6.889: New Developments in Cryptography April 5, 2011

Bounded Memory Leakage

Instructor: Yael Tauman Kalai *Scribe: Raluca Ada Popa*

When designing cryptographic schemes, we usually rely on the assumption that *every bit of the secret key is secret*. However, in practice, loss of secrecy can happen due to side-channel attacks. For example, an adversary can get secret information using timing attacks, acoustic attacks, or even by getting access to parts of the memory used by a cryptographic protocol such as in the "cold-boot attack" demonstrated by Halderman et al. [HSH+09]. With some bits of the secret key revealed, security guarantees may no longer hold.

1 Preliminaries

1.1 Notation

For a distribution X, we use $x \overset{R}{\leftarrow} X$ to denote that x is a sample drawn from the distribution X. For a set S, we use $x \overset{R}{\leftarrow} S$ to denote that x is drawn uniformly at random from the set S.

We use $H_\infty(X)$ to denote the min-entropy of a random variable X defined as $H_\infty(X) = \min_{u \in U} \big\{ - \log \Pr[X = u] \big\}$, where U is the set of all values X may take. We use U_d to denote the uniform distribution over $\{0,1\}^d$.

The notation $\bar{s}$ indicates that s is a vector.

If D_1 and D_2 are distributions, the notation $D_1 \approx_\epsilon D_2$ indicates statistical indistinguishability with an advantage of at most ϵ.

1.2 Leftover hash lemma

We recall the leftover hash lemma introduced in previous lectures in a form useful to some of the constructions in these notes.

Theorem 1 (Leftover Hash Lemma)**.** *Fix $\epsilon > 0$. Let X be a random variable on $\{0,1\}^n$ with min-entropy k. Let $\mathcal{H} = \{\mathcal{H}_n\}_{n \in \mathbb{N}}$ where $\mathcal{H}_n = \{h_s\}_{s \in \{0,1\}^d}$ for all n, be a universal hash family with output length $m \leq k - 2\log(1/\epsilon)$. Then,*

$$\{(h(x), h) : x \overset{R}{\leftarrow} X, h \overset{R}{\leftarrow} \mathcal{H}_n\} \approx_\epsilon \{(u, h) : u \overset{R}{\leftarrow} U_m, h \overset{R}{\leftarrow} \mathcal{H}_n\}.$$

2 Semantic Security with λ-bit leakage

We first recall the definition of semantic security and then enhance it with λ-bit leakage resilience.

Definition 2 (**Semantic security**)**.** *A public-key encryption scheme $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is semantically secure if for all PPT $\mathcal{A}$, for any polynomial p, for any sufficiently large $n \in \mathbb{N}$,*

$$|\Pr[\mathsf{Expt}_0(\mathcal{E}, \mathcal{A}, n) = 1] - \Pr[\mathsf{Expt}_1(\mathcal{E}, \mathcal{A}, n) = 1]| < 1/p(n),$$

where $\mathsf{Expt}_b(\mathcal{E}, \mathcal{A}, n)$ is defined as follows. $\mathsf{Expt}_b(\mathcal{E}, \mathcal{A}, n)$:

1. *The challenger generates* $(\mathsf{PK}, \mathsf{SK}) \leftarrow \mathsf{Gen}(1^n)$ *and sends* PK *to* $\mathcal{A}$*.*

2. *The adversary $\mathcal{A}$ replies with (m_0, m_1).*

3. *The challenger computes $y \leftarrow \mathsf{Enc}(\mathsf{PK}, m_b)$, and sends y to $\mathcal{A}$.*

4. *$\mathcal{A}$ outputs b'.*

Let λ be a nonnegative integer indicating the amount of allowed leakage.

