Var-CNN and DynaFlow: Improved Attacks and Defenses for Website Fingerprinting

Sanjit Bhat (PRIMES) David Lu (PRIMES) Albert Kwon (MIT) Srini Devadas (MIT)
sanjit.bhat@gmail.com davidboxboro@gmail.com

April 17th, 2018
MIT

Motivation and Background

Anonymity matters

- Whistleblowers
- Governmental suppression of political opinion
- Censorship circumvention

http://blog.transparency.org/2016/06/20/new-whistleblower-protection-law-in-france-not-yet-fit-for-purpose/

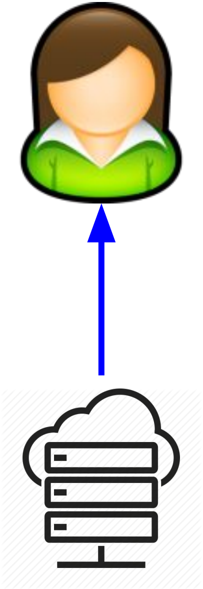
http://facecrooks.com/Internet-Safety-Privacy/To-be-anonymous-or-not-to-be-should-you-use-your-real-name-on-the-Internet.html/

http://www.dmnews.com/social-media/what-if-people-want-their-internet-anonymity-back/article/338654/

The internet provides limited anonymity

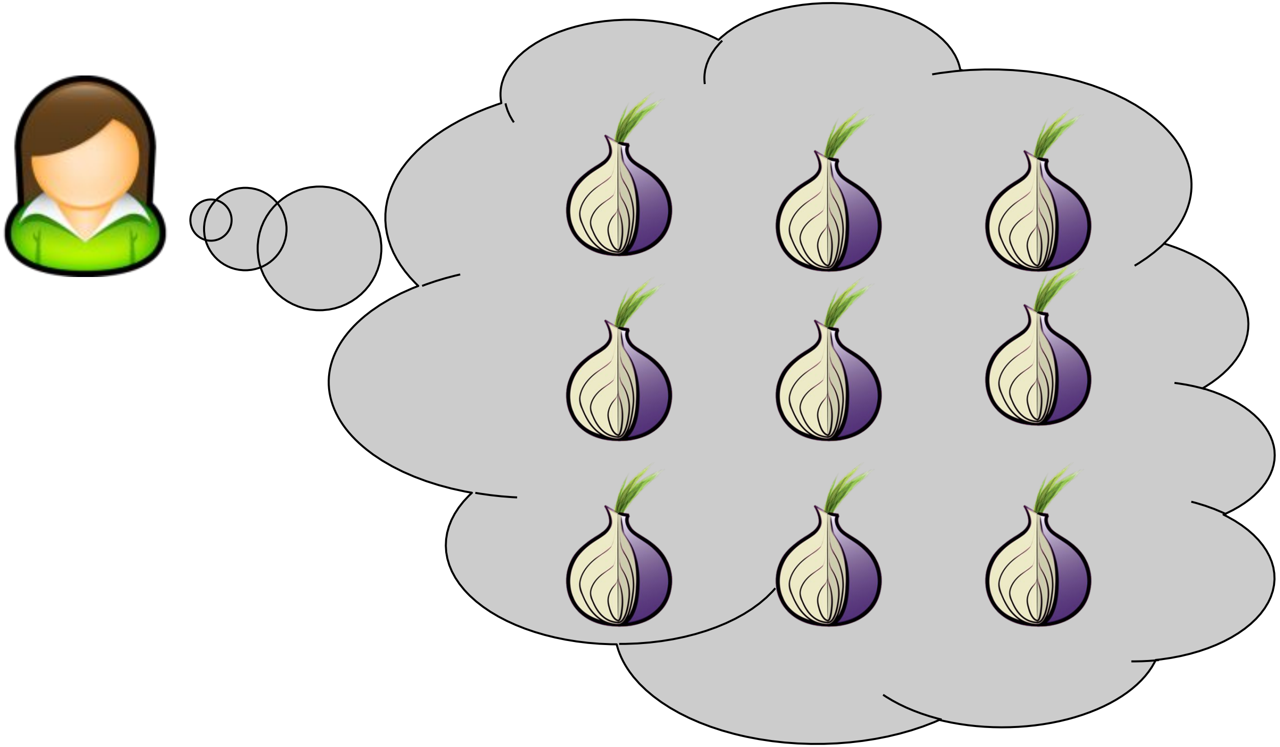
A supposed fix - Tor: The Onion Router

- Alice connects to the Tor network



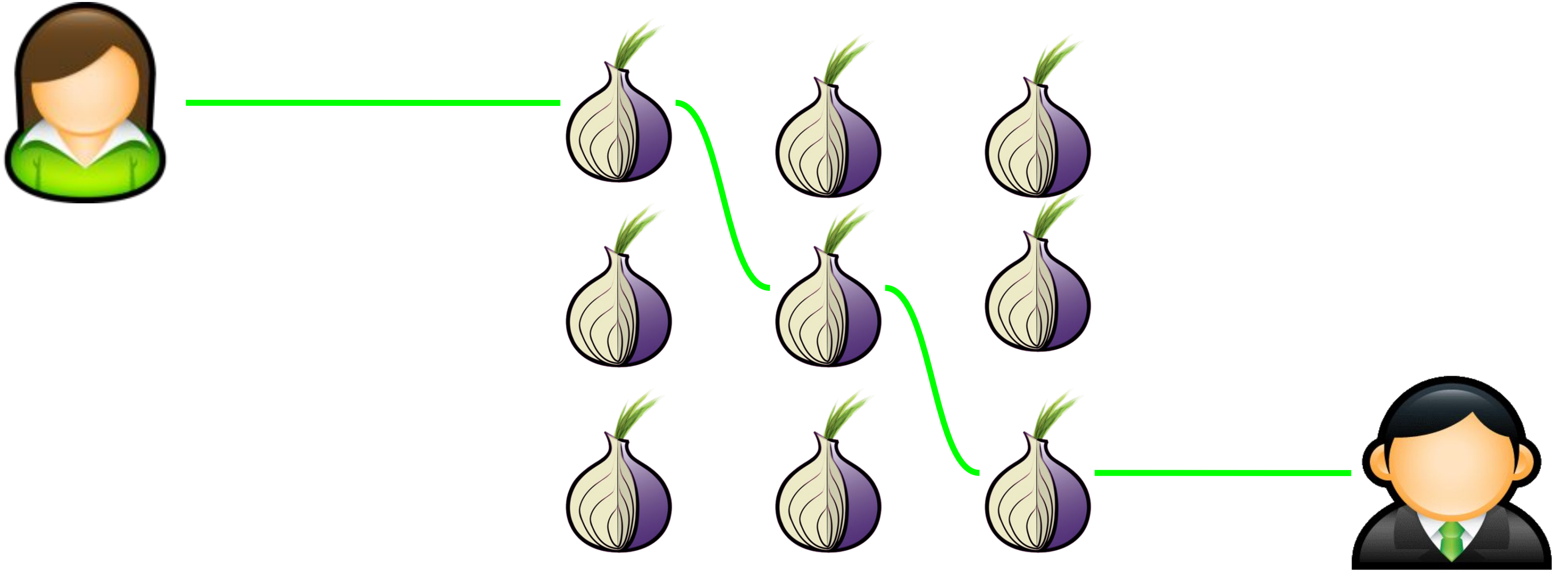
A supposed fix - Tor: The Onion Router

- Alice obtains a list of Tor nodes from the Tor network



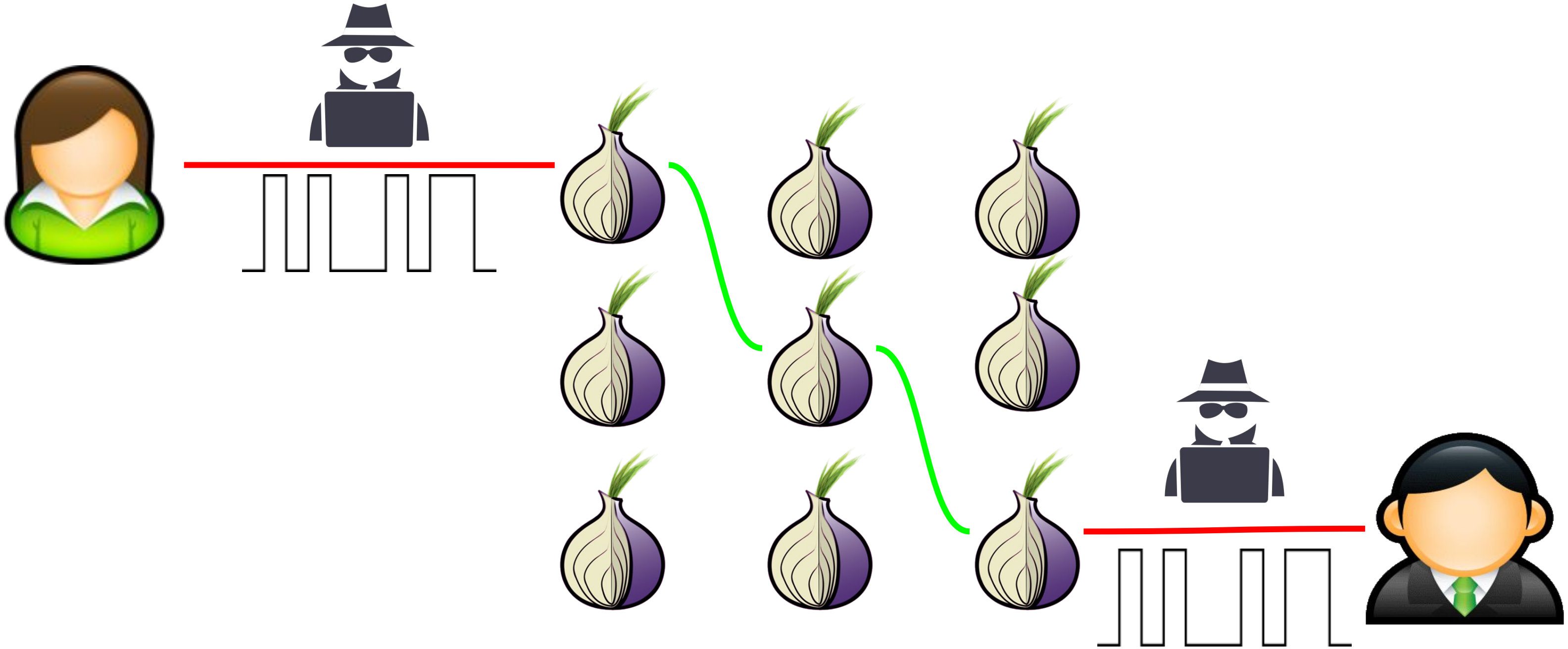
A supposed fix - Tor: The Onion Router

- Alice chooses 3 Tor nodes to make a connection to Bob
- No Tor nodes know the identities of both Bob and Alice



