APPLICATIONS OF LLL: BREAKING REAL-WORLD RSA

PHONG NGUYEN

http://www.di.ens.fr/~pnguyen

November 2024

LATTICE-BASED CRYPTANALYSIS

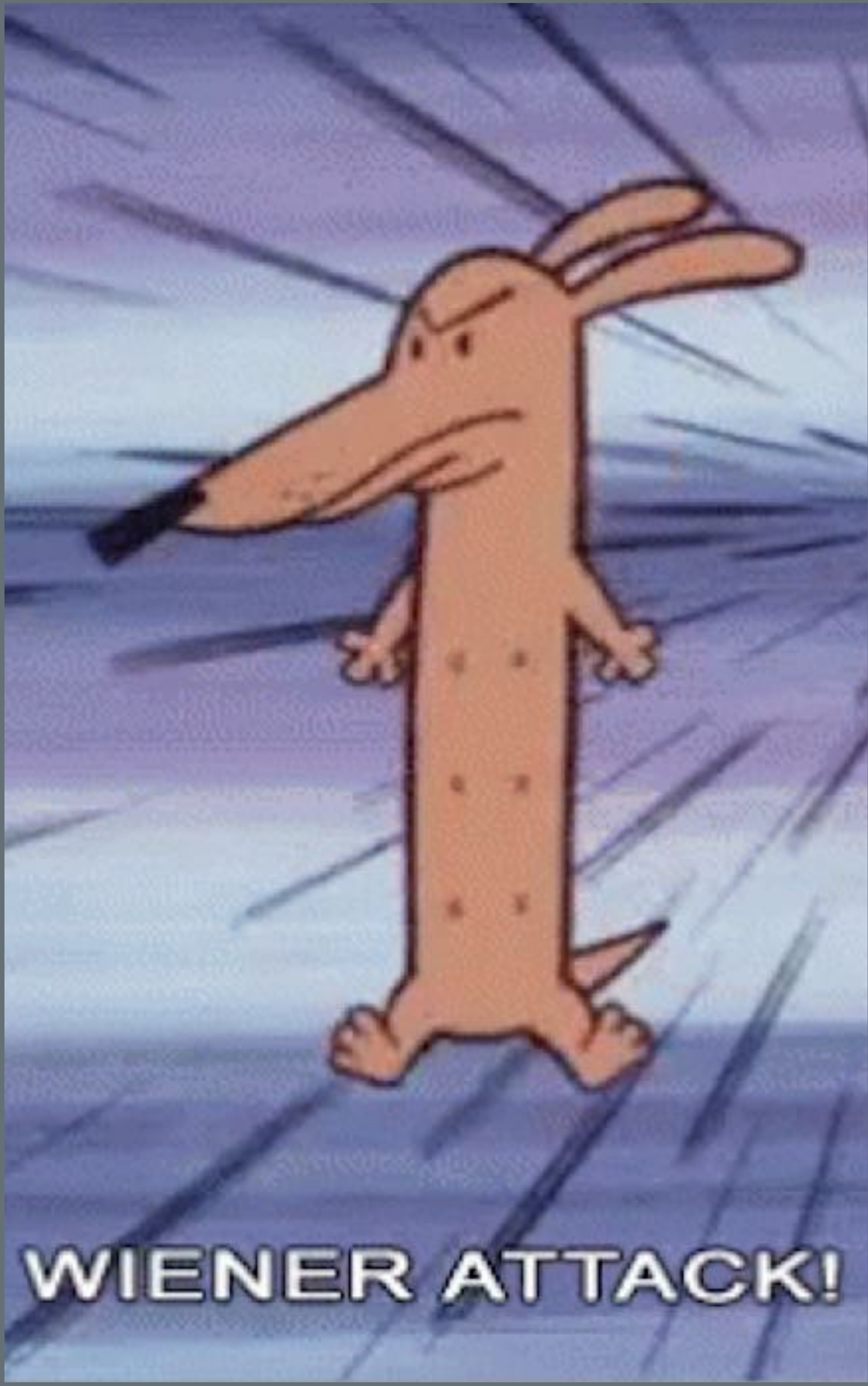
- Lattice algorithms have been used to break many cryptosystems, including:
  - Special settings of RSA: small roots of polynomial equations [Cop96]
    - ROCA [NSSKM17]: Factor N=pq when p is a power of 65537 modulo many small primes.

