ScheduLeak: A Reconnaissance Attack Against Real-Time Embedded Systems

ILLINOIS
SyNeRCys Research Group

Chien-Ying Chen, Sibin Mohan, Negar Kiyavash, Rakesh B. Bobbaz and Rodolfo Pellizzoni

OVERVIEW

- Security in real-time systems (RTS) is overlooked
  - RTS is highly predictable due to its design nature (determinism)
  - RTS is used to control critical systems (e.g. power plants, avionic)
  - Studying possible attacks is crucial to understanding security in RTS
- Reconnaissance attacks
  - A stepping stone to more complex and powerful attacks
  - Stays stealthy while learning system's information
- ScheduLeak Algorithms
  - A set of novel algorithms to reconstruct task schedule information
  - Exploits scheduler-based side-channels
  - Works with periodic and mixed (periodic + sporadic) system model
  - Achieves 97% of inference success rate

SYSTEM AND ATTACKER MODEL

- Fixed-priority real-time systems (RTS)
  - Attacker's task (observer task) *periodic* or *sporadic*
  - Victim task *periodic*
  - Other tasks *periodic* or *sporadic*
- Requirements
  - The attacker knows the victim task's period
  - The observer task has lower priority than the victim task
- Attack Goals
  - Predict the victim task's future arrival points in time.

SCHEDULEAK ALGORITHMS OVERVIEW

- Observe and reconstruct
  - Utilizes a system timer to collect time information
  - Reconstructs the observer task's execution intervals
  - The observation time can be varied depending on the desired attack precision
- Analyze and extract
  - Organizes the reconstructed execution intervals in a schedule ladder diagram
  - Identifies the correlation between the observer and the victim task
- Infer and predict
  - Infers the victim task's arrival window from the schedule ladder diagram
  - Infers the victim task's initial offset
  - Predicts the victim task's future schedule (future arrival instances)

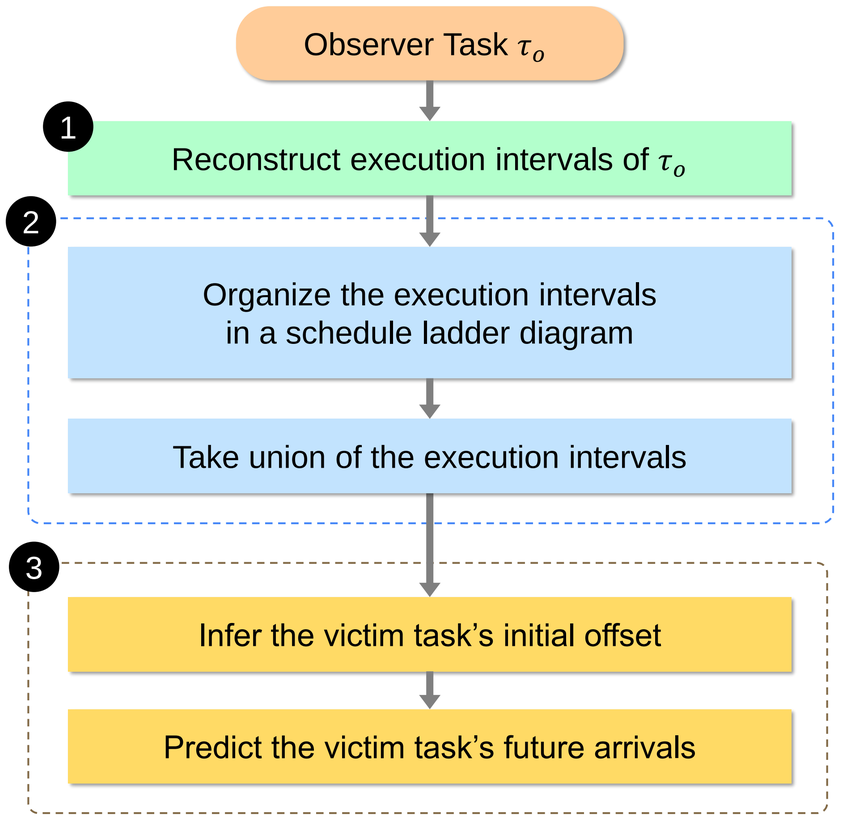
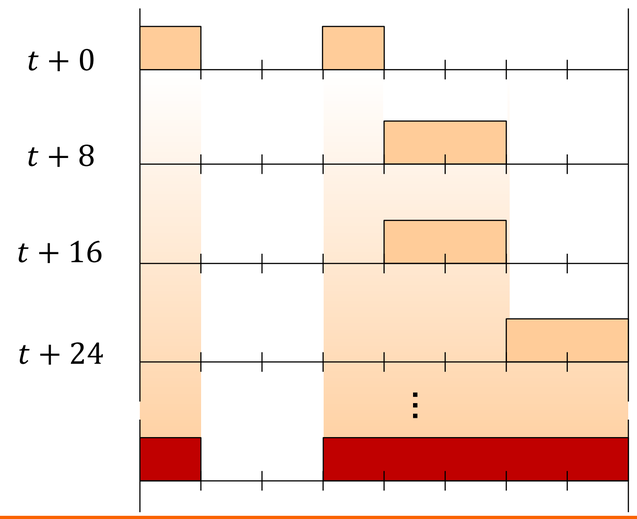