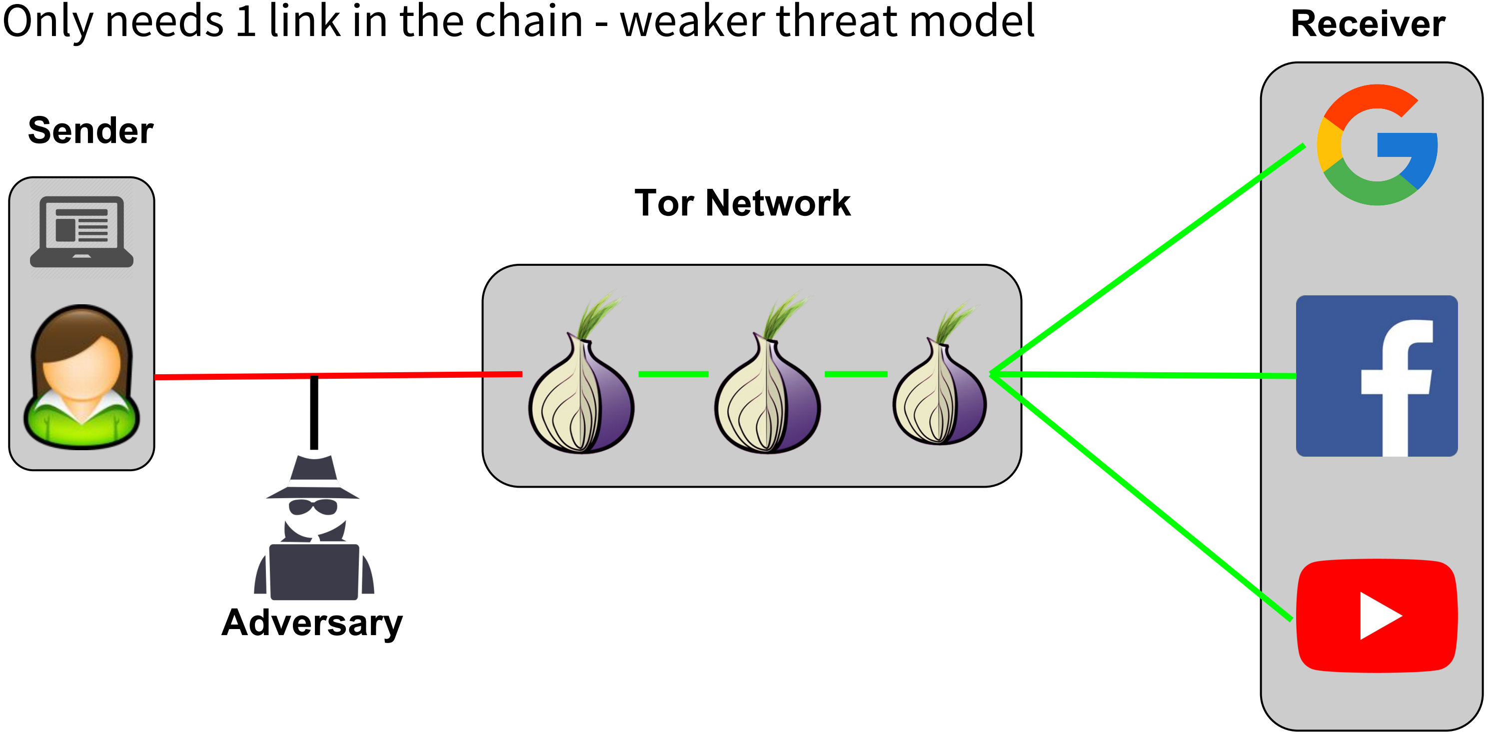
Traffic analysis attacks

- Adversary correlates Alice and Bob's traffic
- Only works when adversary intercepts both entry and exit points



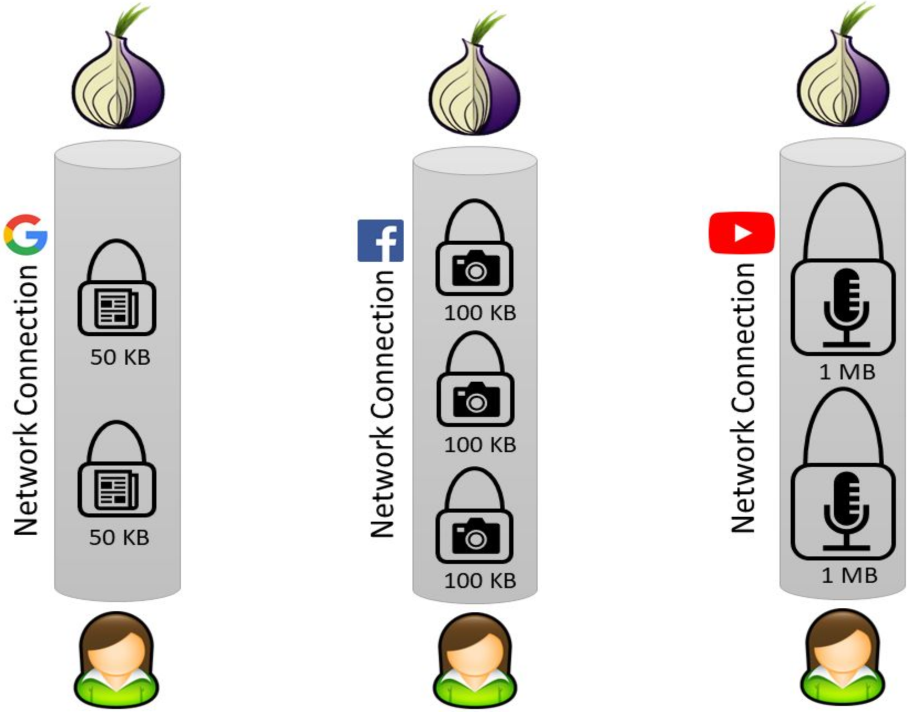
Website fingerprinting (WF) attacks

- Adversary collects database offline and uses it to fingerprint online
- Only needs 1 link in the chain - weaker threat model



Simplified WF attack scenario

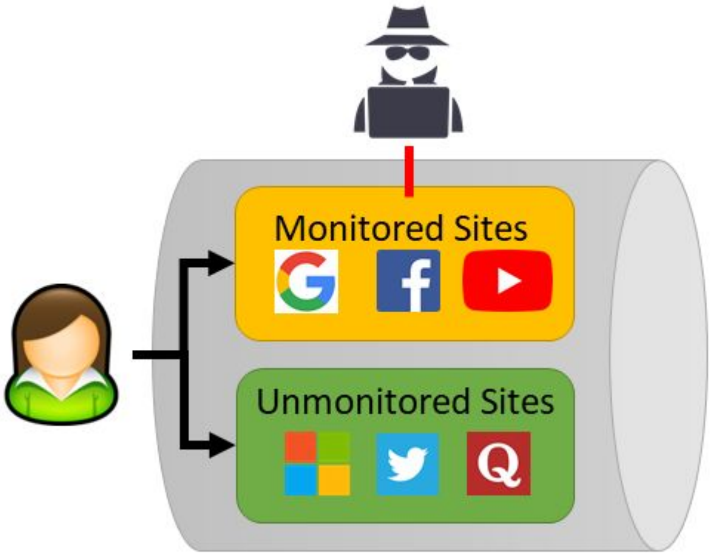
- Each website exhibits characteristic load behavior



Var-CNN: Automated feature extraction using variations on CNNs

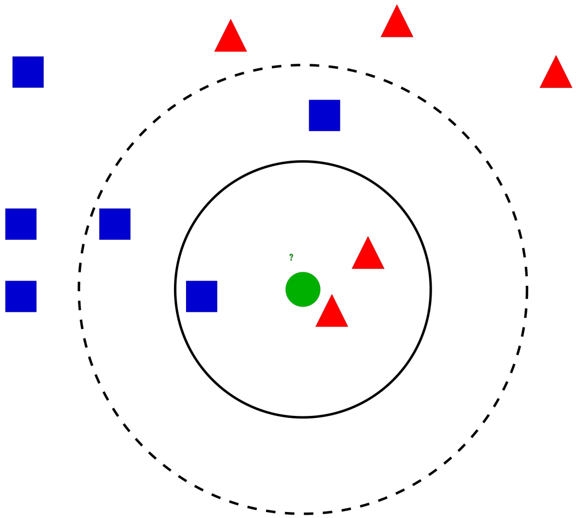
Terminology

- True Positive Rate (TPR) - Proportion of monitored sites correctly classified
- False Positive Rate (FPR) - Proportion of unmonitored sites incorrectly classified



