Formally Bounding the Side-Channel Leakage in Unknown-Message Attacks

Michael Backes[1,2] and Boris Köpf[2]
backes@cs.uni-sb.de bkoepf@mpi-sws.mpg.de

[1] Saarland University
[2] MPI-SWS

Abstract. We propose a novel approach for quantifying a system's resistance to unknown-message side-channel attacks. The approach is based on a measure of the secret information that an attacker can extract from a system from a given number of side-channel measurements. We provide an algorithm to compute this measure, and we use it to analyze the resistance of hardware implementations of cryptographic algorithms with respect to timing attacks. In particular, we show that message-blinding – the common countermeasure against timing attacks – reduces the rate at which information about the secret is leaked, but that the complete information is still eventually revealed. Finally, we compare information measures corresponding to unknown-message, known-message, and chosen-message attackers and show that they form a strict hierarchy.

1 Introduction

Side-channel attacks against cryptographic algorithms aim at breaking cryptography by exploiting information that is revealed by the algorithm's physical execution. Characteristics such as running time [12, 4, 22], power consumption [13], and electromagnetic radiation [11, 24] have all been exploited to recover secret keys from implementations of different cryptographic algorithms. Side-channel attacks are now so effective that they pose a real threat to the security of devices when their physical characteristics can be measured. This threat is not covered by traditional notions of cryptographic security; however, there is a line of research that investigates alternative models for reasoning about the resistance to such attacks [6, 21, 27, 14].

Two quantities determine the effort to successfully mount a side-channel attack and recover a secret key from a given system. The first is the computational power needed to recover the key from the information that is revealed through the side-channel. The second is the number of measurements needed to gather sufficient side-channel information for this task. To prove that a system is resistant to side-channel attacks, one must ensure that the overall effort for a successful attack is out of the range of realistic attackers.

The attacker's computational power is typically not the limiting factor in practice, as many documented attacks show [4, 7, 13, 22]. Hence, the security of a system often entirely depends on the amount of secret information that an attacker can gather in his side-channel measurements. Note that the number of measurements may be bounded – for example, by the number of times the system re-uses a session key – and must be considered when reasoning about a system's vulnerability to side-channel attacks.

A model to express the revealed information as a function of the number of side-channel measurements has recently been proposed, and it has been applied to characterize the resistance of cryptographic algorithms against side-channel attacks [14]. The model captures attackers that can interact with the system by adaptively choosing the messages that the system decrypts (or encrypts).

However, many attack scenarios only allow for *unknown-message* attacks, where the attacker cannot see or control the input that is decrypted (or encrypted) by the system. One type of unknown-message attack is timing attacks against systems that are run with state-of-the-art countermeasures such as message blinding. Quantifying the information that a side-channel reveals in such an attack was an open problem prior to this work.

1.1 Our Contributions

We propose a novel measure for quantifying the resistance of systems against unknown-message side-channel attacks. This measure Λ captures the quantity of secret information that a system reveals as a function of the number of side-channel measurements. Moreover, we provide an explicit formula for Λ when the number of measurements tends to infinity, corresponding to the maximum amount of secret information that is eventually leaked.

In order to apply our measure to realistic settings, we provide algorithms for computing Λ for finite and infinite numbers of measurements, respectively. We subsequently use these algorithms to formally analyze the resistance of a nontrivial hardware implementation to side-channel attacks: we show that a finite-field exponentiation algorithm as used in, e.g., the generalized ElGamal decryption algorithm, falls prey to unknown-message timing attacks in that the key is fully determined by a sufficiently large numbers of measurements.

We use this result to analyze message-blinding, which aims at protecting against timing attacks by decoupling the running time of the algorithm from the secret. We show that, for the analyzed exponentiation algorithm, message-blinding only reduces the rate at which information about the secret is revealed, and that the entire key information is still eventually leaked. This yields the first formal assessment of the (un-)suitability of message-blinding to counter timing attacks.

We conclude by putting our measure Λ into perspective with information measures for different kinds of attacker interactions. The result is a formal hierarchy of side-channel attackers that is ordered in terms of the information they can extract from a system. We distinguish unknown-message attacks, in which the attacker does not even know the messages (as in timing attacks against implementations with message blinding), known-message attacks, in which the attacker knows but cannot influence the messages, and chosen-message attacks, in which the attacker can adaptively choose the messages (as is typically the case in timing attacks against unprotected implementations). As expected, more comprehensive attackers are capable of extracting more information in a given number of measurements. Moreover, we show that this inclusion is strict for certain side-channels. Clarifying the different attack scenarios will provide guidance on which measure to pick for a particular application scenario.

1.2 Outline

The paper is structured as follows. In Section 2, we introduce our models of side-channels and attackers and we review basics of information theory. In Section 3, we present measures for quantifying the information leakage in unknown-message attacks. In Section 4, we show how these measures can be computed for given implementations. We report on experimental results in Section 5 and compare different kinds of side-channel attacks in Section 6. We discuss related work in Section 7 and conclude in Section 8.

2 Preliminaries

We start by describing our models of side-channels and attackers, and we briefly recall some basic information theory.

2.1 Modeling Side-Channels and Attackers

Let K be a finite set of keys, M be a finite set of messages and D be an arbitrary set. We consider systems that compute functions of type $F\colon K \times M \to D$, and we assume that the attacker can make physical observations about F's implementation I_F that are associated with the computation of $F(k, m)$. We assume that the attacker can make one observation per invocation of the function F and that no measurement errors occur. Examples of such observations are the power or the time consumption of I_F during the computation (see [13, 20] and [12, 4, 22], respectively).

Formally, a *side-channel* is a function $f_{I_F}\colon K \times M \to O$, where O denotes the set of possible observations. We assume that the attacker has full knowledge about the implementation I_F, i.e., f_{I_F} is known to the attacker. We will usually leave I_F implicit and abbreviate f_{I_F} by f.

