Implementation Attacks

What you should get out from engaging with this content

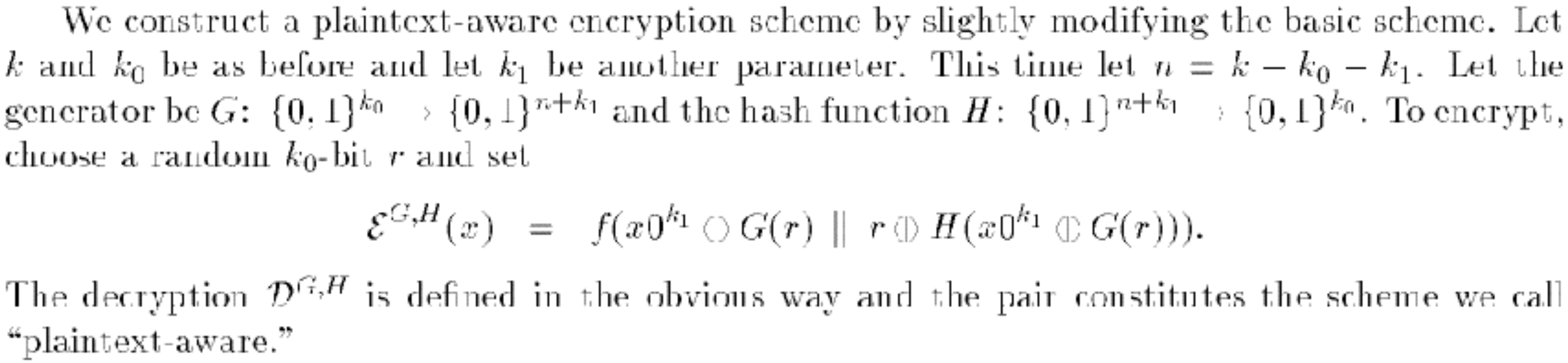
A sense that:

- Implementing cryptography requires leaving the safe haven of theoretical cryptography

- An implementation is unlikely to have the same security as the associated mathematical scheme

- The multiple layers of abstraction encountered from translating a mathematical description into code create opportunities to get things wrong

Crypto implementations start off as mathematical objects with abstract definitions and rigorously defined properties.

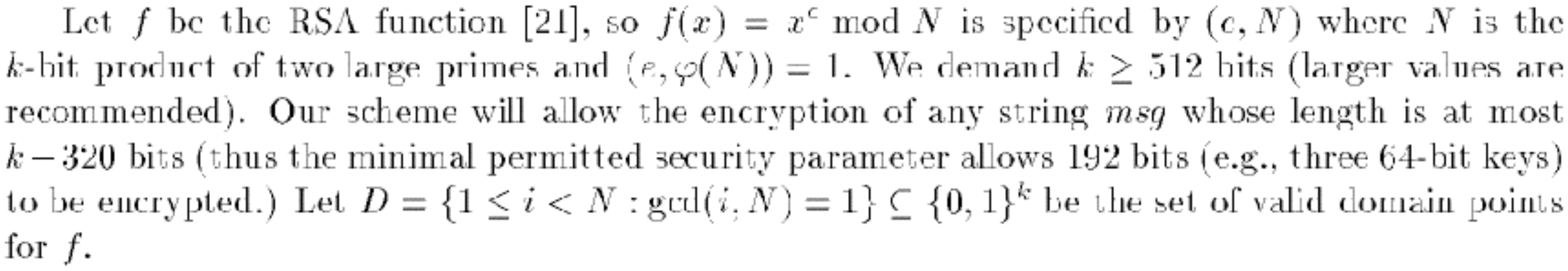


We construct a plaintext-aware encryption scheme by slightly modifying the basic scheme. Let k and k_0 be as before and let k_1 be another parameter. This time let $n = k - k_0 - k_1$. Let the generator be $G\colon \{0,1\}^{k_0} \to \{0,1\}^{n+k_1}$ and the hash function $H\colon \{0,1\}^{n+k_1} \to \{0,1\}^{k_0}$. To encrypt, choose a random k_0-bit r and set

$$\mathcal{E}^{G,H}(x) \;=\; f(x0^{k_1} \oplus G(r) \;\|\; r \oplus H(x0^{k_1} \oplus G(r))).$$

