A Fast and Key-Efficient Reduction of Chosen-Ciphertext to Known-Plaintext Security*

Ueli Maurer and Johan Sjödin

Department of Computer Science, ETH Zurich, CH-8092 Zurich, Switzerland
{maurer, sjoedin}@inf.ethz.ch

Abstract. Motivated by the quest for reducing assumptions in security proofs in cryptography, this paper is concerned with designing efficient symmetric encryption and authentication schemes based on any *weak* pseudorandom function (PRF) which can be much more efficiently implemented than PRFs. Damgård and Nielsen (CRYPTO '02) have shown how to construct an efficient symmetric encryption scheme based on any weak PRF that is provably secure against chosen-*plaintext* attacks. The main ingredient is a range-extension construction for weak PRFs. By using well-known techniques, they also showed how their scheme can be made secure against the stronger chosen-*ciphertext* attacks.

The results of our paper are three-fold. First, we give a range-extension construction for weak PRFs that is optimal within a large and natural class of reductions (especially all known today). Second, we propose a construction of a regular PRF from any weak PRF. Third, these two results imply a (for long messages) much more efficient chosen-ciphertext secure encryption scheme than the one proposed by Damgård and Nielsen. The results also give answers to open questions posed by Naor and Reingold (CRYPTO '98) and by Damgård and Nielsen.

1 Introduction

1.1 Weakening of Cryptographic Assumptions

A general goal in cryptography is to prove the security of cryptographic systems under assumptions that are as weak as possible. Provably secure encryption and authentication schemes based on a *pseudorandom function* (PRF) [11] have been studied extensively [10]. Informally, a PRF is an efficient function with a secret key that cannot be efficiently distinguished from a uniform random function even when it can be queried adaptively (i.e., under a chosen-plaintext attack (CPA)).

The notion of a PRF is very strong and, indeed, it is unclear whether functions such as block ciphers proposed in the literature have this very strong security property.[1] When designing cryptographic schemes, it is prudent to postulate

* This work was partially supported by the Zurich Information Security Center. It represents the views of the authors.

[1] For example, the design criteria for AES did not include a requirement that a candidate proposal be a PRF, only that it be secure as a block cipher in certain modes of operation, against certain types of attacks.

weaker properties as this makes it more likely that a certain function has such properties or, equivalently, there are potentially more efficient implementations for the weaker requirement compared to the stronger.

A very promising weaker notion of pseudorandomness, proposed by Naor and Reingold [18] (see also [19, 1, 8, 20, 22]), is the *weak* PRF (WPRF). Informally, a WPRF is a function with a secret key that cannot be efficiently distinguished from a uniform random function when given a sequence of *random* inputs and the corresponding outputs (i.e., under a *known-plaintext attack* (KPA)). Highly efficient candidates for WPRFs are described in [7] (cf. [19]), although these are not targeted at this particular security notion explicitly. It is an interesting open problem for further research how much block-cipher design can benefit from this weakening of the desired security goal.

While the design of WPRFs has not been studied as extensively as for PRFs, a concrete argument showing that WPRFs are substantially weaker than PRFs is that WPRFs can have rather strong structural properties which are known to be devastating for PRFs. For instance, if $\mathcal{G}$ is a group of prime order p in which the Decisional Diffie-Hellman (DDH) [9] assumption holds, then

$$F : \mathbb{Z}_p \times \mathcal{G} \to \mathcal{G} \quad \text{defined by} \quad F_k(x) \stackrel{\text{def}}{=} F(k, x) = x^k, \tag{1}$$

where k denotes the secret key, is a WPRF that commutes (i.e., $F_k(F_{k'}(x)) = F_{k'}(F_k(x))$) [17]. A WPRF can also be self inverse (i.e., $F_k(F_k(x)) = x$), have a small fraction of bad points (e.g. $F_k(x) = x$ or $F_k(x) = k$), and have related outputs (e.g. $F_k(x\|1) = F_k(x\|0)$ for all x). Due to such structural flaws, most encryption and authentication schemes based on a PRF become insecure if the PRF is simply replaced by a WPRF (for examples see [8]).

In this paper, we propose provably secure encryption and authentication schemes, for the strongest security notion, under the sole assumption of a WPRF. Of course, the security could be based on even weaker assumptions like the one-wayness of certain functions (as PRFs can be obtained from any one-way function [12, 11]), but these schemes are not of practical interest due to their inefficiency.

1.2 Contributions and Related Work

The main motivation for this paper is Damgård and Nielsen's elegant work on WPRFs [8]. In their paper, the Pseudorandom Tree (PRT) construction was introduced for transforming any WPRF $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ (where the first argument is the key input) into a variable-output-length[2] (VOL) WPRF

$$\mathrm{PRT}^F : \{0,1\}^{3n} \times \{0,1\}^n \times \mathbb{N} \to \{0,1\}^*.$$

They also proposed an efficient CPA-secure[3] symmetric encryption scheme based on PRT^F, that is defined by encrypting a message $m \in \{0,1\}^*$ under a key

[2] For a VOL function family $V : \mathcal{K} \times \{0,1\}^n \times \mathbb{N} \to \{0,1\}^*$, $|V_k(x, l)| = l$ for all k, x, l.

[3] Here, CPA formalizes an adversary's inability, given access to an encryption oracle, to distinguish between two plaintexts given the encryption of one of them.

$k \in \{0,1\}^{3n}$ and some auxiliary uniform randomness $r \in \{0,1\}^n$, as

$$(k, r, m) \mapsto \Big(r, \mathrm{PRT}_k^F(r, |m|) \oplus m\Big). \tag{2}$$

To point out the efficiency of this encryption scheme (and also as a reference for the schemes presented in this work), let us compare it with standard modes of operation such as CBC and CTR. Whereas CBC and CTR invoke the underlying block cipher once per message block to encrypt/decrypt, this scheme invokes the underlying function F once per message block to encrypt/decrypt and roughly $2 \cdot \log_2(b)$ times (where b is the number of message blocks) for generating more key material from the initial key (see below). The key generation can be done offline, so that the throughput is exactly the same as for CBC and CTR. However, whereas CBC and CTR are CPA-secure if the underlying block cipher is a PRF, the Damgård-Nielsen scheme is CPA-secure even when the underlying function is a WPRF, and as WPRFs can be more efficiently implementable than PRFs, their scheme can also be the overall most efficient one. Unfortunately, these modes of operations are not secure against the stronger *chosen-ciphertext attack* (CCA)[4]. In [18, p. 279], Naor and Reingold posed an open problem of how to construct an efficient CCA-secure encryption scheme based on any WPRF. Damgård and Nielsen showed (using well-known techniques) how their CPA-secure scheme can be transformed to a CCA-secure one. Their open question [8, p. 464] whether this can be done more efficiently has been the main motivation for this work.

Before we present our results, let us briefly describe the underlying idea of the PRT-construction (illustrated in Fig. 1(a) on page 9). In a first step, some key material $k_1, \ldots, k_d$ is generated from the initial key k by invoking F in an iterative manner, and then the output blocks are derived by applying F_{k_i}, for $i \in \{1, \ldots, d\}$, iteratively to the input or a previously derived output block. For constructions of this type it is crucial for the security and the efficiency (in terms of the number of applications of F relative to the output length) that this is scheduled in the right way. Recently, two more constructions of this type, the Expanded PRT (ERT) (see Fig. 1(a)) and the Factorial Tree (FCT), were proposed in [16]. However, as we point out in Sect. 3.2, the latter and more efficient construction of the two turns out to be flawed. A natural problem that arises is to find the most efficient VOL-WPRF construction (of this type).

The contributions of this paper are the following:

1. The ICT-Construction – A VOL-WPRF from any WPRF: Our Increasing Chain Tree (ICT) construction (see Fig. 1(b)) is more efficient than PRT and ERT (with d generated keys ICT expands the input by a factor of $2^d - 1$, whereas PRT and ERT expand by roughly $1.44^d - 1$ and $1.73^d - 1$, respectively), and ICT also uses a shorter initial key (by a factor of 3). Interestingly, the generated key sequence $k_1, \ldots, k_d$ is not pseudorandom as opposed to the case for PRT and ERT. Indeed, we give strong arguments that ICT is optimal within the large and natural class of constructions described above, and hence also that it is optimal to use ICT instead of PRT in (2).

[4] In a CCA, the adversary has access to an encryption and decryption oracle.

2. The IC-Construction – A PRF from any WPRF: Our Increasing Chain (IC) construction is similar in nature to Goldreich, Goldwasser, and Micali's (GGM) [11] construction of a PRF from any PRG, but it is more than twice as efficient than first transforming the WPRF into a PRG and then applying the GGM-construction. It is also more efficient than the strengthening of a WPRF to a PRF given in [19][5]. This solves their open problem [18, p. 278] whether a more efficient strengthening exists positively. Interestingly, if we instantiate the IC-construction with the DDH-based WPRF F defined in (1), we get Naor and Reingold's [20] highly efficient PRF based on the DDH assumption but with a non-trivial[6] reduction of the key-material by a factor of roughly the input length of the PRF.
3. CCA-Secure Encryption based on any WPRF: The above results combined with a Wegman-Carter [25] based message authentication code (MAC) and the well-known encrypt-then-MAC method [15, 5], yield a CCA-secure encryption scheme from any WPRF that is substantially more efficient than the CCA-secure encryption scheme proposed by Damgård-Nielsen (their number of applications to the WPRF for the MACing is linear in the message length whereas ours is constant). We observe that for our purposes a much weaker primitive than the MAC, namely a *weak* MAC (WMAC)[7], is sufficient (encrypt-then-WMAC actually does the job). This raises the question of constructing possibly efficient WMACs from any WPRF.
4. Non-adaptive[8] CCA-Secure Encryption based on any WPRF and WMAC: Although this type of security may (like CPA-security) be unsatisfactory in practice, the exact requirements for achieving standard security notions are interesting in their own right. It might also motivate further research on basing strong primitives on weak assumptions. Non-adaptive CCA-security has been studied under stronger assumptions in [18].

