An Analytical Model for Time-Driven Cache Attacks

Kris Tiri
Onur Acıiçmez
Michael Neve
Flemming Andersen

Outline

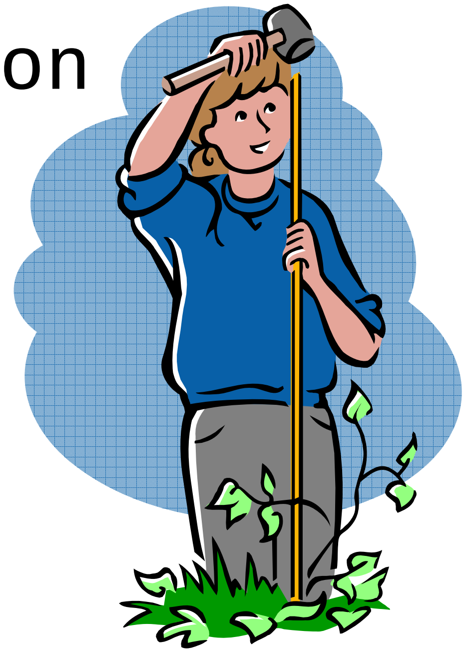
- Motivation
- Cache attacks: origins, time-driven attack
- Strength of an implementation
- Analytical model of time-driven attack
- Experimental results
- Conclusions

Side-Channels

- Information leakage from implementation
  - Example: safecracker “feels” tumblers impacting
  - Covert channel without conspiracy or consent
- Cache Side-Channel Attacks
  - 1996: presumed possible [Kocher]
  - 2002: theoretical work [Page]
  - 2003: first practical results on DES [Tsunoo]
  - 2005: first practical results on AES, RSA [Bernstein][Osvik][Percival]

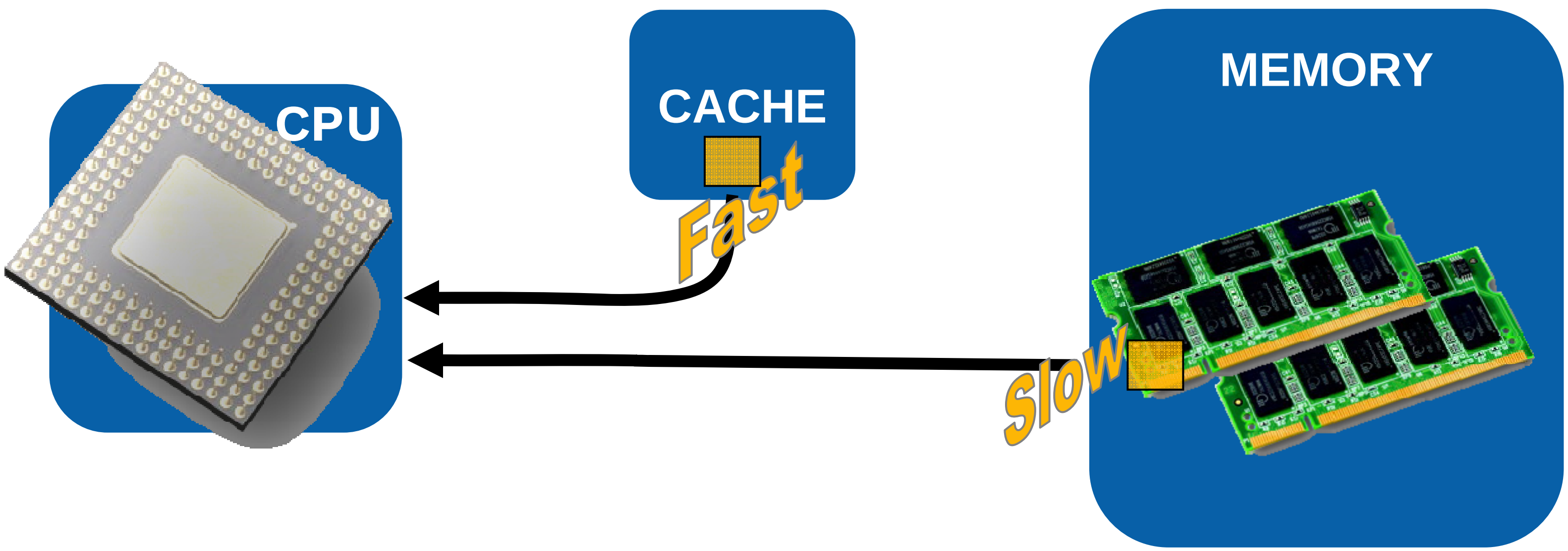
Motivation

- Attack depends on crypto implementation and on cache architecture
- Experimental results cumbersome to obtain
- Can we put a stake in the ground on strength of <u>any implementation</u> of <u>any symmetric key algorithm</u> running on <u>any microprocessor</u> w.r.t. a time-driven cache attack?