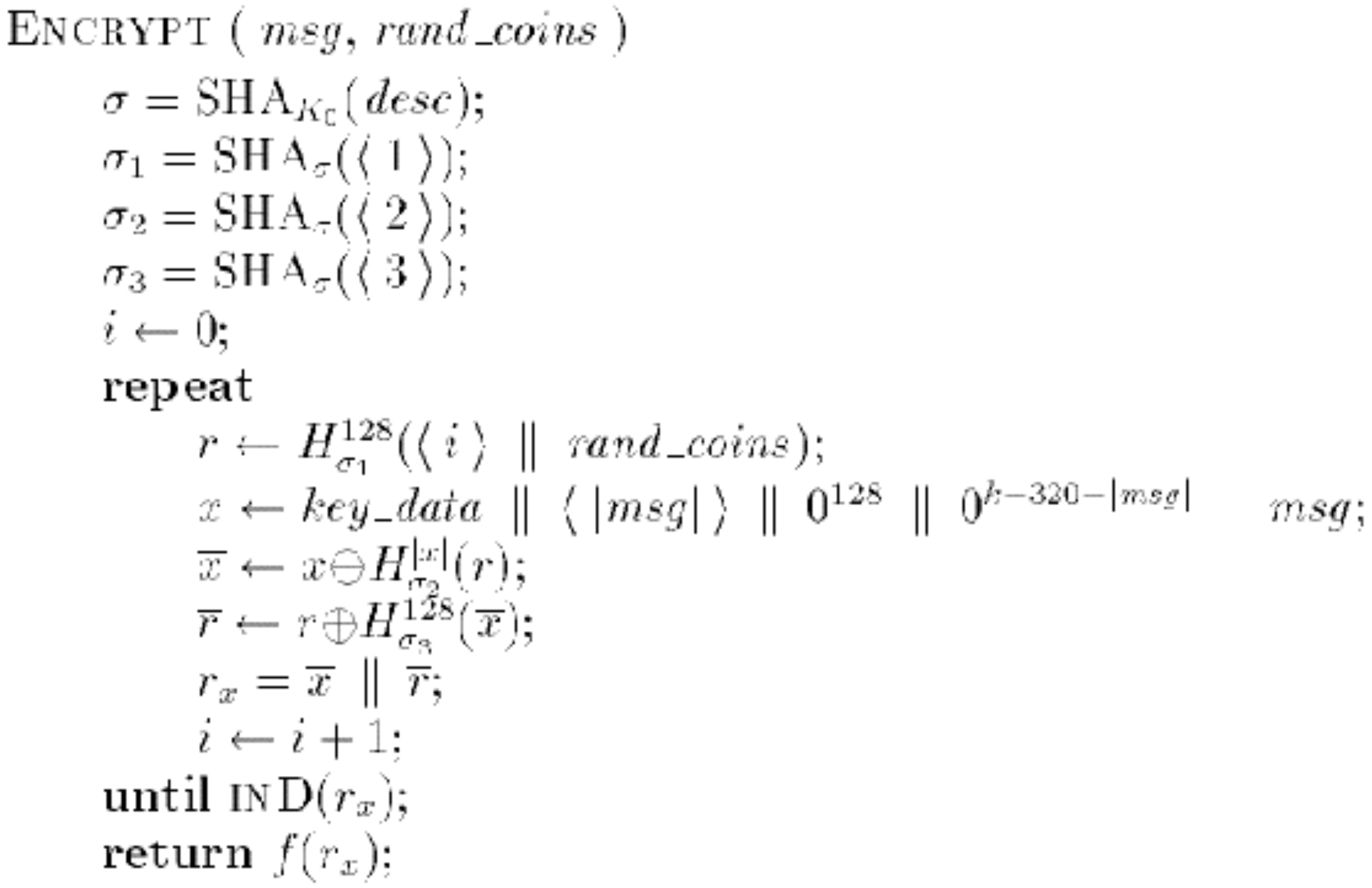
The decryption $\mathcal{D}^{G,H}$ is defined in the obvious way and the pair constitutes the scheme we call "plaintext-aware."

But for many of the objects in definitions and proofs we don't know if they exist. Thus we cannot instantiate these objects as such, and instead need to use 'a best guess' as to what concrete objects resemble their properties best.



