DeepSE-WF

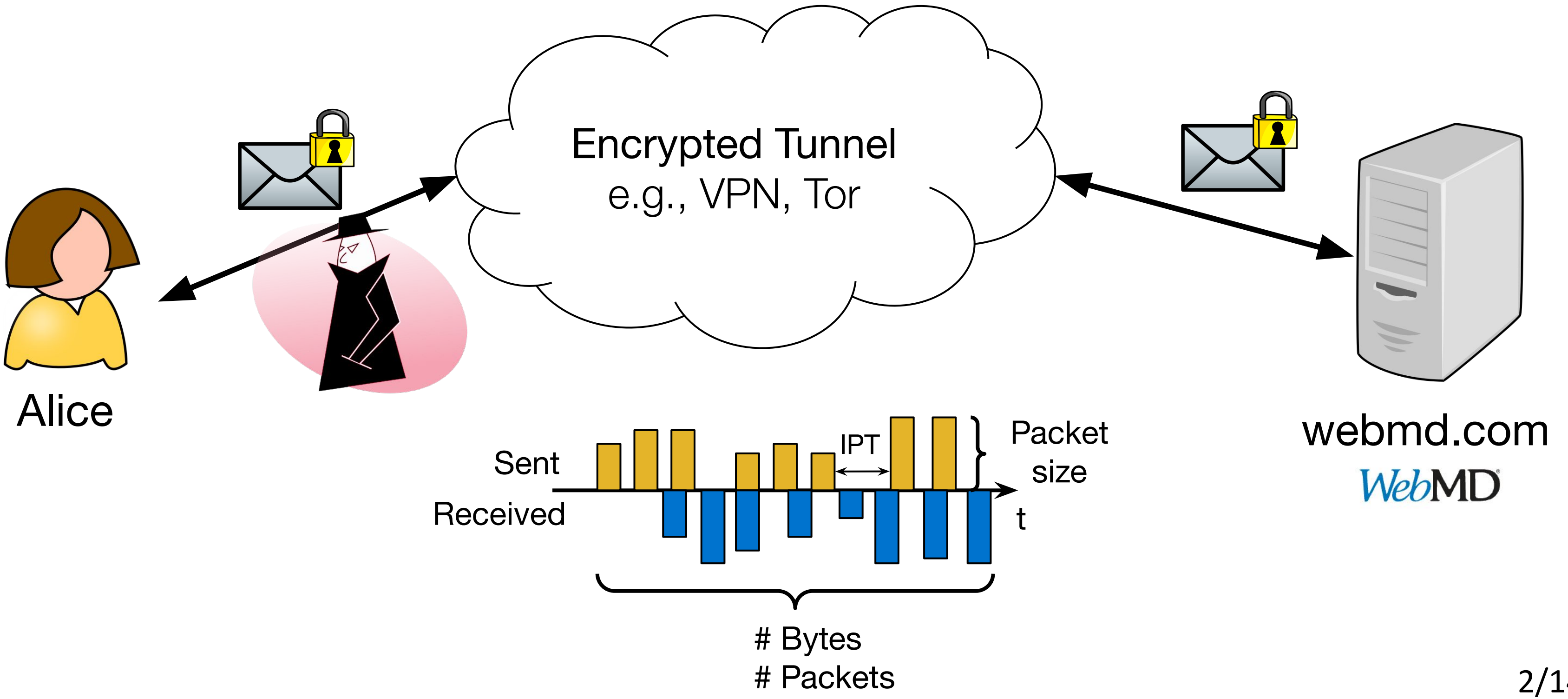
Unified Security Estimation for Website Fingerprinting Defenses