Definition 3 (**Semantic Security with λ–bit Leakage**). *A public-key encryption scheme $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is semantically secure with λ-bit leakage if, for all PPT $\mathcal{A}$, for any polynomial p, for all sufficiently large $n \in \mathbb{N}$, we have*

$$|\Pr[\mathsf{Expt}_0^{\lambda}(\mathcal{E}, \mathcal{A}, n) = 1] - \Pr[\mathsf{Expt}_1^{\lambda}(\mathcal{E}, \mathcal{A}, n) = 1]| < 1/p(n),$$

where $\mathsf{Expt}_b^{\lambda}(\mathcal{E}, \mathcal{A}, n)$ is the output of the following game between $\mathcal{A}$ and a challenger:

1. *The adversary $\mathcal{A}$ selects a leakage function $L : \{0,1\}^* \to \{0,1\}^{\lambda}$ and sends it to the challenger.*

2. *Challenger generates $(\mathsf{SK}, \mathsf{PK}) \leftarrow \mathsf{Gen}(1^n)$ and sends $(\mathsf{PK}, L(\mathsf{SK}, \mathsf{PK}))$ to $\mathcal{A}$.*

3. *$\mathcal{A}$ chooses two messages m_0 and m_1 such that $|m_0| = |m_1|$ and sends (m_0, m_1) to the challenger.*

4. *Challenger sends $C \leftarrow \mathsf{Enc}(\mathsf{PK}, m_b)$ to $\mathcal{A}$.*

5. *$\mathcal{A}$ outputs b'.*

Note that this definition is similar to the definition of semantic security with the addition of the leakage function L. Even though the adversary chooses L before receiving PK, the definition is still adaptive with respect to PK: the leakage function L can first look at the value of PK and choose the actual leakage based on the value of PK. The adversary $\mathcal{A}$ is a PPT algorithm so any function that it would run to compute a leakage function based on PK must be polynomial-time as well, and thus can be incorporated in L.

2.1 Decisional Diffie-Hellman Assumption

We recall the DDH assumption as well as present a more general form proven equivalent by Naor and Reingold [NR04].

Let $\mathsf{GroupGen}$ be a probabilistic polynomial-time algorithm that takes as input a security parameter 1^n for some positive integer n, and outputs $(\mathbb{G}, q, g)$, where q is an n-bit prime number, $\mathbb{G}$ is a group of order q, and g is a generator of $\mathbb{G}$.

Decisional Diffie-Hellman Assumption (DDH). The DDH assumption is that the ensembles $\left\{(\mathbb{G}, g_1, g_2, g_1^r, g_2^r)\right\}_{n\in\mathbb{N}}$ and $\left\{(\mathbb{G}, g_1, g_2, g_1^{r_1}, g_2^{r_2})\right\}_{n\in\mathbb{N}}$ are computationally indistinguishable, where $(\mathbb{G}, q, g) \leftarrow \mathsf{GroupGen}(1^n)$, and the elements $g_1, g_2 \in \mathbb{G}$ and $r, r_1, r_2 \in \mathbb{Z}_q$ are chosen independently and uniformly at random.

Naor and Reingold [NR04] showed that, if DDH holds, so does the following generalization of DDH considering $\ell > 2$ generators.

Lemma 4 ([NR04]). *Under the DDH assumption, for any positive integer ℓ, the ensembles*

$$\left\{(g_1, \ldots, g_\ell, g_1^r, \ldots, g_\ell^r) : g_i \overset{R}{\leftarrow} \mathbb{G}, r \overset{R}{\leftarrow} \mathbb{Z}_q\right\} \text{ and } \left\{(g_1, \ldots, g_\ell, g_1^{r_1}, \ldots, g_\ell^{r_\ell}) : g_i \overset{R}{\leftarrow} \mathbb{G}, r_i \overset{R}{\leftarrow} \mathbb{Z}_q\right\}$$

are computationally indistinguishable, where $(\mathbb{G}, q, g) \leftarrow \mathsf{GroupGen}(1^n)$.

2.2 Difficulties with straightforward reductions

Before showing a construction of a leakage resilient encryption scheme, we give some intuition showing why a typical security reduction is unlikely to yield a proof of security.

