GPGPU for side channel attacks

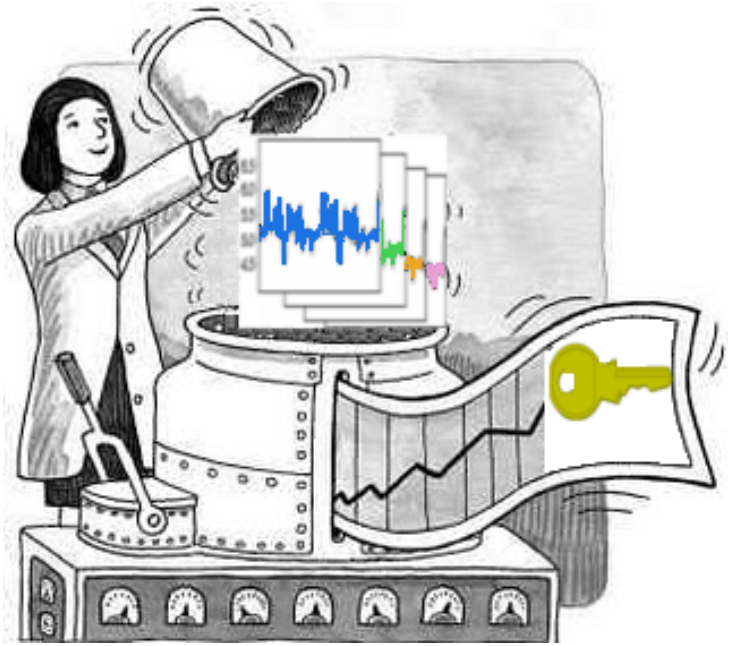
Yanis LINGE[1,2], Cécile DUMAS[1], Sophie LAMBERT-LACROIX[2]

CEA-LETI/MINATEC

UJF-Grenoble 1 / CNRS / UPMF / TIMC-IMAG.

Plan

Context : Side Channel Analysis



- The required trace number grows faster than the computation power.
- Need for new implementation techniques.
  ⇒An easy solution : OpenMP, API for parallel programming on multicore CPU.
  ⇒A more efficient solution : GPU.

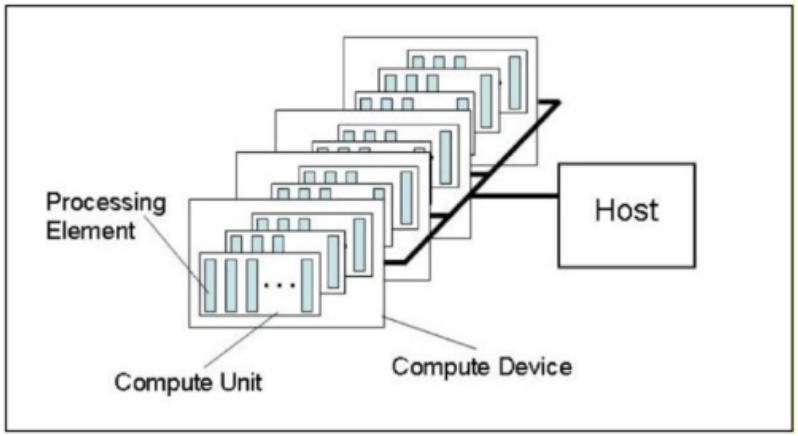
GPU

- GPU : Graphic Processing Unit.
- Specialized electronic circuit designed to rapidly manipulate and alter memory in a frame buffer.
- Present in : graphic card, tablet, recent smartphone, etc.

GPU API

- NVIDIA : CUDA.
- ATI : ATI Stream.
- Standard : OpenCL.