Alexander Veicht, Cedric Renggli, Diogo Barradas

Privacy Enhancing Technologies Symposium
Lausanne, Switzerland
11 July, 2023

ETH zürich
Universität Zürich UZH
UNIVERSITY OF WATERLOO

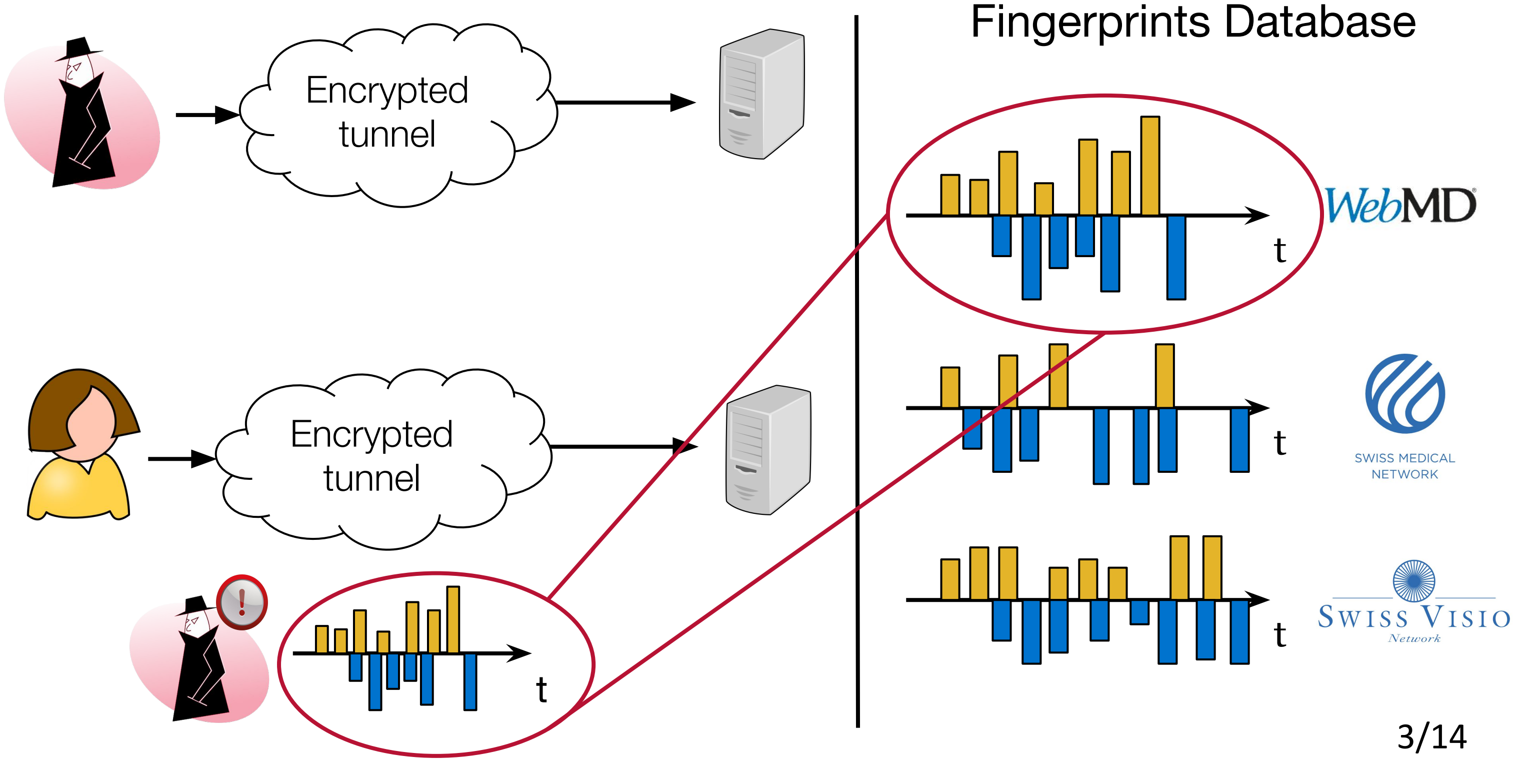
Encrypted Connections Leak Metadata