Let f be the RSA function [21], so $f(x) = x^e \bmod N$ is specified by (e, N) where N is the k-bit product of two large primes and $(e, \varphi(N)) = 1$. We demand $k \geq 512$ bits (larger values are recommended). Our scheme will allow the encryption of any string *msg* whose length is at most $k - 320$ bits (thus the minimal permitted security parameter allows 192 bits (e.g., three 64-bit keys) to be encrypted.) Let $D = \{1 \leq i < N : \gcd(i, N) = 1\} \subseteq \{0,1\}^k$ be the set of valid domain points for f.

The instantiation with concrete objects then needs to be translated into some algorithmic form. Hereby data formats need to be unambiguous, key sizes need to be determined, to get as close as possible to a concrete, concise, and still correct description of the original idea/construction.

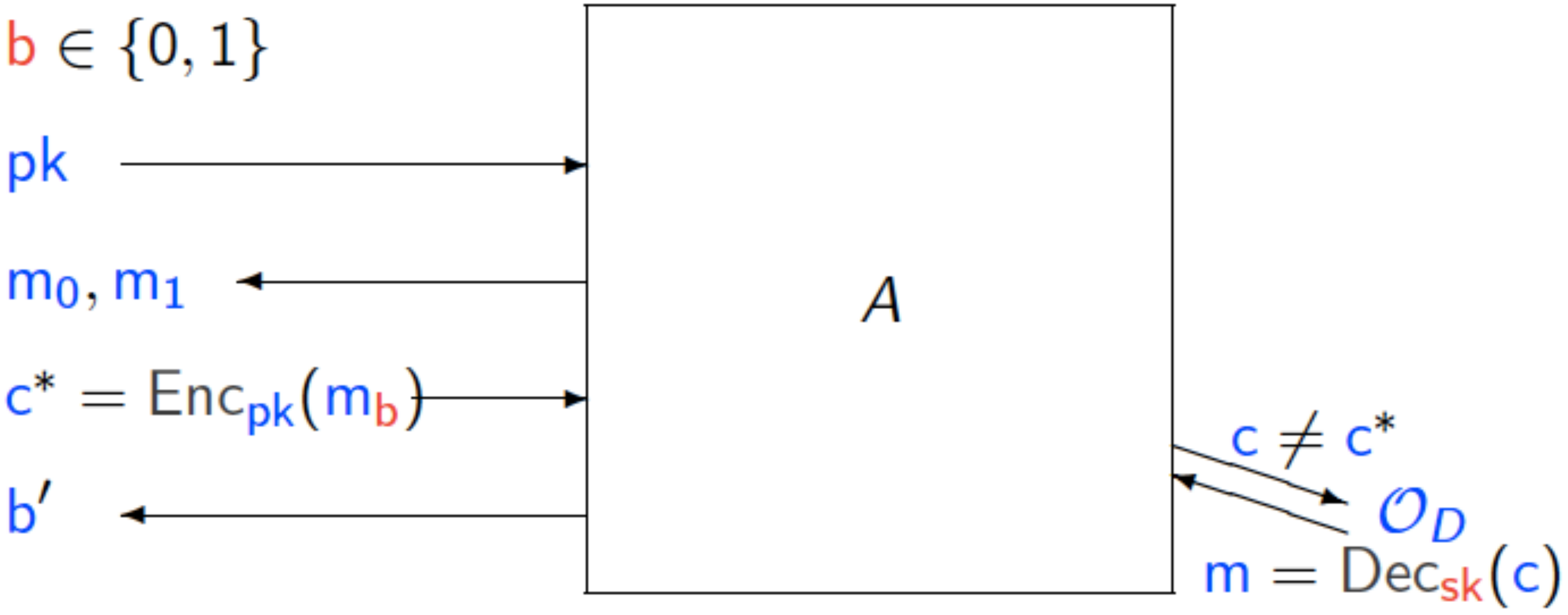


```
ENCRYPT ( $msg$, $rand\_coins$ )
    $\sigma = \mathrm{SHA}_{K_C}(desc)$;
    $\sigma_1 = \mathrm{SHA}_\sigma(\langle 1 \rangle)$;
    $\sigma_2 = \mathrm{SHA}_\sigma(\langle 2 \rangle)$;
    $\sigma_3 = \mathrm{SHA}_\sigma(\langle 3 \rangle)$;
    $i \leftarrow 0$;
    repeat
        $r \leftarrow H^{128}_{\sigma_1}(\langle i \rangle \ \| \ rand\_coins)$;
        $x \leftarrow key\_data \ \| \ \langle |msg| \rangle \ \| \ 0^{128} \ \| \ 0^{k-320-|msg|} \qquad msg$;
        $\overline{x} \leftarrow x \ominus H^{|x|}_{\sigma_2}(r)$;
        $\overline{r} \leftarrow r \oplus H^{128}_{\sigma_3}(\overline{x})$;
        $r_x = \overline{x} \ \| \ \overline{r}$;
        $i \leftarrow i + 1$;
    until IND$(r_x)$;
    return $f(r_x)$;
```

