An Efficient Profiling-Based Side-Channel Attack on Graphics Processing Units

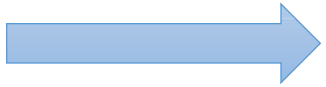
Xin Wang, Wei Zhang
Department of Electrical and Computer Engineering
Virginia Commonwealth University
Richmond, VA 23284
wzhang4@vcu.edu

Outline

- Introduction
- Background
- Profiling-based side-channel attack
- Experimental Results
- Conclusion

Introduction

- GPU (Graphics Processing Unit)
  - The graphic-oriented



high-throughput

- GPGPU (General Purpose Graphics Processing Unit)
  - Real-time computing, e.g. autonomous driving
  - General purpose computing, e.g. compute-intensive data-parallel scientific computing programs
- Particularly, security service, e.g. the encryption/decryption algorithms

Introduction

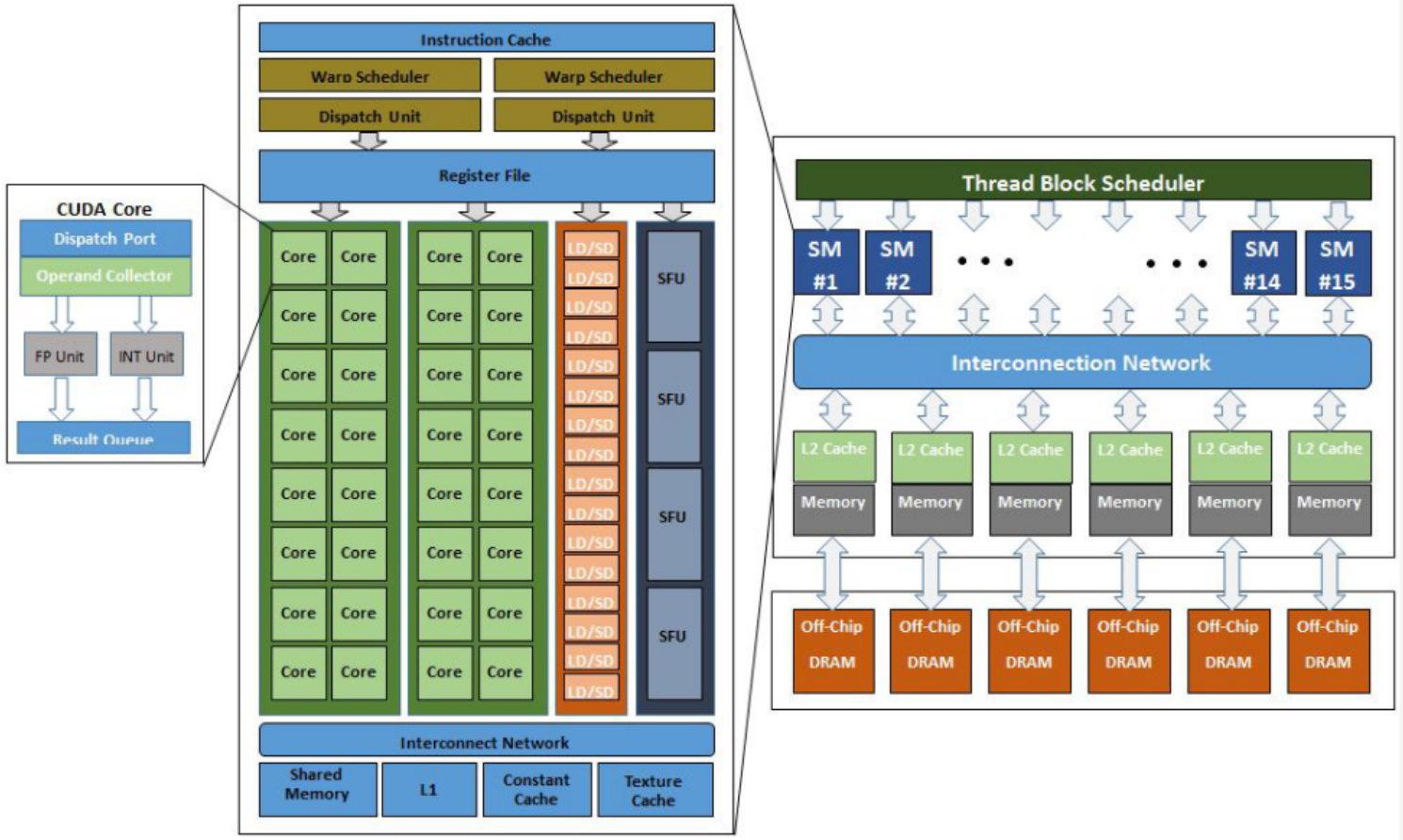
- GPU's new roles post new requirements
  - Time-predictability
  - Energy-efficiency
  - Security (focus of this work)