Example 1. Suppose that F is implemented in synchronous (clocked) hardware and that the attacker is able to determine I_F's running times up to single clock ticks. Then the timing side-channel of I_F can be modeled as a function $f\colon K \times M \to \mathbb{N}$ that represents the number of clock ticks consumed by an invocation of F. A hardware simulation environment can be used to compute f.

Example 2. Suppose F is given in a description language for synchronous hardware. Power estimation techniques such as can be used to determine a function $f\colon K \times M \to \mathbb{R}^n$ that estimates an implementation's power consumption during n points in time (see, e.g., [17] and Section 5.3).

In a side-channel attack, a malicious agent gathers side-channel observations $f(k, m_1), \ldots, f(k, m_n)$ for deducing k or narrowing down its possible values. Depending on the attack scenario, the attacker might additionally be able to see or choose the messages $m_i \in M$: an attack is *unknown-message* if the attacker cannot observe $m_i \in M$; an attack is *known-message* if the attacker can observe but cannot influence the choice of $m_i \in M$; an attack is *chosen-message* if the attacker can choose $m_i \in M$.

In this paper, we focus on the open problem of giving bounds on the side-channel leakage in unknown-message attacks. In Section 6, we will come back to the distinction between different attack types and formally compare them with respect to the quantity of information that they can extract from a system.

2.2 Information Theory Basics

Let A be a finite set and $p \colon A \to \mathbb{R}$ a probability distribution. For a random variable $\mathcal{X} \colon A \to X$, we define $p_{\mathcal{X}} \colon X \to \mathbb{R}$ as $p_{\mathcal{X}}(x) = \sum_{a \in \mathcal{X}^{-1}(x)} p(a)$, which is often denoted by $p(\mathcal{X} = x)$ in the literature.

The *(Shannon) entropy* of a random variable $\mathcal{X} \colon A \to X$ is defined as

$$H(\mathcal{X}) = -\sum_{x \in X} p_{\mathcal{X}}(x) \log_2 p_{\mathcal{X}}(x) \ .$$

The entropy is a lower bound for the average code length of any binary encoding scheme for $\mathcal{X}$. An encoding scheme can be seen as a strategy in which each bit corresponds to a binary test that narrows down the set of the remaining candidate values. Thus, in terms of guessing, the entropy $H(\mathcal{X})$ is a lower bound for the average number of binary questions that need to be asked to determine $\mathcal{X}$'s value [5]. If $\mathcal{Y} \colon A \to Y$ is another random variable, $H(\mathcal{X}|\mathcal{Y} = y)$ denotes the entropy of $\mathcal{X}$ given $\mathcal{Y} = y$, i.e., with respect to the distribution $p_{\mathcal{X}|\mathcal{Y}=y}$. The *conditional entropy* $H(\mathcal{X}|\mathcal{Y})$ of $\mathcal{X}$ given $\mathcal{Y}$ is defined as the expected value of $H(\mathcal{X}|\mathcal{Y} = y)$ over all $y \in Y$, namely,

$$H(\mathcal{X}|\mathcal{Y}) = \sum_{y \in Y} p_{\mathcal{Y}}(y) H(\mathcal{X}|\mathcal{Y} = y) \ .$$

Entropy and conditional entropy are related by the equation $H(\mathcal{X}\mathcal{Y}) = H(\mathcal{Y}) + H(\mathcal{X}|\mathcal{Y})$, where $\mathcal{X}\mathcal{Y}$ is the random variable defined as $\mathcal{X}\mathcal{Y}(k) = (\mathcal{X}(k), \mathcal{Y}(k))$. *The mutual information* $I(\mathcal{X}; \mathcal{Y})$ of $\mathcal{X}$ and $\mathcal{Y}$ is defined as the reduction of uncertainty about $\mathcal{X}$ if one learns $\mathcal{Y}$, i.e., $I(\mathcal{X}; \mathcal{Y}) = H(\mathcal{X}) - H(\mathcal{X}|\mathcal{Y})$.

3 Information Leakage in Unknown-Message Attacks

In this section, we first propose a novel measure that expresses the information gain of an unknown-message attacker as a function of the number of side-channel observations made. Subsequently, we derive an explicit representation for the limit of this information gain for an unbounded number of observations. This representation provides a characterization of the secret information that the side-channel eventually leaks. Moreover, it leads to a simple algorithm for computing this information.

3.1 Information Gain in n Observations

In the following, let $p_K \colon K \to \mathbb{R}$ and $p_M \colon M \to \mathbb{R}$ be probability distributions and let the random variables $\mathcal{K} = id_K$, $\mathcal{M} = id_M$ model the random choice of keys and messages, respectively; we assume that p_M and p_K are known to the attacker. For $n \in N$, let $\mathcal{O}_n \colon K \times M^n \to O^n$ be defined by $\mathcal{O}_n(k, m_1, \ldots, m_n) = (f(k, m_1), \ldots, f(k, m_n))$, where $p_{KM^n}(k, m_1, \ldots, m_n) = p_K(k) p_M(m_1) \ldots p_M(m_n)$ is the probability distribution on $K \times M^n$. The variable $\mathcal{O}_n$ captures that k remains fixed over all invocations of f, while the messages $m_1, \ldots, m_n$ are chosen independently.