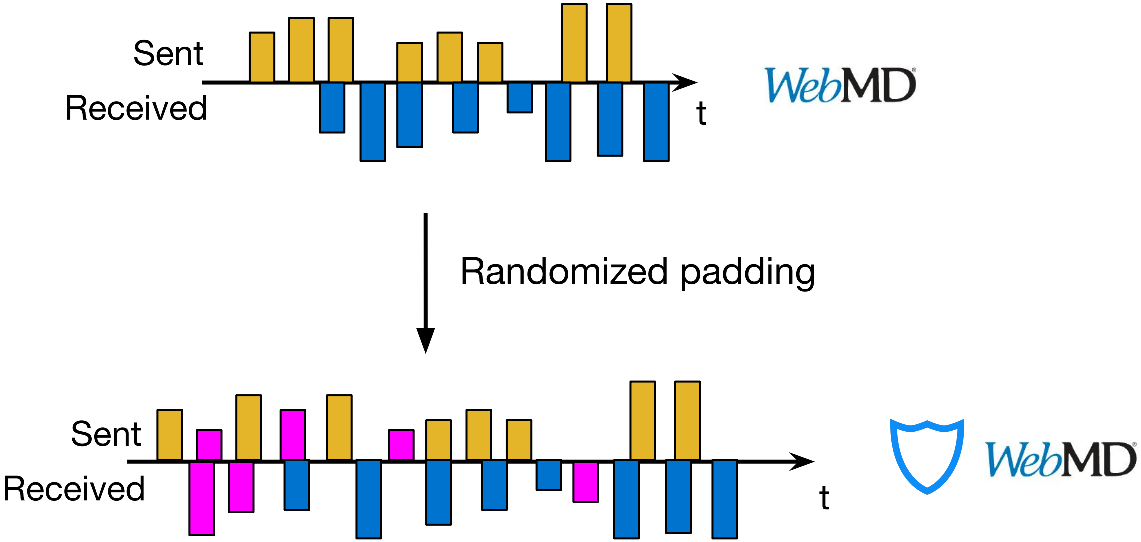
Website Fingerprinting (WF)

Step 1:
Build fingerprint database

Step 2:
Match Alice's traffic

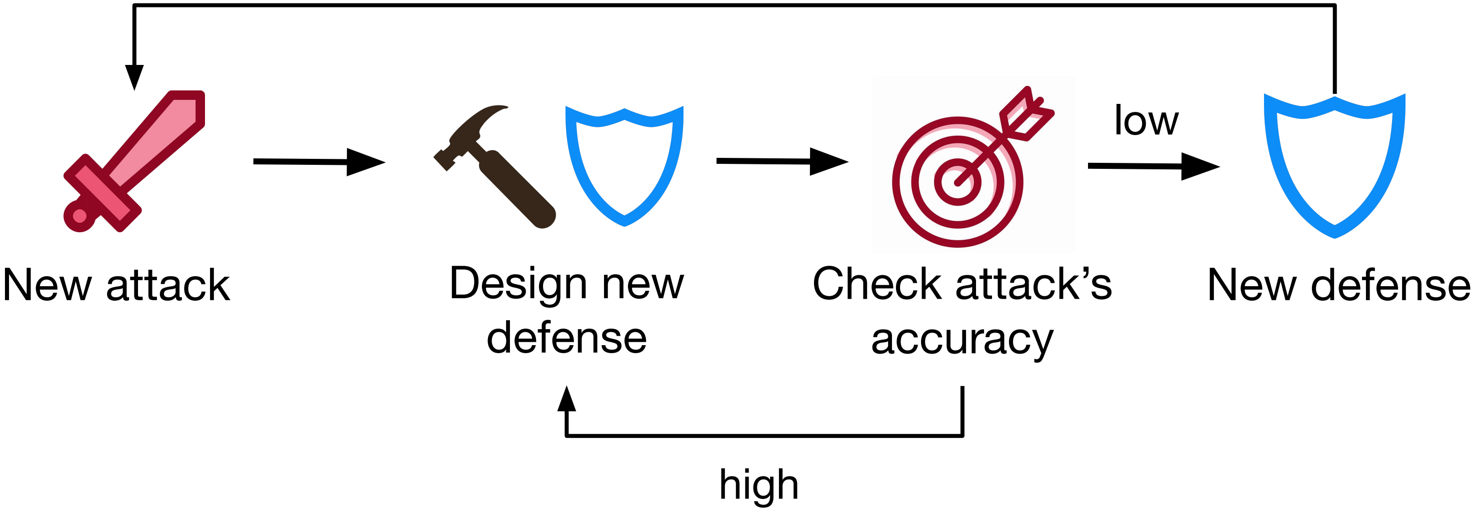


Defenses against Website Fingerprinting



How can we tell how good a defense is?