OpenCL platform model



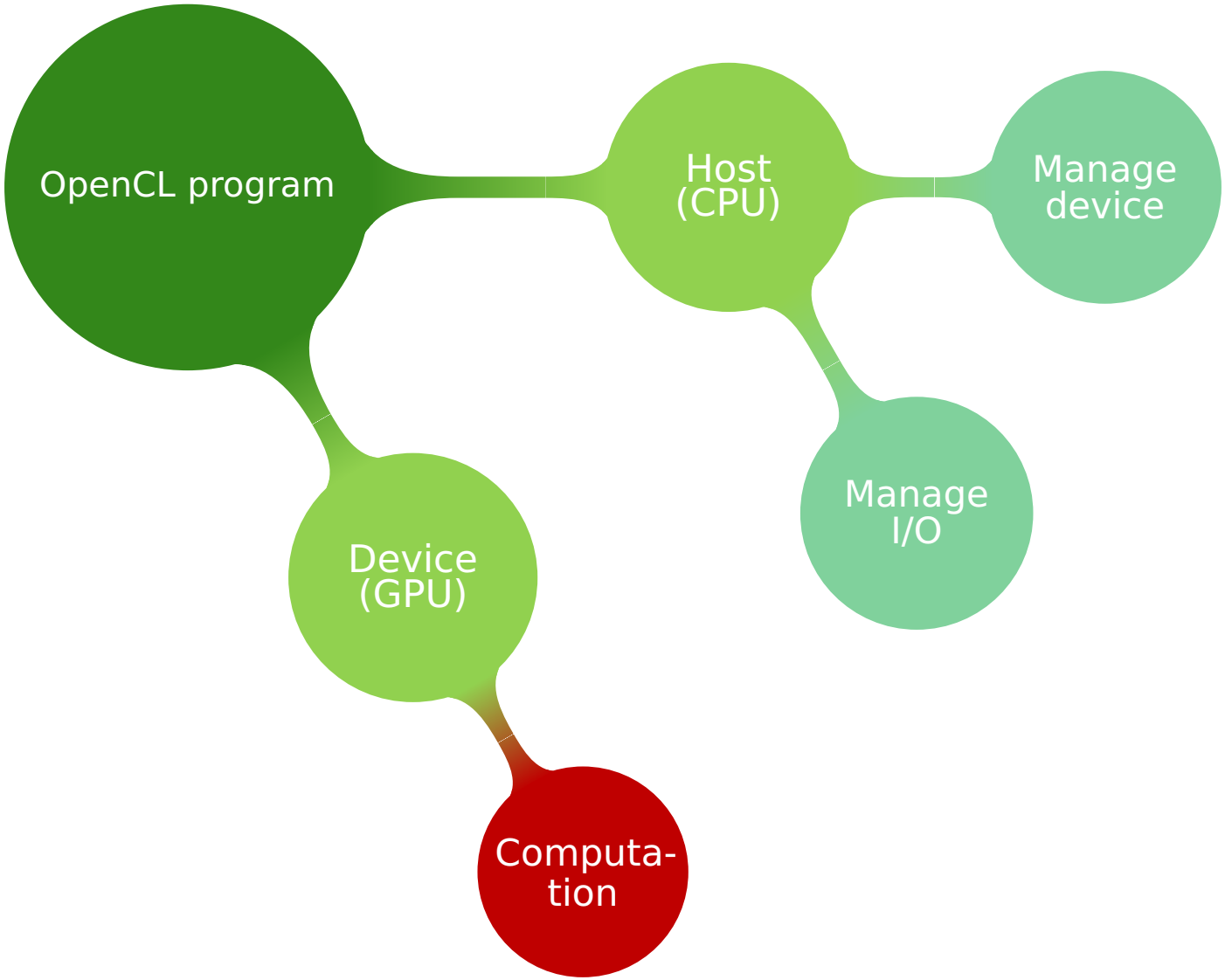
- One host...
- ...connected to computing devices (for example GPU)...
- ...regrouping several computing units.

- For GPU : Single Instruction Multiple Data (SIMD).

High computing power but not for free :

- Parallelization constrained by SIMD programming model.
  $\Rightarrow$ same instruction computed by the computing units

OpenCL program



Example : Correlation Power Analysis (Brier *et al*. 2004)

- N traces C_i associated to a model $H_k(i)$ for each key hypothesis k
- The Pearson coefficient is computed for each instant x of the traces.

$$\rho(x)_k = \frac{N.\sum_{i=1}^{N} C_i(x).H_k(i) - \sum_{i=1}^{N} C_i(x).\sum_{i=1}^{N} H_k(i)}{\sqrt{N.\sum_{i=1}^{N} C_i(x)^2 - \left(\sum_{i=1}^{N} C_i(x)\right)^2}.\sqrt{N.\sum_{i=1}^{N} H_k(i)^2 - \left(\sum_{i=1}^{N} H_k(i)\right)^2}}$$