An unknown-message attacker making n side-channel observations $\mathcal{O}_n$ may learn information about the value of $\mathcal{K}$, i.e., about the secret key. This information can be expressed as the reduction in uncertainty about the value of $\mathcal{K}$, i.e., $I(\mathcal{K}; \mathcal{O}_n) = H(\mathcal{K}) - H(\mathcal{K}|\mathcal{O}_n)$. An alternative is to use the attacker's remaining uncertainty about the key $H(\mathcal{K}|\mathcal{O}_n)$ as a measure for quantifying the system's resistance to an attack. Focusing on $H(\mathcal{K}|\mathcal{O}_n)$ has the advantage of a precise interpretation in terms of guessing: it is a lower bound on the average number of binary questions that the attacker still needs to ask to determine $\mathcal{K}$'s value [5].

Definition 1. *We define $\Lambda(n) = H(\mathcal{K}|\mathcal{O}_n)$ as the* resistance to unknown-message attacks *of n steps.*

Two measures that are closely related to Λ have been proposed in [8] and [27]. The measure from [8] captures only single measurements, i.e., it corresponds to $\Lambda(1)$. The information-theoretic metric from [27] captures multiple measurements, but with respect to stronger, chosen-message adversaries.

The function Λ is monotonically decreasing, i.e., more observations can only reduce the attacker's uncertainty about the key. If $\Lambda(n) = H(\mathcal{K})$, the first n side-channel observations contain no information about the key. Clearly, $\Lambda(0) = H(\mathcal{K})$. If $\Lambda(n) = 0$, the key is completely determined by n side-channel observations.

Since $\Lambda(n)$ is defined as the expected value of $H(\mathcal{K}|\mathcal{O}_n = o)$ over all $o \in O^n$, it expresses whether keys are, on the average, hard to determine after n side-channel observations. It is straightforward to adapt the resistance to accommodate worst-case guarantees [14] or to use alternative notions of entropy that correspond to different kinds of guessing [5]. For example, by using the guessing entropy instead of the Shannon entropy, one can express the remaining uncertainty about the key in terms of the average number of questions of the kind "does $\mathcal{K} = k$ hold" that must be asked to guess $\mathcal{K}$'s value correctly [18].

In Section 4, we will give an algorithm for computing the resistance $\Lambda(n)$ to unknown-message attacks. The time complexity of this algorithm is, however, exponential in n, rendering computation for large values of n infeasible. To remedy this problem, we will now establish an explicit formula for $\lim_{n\to\infty} \Lambda(n)$, which will allow us to compute limits for the resistance without being faced with the exponential increase in n.

3.2 Bounds for Unlimited Observations

The core idea for computing the limit of Λ can be described as follows: for a large number $o_1, \dots, o_n$ of side-channel observations and a fixed key k, the relative frequency of each $o \in O$ converges to the probability $p_{\mathcal{O}|\mathcal{K}=k}(o)$. Thus, making an unbounded number of observations corresponds to learning the distribution $p_{\mathcal{O}|\mathcal{K}=k}$. We next give a formal account of this idea.[3]

Define $k_1 \equiv k_2$ if and only if $p_{\mathcal{O}|\mathcal{K}=k_1} = p_{\mathcal{O}|\mathcal{K}=k_2}$. Then $\equiv$ constitutes an equivalence relation on K, and $K/_\equiv$ denotes the set of equivalence classes. The random variable $\mathcal{V}\colon K \to K/_\equiv$ defined by $\mathcal{V}(k) = [k]_\equiv$ maps every key to its $\equiv$-equivalence class. Knowledge of the value of $\mathcal{V}$ hence corresponds to knowledge of the distribution $p_{\mathcal{O}|\mathcal{K}=k}$ associated with k. Intuitively, an unbounded number of observations contains as much information about the key as the key's $\equiv$-equivalence class. This is formalized by the following theorem.

Theorem 1. *Let $\mathcal{K}, \mathcal{V}$ and $\mathcal{O}_n$ be defined as above. Then*

$$\lim_{n\to\infty} H(\mathcal{K}|\mathcal{O}_n) = H(\mathcal{K}|\mathcal{V}) \ . \tag{1}$$

The proof of Theorem 1 can be found in the full version of this paper [1]. A straightforward calculation shows that, for uniformly distributed keys, $H(\mathcal{K}|\mathcal{V}) = \frac{1}{|K|}\sum_{B\in K/_\equiv} |B| \log_2 |B|$. Consequently, Theorem 1 enables us to compute $\lim_{n\to\infty} H(\mathcal{K}|\mathcal{O}_n)$ from the sizes of the $\equiv$-equivalence classes. This is illustrated by the following example.

Example 3. Let $n \in \mathbb{N}$, $K = \{0,1\}^n$, $M = \{1, \dots, n\}$, and $O = \{0,1\}$. Consider the function $f\colon K \times M \to O$ defined by $f(k,m) = k_m$, where $k = (k_1, \dots, k_n)$. Theorem 1 implies that $H(\mathcal{K}|\mathcal{V})$ captures the information about k that f eventually leaks to an unknown-message attacker. For computing $H(\mathcal{K}|\mathcal{V})$, observe that for $k_1, k_2 \in K$, $p_{\mathcal{O}|\mathcal{K}=k_1} = p_{\mathcal{O}|\mathcal{K}=k_2}$ if and only if the number of 1-bits in k_1 and k_2 is equal, i.e., if k_1 and k_2 have the same Hamming weight. The number of n-bit values with Hamming weight h is given by $\binom{n}{h}$. Hence, $\lim_{n\to\infty} H(\mathcal{K}|\mathcal{O}_n) = \frac{1}{2^n}\sum_{h=0}^{n} \binom{n}{h} \log_2 \binom{n}{h}$.

4 Computing the Resistance to Unknown-Message Attacks

In this section, we show how $\Lambda(n)$ and $\lim_{n\to\infty} \Lambda(n)$ can be computed for given implementations I_F of cryptographic functions F. For this, we first need a representation of the side-channel $f = f_{I_F}$; second, we need to compute Λ from this representation.