WF Defenses' Evaluation Lifecycle



Highly dependent on new attacks (or classifiers)

Attack-independent Defense Evaluation

Bayes Error Rate - BER

(**WFES**, Cherubin, PoPETs’17)

- Estimate **smallest achievable error**
- Uses error of 1-NN classifier as a proxy to estimate a lower bound for the error of ***any*** classifier on predefined features

Mutual Information - MI

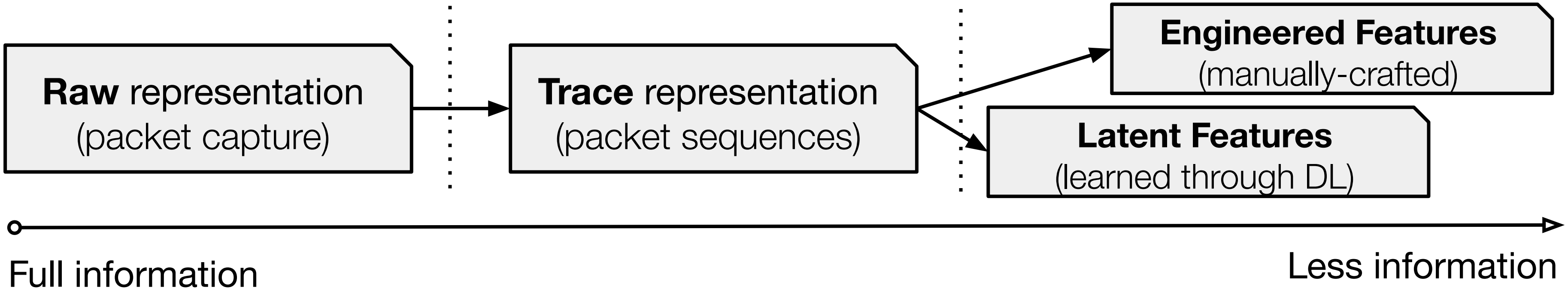
(**WeFDE**, Li et al., CCS’18)

- Estimate **information leakage**
- Uses adaptive KDE to model the probability density function of features
- Computes features’ mutual information

Both approaches focus on the analysis of manually-engineered features

Pitfalls of WF Defenses' Security Evaluation

Main issue: Mismatch of features used in attacks, defenses, and estimators



Features used in security estimation methods are less expressive and thus less informative

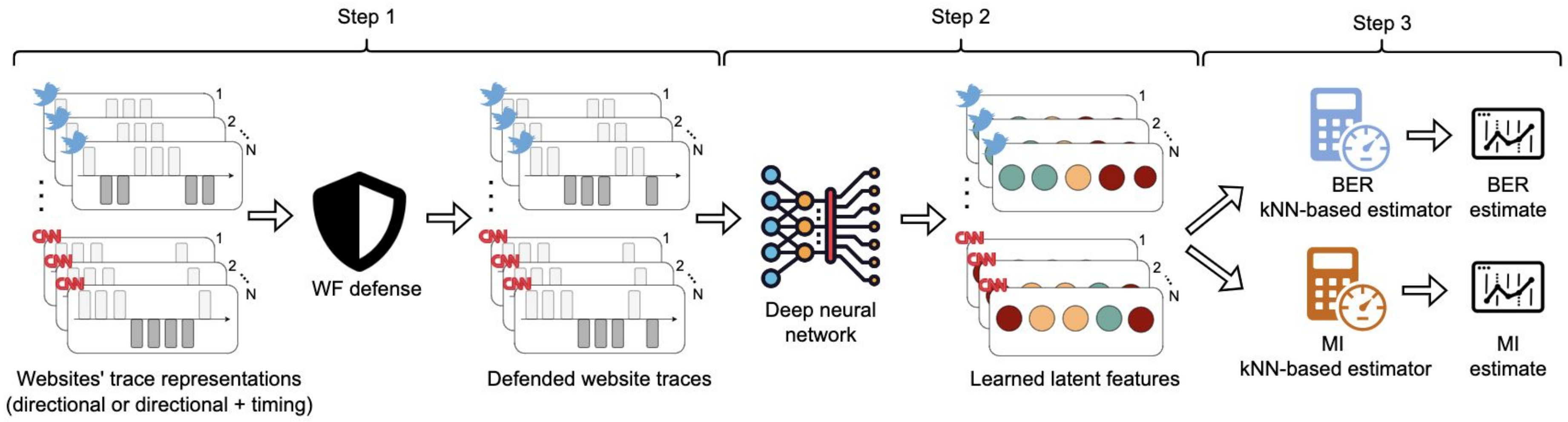
Main Contributions

- **DeepSE-WF**: a new security estimation framework that leverages learned latent feature spaces to jointly estimate the BER and MI of WF defenses

- Implementation and evaluation of DeepSE-WF
  - experiments conducted on defended Tor traffic

DeepSE-WF – Overview

Generate defended traces	Learn latent features	Estimate BER/MI
(via defense simulators)	(for different DL architectures)	(take ***min*** & ***max***)



