Secure Hashed Diffie-Hellman over Non-DDH Groups

Rosario Gennaro[1], Hugo Krawczyk[2], and Tal Rabin[1]

[1] IBM T.J.Watson Research Center, PO Box 704, Yorktown Heights, NY 10598, USA. E-mail: {rosario,talr}@watson.ibm.com

[2] Department of Electrical Engineering, Technion, Haifa 32000, Israel, and IBM T.J. Watson Research Center, New York, USA. E-mail: hugo@ee.technion.ac.il

Abstract. The Diffie-Hellman (DH) transform is a basic cryptographic primitive used in innumerable cryptographic applications, most prominently in discrete-log based encryption schemes and in the Diffie-Hellman key exchange. In many of these applications it has been recognized that the direct use of the DH output, even over groups that satisfy the strong Decisional Diffie-Hellman (DDH) assumption, may be insecure. This is the case when the application invoking the DH transform requires a value that is pseudo-randomly distributed over a set of strings of some length rather than over the DH group in use. A well-known and general solution is to hash (using a universal hash family) the DH output; we refer to this practice as the "hashed DH transform".

The question that we investigate in this paper is to what extent the DDH assumption is required when applying the hashed DH transform. We show that one can obtain a secure hashed DH transform over a non-DDH group G (i.e., a group in which the DDH assumption does not hold); indeed, we prove that for the hashed DH transform to be secure it suffices that G contain a sufficiently large DDH *subgroup*. As an application of this result, we show that the hashed DH transform is secure over Z_p^* for random prime p, provided that the DDH assumption holds over the large prime-order subgroups of Z_p^*. In particular, we obtain the same security working directly over Z_p^* as working over prime-order subgroups, without requiring any knowledge of the prime factorization of $p-1$ and without even having to find a generator of Z_p^*.

Further contributions of the paper to the study of the DDH assumption include: the introduction of a DDH relaxation, via computational entropy, which we call the "t-DDH assumption" and which plays a central role in obtaining the above results; a characterization of DDH groups in terms of their DDH subgroups; and the analysis of of the DDH (and t-DDH) assumptions when using short exponents.

1 Introduction

The Diffie-Hellman Transform and DDH Assumption. The *Diffie-Hellman transform* is one of the best-known and fundamental cryptographic primitives. Its discovery by Whitfield Diffie and Martin Hellman [DH76] revolutionized the science of cryptography and marked the birth of Modern Cryptography. Even today, almost 30 years later, the DH transform remains the basis of some of the most widely used cryptographic techniques. In particular, it underlies the Diffie-Hellman key exchange and the ElGamal encryption scheme [ElG85], and is used over a large variety of mathematical groups. In its basic form the Diffie-Hellman (or DH for short) transform maps a pair of elements g^a, g^b drawn from a cyclic group G generated by the element g into the group element[1] g^{ab}. The usefulness of this transform was originally envisioned under the conjecture, known as the *Computational Diffie-Hellman (CDH)* assumption, that states the infeasibility of computing the value g^{ab} given only the exponentials g^a and g^b. Namely, the value g^{ab} should be computable only by those knowing one of the exponents a or b. Note that the CDH assumption implies the difficulty of computing discrete logarithms over the group G (the converse, however, is unknown for most practical groups).

Over time it was realized that the CDH assumption is insufficient to guarantee the security of most DH applications (in particular those mentioned above). For this reason a much stronger assumption was introduced: the *Decisional Diffie-Hellman (DDH)* assumption postulates that given the values g^a and g^b not only it is computationally hard to derive the value g^{ab} but even the seemingly much easier task of distinguishing g^{ab} from random group elements is infeasible [Bra93] (see [Bon98] for a survey on the DDH assumption). On the basis of this assumption one can consider the DH transform as a good generator of pseudorandomness as required in key-exchange, encryption and other cryptographic applications. Hereafter we refer to groups in which the DDH assumption holds as *DDH groups*. The need to rely on the DDH disqualifies many natural groups where the assumption does not hold. For example, any group whose order is divisible by small factors, such as the classic groups Z_p^* of residues modulo a large prime p; in this case the group's order, $p-1$, is always divisible by 2 and thus the DDH assumption does not hold. Moreover, for randomly generated primes p, $p-1$ has (with very high probability) additional small factors. Due to the *perceived* need to work over DDH groups it is often recommended in the cryptographic literature that one work over subgroups of large prime order where no attacks are known on the DDH assumption.

The Need for Hashing the Diffie-Hellman Result. Interestingly, the DDH assumption, while apparently necessary, turns out to be insufficient for guaranteeing the security of some of the most basic applications of the DH transform. Consider for example the ElGamal encryption scheme: Given a public key $y = g^a$ (for secret a), a message $m \in G$ is encrypted by the pair (g^b, my^b) where the value b is chosen randomly anew for each encryption. In this case, the DDH assumption guarantees the semantic security ([GM84]) of the scheme (against chosen-plaintext attacks) *provided that the plaintexts m are elements of the group G*. However, if the message space is different, e.g. the set of strings of some length smaller than $\log(|G|)$, then the above encryption scheme becomes problematic. First of all, you need to encode messages m as group elements in G and that could be cumbersome. If G is a subgroup of prime order of Z_p^*, a naive (and common) approach would be to trivially encode m as an integer and perform the multiplication my^b modulo p. But now the scheme is *insecure even if the*

[1] Here we use the exponential notation that originates with multiplicative groups but our treatment applies equally to additive groups such as Elliptic Curves.

group G does satisfy the DDH assumption. A good illustration of the potential weaknesses of this straightforward (or "textbook") application of ElGamal is presented in [BJN00]. It is shown that if the space of plaintexts consists of random strings of length shorter than $|G|$ (e.g., when using public key encryption to encrypt symmetric keys) the above scheme turns out to be insecure even under a ciphertext-only attack and, as said, even if the group G is DDH. For example, if the plaintexts to be encrypted are keys of length 64, an attacker that sees a ciphertext has a significant probability of finding the plaintext with a work factor in the order of 2^{32} operations and comparable memory; for encrypted keys of length 128 the complexity of finding the key is reduced to 2^{64}.

A general and practical approach to solving these serious security weaknesses is to avoid using the DH value itself to "mask" m via multiplication, but rather to *hash* the DH value g^{ab} to obtain a pseudorandom key K of suitable length which can then be used to encrypt the message m under a particular encryption function (in particular, K can be used as a one-time pad). In this case the hash function is used to *extract the (pseudo) randomness* present in the DH value. Suitable hash functions with provable extraction properties are known, for example *universal hash functions* [CW79,HILL99]. The above considerations are common to many other applications of the DH transform, including encryption schemes secure against chosen-ciphertext attacks [CS98] and, most prominently, the Diffie-Hellman key-exchange protocol (in the latter case one should not use the DH output as a cryptographic key but rather derive the agreed shared keys via a hashing of the DH result); see Section 3.2 for a discussion on how these applications choose a random hash function out of a given family. For additional examples and justification of the need for hashing the DH output see [Bon98,NR97,CS98,ABR01]. In the sequel we refer to the combination of the DH transform with a (universal) hash function as the *hashed DH transform.*

1.1 Our Results

The Security of the Hashed DH Transform over non-DDH Groups. In light of the need to hash the DH value, some natural questions arise: when applying the hashed DH transform, is it still necessary to work over groups where the DDH assumption holds, or can this requirement be relaxed? Can one obtain a secure (hashed) DH transform over a non-DDH group, and specifically, is doing hashed DH over Z_p^* secure? In this paper we provide answers to these questions. Our main result can be informally stated as follows: *For any cyclic group G, applying the hashed DH transform over G has the same security as applying the hashed DH transform directly over the <u>maximal</u> DDH subgroup of G.* In particular, one can obtain secure applications of the hashed DH transform over non-DDH groups; the only requirement is that G contain a (sufficiently large) DDH subgroup (see below for the exact meaning of "sufficiently large" and other parameter size considerations). A significant point is that we are only concerned with the *existence* of such a subgroup; there is no need to know the exact size or structural properties of, nor to be able to construct, this specific (maximal) DDH subgroup.