Suppose we would like to prove that, under the DDH assumption, $\mathcal{E}$ is secure with λ-bits of leakage. The proof would typically proceed by contradiction: suppose there exists a PPT $\mathcal{A}$ that breaks $\mathcal{E}$, and we construct a PPT $\mathcal{B}$ that breaks the DDH assumption. The adversary $\mathcal{A}$ behaves as in Definition 3 so it will first provide $\mathcal{B}$ with the leakage function L. The reduction $\mathcal{B}$ is supposed to return PK and $L(\mathsf{SK}, \mathsf{PK})$. In order to use L as a black box, $\mathcal{B}$ needs to generate SK and PK because L may be checking that SK is correct. If L checks SK and finds that it is not a correct secret key, it may only output certain values which $\mathcal{A}$ will recognize and for which it will not break $\mathcal{E}$. Now that $\mathcal{B}$ had to generate SK and PK himself, it seems that $\mathcal{A}$ can no longer help $\mathcal{B}$ because it will not tell $\mathcal{B}$ anything about the secret key that $\mathcal{B}$ could not compute himself from SK and PK. Thus, it seems that $\mathcal{B}$ cannot exploit the power of $\mathcal{A}$.

However, the above intuition is incorrect. The insight in these proofs is for $\mathcal{B}$ to generate an improper encryption C' instead of $C = \mathsf{Enc}(\mathsf{PK}, m_b)$; $\mathcal{B}$ computes C' using some information from the DDH instance it is trying to solve and hopefully leverage $\mathcal{A}$'s power to learn some new information.

2.3 Construction: BHHO

Naor and Segev [NS09] show that the scheme of Boneh et al. [BHHO08] can be made secure against bounded leakage. Boneh et al. have proposed this scheme as a circular-secure encryption scheme and it can be thought of as an extension of the El-Gamal scheme.

A slightly modified version of the BHHO encryption scheme is defined as follows:

- $\mathsf{Gen}(1^n)$: Choose $s \in \mathbb{Z}_q^\ell$, where ℓ is some polynomial in n depending on the desired resilience to leakage, and $g_1, \ldots, g_\ell \overset{R}{\leftarrow} \mathbb{G}$. Let $y = \prod_{i=1}^{\ell} g_i^{s_i}$ and output keys $\mathsf{SK} = s$ and $\mathsf{PK} = (g_1, \ldots g_\ell, y)$.
- $\mathsf{Enc}(\mathsf{PK}, m)$ for $m \in \mathbb{G}$, performs: choose $r \overset{R}{\leftarrow} \mathbb{Z}_q$ and output $(c_1, \ldots, c_{l+1}) = (g_1^r, \ldots, g_\ell^r, y^r \cdot m)$ as the encryption of m.
- $\mathsf{Dec}(\mathsf{SK}, c_1, \ldots, c_\ell, c_{l+1}) = \dfrac{c_{l+1}}{(\prod_{i=1}^{\ell} c_i^{s_i})}$.

The completeness property of the encryption scheme is immediate:

$$\mathsf{Dec}(\mathsf{SK}, c_1, \ldots, c_\ell, c_{l+1}) = \frac{c_{l+1}}{\prod_{i=1}^{\ell} c_i^{s_i}} = \frac{\left(\prod_{i=1}^{\ell} g_i^{rs_i}\right) \cdot m}{\prod_{i=1}^{\ell} g_i^{rs_i}} = m$$

Claim 5. *Under the DDH assumption, BHHO is a semantically secure encryption scheme with $\lambda = |\mathsf{SK}|(1 - o(1))$ bits of leakage, where $|\mathsf{SK}|$ is the length of the secret key.*

Proof. We proceed by contradiction: assume there is a polynomial $p(n)$ and a PPT $\mathcal{A}$ that breaks the BHHO scheme above with at least $1/p(n)$ advantage and construct a PPT $\mathcal{B}$ that breaks DDH. The algorithm $\mathcal{B}$ is defined as follows:

Algorithm 1 ($\mathcal{B}$ on input $(g_1, \ldots, g_\ell, c_1, \ldots, c_\ell)$ for security parameter n)**.** Recall that $\mathcal{B}$ has to decide if the input is of the form $(g_1, \ldots, g_\ell, g_1^r, \ldots, g_\ell^r)$ or $(g_1, \ldots, g_\ell, g_1^{r_1}, \ldots, g_\ell^{r_\ell})$. The reduction $\mathcal{B}$ emulates the security game for $\mathcal{A}$ as follows:

1. Receive L from $\mathcal{A}$.
2. Run $\mathsf{Gen}(1^n)$ to obtain $(\mathsf{SK}, \mathsf{PK})$ and send PK and $L(\mathsf{SK}, \mathsf{PK})$ to $\mathcal{A}$.
3. $\mathcal{B}$ receives m_0 and m_1 from $\mathcal{A}$.
4. $\mathcal{B}$ flips a coin b and computes $C' = \mathsf{Enc}^*(\mathsf{SK}, m_b) = (c_1, \ldots, c_\ell, \prod_{i=1}^{\ell} c_i^{s_i} \cdot m_b)$
5. $\mathcal{A}$ replies with its guess b'.
6. If $\mathcal{A}$ guesses right (that is, $b = b'$), $\mathcal{B}$ outputs 0 (meaning the input is of the first form), else $\mathcal{B}$ outputs 1 (meaning the input is of the second form).

Let us compute the probability of $\mathcal{B}$ distinguishing correctly. There are two cases for the inputs to $\mathcal{B}$, each equally likely.

Case 1: $(c_1 = g_1^r, \ldots, c_\ell = g_\ell^r)$. We can see that, in this case, $\mathcal{A}$ receives the right distribution of inputs it expects and therefore, it can distinguish with probability at least $1/2 + 1/p(n)$; thus, $\mathcal{B}$ will also make the correct decision with probability at least $1/2 + 1/p(n)$.

Case 2: $(c_1 = g_1^{r_1}, \ldots, c_\ell = g_\ell^{r_\ell})$. In this case, we would like to make sure that $\mathcal{A}$ does not guess b' correctly too often because this would cause $\mathcal{B}$ to output the wrong answer. The approach is to show that C' hides b information-theoretically even with leakage, and thus A will guess the correct answer with probability at most 1/2 plus a negligible amount. In this case, the view of $\mathcal{A}$ can be summarized by $(\mathsf{PK}, L(\mathsf{PK}, \mathsf{SK}), g_1^{r_1}, \ldots, g_\ell^{r_\ell}, \prod_{i=1}^{\ell} g_i^{r_i s_i} \cdot m_b)$ and we want to argue that it is statistically indistinguishable from $(\mathsf{PK}, L(\mathsf{PK}, \mathsf{SK}), g_1^{r_1}, \ldots, g_\ell^{r_\ell}, U)$ with some small error.

Note that distinguishing between these two ensembles is at least as hard as distinguishing between a second pair of ensembles, $(\mathsf{PK},\ L(\mathsf{SK}, \mathsf{PK}),\ r_1,\ \ldots,\ r_\ell,\ \langle \bar{r}, \bar{s} \rangle)$ and $(\mathsf{PK},\ L(\mathsf{SK}, \mathsf{PK}),\ r_1,\ \ldots,\ r_\ell,\ U)$, as follows. For some generator g of $\mathbb{G}$, consider replacing each $g_i = g^{\delta_i}$ for some δ_i in the first pair of ensembles and rewriting the ensembles. The reduction now becomes straightforward: it consists of simply raising g to the power of r_i and $\langle \bar{r}, \bar{s} \rangle$.

To prove that $(\mathsf{PK},\ L(\mathsf{SK}, \mathsf{PK}),\ r_1,\ \ldots,\ r_\ell,\ \langle \bar{r}, \bar{s} \rangle)$ and $(\mathsf{PK},\ L(\mathsf{SK}, \mathsf{PK}),\ r_1,\ \ldots,\ r_\ell,\ U)$ are statistically close, we apply the leftover hash lemma, Theorem 1. Consider the collection of hash functions $\mathcal{H}_n$ consisting of $h_{\bar{r}}(\bar{s}) = \langle \bar{r}, \bar{s} \rangle_q$. We can see that this is a universal hash family. To apply the leftover hash lemma we need $H_\infty(\bar{s}) \geq \log q + 2\log(1/\epsilon)$. We have that $H_\infty(\bar{s}) = \ell \log q - \lambda - \log q$ because $\bar{s}$ (the secret key) is $\ell \log q$ bits long, λ of them leak due to L, and $\log q$ of them leak due to $\langle \bar{r}, \bar{s} \rangle$. Therefore, as long as $\lambda \leq \ell \log q - 2\log q - 2\log 1/\epsilon = |\mathsf{SK}|(1 - o(1))$, by Theorem 1, the distributions in question have statistical difference at most $\epsilon/2$.