Figure 1: *A sample instantiation of the plaintext-aware encryption scheme.*

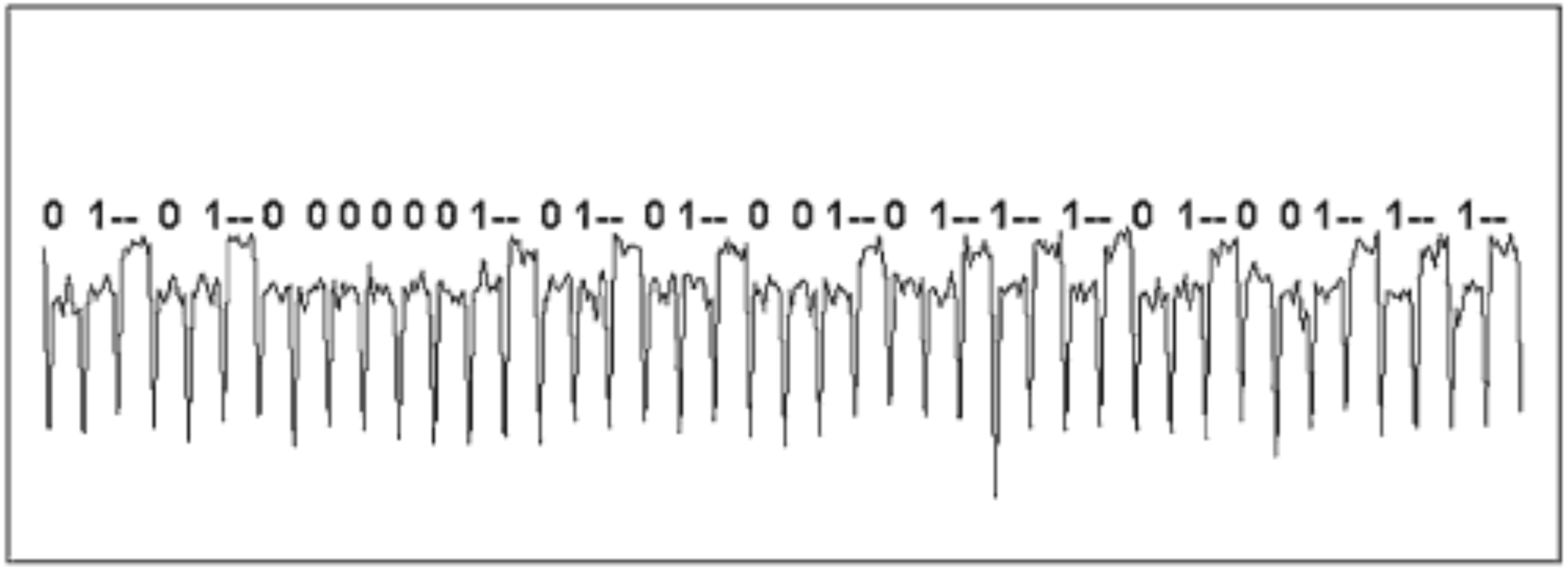
Text snippets taken from:
Optimal Asymmetric Encryption —How to Encrypt with RSA,
Bellare and Rogaway, 1994

Security definition/Threat model in theory: an adversary behaves in a precisely defined way. Adversary interacts with crypto primitive only via defined interfaces.



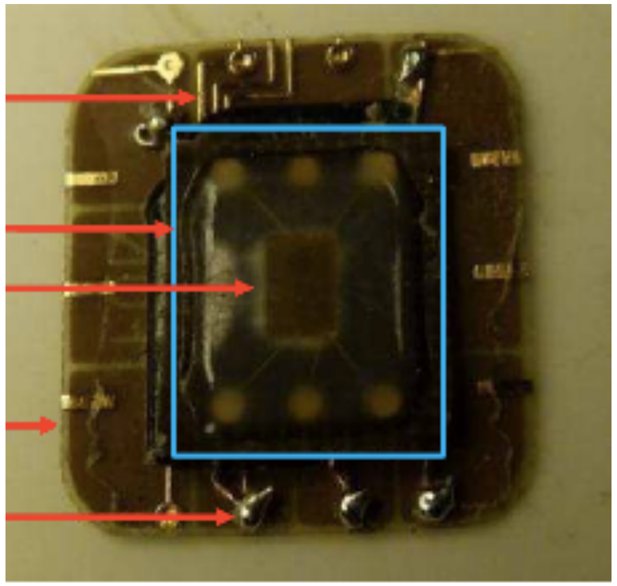
Picture from Crypto A (Bristol) unit

But with some equipment, and ignoring the rules of the security game an adversary can capture power or EM traces such as below.