  - Special settings of Discrete Log: small roots of linear congruences
    - Attacking DSA/ECDSA with hints on nonces, such as in Bitcoin/TLS/SSH.

TODAY

- Wiener's Attack
- Small-Roots Attack

WIENER'S ATTACK



REMEMBER RSA

- $N=pq$ product of two large random primes.
- $ed \equiv 1 \pmod{\phi(N)}$ where $\phi(N)=(p-1)(q-1)$
  - e is the public exponent
  - d is the secret exponent
- Then $m \to m^e$ is a trapdoor one-way permutation over $\mathbf{Z}/N\mathbf{Z}$, whose inverse is $c \to c^d$.

SHORT-SECRET RSA

- To speed-up RSA secret operations, we want to select a short d.
  - Assume $d \ll N$
    - Can we recover d from (e,N)?
  - $ed = 1+k\phi(N)$ where $\phi(N)=(p-1)(q-1)=N+O(\sqrt{N})$
  - So, $k=O(d)$ and $ed \approx kN$, namely $ed-kN=O(d\sqrt{N})$.

LATTICES AND SHORT-SECRET RSA

- Consider the 2-dim lattice L spanned by:

$$\begin{bmatrix} e & \sqrt{N} \\ N & 0 \end{bmatrix}$$

- It contains the vector t=dx(1st row)-kx(2nd row).

LATTICES AND SHORT-SECRET RSA

- How short is $t=d\times(\text{1st row})-k\times(\text{2nd row})$?
    - Its 1st coordinate is $ed-kN=O(d\sqrt{N})$.
    - Its 2nd coordinate is $d\sqrt{N}$.

- So $||t||=O(d\sqrt{N})$.

- This is unusually short if $||t||\leq vol(L)^{1/2}=N^{3/4}$ i.e. $d\leq O(N^{1/4})$, then t is ``likely'' to be a shortest vector of L.

LATTICE ATTACK ON SHORT-SECRET RSA

- Compute a shortest vector of the 2-dim lattice L:

➤ This only takes polynomial-time, less than 1s for 2048-bit RSA.

- If it is $\pm t$, recover (k,d): how?

- Check that (k,d) is correct: how?

LATTICE ATTACK ON SHORT-SECRET RSA

- If it is $\pm t$, recover (k,d): how?
    - Divide the 2nd coordinate by $\sqrt{N}$.

- Check that (k,d) is correct: how?
    - $ed-kN=1-k(p+q-1)$.
    - Derive $p+q$.
    - Recover p and q by solving $X^2-(p+q)X+N=0$.

WIENER'S ATTACK (1989)

- Using continued fractions instead of lattices, Wiener showed:

Michael J. Wiener

- Theorem: If $q<p<2q$ and $1\leq d\leq N^{1/4}/3$, one can recover p and q in polynomial time from (N,e).

- [BonehDurfee1999]: There is a heuristic (lattice) attack recovering p and q in polynomial time from (N,e) if $d\leq N^{0.292\ldots}$

SMALL-ROOTS ATTACKS

BREAKING RSA WITHOUT FACTORING

- In 1996, Coppersmith showed how to solve two problems in polynomial time using lattices:

Don Coppersmith

  - Given a monic polynomial P in **Z**[X] and an integer N, find all "small" integers x s.t. $P(x) \equiv 0 \pmod N$.

  - Given an irreducible polynomial P in **Z**[X,Y], find all "small" integers x and y s.t. $P(x,y) = 0$.

APPLICATIONS TO RSA

- This and generalizations lead to breaking many special cases of RSA
  - When the secret exponent d is too small.
  - When half of the bits of p are known.
  - When the public exponent e is small, and only a fraction of the plaintext is unknown.

STEREOTYPED ATTACK

- Assume that $e=3$, N is 2048-bit, and that we encrypt a 128-bit AES key m by padding a known constant like « Today's key is ».
  - $c=(m+b)^e \pmod N$.
  - What is the problem?