[3] For probabilities, this is a consequence of the law of large numbers. We are not aware of a corresponding result for the conditional entropy.

4.1 Determining Time Consumption

We focus on implementations in synchronous (clocked) hardware and we assume that the attacker can determine the system's time consumption up to single clock ticks. We use the hardware design environment Gezel [25] for describing circuits and for building up value table representations of f. Here, the value $f(k,m)$ is the number of clock ticks consumed by the computation of $F(k,m)$ and can be determined by the simulation environment. Specifications in the Gezel language can be mapped into a synthesizeable subset of Vhdl, an industrial-strength hardware description language. The mapping preserves the circuit's timing behavior within the granularity of clock ticks. In this way, the guarantees obtained by formal analysis translate to silicon implementations.

We next show how $\mathit{\Lambda}(n)$ can be computed from the value table representation of f.

4.2 Computing $\mathit{\Lambda}(n)$

For computing $\mathit{\Lambda}(n)$ we first show how $\mathit{\Lambda}(n) = H(\mathcal{K}|\mathcal{O}_n)$ can be decomposed into a sum of terms of the form $p_{\mathcal{O}|\mathcal{K}=k}(o)$, with $k \in K$ and $o \in O$. Subsequently, we sketch how this decomposition can be used to derive a simple implementation for computing $\mathit{\Lambda}(n)$.

We have the following equalities

$$H(\mathcal{K}|\mathcal{O}_n) = -\sum_{o \in O^n} p_{\mathcal{O}_n}(o) \sum_{k \in K} p_{\mathcal{K}|\mathcal{O}_n=o}(k) \, \log_2 p_{\mathcal{K}|\mathcal{O}_n=o}(k) \tag{2}$$

$$p_{\mathcal{K}|\mathcal{O}_n=o}(k) = \frac{p_{\mathcal{O}_n|\mathcal{K}=k}(o) p_{\mathcal{K}}(k)}{p_{\mathcal{O}_n}(o)} \tag{3}$$

$$p_{\mathcal{O}_n}(o) = \sum_{k \in K} p_{\mathcal{O}_n|\mathcal{K}=k}(o) p_{\mathcal{K}}(k) \tag{4}$$

$$p_{\mathcal{O}_n|\mathcal{K}=k}(o_1, \ldots, o_n) = \prod_{i=1}^{n} p_{\mathcal{O}|\mathcal{K}=k}(o_i) \ , \tag{5}$$

where (3) is Bayes' formula and (5) holds because, for a fixed key, the observations are independent and identically distributed. Furthermore, for uniformly distributed messages, $p_{\mathcal{O}|\mathcal{K}=k}(o) = |\{m \mid f(k,m) = o)\}|/|M|$, which can be computed using the value table representation of f given by Gezel.

The decomposition in (2)-(5) of $H(\mathcal{K}|\mathcal{O}_n)$ into a combination of terms of the form $p_{\mathcal{O}|\mathcal{K}=k}(o)$ and $p_{\mathcal{K}}(k)$ for $k \in K$ and $o \in O$ can be expressed by list comprehensions. This is illustrated by the following code snippet in Haskell [3]. Here, `pO` computes $p_{\mathcal{O}_n}(o)$ according to (4) and (5) from a list of observations `obs`, a list representation `keys` of K, and an array `p` that stores the values $p_{\mathcal{O}|\mathcal{K}=k}(o)$:

```
pO obs = sum [ product [ p!(o,k) | o <- obs ]| k <- keys ]
            / length keys
```

The computation of $\varLambda(n)$ according to (2) and (3) can be encoded in a similarly concise way. We have implemented this in Haskell and use this implementation to perform experiments in Section 5.

4.3 Computing $\lim_{n\to\infty} \boldsymbol{\varLambda(n)}$

From Theorem 1 it follows that $\lim_{n\to\infty} \varLambda(n) = H(\mathcal{K}|\mathcal{V})$, where $\mathcal{V}(k) = [k]_{\equiv}$ and $k_1 \equiv k_2$ if and only if $p_{\mathcal{O}|\mathcal{K}=k_1} = p_{\mathcal{O}|\mathcal{K}=k_2}$. We have $H(\mathcal{K}|\mathcal{V}) = \frac{1}{|K|}\sum_{B\in K/\equiv} |B| \log_2 |B|$ for uniformly distributed keys. Hence, for computing $H(\mathcal{K}|\mathcal{V})$ it suffices to determine the sizes of the $\equiv$-equivalence classes.

The equivalence classes of an equivalence relation form a partition of the relation's domain. We compute the partition of K corresponding to $\equiv$ by refinement. For this, consider the equivalence relations $\equiv_o$ defined by $k_1 \equiv_o k_2$ if and only if $p_{\mathcal{O}|\mathcal{K}=k_1}(o) = p_{\mathcal{O}|\mathcal{K}=k_2}(o)$. Clearly, $k_1 \equiv k_2$ if and only if $\forall o \in O. k_1 \equiv_o k_2$.

For partitioning a set $B \subseteq K$ with respect to $\equiv_o$, group together all $k \in B$ with the same value of $p_{\mathcal{O}|\mathcal{K}=k}(o)$. For refining a given partition P of K with respect to $\equiv_o$, partition all $B \in P$ according to $\equiv_o$. Finally, for computing the partition corresponding to $\equiv$, successively refine the partition $\{K\}$ with respect to all $o \in O$. The following Haskell program implements this idea:

```
partKeys keys obs = foldr refineBy [keys] obs
  where refineBy o part = concat (map (splitBlockByObs o) part)
```

Here, the refinement of a block by an observation is accomplished by the function `splitBlockByObs`. The function `refineBy` applies this procedure to every block in a given partition. The function `partKeys` refines the partition `[keys]` by all observations in the list `obs`.