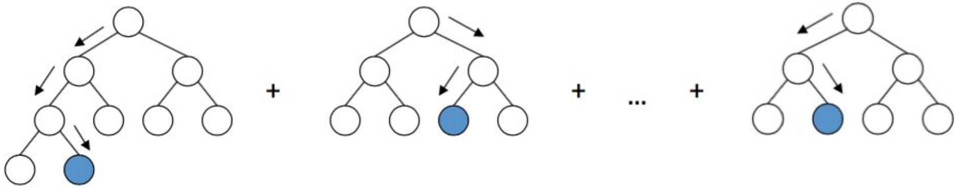
Open-World

Prior attacks



K-Nearest Neighbors (Wang et al. *k*-NN)

By Antti Ajanki AnAj - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=2170282



Random Forest (Hayes et al. *k*-FP)

“Brilliantly Wrong”, Alex Rogozhnikov

Prior attacks

Pros:

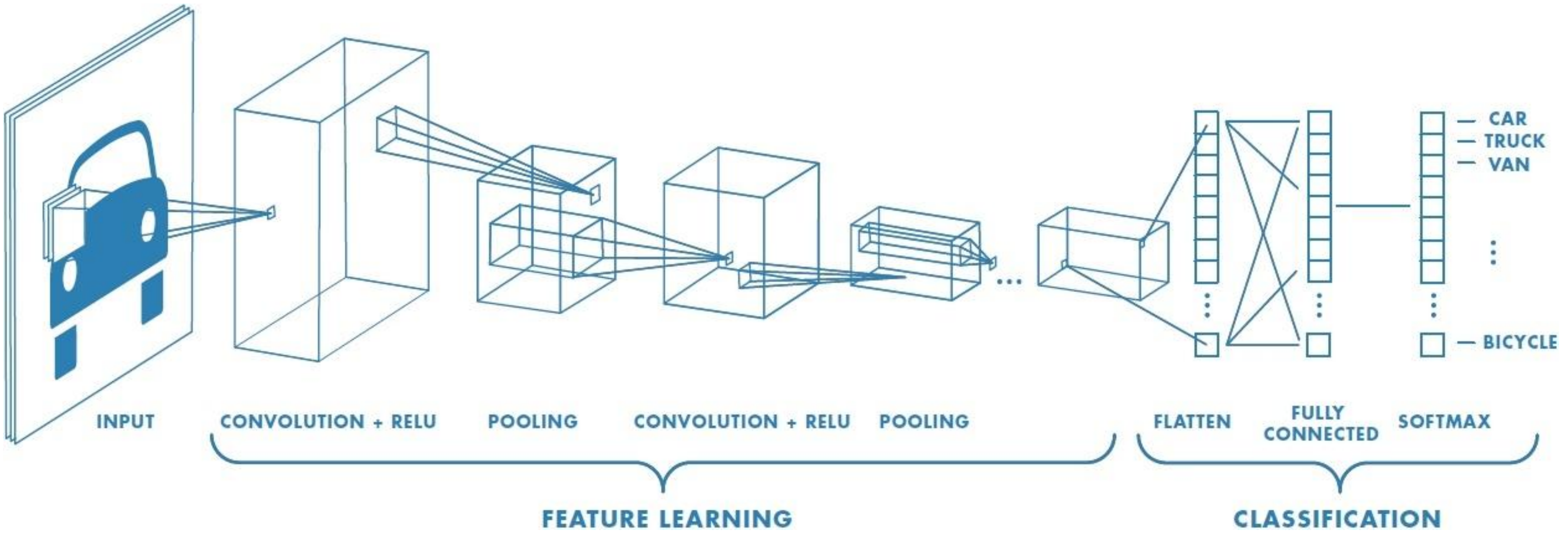
- Use well-studied machine learning techniques
- Quick to run
- Usually require small amounts of data

Cons:

- Pre-defined features as input
  - Number of packets
  - Packet orderings
  - Burst patterns, etc.
- Switching to other protocols requires feature re-design
- Features might not be optimal

Why deep learning?

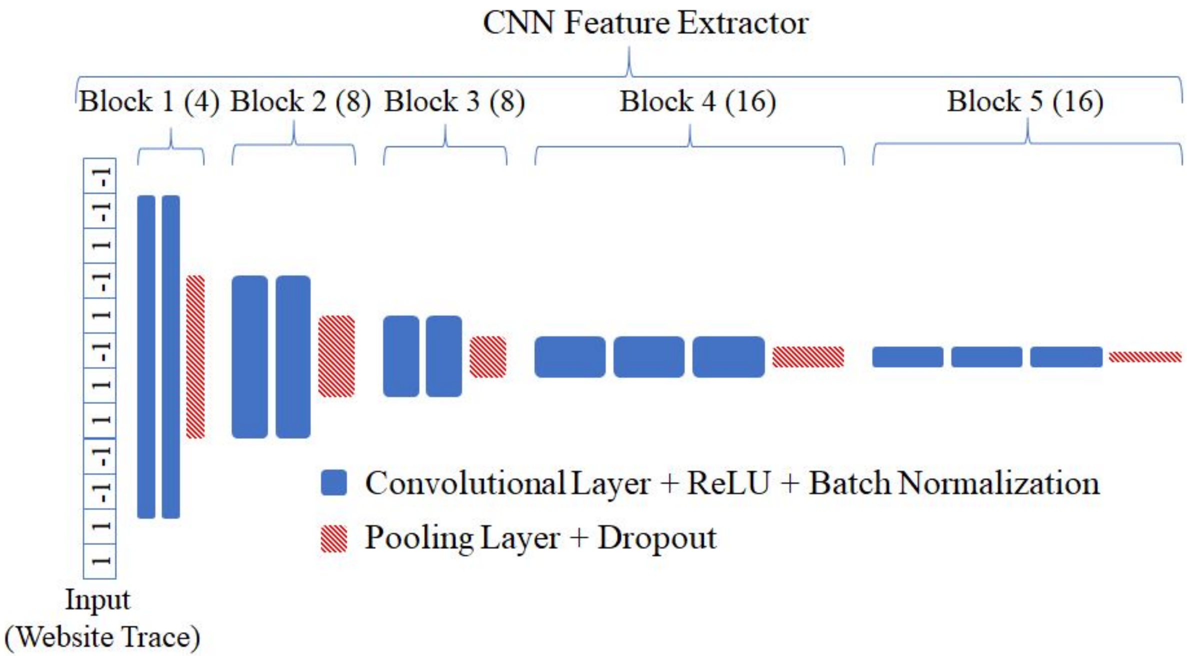
- Automated feature extraction
- Resistant to network protocol changes
- Discover more optimal features than humans could define