FACTORING WITH A HINT [COP96]

- $N=pq$ where $p=p_0+\varepsilon$ for some small ε.
- Let $f(x)=p_0+x$.
- Then $\gcd(f(\varepsilon),N)=p$ is large.
- Can recover ε and p if $|\varepsilon|\leq N^{1/4}$

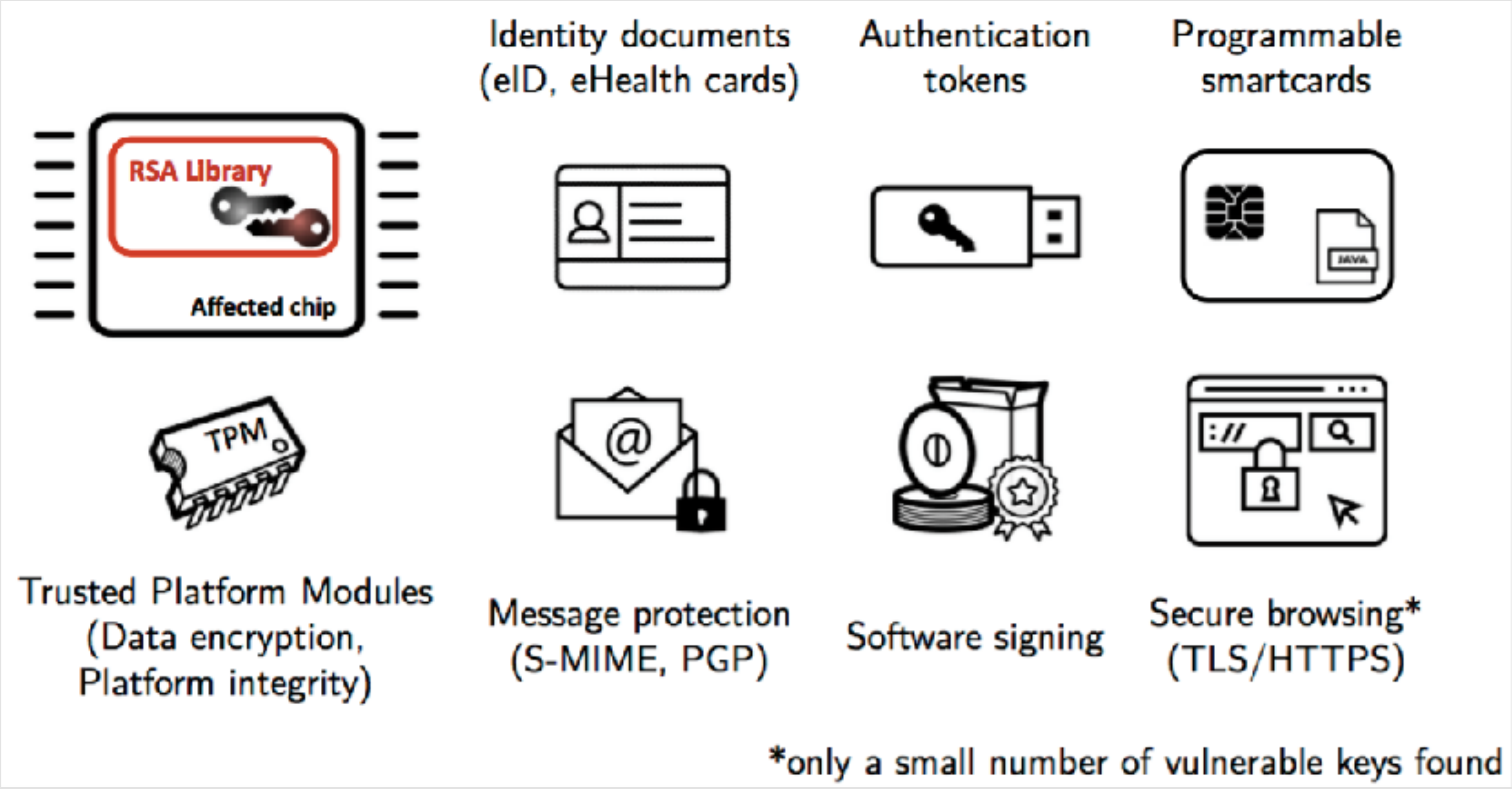
ANOTHER REAL-WORLD ATTACK

- Attack on Infineon RSA keys.

- See ACM CCS '17:

- The Return of Coppersmith's Attack: Practical Factorization of Widely Used RSA Moduli by Matus Nemec, Marek Sys, Petr Svenda, Dusan Klinec, Vashek Matyas (Masaryk University).

IMPACT



- Ex: Estonia's 750,000 ID cards.