Estimation Methodology – BER

Based on 1-NN
(Cover and Hart, '67)

$$\min_f \widehat{(R_{f(X)})}_{n,1} = \min_f \left(\frac{(R_{f(X)})_{n,1}}{1 + \sqrt{1 - \frac{C (R_{f(X)})_{n,1}}{C-1}}} \right)$$

where:

f : each of the learned feature representations

$(R_{f(X)})_{n,1}$: 1-NN accuracy using f

C : number of classes

Transformations can only increase the BER (Rimanic et al.'20)

DeepSE-WF keeps theoretical guarantees on **any** possible feature transformation

(take ***min*** over all possible f)

Estimation Methodology – MI

Based on k-NN
(Ross, '14)

$$\max_f \hat{I}(f(X);Y) = \max_f \left(\psi(N) - \langle\psi(N_x)\rangle + \psi(k) - \langle\psi(m_f)\rangle\right)$$

where:

ψ : digamma function

N : # of samples

N_x : # of samples/class averaged over all classes

k : hyperparameter (usually small – we use k = 5)

m : avg. # of samples in the radius defined by the k nearest samples of the same class for every data point

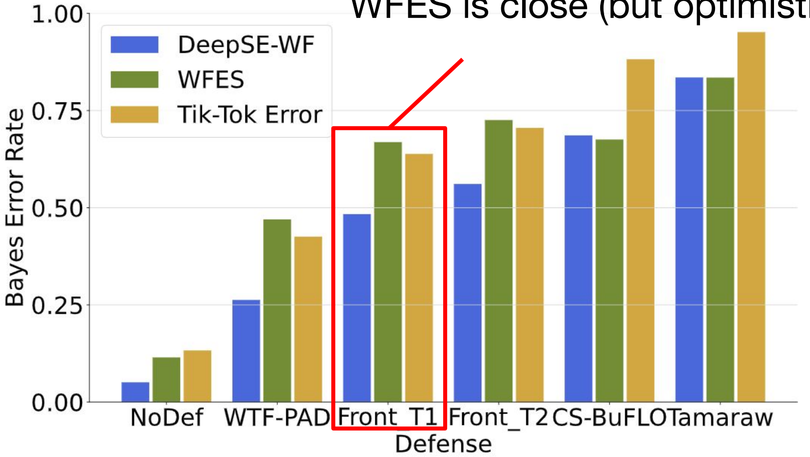
Transformations can only decrease MI
(proof in the paper)

DeepSE-WF keeps these guarantees on **any** possible feature transformation
(take ***max*** over all possible f)

BER – Comparison with WFES

(using the DF/Tik-Tok DNN architecture)

WFES is close (but optimistic)



AWF100x**90**

WFES is OOM for larger # of traces

For AWF100x**4500**:

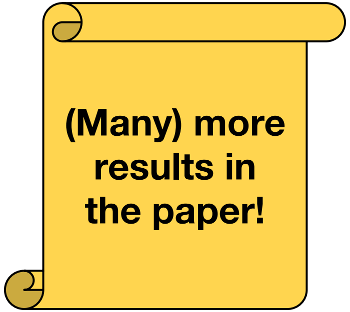
DeepSE-WF = 0.10

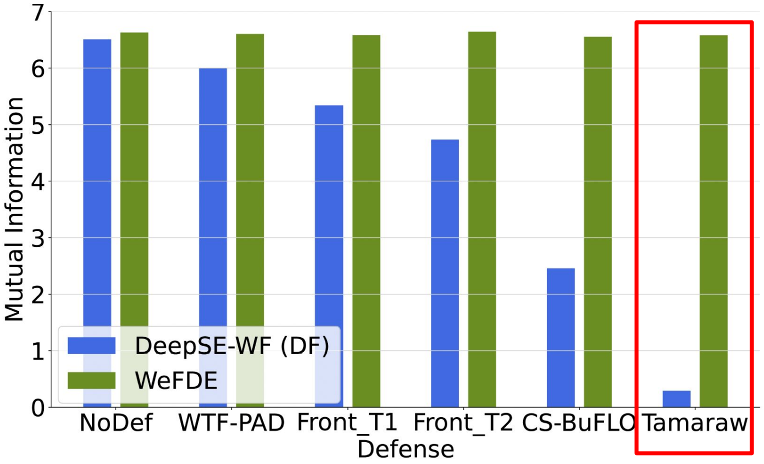
Tik-Tok Error = 0.16

DeepSE-WF produces tighter BER estimates (and scales to a larger number of samples)

