Deep Fingerprinting: Undermining Website Fingerprinting Defenses with Deep Learning

Payap Sirinam  Marc Juarez  Mohsen Imani  Matthew Wright
Problem:
Could website fingerprinting (WF) attacks be improved to undermine the effectiveness of the new defenses?

Key contribution:
A new WF attack based on Convolutional Neural Network (CNN) that achieves very high accuracy in both a closed-world setting and a more realistic setting.
Threat Model
Threat Model: What is Tor?

A random three-circuit hop between client and server
How does WF work?

- Traffic analysis attack to recover browsing history of a client
- Local and passive adversary
- A classification problem
Threat Model: Assumptions

**Client-setting:**
- Closed- vs open-world
- Browsing behavior

**Web:**
- Template websites

**Adversary:**
- Page load parsing
- No background traffic
- Replicability
### Threat Model: Assumptions

<table>
<thead>
<tr>
<th>Item</th>
<th>Closed-world</th>
<th>Open-world</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Set</strong></td>
<td>Finite set of websites</td>
<td>- Monitored set</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Unmonitored set</td>
</tr>
<tr>
<td><strong>Classification</strong></td>
<td>Multi-class (websites)</td>
<td>Binary</td>
</tr>
<tr>
<td><strong>Goal</strong></td>
<td>Predict website</td>
<td>Predict if a monitored or unmonitored website</td>
</tr>
</tbody>
</table>

**Closed-world**

- Universe
- \( M \) (finite)
- \( M' \) (infinite & diverse)

**Open-world**
**Threat Model: Performance Metrics**

**Closed-world evaluation:**

\[ \text{Accuracy} = \frac{P_{\text{correct}}}{N} \]

**Open-world evaluation:**

<table>
<thead>
<tr>
<th>True condition</th>
<th>Test result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Positive (monitored)</td>
<td>Positive (monitored) = True Positive (TP)</td>
</tr>
<tr>
<td>Negative (unmonitored)</td>
<td>False Positive (FP)</td>
</tr>
</tbody>
</table>

\[ \text{Precision} = \frac{TP}{TP + FP} \]

\[ \text{Recall} = \text{Sensitivity} = TPR = \frac{TP}{TP + FN} \]

\[ FPR = \frac{FP}{FP + TN} \]
Prior Work
### Prior Work: WF attacks (state-of-the-art)

<table>
<thead>
<tr>
<th>Name</th>
<th>Assumption</th>
<th>Algorithm</th>
<th>Features</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>k-NN</td>
<td>Closed-world</td>
<td>k-NN</td>
<td>Packet ordering, # incoming &amp; outgoing cells, # bursts, etc.</td>
<td>91% accuracy</td>
</tr>
<tr>
<td></td>
<td>Open-world</td>
<td></td>
<td></td>
<td>86% TPR 0.6% FPR</td>
</tr>
<tr>
<td>CUMUL</td>
<td>Closed-world</td>
<td>SVM</td>
<td>Cumulative sum of packet lengths</td>
<td>91% accuracy</td>
</tr>
<tr>
<td></td>
<td>Open-world</td>
<td></td>
<td></td>
<td>- Multi-class: 96% TPR 9.61% FPR</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- Two-class: 96% TPR 1.9% FPR</td>
</tr>
<tr>
<td>k-FP</td>
<td>Closed-world</td>
<td>RF</td>
<td># packets in a sequence, packet inter-arrival time</td>
<td>91% accuracy</td>
</tr>
<tr>
<td></td>
<td>Open-world</td>
<td>k-NN</td>
<td></td>
<td>88% TPR 0.5% FPR</td>
</tr>
</tbody>
</table>
**Prior Work: WF defenses**

<table>
<thead>
<tr>
<th>Fundamental strategy:</th>
<th>Challenge:</th>
<th>Lightweight systems:</th>
</tr>
</thead>
</table>
| Add dummy packets and/or delay packets to make WF features less distinctive | Incur high bandwidth and latency overheads | • WTF-PAD  
• Walkie-Talkie |
Deep Learning Models
Deep Learning Models: Techniques

Stacked

Denoising

Autoencoders

(SDAE)
Convolutional Neural Network (CNN)
Deep Learning Models: Data Collection

- 10 low-end machines
- Visits were sequential
- Closed-world: visits to each site were split in 5 chunks, in a round-robin fashion; Open-world: only visited each site once then took a screenshot of the homepages
- Discarded corrupted traffic traces