"Convolutional Neural Network", MathWorks

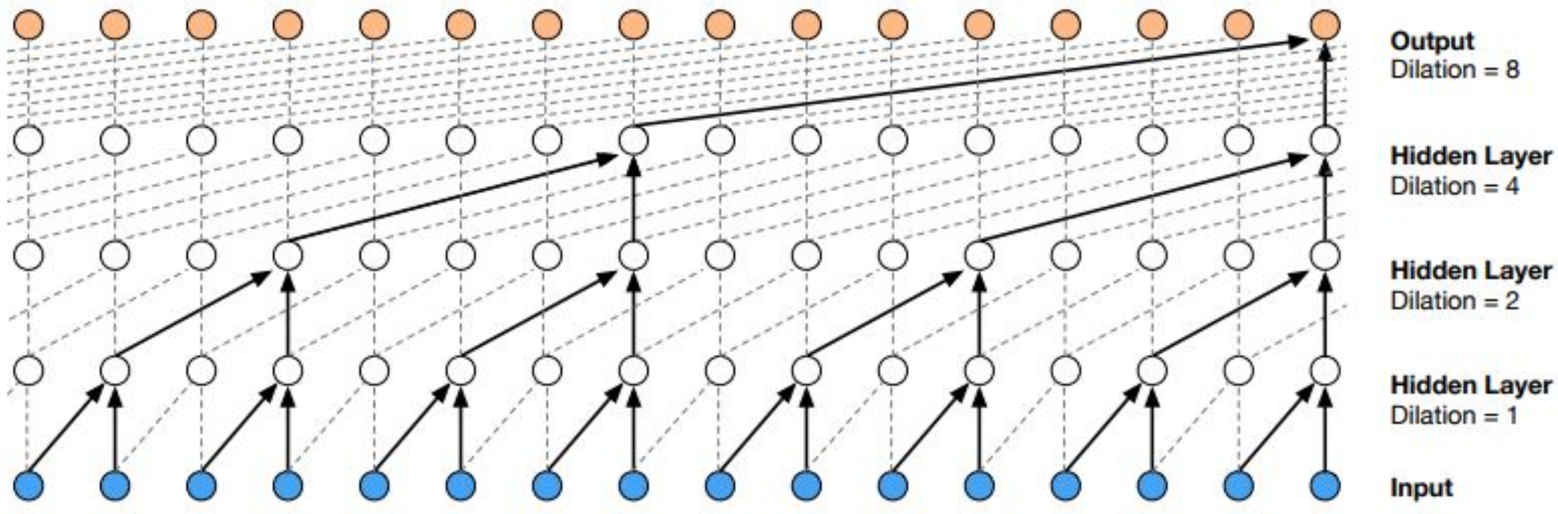
Var-CNN architecture

- VGG-16 Convolutional Neural Network (CNN) - ImageNet competition
- Multiple blocks composed of multiple layers for deeper feature extraction



Dilated convolutions

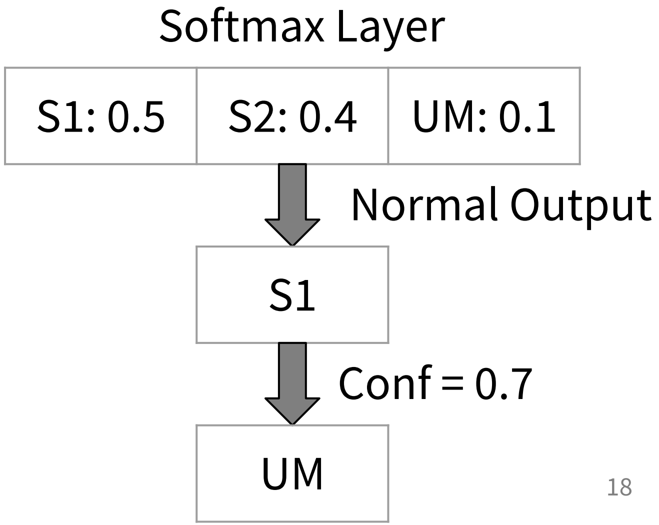
- Packet sequence inherently time-dependent
- Sacrifice fine-grain detail for broader field of view



A. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. arXiv, 2016.

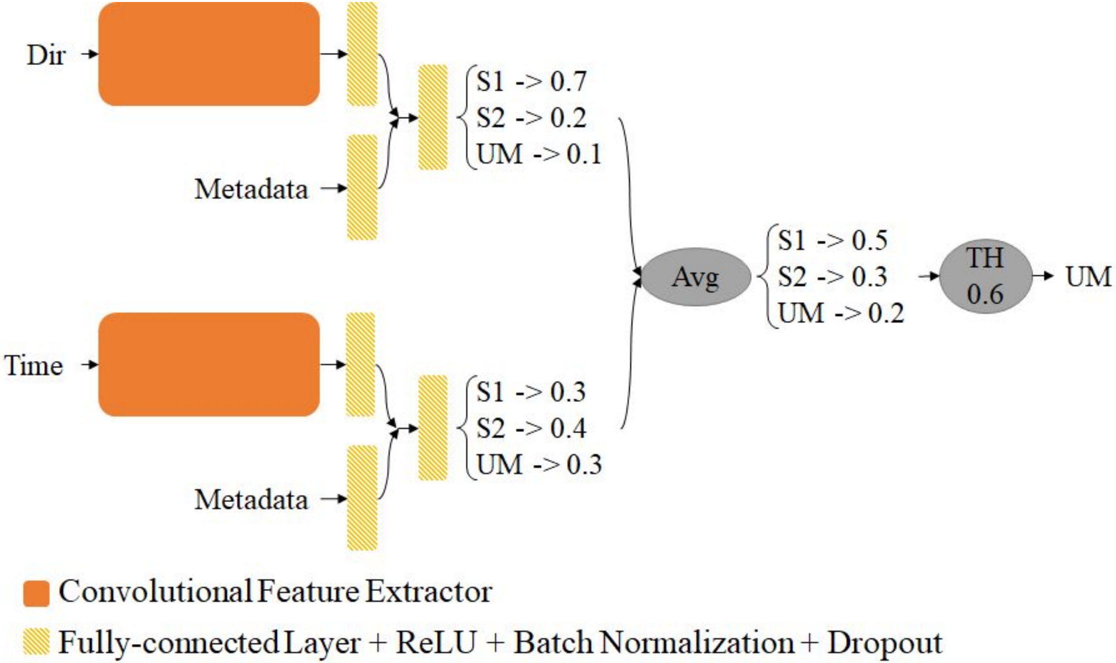
Other techniques

- Cumulative features
  - Total number of packets
  - Number of incoming and outgoing
  - Ratio of incoming to total and outgoing to total
  - Total transmission time
  - Average number of packets per second
- Confidence thresholds
  - Threshold for attacker certainty
  - Adjust TPR-FPR trade-off



Ensemble model

- Utilizing timing leakage should yield a stronger model
- No past pre-extracted timing features performed well