Finally, we can compute $H(\mathcal{K}|\mathcal{V}) = \frac{1}{|K|}\sum_{B\in K/\equiv} |B| \log_2 |B|$ from the partition `part` returned by `partKeys`:

```
entropy part = sum [ b * logBase 2 b | x <- bs ] / sum bs
  where bs = map length part
```

We use this simple prototype implementation in our experiments below.

5 Experimental Results

We now report on a case study where we analyze the implementation of a circuit for exponentiation in finite fields with respect to its resistance to timing attacks. Finite-field exponentiation is relevant, for example, in the generalized ElGamal encryption scheme [19]. Furthermore, we show how this result can be used for evaluating state-of-the-art countermeasures to timing attacks.

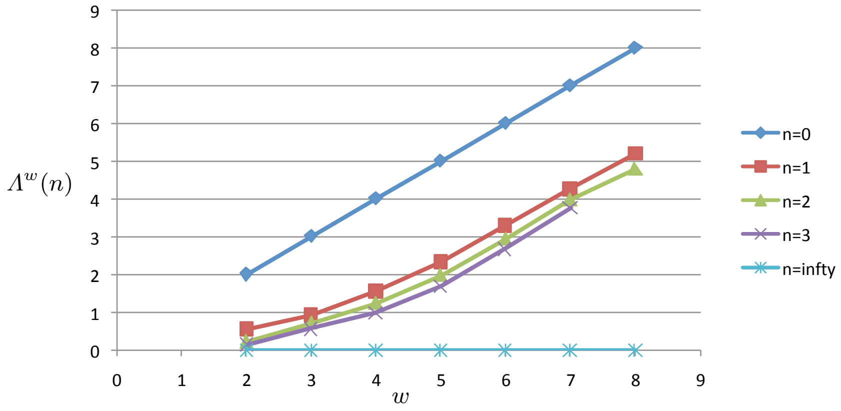

Fig. 1. Resistance of a finite-field exponentiation algorithm to unknown-message timing attacks

5.1 Timing Analysis of a Finite-Field Exponentiation Algorithm

We have analyzed a Gezel implementation of the finite-field exponentiation algorithm from [10]. It takes two arguments m and x and computes m^x in $\mathbb{F}_{2^w}$. The exponentiation is performed by square-and-multiply, where each multiplication corresponds to a multiplication of polynomials. The entire algorithm consists of three nested loops.

Computing $\Lambda(n)$ with the implementation presented in Section 4 is expensive and does not scale to large values of n and operands of large bit-widths. To overcome this problem, we use the following approximation technique: we parameterize each algorithm by the bit-width w of its operands. Our working assumption is that regularity in the values of Λ for $w \in \{2, \ldots, w_{\max}\}$ reflects the structural similarity of the algorithms. This permits the extrapolation to values of w beyond $w_{\max}$. To make this explicit, we will write Λ^w to denote that Λ is computed on w-bit operands.

Results of the Analysis The results of our analysis are given in Figure 1. The bit-width w of the operands is depicted along the horizontal axis and the entropy is depicted along the vertical axis. The different curves represent $\Lambda^w(n)$ for $n \in \{0, 1, 2, 3, \infty\}$.

We can draw the following conclusion from our data: the first timing observation reveals almost half of the secret information about the key. Subsequent observations reduce the uncertainty at a significantly slower rate. In the long run, however, the entire key information is leaked. Hence the circuit is vulnerable to unknown-message timing attacks.

5.2 Implications for the Security of Message-blinding

Timing attacks typically rely on the fact that the attacker can choose the input $m \in M$ and can measure the corresponding running time. Message-blinding, the state-of-the art countermeasure against timing attacks, renders this type of attack impractical by decoupling the algorithm's running time from m. Message-blinding has been proposed for exponentiation modulo n [12], but it can directly be applied to exponentiation in the field $\mathbb{F}_{2^w}$. We illustrate message-blinding for the common case of RSA.

Example 4. Consider an RSA decryption $x = m^k \mod n$, where m is chosen by the attacker, x the plaintext, n the modulus and k the secret key. Message-blinding decouples the running time of the exponentiation from m: in the *blinding* phase one computes $m \cdot r^e \mod n$, where r is random and relatively prime to n, and e is the public key. The result of the decryption is $(m \cdot r^e)^k = x \cdot r \mod n$, which yields x after *unblinding*, i.e., after multiplication with $r^{-1} \mod n$.

The belief that message-blinding is secure is based on the assumption that the blinding and unblinding steps do not introduce new side-channels, and that $m \cdot r^e$ is sufficiently random. Analyzing the resistance of an exponentiation algorithm with respect to unknown-message attackers and uniformly distributed messages thus corresponds to analyzing the implementation with idealized message-blinding and with respect to chosen-message attacker.

This correspondence enables us to use Λ for evaluating the quality of message-blinding as a countermeasure for timing attacks against the finite-field exponentiation circuit from Section 5.1. Figure 2 is based on data from [14] and depicts the resistance of the same exponentiation algorithm with respect to *chosen-message* attacks. Here, $\Phi^w(n)$ denotes the remaining uncertainty after n steps of a chosen-message attack. The value $\Lambda^w(n) - \Phi^w(n)$, i.e., the difference between the curves in Figures 1 and 2, gives a formal account of what is gained by applying message-blinding as a countermeasure, namely that the information is leaked at a significantly slower rate. Figure 1 shows that $\lim_{n\to\infty} \Lambda(n) = 0$. This implies that, even with message-blinding applied, the timing side-channel eventually leaks the entire key information. To our knowledge, this is the first formal analysis of a countermeasure against timing attacks.