A particularly interesting consequence of the above result is that assuming that DDH holds on large subgroups of Z_p^* (we will see later that it is sufficient to assume that DDH holds on large prime-order subgroups of Z_p^*), one can build secure (hashed) DH applications working directly over Z_p^*, where p is an unconstrained random prime. Only the length of the prime is specified, while other common requirements such as the knowledge of the partial or full factorization of $p-1$, insisting that $p-1$ has a prime factor of a particular size, or disqualifying primes for which $(p-1)/2$ has a smooth part, are all avoided here. Moreover, we show that there is no need to find a generator of Z_p^*; instead we prove that a randomly chosen element from Z_p^* will span a (probably non-DDH) subgroup with a large enough DDH subgroup. In particular, the DH security is preserved even if the order of the chosen element has small factors or if it misses some prime divisors of $p-1$. Note that avoiding the need to find a generator for Z_p^* allows us to work with primes p with unknown factorization of $p-1$ (which is otherwise required to find a Z_p^* generator).

The t-DDH Assumption. In order to prove our main result (i.e., that the hashed DH transform is secure over any group G, not necessarily a DDH group, that contains a large enough DDH subgroup), we introduce a relaxation of the DDH assumption which we call the *t-DDH assumption.* Informally, a group G satisfies the t-DDH assumption (where $0 \leq t \leq |G|$) if given the pair (g^a, g^b) (where g is a generator of G) the value g^{ab} contains t *bits of computational entropy.* The notion of computational entropy, introduced in [HILL99], captures the amount of computational hardness present in a probability distribution. In other words, we relax the "full hardness" requirement at the core of the DDH assumption, and assume partial hardness only. Moreover, we do not care about the exact subsets of bits or group elements where this hardness is contained, but only assume their existence. On this basis, and using the entropy-smoothing theorem from [HILL99] (also known as the leftover hash lemma), we obtain a way to efficiently transform (via universal hashing) DH values over groups in which the t-DDH assumption holds into shorter outputs that are computationally indistinguishable from the uniform distribution. The maximal length of (pseudorandom) strings that one can obtain as output from the hashed DH transform depends on the maximum value of t for which the t-DDH holds in G. In particular, in order to be 2^{-k}-computationally close to uniform one can output up to $t-2k$ pseudorandom bits (e.g., to produce 128-bit keys with a security parameter of $k=80$ the group G should be 288-DDH, while for $k=128$, G is to be 384-DDH).

After defining the t-DDH assumption and showing its usefulness in extracting random bits from t-DDH groups, we show that *if G contains a DDH subgroup of order m then G is $|m|$-DDH.* This forms the basis for our main result as stated above. Indeed, it suffices that G has a suitably large-order DDH subgroup to ensure that hashing the DH output results in pseudorandom outputs of the required length. Again, it is important to stress that we do not need to know the specific DDH subgroup or its order, only (assume) its existence.

A Direct Product Characterization of the DDH Assumption. A further contribution of our work is in providing a characterization of the DDH assumption in a given group in terms of its DDH subgroups. Specifically, we show that *a group is DDH if and only if it is the direct product of (disjoint) prime power DDH groups.* In other words, a group G is DDH if and only if all its prime power subgroups are DDH. Moreover, for any cyclic group G, the maximal DDH group in G is obtained as the product of all prime power DDH subgroups in G. Beyond its independent interest, this result plays a central role in our proof that the hashed DH transform over Z_p^* is secure as long as the DDH assumption holds in the subgroups of Z_p^* of large prime order. In particular, this allows us to expand significantly the groups in which one can work securely with the hashed DH transform without having to strengthen the usual assumption that DDH holds in large prime order subgroups.

Some Practical Considerations. Beyond the theoretical interest in understanding the role of the DDH assumption and proving the usefulness of relaxed assumptions, our results point out some practical issues that are worth discussing. In this respect, one significant contribution is the justification of the use of non-DDH groups in applications of DH that hash their output. It needs to be noted that in spite of an extensive crypto literature regarding the use of prime order subgroups for performing DH, many real-world instantiations of this primitive work over non-DDH groups (e.g. Z_p^*). Examples include the widespread SSH and IPsec standards. Interestingly, the latter has standardized a set of groups for use with the IKE Diffie-Hellman key-exchange protocols [RFC2409], none of which constitute a DDH group. However, since the IKE protocol takes care of hashing the output of the DH transform before generating the cryptographic keys (see [Kra03]), then our results serve to justify the security of this mechanism[2].

In addition, and as pointed out before, our results also show that under the sole assumption that the DDH holds in groups of large prime order one can work directly over Z_p^* for a random prime p, without having to know the factorization of $p-1$ and without having to find a generator of Z_p^*. Moreover, the ability to work over non-prime order groups has the benefit of eliminating the attacks on the DH transform described in [LL97], without having to search for primes of a special form (and without necessitating special parameter checks when certifying public keys [LL97]).

Short-Exponent Diffie-Hellman. One important practical consideration is the length of exponents used when applying the DH transform. Full exponents when working over Z_p^* are, typically, of size 1024 or more. Even if one works over a prime-order subgroup, one still needs to use relatively large orders (e.g. 288-bit long primes), with their correspondingly large exponents, to ensure a hashed output (say of 128 bits) that is indistinguishable from uniform. (This requirement for large computational entropy is often overlooked; indeed, the usual practice

[2] In IKE, the family of hash functions used for extracting a pseudorandom key from the DH value are implemented using common pseudorandom function families keyed with random, but known, keys. The randomness extraction properties of the latter families are studied in [GHKR04].

of using 160-bit prime-order groups, which originates with Schnorr's signatures, is inappropriate for hashed DH-type applications.)

Motivated by the significant cost of exponentiation using long exponents, we investigate whether one can use short exponents (e.g. as in [RFC2409]) and still preserve the security of the hashed DH transform. An obviously necessary requirement for the short exponent practice to be secure is the assumption that the discrete log problem is hard when exponents are restricted to a short length (say of s bits). We show that this requirement (called the s-DLSE assumption) is sufficient for the secure use of short exponents in the setting of the DH transform; more precisely, we prove (based on [Gen00]) that if the s-DLSE assumption holds in a group G, then the hashed DH transform in G is as secure with full exponents as with s-bit exponents. As a consequence, one can analyze the security of the hashed DH transform in the group G with full exponents and later replace the full exponents with much shorter ones without sacrificing security. In this case the important parameter is s; we note that the appropriate value of s depends on the underlying group. See [vOW96] for an extensive study of the plausible value of s for different groups.

Paper's Organization. In Section 2 we recall the DDH Assumption and prove the DDH Characterization Theorem. In Section 3 we introduce the t-DDH Assumption and its application to the hashed DH transform, and prove the central Max-Subgroup Theorem. In Section 4 we investigate the security of the hashed DH transform when using short exponents. We conclude in Section 5 by describing the applicability of our results to the hashed DH transform over non-DDH groups.

Notation. The formal treatment in this paper often involves sequences of probability distributions $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$ to which we refer as *probability ensembles* (or simply as "ensembles"). We adopt the convention that by the "probability distribution $\mathcal{D}_n$" we mean the specific element (distribution) $\mathcal{D}_n$ in the above sequence, while the term "probability ensemble $\mathcal{D}_n$" is short for "probability ensemble $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$". We also assume that each distribution $\mathcal{D}_n$ is taken over a set $A_n \subset \{0,1\}^{n'}$ where n' is polynomial in n (i.e., each ensemble has a fixed polynomial in n that determines the value n'). The notation $x \in_{\mathcal{D}_n} A_n$ is to be read as x chosen in A_n according to the distribution $\mathcal{D}_n$, and $x \in_R S$ means choosing x with uniform distribution over the set S. Finally if m is an integer, we denote with $|m|$ its binary length.

2 A Direct-Product DDH Characterization

We consider a (infinite) family of cyclic groups $\mathcal{G} = \{G_n\}_n$. Denote with g_n and m_n a generator and the order of G_n, respectively, where $|m_n|$ is bounded by a polynomial in n.