Cache attack origins

- Information leaks resulting from the implementation of the cache



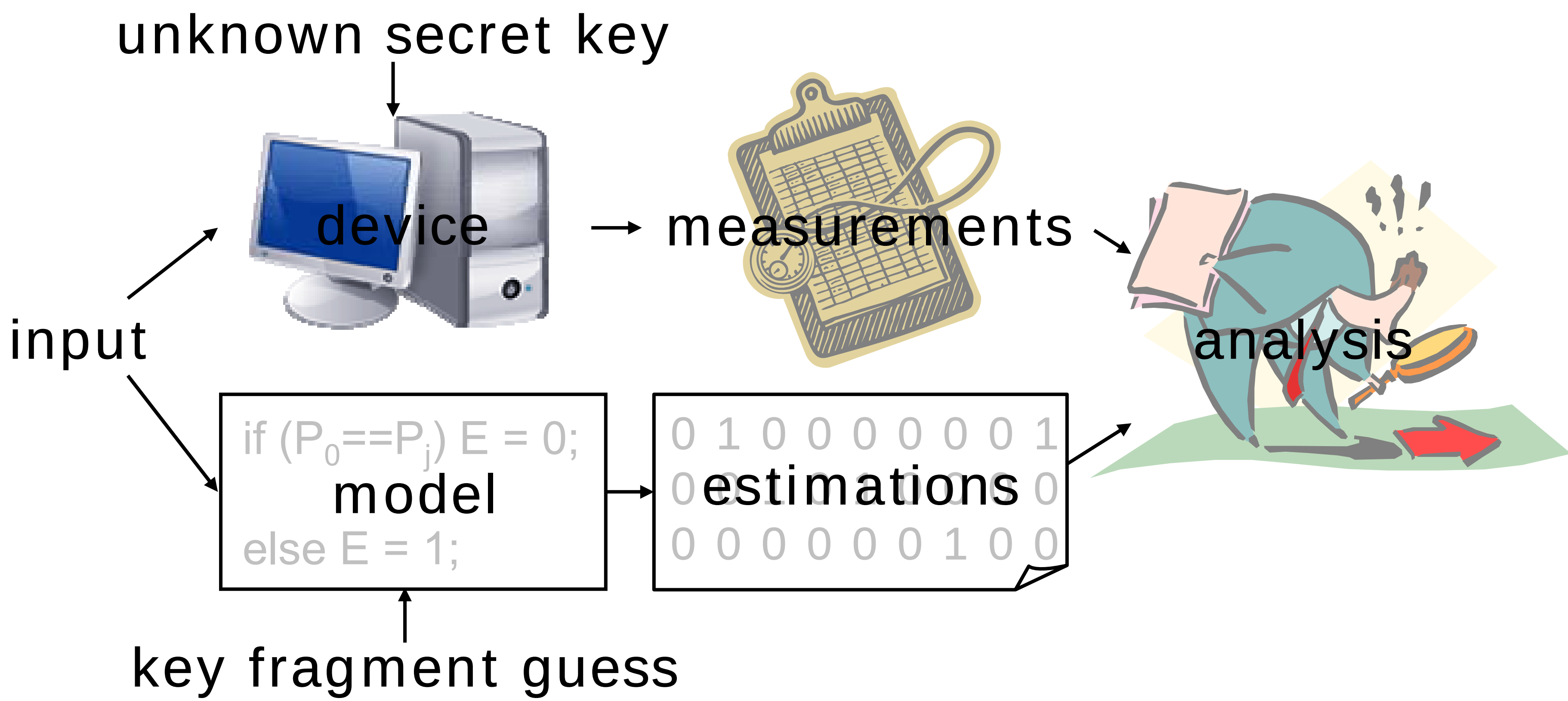
- Difference between cache hit & cache miss is observable/measurable

Cache attacks in a nutshell

- Cache is shared between processes
- Cache state persists despite context switch
- Data is protected, metadata is unprotected
- Cache access pattern depends on cache state and processed data
- *Spy*-process can observe key-dependent cache accesses of *crypto*-process
- Observation techniques: time-driven attack, trace-driven attack, access-driven attack

