Stealthy Tracking of Autonomous Vehicles with Cache Side Channels

Mulong Luo, Andrew C. Myers, G. Edward Suh
Cornell University
29th USENIX Security Symposium 2020
Privacy in Cyber-Physical Systems (CPS)

- Autonomous vehicles and other CPS (e.g. UAV) are widely used
- Privacy in CPS involves physical information (e.g., location)
- CPS as computer systems, are vulnerable to cache side-channel attack

1. waymo.com
2. dji.com
Main Contributions

• We identified CPS as victims that can expose physical privacy information
  – Traditional victims of cache side-channel attack usually leaks non-physical information (e.g., crypto keys)

• We showed adaptive algorithms whose computation depends on inputs in CPS are vulnerable to cache side-channel attack

• We demonstrated a concrete example of cache side-channel attack on autonomous vehicles for route prediction
Threat Model

- Attack software shares computer with victim software on the vehicle under attack
- Attack software has no permission to access location service (e.g., GPS)
- Attack software uses cache side-channel attack to infer the route of the vehicle from a set of known routes
Autonomous Vehicle Systems

- Main components
  - Sensors
  - Actuators
  - Computer

- Major navigation tasks
  - Estimation/localization software
  - Control/decision software
Attack Outline

- Adaptive Monte-Carlo Localization (AMCL): memory access pattern is affected by the physical sensor inputs
- Cache side-channel attack is used to extract the memory access patterns
- Machine learning (ML) models are used to infer the number of AMCL samples and the route of the vehicle
Adaptive Monte-Carlo Localization (AMCL)

- Input: sensor image, and a map
- Output: location on the map
- Process:
  - Randomly sample locations on the map, calculate the similarity with the sensor image
  - Use k-means to find cluster center
- Adaptivity: the number of samples varies adaptively depending on uncertainty\(^1\)
  - High curvature (uncertainty) \(\rightarrow\) more AMCL samples \(\rightarrow\) more memory access

---

• Adaptive Monte-Carlo Localization (AMCL): memory access pattern is affected by the physical sensor inputs
• Cache side-channel attack is used to extract the memory access patterns
• Machine learning (ML) models are used to infer the number of AMCL samples and the route of the vehicle
Cache Side-Channel Attack

- Sharing of cache
- Memory accesses patterns from AMCL are left in the cache
- Attack program access the cache and infer AMCL accesses through timing
  - Prime+probe
Cache Prime+Probe

- For each prime+probe trial
  - Probe each cache set and record the probing time
  - High probing time (dark cell) $\rightarrow$ cache set accessed by a victim
  - #dark cells $\rightarrow$ #AMCL samples
- ML model to predict number of samples
Attack Outline

• Adaptive Monte-Carlo Localization (AMCL): memory access pattern is affected by the physical sensor inputs
• Cache side-channel attack is used to extract the memory access patterns
• Machine learning (ML) models are used to infer the number of AMCL samples and the route of the vehicle
ML Model: #AMCL Samples Predictor

- Formulated as a binary classification problem
  - Input: a window of cache probing time matrix
  - Output: two classes \{High, Low\} representing the number of samples
  - RUSBoost$^1$ binary predictor is used

Attack Outline

- Adaptive Monte-Carlo Localization (AMCL): memory access pattern is affected by the physical sensor inputs
- Cache side-channel attack is used to extract the memory access patterns
- Machine learning (ML) models are used to infer the number of AMCL samples and the route of the vehicle
ML Model: Route Predictor

- Formulated as a classification problem
  - Input: a series of \{High, Low\} which indicates the number of samples
  - Output: label of the route
  - Decision tree/forest is used
Evaluation: Datasets

• Two datasets used
  – Gazebo/ROS\(^1\)
    • Simulated robot in a maze
  – Oxford\(^2\)
    • Data collected on a car driving around Oxford, UK

• Processors for evaluation
  – Intel i5-3317u, and Intel Xeon E5-1270 (recommended by Apollo\(^3\))

---

2. https://robotcar-dataset.robots.ox.ac.uk/
3. https://apollo.auto
Evaluation: Route Prediction - Gazebo

- 7 randomly-selected routes with similar lengths
- Perform route prediction using L1D or LLC probing data
  - Accuracy: L1D: 81.4%, LLC 75%

Maze and 7 routes
L1D prediction results
LLC prediction results
Evaluation: Route Prediction - Oxford

- 7 routes with similar lengths from the Oxford dataset
- Perform route prediction using L1D or LLC probing data
  - Accuracy: L1D: 74.6%, LLC 73.0%
More In the Paper…

• Evaluation: Host-client testbed architecture (Section 5.1.1)
• Location prediction (Section 5.3.2)
• L1D & LLC comparison (Section 5.3.3)
• Discussion (Section 6)
  – Processor architecture (Section 6.1)
  – Generality of the vulnerability (Section 6.2)
  – Limitations of the attack model (Section 6.3)
  – Difficult-to-predict routes (Section 6.4)
Conclusion

• We identified CPS as victim that can expose physical privacy information

• We showed adaptive algorithms in CPS are vulnerable to cache side-channel attack

• We demonstrated a concrete example of cache side-channel attack on autonomous vehicles for route prediction

• Cache side-channel protection necessary for strong CPS physical privacy
Thanks for listening!

ml2558@cornell.edu