Consider the following problem: Given a pair g_n^a, g_n^b compute the value g_n^{ab}. If this problem is intractable over a family $\mathcal{G}$ then we say that the Computational Diffie-Hellman (CDH) assumption holds (over $\mathcal{G}$).

A much stronger, but also more useful, assumption is the following. Consider the family of sets $G_n^3 = G_n \times G_n \times G_n$ and the following two probability ensembles over it:

$$\mathcal{R}_n = \{(g_n^a,\ g_n^b,\ g_n^c) \ \text{ for } \ a, b, c \in_R [0..m_n]\}$$

and

$$\mathcal{DH}_n = \{(g_n^a,\ g_n^b,\ g_n^{ab}) \ \text{ for } \ a, b \in_R [0..m_n]\}$$

Definition 1. *We say that the* Decisional Diffie-Hellman (DDH) Assumption *holds over $\mathcal{G}$ if the ensembles $\mathcal{R}_n$ and $\mathcal{DH}_n$ are computationally indistinguishable (with respect to non-uniform distinguishers)*[3]*. If $\mathcal{G}$ satisfies the DDH assumption, we call $\mathcal{G}$ a* DDH group (family).

Informally what the above assumption requires is that no polynomial time judge can *decide* if the third element of the triple (g_n^a, g_n^b, g_n^c) is the result of the Diffie-Hellman transform applied to g_n^a, g_n^b or a randomly chosen group element. Clearly this is a much weaker requirement from the attacker than computing the value g_n^{ab} from g_n^a, g_n^b. And therefore, as a general hardness assumption, DDH is (much) stronger than the CDH.

The group family $\mathcal{G}$ over which the two distributions $\mathcal{R}_n$ and $\mathcal{DH}_n$ are defined is very important and indeed it makes a difference for the validity of the assumption.

Example 1: *A group where the DDH assumption does not hold.* Consider the following group family; for each n take an n-bit prime p_n and the group $G_n = Z_{p_n}^*$. Since testing for quadratic residuosity over $Z_{p_n}^*$ is easy, by computing $(\frac{\cdot}{p_n})$ (the Legendre symbol), then we immediately get a distinguisher against DDH in this group: by mapping the Legendre symbol of 1 (i.e. quadratic residues) to 0, and the Legendre symbol of -1 to 1, we can simply check that $(\frac{g_n^a}{p_n})(\frac{g_n^b}{p_n}) = (\frac{g_n^c}{p_n})$, and output "$\mathcal{DH}_n$" if it holds and "$\mathcal{R}_n$" otherwise. Clearly, if the triple is a legal DH triple then the distinguisher outputs $\mathcal{DH}_n$ with probability 1, while in the other case the probability is only 1/2.

Example 2: *A group where the DDH is conjectured to hold.* For each integer n consider an n-bit prime q_n and $poly(n)$-bit prime p_n such that q_n divides $p_n - 1$. The group G_n is the subgroup of prime order q_n in $Z_{p_n}^*$. In this case no efficient distinguisher against the DDH is known.

An important remark about our formalism. We assume a notion of computational indistinguishability under non-uniform distinguishers. In particular, such a distinguisher may be given an "auxiliary input" for each group G_n in the family $\mathcal{G}$. This approach allows us to keep the simplicity of arguments in the asymptotic polynomial-time model while capturing the fact that we are interested in the security of individual groups for which the attacker may have some side information. A particularly important example of such "side information" is

[3] The notion of computational indistinguishability is recalled in Appendix A; see also the remark below regarding our non-uniform formalism.

the possible knowledge by the attacker of the group order and its factorization. Our results do assume that such factorization may be given to the attacker (as part of the non-uniform auxiliary input). In particular, this assumption plays an important role in the proof of the following theorem, which does not necessarily hold when the factorization of $ord(G)$ is unknown (as it may be the case when working over Z_N^* where $N = pq$ is a modulus of unknown factorization).
Due to our focus on the security of specific groups we will often omit the subscript n in the notation of groups, generators, etc.

The next theorem provides a full characterization of DDH groups in terms of their prime order subgroups (as remarked above, the proof of this theorem assumes that the distinguisher is given the factorization of $ord(G)$).

Theorem 1 (Direct Product Characterization Theorem.). *A cyclic group G is DDH if and only if all its prime power subgroups are DDH.*

The proof follows from Lemmas 1 and 2.

Lemma 1. *If the DDH assumptions holds in a group G then it holds in all the subgroups of G.*

Proof. Let G be a DDH (cyclic) group of order order $m = m_1 m_2$, and let G_1 be a subgroup of G of order m_1. Let g be a generator of G and $g_1 = g^{m_2}$ be a generator of G_1. Assume by contradiction that the DDH does not hold in G_1, i.e. there is a distinguisher D_1 that upon receiving a triple $(A_1 = g_1^{a_1}, B_1 = g_1^{b_1}, C_1 = g_1^{c_1}) \in G_1^3$, can distinguish whether it came from the distribution $\mathcal{R}_{G_1}$ or $\mathcal{DH}_{G_1}$ with non-negligible advantage ϵ. We build a distinguisher D for G which distinguishes between the distributions $\mathcal{DH}_G$ and $\mathcal{R}_G$ with the same probability ϵ.

Upon receiving a triple $(A = g^a, B = g^b, C = g^c)$, where $a, b \in_R Z_{m_1 m_2}$ and c is either the product of ab or picked uniformly at random in $Z_{m_1 m_2}$, the distinguisher D :

1. Computes (A_1, B_1, C_1) by setting $A_1 = A^{m_2}, B_1 = B^{m_2}$, and $C_1 = C^{m_2}$.
2. Passes the triple (A_1, B_1, C_1) to D_1
3. Outputs the same output bit as D_1.

Note that by construction the values A_1, B_1, C_1 equal $g_1^{a_1}, g_1^{b_1}, g_1^{c_1}$, respectively, where $a_1 = a \bmod m_1, b_1 = b \bmod m_1, c_1 = c \bmod m_1$. Since $a, b \in_R Z_{m_1 m_2}$ then $a_1, b_1 \in_R Z_{m_1}$. Also, if $c = ab \bmod m_1 m_2$ then $c_1 = a_1 b_1 \bmod m_1$, while if $c \in_R Z_{m_1 m_2}$ then $c_1 \in_R Z_{m_1}$. In other words, whenever the triple (A, B, C) is distributed according to $\mathcal{DH}_G$ then the triple (A_1, B_1, C_1) is distributed according to $\mathcal{DH}_{G_1}$, while if (A, B, C) is distributed according to $\mathcal{R}_G$ then the triple (A_1, B_1, C_1) is distributed according to $\mathcal{R}_{G_1}$. Therefore, D distinguishes between the distributions $\mathcal{DH}_G$ and $\mathcal{R}_G$ with the same probability ϵ that D_1 distinguishes between $\mathcal{DH}_{G_1}$ and $\mathcal{R}_{G_1}$. □

Lemma 2. *Let G be a cyclic group of order $m = m_1 m_2$, where $(m_1, m_2) = 1$, and let G_1 and G_2 be the subgroups of G of orders m_1, m_2 resp. If DDH holds in G_1 and G_2 then DDH holds in G.*

Proof. Let g, g_1, g_2 be generators of G, G_1, and G_2, respectively; in particular, $g_1 = g^{m_2}$ and $g_2 = g^{m_1}$. Given a triple $t_1 = (A_1 = g_1^{a_1}, B_1 = g_1^{b_1}, C_1 = g_1^{c_1}) \in G_1^3$ and a triple $t_2 = (A_2 = g_2^{a_2}, B_2 = g_2^{b_2}, C_2 = g_2^{c_2}) \in G_2^3$ we define the following transformation T which "lifts" this pair of triples into a triple in G^3. (T is the standard isomorphism between the group G and its product group representation as determined by the Chinese Reminder Theorem.) On input t_1, t_2, $T(t_1, t_2)$ outputs a triple $(A = g^a, B = g^b, C = g^c) \in G^3$ defined as follows:

1. Let r_1, r_2 be such that $r_1 m_1 + r_2 m_2 = 1$ (i.e., $r_1 = m_1^{-1} \bmod m_2$ and $r_2 = m_2^{-1} \bmod m_1$)
2. Set $A = A_1^{r_2} A_2^{r_1} = g^{a_1 m_2 r_2 + a_2 m_1 r_1} \in G$, i.e. $a = a_1 m_2 r_2 + a_2 m_1 r_1 \bmod m$
3. Set $B = B_1^{r_2} B_2^{r_1} = g^{b_1 m_2 r_2 + b_2 m_1 r_1} \in G$, i.e. $b = b_1 m_2 r_2 + b_2 m_1 r_1 \bmod m$
4. Set $C = C_1^{m_2 r_2^2} C_2^{m_1 r_1^2} = g^{c_1 m_2^2 r_2^2 + c_2 m_1^2 r_1^2} \in G$, i.e. $c = c_1 m_2^2 r_2^2 + c_2 m_1^2 r_1^2 \bmod m$

Note the following facts about the triple (A, B, C) which result from the above transformation:

Fact 1 If $a_1, b_1 \in_R Z_{m_1}$, and $a_2, b_2 \in_R Z_{m_2}$, then $a, b \in_R Z_m$.
Fact 2 $c - ab \equiv c_1 - a_1 b_1 \bmod m_1$ and $c - ab \equiv c_2 - a_2 b_2 \bmod m_2$
Fact 3 Following Facts 1 and 2, if the triple t_1 is chosen according to distribution $\mathcal{DH}_{G_1}$ and t_2 according to distribution $\mathcal{DH}_{G_2}$, then the triple (A, B, C) is distributed according to the distribution $\mathcal{DH}_G$. Similarly, if t_1, t_2 are distributed according to $\mathcal{R}_{G_1}$ and $\mathcal{R}_{G_2}$, respectively, then (A, B, C) is distributed according to $\mathcal{R}_G$.

For probability distributions $\mathcal{P}_1, \mathcal{P}_2$ we denote by $T(\mathcal{P}_1, \mathcal{P}_2)$ the probability distribution induced by the random variable $T(x_1, x_2)$ where x_1, x_2 are random variables distributed according to $\mathcal{P}_1, \mathcal{P}_2$, respectively, and T is the above defined transform. Using this notation and Fact 3 we get: $\mathcal{DH}_G = T(\mathcal{DH}_{G_1}, \mathcal{DH}_{G_2})$ and $\mathcal{R}_G = T(\mathcal{R}_{G_1}, \mathcal{R}_{G_2})$.

Let us now consider the "hybrid" probability distribution $T(\mathcal{R}_{G_1}, \mathcal{DH}_{G_2})$. Note that this distribution is computationally indistinguishable from $T(\mathcal{DH}_{G_1}, \mathcal{DH}_{G_2})$. Indeed, since the distribution $\mathcal{DH}_{G_2}$ is efficiently samplable and the transformation T is efficiently computable, then one can transform any efficient distinguisher between the above two distributions into an efficient distinguisher between $\mathcal{R}_{G_1}$ and $\mathcal{DH}_{G_1}$, in contradiction to the Lemma's premise that the distributions $\mathcal{R}_{G_1}$ and $\mathcal{DH}_{G_1}$ are indistinguishable. Similarly, we have that the hybrid distribution $T(\mathcal{R}_{G_1}, \mathcal{DH}_{G_2})$ is indistinguishable from $T(\mathcal{R}_{G_1}, \mathcal{R}_{G_2})$. Summarizing, we have that:

$$\mathcal{DH}_G = T(\mathcal{DH}_{G_1}, \mathcal{DH}_{G_2}) \overset{c}{\approx} T(\mathcal{R}_{G_1}, \mathcal{DH}_{G_2}) \overset{c}{\approx} T(\mathcal{R}_{G_1}, \mathcal{R}_{G_2}) = \mathcal{R}_G$$

where $\overset{c}{\approx}$ denotes computational indistinguishability. Therefore by a standard hybrid argument (or the triangle inequality for computational indistinguishability) we get that, provided that the DDH assumption holds in G_1 and G_2, then $\mathcal{DH}_G$ and $\mathcal{R}_G$ are computationally indistinguishable, i.e. G is DDH. □

3 The t-DDH Assumption and the Hashed DH Transform

In this section we introduce an intractability assumption that is, in general, weaker than the DDH assumption, yet it suffices for ensuring DH outputs from which a large number of pseudorandom bits can be extracted. We start by recalling the notions of computational entropy and entropy smoothing. We use the notations introduced at the end of Section 1.

3.1 Computational Entropy and Entropy Smoothing

Definition 2. *Let $\mathcal{X}_n$ be a probability ensemble over A_n. The min-entropy of $\mathcal{X}_n$ is the value*

$$\mathsf{min\text{-}ent}(\mathcal{X}_n) = \mathsf{min}_{x \in A_n : Prob_{\mathcal{X}_n}[x] \neq 0}(-\log(Prob_{\mathcal{X}_n}[x]))$$

Note that if $\mathcal{X}_n$ has min-entropy $t(n)$ then for all $x \in A_n$, $Prob_{\mathcal{X}_n}[x] \leq 2^{-t(n)}$.

The notion of min-entropy provides a measurement of the amount of randomness present in a probability distribution. Indeed, the Entropy Smoothing Theorem (see below) shows that if $\mathcal{X}_n$ has min-entropy $t(n)$ it is possible to construct from $\mathcal{X}_n$ an (almost) uniform distribution over (almost) $t(n)$ bits, by simply hashing elements chosen according to $\mathcal{X}_n$. The basic hashing tool to do this uses the following notion of universal hashing.

Definition 3. *Let $\mathcal{H}_n$ be a family of functions, where each $H \in \mathcal{H}_n$ is defined as $H : A_n \to \{0,1\}^{m(n)}$. We say that $\mathcal{H}_n$ is a* family of (pairwise-independent) universal hash functions *if, for all $x, x' \in A_n$, $x \neq x'$, and for all $a, a' \in \{0,1\}^{m(n)}$ we have*

$$Prob_{H \in \mathcal{H}_n}[H(x) = a \text{ and } H(x') = a'] = 2^{-2m(n)}.$$

That is, a randomly chosen H will map any pair of distinct elements independently and uniformly.

Our techniques use as a central tool the following Entropy Smoothing Theorem from [HILL99] (see also [Lub96]). The definition of statistical distance used below is recalled in Appendix A.

Theorem 2 (Entropy Smoothing Theorem [HILL99].**).** *Let t be a positive integer and let $\mathcal{X}$ be a random variable defined on $\{0,1\}^n$ such that* $\mathsf{min\text{-}ent}(\mathcal{X}) > t$. *Let $k > 0$ be an integer parameter. Let $\mathcal{H}$ be a family of universal hash functions such that $h \in \mathcal{H}$, $h : \{0,1\}^n \to \{0,1\}^{t-2k}$. Let $\mathcal{U}$ be the uniform distribution over $\{0,1\}^{t-2k}$. Then, the distributions $[< h(\mathcal{X}), h >]_{h \in_R \mathcal{H}}$ and $[< \mathcal{U}, h >]_{h \in_R \mathcal{H}}$ have statistical distance at most $2^{-(k+1)}$.*

Thus, the Entropy Smoothing Theorem guarantees that if $\mathcal{X}_n$ is a probability ensemble over A_n with min-entropy of at least $t(n)$, and $\mathcal{H}_n$ a family of universal hash functions from A_n to $\{0,1\}^{t(n)-2k(n)}$, then the random variable $H(x)$, where $H \in_R \mathcal{H}_n$ and x is chosen according to the distribution $\mathcal{X}_n$, is "almost" uniformly distributed over $\{0,1\}^{t(n)-2k(n)}$ even when the hash function H is given. Here, "almost" means a statistical distance of at most $2^{-k(n)}$. Therefore, if one sets $k(n) = \omega(\log n)$, then the statistical distance of $H(x)$ from uniform becomes negligible.

The following notion represents a computational analogue of the notion of min-entropy; it is due to [HILL99].