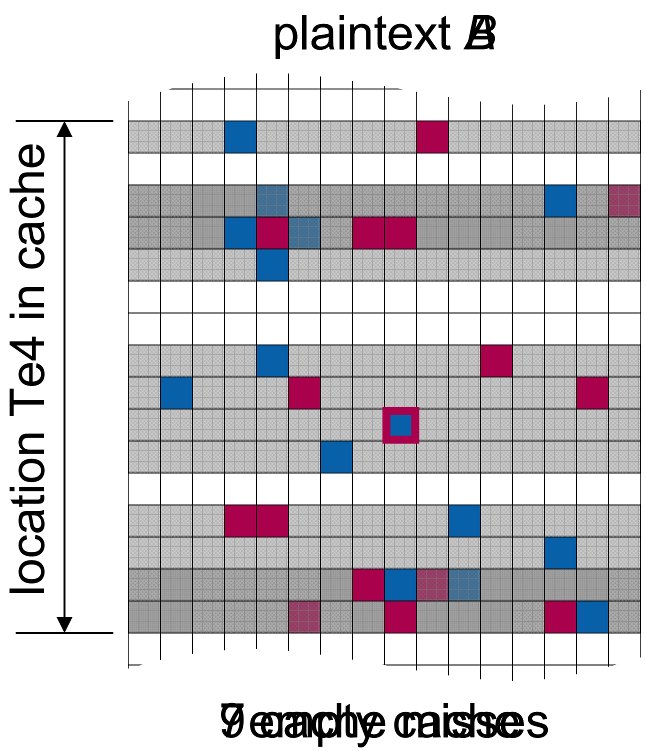
Time-driven cache attacks

- Leakage: number of cache misses depend on data



Example: last round attack on AES

- OpenSSL: 5 tables (Te0..4) of 1024 bytes
  - 16 accesses to table Te4 in last round



- device:
  execution time ~ all cache misses
- model:
  if (collision) estimation = 0;
  else estimation = 1;
- cache line estimation
  $<sbox^{-1}(RK_0^{(10)} \oplus C_0)> == <sbox^{-1}(RK_i^{(10)} \oplus C_i)>$
- table index estimation
  $C_0 == RK_{0i}^{(10)} \oplus C_i$ *with* $RK_{0i}^{(10)} = RK_0^{(10)} \oplus RK_i^{(10)}$

Strength / Resistance of an implementation

- How many measurements are required?

[Mangard2005]

$$N = \frac{2 . Z_{\alpha}^{2}}{\rho^{2}}$$

- Quantile of standard normal distribution for probability α

  How sure do you want to be?

- Correlation coefficient between estimations and measurements

  How accurate is your model?

1. model the measurements
2. compute ρ between estimations and modeled measurements

Model the measurements

Assumptions:

1. Cache is clean before cipher operation
2. No collision between lookup tables
3. Cache accesses are random, independent
4. Cipher operation operates uninterrupted
5. Execution time proportional to number of cache misses

Compute ρ between estimations and modeled measurements

$$\rho = \frac{E(E_{K_{secret}}.M) - E(E_{K_{secret}}).E(M)}{\sqrt{E(E^2_{K_{secret}}) - E(E_{K_{secret}})^2}\sqrt{E(M^2) - E(M)^2}}$$

- time ~ cache misses:

$$\rho(E, M_{time}) = \rho(E, M_{misses})$$

- independent accesses to T tables:

$$E(M) = \sum_{t=1}^{T} E(M_t)$$

- measurement model with *k* accesses to *l* lines:

$$\mu_M(k,l) = \sum_{j=1}^{l} j.P_{k,l}(j)$$

$$\sigma^2_M(k,l) = \sum_{j=1}^{l} j^2.P_{k,l}(j) - \mu^2_M(k,l)$$

Compute ρ between estimations and modeled measurements

$$\rho = \frac{E(E_{K_{secret}}.M) - E(E_{K_{secret}}).E(M)}{\sqrt{E(E_{K_{secret}}^2) - E(E_{K_{secret}})^2}\sqrt{E(M^2) - E(M)^2}}$$

- let's estimate cache hits

$$|\rho(E_{miss}, M)| = |\rho(E_{hits}, M)|$$

to ease

$$E(E) = 1.P(E=1) + 0.P(E=0)$$

$$\frac{1}{r_T} \xleftarrow{TIE} P(E=1) \xrightarrow{CLE} \frac{1}{l_T}$$

- independent accesses
- correct prediction

$$E(E_{K_{secret}}.M) = E(E_{K_{secret}}.M_T) + \sum_{t=1}^{T-1} E(E_{K_{secret}}).E(M_t)$$

$$\mu_H(k,l) = \mu_M(k-1,l)$$

Putting the pieces together…

analytical model for time-driven cache attacks

$$N = \frac{2.Z_{\alpha}^{2}}{\frac{\mu_E^2}{\sigma_E^2} \cdot \frac{\mu_D^2(k_T, l_T)}{\sum_{t=1}^{T} \sigma_M^2(k_t, l_t)}}$$