Introduction

- Porting the encryption/decryption algorithms to GPUs
  - Advantage: Performance profit
  - Disadvantage: Security issues have not been paid enough attention
    - e.g. vulnerability to the side-channel attacks
- This work demonstrates a ***Profiling-based Side-Channel Attack*** which can rebuild the secure key of the AES algorithm running on GPUs in less than 30 seconds

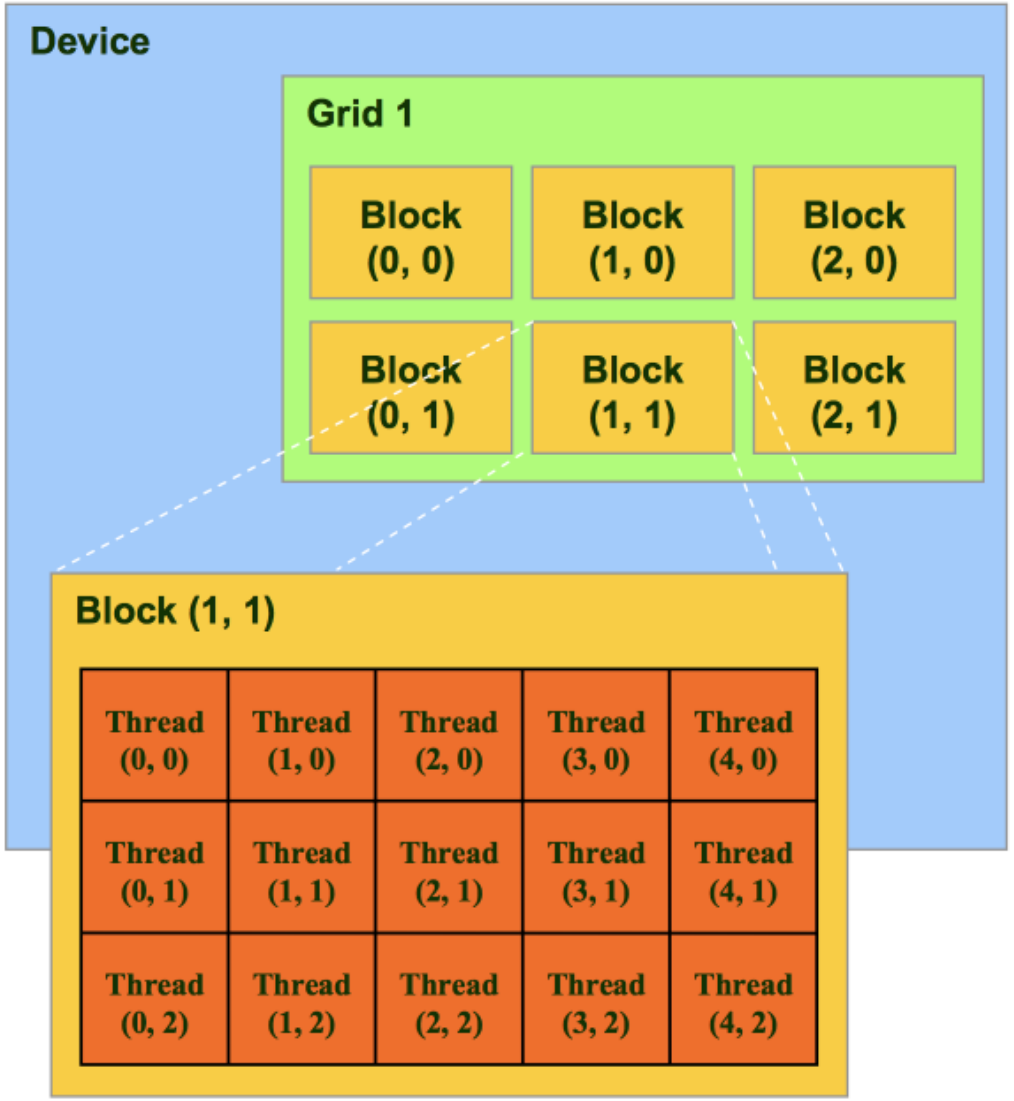
Background

- GPU architecture
  - Fermi like
  - 16 Streaming Multiprocessors
