td>
</tr>
<tr>
<td>BitTorrent</td>
<td>0.9992</td>
<td>Apple</td>
<td>0.964</td>
</tr>
<tr>
<td>MySQL</td>
<td>0.9989</td>
<td>SSL</td>
<td>0.9513</td>
</tr>
<tr>
<td>DNS</td>
<td>0.9989</td>
<td>HTTP_Proxy</td>
<td>0.9174</td>
</tr>
</tbody>
</table>
Automatic Feature Learning

- take the sum of all absolute weights $|W_{ij}^{(1)}|$ with regard to every node in the input layer as the value

$$v_j = \sum_{i=1}^{n} |w_{ij}^{(1)}|$$

- $v_j$: the larger, the more important the j-th feature is.
Automatic Feature Learning

- The distribution of TOP 25 (A) & 100 (B) important features

- A

- B
Automatic Feature Learning

- The distribution of 300 least important features

- C
Application Identification

Process:
- svchost.exe
- Thunder.exe
- lsass.exe
- outlook.exe
- iexplore.exe

Payloads:
- 474554202f7461736b...
- 30840000068702020c...
- 54545033a31353a323...
- 727665720020732048...
- 47455420687445d4a1...

Deep Learning Model

New payloads

Which application?
Application Identification

- More than 800 applications in our training data
- Precision: 96.3% (testing on applications that are more than 200 records)

<table>
<thead>
<tr>
<th>Application</th>
<th>Precision</th>
<th>Protocol</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>foxmail.exe</td>
<td>1.0000</td>
<td>xshell.exe</td>
<td>0.9813</td>
</tr>
<tr>
<td>wpservice.exe</td>
<td>1.0000</td>
<td>baidumusic.exe</td>
<td>0.9808</td>
</tr>
<tr>
<td>taobaoprotect.exe</td>
<td>0.9984</td>
<td>fection.exe</td>
<td>0.9779</td>
</tr>
<tr>
<td>wechat.exe</td>
<td>0.9983</td>
<td>qqmusic.exe</td>
<td>0.9730</td>
</tr>
<tr>
<td>liebao.exe</td>
<td>0.9978</td>
<td>qqdownload.exe</td>
<td>0.9615</td>
</tr>
<tr>
<td>weibo2015.exe</td>
<td>0.9974</td>
<td>yodaodict.exe</td>
<td>0.9542</td>
</tr>
<tr>
<td>lsass.exe</td>
<td>0.9945</td>
<td>itunes.exe</td>
<td>0.9429</td>
</tr>
<tr>
<td>sogoucloud.exe</td>
<td>0.9897</td>
<td>outlook.exe</td>
<td>0.9219</td>
</tr>
<tr>
<td>qq.exe</td>
<td>0.9884</td>
<td>thunder.exe</td>
<td>0.9168</td>
</tr>
<tr>
<td>pplive.exe</td>
<td>0.9870</td>
<td>iexplore.exe</td>
<td>0.8860</td>
</tr>
</tbody>
</table>
Unknown Protocol Identification

- Randomly choose 10,000 records that labeled “unknown” by traditional ways
- our method can also find out 6,337 of them

<table>
<thead>
<tr>
<th>Protocol</th>
<th>number</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL</td>
<td>1956</td>
<td>29.12%</td>
</tr>
<tr>
<td>DCE_RPC</td>
<td>1454</td>
<td>21.65%</td>
</tr>
<tr>
<td>Skype</td>
<td>873</td>
<td>13.00%</td>
</tr>
<tr>
<td>Kerberos</td>
<td>517</td>
<td>7.70%</td>
</tr>
<tr>
<td>MSN</td>
<td>360</td>
<td>5.36%</td>
</tr>
<tr>
<td>Google</td>
<td>311</td>
<td>4.63%</td>
</tr>
<tr>
<td>DNS</td>
<td>260</td>
<td>3.87%</td>
</tr>
<tr>
<td>RTMP</td>
<td>234</td>
<td>3.48%</td>
</tr>
<tr>
<td>TDS</td>
<td>202</td>
<td>3.01%</td>
</tr>
<tr>
<td>H323</td>
<td>170</td>
<td>2.53%</td>
</tr>
</tbody>
</table>
Conclusions and Future Work

• The Applications of Deep Learning on Traffic Identification
  • Protocol Classification
  • Automatic Feature Learning
  • Application Identification
  • Unknown Protocol Identification

• Future Work
  • Applying Convolutional Neural Networks (CNN) model
  • Analysis of encrypted traffics
Thanks!

Zhanyi Wang
wangzhanyi@360.cn