Figure: Overview of the ScheduLeak attack algorithms

SCHEDULEAK EXAMPLE

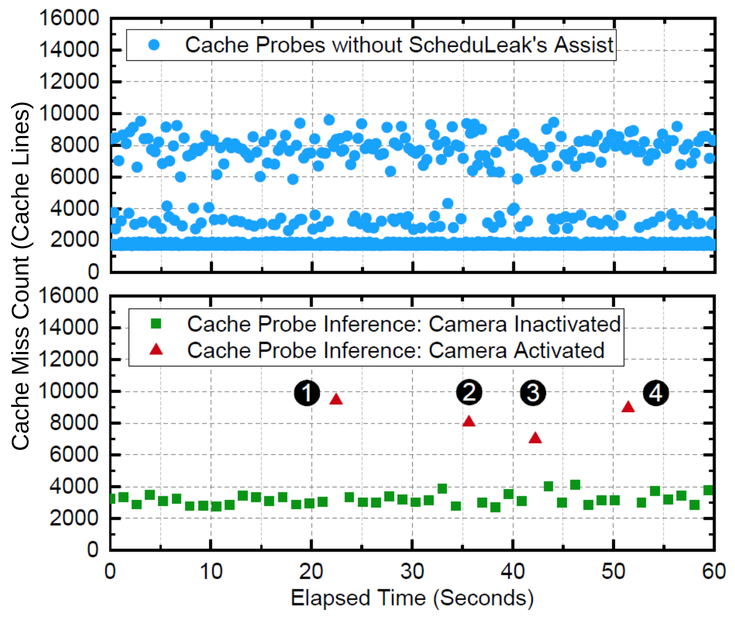
Task ID	Period	Exec. Time
Task 1	15	1
Observer Task	10	2
Victim Task	8	2
Task 4	6	1