5.3 On Formal Bounds for Power Analysis Attacks

Our measure Λ can also be applied to analyze the resistance of systems to power analysis attacks. As a proof of concept, we have applied our model to compute the resistance of a hardware implementation of an AES SBox with respect to power analysis attacks. The results can be found in the full version of this paper [1].

However, the formal bounds derived for power analysis attacks have to be carefully translated to real-world situations. First, power models typically abstract from certain electrical effects [17] so that formal bounds derived using such models (including ours) do not take into account attackers that exploit these elided effects. Second, in many attack scenarios, the attacker can observe the device's power consumption as a function of time. This function is typically approximated by the vector of the power measurements at n fixed time instants.

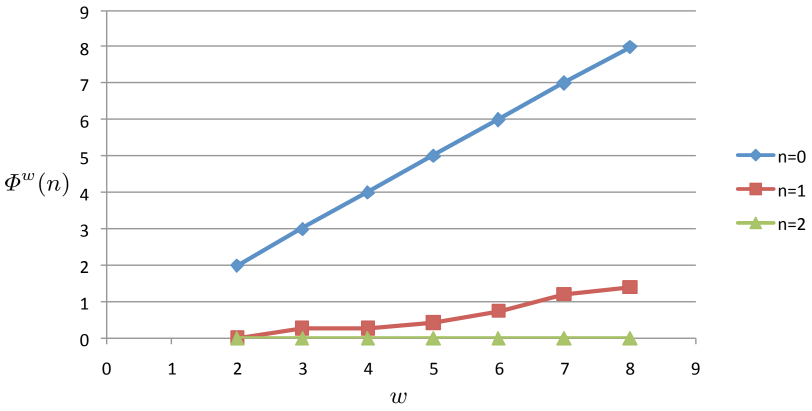

Fig. 2. Resistance of a finite-field exponentiation algorithm to *chosen-message* timing attacks

In our model, such an approximation can be captured by a side-channel of type $f\colon K \times M \to \mathbb{R}^n$. Bounds derived from this approximation do not take into account attackers that measure the power consumption at other points in time.

6 A Hierarchy of Side-Channel Attackers

In this section, we formally relate unknown-message, known-message and chosen-message attackers with respect to the information that they can extract from a given side-channel $f\colon K \times M \to O$. The main purpose of this comparison is a unified presentation that simplifies the task of picking the appropriate measure for a given attack scenario.

The result of the comparison is as expected: chosen-message attackers are stronger than known-message attackers, which are stronger than unknown-message attackers. All inclusions are shown to be strict. Before we formally state and prove this result, we begin with definitions of the resistance to known-message and chosen-message attacks.

6.1 Known-Message and Chosen-Message Attacks:

We define the resistance to known-message attacks along the lines of Definition 1, where we express that the attacker knows the messages by conditioning the entropy of $\mathcal{K}$ on $\mathcal{M}_n$. Here, $\mathcal{M}_n$ models the n independent choices of messages from M.

Definition 2. *We define* $\Delta(n) = H(\mathcal{K}|\mathcal{O}_n\mathcal{M}_n)$ *as the* resistance to known-message attacks *of n steps.*

Note that Δ is an average-case measure, as $H(\mathcal{K}|\mathcal{O}_n\mathcal{M}_n)$ is the expected remaining uncertainty about $\mathcal{K}$ if the values of $\mathcal{O}_n$ and $\mathcal{M}_n$ are known. It can be adapted to accommodate worst-case guarantees by replacing the expected value by the minimal value over all n-tuples of messages or observations.

A measure for the resistance to chosen-message attacks has been defined in [14]. We next give a short account of this definition. A chosen-message attack is formalized as a tree whose nodes are labeled with subsets of K. In this tree, an attack step is represented by a node v together with its children. The label A of v is the set of keys that could have led to the attacker's previous observations. The labels of the children of v form a partition of A. We require that this partition is of the form $\{A \cap f_m^{-1}(o) \mid o \in O\}$ for some $m \in M$, where $f_m(k) = f(k, m)$. This corresponds to the attacker's choice of a query m. By observing o, the attacker can narrow down the set of possible keys from A to $A' = f_m^{-1}(o) \cap A$. The child of v with label A' is the starting point for subsequent attack steps.

Definition 3 ([14]). *An* attack strategy *against f is a triple (T, r, L), where $T = (V, E)$ is a tree, $r \in V$ is the root, and $L \colon V \to 2^K$ is a node labeling with the following properties:*

1. *$L(r) = K$, and*
2. *for every $v \in V$, there is an $m \in M$ with $\{L(v) \cap f_m^{-1}(o) \mid o \in O\} = \{L(w) \mid (v, w) \in E\}$.*

An attack strategy is of length *l if T has height l.*

A simple consequence of requirements 1 and 2 is that the labels of the leaves of an attack strategy $\mathfrak{a} = (T, r, L)$ form a partition $P_{\mathfrak{a}} = \{L(v) \mid v \text{ is a leaf of } T\}$ (the *induced partition*) of K. We denote by $\mathcal{V}_{\mathfrak{a}}$ the random variable that maps $k \in K$ to its enclosing block in $P_{\mathfrak{a}}$.

Definition 4 ([14]). *We define $\Phi(n) = \min\{H(\mathcal{K}|\mathcal{V}_{\mathfrak{a}}) \mid \mathfrak{a} \text{ is of length } n\}$ as the* resistance to chosen-message attacks *of length n.*

6.2 Comparing Side-Channel Attackers

The following theorem gives a formal account of the intuition that more comprehensive attackers can extract more information from a system.