    - 32 shader processors (SPs)
    - 16 LD/ST units
    - 4 special function units (SFUs)
    - 2 warp scheduler
    - Shared storage
      - ➢ L1 cache
      - ➢ Register file
      - ➢ Shared memory



Background

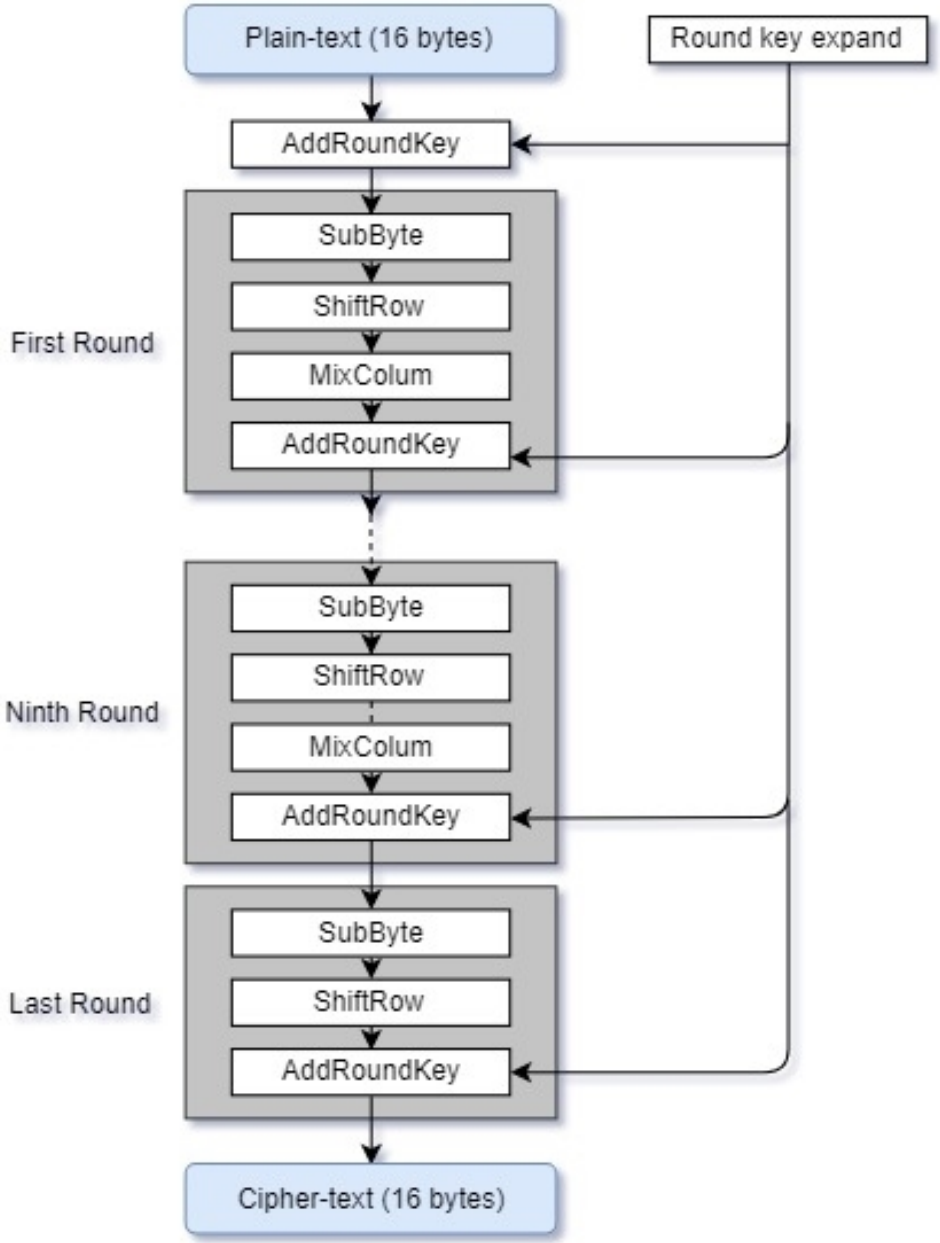
- CUDA Programming Model
  - Grid → Thread block → Warp (32 threads)
  - Warp execution:
    - A single instruction multiple threads (SIMT) way
    - One Program Counter (PC)
    - Switch between warps to hide latency
    - Memory coalescing