IDENTIFYING RSA KEYS

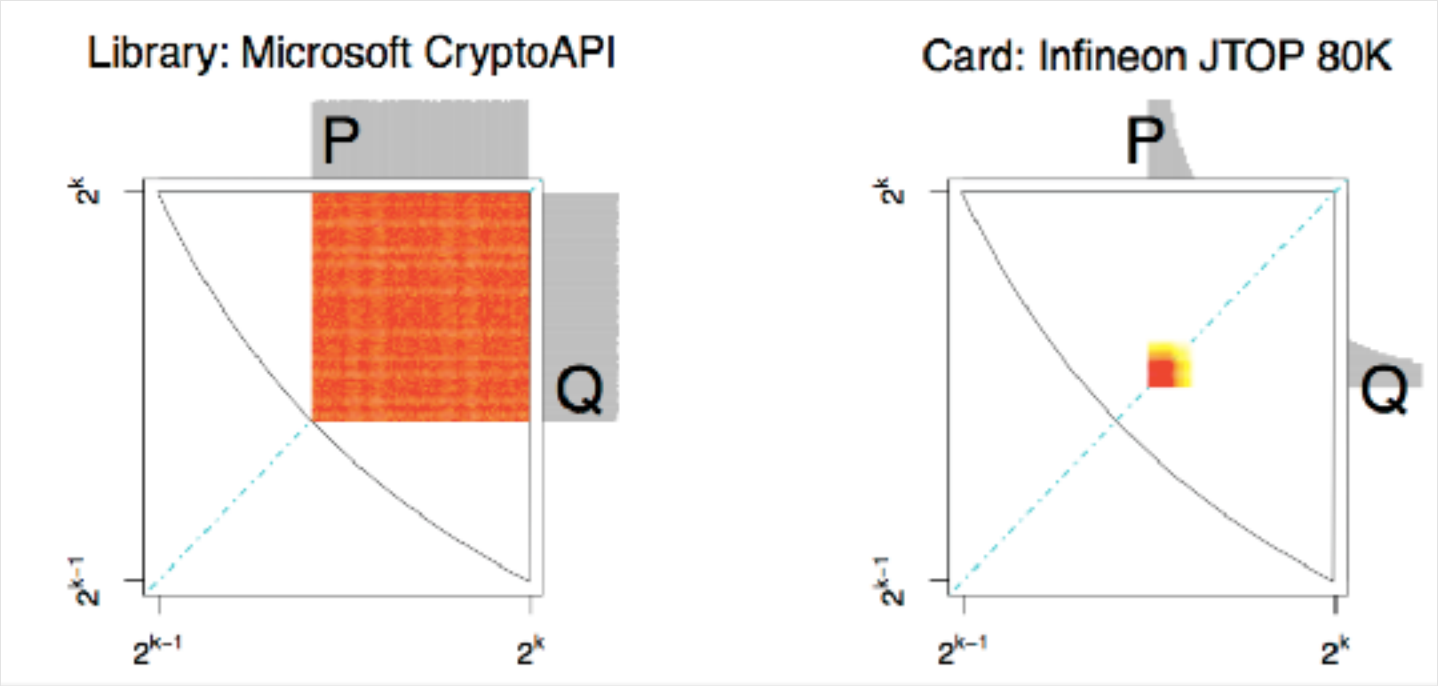
- Svenda et al. analyzed 60 millions fresh keys produced by 22 libraries and 16 smartcards from 6 manufacturers.

- Most distributions of $N=pq$ and/or p were different and could be identified!

WHY?