Theorem 2. *Let $f \colon K \times M \to O$ be a side-channel. Then, for all $n \in \mathbb{N}$,*

$$\Phi(n) \;\leq\; \Delta(n) \;\leq\; \Lambda(n) \;.$$

Proof. Conditioning on $\mathcal{M}_n$ does not increase the entropy, hence we have $\Delta(n) = H(\mathcal{K}|\mathcal{O}_n\mathcal{M}_n) \leq H(\mathcal{K}|\mathcal{O}_n) = \Lambda(n)$ for all $n \in \mathbb{N}$. For showing $\Phi(n) \leq \Delta(n)$, let $(m_1, \ldots, m_n) = argmin_{m \in M^n} H(\mathcal{K}|\mathcal{O}_n(\mathcal{M}_n = m))$ and observe that $H(\mathcal{K}|\mathcal{O}_n(\mathcal{M}_n = m)) \leq H(\mathcal{K}|\mathcal{O}_n\mathcal{M}_n)$. Define $\mathfrak{a}$ as the attack strategy where, for each node of distance i from the root, the message m_i is chosen as a query. A simple calculation shows that $H(\mathcal{K}|\mathcal{V}_{\mathfrak{a}}) = \sum_{B \in P} p(B) H(\mathcal{K}|\mathcal{V}_{\mathfrak{a}} = B) = H(\mathcal{K}|\mathcal{O}_n(\mathcal{M}_n = (m_1, \ldots, m_n)))$ holds, where P is the partition of K given by $\bigcap_{i=1}^n \{f_{m_i}^{-1}(o) \mid o \in O\}$. Here, $\cap$ denotes the intersection of partitions, which is defined by $Q \cap Q' = \{B \cap B' \mid B \in Q, B' \in Q'\}$. Then $\Phi(n) \leq H(\mathcal{K}|\mathcal{V}_{\mathfrak{a}}) = H(\mathcal{K}|\mathcal{O}_n(\mathcal{M}_n = (m_1, \ldots, m_n))) \leq H(\mathcal{K}|\mathcal{O}_n\mathcal{M}_n) = \Delta(n)$, which concludes this proof.

The inequalities in Theorem 2 are strict for some side-channels f, as the following example shows.

Example 5. Let $K = \{1, 2, 3, 4\}$, $M = \{m_1, m_2\}$, $O = \{1, 2\}$, and $f \colon K \times M \to O$ such that $f_{m_1}^{-1}(1) = \{1, 2\}$ and $f_{m_2}^{-1}(1) = \{2, 3\}$. With a uniform distribution on K, $\Phi(1) = 1$ and $\Phi(n) = 0$, for $n > 1$. According to Theorem 1, $\Lambda(n)$ is bounded from below by $H(\mathcal{K}|\mathcal{V})$. With a uniform distribution on M, we have $p_{\mathcal{O}|\mathcal{K}=1} = p_{\mathcal{O}|\mathcal{K}=3}$, hence $\Lambda(n) \geq H(\mathcal{K}|\mathcal{V}) = \frac{1}{2}H(\mathcal{K}|\mathcal{V} = [1]_\equiv) = \frac{1}{2}$. We have $\lim_{n\to\infty} \Delta(n) = 0$, but Δ will not reach its limit for a finite n as, e.g, $\mathcal{M}_n = (m_1, m_1, \ldots, m_1)$ is a possible choice of messages. Hence, $\Phi(n) < \Delta(n) < \Lambda(n)$ for the given f and large enough n.

We conclude that chosen-message attackers, known-message attackers, and unknown message attackers form a strict hierarchy in terms of the information that they can extract from a given side-channel.

7 Related Work

While there has been substantial work in information-flow security on detecting or quantifying information leaks, there are no results for quantifying the information leakage in unknown-message attacks. Lowe [15] quantifies information flow in a possibilistic process algebra by counting the number of distinguishable behaviors. Clarkson et al. [9] develop a model for reasoning about an adaptive attacker's beliefs about the secret, which may be right or wrong. The information measure proposed by Clark et al. [8] is closely related to ours, however, it is not applicable to side-channel attacks as it does not capture multiple computations with the same key.

There is a large body of work on side-channel cryptanalysis, in particular on attacks and countermeasures. However, there are only a few approaches that give theoretical bounds on what side-channel attackers can, in principle, achieve. Chari et al. [6] are the first to investigate methods for proving hardware implementations secure with respect to power attacks. They propose a generic countermeasure for power attacks and prove that it resists a given number of side-channel measurements. Micali et al. [21] propose physically observable cryptography, a mathematical model that aims at providing provably secure cryptography on hardware that is only partially shielded.

The model of Micali et al. has been been specialized to a framework for the evaluation of side-channel attacks by Standaert, Malkin, and Yung [27] (henceforth called the SMY-model), with applications described in [26, 16, 23]. An analysis with the SMY-model is based on the probability distribution of the attacker's side-channel measurements. These distributions can be obtained from real measurement data, which ensures the validity of the analysis. The SMY-model uses two largely independent metrics for the evaluation of systems. The information-theoretic metric considers only non-adaptive chosen-message adversaries and is not given a direct interpretation in terms of security. The security metric characterizes the security of a system in terms of the success rate for recovering the correct key when applying a given algorithm (e.g., Bayesian classification) to the measurement data. In this way, an analysis with the SMY-model yields meaningful assertions about the effectiveness of the chosen algorithm, but not necessarily worst-case bounds.

By contrast, our metrics abstract from any concrete statistical analysis technique and explicitly consider the way the attacker interacts with the system. This enables us to derive sound worst-case bounds for what can, in principle, be achieved in a side-channel attack. Clearly, such formal bounds are practically relevant only if they are based on a valid system model. For power analysis, the practical implications of the bounds derived using our model require further investigation (see Section 5.3). For timing analysis, the number of clock ticks provides a reasonable and deterministic abstraction of time. For this application domain, our metrics offer the advantage of quantitative bounds that are sound with respect to arbitrary statistical analysis techniques and different kinds of attacker interactions.