MI – Comparison with WeFDE

(using the DF/Tik-Tok DNN architecture)





Tamaraw is a strong defense

AWF100x**500**

DeepSE-WF provides more reasonable results than WeFDE when estimating the leakage caused by all features

Takeaways

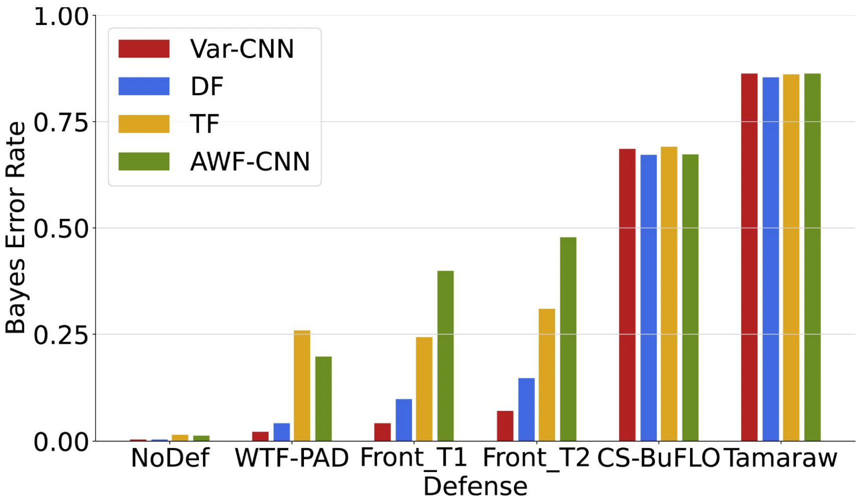
- Current security estimators do not provide tight bounds for the protection offered by existing WF defenses

- We proposed DeepSE-WF, a novel WF security estimator
  - Based on k-NN BER and MI estimators on latent feature spaces
  - Computes tighter security bounds, more efficiently

- However, DeepSE-WF estimates are not:
  - Attack-agnostic
  - Able to provide interpretable information about features
  - Geared towards the open-world setting

Thank you!

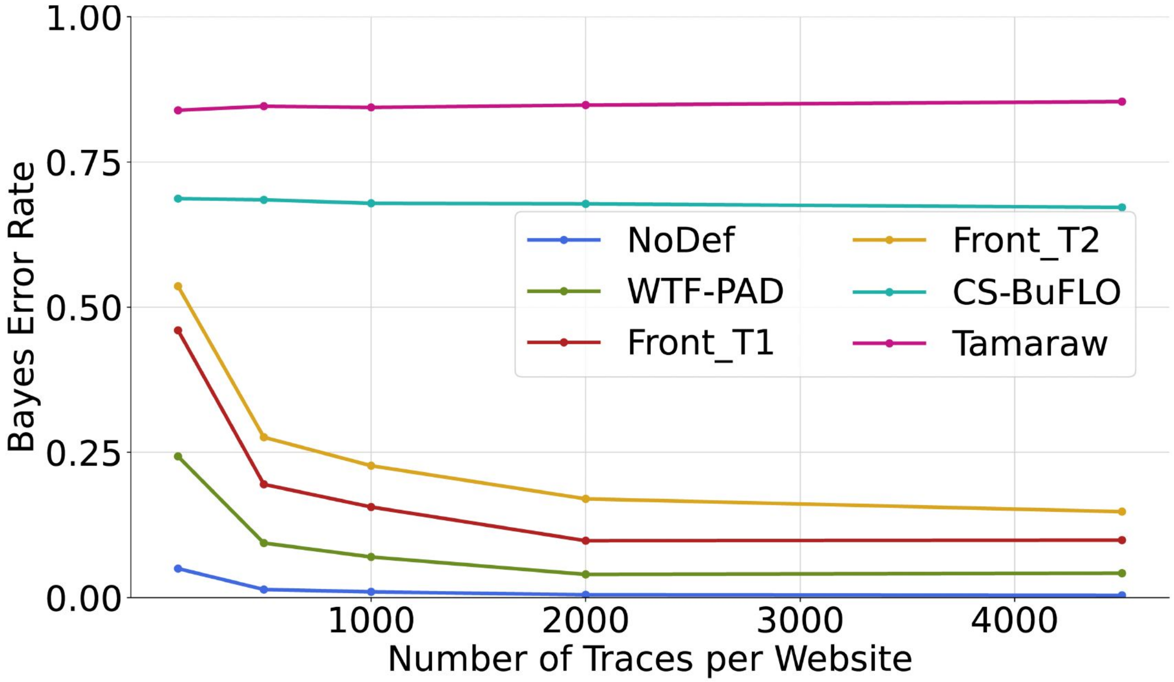
Impact of DNNs in the BER Estimates
(backup slide)