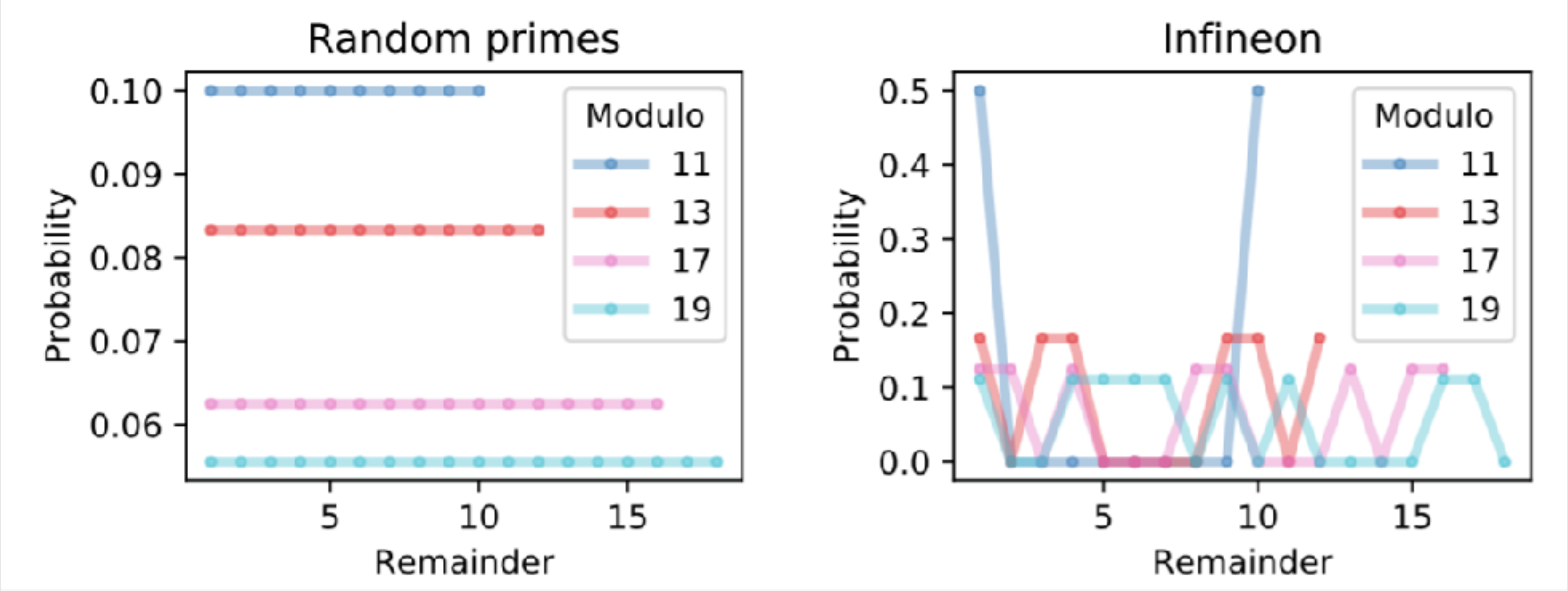
- If p and q are random primes, then (p-1)(q-1) may not be coprime with e, and N=pq will not have a fixed bit-length.

- Each manufacturer/library typically has their own distribution.



EX: INFINEON

- Infineon primes are « not random »



WHAT IS GOING ON?

- If p_i is a small prime then p mod p_i is not uniform over $\{1,\ldots, p_i-1\}$.

- It seems to be uniform over some small subgroup of $(\mathbf{Z}/p_i\mathbf{Z})^*$.

WHY?

- Typically, one generates primes as:
  - Repeat
    - Generate a large random number p
  - Until p is prime
- In practice, primality testing is a few modular exponentiations. One can increase the probability by making p not divisible by all small p_i.

GENERATION

- The subgroup of $(\mathbf{Z}/p_i\mathbf{Z})^*$ is the one generated by 65537.

- p and q are of the form:
  - $p=kM+(65537^a \bmod M)$, where M is the product of the first n primes: 2x3x5x…
  - n depends on the size of N.

- Hence, N mod M is a power of 65537, which can easily be checked: can we factor such N?

VALUES OF M

- M is the product of the first n primes.

Bit-length(N)	Number of primes
512-960	39
992-1952	71
1984-3936	126
3968-4096	225

BREAKING INFINEON-RSA

- $p = kM + (65537^a \bmod M)$
- If one can guess the exponent a, then $p \bmod M$ is known.
- From Coppersmith's 1996 work: if $M \geq N^{0.25}$, lattice attacks recover p in poly-time from N.

N	512-bit	1024-bit	2048-bit	3072-bit	4096-bit
$(\log_2 M)/(\log_2 N)$	0.43	0.46	0.47	0.32	0.48

LATTICE ATTACKS

- If p mod M is known, one knows a linear polynomial $f(X)\in\mathbf{Z}[X]$ s.t. $\gcd(f(x_0),N)=p$ is large, where x_0 is a small integer: it is small if M is large.
- This can be solved by lattice techniques [Cop1996].

THE TRICK

- Guessing a depends on the order of 65537 in $(\mathbf{Z}/M\mathbf{Z})^*$, which might be as big as $M \geq N^{0.4}$: exhaustive search too expensive!