SIMT + Memory coalescing = vulnerability to side-channel attack !

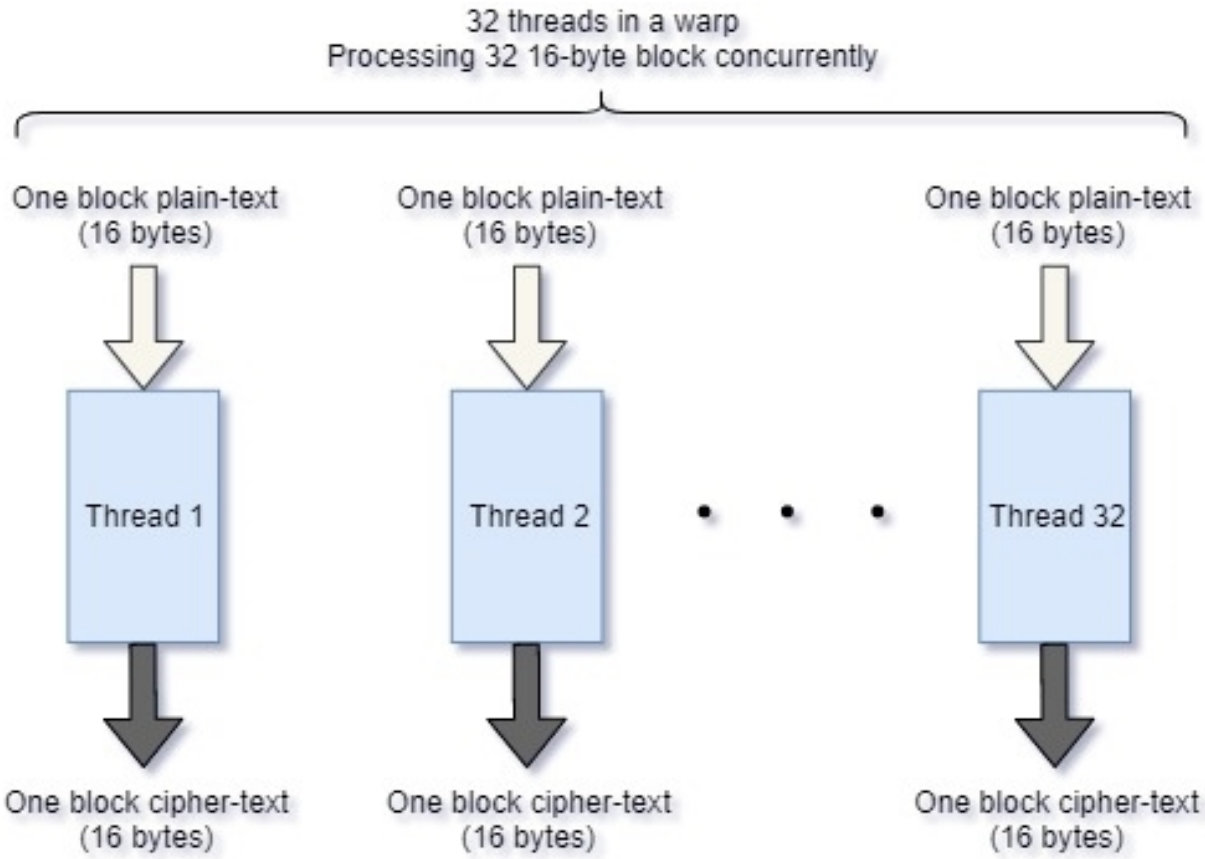
Background

- AES algorithm
  - The basic encryption unit fixed to 128 bits
  - The length of the AES key
    - 128-bit (10 encryption rounds)
    - 192-bit (12 encryption rounds)
    - 256-biy (14 encryption rounds)
  - Four operations each round:
    - SubByte
    - ShiftRow
    - MixColumn
    - AddRoundKey
  - Last round skips MixColumn operation