- LCM(10, 8) = 40
- Inferred arrival window: [1, 3)
- Inferred initial offset: 1



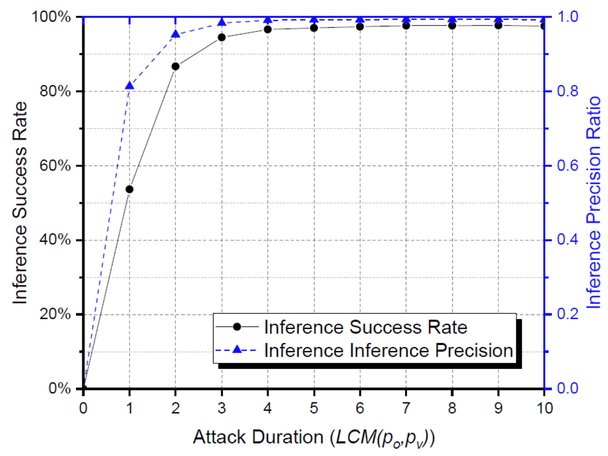
CASE STUDY

- Improving precision of cache-timing side-channel attacks
  - Attacks implemented on a hardware-in-the-loop UAV platform
    - Zedboard running FreeRTOS and a GPS trace simulator
  - Brute force attacks
    - No distinguishable point
  - With ScheduLeak
    - Points of interest identified

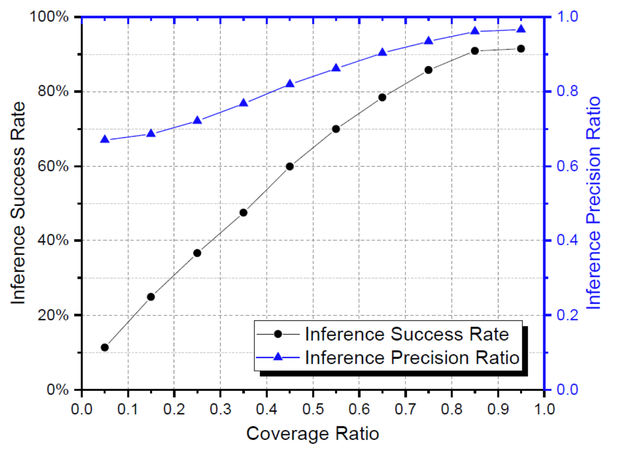


PERFORMANCE EVALUATION

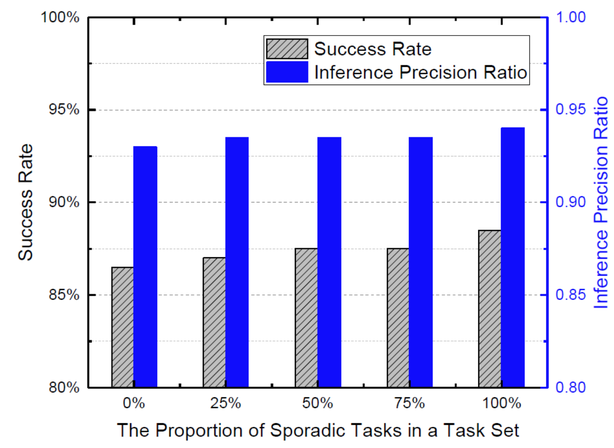
Attack Duration



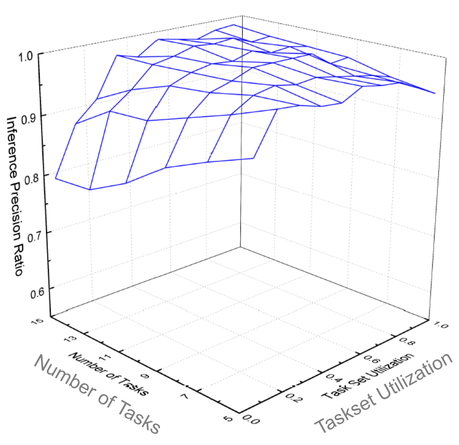
Coverage Ratio



Impact of Sporadic Tasks



Impact of Task Parameters



- Both success rate and precision ratio are stabilized after $5 \times LCM(p_o, p_v)$
  - Success Rate: 97%; Precision Ratio: 0.99
- Higher coverage ratio yields better success rate and inference precision.
  - The success rate is about 59.9% (precision ratio is 0.819) when the coverage ratio is around 0.5.
- The algorithms perform better with sporadic tasks, with a ascending trend as the proportion of sporadic tasks increases.

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
DEPARTMENT OF COMPUTER SCIENCE
COORDINATED SCIENCE LABORATORY