Defending Against Packet-Size Side-Channel Attacks in IoT Networks

Motivation

IoT devices are ubiquitous

- These devices communicate by sending packets over wireless networks

Interception of packets can reveal private information, <u>regardless of connection encryption</u>

- e.g., packets sent from a smartwatch may reveal activity status (walking, running, sleeping)

Side-channel attacks

Use of statistical properties of packets to reveal sensitive information

- Properties are observable despite content encryption

This paper’s focus: size-based side-channel attacks

- i.e., using packet size to infer information

Preliminaries

We observe a sequence of packets

- $X_1, X_2, \ldots, X_n$
- $X_i \in \mathcal{X}$, where $\mathcal{X}$ is the set of all possible packet sizes
  - i.e., we observe the size of each packet rather than content

Each packet follows a distribution $p_v(X_i = x)$

- p_v comes from a family of distribtions, $\mathcal{P} = \{p_v\}_{v=1}^{S}$
  - $\mathcal{P}$ can represent different IoT devices, or a single IoT device and its different states
- ***Packets are assumed to be i.i.d***

$\mathcal{P}$ and $P(X \sim p_v)$ is known by the adversary

Goal

$\mathcal{P}$ and $P(X \sim p_v)$ is known by the adversary

- Given a raw packet stream, an adversary can determine which IoT device generated the stream, as well as its state

We want perturb the packet size to make inferring $P(X \sim p_v)$ difficult

Packet Padding Obfuscator

Methodology:

- Define a conditional probability distribution $q(\hat{x}|x)$ that maps x to $\hat{x}$
- $\hat{x} \in \widehat{\mathcal{X}}$ i.e., the set of packet sizes and obfuscated packet sizes to not have to be the same

How do we select the values of q?

- Naïve: assign all packets the max size
  - Not a great solution, requires too much additional bandwidth overhead

$$
\begin{array}{cc}
 & \hat{X} \\
X & \begin{bmatrix} 0.1 & 0.7 & 0.2 \\ 0 & 0.6 & 0.4 \\ 0 & 0 & 1 \end{bmatrix}
\end{array}
$$

Example $q(\hat{X} = \hat{x}|X = x)$

Optimal $q(\hat{x}|x)$: Heuristics

Two heuristics are defined to address bandwidth overhead

$\widehat{W}_{avg}(q(\hat{x}|x))$: the expected size per packet

$\widehat{W}_{worst}\left(q(\hat{x}|x)\right)$: the maximum potential packet size

Optimal $q(\hat{x}|x)$: Guaranteeing Privacy

Local Differential Privacy

- More stringent than classical differential privacy
- Protects individual data rather than aggregate data

Probability of a packet following distribution p_v **after observing** the obfuscated size

$$\frac{\boxed{P(X \sim p_v \mid \hat{X} = \hat{x})}}{P(X \sim p_{v'} \mid \hat{X} = \hat{x})} \le \frac{\boxed{P(X \sim p_v)}}{P(X \sim p_{v'})} \cdot e^{\epsilon}$$

Probability of a packet following distribution p_v

We can use Bayes Theorem to rewrite this guarantee in terms of $q(\hat{x}|x)$

Optimal $q(\hat{x}|x)$: The Optimization Problem

Our goal is to find $q(\hat{x}|x)$ that minimizes $\widehat{W}_{avg}$ or $\widehat{W}_{worst}$ such that:

- $\boxed{0 \le q(\hat{x}_j|x_i) \le 1,\ \forall i, j}$
- $\boxed{\sum_{j=1}^{|\hat{\mathcal{X}}|} q(\hat{x}_j|x_i) = 1,\ \forall i}$

Ensure q is a proper probability distribution

- $\boxed{q(\hat{x}_j|x_i) = 0,\ \forall x_i > \hat{x}_j}$ Ensure q can not decrease the packet size

- $\boxed{\frac{\sum_{i=1}^{|\hat{\mathcal{X}}|} p_v(x_i) \cdot q(\hat{x}_j \mid x_i)}{\sum_{i=1}^{|\hat{\mathcal{X}}|} p_{v'}(x_i) \cdot q(\hat{x}_j \mid x_i)} \le e^{\epsilon},\ \forall j, v, v'}$ Ensure q meets the privacy guarantee

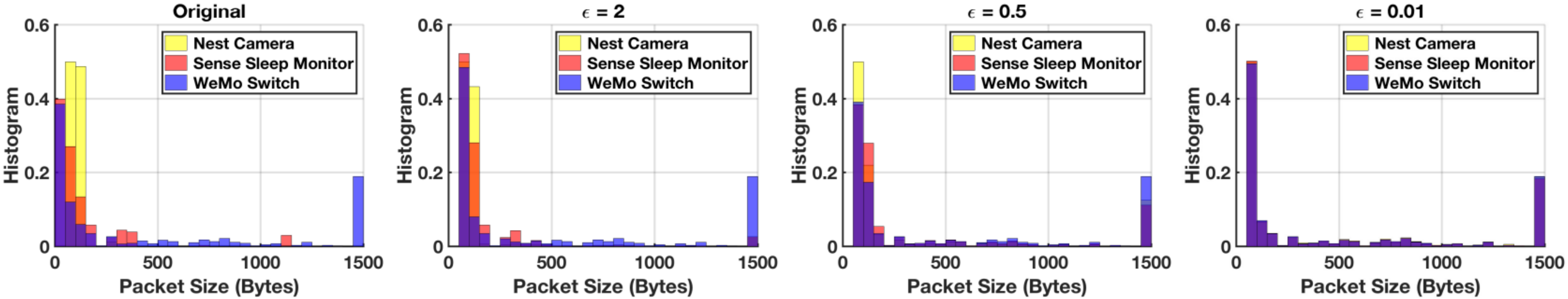
Privacy guarantee after applying Bayes Theorem

$$\begin{array}{cc} & \hat{X} \\ X & \begin{bmatrix} 0.1 & 0.7 & 0.2 \\ 0 & 0.6 & 0.4 \\ 0 & 0 & 1 \end{bmatrix} \end{array}$$