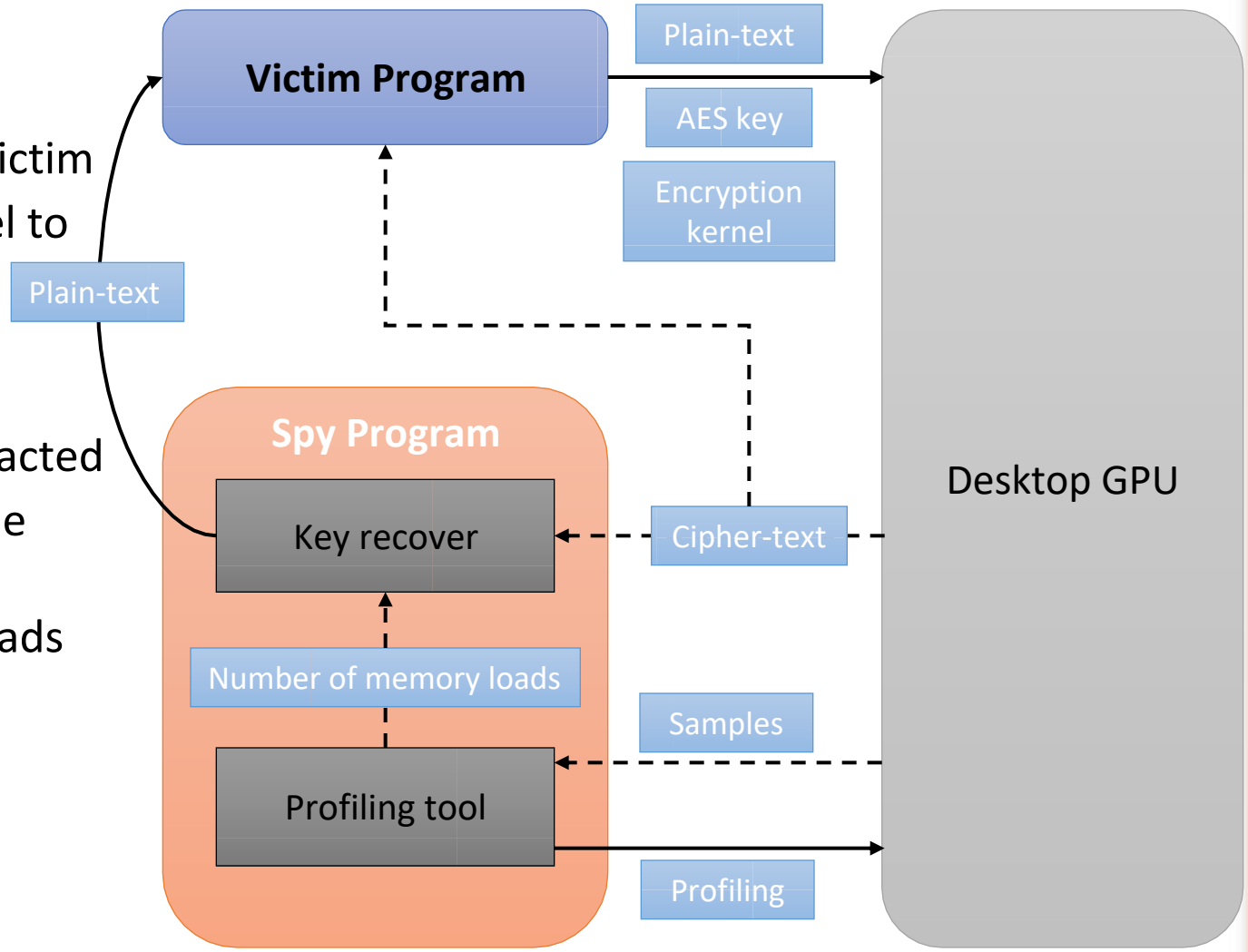
Background

- AES GPU implementation
  - The LUT (Look Up Table) based AES implements
    - Four operations performed by Four table lookups
    - The last round accesses a particular table
    - GPU implementation enables each thread to process a basic 16 bytes plain-text block