Three kernels :

- Compute : $\sum_{i=1}^{N} H_k(i)$ and $\sum_{i=1}^{N} H_k(i)^2$
- Compute : $\sum_{i=1}^{N} C_i(x)$, $\sum_{i=1}^{N} C_i(x)^2$ and $\sum_{i=1}^{N} C_i(x)H_k(i)$
- Compute the Pearson coefficient for each k and each x

Comparisons of different implementations

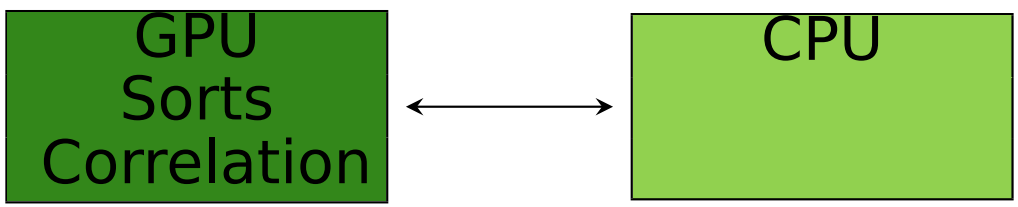
- Hardware :
  - CPU : XEON X3430
    - 4 cores
    - clock speed : 2.4GHz
    - cost : $\approx$ 200$
  - GPU : ENGTS450
    - 192 CUDA cores
    - clock speed : 0.78GHz
    - cost : $\approx$ 200$
- Results : order of magnitude

Algorithm	Sequential	OpenMP	OpenCL
CPA	T	$T/3$	$T/12$
Variance $\overline{C(x)^2} - \overline{C(x)}^2$	T'	$T'/3$	$T'/12$

Example : Rank Correlation (Batina *et al*. 2008)

- Same computation than the CPA but the values of $C_i(x)$ and $H_k(i)$ are replaced by their rank.
- Many sets of size N have to be sorted.
- SIMD not designed for conditional function and sorting.

$\Rightarrow$ ”only” 5 times faster than the sequential version.



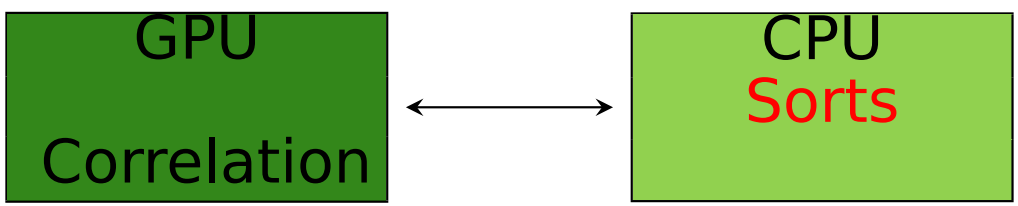
Example : Rank Correlation (Batina *et al*. 2008)

- Same computation than the CPA but the values of $C_i(x)$ and $H_k(i)$ are replaced by their rank.
- Many sets of size N have to be sorted.
- SIMD not designed for conditional function and sorting.

  $\Rightarrow$ ”only” 5 times faster than the sequential version.

- Solution : Sorts by the CPU and the correlation computation by the GPU.

  $\Rightarrow$ 8 times faster than the sequential version.



Conclusion

- GPU usefull for SCA.
- Many algorithms from the Side Channel Analysis and from the Signal Processing.
- But not all of them.

$\Rightarrow$ It is important to respect SIMD concepts and correctly determine the parallelism in the algorithm.
$\Rightarrow$ Need for a prior analysis.

Leti

LABORATOIRE D'ÉLECTRONIQUE
ET DE TECHNOLOGIES
DE L'INFORMATION

CEA-Leti
MINATEC Campus, 17 rue des Martyrs
38054 GRENOBLE Cedex 9
Tel. +33 4 38 78 36 25

www.leti.fr

Thank you !

contact : yanis.linge@cea.fr