Var-CNN Results

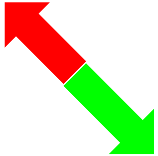
Experimental setup

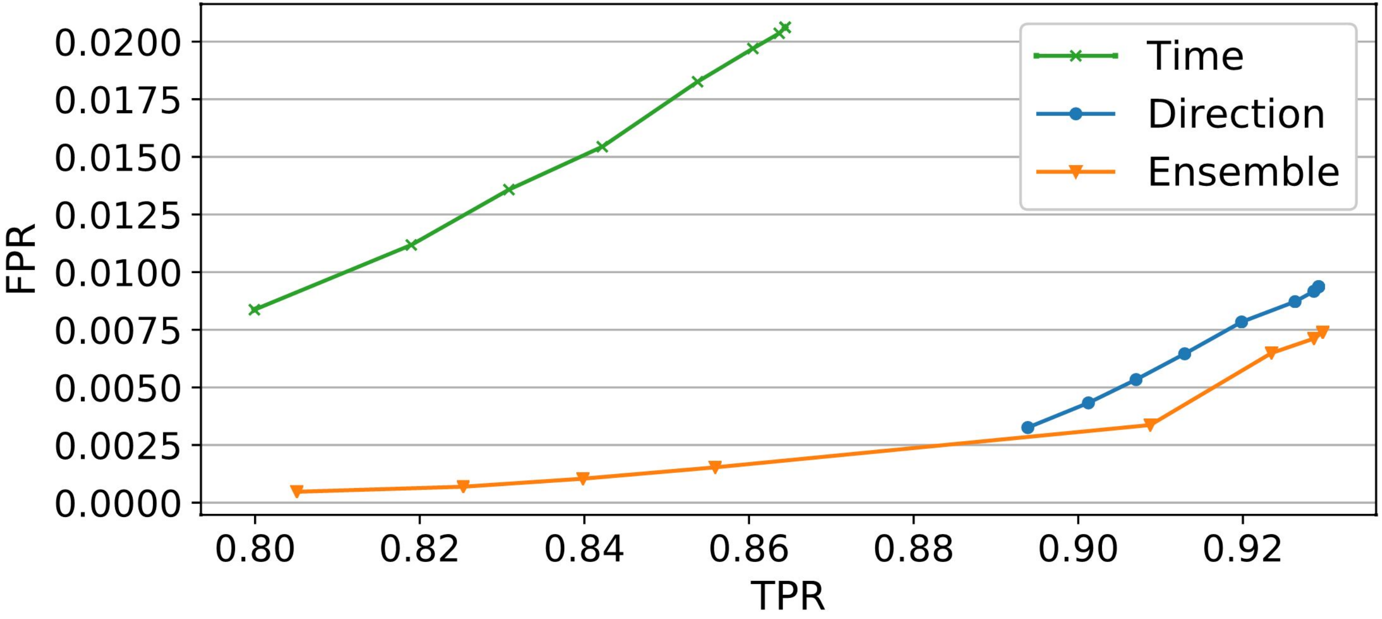
- Wang et al. *k*-NN data set
  - 100 monitored sites (90 instances) - Blocked pages from around the world
  - 9000 unmonitored sites - Alexa most popular pages
- <= training data used by competing attacks
- Re-randomize train/test sets and average results over 10 trials

Ensemble model and confidence threshold

- Alone, time model is worse than direction model
- However, their performance is additive

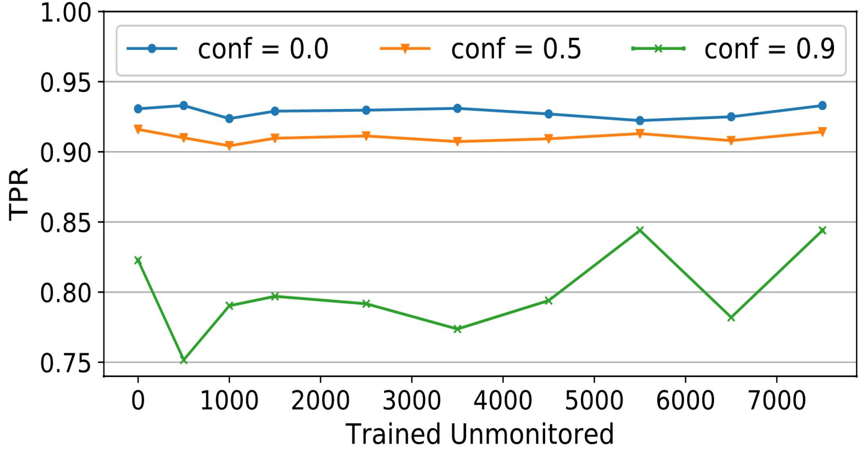
- TPR and FPR decrease as confidence threshold increases