2 Preliminaries

2.1 Notation and Definitions

Let $s \xleftarrow{\$} S$ denote that s is selected uniformly at random from the set S. If $\mathcal{D}$ is a probability distributions over S then $s \leftarrow \mathcal{D}$ denotes the operation of selecting s at random according to $\mathcal{D}$. If x and y are two bitstrings, $x\|y$ denotes their concatenation, $x[i]$ the i-th bit of x, $x[i,j] \stackrel{\text{def}}{=} x[i]\|x[i+1]\|\cdots\|x[j]$ for $i < j$, and $x[i,i] \stackrel{\text{def}}{=} x[i]$. For two functions f and g, $f \circ g\,(x) \stackrel{\text{def}}{=} f(g(x))$. A function has *variable-input-length* (VIL) if the domain is $\{0,1\}^{\leq N} \stackrel{\text{def}}{=} \cup_{i=1}^{N}\{0,1\}^i$ (for some $N > 1$), and a function $f : \{0,1\}^n \times \mathbb{N} \to \{0,1\}^*$ has *variable-output-length* (VOL) if for all all x and l, $|f(x,l)| = l$ and $f(x, l+1) = f(x,l)\|b$ for some bit b. Let $\mathcal{R}_{N,n}$ and $\mathcal{R}_{\leq N,n}$ denote uniform random functions with

[5] In that work, the PRF is reduced – via a pseudorandom synthesizer – to a WPRF.
[6] The key is not replaced by a pseudorandom sequence based on F.
[7] A WMAC is unforgeable under a *known-plaintext attack* (see [18]).
[8] Here the adversary has no oracle access after the challenge (ciphertext) is presented.

range $\{0,1\}^n$, and domain $\{0,1\}^N$ and $\{0,1\}^{\leq N}$, respectively. Let $\mathcal{R}_{n,*}$ denote a VOL-function $\{0,1\}^n \times \mathbb{N} \to \{0,1\}^*$ for which $\mathcal{R}_{n,*}(\cdot, l)$ is a uniform random function $\{0,1\}^n \to \{0,1\}^l$ for all l. Abusing notation, we refer to $\mathcal{R}_{n,*}$ as a uniform random VOL-function. We let $\Pr[\Pi : \mathcal{E}]$ denote the probability of event $\mathcal{E}$ in random experiment Π. $A^{\mathcal{O}}$ denotes an algorithm A with access to an oracle $\mathcal{O}$.

2.2 Cryptographic Functions

Concrete Security. We state our results in the concrete security framework, which was formalized for the following primitives by Bellare, Kilian, and Rogaway [4]. Let $\mathcal{O}^f$ denote the oracle which, if invoked, returns $(r, f(r))$ for a uniform random input r of the function f. The **w**-*advantage* of adversary A for $F : \mathcal{K} \times \{0,1\}^N \to \{0,1\}^n$ with $\mathbf{w} \in \{\mathbf{prf}, \mathbf{wprf}, \mathbf{mac}, \mathbf{wmac}\}$ is defined as:

$$\mathbf{Adv}^{\mathbf{prf}}_{F,A} \stackrel{\text{def}}{=} \left|\Pr\left[k \stackrel{\$}{\leftarrow} \mathcal{K}, b \leftarrow A^{F_k} : b = 1\right] - \Pr\left[\mathbf{R} \leftarrow \mathcal{R}_{N,n}, b \leftarrow A^{\mathbf{R}} : b = 1\right]\right|$$

$$\mathbf{Adv}^{\mathbf{wprf}}_{F,A} \stackrel{\text{def}}{=} \left|\Pr\left[k \stackrel{\$}{\leftarrow} \mathcal{K}, b \leftarrow A^{\mathcal{O}^{F_k}} : b = 1\right] - \Pr\left[\mathbf{R} \leftarrow \mathcal{R}_{N,n}, b \leftarrow A^{\mathcal{O}^{\mathbf{R}}} : b = 1\right]\right|$$

$$\mathbf{Adv}^{\mathbf{mac}}_{F,A} \stackrel{\text{def}}{=} \left|\Pr\left[k \stackrel{\$}{\leftarrow} \mathcal{K}, (m,\tau) \leftarrow A^{F_k}, b = \begin{cases} 1 \text{ if } \tau = F_k(m), m \text{ ``new''} \\ 0 \text{ otherwise} \end{cases} : b = 1\right]\right|$$

$$\mathbf{Adv}^{\mathbf{wmac}}_{F,A} \stackrel{\text{def}}{=} \left|\Pr\left[k \stackrel{\$}{\leftarrow} \mathcal{K}, (m,\tau) \leftarrow A^{\mathcal{O}^{F_k}}, b = \begin{cases} 1 \text{ if } \tau = F_k(m), m \text{ ``new''} \\ 0 \text{ otherwise} \end{cases} : b = 1\right]\right|$$

where "*m new*" stands for the event that m is distinct from the inputs to F_k. The maximal **w**-advantages are defined as $\mathbf{Adv}^{\mathbf{w}}_{F}(t,q) \stackrel{\text{def}}{=} \max_A\{\mathbf{Adv}^{\mathbf{w}}_{F,A}\}$, where the maximum is taken over all A restricted to time-complexity[9] t and q (respectively $q-1$ if $\mathbf{w} \in \{\mathbf{mac}, \mathbf{wmac}\}$) invocations of its oracle.

VIL-Function Families. For a VIL-function family $F : \mathcal{K} \times \{0,1\}^{\leq N} \to \{0,1\}^n$, the **vil-mac**-advantage $\mathbf{Adv}^{\mathbf{vil\text{-}mac}}_{F,A}$ is defined like the **mac**-advantage, except that the adversary A may query inputs of any length ($\leq N$). Let $\mathcal{O}^f_{\mathbf{vil}}$ (for some VIL-function f) denote an oracle that on input $l \leq N$ generates a uniform random input $r \in \{0,1\}^l$ and outputs $(r, f_k(r))$. The **vil-wmac**-advantage $\mathbf{Adv}^{\mathbf{vil\text{-}wmac}}_{F,A}$ is defined like the **wmac**-advantage except that the oracle $\mathcal{O}$ is replaced by $\mathcal{O}_{\mathbf{vil}}$. For $\mathbf{w} \in \{\mathbf{mac}, \mathbf{wmac}\}$, the maximal advantage is defined as $\mathbf{Adv}^{\mathbf{vil\text{-}w}}_{F}(t,q,\mu) \stackrel{\text{def}}{=} \max_A\{\mathbf{Adv}^{\mathbf{vil\text{-}w}}_{F,A}\}$, where the maximum is taken over all A with time-complexity t, making at most $q-1$ oracle invocations such that the total length of the inputs to F (including the forgery message) is at most μ bits.

VOL-Function Families. Let $\mathcal{O}^f_{\mathbf{vol}}$ denote the oracle that on input $l \in \mathbb{N}$ outputs $(r, f(r,l))$ for a uniform random $r \in \{0,1\}^n$. For a VOL-function family $F : \mathcal{K} \times \{0,1\}^n \times \mathbb{N} \to \{0,1\}^*$, the **vol-wprf**-advantage of A for F is

$$\mathbf{Adv}^{\mathbf{vol\text{-}wprf}}_{F,A} \stackrel{\text{def}}{=} \Pr\left[k \leftarrow \mathcal{K}, b \leftarrow A^{\mathcal{O}^{F_k}_{\mathbf{vol}}} : b = 1\right] - \Pr\left[\mathbf{R} \leftarrow \mathcal{R}_{n,*}, b \leftarrow A^{\mathcal{O}^{\mathbf{R}}_{\mathbf{vol}}} : b = 1\right],$$

[9] I.e., t is the worst-case total running time (including the length of A) of the experiment in which A interacts with its oracle (in some fixed RAM model of computation).

and by maximizing over all A, restricted to time-complexity t and at most q oracle queries whose sum totals at most μ, we get the maximal **vol-wprf**-advantage $\mathbf{Adv}_F^{\textbf{vol-wprf}}(t,q,\mu) \stackrel{\text{def}}{=} \max_A\{\mathbf{Adv}_{F,A}^{\textbf{vol-wprf}}\}$.