- probability α to find key
- k_t accesses to table t consisting of r_t *elements* occupying l_t cache lines
- T tables in cipher operation
- table T is table of interest

Example: attack on last round AES

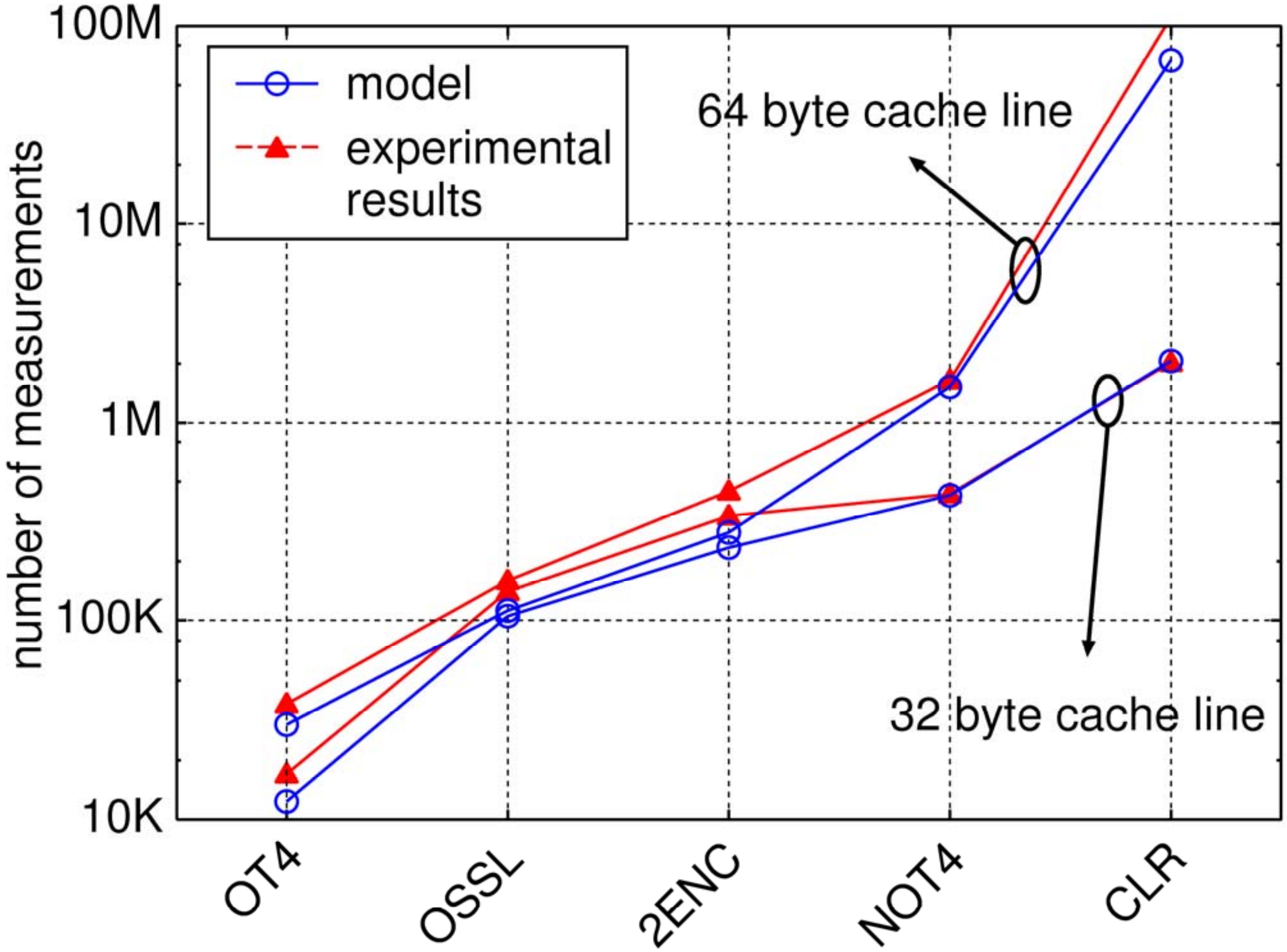
$$N = \frac{2.Z_\alpha^2}{\dfrac{\mu_E^2}{\sigma_E^2} \cdot \dfrac{\mu_D^2(k_T, l_T)}{\sum_{t=1}^{T} \sigma_M^2(k_t, l_t)}}$$

