Bayes, not Naïve
Provable Security of Website Fingerprinting Defences

Giovanni Cherubin

ISG Seminar, Royal Holloway University of London
9 February, 2017
- Background
- Provable bounds by Cai et al.
- Bayes-based bounds, $(\varepsilon, \Phi)$-privacy
- Dependence on features
Arms Race

Major attacks and defences up to February 2017

Back et al. introduce WF

Liberatore and Levine (Naive Bayes)

Wright et al. (Naive Bayes)

Herrman (Multinomial Naive Bayes)

Panchenko et al. (Naive Bayes)

Dyer et al. (k-NN)

Wang et al. (SVM)

Hayes et al. (RF)

2001

2006


Cherubin et al., ALPaCA

LLAMA

Juarez et al., WTF-PAD

Dyer et al., BuFLO

Cai et al., Tamaraw

Wang et al., Supersequences

Panchenko et al., Decoy Pages

Wright et al., traffic morphing

Luo et al., HTTPOS
Website Fingerprinting (WF)
Website Fingerprinting (WF)
Threat model
Threat model

Adversary:
Threat model

Adversary:

• **Locally** and **passively** collects network traffic
Threat model

Adversary:

- **Locally** and **passively** collects network traffic
- Does **not** know the **web server’s IP** (e.g., victim uses Tor/VPN)
Threat model

Adversary:

- **Locally** and **passively** collects network traffic
- Does **not** know the **web server’s IP** (e.g., victim uses Tor/VPN)
- Cannot decrypt traffic. Only information he gets from a page load is a **packet sequence**: 

```
↑   ↓   ↑   ↑   ↓   ↓
```

```
t```

---

**↑**  **↓**  **↑**  **↑**  **↓**  **↓**

```
t```
Threat model

Adversary:

• **Locally** and **passively** collects network traffic

• Does **not** know the **web server’s IP** (e.g., victim uses Tor/VPN)

• Cannot decrypt traffic. Only information he gets from a page load is a **packet sequence**:

  ![Diagram](image)

• Knows set of web **pages** the user may visit (Closed World)
WF Adversary

= (Φ, Train)

Training algorithm for an ML classifier (e.g., Logistic Regression, Random Forest, SVM)
WF Adversary

$WF = (\Phi, \text{Train})$

Training algorithm for an ML classifier (e.g., Logistic Regression, Random Forest, SVM)

$x = \Phi(\text{feature object})$
WF Adversary

\[ x = \Phi (f) \]

Training algorithm for an ML classifier (e.g., Logistic Regression, Random Forest, SVM)

\[ f = \text{Train}((x_1, y_1), \ldots, (x_n, y_n)) \]

feature object

ML classifier
Evaluating WF Attacks

\[(x_1, y_1), \ldots, (x_n, y_n)\]

training set

\[f(x)\]

test object

\[R_f = \Pr (f(x) \neq y)\]

Implicit: \(x_i = \Phi(p_i)\)
Probabilistic & Deterministic Defences

Original
Probabilistic & Deterministic Defences

Original

BuFLO $(\sigma, \theta, \tau)$
Probabilistic & Deterministic Defences

Original

BuFLO ($\sigma$, $\theta$, $\tau$)

Decoy Pages
- Background

- **Provable bounds by Cai et al.**

- Bayes-based bounds, \((\varepsilon, \Phi)\)-privacy

- Dependence on features
“Lookup-Table” Approach

(Cai et al., ’14)

Idealised Adversary: knows exactly what packet sequences each web page may generate. Counts the collisions.
“Lookup-Table” bound

- Highly affected by noise
- Needs to be computed on partial data
- Only deterministic defences
Background

Provable bounds by Cai et al.

Bayes-based bounds, $(\varepsilon, \Phi)$-privacy

Dependence on features
“Bayes error” approach

= (Φ, Train)
Distinguishing Web Pages

\[ P_x | y=\text{google.com} \]

\[ P_x | y=\text{freeimages.com} \]

Total communication time
Distinguishing Web Pages

\[ P_x | y=\text{google.com} \quad \text{and} \quad P_x | y=\text{freeimages.com} \]

Total communication time
Distinguishing Web Pages

$P_x | y=\text{google.com}$  $P_x | y=\text{freeimages.com}$

Total communication time
Distinguishing Web Pages

\[ P_x | y=\text{google.com} \quad P_x | y=\text{freeimages.com} \]