3 The IC- and ICT-Construction

In this section, we propose the IC-construction, for transforming a WPRF into a PRF, and the ICT-construction, for transforming a WPRF into a VOL-WPRF. Throughout, let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ denote a function family.[10]

3.1 A PRF from any WPRF

The IC-construction transforms $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ into

$$\mathrm{IC}^F : (\{0,1\}^n \times \{0,1\}^n \times \{0,1\}^n) \times \{0,1\}^N \to \{0,1\}^n,$$

for some fixed N, where $\mathrm{IC}^F_{k_1,r,\tau_1}(x)$ is defined by the following algorithm:

if $|x| > 1$ **then**
  for $i = 2$ to $|x|$ **do** $k_i = F_{k_{i-1}}(r)$
for $i = 1$ to $|x|$ **do**
  if $x[i] = 1$ **then**
    $\tau_{i+1} = F_{k_i}(\tau_i)$
  else
    $\tau_{i+1} = \tau_i$
return $\tau_{|x|}$

The following theorem states that IC^F is a PRF if F is a WPRF, even if the r-value of the initial key is not kept secret. Note that F is invoked at most $2N-1$ times. However, the first $N-1$ invocations can be pre-processed and cached, and hence at most N invocations are necessary or, to be precise, as many invocations as there are ones in the input.

Theorem 1. *For any t, q, and input length N of* IC^F

$$\mathbf{Adv}_{\mathrm{IC}^F}^{\textbf{prf}}(t,q) \leq N \cdot \left(\mathbf{Adv}_F^{\textbf{wprf}}(t,q) + \frac{q(q+1)}{2^{n+1}}\right).$$

Proof. Let Π_0 denote the following random experiment for an adversary A with time-complexity t which makes at most q queries to its oracle:

$$(k_1, r, \tau_1) \stackrel{\$}{\leftarrow} \{0,1\}^n \times \{0,1\}^n \times \{0,1\}^n,\ b \leftarrow A^{\mathrm{IC}^F_{k_1,r,\tau_1}}.$$

Note that for any query x issued by A and any $s \in \{1,\ldots,N\}$, the sequence $(\tau_1,\ldots,\tau_s)$ (resulting from the second for-loop) does not depend on $x[s,N]$.

[10] For simplicity, we choose the key-length to be the same as the input length. We refer to [8] for constructing such a WPRF from any WPRF.

Hence, $(\tau_1, \ldots, \tau_s)$ can be reused for any other query x' for which $x'[1, s-1] = x[1, s-1]$. We assume that $\mathrm{IC}^F_{k_1,r,\tau_1}$ reuses previously computed τ-values (for saving calls to F) whenever possible, by maintaining a look-up table with all the entries $(x[1,s], \tau_{s+1})$ for which x is a query to $\mathrm{IC}^F_{k_1,r,\tau_1}$, $s \in \{1, \ldots, N\}$, and $x[s] = 1$. We also assume that the calls to F in the first for-loop are preprocessed and cached. For $j = 1, \ldots, N$, let Π_{2j-1} be the same experiment as Π_{2j-2} except that F_{k_j} is replaced by a random function $\mathbf{R}_j$, and let Π_{2j} be the same experiment as Π_{2j-1} except that for each query x issued by A, for which $x[j] = 1$ and $x[1,j]$ is not in the look-up table, the output of $\mathbf{R}_j$ is replaced by a uniform random $R \in \{0,1\}^n$ and $(x[1,j], R)$ is inserted into the table. Let S_i be the event that $b = 1$ in Π_i, for $i = 0, \ldots, 2N$. Now, as Π_{2N} is equivalent to $[\mathbf{R} \leftarrow \mathcal{R}_{N,n},\ b \leftarrow A^{\mathbf{R}}]$, we get

$$
\begin{aligned}
&\mathbf{Adv}^{\mathbf{prf}}_{\mathrm{IC}^F,A} \stackrel{\mathrm{def}}{=} \big|\Pr[S_0] - \Pr[S_{2N}]\big| \\
&\le \sum_{j=1}^{N} \big|\Pr[S_{2j-2}] - \Pr[S_{2j-1}]\big| + \sum_{j=1}^{N} \big|\Pr[S_{2j-1}] - \Pr[S_{2j}]\big| \\
&\le \sum_{j=1}^{N} \mathbf{Adv}^{\mathbf{wprf}}_{F}\big(t, \min\{q+1, 2^{j-1}+1\}\big) + \sum_{j=1}^{N} \min\left\{\frac{(q+1)q}{2^{n+1}}, \frac{(2^{j-1}+1)2^{j-1}}{2^{n+1}}\right\} \\
&\le N \cdot \left(\mathbf{Adv}^{\mathbf{wprf}}_{F}(t, q+1) + \frac{(q+1)q}{2^{n+1}}\right),
\end{aligned}
$$

due to the triangle inequality and the following two facts. First, for $j = 1, \ldots, N$, A can be transformed to a WPRF distinguisher A' for F with time-complexity t, making at most $\min(q+1, 2^{j-1}+1)$ oracle invocations, and having advantage at least $|\Pr[S_{2j-2}] - \Pr[S_{2j-1}]|$. A' with oracle T, simulates the experiment Π_{2j-2} if T is an instance of F and Π_{2j-1} if T is a random function $\mathbf{R}$ (which is possible as all queries to F_{k_j} in Π_{2j-2} and to $\mathbf{R}_j$ in Π_{2j-1} are distributed uniformly at random). Finally, A' decides as A does. Second, Π_{2j-1} and Π_{2j} are equivalent experiments as long as no collision among the inputs on which $\mathbf{R}_j$ is invoked occurs. As $\mathbf{R}_j$ is invoked on at most $\min\{q+1, 2^{j-1}+1\}$ inputs and these are all random, the probability of this event is upper bounded by $\min\big\{(q+1)q/2^{n+1}, (2^{j-1}+1)2^{j-1}/2^{n+1}\big\}$. □

Key Reduction of Naor-Reingold's DDH-Based PRF. In [20], Naor and Reingold presented an efficient construction of a PRF based on the DDH assumption. It is easy to verify, that IC^F with F as defined in (1) is the same PRF but with a significantly shorter key by a factor of roughly N (recall that N is the input length of IC^F). To be more precise, the first for-loop generates a sequence $k_1, \ldots, k_N$ of keys from the initial key (k_1, r, τ_1) and the second for-loop exactly corresponds to the Naor-Reingold construction with $k_1, \ldots, k_N$ as its key. Note that the reduction is non-trivial in the sense that $k_1, \ldots, k_N$ is not generated from a PRG based on F. For instance $F^{-1}_{k_1}(k_2) = F^{-1}_{k_2}(k_3)$ holds which can easily be verified given k_1, k_2, and k_3.

THE GGM-APPROACH. An alternative (but less efficient) approach to obtain a PRF from any WPRF F is to first transform F into a pseudorandom generator (PRG) and then apply the so-called GGM-construction [11] (which transforms a PRG into a PRF). Informally, a PRG is an efficient deterministic function mapping a truly random string (or seed) to a longer string which is computationally indistinguishable from random. Let us briefly describe the GGM-construction. It transforms a length-doubling PRG G into a PRF (say with N-bits input) as

$$GGM_k(x) \stackrel{\text{def}}{=} G_{x[1]} \circ \ldots \circ G_{x[N]}(k),$$

where $G_0(k)$ and $G_1(k)$ denote the left and right half of $G(k)$, respectively. The most efficient construction of a length doubling PRG G from F, that we are aware of, uses 3 and 4 invocations to F, respectively, for computing G_0 and G_1:

$$G(k_1\|r\|x) \stackrel{\text{def}}{=} x\|F_{k_1}(x)\|F_{k_2}(x)\|F_{k_2} \circ F_{k_1}(x)\|F_{k_3}(x)\|r,$$

where $k_2 = F_{k_1}(r)$ and $k_3 = F_{k_2}(r)$. The proof that G is a PRG if F is a WPRF follows directly from Theorem 2 (see next section) and the fact that $G(k_1\|r\|x) = x\|\,\mathrm{ICT}^F_{k,r}(x, 4n)\|r$. Hence, to get a PRF with N-bits input and n-bits output, we roughly need $4N$ invocations of F per call in the worst case (cf. the efficiency of IC^F above).

Remark 1. Let us briefly point out a method for improving the computation time of IC^F, at the cost of generating and storing more keys (say N' keys instead of N). On input x (of length N), x is first injectively mapped to a N'-bit string x' of Hamming weight at most some (fixed) c, satisfying

$$\sum_{i=0}^{c} \binom{N'}{i} \geq 2^N.$$

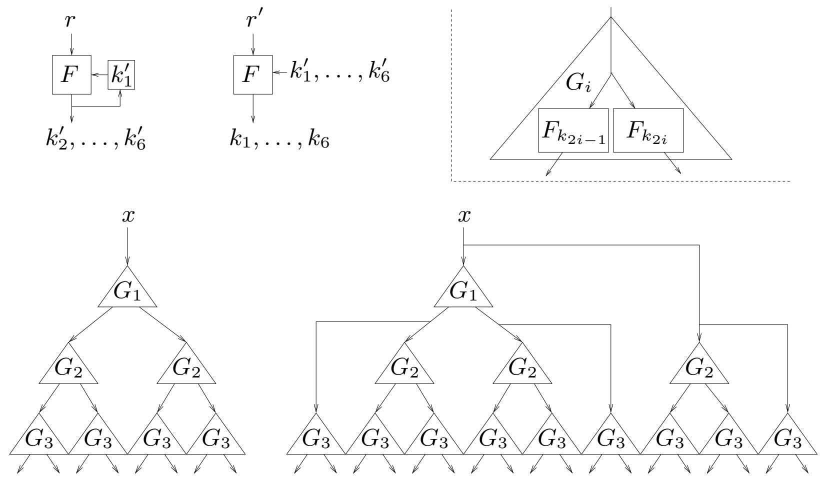
Then IC^F is invoked on x' and the result is output. Here, F is invoked at most c (as opposed to N) times, as there are at most c ones in the input.