**Closed-world dataset:** 95 sites with 1,000 visits

**Open-world dataset:** 5,000 sites with 40,716 traffic traces

**Defended dataset:** closed- and open-world datasets of size similar to the undefended ones, but now with traces protected
A website trace is represented as a sequence of tuples:

\(< \text{timestamp}, \pm \text{packet\_size} >\)

Hyperparameters selection for DF model from Extensive Candidates Search method
Deep Learning Models: DF Model Architecture

**Block 1**
- Convolutional 1D
- Batch Normalization
- Activation Layer: ELU (alpha = 1.0)
- Convolutional 1D
- Batch Normalization
- Activation Layer: ELU (alpha = 1.0)
- Max Pooling
- Dropout

**Block 2**
- Convolutional 1D
- Batch Normalization
- Activation Layer: ReLU
- Convolutional 1D
- Batch Normalization
- Activation Layer: ReLU
- Max Pooling
- Dropout

**Block 3**
- Convolutional 1D
- Batch Normalization
- Activation Layer: ReLU
- Convolutional 1D
- Batch Normalization
- Activation Layer: ReLU
- Max Pooling
- Dropout

**Block 4**
- Convolutional 1D
- Batch Normalization
- Activation Layer: ReLU
- Convolutional 1D
- Batch Normalization
- Activation Layer: ReLU
- Max Pooling
- Dropout

**Fully-Connected (FC) Layers**
- **FC Layer 1**
  - 512 hidden units
  - Batch Normalization
  - Activation Layer
  - Dropout
  - Rate = 0.7
- **FC Layer 2**
  - 512 hidden units
  - Batch Normalization
  - Activation Layer
  - Dropout
  - Rate = 0.5

**Output Prediction**
- FC Layer
- Activation Layer
- N hidden units
- Softmax
Accuracy on the non-defended dataset for state-of-the-art attacks

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDAE</td>
<td>92.3</td>
</tr>
<tr>
<td>DF</td>
<td><strong>98.3</strong></td>
</tr>
<tr>
<td>K-NN</td>
<td>95.0</td>
</tr>
<tr>
<td>CUMUL</td>
<td>97.3</td>
</tr>
<tr>
<td>K-FP</td>
<td>95.5</td>
</tr>
</tbody>
</table>
Deep Learning Models: Evaluation in Closed-world

Impact of the number of training epochs and number of training traces on DF accuracy
Deep Learning Models: Evaluation in Closed-world

<table>
<thead>
<tr>
<th>Defenses</th>
<th>Overhead (%)</th>
<th></th>
<th>Accuracy of WF attacks on defended datasets (%)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Band-width</td>
<td>Latency</td>
<td>SDAE</td>
<td><strong>DF</strong></td>
</tr>
<tr>
<td>WTF-PAD</td>
<td>64</td>
<td>0</td>
<td>36.9</td>
<td><strong>90.7</strong></td>
</tr>
<tr>
<td>Walkie-Talkie</td>
<td>31</td>
<td>34</td>
<td>23.1</td>
<td><strong>49.7</strong></td>
</tr>
</tbody>
</table>
Deep Learning Models: Evaluation in Open-world

The impact of the amount of unmonitored training data on TPR and FPR (non-defended dataset)
Critical Evaluation
Critical Evaluation: Unrealistic Assumptions

- User settings (open world, multi-tab browsing behavior)
- Adversary capabilities (access to TBB version, Internet connection)
- Nature of the Web (website variance over time)
Critical Evaluation: Base-rate Fallacy

Bayes Detection Rate (BDR) = \( P(M|C) = \frac{P(C|M)P(M)}{P(M)P(C|M) + P(\neg M)P(C|\neg M)} \)

where:

- \( P(M|C) \): probability that a traffic trace actually corresponds to a monitored webpage given that the classifier recognized it as monitored
- \( P(C|M) \): TPR
- \( P(C|\neg M) \): FPR
- \( P(M) = \frac{|Monitored|}{|World|} \) (assuming uniform distribution of pages)
Future Directions & Conclusions
Future Directions & Conclusions

- Improving open-world classification
- Exploring the trade-offs between scalability (due to attack costs) and accuracy

→ The need to improve WF defenses to be more robust against attacks using deep learning
Thank you!
Deep Learning Models: Evaluation in Open-world

**Precision – Recall curves**

(a) Non-defended dataset

(b) WTF-PAD dataset

(c) W-T dataset