Putting together these two cases:

$$\begin{aligned}\Pr[\mathcal{B} \text{ wins}] &= 1/2 \Pr[\mathcal{B} \text{ wins} \mid \text{Case } 1] \ + 1/2 \Pr[\mathcal{B} \text{ wins} \mid \text{Case } 2] \\ &\geq 1/2(1/2 + 1/p(n)) + 1/2(1/2 - \epsilon/2) = 1/2 + 1/2(1/p(n) - \epsilon/2).\end{aligned}$$

We can choose ϵ to be $n^{-\log n}$; in this case, $\mathcal{B}$'s advantage remains nonnegligible at breaking the DDH assumption (by contradicting Lemma 4) and the leakage tolerated becomes $\ell \log q - 2\log q - 2\log^2 n = |\mathsf{SK}|(1 - o(1))$.

□

In the construction of this proof, we allowed the leakage λ to be $|\mathsf{SK}|(1 - o(1))$. If $n \approx \log q$ is the size of the security parameter, and the size of the secret key ℓ is a polynomial in this size, say n^c, we allow leakage of at most $n^{c+1} - 2n$; this amount of leakage is significant and can be made larger than any fraction of the secret key's length.

The following question now arises naturally: what if we allow a larger leakage to happen as long as it is still computationally infeasible for an adversary to find SK? We may give away SK information-theoretically, but a polynomially-bounded adversary may still not be able to compute

SK. In the proof above, we provided information-theoretic guarantees, but we may be able to provide computational guarantees with such a setting. We explore this direction next.

3 Semantic security with auxiliary input

Semantic security with respect to auxiliary inputs was first defined by Dodis et al. [DKL09].

Definition 6 (**Semantic security with $2^{-\lambda}$-hard-to-invert auxiliary input**). *A public-key encryption scheme $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with message space $\mathcal{M} = \{\mathcal{M}_n\}_{n \in N}$ is semantically secure with auxiliary input if for any PPT adversary $\mathcal{A}$, any polynomial p, and any sufficiently large $n \in \mathbb{N}$,*

$$\Pr[\mathsf{win}(\mathcal{E}, \mathcal{A}, n) = 1] < 1/2 + 1/p(n),$$

where $\mathsf{win}(\mathcal{E}, \mathcal{A}, n)$:

- *Adversary $\mathcal{A}$ chooses a leakage function L and sends it to the challenger.*
- *Challenger computes $(\mathsf{SK}, \mathsf{PK}) \leftarrow \mathsf{Gen}(1^n)$ and sends $(\mathsf{PK}, L(\mathsf{SK}, \mathsf{PK}))$ to the adversary $\mathcal{A}$.*
- *$\mathcal{A}$ replies with two messages m_0 and m_1.*
- *The challenger flips a coin b and sends $\mathsf{Enc}(\mathsf{PK}, m_b)$ to $\mathcal{A}$.*
- *$\mathcal{A}$ replies with b', its guess for b.*

If $b = b'$ and L is $2^{-\lambda}$-hard-to-invert (A wins), output 1 else output 0.

By $2^{-\lambda}$-hard-to-invert, we mean that for all PPT $\mathcal{B}$, $\Pr[(\mathsf{PK}, \mathsf{SK}) \leftarrow \mathsf{Gen}(1^n), \mathcal{B}(\mathsf{PK}, L(\mathsf{PK}, \mathsf{SK})) = \mathsf{SK}] \leq 1/2^{\lambda}$.

Dodis et al. [DGK+10] show that BHHO is secure with $2^{-l^{\lambda}}$-hard-to-invert auxiliary input. As part of their proof, they extend the Goldreich-Levin theorem to large fields. Thus, let us first state this theorem:

Theorem 7 (Goldreich-Levin for large fields [DGK+10]). *Let q be a prime, and let H be an arbitrary subset of $\mathrm{GF}(q)$. Let $f : H^n \leftarrow \{0,1\}^*$ be any (possibly randomized) function. If there is a distinguisher $\mathcal{D}$ that runs in time t such that*

$$\begin{aligned}&\big|\Pr[\bar{s} \leftarrow H^n, y \leftarrow f(\bar{s}), \bar{r} \leftarrow \mathsf{GF}(q)^n : D(y, \bar{r}, \langle \bar{r}, \bar{s} \rangle) = 1]\\ &- \Pr[\bar{s} \leftarrow H^n, y \leftarrow f(\bar{s}), \bar{r} \leftarrow \mathsf{GF}(q)^n, u \leftarrow \mathsf{GF}(q) : D(y, \bar{r}, u) = 1]\big| = \epsilon,\end{aligned}$$

then there is an inverter $\mathcal{A}$ that runs in time $t' = t \cdot \mathrm{poly}(n, |H|, 1/\epsilon)$ such that

$$\Pr[\bar{s} \leftarrow H^n, y \leftarrow f(\bar{s}) : \mathcal{A}(y) = \bar{s}] \geq \frac{\epsilon^3}{512 \cdot n \cdot q^2}.$$

Claim 8 ([DGK+10]). *Under the DDH assumption, BHHO is secure with $2^{-\ell^{\lambda}}$-hard-to-invert auxiliary input.*

Proof. The construction of the reduction is the same as in the proof of Claim 5: Algorithm 1.

We now consider a sequence of four experiments with either the same or computationally indistinguishable input distributions to the adversary. The last distribution will enable us to prove the claim easily. Let $\mathsf{Adv}_{\mathcal{A}}^{(i)}(n)$ be the advantage of the adversary $\mathcal{A}$ in guessing right in Experiment i for security parameter n.

Experiment 0: This experiment is the same as in Definition 6.

Experiment 1: This is the same experiment as Experiment 0, except that instead of $C = \mathsf{Enc}(\mathsf{PK}, m_b)$, the challenger sends

$$C' = \mathsf{Enc}^*(\mathsf{SK}, m_b) = (g_1^r, \ldots, g_\ell^r, c = \prod_{i=1}^{\ell} g_i^{rs_i} \cdot m_b).$$

Now let us argue that the input distributions to the adversary $\mathcal{A}$ are the same in Experiment 0 and Experiment 1. We can see that for both $\mathsf{Enc}(\mathsf{PK}, m_b)$ and $\mathsf{Enc}^*(\mathsf{SK}, m_b)$, the challenger chooses $r \overset{R}{\leftarrow} \mathbb{Z}_q$. For the same r, we can see that $\mathsf{Enc}^*(\mathsf{SK}, m_b) = \mathsf{Enc}(\mathsf{PK}, m_b)$.

Experiment 2: In this experiment, we have $c_i \overset{R}{\leftarrow} \mathbb{G}$ for $i = 1, \ldots, l$ and $C' = \left(\prod_{i=1}^{\ell} c_i^{s_i}\right) m_b$.

We would like to claim that the advantage of the adversary in Experiments 1 and 2 only differs by a negligible amount.

Claim: If DDH is hard for $\mathbb{G}$, then for every PPT $\mathcal{A}$,

$$|\mathsf{Adv}_{\mathcal{A}}^{(1)}(n) - \mathsf{Adv}_{\mathcal{A}}^{(2)}(n)| \leq \mathsf{negl}(n).$$

Proof: We would like to show that $(\mathsf{PK}, L(\mathsf{SK}, \mathsf{PK}), c_1, \ldots, c_\ell, \prod_{i=1}^{\ell} c_i^{s_i} : c_i \overset{R}{\leftarrow} \mathbb{G})$ (Exp.1) and $(\mathsf{SK}, L(\mathsf{SK}, \mathsf{PK}), g_1^r, \ldots, g_\ell^r, \prod_{i=1}^{\ell} g_i^{rs_i} : r \overset{R}{\leftarrow} \mathbb{Z}_q)$ (Exp. 2) are computationally indistinguishable (where we omitted the distributions from which some random variables are drawn for brevity). Assuming there exists a distinguisher for these two distributions $\mathcal{D}_{12}$, we want to construct a distinguisher $\mathcal{D}_{DDH}$ that breaks the DDH assumption from Lemma 4. Upon receiving input for the general DDH problem $(g_1, \ldots g_\ell, c_1, \ldots c_\ell)$, $\mathcal{D}_{DDH}$ simply generates SK and PK as in the case of the BHHO scheme using $g_1, \ldots, g_\ell$, and provides to $\mathcal{D}_{12}$ $(\mathsf{PK}, L(\mathsf{SK}, \mathsf{PK}), c_1, \ldots, c_\ell, \prod_{i=1}^{\ell} c_i^{s_i})$. $\mathcal{D}_{DDH}$ outputs exactly what $\mathcal{D}_{12}$ outputs and we can see that they have the same winning probability. Therefore $\mathcal{D}_{DDH}$ has nonnegligible advantage of breaking the general DDH problem; by Lemma 4 and assuming DDH, we reach a contradiction.