3.2 A VOL-WPRF from any WPRF

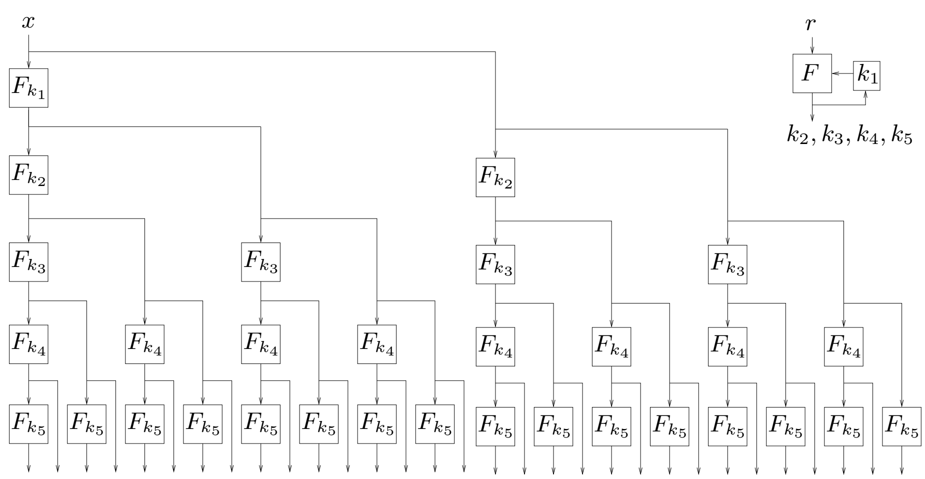
The ICT-construction is illustrated in Fig. 1(b) and is defined as

$$\begin{aligned}\mathrm{ICT}^F : (\{0,1\}^n \times \{0,1\}^n) \times \{0,1\}^n \times \mathbb{N} &\to \{0,1\}^* \\ ((k,r),x,l) &\mapsto \left(\mathrm{IC}^F_{k,r,x}(\langle 1\rangle)\big\|\,\mathrm{IC}^F_{k,r,x}(\langle 2\rangle)\big\|\cdots\big\|\,\mathrm{IC}^F_{k,r,x}(\langle\lceil l/n\rceil\rangle)\right)[1,l],\end{aligned}$$

where $\langle i\rangle$ denotes the reversed standard bit encoding of the integer i (e.g. $\langle 0\rangle = 0, \langle 1\rangle = 1, \langle 2\rangle = 01, \langle 3\rangle = 11, \langle 4\rangle = 001$). Note that $\mathrm{IC}^F_{k,r,x}(\langle 0\rangle) = x$ can not be part of the output, as x is the input. It is easy to verify, see Fig. 1(b), that $\mathrm{ICT}^F_{k,r}(x,l)$ needs $d-1 = \lfloor\log_2(\lceil l/n\rceil)\rfloor$ calls to F for computing (or precomputing) the needed keys $k_1, \ldots, k_d$ and further $\lceil l/n\rceil$ calls for computing the output (i.e., one call per output block). The next theorem states that ICT^F is a VOL-WPRF if F is a WPRF. As for IC, the r-value of the key need not be kept secret.



(a) Computation of $\mathrm{PRT}^F_{k_1',r,r'}(x, 14n)$ (bottom left) and $\mathrm{ERT}^F_{k_1',r,r'}(x, 26n)$ (bottom right), i.e., the maximal sized output using 6 generated keys $k_1, \ldots, k_6$ (upper left). Here every output of G_i (defined upper right) for $i = 1, 2, 3$ is part of the global output.



(b) Computation of $\mathrm{ICT}^F_{k_1,r}(x, 31n)$, i.e., the output of maximal size using 5 generated keys $k_1, \ldots, k_5$ (upper right). Here every output of F – except for the generated keys (upper right) – is part of the global output. We stress that the order of the output blocks are not the same as presented in the text.

Fig. 1. Illustration of (a) PRT [8], ERT [16], and (b) ICT (of this paper). The generated key sequence $k_1, k_2, \ldots$ is not pseudorandom in (b) as opposed to in (a) (see Sect. 3.1).

Theorem 2. *For any $t, q,$ and μ*

$$\mathbf{Adv}^{\textbf{vol-wprf}}_{\mathrm{ICT}^F}(t,q,\mu) \le d_{\max} \cdot \mathbf{Adv}^{\textbf{wprf}}_F(t, q(2^{d_{\max}-1}+1)) + \frac{4^{d_{\max}} \cdot q^2}{2^n},$$

where $d_{\max} = \lfloor \log_2(\lceil l_{\max}/n \rceil) \rfloor + 1$ and $l_{\max} \le \mu$ is the maximum allowed output length of ICT^F.

Proof. Let Π_0 denote the following random experiment for an adversary A with time-complexity t that make at most q queries whose sum is at most μ:

$$(k,r) \xleftarrow{\$} \{0,1\}^n \times \{0,1\}^n, b \leftarrow A^{\mathcal{O}_{\textbf{vol}}^{\mathrm{ICT}^F_{k,r}}}.$$

Let d denote the maximal number of generated keys (for F), needed for answering the queries to $\mathrm{ICT}^F_{k,r}$ issued by A. Note that the j-th instantiation of F, i.e., F_{k_j}, for $j \in \{1,\ldots,d\}$, is queried at most $q_j = q \cdot (2^{j-1}+1)$ times. For $j = 1,\ldots,d$, let Π_{2j-1} denote the same random experiment as Π_{2j-2} except that F_{k_j} is replaced by a random function $\mathbf{R}_j$, and let Π_{2j} be the same experiment as Π_{2j-1} except that the outputs of $\mathbf{R}_j$ are replaced by uniform random n-bit strings. Furthermore, let Π_{2d+1} denote the random experiment $[\mathbf{R} \xleftarrow{\$} \mathcal{R}_{n,*}, b \leftarrow A^{\mathcal{O}^{\mathbf{R}}_{\textbf{vol}}}]$. Now, for $i = 0,\ldots,2d+1$, let S_i denote the event that $b = 1$ in Π_i. We get

$$\begin{aligned}
&\mathbf{Adv}^{\textbf{vol-wprf}}_{A,\mathrm{ICT}^F} = \left|\Pr[S_0] - \Pr[S_{2d+1}]\right| \\
&\le \sum_{j=1}^{d} \left|\Pr[S_{2j-2}] - \Pr[S_{2j-1}]\right| + \sum_{j=1}^{d} \left|\Pr[S_{2j-1}] - \Pr[S_{2j}]\right| + \Big|\Pr[S_{2d}] - \Pr[S_{2d+1}]\Big| \\
&\le \sum_{j=1}^{d} \mathbf{Adv}^{\textbf{wprf}}_F(t, q_j) + \sum_{j=1}^{d} \frac{q_j^2}{2^{n+1}} + \frac{q^2}{2^{n+1}} \le d \cdot \mathbf{Adv}^{\textbf{wprf}}_F\left(t, q(2^{d-1}+1)\right) + \frac{q^2 4^d}{2^n},
\end{aligned}$$

using the triangle inequality and the following facts. As Π_{2d} and Π_{2d+1} are equivalent experiments as long as the input part of the samples returned by the oracle are distinct, we get $|\Pr[S_{2d}] - \Pr[S_{2d+1}]| \le q^2/2^{n+1}$. Furthermore, as Π_{2j-1} and Π_{2j} are equivalent as long as the random inputs to $\mathbf{R}_j$ are all distinct, it holds that $|\Pr[S_{2j-1}] - \Pr[S_{2j}]| \le q_j^2/2^{n+1}$. Finally, A can be transformed into a WPRF distinguisher A' for F with time-complexity t, that makes q_j oracle queries and has advantage $|\Pr[S_{2j-2}] - \Pr[S_{2j-1}]|$. A' with oracle $\mathcal{O}^T$ simply simulates the random experiment that is equivalent to Π_{2j-2} if T is an instance of F and to Π_{2j-1} if T is a random function $\mathbf{R}$ (this is possible as the inputs to F_{k_j} in Π_{2j-2} and to $\mathbf{R}_j$ in Π_{2j-1} are distributed uniformly at random). Finally, A' decides as A does. □

THE FCT-CONSTRUCTION IS FLAWED. Let us point out that the security proof of FCT (in [16]) is flawed. The maximal sized output of FCT^F for two generated keys k_1 and k_2 is defined as

$$x \mapsto F_{k_1}(x) \| F_{k_2}(x) \| F_{k_2} \circ F_{k_1}(x) \| F_{k_1} \circ F_{k_2}(x). \tag{3}$$

Clearly, the construction is insecure for any WPRF F that commutes (i.e., for which $F_{k_2} \circ F_{k_1}(x) = F_{k_1} \circ F_{k_2}(x)$ for all k_1, k_2, and x). As such WPRFs exist under the DDH assumption (see (1)), a fix of the security proof would contradict the assumption and thus be a major breakthrough in number theory.[11]