PROFILING-BASED SIDE-CHANNEL ATTACK

- The attack scenario
  - The spy keeps sending plaint-text to the victim
  - The victim sends plain-text, key and kernel to GPU
  - The spy launches the profiling tool to get samples
  - The number of memory loads then is extracted
  - The spy gets cipher-text from output of the victim
  - The spy utilizes the number of memory loads and cipher-text to recovery AES key



PROFILING-BASED SIDE-CHANNEL ATTACK

- The attack is based on three findings
  - Finding one
    - To generate the entire 16 bytes block cipher-text, 16 load and 16 store instructions are executed alternately
    - The store instruction is dependent on the previous load instruction
    - The store instructions are independent with each other

The load instructions are separated by store instructions and the 16 bytes AES key can be revealed byte by byte independently

PROFILING-BASED SIDE-CHANNEL ATTACK

- The attack is based on three findings
  - Finding two
    - The number of coalesced memory requests of one load instruction of 32 threads in a warp is highly dependent on the table indexes
    - The table indexes can be calculated with cipher-text byte and corresponding round key using an inverse lookup table

The number of memory loads can be speculatively calculated to leak the key related information

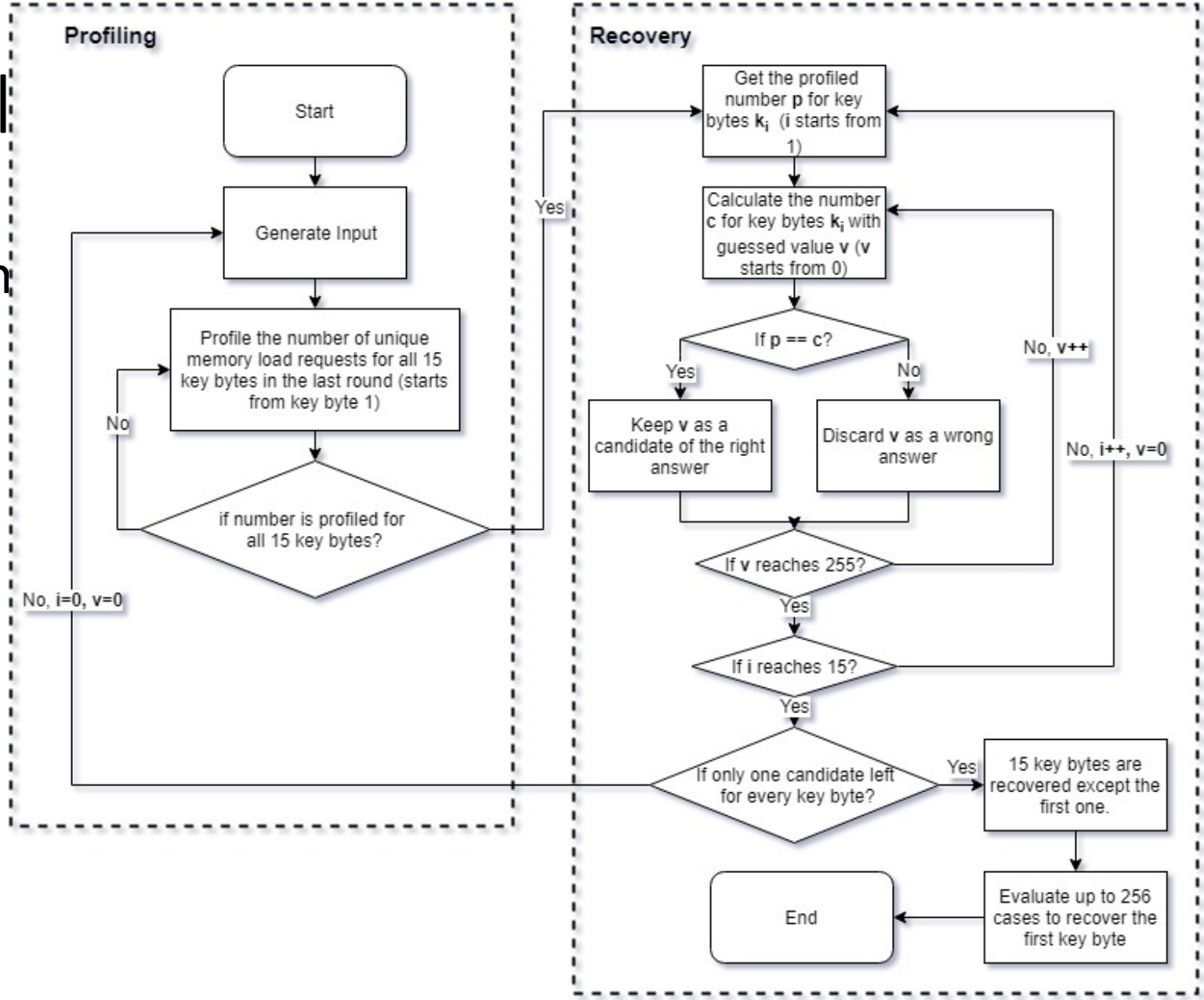
PROFILING-BASED SIDE-CHANNEL ATTACK

- The attack is based on three findings
  - Finding three
    - 64 transactions are used to store 32*16 bytes data for 32 threads
    - Store 32*1 bytes costs 4 transactions

The number of memory stores can be used to distinguish the number of memory loads for each load instruction in the last encryption round

PROFI ACK

- Flow ch



Experimental results

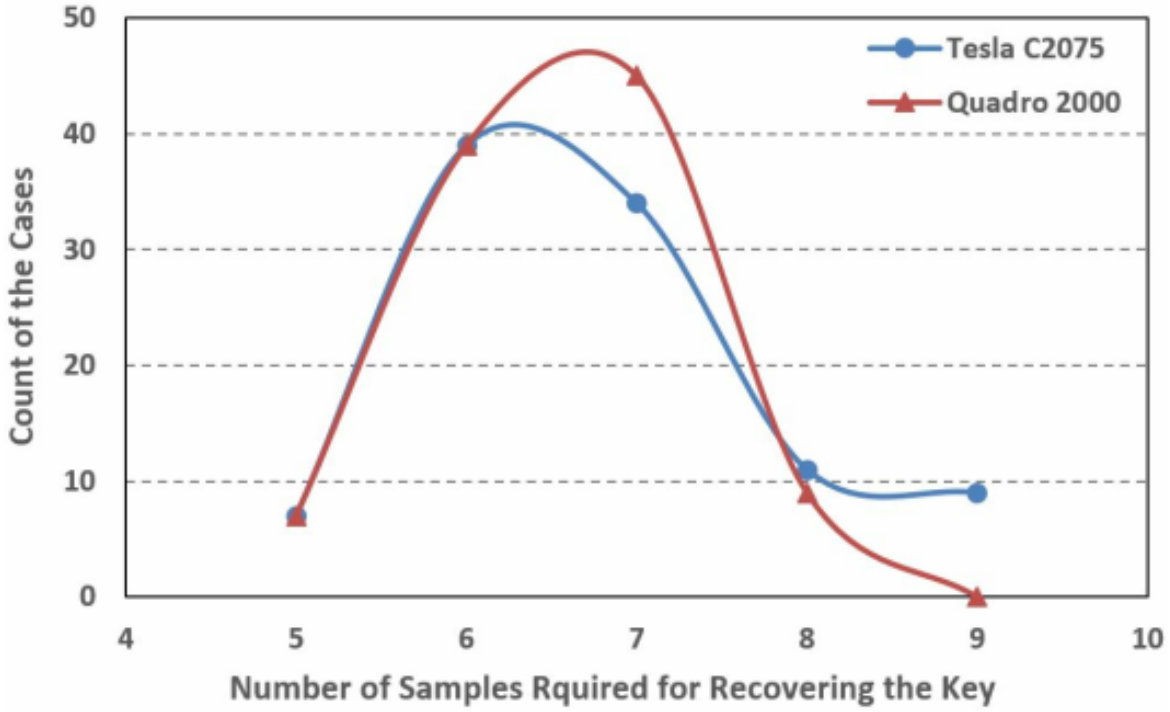
- The pSCA is evaluated on two GPU cards
  - NVIDIA Quadro 2000
  - NVIDIA Tesla C2075