Example $q(\hat{X} = \hat{x}|X = x)$

If our chosen heuristic is $\widehat{W}_{avg}$ we can solve the problem using linear programming
If our chosen heuristic is $\widehat{W}_{worst}$ we can solve the optimization problem is convex

Results



(a) Original (b) $\epsilon = 2$ (c) $\epsilon = 1$ (d) $\epsilon = 0.01$

Privacy level

Experiment Setup

- Use real and synthetically generated packet data
  - Real: PMF of 3 IoT devices
  - Synthetic: generated from Zipf, Poisson (and mix of Zipf/Poisson)
  - i.e., there are 4 families of distributions, $\mathcal{P}_{1:4}$, each family has 3 “sub”-distributions

- Explore how different priors, $P(X \sim p_v)$, affect additional bandwidth of obfuscated packets. Assume 3 different priors:
  - **SAND**: assume highest probability on lowest bandwidth source
  - **UNIF**: assume uniform probability on all sources
  - **ROCK**: assume highest probability on highest bandwidth source

Results

β is the multiple of the non-private bandwidth required to ensure privacy

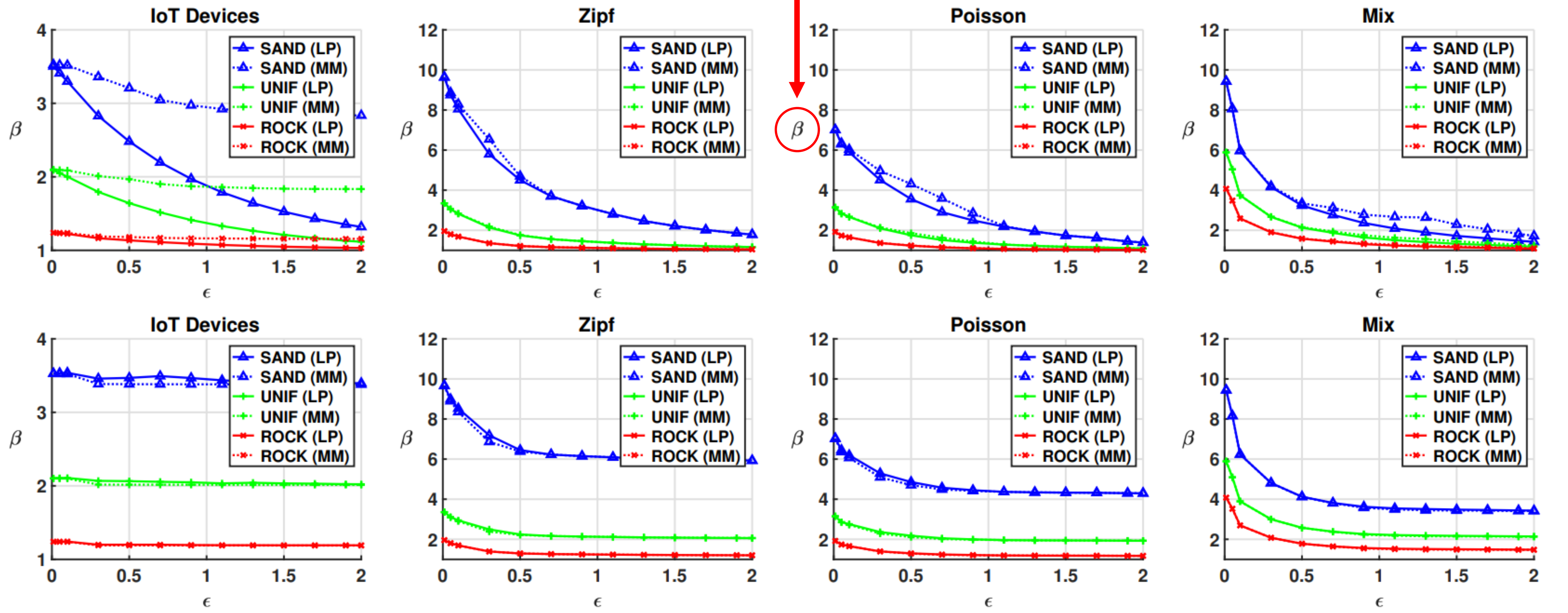


Fig. 1: On-Average and Worst-Case Privacy-Bandwidth Tradeoffs. Each of the 4 subfigures on the top row compares the *on-average* tradeoffs ϵ-$\hat{W}_{\text{avg}}(q^*_{\text{LP}})$ (solid line) and ϵ-$\hat{W}_{\text{avg}}(q^*_{\text{MM}})$ (dotted line), under SAND, UNIF, and ROCK priors for the corresponding family of PMFs. Each subfigure on the bottom row compares *worst-case* tradeoffs ϵ-$\hat{W}_{\text{worst}}(q^*_{\text{LP}})$ and ϵ-$\hat{W}_{\text{worst}}(q^*_{\text{MM}})$, accordingly.

Conclusion

- Using a heuristic and privacy constraint, we can create a provably private mechanism by solving the emerging optimization problem

- Notes:
  - Assumption that the data is i.i.d
    - Many sequences will not follow this
  - Assumption (in experiments) that the adversary will only observe a single packet at a time
    - If an adversary observed N packets, privacy leakage would be $N \cdot \epsilon$ (sequential composition)