Comparing ICT with other Constructions. The idea behind PRT of [8], ERT of [16], and ICT is to first generate keys $k_1, \ldots, k_d$ from the initial key (and F) and then to derive the output blocks sequentially by invoking F_{k_i}, with $i \in \{1, \ldots, d\}$, to the input or a previously computed output block (see Fig. 1). ICT is superior to PRT and ERT for three reasons. First, the initial key of ICT is n bits (plus n bits that may be publicly known) versus $3n$ bits for PRT and ERT. Second, ICT needs $d-1$ invocations of F to generate the d keys $k_1, \ldots, k_d$ whereas PRT and ERT needs $2d-1$. Third, the maximal output size using $k_1, \ldots, k_d$ is $(2^d-1)n$ for ICT, roughly $(3^{\frac{d}{2}}-1)n$ for ERT, and roughly $(2^{\frac{d}{2}+1}-2)n$ for PRT.[12] For all constructions, the keys needed for computing outputs of length bounded by some fixed value (say $l_{\max}$) can be pre-processed, such that one call of F is needed per output block. But whereas ICT needs to store say $s \stackrel{\text{def}}{=} \lfloor \log_2(\lceil l_{\max}/n \rceil) \rfloor + 1$ keys, ERT and PRT store about $\lceil 1.26s \rceil$ and $2s$ keys, respectively. The factor in front of the **wprf**-advantage in the security reduction reduces correspondingly, i.e., for s as defined above we roughly have

$$\begin{aligned}
\mathbf{Adv}^{\textbf{vol-wprf}}_{\mathrm{ICT}^F}(t,q) &\leq \quad s \cdot \mathbf{Adv}^{\textbf{wprf}}_{F}(t, 2^{s-1}q) \quad + \; 4^s q^2/2^n \\
\mathbf{Adv}^{\textbf{vol-wprf}}_{\mathrm{ERT}^F}(t,q) &\leq 1.26s \cdot \mathbf{Adv}^{\textbf{wprf}}_{F}(t, 2^{s-1}q/3) + 4^s q^2/(2^n \cdot 9) \\
\mathbf{Adv}^{\textbf{vol-wprf}}_{\mathrm{PRT}^F}(t,q) &\leq \quad 2s \cdot \mathbf{Adv}^{\textbf{wprf}}_{F}(t, 2^{s-1}q/4) + 4^s q^2/(2^n \cdot 16).
\end{aligned}$$

Optimality of the ICT-Construction. In [22], it is shown that there is no black-box proof of the security for constructions that expands more than ICT (for any fixed number of generated keys). Here, we show something stronger for the constructions with log-time random access to output blocks, i.e., for the rather balanced constructions where the maximal length of the composition chains are in $O(\log(l))$ for output length l, namely that ICT is optimal within that class of constructions under the *inverse* DDH (IDDH) assumption [2].

To be more precise, note that – for $l = 3n$ – the value $\mathrm{ICT}^F_{k_1,r}(x,l)$ is derived by first computing $k_2 = F_{k_1}(r)$ and then returning

$$y := F_{k_1}(x) \| F_{k_2}(x) \| F_{k_2} \circ F_{k_1}(x).$$

For $l = 7n$, an extra key $k_3 = F_{k_2}(r)$ is derived and

$$y \| F_{k_3}(x) \| F_{k_3} \circ F_{k_1}(x) \| F_{k_3} \circ F_{k_2}(x) \| F_{k_3} \circ F_{k_2} \circ F_{k_1}(x)$$

is returned. A natural question is whether more can be output before a new key needs to be generated, i.e., for a fixed number of generated keys (say k_1, k_2,

[11] However, information theoreticly (and even in Minicrypt, i.e., under the assumption that one-way functions exist but no public-key cryptography) (3) is secure [21, 22].

[12] The latter two values are exact if d is even. Otherwise $(2 \cdot 3^{\frac{d-1}{2}}-1)n$ and $(3 \cdot 2^{\frac{d-1}{2}}-2)n$ are exact, respectively.

and k_3), can we output more than ICT^F maximally can (i.e., more than $7n$ bits) by invoking the instantiations (i.e., $F_{k_1}, F_{k_2}, F_{k_3}$) *one* more time than ICT^F does (i.e., 8 times instead of 7). The answer turns out to be “no” unless the IDDH assumption is false, since otherwise there is a WPRF F, described in (4), which with high probability both commutes and is self inverse, i.e., for all $k \neq k'$

$$\Pr_x[F_k \circ F_{k'}(x) = F_{k'} \circ F_k(x)] \approx 1/4 \quad \text{and} \quad \Pr_x[F_k \circ F_k(x) = x] \approx 1/2.$$

If F is used and more is output at least two output blocks will (by the pigeonhole principle) have the same value with high probability (which is unlikely for a uniform random VOL-function). F is defined for a group $\mathcal{G}$ of prime order ρ as

$$F : \mathbb{Z}_\rho \times \mathcal{G} \to \mathcal{G} \quad \text{and} \quad F_k(x) \stackrel{\text{def}}{=} \begin{cases} x^k & \text{if } x \in P_1 \\ x^{k^{-1}} & \text{if } x \in P_2 \end{cases}, \tag{4}$$

where $k \cdot k^{-1} = 1 \pmod{\rho}$ and $\{P_1,\, P_2\}$ is a partition of $\mathcal{G}$ in roughly equal sized sets (where we assume that it is efficient to decide whether $x \in P_1$ or not). A proof that F is a WPRF if the IDDH assumption holds in $\mathcal{G}$ is given in [14].

4 Applications

In this section, we optimize Damgård and Nielsen's CPA-secure encryption scheme (2) by using ICT instead of PRT. Then, we first make the scheme CCA-secure by applying IC and the well-known encrypt-then-MAC technique (actually we prove that what we call encrypt-then-WMAC does the job here), and, second, we make it non-adaptive CCA-secure by using a (fixed-input-length) WMAC for authenticating the auxiliary uniform randomness.

4.1 Symmetric Encryption

A symmetric encryption scheme $\mathcal{SE} = (E, D)$ consists of two efficient algorithms. The (randomized) encryption algorithm E maps a key k and a message m to a ciphertext $c = E_k(m)$, and the deterministic decryption algorithm D maps a key k and a ciphertext $c = E_k(m)$ to the message $m = D_k(c)$. There are several notions for privacy and integrity of $\mathcal{SE}$ (for an overview, we refer to [5, 13, 3]). We consider the IND-PX-CY notion (for $\mathsf{X},\mathsf{Y} \in \{0,1,2\}$), introduced in [13].

Definition 1 (IND-PX-CY). *For an encryption scheme $\mathcal{SE} = (E, D)$ (with message space $\mathcal{M}$ and keyspace $\mathcal{K}$), the* **ind-p***x***-c***y-advantage of an adversary A (with $x, y \in \{0,1,2\}$) is defined as follows (where $\perp$ denotes no oracle):*

$$\mathbf{Adv}^{\mathbf{ind\text{-}p}x\mathbf{\text{-}c}y}_{\mathcal{SE},A} \stackrel{\text{def}}{=} 2\cdot\Pr\Big[k \xleftarrow{\$} \mathcal{K}, (m_0, m_1) \leftarrow A^{\mathcal{O}_1,\mathcal{O}_2}, b \xleftarrow{\$} \{0,1\}, c \leftarrow E_k(m_b), \hat{b} \leftarrow A^{\mathcal{O}'_1,\mathcal{O}'_2}(c) : \hat{b} = b\Big] - 1,$$

where $(\mathcal{O}_1, \mathcal{O}'_1) = \begin{cases} (\perp,\perp) & \textit{if } x = 0 \\ (E_k,\perp) & \textit{if } x = 1 \\ (E_k, E_k) & \textit{if } x = 2 \end{cases}, \quad (\mathcal{O}_2, \mathcal{O}'_2) = \begin{cases} (\perp,\perp) & \textit{if } y = 0 \\ (D_k,\perp) & \textit{if } y = 1 \\ (D_k, D_k) & \textit{if } y = 2 \end{cases},$

$m_0, m_1 \in \mathcal{M}$ with $|m_0| = |m_1|$, and A does not query $\mathcal{O}'_2$ with c. By maximizing over all A restricted to time-complexity t, at most $q-1$ encryption queries of total length at most $(\mu - |m_0|)$ bits, and q' decryption queries of total length at most μ' bits, we let $\mathbf{Adv}^{\textbf{ind-p}x\textbf{-c}y}_{\mathcal{SE}}(t, q, \mu, q', \mu') \stackrel{def}{=} \max_A\{\mathbf{Adv}^{\textbf{ind-p}x\textbf{-c}y}_{\mathcal{SE},A}\}$ (where one typically drops the parameters (q', μ') if $y = 0$).

The IND-P2-C0, IND-P2-C2, and IND-P1-C1 notions are often referred to as IND-CPA, (adaptive) IND-CCA, and non-adaptive IND-CCA, respectively.