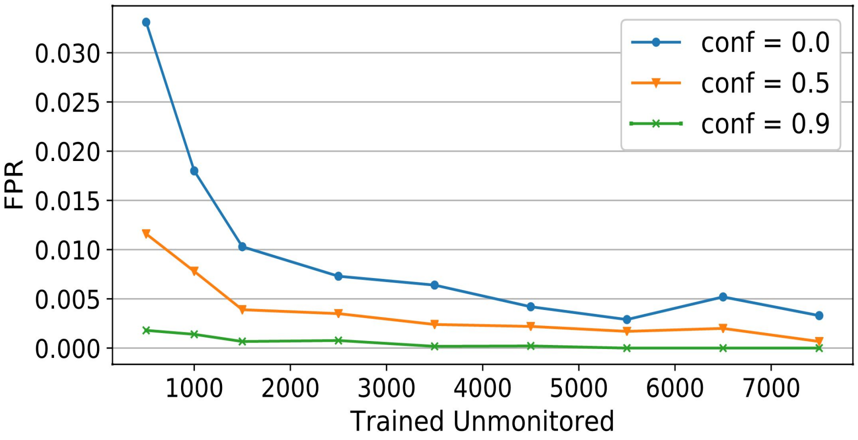




Scaling performance - FPR

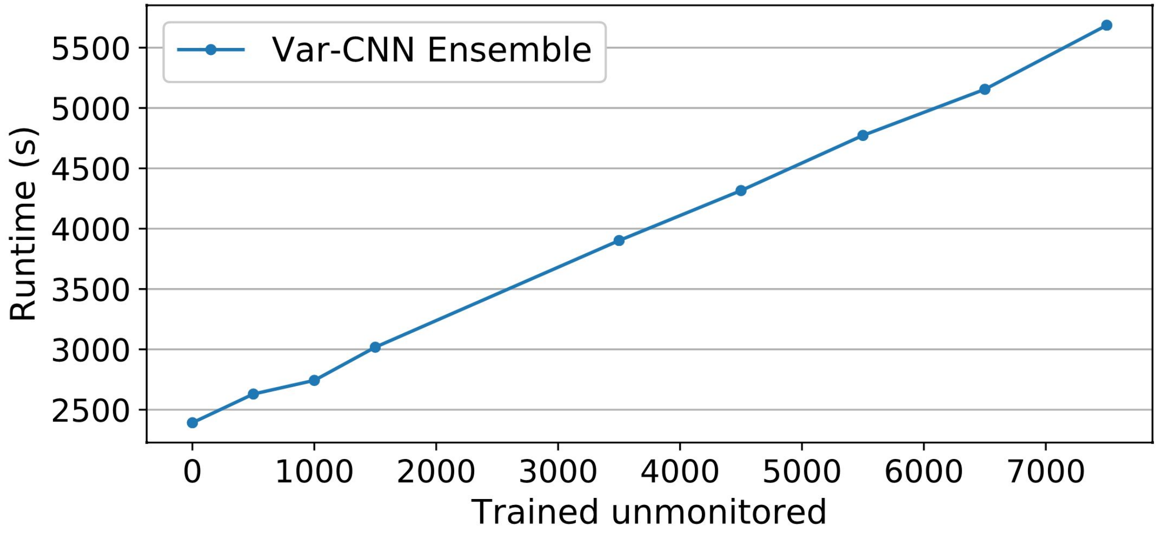
- FPR is incredibly important as open-world size increases
- Training on greater numbers of unmonitored sites retains TPR while reducing FPR
- Var-CNN scales better to larger open-worlds than prior-art attacks





Scaling performance - runtime

- Runtime scales linearly, better than prior models



Open-world performance

- 5% better TPR than SDAE
- Over a sixth the FPR of SDAE
- 3% better TPR than *k*-FP
- Nearly half the FPR of *k*-FP

All values are in %.

Attack	Auto. Feature Extraction	Accuracy (Closed)	TPR (Open)	FPR (Open)	Precision (Open)
k-NN [40]	✗	91 ± 3	85 ± 4	0.6 ± 0.4	—
k-FP [14]	✗	91 ± 1	88 ± 1	0.5 ± 0.1	—
SDAE [4]	✓	88	86	2	—
Var-CNN Ensemble (conf. threshold = 0.0)	✓	93.2 ± 0.5	93.0 ± 0.5	0.7 ± 0.1	98.6
Var-CNN Ensemble (conf. threshold = 0.5)	✓	93.2 ± 0.5	90.9 ± 0.5	0.3 ± 0.1	99.3

Background: WF Defenses

Limited defenses

Designed to counter existing attacks.

Examples:

- **LLaMA**: adds delays between requests
- **Decoy pages**: loads another page in parallel with the desired website
- **WTF-PAD**: adds dummy packets to hide unlikely time gaps

Main drawback: no provable guarantees.

Supersequence defenses

Overview:

1) Collect a database of traffic traces of many different websites
2) Group the traces into sets
3) Compute “supersequence” of each set
    a) Each sequence is a subsequence of the supersequence
4) Pad each trace to its supersequence

Examples: Supersequence, Glove, Walkie-Talkie

Drawbacks:

1) Requires a large and constantly-updated trace database
2) Protects only static content (no AJAX, Javascript)

Constant-flow defenses

Overview: Flood the network with a continuous stream of packets.

BuFLO:

- First constant-flow defense
- Leaked length of each trace

Tamaraw:

- Pads trace lengths
- High overheads: minimum of 100-200%
  - Time overheads
  - Bandwidth overheads

Advantages of DynaFlow

	Low Latency	Low Bandwidth Usage	Strong Security Guarantees	Protects Dynamic Content	No Database Required	Highly Tunable
DynaFlow	✓	✓	✓	✓	✓	✓
BuFLO [13]	✗	✗	✗	✓	✓	✗
Tamaraw [7]	✗	✗	✓	✓	✓	✗
Supersequence [40]	✗	✗	✓	✗	✗	✗
Walkie-Talkie [42]	✓	✓	✓	✗	✗	✓
Glove [29]	✗	✗	✓	✗	✗	✗
WTF-PAD [21]	✓	✓	✗	✓	✓	✗
Decoy Pages [32]	✓	✗	✗	✓	✓	✗
LLaMA [10]	✓	✓	✗	✗	✗	✗

DynaFlow: a new defense based on dynamically-adjusting flows

Overview of DynaFlow

Our goal: construct a defense with similar guarantees as Tamaraw but with significantly lowered overheads.

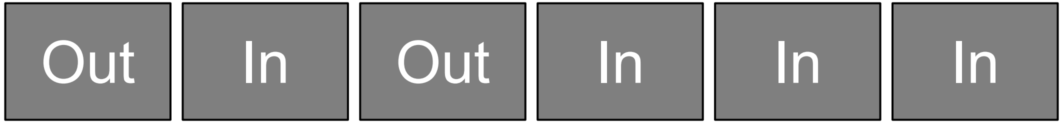
Three Components:

1) Burst-pattern morphing
2) Constant traffic flow with dynamically changing intervals
3) Padding the number of bursts

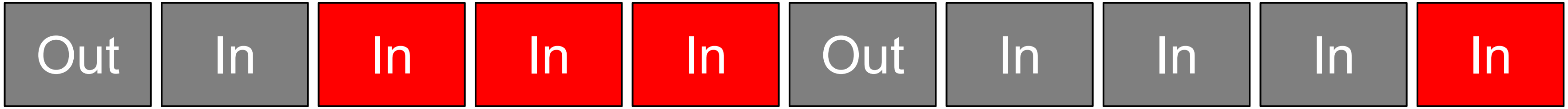
Burst-pattern morphing

- Traffic is morphed into fixed **bursts**: *o* outgoing packets followed by *i* incoming packets
- Setting $o = 1$ and $i = 4$ minimized overhead
- Dummy packets added to morph traffic

Before padding:



After padding (red packets are dummy packets):



Inter-packet timing

- Packets are sent every ***t*** seconds
- The value of ***t*** dynamically changes to fit the loading page
- There are three tunable parameters: ***a, b, T***
  - The value of ***t*** changes every ***b*** bursts
  - Up to ***a*** adjustments total
  - The value of ***t*** is chosen from the set $\boldsymbol{T = \{t_1, \dots , t_k\}}$

The number of bursts

- The number of bursts is padded to ***{[m], [m²], [m³], … }***
- Advantages of padding to a power of *m*
  - Significantly mitigate privacy loss
  - Incur reasonably-small overhead
- Example: when ***m = 2***, the bandwidth overhead is at most 100%

DynaFlow Results

Open-world eval. against existing attacks

DynaFlow against existing attacks. All values are in %.