Definition 4. *A probability ensemble $\mathcal{Y}_n$ has* computational entropy $t(n)$ *if there exists a probability ensemble $\mathcal{X}_n$ such that*

- $\mathsf{min\text{-}ent}(\mathcal{X}_n) \geq t(n)$
- *$\mathcal{X}_n$ and $\mathcal{Y}_n$ are computationally indistinguishable*

Using a standard hybrid argument it is easy to show that the Entropy Smoothing Theorem, as discussed above, can be generalized to probability ensembles $\mathcal{X}_n$ that have computational entropy $t(n)$. In this case, applying a (randomly chosen) universal hash function with $k(n) = \omega(\log n)$ to $\mathcal{X}_n$ results in a *pseudorandom* ensemble, namely, an ensemble which is computationally indistinguishable from the uniform distribution.

3.2 t-DDH: A Relaxed DDH Assumption

We proceed to define the t-DDH assumption. The intuition behind this assumption is that if the Computational Diffie-Hellman Assumption holds in a group G generated by a generator g, then the DH value g^{ab} must have some degree of unpredictability (or "partial hardness") even when g^a and g^b are given. Specifically, we say that the t-DDH Assumption holds in the group G if the Diffie-Hellman output g^{ab} has t bits of computational entropy (here $0 \leq t \leq \log(G)$). Formally:

Definition 5. *We say that the $t(n)$-DDH Assumption holds over a group family $\mathcal{G} = \{G_n\}_n$ if for all n there exists a family of probability distributions $\mathcal{X}_n(g_n^a, g_n^b)$ over G_n (one distribution for each pair g_n^a, g_n^b) such that*

- $\mathsf{min\text{-}ent}(\mathcal{X}_n(g_n^a, g_n^b)) \geq t(n)$
- *The probability ensemble $\mathcal{DH}_n$ (see Section 2) is computationally indistinguishable from the ensemble*

$$\mathcal{R}_n^* = \{(g_n^a, g_n^b, C) \text{ for } a, b \in_R \mathrm{ord}(G_n) \text{ and } C \in_{\mathcal{X}_n(g_n^a, g_n^b)} G_n\}$$

It is important to note that the distributions $\mathcal{X}_n(g^a, g^b)$ in the above definition may be different for each pair of values g^a, g^b. Requiring instead a single distribution $\mathcal{X}$ for all pairs g^a, g^b (as may seem more natural at first glance) results in a significantly stronger, and consequently less useful, assumption.

Consider Example 1 from Section 2: over Z_p^* one can break the DDH by detecting if the quadratic residuosity character of C is consistent with the one induced by g^a, g^b. Yet, Z_p^* can satisfy the t-DDH assumption even for high values of t. For example, if for all a, b for which one of a, b is even we define $\mathcal{X}_n(g^a, g^b)$ to be the set of quadratic residues in Z_p^*, and for all other pairs g^a, g^b we define $\mathcal{X}_n(g^a, g^b)$ to be the set of quadratic non-residues in Z_p^*, then the trivial break of DDH in the above example does not hold against these distributions. More generally, if we consider a prime p of the form $2^u q + 1$ where q is a prime then we can get that (given current knowledge) the t-DDH assumption holds for Z_p^* for $t = |p| - u$, while clearly the DDH assumptions does not hold over this group.

Note that the DDH assumption can also be stated in terms of computational entropy. Indeed the DDH assumption over a group G is equivalent to the t-DDH assumption over G for $t = \log(\mathrm{ord}(G))$.

Sampling $\mathcal{X}_n(g^a, g^b)$. The t-DDH Assumption as stated above makes no requirement of efficient samplability for $\mathcal{X}_n(g^a, g^b)$. It is possible to strengthen the assumption by requiring that $\mathcal{X}_n(g^a, g^b)$ be efficiently samplable. We say that the samplable [resp. semi-samplable] t-DDH Assumption holds over $\mathcal{G}$, if the t-DDH Assumption holds over $\mathcal{G}$ and the underlying distributions $\mathcal{X}_n(g^a, g^b)$ are polynomial-time samplable [resp. polynomial-time samplable when either exponent a or b is known].

As a direct consequence of the Entropy Smoothing Theorem and the definition of t-DDH we have:

Lemma 3. *Let $\mathcal{G} = \{G_n\}_n$ be a group family in which the $t(n)$-DDH Assumption holds, and let $\{\mathcal{H}_n\}_n$ be a family of universal hash functions such that for all $h \in \mathcal{H}_n$, $h : G_n \to \{0,1\}^{t'(n)}$ where $t'(n) = t(n) - \omega(\log n)$. Then the induced distribution of $h(g_n^{ab})$, for $a, b \in_R [1..\mathrm{ord}(G_n)]$ and $h \in_R \mathcal{H}_n$, is computationally indistinguishable from the uniform distribution over $\{0,1\}^{t'(n)}$ even when h, g_n^a and g_n^b are given to the distinguisher.*

Notice that the above lemma requires the hash function h to be chosen at random for each application. This is the case in several practical protocols (such as the case of IKE [RFC2409], mentioned in the Introduction, in which a key to the hash function is chosen by the communicating parties anew with each run of the protocol). However, it is also possible to fix a randomly chosen hash function and apply it repeatedly to different DH values. An example of such an application would be its use in the context of the Cramer-Shoup CCA-secure cryptosystem [CS98] (also discussed in the Introduction) in which the specific hash function h would be chosen at random from the family $\mathcal{H}$ by the owner of the decryption key, and published as part of the public key parameters. In this case, the security of the repeated use of the same hash function h can be proved via a standard simulation argument.

Finally we point out that for groups of prime order, the t-DDH Assumption is equivalent to the full DDH. The proof of this fact can be obtained by a standard random self-reducibility argument.

Lemma 4. *Let G be a group of prime order q. If the t-DDH Assumption holds in G for $t > 0$ then the DDH Assumption holds in G as well.*

This yields an interesting 0-1 law for prime order groups, in which either the DDH Assumption holds, and thus the DH output has $\log(q)$ bits of computational entropy, or we cannot claim that the DH output has *any* bits of computational entropy.

3.3 The Max-Subgroup Theorem

We now proceed to prove our main theorem concerning the t-DDH assumption. The significance of the theorem below is that we can claim that a cyclic group is t-DDH if t is the order of the maximal subgroup of G where the DDH holds.

Theorem 3. *Let G be a cyclic group of order $m = m_1 m_2$ where $(m_1, m_2) = 1$, and G_1 be a sub-group of order m_1 in G. If the DDH Assumption holds over G_1 then the $|m_1|$-DDH Assumption holds in G.*

Proof. An initial intuition behind the correctness of the theorem is that the hardness hidden in G_1 could be "sampled" when applying a hash function to the DH values over G. This however is incorrect: the size of G_1 may be negligible in relation to $|G|$ and as such the probability to sample a triple (g^a, g^b, g^{ab}) from G_1 is negligible too. The actual argument, presented next, uses the observation that the "hardness" present in G_1 can be extended to its cosets in G.

Let g be a generator of G and $g_1 = g^{m_2}$ be a generator of order m_1 of G_1. Given $g^a, g^b \in G$, we define the distribution $\mathcal{X}(g^a, g^b)$ to be the uniform distribution over $\{C = g^c \in G$ such that $c \in Z_m$ and $c \equiv ab \bmod m_2\}$ Thus, it is easy to see that $\mathcal{X}(g^a, g^b)$ has $|m_1|$ bits of min-entropy (since the above set has m_1 elements). Let $\mathcal{R}^*$ denote the probability distribution $\{(g^a, g^b, C) : a, b \in_R Z_m$ and $C \in_{\mathcal{X}(g^a, g^b)} G\}$.

We assume by contradiction that the $|m_1|$-DDH assumption does not hold in G, and thus we have a distinguisher D between the distributions $\mathcal{DH}_G$ and $\mathcal{R}^*$. Using D we build a distinguisher D_1 that distinguishes between the distributions $\mathcal{DH}_{G_1}$ and $\mathcal{R}_{G_1}$.