The *integrity of ciphertext* (INT-CTXT) [5] notion formalizes an adversary's inability – given access to an encryption oracle – to create a new valid ciphertext:

Definition 2 (INT-CTXT). *For $\mathcal{SE} = (E, D)$ (with message space $\mathcal{M}$ and key space $\mathcal{K}$), let D^*_k denote an algorithm that on input c outputs 1 iff c is a valid ciphertext under the key k, i.e., there exists $m \in \mathcal{M}$ such that $D_k(c) = m$.*

$$\mathbf{Adv}^{\textbf{int-ctxt}}_{\mathcal{SE},A} \stackrel{def}{=} \Pr\left[k \stackrel{\$}{\leftarrow} \mathcal{K}, A^{E_k, D^*_k}, b \stackrel{def}{=} \begin{cases} 1 & \text{If } \exists i\ \forall j : D^*_k(y_i) = 1 \wedge y_i \neq c_j \\ 0 & \text{otherwise} \end{cases} : b = 1\right],$$

where $c_1, \ldots, c_q$ denote the outputs from E_k and $y_1, \ldots, y_{q'}$ denote A's queries to D^*_k. By maximizing over all A with time-complexity t, that makes at most q queries to E_k of total length at most μ bits, and at most q' queries to D^*_k of total length at most μ' bits, we let $\mathbf{Adv}^{\textbf{int-ctxt}}_{\mathcal{SE}}(t, q, \mu, q', \mu') \stackrel{def}{=} \max_A\{\mathbf{Adv}^{\textbf{int-ctxt}}_{\mathcal{SE},A}\}$.

4.2 A CPA-Secure Encryption Scheme

In [8], Damgård and Nielsen introduced an IND-P2-C0-secure encryption scheme based on any VOL-WPRF $V : \{0,1\}^\kappa \times \{0,1\}^n \times \mathbb{N} \to \{0,1\}^*$. To be precise, their encryption scheme $\mathcal{SE}_1$ is defined by encrypting a message $m \in \{0,1\}^*$, under the key $k \in \{0,1\}^\kappa$ and some auxiliary uniform randomness $r \in \{0,1\}^n$ as

$$(k, r, m) \mapsto \Big(r, V_k(r, |m|) \oplus m\Big). \qquad (\mathcal{SE}_1) \quad (5)$$

The following proposition originates from [8]. We give the proof for completeness.

Proposition 1. *For any t, q, and μ*

$$\mathbf{Adv}^{\textbf{ind-p2-c0}}_{\mathcal{SE}_1}(t, q, \mu) \leq 2 \cdot \mathbf{Adv}^{\textbf{vol-wprf}}_{V}(t, q, \mu) + \frac{q-1}{2^{n-1}}.$$

Proof. For $\mathcal{SE}_1 = (E, D)$, let $\mathit{\Pi}_0$ denote the IND-P2-C0 random experiment

$$k \stackrel{\$}{\leftarrow} \{0,1\}^\kappa,\ (x_0, x_1) \leftarrow A^{E_k},\ b \stackrel{\$}{\leftarrow} \{0,1\},\ y \leftarrow E_k(x_b),\ \hat{b} \leftarrow A^{E_k}(y),$$

where A is any adversary with resources (t, q, μ). Furthermore, let $\mathit{\Pi}_1$ be the same experiment as $\mathit{\Pi}_0$, except that V_k is replaced by a uniform random VOL-function $\mathbf{R}_{n,*}$. Let $\mathit{\Pi}_2$ be the same experiment as $\mathit{\Pi}_1$, except that the input y

to the adversary is replaced by a truly random string y' (of length $|y|$). For $i = 0, 1, 2$, let S_i denote the event that $\hat{b} = b$ in experiment Π_i. Then

$$\begin{aligned}\mathbf{Adv}^{\textbf{ind-p2-c0}}_{\mathcal{SE}_1,A} &\stackrel{\text{def}}{=} 2 \cdot \Pr[S_0] - 1 = 2 \cdot \Pr[S_2] - 1 + 2 \cdot \sum_{i=0}^{1} (\Pr[S_i] - \Pr[S_{i+1}]) \\ &\leq 2 \cdot \frac{1}{2} - 1 + 2 \cdot \mathbf{Adv}^{\textbf{vol-wprf}}_{V}(t, q, \mu) + 2 \cdot \frac{q-1}{2^n},\end{aligned}$$

where the inequality follows from the following three facts. First, A can be transformed into VOL-WPRF distinguisher A' for V with advantage $\Pr[S_0] - \Pr[S_1]$ and resources (t, q, μ). A' with oracle T simply simulates the experiment Π_0 if T is an instance of V and Π_1 if T is a uniform random VOL-function $\mathbf{R}$ (this is possible as the inputs to V_k in Π_0 and to $\mathbf{R}_{n,*}$ in Π_1 are distributed uniformly at random), and then A' returns whatever A does. Second, Π_1 and Π_2 are equivalent experiments as long as the random input to $\mathbf{R}_{n,*}$ in the computation of y is different from the other random inputs to $\mathbf{R}_{n,*}$, an event upper bounded by $(q-1)/2^n$. Third, $\Pr[S_2] = 1/2$ since b is independent of y. □

Remark 2. Given the strong optimality arguments for ICT, it is clear that (2) is optimal when ICT is used (instead of PRT) unless a significantly different approach for range extension of WPRFs is invented.

4.3 A CCA-Secure Encryption Scheme

The well-known encrypt-then-MAC method is a general technique for constructing an INT-CTXT- and IND-P2-C2-secure encryption scheme from any IND-P2-C0-secure encryption scheme $\mathcal{SE} = (Enc, Dec)$ and any VIL-MAC W. The idea is to simply encrypt with Enc and then authenticate the ciphertext using W [15, 5]. Here, we note that for the IND-P2-C0-secure scheme $\mathcal{SE}_1$ based on any VOL-WPRF $V : \{0,1\}^{\kappa_1} \times \{0,1\}^n \times \mathbb{N} \to \{0,1\}^*$, it is sufficient if $W : \{0,1\}^{\kappa_2} \times \{0,1\}^* \to \{0,1\}^\ell$ is a VIL-WMAC (as the ciphertexts of $\mathcal{SE}_1$ are pseudorandom). To be precise, the scheme $\mathcal{SE}_2$, defined by encrypting $m \in \{0,1\}^*$ under a key $(k_1, k_2) \in \{0,1\}^{\kappa_1} \times \{0,1\}^{\kappa_2}$ and auxiliary uniform randomness $r \in \{0,1\}^n$ as

$$\big((k_1, k_2), r, m\big) \mapsto \Big(r, \underbrace{V_{k_1}(r, |m|) \oplus m}_{c}, W_{k_2}(r \| c)\Big), \qquad (\mathcal{SE}_2) \quad (6)$$

is IND-P2-C2 secure if V is a VIL-WPRF and W is a VIL-WMAC:

Theorem 3. *For any t, q, μ, q', and μ'*

$$\begin{aligned}\mathbf{Adv}^{\textbf{int-ctxt}}_{\mathcal{SE}_2}(t, q, \mu, q', \mu') \leq \min\Big\{ & q' \cdot \mathbf{Adv}^{\textbf{vil-mac}}_{W}(t, q, \mu{+}qn{+}\mu'), \\ & \mathbf{Adv}^{\textbf{vol-wprf}}_{V}(t, q, \mu) + \frac{q^2}{2^{n+1}} + q' \cdot \mathbf{Adv}^{\textbf{vil-wmac}}_{W}(t, q, \mu{+}qn{+}\mu') \Big\}\end{aligned}$$

$$\mathbf{Adv}^{\textbf{ind-p2-c2}}_{\mathcal{SE}_2}(t, q, \mu, q', \mu') \leq 2\,\mathbf{Adv}^{\textbf{int-ctxt}}_{\mathcal{SE}_2}(t, q, \mu, q', \mu') + \mathbf{Adv}^{\textbf{ind-p2-c0}}_{\mathcal{SE}_1}(t, q, \mu).$$

Proof. The proof of the first inequality consists of two parts. For the first part, i.e., $\mathbf{Adv}^{\textbf{int-ctxt}}_{\mathcal{SE}_2}(t,q,\mu,q',\mu') \le q' \cdot \mathbf{Adv}^{\textbf{vil-mac}}_{W}(t,q,\mu+qn+\mu')$, we refer to [5]. For the second part, let $\mathit{\Pi}_0$ denote the INT-CTXT random experiment

$$(k_1,k_2) \xleftarrow{\$} \{0,1\}^{\kappa_1} \times \{0,1\}^{\kappa_2},\ A^{E_{k_1,k_2}, D^*_{k_1,k_2}}$$

for $\mathcal{SE}_2 = (E,D)$ and any adversary A with resources (t,q,μ,q',μ'). Furthermore, let $\mathit{\Pi}_1$ be defined as $\mathit{\Pi}_0$ except that V_{k_1} has been replaced by a uniform random VOL-function $\mathbf{R}_{n,*}$ and let $\mathit{\Pi}_2$ be defined as $\mathit{\Pi}_1$ except that the output of $\mathbf{R}_{n,*}$ is replaced by a truly random string (no matter of the input). For $i = 0,1,2$, let $\mathcal{E}_i$ denote the event that $D^*_{k_1,k_2}$ outputs 1 in $\mathit{\Pi}_i$. Then