Total communication time

\[ R^*: \text{Bayes Error} \]
“Bayes error” approach

\[ f = \text{Train} \left( (x_1, y_1), \ldots, (x_n, y_n) \right) \]

\[ R^f = \text{Pr} \left( f (x_{n+1}) \neq y_{n+1} \right) \]

Implicit: \( x_i = \Phi(p_i) \)
“Bayes error” approach

\[ f = \text{Train} \left( (x_1, y_1), \ldots, (x_n, y_n) \right) \]

\[ R^f = \text{Pr} \left( f \left( x_{n+1} \right) \neq y_{n+1} \right) \]

\[ R^f \geq R^* \]

Implicit: \( x_i = \Phi(p_i) \)
"Bayes error" approach

\[ f = \text{Train} \left( (x_1, y_1), \ldots, (x_n, y_n) \right) \]

\[ R_f = \Pr \left( f(x_{n+1}) \neq y_{n+1} \right) \]

\[ R_f \geq R^* \geq \hat{R}^* \]

Implicit: \( x_i = \Phi(p_i) \)
Bayes Error Estimate
(Cover & Hart, ’67)

Asymptotically,

\[ R^* \geq \frac{L - 1}{L} \left( 1 - \sqrt{1 - \frac{L}{L - 1} R^{NN}} \right) \]
NN-based estimate
(\(\varepsilon, \Phi\))-privacy

**Problem** An error estimate \(\hat{R}^*\) alone does not convey information about the setting

\[
\varepsilon = \frac{\hat{R}^*}{R^G}
\]
Background

Provable bounds by Cai et al.

Bayes-based bounds, $(\varepsilon, \Phi)$-privacy

**Dependence on features**
Do We Need Features?

**Theorem** Using the full original data performs no worse than using any transformation (i.e., feature) of the original data *asymptotically*.

**Problem** *Curse of dimensionality* and difficult *separability* in original space.
Did Feature Sets Improve?

(How much)

Liberatore & Levine
Dyer et al.
Wang et al.
Panchenko et al.
Hayes & Danezis

Bayes Error Estimate

No Defence
Decoy Pages
BuFLO
Tamaraw

Attack's Year

2006
2012
2014
2016
2017
Features

Determine an **efficient** set of features $\Phi$:

- low computational-memory *complexity* to extract objects
- good *accuracy* w.r.t. the optimal one: balance dimensionality and information

$$R^\Phi \leq R^{\Phi'} + \varepsilon$$
TL;DL

• WF attacks too long without security proofs

• Bound based on lookup-table suffer from noise

• $R^*$ estimate bounds an adversary $A = (\Phi, \cdot)$

• $R^*$ can be used to evaluate features

• Improving $\Phi$ is becoming more difficult
Future Directions

Efficient feature set:

• how to show that a feature set is efficient?

• can we construct one automatically?

New estimates of $R^*$:

• tighter bounds

• weaker assumptions on data

Other applications of the method
Bayes, not Naïve

Provable Security of Website Fingerprinting Defences

Giovanni Cherubin

ISG Seminar, Royal Holloway University of London

9 February, 2017
## Comparision with Cai et al.

<table>
<thead>
<tr>
<th>Defence</th>
<th>$R^*$ estimate</th>
<th>Cai et al.</th>
<th>Cai et al. (full information)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BuFLO</td>
<td>57%</td>
<td>53%</td>
<td>19%</td>
</tr>
<tr>
<td>Tamaraw</td>
<td>69%</td>
<td>91%</td>
<td>11%</td>
</tr>
</tbody>
</table>
### (ε, Φ)-privacy

<table>
<thead>
<tr>
<th>Defence</th>
<th>(ε, Φ)-privacy</th>
<th>Time OH</th>
<th>Packet OH</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Defence</td>
<td>(0.06, k-NN)</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Decoy Pages</td>
<td>(0.43, k-NN)</td>
<td>29%</td>
<td>98%</td>
</tr>
<tr>
<td>BuFLO</td>
<td>(0.58, k-FP)</td>
<td>24%</td>
<td>11%</td>
</tr>
<tr>
<td>Tamaraw</td>
<td>(0.70, k-NN)</td>
<td>334%</td>
<td>161%</td>
</tr>
</tbody>
</table>
Other applications

\[(x_1, y_1), \ldots, (x_n, y_n)\]

training set

\[x\]

test object

\[f(x)\]

\[R^f = P(f(x) \neq y)\]
Bayes, not Naïve
Provable Security of Website Fingerprinting Defences

Giovanni Cherubin

ISG Seminar
9 February, 2017