Experiment 3: In this experiment, C is replaced with $C' = (g^{r_1}, \ldots, g^{r_\ell}, g^u)$ for $r_i \overset{R}{\leftarrow} \mathbb{Z}_q$, $u \overset{R}{\leftarrow} \mathbb{Z}_q$, and some fixed generator g of $\mathbb{G}$.

Now we claim that the advantage of the adversary in Experiments 2 and 3 only differs by a negligible factor:

Claim: For every PPT $\mathcal{A}$, $|\mathsf{Adv}_{\mathcal{A}}^{(2)}(n) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(n)| \leq \mathsf{negl}(n)$.

Proof: In Experiment 2, choosing $c_1, \ldots, c_\ell \overset{R}{\leftarrow} \mathbb{G}$ is equivalent to choosing $r_1, \ldots, r_\ell \overset{R}{\leftarrow} \mathbb{Z}_q$ for some fixed generator g of $\mathbb{G}$. Therefore, we need to prove that $(\mathsf{PK}, L(\mathsf{SK}, \mathsf{PK}), g^{r_1}, \ldots, g^{r_\ell}, \prod_{i=1}^{\ell} g^{r_i s_i} \cdot m_b)$ (see Exp. 2) and $(\mathsf{PK}, L(\mathsf{SK}, \mathsf{PK}), g^{r_1}, \ldots, g^{r_\ell}, \prod_{i=1}^{\ell} g^u)$ (see Exp. 3) are computationally indistinguishable.

Note that it is enough to prove $D_1 = (\mathsf{PK}, L(\mathsf{SK}, \mathsf{PK}), , r_1, \ldots, r_\ell, \langle \overline{r}, \overline{s} \rangle)$ is computationally indistinguishable from $D_2 = (\mathsf{PK}, L(\mathsf{SK}, \mathsf{PK}), , r_1, \ldots, r_\ell, u)$. The reason is that we can reduce the computational indistinguishability of the initial distributions to the computational indistinguishability of D_1 and D_2. The reduction would simply consist of raising g to the power of r_i before feeding to a distinguisher for the second pair of distributions.

We now use Goldreich-Levin for large numbers. From Theorem 7, it follows that if we can distinguish D_1 and D_2 with $\delta > 2^{-l^\lambda/4}$ advantage, we can invert L and obtain SK with probability:

$$q \cdot \frac{\delta^3}{512nq^3} > q \cdot \frac{1}{512n2^{3l\lambda/4}\text{poly}(n)} > q2^{-l\lambda}$$

This contradicts our computational hardness assumption about L; we can thus conclude that Experiments 2 and 3 are computationally indistinguishable.

Note that in Experiment 3, the ciphertext C' sent by the challenger is a random value independent of the bit b, and therefore the adversary has zero advantage of guessing this bit.

By following the sequence of experiment indistinguishability we proved above, the overall adversary advantage of breaking BHHO is at most negligible, thus concluding our proof.

□

4 The GPV Cryptosystem

The GPV cryptosystem [GPV08] is a construction based on lattices. Before we present the cryptosystem, let us present the *Learning with Errors Assumption* on which it is based.

Learning with Errors Assumption (LWE). Consider integers n, m, q and a probability distribution χ on $\mathbb{Z}_q$, typically taken to be a normal distribution that has been discretized. The input is a pair $(A, \bar{v})$ where $A \in \mathbb{Z}_q^{m\times n}$ is chosen uniformly, and $\bar{v}$ is either chosen uniformly from $\mathbb{Z}_q^m$ or chosen to be $A\bar{s} + \bar{x}$ for a uniformly chosen $\bar{s} \in \mathbb{Z}_q^n$ and a vector $\bar{x} \in Z_q^m$ chosen according to χ_m. The assumption is that no PPT $\mathcal{A}$ can distinguish with some non-negligible probability between these two cases.