$$\mathbf{Adv}^{\textbf{int-ctxt}}_{\mathcal{SE}_2,A} \stackrel{\text{def}}{=} \Pr[\mathcal{E}_0] = \Big(\Pr[\mathcal{E}_0] - \Pr[\mathcal{E}_1]\Big) + \Big(\Pr[\mathcal{E}_1] - \Pr[\mathcal{E}_2]\Big) + \Pr[\mathcal{E}_2]$$
$$\le \mathbf{Adv}^{\textbf{vol-wprf}}_{V}(t,q,\mu) + \frac{(q-1)q}{2^{n+1}} + q' \cdot \mathbf{Adv}^{\textbf{vil-wmac}}_{W}(t,q,\mu+qn+\mu'),$$

due to the following three facts. First, A implies a VOL-WPRF distinguisher A' for V with advantage $|\Pr[\mathcal{E}_0] - \Pr[\mathcal{E}_1]|$ and resources (t,q,μ). A' with oracle T simply simulates $\mathit{\Pi}_0$ if T is an instance of V and $\mathit{\Pi}_1$ if T is a uniform random VOL-function $\mathbf{R}$ (this is possible as the inputs to V_{k_1} in $\mathit{\Pi}_0$ and to $\mathbf{R}_{n,*}$ in $\mathit{\Pi}_1$ are distributed uniformly at random), and then A' outputs 1 if and only if A is successful. Second, $\mathit{\Pi}_1$ and $\mathit{\Pi}_2$ are equivalent experiments unless the auxiliary random r-values returned by the encryption oracle are not all distinct, an event upper bounded by $q(q-1)/2^{n+1}$. Third, from A we can construct a VIL-WMAC-forger A'' for W with advantage $\Pr[\mathcal{E}_2]/q'$ and resources $(t,q,\mu+q\mu+\mu')$. A'' simply picks a random element $i \in \{1,\dots,q'\}$ and starts simulating $\mathit{\Pi}_2$ – except for invoking $D^*_{k_1,k_2}$ on A's queries – by using its own oracle in place of W_{k_2} (this is possible as all inputs to W_{k_2} in $\mathit{\Pi}_2$ are distributed uniformly at random). However, once A makes its i-th query to $D^*_{k_1,k_2}$ (if at all), A'' stops the simulation and returns it as its forgery.

For proving the second inequality, let $\mathit{\Pi}'_0$ denote the IND-P2-C2 experiment

$$(k_1,k_2) \xleftarrow{\$} \{0,1\}^{\kappa_1} \times \{0,1\}^{\kappa_2},$$
$$(x_0,x_1) \leftarrow A^{E_{k_1,k_2},D_{k_1,k_2}},\ b \xleftarrow{\$} \{0,1\}, y \leftarrow E_{k_1,k_2}(x_b),\ \hat{b} \leftarrow A^{E_{k_1,k_2},D_{k_1,k_2}}(y),$$

for $\mathcal{SE}_2 = (E,D)$ and any adversary A with resources (t,q,μ,q',μ'). Without loss of generality, we assume that A does not query D_{k_1,k_2} with an output from E_{k_1,k_2}. Let $\mathit{\Pi}'_1$ be the same experiment as $\mathit{\Pi}'_0$, except that all queries to D_{k_1,k_2} are rejected. Moreover, for $i = 0,1$, let S_i denote the event that $\hat{b} = b$ in $\mathit{\Pi}'_i$. Then

$$\mathbf{Adv}^{\textbf{ind-p2-c2}}_{\mathcal{SE}_2,A} \stackrel{\text{def}}{=} 2 \cdot \Pr[S_0] - 1 = 2 \cdot \Big(\Pr[S_0] - \Pr[S_1]\Big) + 2 \cdot \Pr[S_1] - 1$$
$$\le 2 \cdot \Pr[\mathcal{E}] + \mathbf{Adv}^{\textbf{ind-p2-c0}}_{\mathcal{SE}_2}(t,q,\mu) \le 2 \cdot \Pr[\mathcal{E}] + \mathbf{Adv}^{\textbf{ind-p2-c0}}_{\mathcal{SE}_1}(t,q,\mu),$$

where $\mathcal{E}$ denotes the event that a query to D_{k_1,k_2} in $\mathit{\Pi}'_1$ (or $\mathit{\Pi}'_0$) is a valid ciphertext. The first inequality follows from the the fact that $\mathit{\Pi}'_0$ and $\mathit{\Pi}'_1$ are equivalent experiments unless $\mathcal{E}$ occurs, and that Π_1' is equivalent to the corresponding IND-P2-C0 experiment for $\mathcal{SE}_2$ (in which the VIL-WMAC is superfluous by Proposition 1). It remains to show that

$$\Pr[\mathcal{E}] \leq \mathbf{Adv}^{\textbf{int-ctxt}}_{\mathcal{SE}_2}(t, q, \mu, q', \mu').$$

This is the case as A can trivially be transformed into a INT-CTXT adversary A''' (for $\mathcal{SE}_2$) using the same resources and having advantage $\Pr[\mathcal{E}]$. A''' simply runs A, by answering its encryption queries with its own encryption oracle and rejecting all decryption queries. In addition, A''' forwards A's decryption queries to its D^* oracle. If A presents its challenge input (m_0, m_1), A''' flips a coin b, queries its encryption oracle with m_b, and returns the result to A. □

Remark 3. The above result leads to an interesting open question for further research, namely, how efficient constructions are there of a VIL-WMAC W based on any WPRF F. One approach – for constructing W – would be to first transform F into the PRF $\mathrm{IC}^F : \{0,1\}^{3n} \times \{0,1\}^N \to \{0,1\}^n$ (see Sect. 3.1) and then apply the following rather standard method [25, 23, 6] for constructing a VIL-MAC (and thus also a VIL-WMAC) from any PRF. Simply hash the message using an ε-almost universal (AU) hash function $H : \mathcal{K} \times \{0,1\}^* \to \{0,1\}^N$ (i.e., for all distinct $m, m' \in \{0,1\}^*$, $\Pr[k' \leftarrow \mathcal{K} : H_{k'}(m) = H_{k'}(m')] \leq \varepsilon$ [24]) and then apply IC^F to the result: $W_{k,k'}(x) \stackrel{\text{def}}{=} \mathrm{IC}^F_k \circ H_{k'}(x)$.[13] This method is appealing since H exists unconditionally and IC^F is invoked on "short" inputs (of size N). There are 2^{1-N}-AU hash functions, with $5N$-bit key size and maximal input length 2^N, that should do for most practical applications (see [25]).

Remark 4. By combining (6) with $V = \mathrm{ICT}^F$ and a W (as defined above), we get a CCA-secure encryption scheme from any WPRF F. In [8], Damgård and Nielsen also proposed to use the encrypt-then-MAC method for achieving CCA-security of $\mathcal{SE}_1$. However, their approach for constructing the VIL-MAC from any WPRF introduces a too large overhead for the solution to be practical. The number of applications of the WPRF per evaluation is in the order of the message length. The approach we give in Remark 3 is more efficient using at most N applications of the WPRF independently of the message length, where typically $N \ll n$ (recall that n is the block length of F). Whereas this additive overhead is of little concern for "long" messages, it is an open problem whether it can be improved for "short" messages.

4.4 A Non-Adaptive CCA-Secure Encryption Scheme

To achieve IND-P2-C1-security of $\mathcal{SE}_1$, we note that it is sufficient to WMAC the auxiliary randomness r. This has the advantage (over $\mathcal{SE}_2$) that the WMAC does not need to have VIL. To be precise, for $V : \{0,1\}^{\kappa_1} \times \{0,1\}^n \times \mathbb{N} \to \{0,1\}^*$ and $W : \{0,1\}^{\kappa_2} \times \{0,1\}^n \to \{0,1\}^\ell$, let $\mathcal{SE}_3$ denote the encryption scheme defined

[13] For any $Q : \mathcal{K}' \times \{0,1\}^N \to \{0,1\}^n$ and ϵ-AU hash function $H : \mathcal{K} \times \{0,1\}^* \to \{0,1\}^N$, $\mathbf{Adv}^{\textbf{vil-mac}}_{Q \circ H}(t, q, \mu) \leq \mathbf{Adv}^{\textbf{prf}}_{Q}(t, q) + q(q-1)\varepsilon/2 + 1/2^n$ (see [6]).

by encrypting a message $m \in \{0,1\}^*$ under the key $(k_1, k_2) \in \{0,1\}^{\kappa_1} \times \{0,1\}^{\kappa_2}$ and some auxiliary uniform random string $r \in \{0,1\}^n$ as

$$((k_1, k_2), r, m) \mapsto \Big(r, V_{k_1}(r, |m|) \oplus m, W_{k_2}(r)\Big). \qquad (\mathcal{SE}_3) \quad (7)$$

Theorem 4. *For any t, q, μ, q', and μ'*

$$\mathbf{Adv}^{\textbf{ind-p2-c1}}_{\mathcal{SE}_3}(t, q, \mu, q', \mu') \leq 2 \cdot q' \cdot \mathbf{Adv}^{\textbf{wmac}}_{W}(t, q) + \mathbf{Adv}^{\textbf{ind-p2-c0}}_{\mathcal{SE}_1}(t, q, \mu{+}q\mu').$$