- However, no need to take M: take any divisor M′ of M s.t. $M' \geq N^{1/4}$ and the order of 65537 in $(\mathbf{Z}/M'\mathbf{Z})^*$ is small.

- Ex: 20-bit order for 512-bit N, 30-bit order for 1024-bit.

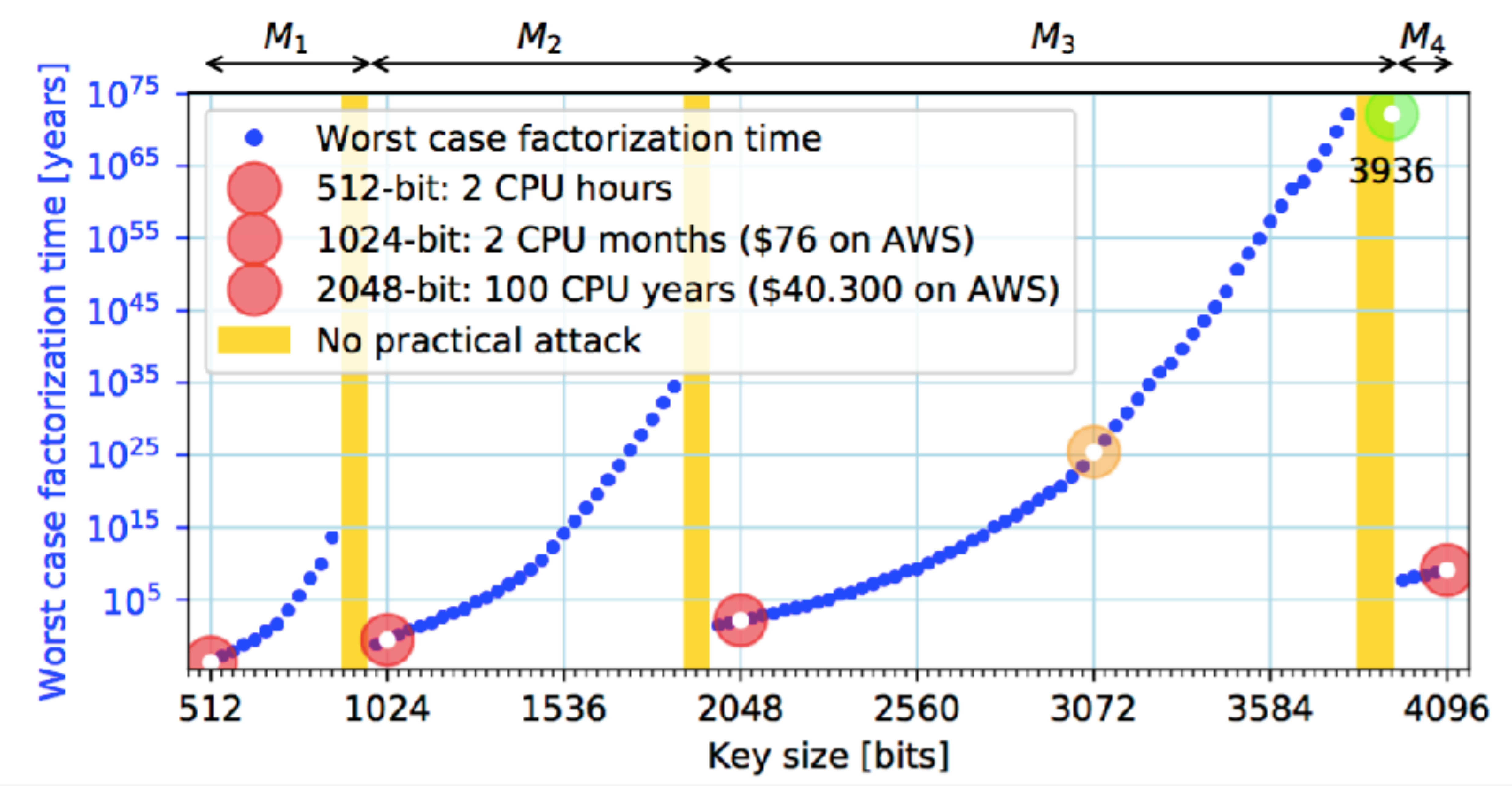
EXPLANATION

- M is the product of the first n primes p_i.