	k-NN [40]		*k*-FP [14]		Var-CNN		TOH	BWOH
	TPR	FPR	TPR	FPR	TPR	FPR		
No defense:	84.5	2.5	86.3	1.6	89.1	0.7	0	0
Medium security:	15.4	20.6	5.0	1.6	10.8	3.0	23	59
High security:	5.9	69.0	4.4	40.1	0.6	0.9	28	112

The optimal attacker

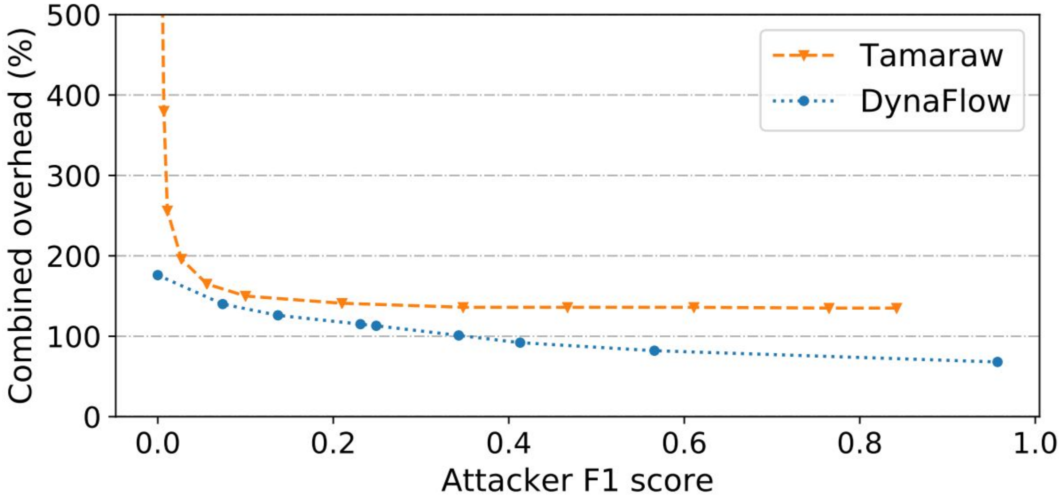
Overview:

- Knows the exact probability that a website ***w*** is visited, generating defended trace ***t***
- Uses this information to make the best guess for which website ***w*** is visited when he sees a trace ***t***
- We can use this information to calculate what the optimal attacker would guess.

Measuring accuracy:

- **F1-score** — harmonic mean of precision and recall (TPR)

Open-world eval. against optimal attacker



- 31% F1 score: 29% TPR, 11% FPR
  - DynaFlow: 101% overhead (29% TOH, 73% BWOH)
  - Tamaraw: 138% overhead (40% TOH, 98% BWOH)
- Gap increases for larger F1 scores

Conclusion

- Var-CNN uses novel variants of CNNs to do the following:
    - Scale well in large open-worlds, both in runtime and in FPR
    - Be highly tunable in terms of TPR-FPR trade-off
    - Outperform all prior attacks, all while using <= amount of training data
- DynaFlow overcomes challenges of prior WF defenses:
    - Lower overhead than prior work
    - Strong, provable privacy guarantees
    - Protects dynamic content
    - No database required
- Current status
    - Preprint on arXiv
    - Under review as conference paper in USENIX Security Symposium
    - All code and data sets publically available

Future work

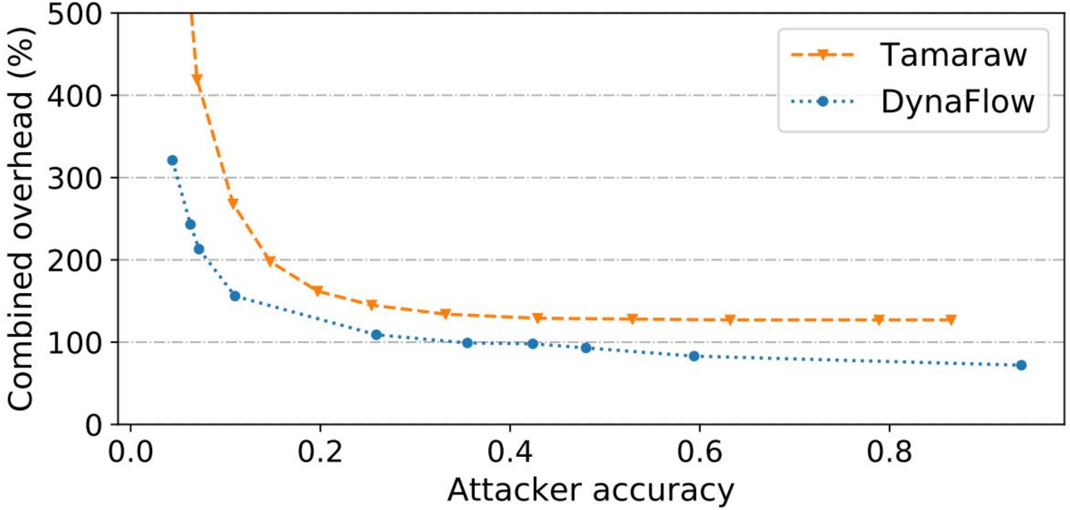
- More powerful deep learning models for Var-CNN
    - Computer vision architectures - DenseNet
    - Recurrent Neural Network architectures - LSTM with Synthetic Gradients
- Find a better way to determine optimal DynaFlow parameters
    - Currently, we sweep parameters one at a time
- Further reduce DynaFlow overheads
    - Total overhead sum can still exceed 100% for stronger configurations

Acknowledgements

Thank you to:

- Our parents
- Albert Kwon, for providing advice every step of the way
- Prof. Devadas, for giving feedback on the paper and running PRIMES CS
- The PRIMES program and Dr. Gerovitch, for providing research opportunities to high school students and sponsoring AWS bills and a GPU :-)

Closed-world (optimal attacker)



- 50% accuracy with 93% total overhead (Tamaraw: 127% overhead)
- 20% accuracy with 121% total overhead (Tamaraw: 162% overhead)
- 7% accuracy with 213% total overhead (Tamaraw: 419% overhead)

Closed-world (existing attacks)

DynaFlow against existing attacks. All values are in %.

Config.	Parameters	k-NN [40]	k-FP [14]	Var-CNN	TOH	BWOH
Baseline	N/A	88.0	94.3	95.2	0	0
1	$o = 1, i = 4, t_i = 0.012, b = 160, a = 6$ $m = 1.2, T = \{0.0012, 0.005\}$	17.5	45.0	46.8	31	53
2	$o = 1, i = 4, t_i = 0.012, b = 80, a = 1$ $m = 1.2, T = \{0.0015\}$	6.0	18.4	18.4	38	84