$$\frac{1/l_T^2}{1/l_T - 1/l_T^2}$$

- cache line estimation
- 99% success
- 16 accesses to table of interest Te4 of 16 lines
- 36 accesses to 4 tables Te0..3 each of 16 lines
- measured: 10000

$$N = \frac{11}{\dfrac{1/16^2}{1/16 - 1/16^2} \cdot \dfrac{\mu_D^2(16,16)}{4.\sigma_M^2(36,16) + \sigma_M^2(16,16)}} = 6592$$

Experimental results
last round, table index estimation



setup:

- single process
- perf-counters

experiments:

1. observe only Te4
2. OpenSSL version
3. 2 encryptions
4. no Te4
5. compact last round

Further insights

- Cache line estimation is l_T/r_T times more effective than table index estimation
- Yet 2^{16} key search space instead of 2^8

$$\frac{N|_{TIE}}{N|_{CLE}} = \frac{\left.\mu_E^2 \middle/ \sigma_E^2\right|_{CLE}}{\left.\mu_E^2 \middle/ \sigma_E^2\right|_{TIE}} \approx \frac{r_T}{l_T}$$

e.g. 64 byte cache line:

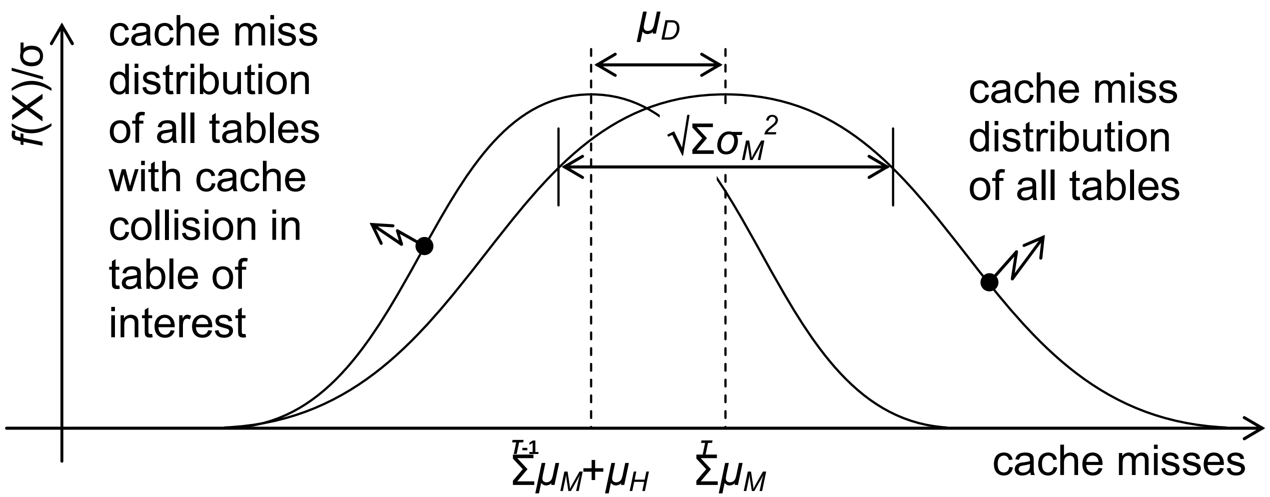
$time_{TIE} = 16.N.2^8.\Delta_{time}$

$time_{CLE} = N.2^{16}.\Delta_{time}$

Universal model

- Metric is based on signal-to-noise ratio

$$\frac{N_B}{N_A} = \frac{\mu_D^2\left(k_{T_A}, l_{T_A}\right)}{\sum_{t_A=1}^{T_A} \sigma_M^2\left(k_{t_A}, l_{t_A}\right)} \Bigg/ \frac{\mu_D^2\left(k_{T_B}, l_{T_B}\right)}{\sum_{t_B=1}^{T_B} \sigma_M^2\left(k_{t_B}, l_{t_B}\right)} = \frac{SNR_A}{SNR_B}$$



Conclusions

- Analytical model forecasts resistance of block cipher implementations against time-driven cache attacks using:
    1. Number of lookup tables
    2. Size of lookup tables
    3. Size of cache line
- Model accuracy verified with measurement results for different implementations, attack scenarios and platforms