High accuracy and quick recovery

	Tesla C2075	**Quadro 2000**
Key recovery done	success:100	success:100
Recovery success rate	100%	100%
Profiling done	success:676	success:656, fail:1
Profiling accuracy	100%	99.85%
The average profiling time	13814.99 ms	29837.02 ms
The average recovery time	93389.37 ms	196029.20 ms
The average inputs profiled	6.76	6.57

Experimental results

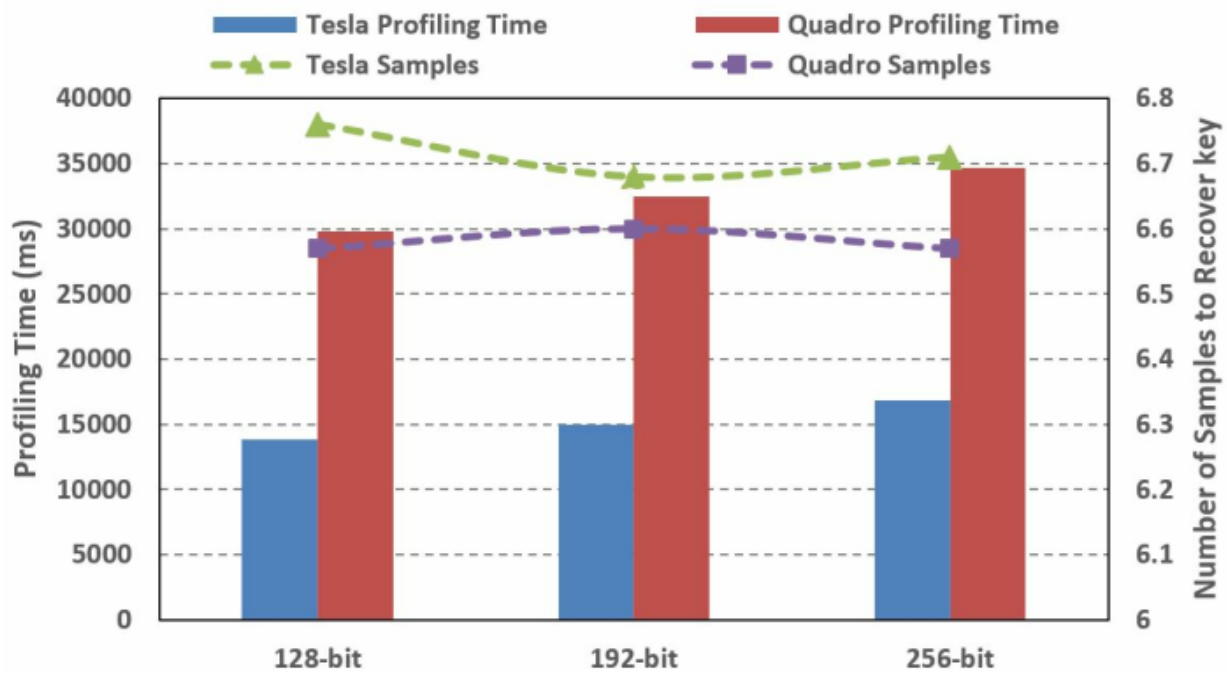
- The number of samples required to recover an entire AES key
  - In most cases, information retrieved from 6 or 7 samples is enough
  - The average number of samples is 6.76 and 6.57 for Tesla C2075 and



The number of samples is small

Experimental results

- The performance as the length of the AES key is increased to 192-bit and 256-bit
  - The extended AES key does not increase the number of samples for the key recovery
  - The time to profile one sample increases a little bit



Good scalability

Conclusion

- The Profiling-based Side-channel attack provides,
  - Significantly reduced number of samples
    - 10 samples vs. 1000000 samples for existing SCA
  - Straightforward key recovery procedure
    - No statistical analysis model is required
  - Guaranteed accuracy
    - Approaching 100%
  - Good scalability
    - Reveal longer AES key with similar number of samples

Conclusion

- The Profiling-based Side-channel attack exposes,
  - Serious threats of the SCA on GPUs
    - The AES key recovery procedure costs as short as 30 seconds with accuracy approaching 100%.

Thank you for you attention!