Proof. For $\mathcal{SE}_3 = (E, D)$, let Π_0 denote the IND-P2-C1 random experiment for any adversary A with resources (t, q, μ, q', μ'), i.e.,

$$(k_1, k_2) \stackrel{\$}{\leftarrow} \{0,1\}^{\kappa_1} \times \{0,1\}^{\kappa_2},$$
$$(x_0, x_1) \leftarrow A^{E_{k_1,k_2}, D_{k_1,k_2}}, b \stackrel{\$}{\leftarrow} \{0,1\}, y \leftarrow E_{k_1,k_2}(x_b),\ \hat{b} \leftarrow A^{E_{k_1,k_2}}(y).$$

Let Π_1 be the same same random experiment as Π_0 except for replacing A with an adversary B (described next) that has the same advantage as A and does not issue any query to D_{k_1,k_2} for which the auxiliary random part is the same as for a ciphertext returned previously by E_{k_1,k_2}. To be precise, let $\ell_{\max}$ denote the maximal length of the second input part of the decryption queries issued by A (clearly $\ell_{\max} < \mu'$). The adversary B simply runs A and for each encryption query m issued by A, B appends zeroes such that it is of length $l_{\max}$, i.e., $m' := m \| 0^{\ell_{\max} - |m|}$, and then queries the encryption oracle with m'. On output (r, c', w) from the encryption oracle, B returns $(r, c'[1, |m|], w)$ to A (and stores $(m', (r, c', w))$ in a look-up table). If A queries some decryption query, say (r, c, w'), for which r occurs in the look-up table as $(m', (r, c', w))$, B returns $c \oplus c'[1, |c|] \oplus m'[1, |c|]$ if $w = w'$ and otherwise rejects. When A presents its challenge input (m_0, m_1), B flips a coin b, queries its encryption oracle with m_b, and returns the result to A. Finally, B decides as A does. Further, let Π_2 be the same experiment as Π_1 except that all queries to D_{k_1,k_2} are rejected.

Moreover, for $i = 0, 1$, let S_i denote the event that $\hat{b} = b$ in Π_i. Then

$$\mathbf{Adv}^{\textbf{ind-p2-c1}}_{\mathcal{SE}_3, A} \stackrel{\text{def}}{=} 2 \cdot \Pr[S_0] - 1 = 2 \cdot \Pr[S_2] - 1 + 2 \cdot \sum_{i=0}^{1} \Big(\Pr[S_i] - \Pr[S_{i+1}]\Big)$$
$$\leq \mathbf{Adv}^{\textbf{ind-p2-c0}}_{\mathcal{SE}_3}(t, q, \mu{+}q\mu') + 2 \cdot \Pr[\mathcal{E}] \leq \mathbf{Adv}^{\textbf{ind-p2-c0}}_{\mathcal{SE}_1}(t, q, \mu{+}q\mu') + 2 \cdot \Pr[\mathcal{E}],$$

by the following three facts. First, $\Pr[S_0] = \Pr[S_1]$ as B decides as A does. Second, Π_1 and Π_2 are equivalent experiments unless the event $\mathcal{E}$ occurs that B queries a valid ciphertext to its decryption oracle. It follows that

$$\Pr[S_1] - \Pr[S_2] \leq \Pr[\mathcal{E}] \leq q' \cdot \mathbf{Adv}^{\textbf{wmac}}_{W}(t, q),$$

as B can be transformed to the following forger B' for W with advantage at least $\Pr[\mathcal{E}]/q'$. B' simply picks a random $i \in \{1, \ldots, q'\}$ and starts running B, answering its encryption queries with help of its own oracle and the decryption queries by rejection. When B (if at all) issues its i-th decryption query (r_i, c_i, w_i), B' returns (r_i, w_i) as its forgery (without making any extra calls to its encryption oracle). Third, Π_2 corresponds to the IND-P2-C0 experiment (in which the WMAC W is superfluous by Proposition 1). □

Remark 5. Combining (7) with $V = \mathrm{ICT}^F$ and $W = \mathrm{IC}^F \circ H$ results in an IND-P2-C1-secure scheme based on any WPRF F, but with the advantage that the ε-AU hash function H only is applied on fixed-sized strings (of length n). Alternatively, using $W = \mathrm{IC}^F$ saves the call to H and results in $n/2$ overhead applications on average (as IC^F is then invoked on random inputs).

5 Conclusions

We have proposed two constructions, the Increasing Chain Tree (ICT) and the Increasing Chain (IC). Whereas ICT extends the output length of weak PRFs in an optimal way (within a natural class of extensions) and optimizes Damgård and Nielsen's CPA-secure encryption scheme based on any weak PRF [8], IC is a construction of a regular PRF from any weak PRF that, in particular, reduces the key-material of Naor-Reingold's efficient PRF based on the DDH assumption [20]. By combining IC and ICT, we get a CCA-secure encryption scheme based on any weak PRF that is indeed much more efficient than the CCA-secure scheme proposed by Damgård and Nielsen (especially for "long" messages). It is an open problem to construct efficient schemes for "short" messages. Another interesting question is how to construct efficient weak MACs based on weak PRFs.

Although several highly efficient candidates for weak PRFs exist, none were targeted at this particular security notion explicitly. It is an interesting open problem for further research how much block-cipher design can benefit from this weakening of the desired security goal.

References

1. W. Aiello, S. Rajagopalan, and R. Venkatesan. High-speed pseudorandom number generation with small memory. In *Fast Software Encryption*, volume 1636 of *LNCS*, pages 290–304. Springer, 1999.
2. F. Bao, R. H. Deng, and H. Zhu. Variations of Diffie-Hellman problem. In *ICICS '03*, volume 2836 of *LNCS*, pages 301–312. Springer, 2003.
3. M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. A concrete security treatment of symmetric encryption. In *Proc. of the 38th Symposium on Foundations of Computer Science*, pages 394–403. IEEE, 1997.
4. M. Bellare, J. Kilian, and P. Rogaway. The security of cipher block chaining. In *Advances in Cryptology — CRYPTO '94*, volume 839 of *LNCS*, pages 341–358. Springer, 1994.
5. M. Bellare and C. Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In *Advances in Cryptology — ASIACRYPT '00*, volume 1976 of *LNCS*, pages 531–545. Springer, 2000.

6. J. Black, S. Halevi, H. Krawczyk, T. Krovetz, and P. Rogaway. Umac: Fast and secure message authentication. In *Advances in Cryptology — CRYPTO '99*, volume 1666 of *LNCS*, pages 313–328. Springer, 1999.
7. A. Blum, M. L. Furst, M. J. Kearns, and R. J. Lipton. Cryptographic primitives based on hard learning problems. In *Advances in Cryptology — CRYPTO '93*, volume 773 of *LNCS*, pages 278–291. Springer, 1993.
8. I. Damgård and J. B. Nielsen. Expanding pseudorandom functions; or: From known-plaintext security to chosen-plaintext security. In *Advances in Cryptology — CRYPTO '02*, volume 2442 of *LNCS*, pages 449–464. Springer, 2002.
9. W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, 1976.
10. O. Goldreich. *Foundations of Cryptography – Volume II – Basic Applications*. Cambridge University Press, 2004.
11. O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986.
12. J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.
13. J. Katz and M. Yung. Complete characterization of security notions for probabilistic private-key encryption. In *Proc. of the 32nd Annual Symposium on Theory of Computing*, pages 245–254. ACM, 2000.
14. M. Keller. Constructing weak pseudorandom functions with prescribed structure, 2006. Semester Thesis, ETH Zurich.
15. S. Kent and R. Atkinson. IP encapsulating security payload (ESP), November 1998. Request for Comments 2406.
16. K. Minematsu and Y. Tsunoo. Expanding weak PRF with small key size. In *ICISC '05*, volume 3935 of *LNCS*, pages 284–298. Springer, 2005.
17. M. Naor, B. Pinkas, and O. Reingold. Distributed pseudo-random functions and KDCs. In *Advances in Cryptology — EUROCRYPT '99*, volume 1592 of *LNCS*, pages 327–346. Springer, 1999.
18. M. Naor and O. Reingold. From unpredictability to indistinguishability: A simple construction of pseudo-random functions from MACs. In *Advances in Cryptology — CRYPTO '98*, LNCS, pages 267–282. Springer, 1998.
19. M. Naor and O. Reingold. Synthesizers and their application to the parallel construction of pseudo-random functions. *J. Comp. Sys. Sci.*, 58(2):336–375, 1999.
20. M. Naor and O. Reingold. Number-theoretic constructions of efficient pseudo-random functions. *J. of the ACM*, 51(2):231–262, 2004.
21. K. Pietrzak and J. Sjödin. Weak pseudorandom functions in minicrypt, November 2006. Manuscript.
22. K. Pietrzak and J. Sjödin. Domain extension for weak PRFs; the good, the bad, and the ugly. In *Advances in Cryptology — EUROCRYPT '07*, LNCS. Springer, 2007. This proceedings.
23. V. Shoup. On fast and provably secure message authentication based on universal hashing. In *Advances in Cryptology — CRYPTO '96*, volume 1109 of *LNCS*, pages 313–328. Springer, 1996.
24. D. R. Stinson. Universal hashing and authentication codes. In *Advances in Cryptology — CRYPTO '91*, volume 576 of *LNCS*, pages 74–85. Springer, 1992.
25. M. N. Wegman and J. L. Carter. New hash functions and their use in authentication and set equality. *J. Comp. Sys. Sci.*, 22:265–279, 1981.