AWF100x**4500**

Different learned representations lead to different BER estimates (and tighter bounds for some defenses)

Convergence behavior
(backup slide)



Laboratory Testbed
(backup slide)

Assumptions:

- Closed-world setting – accesses to monitored websites equally likely
- Attacker perfectly separates website traces

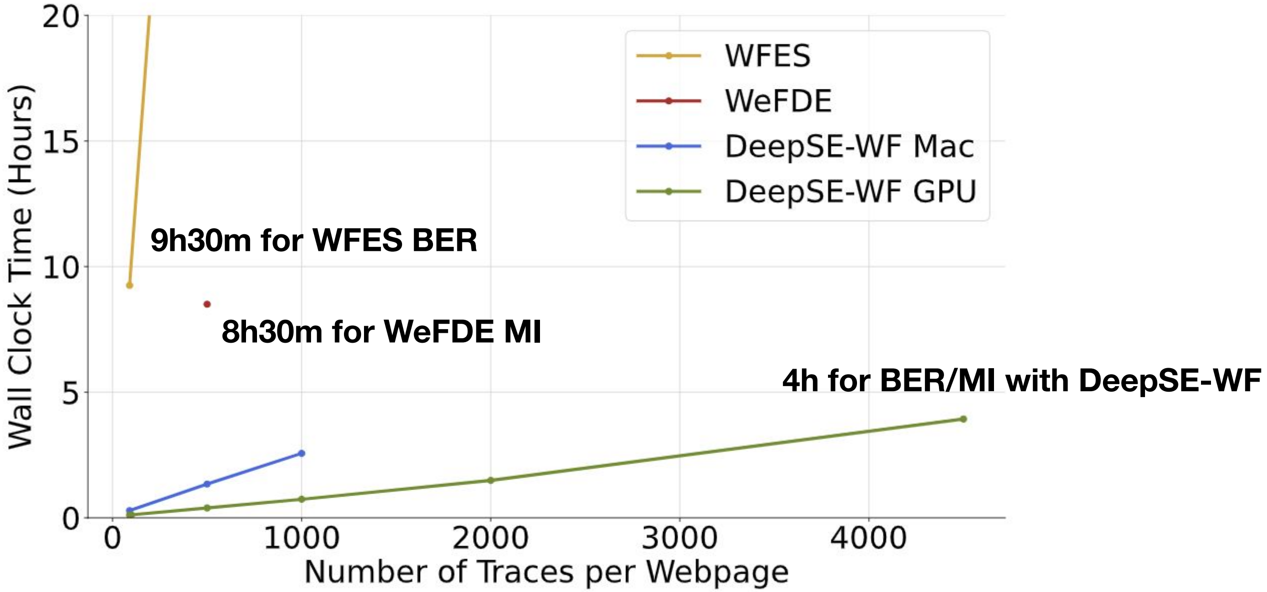
Datasets:

- Rimmer et al. ‘17 (AWF) – 100 websites * 4500 traces
- Gong and Wang ‘20 (DS19) – 100 websites * 100 traces

Testbed:

- MacBook Pro – M1 Pro CPU, 32GB of RAM
- Server – 40 Intel Xeon E5-262 CPU cores, NVIDIA TITAN X GPU, 256GB RAM

How Scalable is DeepSE-WF?
(backup slide)



DeepSE-WF is substantially more lightweight than WFES and WeFDE

DeepSE-WF BER vs. Attacks' Error
(backup slide)