8 Future Work and Conclusions

We have presented a novel approach to quantify the secret information that is revealed to unknown-message side-channel attackers. We have applied it to analyze the vulnerability of a finite-field exponentiation algorithm to unknown-message timing attacks. In particular, we have used it to perform the first formal analysis of message-blinding as a countermeasure against timing attacks. Finally, we have given a formal account of the intuition that more comprehensive attackers can extract more information from a given side-channel.

As future work, we plan to investigate whether techniques for entropy estimation [2] can be used to approximate the value of Λ for implementations with operands of larger bit-widths. Another possibility for future work is to investigate whether Λ can be approximated by language-based techniques, e.g., by a type system. This would enable us to derive bounds for systems with larger or infinite state spaces. Finally, it is an open problem to determine information-theoretic bounds for systems that incorporate common components such as cache architectures.

Acknowledgments We thank François-Xavier Standaert for valuable feedback.

References

1. M. Backes and B. Köpf. Formally Bounding the Side-Channel Leakage in Unknown-Message Attacks. Cryptology ePrint Archive, Report 2008/162, 2008.
2. T. Batu, S. Dasgupta, R. Kumar, and R. Rubinfeld. The complexity of approximating entropy. In *Proc. STOC '02*, pages 678–687. ACM, 2002.
3. R. Bird. *Introduction to Functional Programming using Haskell.* Prentice Hall, second edition, 1998.
4. D. Boneh and D. Brumley. Remote Timing Attacks are Practical. In *Proc. USENIX Security Symposium '03.*
5. C. Cachin. Entropy Measures and Unconditional Security in Cryptography. PhD thesis, ETH Zürich, 1997.

6. S. Chari, C. S. Jutla, J. R. Rao, and P. Rohatgi. Towards Sound Approaches to Counteract Power-Analysis Attacks. In *Proc. CRYPTO '99*, LNCS 1666, pages 398–412. Springer.
7. S. Chari, J. R. Rao, and P. Rohatgi. Template Attacks. In *Proc. CHES '02*, LNCS 2523, pages 13–28. Springer.
8. D. Clark, S. Hunt, and P. Malacaria. Quantitative Information Flow, Relations and Polymorphic Types. *J. Log. Comput.*, 18(2):181–199, 2005.
9. M. Clarkson, A. Myers, and F. Schneider. Belief in Information Flow. In *Proc. CSFW '05*, pages 31– 45. IEEE.
10. M. Davio, J. P. Deschamps, and A. Thayse. *Digital Systems with Algorithm Implementation.* John Wiley & Sons, Inc., 1983.
11. K. Gandolfi, C. Mourtel, and F. Olivier. Electromagnetic analysis: Concrete results. In *Proc. CHES '01*, LNCS 2162, pages 251–261. Springer.
12. P. Kocher. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In *Proc. CRYPTO '96*, LNCS 1109, pages 104–113. Springer.
13. P. Kocher, J. Jaffe, and B. Jun. Differential Power Analysis. In *Proc. CRYPTO '99*, LNCS 1666, pages 388–397. Springer.
14. B. Köpf and D. Basin. An Information-Theoretic Model for Adaptive Side-Channel Attacks. In *Proc. CCS '07*, pages 286 – 296. ACM.
15. G. Lowe. Quantifying Information Flow. In *Proc. CSFW '02*, pages 18–31. IEEE.
16. F. Mace, F.-X. Standaert, and J.-J. Quisquater. An Informtion Theoretic Evaluation of Side-Channel Resistant Logic Styles. In *Proc. CHES '07*, LNCS 4727, pages 427–442. Springer.
17. S. Mangard, E. Oswald, and T. Popp. *Power Analysis Attacks: Revealing the Secrets of Smart Cards.* Springer, 2007.
18. J. L. Massey. Guessing and Entropy. In *Proc. IEEE Int. Symp. on Info. Th. '94*, page 204. IEEE.
19. A. Menezes, P. van Oorschot, and S. Vanstone. *Handbook of Applied Cryptography.* CRC Press, 1996.
20. T. S. Messerges, E. A. Dabbish, and R. H. Sloan. Power Analysis Attacks of Modular Exponentiation in Smartcards. In *Proc. CHES '99*, LNCS 1717, pages 144–157. Springer.
21. S. Micali and L. Reyzin. Physically Observable Cryptography (Extended Abstract). In *Proc. TCC '04*, LNCS 2951, pages 278–296. Springer.
22. D. A. Osvik, A. Shamir, and E. Tromer. Cache Attacks and Countermeasures: the Case of AES. In *Proc. CT-RSA '06*, LNCS 3860, pages 1–20. Springer.
23. C. Petit, F.-X. Standaert, O. Pereira, T. G. Malkin, and M. Yung. A Block Cipher based Pseudo Random Number Generator Secure Against Side-Channel Key Recovery. In *Proc. AsiaCCS '08*, pages 56–65. ACM.
24. J.-J. Quisquater and D. Samyde. ElectroMagnetic Analysis (EMA): Measures and Couter-Measures for Smard Cards. In *Proc. E-smart '01*, LNCS 2140, pages 200–210. Springer.
25. P. Schaumont, D. Ching, and I. Verbauwhede. An Interactive Codesign Environment for Domain-Specific Coprocessors. *ACM Transactions on Design Automation for Electronic Systems*, 11(1):70–87, 2006.
26. F.-X. Standaert, E.Peeters, C. Archambeau, and J.-J. Quisquater. Towards Security Limits in Side-Channel Attacks. In *Proc. CHES '06*, LNCS 4249, pages 30–45. Springer.
27. F.-X. Standaert, T. G. Malkin, and M. Yung. A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks. Cryptology ePrint Archive, Report 2006/139, 2006.