Given a triple (A_1, B_1, C_1) where $A_1 = g_1^{a_1}, B_1 = g_1^{b_1}$, and C_1 either equals $g_1^{a_1 b_1}$ or $g_1^{c_1}$ for $c_1 \in_R Z_{m_1}$, the distinguisher D_1 does the following:

1. Chooses $i, j \in_R Z_m$
2. Sets $A = A_1 g^i, B = B_1 g^j$ and $C = C_1^{m_2} A_1^j B_1^i g^{ij}$ computed in G
3. Hands D the triple (A, B, C)
4. Outputs the same output bit as D.

Let's examine the distribution of the triple (A, B, C). Consider first A. This value is set to $A = A_1 g^i = g_1^{a_1} g^i = g^{m_2 a_1 + i}$ thus $a = m_2 a_1 + i$. Since $i \in_R Z_m$ then also $a \in_R Z_m$. Similarly for $B = g^b$ we get $b \in_R Z_m$. In the case of C we have $C = C_1^{m_2} A_1^j B_1^i g^{ij} = g^{c_1 m_2^2 + m_2 a_1 j + m_2 b_1 i + ij}$, thus $c = c_1 m_2^2 + m_2 a_1 j + m_2 b_1 i + ij$. In addition, we have that $ab = (m_2 a_1 + i)(m_2 b_1 + j) = m_2^2 a_1 b_1 + m_2 a_1 j + m_2 b_1 i + ij$. Thus

$$c - ab = m_2^2 c_1 + m_2 a_1 j + m_2 b_1 i + ij - (m_2^2 a_1 b_1 + m_2 a_1 j + m_2 b_1 i + ij) = m_2^2 c_1 - m_2^2 a_1 b_1$$

which implies $c = m_2^2(c_1 - a_1b_1) + ab \bmod m$. Therefore, if $c_1 = a_1b_1$ then $c = ab$, while if $c_1 \in_R Z_{m_1}$ then $c_1 - a_1b_1 \in_R Z_{m_1}$, and consequently C is distributed according to the distribution $\mathcal{X}(g^a, g^b)$. In other words, the triple (A, B, C) is distributed according to $\mathcal{DH}_G$ if (A_1, B_1, C_1) came from $\mathcal{DH}_{G_1}$, and it is distributed according to $\mathcal{R}^*$ if (A_1, B_1, C_1) came from $\mathcal{R}_{G_1}$. Therefore, D_1 distinguishes between $\mathcal{DH}_{G_1}$ and $\mathcal{R}_{G_1}$ with the same probability that D distinguishes between $\mathcal{DH}_G$ and $\mathcal{R}^*$. Since we assumed the latter probability to be non-negligible we reached a contradiction with the premise that G_1 is a DDH group. □

Remark on samplability. The distributions $\mathcal{X}(g^a, g^b)$ defined in the above proof are efficiently samplable given m_1, m_2 and at least one of a, b. Indeed given, say, $a, B = g^b$ we can sample $\mathcal{X}(g^a, g^b)$ by choosing $k \in_R Z_{m_1}$ and setting $C = g^{km_2}B^a$. In other words, provided that m_1, m_2 are given, Theorem 3 (and its corollary below) can be strengthened to claim that the *semi-samplable* $|m_1|$-DDH Assumption holds in G. We will use this stronger version of the theorem in Section 5.

From the above theorem and the Characterization Theorem we get:

Corollary 1. *For any cyclic group G, G is $|m|$-DDH where m is the order of the* maximal *DDH subgroup of G.*

4 DDH and t-DDH with Short Exponents

In this section we investigate the use of the DDH and t-DDH assumptions in conjunction with the so called "short-exponent discrete-log" assumption.

The Short-Exponent Discrete-Log Assumption. A common practice for increasing the efficiency of exponentiation in cryptographic applications based on the hardness of computing discrete logarithms, and in particular those using the Diffie-Hellman transform, is to replace full-length exponents (i.e. of length logarithmic in the group order) with (significantly) shorter exponents. The security of this practice cannot be justified by the usual assumption that computing discrete logarithms (with full-length exponents) is hard, but rather requires a specific assumption first analyzed in [vOW96] and formalized (as follows) in [PS98].

Assumption 4 (s-DLSE [PS98]) *Let $\mathcal{G} = \{G_n\}_n$ be a family of cyclic groups where each G_n has a generator g_n and $ord(G_n) = m(n) > 2^n$. We say that the s-DLSE Assumption holds in G if for every probabilistic polynomial time Turing machine I, for every polynomial $P(\cdot)$ and for all sufficiently large n we have that $Prob_{x \in_R [1..2^s]}(I(g_n, m(n), s, g_n^x) = x) \leq 1/P(n)$.*

Current knowledge points to the plausibility of the above assumption even for exponents s significantly shorter than $\log(ord(g))$. The exact values of s for which the assumption seems to hold depends on the group generated by the element g. An obvious lower bound on s, if one wants to achieve security against 2^n-complexity attacks, is $s \geq 2n$ which is necessary to thwart the usual square-root attacks such as Shanks and Pollard methods. However, as it was pointed out in [vOW96], there are cases where s needs to be chosen larger than $2n$. Specifically, they show how to use a Pohlig-Hellman decomposition to obtain some of the bits of the exponent. The power of the attack depends on the (relatively) small prime factors of the group order. For example, when working over Z_p^* with a random prime p, the [vOW96] results indicate the use of $s \approx 4n$ (e.g., with a security parameter of 80 one should use $s = 320$ which is much shorter than the 1024 or 2048 bits of p, yet twice as much as the bare minimum of $s = 160$). If one wants to use $s = 2n$ (i.e. assume the $2n$-DLSE), it is necessary to work in special groups such as those of prime order or Z_p^* with p a safe prime (i.e. $p = 2q + 1$, and q prime).

From Hardness to Indistinguishability. Gennaro [Gen00] proves that if the s-DLSE assumption holds in $G = Z_p^*$ with p a safe prime then the distribution over G generated by g^x for $x \in_R [1..2^s]$ is computationally indistinguishable from the uniform distribution over G. Here we use a generalization of this result that we summarize in the following proposition (see the full version of this paper [GKR04] for a proof of this Proposition).

Proposition 1. *Let G be a cyclic group of order m generated by g, such that m is odd or $m/2$ is odd. If the s-DLSE Assumption holds in G, then the following two distributions $\mathcal{S}_G = \{g^x \ : \ x \in_R [1..2^s]\}$ and $\mathcal{U}_G = \{g^x \ : \ x \in_R Z_m\}$ are computationally indistinguishable.*

Next we show that if in a group G, both the s-DLSE and the t-DDH Assumptions hold, then performing the Diffie-Hellman transform with short exponents a and b, yields a DH output with t bits of computational entropy. In other words, the security of the hashed DH transform over such groups when using s-bit long exponents is essentially equivalent to that of using full exponents.

Theorem 5. *Let G be a cyclic group of order m generated by g, such that m is odd, or $m/2$ is odd. Let s, t be such that the s-DLSE and the semi-samplable t-DDH Assumptions hold in G. Denote with $\mathcal{X}(g^a, g^b)$ the family of distributions induced by the t-DDH assumption over G (see Def. 5). Then the following two distributions*

$$\mathcal{SDH} = \{(g^a, g^b, g^{ab}) \ \textit{for} \ a, b \in_R [1..2^s]\}$$

and

$$\mathcal{SR}^* = \{(g^a, g^b, C) \ \textit{for} \ a, b \in_R [1..2^s] \ \textit{and} \ C \in_{\mathcal{X}(g^a, g^b)} G\}$$

are computationally indistinguishable.

Proof. Recall that if the t-DDH Assumption holds over the group G of order m, then there exists a family of probability distributions $\mathcal{X}(g^a, g^b)$ with min-entropy t (one distribution for each pair g^a, g^b) over G such that the distributions

$$\mathcal{DH} = \{(g^a, g^b, g^{ab}) \text{ for } a, b \in_R Z_m\}$$

and

$$\mathcal{R}^* = \{(g^a, g^b, C) \text{ for } a, b \in_R Z_m \text{ and } C \in_{\mathcal{X}(g^a, g^b)} G\}$$

are computationally indistinguishable.