Attacks & Estimators	NoDef	WTF-PAD	Front_T1	Front_T2	CS-BuFLO	Tamaraw
k-FP	04.1 ± 0.0	33.0 ± 0.0	41.2 ± 0.2	46.3 ± 0.1	80.9 ± 0.1	93.2 ± 0.1
AWF-CNN	03.5 ± 0.1	37.5 ± 0.9	51.0 ± 0.5	60.7 ± 0.4	84.6 ± 0.5	94.9 ± 0.1
DF	00.7 ± 0.0	07.4 ± 0.1	15.8 ± 0.1	22.9 ± 0.1	83.0 ± 0.1	94.8 ± 0.1
TF (L2 loss)	02.9 ± 0.4	45.4 ± 2.0	42.6 ± 2.1	52.2 ± 4.8	90.0 ± 0.1	97.3 ± 0.3
Var-CNN	00.7 ± 0.1	03.3 ± 0.1	06.4 ± 0.2	11.1 ± 1.3	83.0 ± 0.0	96.0 ± 2.0
Tik-Tok	01.0 ± 0.1	06.5 ± 0.2	15.9 ± 0.6	22.3 ± 0.2	82.8 ± 0.1	94.8 ± 0.1
DeepSE-WF (AWF-CNN)	01.3 ± 0.1	19.9 ± 0.2	39.9 ± 0.2	47.8 ± 0.5	67.3 ± 0.1	86.3 ± 1.1
DeepSE-WF (DF)	**00.4 ± 0.0**	04.2 ± 0.2	09.9 ± 0.2	14.8 ± 0.2	**67.2 ± 0.1**	**85.4 ± 1.1**
DeepSE-WF (TF - L2 loss)	01.5 ± 0.2	25.9 ± 1.3	24.3 ± 1.4	31.0 ± 3.7	69.1 ± 0.2	86.1 ± 1.2
DeepSE-WF (Var-CNN)	**00.4 ± 0.0**	**02.2 ± 0.1**	**04.2 ± 0.1**	**07.1 ± 0.2**	68.6 ± 0.5	86.3 ± 1.1

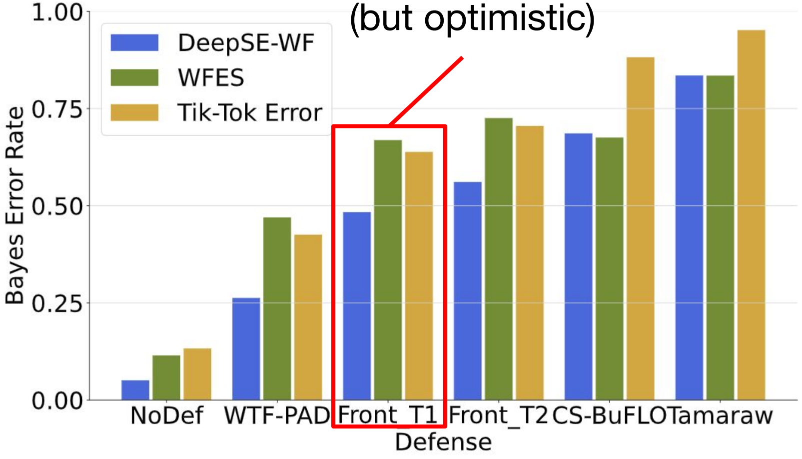
AWF100x4500

Comparison with WFES

(using the Tik-Tok DNN architecture)

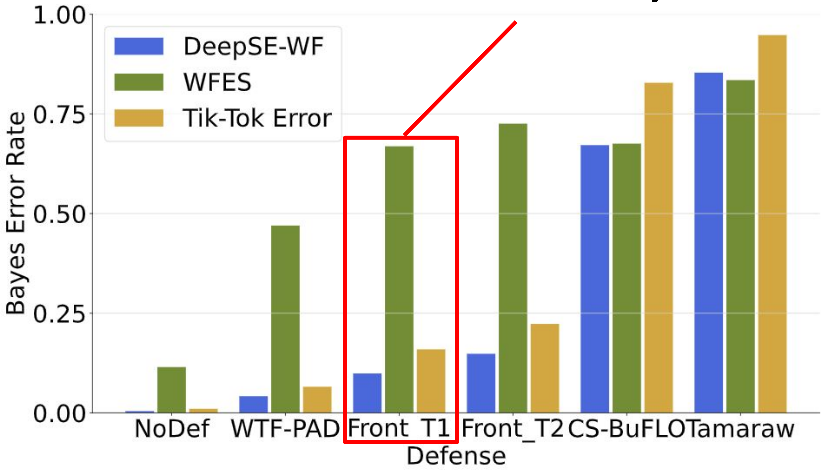
More results in the paper!

WFES is reasonably close (but optimistic)



AWF100x**90**

WFES is OOM for many traces



AWF100x**4500**

DeepSE-WF produces tighter BER estimates (and scales better for a larger number of samples)