Passive Encrypted IoT Device Fingerprinting with Persistent Homology

Joseph Collins, David Chapman
University of Maryland, Baltimore County

Dmitry Cousin, Michaela Iorga
National Institute of Standards and Technology

Abstract

- We present a novel persistent homology based method for the fingerprinting of Internet of Things (IoT) traffic
- IoT devices often present significant security risks to end users
- Inter-packet arrival time (IAT) is an especially useful feature as it is available even in encrypted traffic
- We show that applying persistent homology over IAT packet windows yields powerful discriminative features for device fingerprinting
- The clique complex construction and weighting function we present are efficient to compute and robust to shifts of the packet window

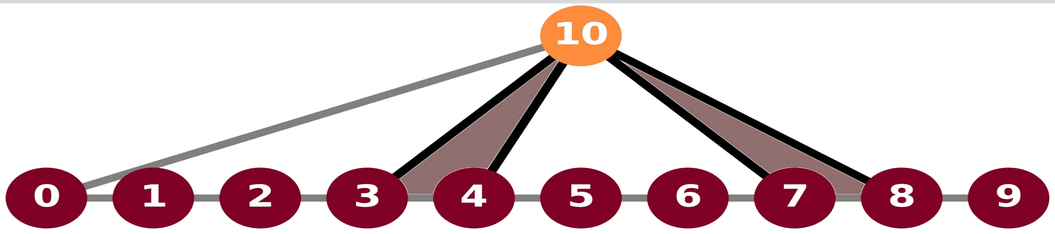
Methods

- Packets of given traffic flow are $(p_1,p_2,...,p_N)$
- IAT is $\Delta_T(p_q) = T(p_q) - T(p_{q-1})$
  - Where $T(p_q)$ is arrival time of p_q
- Take a k-length window of packet starting at packet p_i , $\omega_k(i) = (p_i ,p_{i+1},...,p_{i+1-1})$
- Create filtered simplicial complex with IAT
  - Sequential packet vertices connected at step 0
  - Connect packet vertices to “flow vertex” in order of ascending IAT
- Classify over H_1 persistent images using convolutional neural network

Results

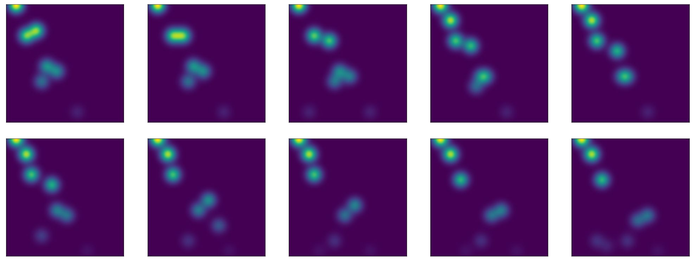
- Achieved competitive performance classifying encrypted traffic from 23 IoT devices (UNSW IoT dataset)
  - Split: 70/30 train/test
  - Accuracy: 95.34%
  - Recall: 95.27%
  - Precision: 95.46
- Comparison difficult due to lack of published details for similar systems
  - Data parsing and splitting changes the problem
  - Similarly constrained approaches achieve accuracy ~91%-96%
- This work confirms viability a windowed persistent homology approach

Example Filtration



An example filtered clique complex of the window $\omega_{10}(0)$ after the first 4 non-zero edges have been added. Packet vertices are in red and the flow vertex is in orange. Dark black edges are non-zero edges that have been added. Gray edges are weight 0 and added at the initial step. The shaded regions represent the 2-simplices.

Example Filtration



Persistence images of the windows corresponding to $\omega_{25}(i)$, $0 \leq i < 10$. Note that as the start of the window moves forward one packet at a time, the persistence image only changes slightly (left to right, top to bottom).