The following standard hybrid argument yields the proof of the Theorem. Consider the intermediate distributions

$$\mathcal{D}_0 = \{(g^a, g^b, g^{ab}) \text{ for } a, b \in_R [1..2^s]\}$$

$$\mathcal{D}_1 = \{(g^\alpha, g^b, g^{\alpha b}) \text{ for } \alpha \in_R Z_m, b \in_R [1..2^s]\}$$

$$\mathcal{D}_2 = \{(g^\alpha, g^\beta, g^{\alpha\beta}) \text{ for } \alpha, \beta \in_R Z_m\}$$

$$\mathcal{D}_3 = \{(g^\alpha, g^\beta, C) \text{ for } \alpha, \beta, \in_R Z_m \text{ and } C \in_{\mathcal{X}(g^\alpha, g^\beta)} G\}$$

$$\mathcal{D}_4 = \{(g^\alpha, g^b, C) \ b \in_R [1..2^s], \alpha \in_R Z_m \text{ and } C \in_{\mathcal{X}(g^\alpha, g^b)} G\}$$

$$\mathcal{D}_5 = \{(g^a, g^b, C) \ : \ a, b \in_R [1..2^s] \text{ and } C \in_{\mathcal{X}(g^a, g^b)} G\}$$

Clearly $\mathcal{D}_0 = \mathcal{SDH}$ while $\mathcal{D}_5 = \mathcal{SR}^*$. If there is an efficient distinguisher between these distributions then, by a standard hybrid argument, there is an efficient distinguisher between $\mathcal{D}_i$ and $\mathcal{D}_{i+1}$ for some $i \in \{0, 1, 2, 3, 4\}$. But under the t-DDH Assumption we know that $\mathcal{D}_2$ is computationally indistinguishable from $\mathcal{D}_3$. Also, under the s-DLSE Assumption we know that $\mathcal{D}_i$ is computationally indistinguishable from $\mathcal{D}_{i+1}$ for $i = 0, 1, 3, 4$ by reduction to Proposition 1 (in the case $i = 3, 4$ one needs $\mathcal{X}(g^a, g^b)$ to be semi-samplable). □

Note that, as a particular case, when $t = \log(m)$ the theorem states that if G is a DDH group in which the s-DLSE assumption holds, then performing the DH transform over G with exponents of size s yields values that are indistinguishable from random elements in G.

5 Hashed DH over Z_p^* and its Subgroups

Here we discuss the security of the hashed DH transform over groups and subgroups of Z_p^* for prime p. Throughout this section we assume that the DDH assumption holds over the large prime-order subgroups of Z_p^*. Under this assumption we immediately get that it is secure to use the hashed DH transform over a subgroup G_q of Z_p^* of order q, provided that q is a sufficiently large prime that divides $p-1$. By sufficiently large we mean that the DDH assumption (plausibly) holds in G_q (for a given security parameter k), and that the computational entropy of q is sufficient for the application. Specifically, if the application requires a pseudorandom output of ℓ bits then q needs to satisfy $|q| \geq \ell + 2k$. Similarly, we get that it is secure to work in any subgroup of Z_p^* whose order m is the product of large primes (each of which divides $p - 1$); also here it is required that $|m| \geq \ell + 2k$, although note that each of the prime factors of m may be smaller than that bound (one usually assumes the DDH to hold on groups of prime order q with $|q| \geq 2k$).

Moreover, one of the most significant contributions of our work is in showing the security of the hashed DH transform also over groups (or subgroups) whose order is divisible by small prime factors (and therefore not satisfying the DDH assumption). In particular, this is necessarily the case for the group Z_p^* with prime p (the order $m = p - 1$ of this group is always divisible by small prime factors, e.g., 2). Our results show that the hashed DH is secure over Z_p^* provided that $p - 1$ has enough prime divisors whose product is larger than the entropy bound $2^{\ell+2k}$, and for which the subgroups of corresponding prime order are DDH. (In particular, the fact that $p - 1$ has additional smaller prime factors does not invalidate the security of the hashed DDH in Z_p^*.)

A particularly interesting group is Z_p^* for $p = 2q + 1$ and q prime. In this case, working directly with the hashed DH over Z_p^* is secure since we are assuming that its subgroup of order q is DDH, and therefore the whole Z_p^* group is $|\frac{p-1}{2}|$-DDH. Working over Z_p^* in this case has several important advantages: (i) one can produce a large (actually, largest) number of pseudorandom bits (specifically, $|p| - 1 - 2k$ bits); (ii) p can be chosen such that 2 is a generator of Z_p^* (which speeds up exponentiation); (iii) the $2k$-DLSE Assumption (see Section 4) is conjectured to hold in these groups [vOW96] and therefore one can use minimal-length exponents (i.e., of length $2k$) in these groups, obtaining yet another significant exponentiation speedup without sacrificing the security of the (hashed) DH transform; and (iv) these groups are free from the potentially serious attacks described in [LL97] (that affect subgroups of prime order q where $(p-1)/q$ has a relatively large smooth factor). Note that items (i) and (iii) follow essentially from our results. The only drawback working over such a group is the cost of generating p's of the above form; this, however is insignificant in typical applications (e.g., IKE [RFC2409]) in which this generation is very rare, and usually done at the set-up of the system and used for a large period of time.

Note that in all of the above examples it is assumed that one knows the full or partial factorization of $p-1$; in particular, the knowledge of this factorization is essential for selecting a generator of the group. It is a theoretically and practically important question to establish whether the knowledge of the factorization of $p-1$ is essential for working securely over Z_p^* or over one of its subgroups. In the rest of this section we show that this knowledge is not essential. Specifically, it follows from our results that if one chooses a random prime p (of a pre-specified size such that the Discreet Logarithm Problem is hard in Z_p^*) and a random element e in Z_p^*, then performing the hashed DH transform over the group generated by e is secure.[4]

Let p be a random prime such that $p - 1 = p_1 p_2 ... p_n$ and $p_1 \leq p_2 \leq ... \leq p_n$ are all (not necessarily different and possibly unknown) primes. Let e be an element randomly chosen from Z_p^*, and let G_e denote the subgroup of Z_p^* generated by

[4] We stress that while the legitimate users of such a scheme do not need to know the factorization of $p - 1$, the scheme remains secure even if this factorization is known to the attacker.

e. We first claim that with overwhelming probability the large prime factors of $p-1$ divide the order of G_e.

Lemma 5. *Let Z_p^* and $p-1 = p_1..p_n$ be as described above. Then for all $1 \leq i \leq n$: $Pr_{e \in_R Z_p^*}[p_i \not| \; ord(e)] \leq 1/p_i$.*

Proof. Let g be a generator of Z_p^*. There are at most $(p-1)/p_i$ elements whose order is not divisible by p_i, and they are the elements of the form g^{jp_i} for $1 \leq j \leq (p-1)/p_i$. When $p_i^2|p-1$ this is a strict upper bound, otherwise this is an exact bound. Thus, the probability to choose e such that $p_i \not| \; ord(e)$ is at most $\frac{(p-1)/p_i}{p-1} = \frac{1}{p_i}$. □

Corollary 2. *For a given bound B, let $p-1 = \Pi_{i=1}^n p_i$ where $p_j, p_{j+1}, ..., p_n > B$. Then*

$$Pr_{e \in_R Z_p^*}[\Pi_{i=j}^n \; p_i \mid ord(e)] \geq 1 - \sum_{i=j}^{n} \frac{1}{p_i} \geq 1 - \frac{n-j}{B} \geq 1 - \frac{\log p}{B}.$$

Thus, for large values of B, the order of a random element e is divisible, with overwhelming probability, by all the prime factors of $p-1$ which are larger than B. Or, equivalently, G_e has as subgroups all the prime-order subgroups of Z_p^* whose order is larger than B.