- We search for a subset I of {1,…,n} s.t. $M' = \prod_{i \in I} p_i$
    - $M' \geq N^{1/4}$
    - $\mathrm{ord}_{M'}(65537) = \mathrm{lcm}_{i \in I} \mathrm{ord}_{p_i}(65537)$ is minimized
  - The underlying optimization problem is NP hard, but we just need to find a solution.

IMPLEMENTATION

- Non-increasing



Finding small roots of polynomial equations using LLL

Recall that using $\varepsilon = 1/4$, given as input a basis of an integer lattice L of rank d, the LLL algorithm outputs in polynomial time a non-zero vector $\vec{u} \in L$ such that $\|\vec{u}\| \leq 2^{(d-1)/4}\mathrm{vol}(L)^{1/d}$.

1. Coppersmith's Theorem. (⋆⋆)

Let $P(x) \in \mathbb{Z}[x]$ be a monic polynomial of degree δ : the coefficient of its x^δ monomial is 1. Let N be a positive integer, whose factorization is unknown. We say that $Q(x) \in \mathbb{Q}[x]$ is (N, P)-good if for every integer $x_0 \in \mathbb{Z}$ such that $P(x_0) \equiv 0 \pmod{N}$, we have $Q(x_0) \in \mathbb{Z}$. If $Q(x) = \sum_{i=0}^{d} q_i x^i \in \mathbb{Q}[x]$, we define $\|Q\| = (\sum_{i=0} q_i^2)^{1/2}$. Let $X > 0$.

1. *Assume that $Q(x) \in \mathbb{Q}[x]$ is (N, P)-good and that $\|Q(xX)\| < 1/\sqrt{n+1}$ where n is the degree of Q. Show that if $P(x_0) \equiv 0 \pmod{N}$ and $|x_0| \leq X$, then $Q(x_0) = 0$.*
2. *For any integers $u, v \geq 0$, define $Q_{u,v}(x) = x^u(P(x)/N)^v$. Show that any integral linear combinations of polynomials $Q_{u,v}(x)$ is (N, P)-good.*

3. *Given as input N and $P(x)$, show that one can find in polynomial time a non-zero (N,P)-good polynomial $Q(x) \in \mathbb{Q}[x]$ such that $Q(x)$ is an integral linear combination of $Q_{0,0}(x), Q_{1,0}(x), \ldots, Q_{\delta-1,0}(x), Q_{0,1}(x)$ and*

$$\|Q(xX)\| \leq 2^{\delta/4} X^{\delta/2} N^{-1/(\delta+1)}.$$

4. *Deduce Håstad's theorem : one can find in polynomial time all the integers $x_0 \in \mathbb{Z}$ such that $|x_0| \leq N^{2/(\delta(\delta+1))}$ and $P(x_0) \equiv 0 \pmod N$.*

5. *Using the polynomials* $Q_{u,v}(x)$ *where* $0 \leq u \leq \delta - 1$ *and* $0 \leq v \leq h$ *for some well-chosen integer* h*, show Coppersmith's theorem : one can find in polynomial time all the integers* $x_0 \in \mathbb{Z}$ *such that* $|x_0| \leq N^{1/\delta}$ *and* $P(x_0) \equiv 0 \pmod{N}$.

6. *What can we do if* $P(x)$ *is not monic ?*

7. *If we want to find all roots* x_0 *such that* $|x_0| \leq C \times N^{1/\delta}$ *for some* $C > 1$*, what can we do ?*

2. The GCD generalization.

$(\star\star\star)$

We take the same notation. Let $\alpha \in \mathbb{Q}$ such that $0 < \alpha \leq 1$. We want to find all $x_0 \in \mathbb{Z}$ such that $\gcd(P(x_0), N) \geq N^{\alpha}$.

---

1. *Consider an integral linear combination $Q(x) \in \mathbb{Q}[x]$ of the $h\delta$ polynomials $Q_{u,v}(x)$ where $0 \leq u \leq \delta - 1$ and $0 \leq v \leq h$ for some well-chosen integer h. Show that if $x_0 \in \mathbb{Z}$ and $\gcd(P(x_0), N) \geq N^{\alpha}$ then the rational $Q(x_0)$ has a denominator $\leq N^{(1-\alpha)h}$.*

2. *Deduce that one can find in polynomial time all the integers $x_0 \in \mathbb{Z}$ such that $\gcd(P(x_0), N) \geq N^{\alpha}$ and $|x_0| \leq N^{\alpha^2/\delta}$.*