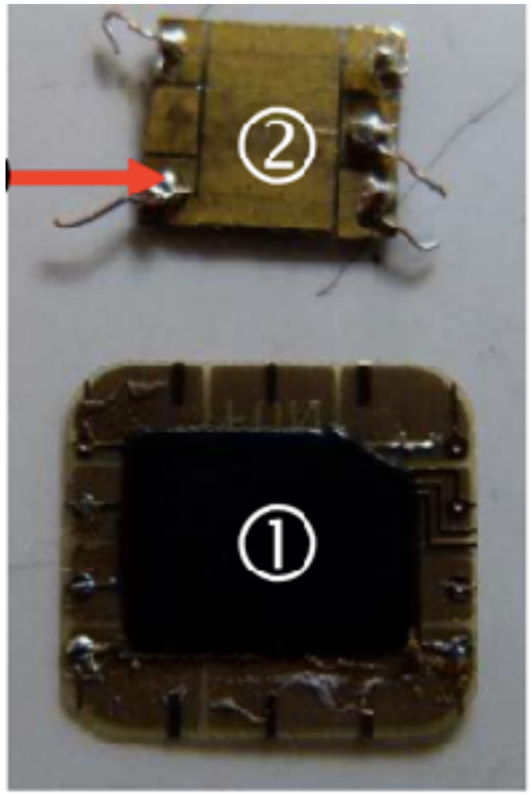


Source: https://www.rambus.com/introduction-to-differential-power-analysis/

Practical attacks: adversaries always choose the most rewarding attack vector and go for the most desirable target. This may be key recovery (thereby leading to a full break of the system), or, like in the attack from which I took the picture below, they might emulate a ‘seemingly theoretical scenario’ like a man in the middle attack.



Microchip piggy backing on another microchip



Microchips separated

Check out:
When Organized Crime Applies Academic Results
A Forensic Analysis of an In-Card Listening Device,
Ferrari et al., 2015

Some (in)famous attacks…

- 1943: NSA becomes aware of ‘compromising’ (electromagnetic) emanations from encryption machines (Tempest) (source: https://www.nsa.gov/Portals/70/documents/news-features/declassified-documents/cryptologic-spectrum/tempest.pdf)

- 1965: MI5 utilises click sounds from Rotor machines as additional information to aid decryption (source: Spy Catcher, Peter Wright)

- 2010: NSA use power and EM analysis to extract keys from TPMs (Trusted Platform modules), which are an integral component for secure boot as well as disk encryption (source: https://theintercept.com/document/2015/03/10/tpm-vulnerabilities-power-analysis-exposed-exploit-bitlocker/)

… some more …

- CRIME: **C**ompression **R**at**i**o Attacks **M**ade **E**asy (https://en.wikipedia.org/wiki/CRIME)

- BREACH: **B**rowser **R**econnaissance and **E**xfiltration via **A**daptive **C**ompression of **H**ypertext (https://en.wikipedia.org/wiki/BREACH)

- Meltdown, Spectre, Foreshadow … (all documented on Wikipedia as well)

If you have fancy equipment then you can do this:

https://www.youtube.com/watch?v=tnY7UVyaFiQ&feature=youtu.be

LiveSlides web content

To view

Download the add-in.

liveslides.com/download

Start the presentation.

Summary

- Cryptographic algorithms are based on mathematical functions for which one can prove certain properties in well defined models.

- Adversaries do not care about what they are `allowed’ to do in those models, they will in fact actively seek to circumvent those restrictions.

- Executing any algorithm changes the `state’ of a computing device: depending on the algorithm and the input data, power/EM varies, timings vary, the state of the cache changes, sounds may arise, etc.

- `Encouraging’ and utilising such extra information leads to incredibly powerful attacks, which we will study in this unit.