Now, if we set our security parameter to k, define $B = 2^{2k}$, and assume that the DDH holds in subgroups of prime order larger than B, then we have that, with overwhelming probability, G_e contains all the prime order DDH subgroups of Z_p^*. In other words, if we denote by P the product of all prime factors of $p-1$ larger than B, we have that G_e contains, by virtue of our DDH Characterization Theorem (Theorem 1) a DDH subgroup of size P, and then by the Max-Subgroup Theorem (Theorem 3) we get that G_e is $|P|$-DDH.

All it is left to argue is that $|P|$ is large enough. For this we use the following Lemma from [vOW96] that provides an upper bound on the expected size of the product of all prime divisors of $p-1$ that are smaller than B (and thus, it provides a lower bound on the expected size of $|P|$).

Lemma 6 ([vOW96]). *For a random prime p (as above) and a fixed bound B, the expected length of $\Pi_i p_i$ where $p_i < B$ is $\log B + 1$.*

In other words, the lemma states that the expected size of $|P|$ is $|p| - |B| = |p| - 2k$.

If, for the sake of illustration, we set $|p| = 1024$ and $k = 80$ we get that we expect G_e to be 864-DDH. However, note that this expected size may vary for specific p's. Yet, note that even if p happens to have a B-smooth part that is 4 times larger than expected (!) we are still left with a 384-DDH subgroup G_e with enough computational entropy for most DH applications (such as deriving a 128-bit pseudorandom key). If one considers 2048-bits and $k = 160$ then the expected amount of entropy is 2048-320=1728 bits which, again, leaves plenty room to compensate for "unlucky choices" of p.

Notice that in order to use short exponents in this case (i.e. random prime p and random generator e), one must make sure that the order m of the group generated by e is either odd, or $m/2$ is odd (so that we can invoke Theorem 5). This can be easily achieved by choosing first a random element e in Z_p^* and then using as the group generator the element $e^{2^f} \bmod p$ where f is the maximal integer such that $2^f|(p-1)$ (the value f is, of course, trivial to obtain without requiring of any significant factorization of $p-1$).

Remark (semi-samplability). In the above discussion we have justified the usage of short exponents on the basis of Theorem 5. Note, however, that this theorem assumes the semi-samplability of the distributions $\mathcal{X}(g^a, g^b)$. Therefore, we need to verify that this semi-samplability property holds for the above applications. This is indeed the case since these applications use the distributions defined in the proof of Theorem 3, which are semi-samplable when the factorization of the group order is known (see the remark following the proof of Theorem 3). Therefore, we obtain that, even though the honest parties can perform the hashed DH transform securely with short exponents, and *without* requiring the knowledge of the factorization of $p-1$, the DH transform remains secure *even if* such factorization is available to the attacker.

Acknowledgment

We thank the anonymous referees for their useful comments.

References

[ABR01] M. Abdalla, M. Bellare, and P. Rogaway. DHIES: An Encryption Scheme Based on the Diffie-hellman Problem. In *CT-RSA '01*, pages 143–158, 2001. LNCS No. 2020.

[BJN00] D. Boneh, A. Joux, and P. Nguyen. Why Textbook ElGamal and RSA Encryption are Insecure . In *AsiaCrypt '00*, pages 30–44, 2000. LNCS No. 1976.

[Bon98] D. Boneh. The Decision Diffie-Hellman Problem. In *Third Algorithmic Number Theory Symposium*, pages 48–63, 1998. LNCS No. 1423.

[Bra93] S. Brands. An Efficient Off-Line Electronic Cash System Based on the Representation Problem. TR CS-R9323, CWI, Holland, 1993.

[CW79] L. Carter and M. N. Wegman. Universal Classes of Hash Functions. *JCSS*, 18(2):143–154, April 1979.

[CS98] R. Cramer and V. Shoup. A Practical Public Key Cryptosystem Provable Secure Against Adaptive Chosen Ciphertext Attack. In *Crypto '98*, pages 13–25, 1998. LNCS No. 1462.

[DH76] W. Diffie and M. E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[ElG85] T. ElGamal. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. *IEEE Trans. Info. Theory*, IT 31:469–472, 1985.

[Gen00] R. Gennaro. An Improved Pseudo Random Generator Based on Discrete Log. In *Crypto '00*, pages 469–481, 2000. LNCS No. 1880.

[GHKR04] R. Gennaro, J. Hastad, H. Krawczyk and T. Rabin. Randomness Extraction and Key Derivation Using the CBC, Cascade and HMAC modes. Manuscript, 2004.

[GKR04] R. Gennaro, H. Krawczyk and T. Rabin. Secure Hashed Diffie-Hellman over Non-DDH Groups. Full version available at `http://eprint.iacr.org/2004/`

[GM84] S. Goldwasser and S. Micali. Probabilistic Encryption. *JCSS*, 28(2):270–299, April 1984.

[HILL99] J. Hastad, R. Impagliazzo, L. Levin, and M. Luby. Construction of a Pseudo-random Generator from any One-way Function. *SIAM. J. Computing*, 28(4):1364–1396, 1999.

[Kra03] H. Krawczyk. SIGMA: The 'SiGn-and-MAc' Approach to Authenticated Diffie-Hellman and Its Use in the IKE Protocols. In *Crypto '03*, pages 400–425, 2003. LNCS No. 2729. `http://www.ee.technion.ac.il/~hugo/sigma.html`

[LL97] C.H. Lim and P.J. Lee. A Key Recovery Attack on Discrete Log-Based Schemes Using a Prime Order Subgroup. In *Crypto '97*, pages 249–263, 1997. LNCS No. 1294.

[Lub96] M. Luby. *Pseudorandomness and Cryptographic Applications.* Princeton Computer Science Note, Princeton University Press, January 1996.

[NR97] M. Naor and O. Reingold. Number-theoretic Constructions of Efficient Pseudorandom Functions. In *Proc. 38th FOCS*, pages 458–467. IEEE, 1997.

[PS98] S. Patel and G. Sundaram. An Efficient Discrete Log Pseudo Random Generator. In *Crypto '98*, pages 304–317, 1998. LNCS No. 1462.

[RFC2409] RFC2409. The Internet Key Exchange (IKE). Authors: D. Harkins and D. Carrel, Nov 1998.

[Sta96] M. Stadler. Publicly Verifiable Secret sharing. In *Eurocrypt '96*, pages 190–199, 1996. LNCS No. 1070.

[vOW96] P.C. van Oorschot and M. Wiener. On Diffie-Helman Key Agreement with Short Exponents. In *Eurocrypt '96*, pages 332–343, 1996. LNCS No. 1070.

A Indistinguishability of Probability Distributions

Definition 6. *Let $\mathcal{X}_n, \mathcal{Y}_n$ be two probability distributions over a support set A_n. We say that $\mathcal{X}_n$ and $\mathcal{Y}_n$ have* statistical distance *bounded by $\Delta(n)$ if $\sum_{x \in A_n} |Prob_{\mathcal{X}_n}[x] - Prob_{\mathcal{Y}_n}[x]| \leq \Delta(n)$. We say that the ensembles $\mathcal{X}_n$ and $\mathcal{Y}_n$ are* statistically indistinguishable *if for every polynomial $P(\cdot)$ and for all sufficiently large n we have that $\Delta(n) \leq \frac{1}{P(n)}$.*

Definition 7. *Let $\mathcal{X}_n, \mathcal{Y}_n$ be two probability ensembles. Given a family of circuits $D = \{D_n\}_n$ (called the distinguisher) consider the following quantities*

$$\delta_{D,\mathcal{X}_n} = Prob_{x \in \mathcal{X}_n}[D_n(x) = 1] \quad \textit{and} \quad \delta_{D,\mathcal{Y}_n} = Prob_{y \in \mathcal{Y}_n}[D_n(y) = 1]$$

We say that the probability ensembles $\mathcal{X}_n$ and $\mathcal{Y}_n$ are computationally indistinguishable *(by non-uniform distinguishers) if for every polynomial-size distinguisher family D, for every polynomial $P(\cdot)$, and for all sufficiently large n we have that $|\delta_{D,\mathcal{X}_n} - \delta_{D,\mathcal{Y}_n}| \leq \frac{1}{P(n)}$*