The GPV cryptosystem is the following bit-encryption scheme. Let n, m, and q be integer parameters of the scheme.

- $\mathsf{Gen}(1^n)$: $\bar{r} \overset{R}{\leftarrow} \{0,1\}^m$, $A \leftarrow \mathbb{Z}_q^{m\times n}$, $\mathsf{SK} = \bar{r}$, $\mathsf{PK} = (A, \bar{r}A)$. Output $(\mathsf{SK}, \mathsf{PK})$.
- $\mathsf{Enc}(\mathsf{PK}, b)$ for $b \in \{0,1\}$: Choose $\bar{s} \overset{R}{\leftarrow} \mathbb{Z}_q^n$, $\bar{x} \overset{R}{\leftarrow} \chi^m$, and $x' \overset{R}{\leftarrow} \chi$. Output $(A\bar{s} + \bar{x}, \bar{r}A\bar{s} + x' + b\lfloor q/2\rfloor)$.
- $\mathsf{Dec}(\mathsf{SK}, (c_1, c_2))$: Compute $c_2 - c_1\bar{r} = b\lfloor q/2\rfloor + x' - \bar{r}\bar{x}$. Output 0 if this value is closer to 0 than to $\lfloor q/2\rfloor$, and output 1 otherwise.

Since $x' - \bar{r}\bar{x}$ is small in comparison to q, we can see that the decryption will return the correct result and the completeness property of $\mathcal{E}$ thus follows.

Claim 9. *GPV is secure with λ-bit of leakage under LWE.*

Proof. As before, we would like to construct a PPT $\mathcal{B}$ that can break LWE with nonnegligible advantage given a PPT $\mathcal{A}$ that can break GPV. $\mathcal{B}$ receives an input of the form (A, y), which could be $(A, \bar{v})$ or $(A, A\bar{s} + \bar{x})$.

The construction for $\mathcal{B}$ is the same as Algorithm 1 except that $\mathsf{Enc}^*_{\mathsf{SK}}(m_b) = (y, \bar{r}y - x' + b\lfloor q/2\rfloor)$, where x' is generated such that the distribution of $rx - x'$ is statistically indistinguishable from the distribution of x' and y is the second term received by $\mathcal{B}$ as input.

Let's consider each case of $\mathcal{B}$'s inputs:

- $\mathcal{B}$ receives $(A, A\bar{s} + \bar{x})$ and therefore $\mathcal{A}$ receives $(A\bar{s} + \bar{x}, A\bar{s}\bar{r} + \bar{x}r - x' + b\lfloor q/2\rfloor)$ for C. Since x' is drawn from a distribution such that $rx - x'$ would induce a statistically indistinguishable distribution, we can see that the inputs to $\mathcal{A}$ will be statistically indistinguishable from what $\mathcal{A}$ expects and therefore, $\mathcal{A}$ will guess the right b with nonnegligible probability.

- $\mathcal{B}$ receives $(A, \bar{v})$. $\mathcal{A}$ receives $(\bar{v}, r\bar{v} - x' + b\lfloor q/2 \rfloor)$. Using the leftover hash lemma, Theorem 1, we can bound by $1/2 + \epsilon/2$ the probability with which $\mathcal{A}$ succeeds in guessing the right b and hence mislead $\mathcal{B}$ into outputting an incorrect bit. We can choose $\epsilon = n^{-\log n}$ to enable $\mathcal{B}$ to maintain the nonnegligible advantage given by the first case.

Combining the two steps, we can see that $\mathcal{B}$ will have nonnegligible probability of breaking LWE, thus reaching a contradiction.

□

When applying LHL, Theorem 1, we obtain $m - n\log q - \lambda \geq 2\log(1/\epsilon)$, therefore, enabling $\lambda \leq m - n\log q - 2\log(1/\epsilon)$ leakage. The GPV cryptosystem can also be proven secure with auxiliary input.

References

[BHHO08] Dan Boneh, Shai Halevi, Mike Hamburg, and Rafail Ostrovsky. Circular-secure encryption from decision Diffie-Hellman. In *Proceedings of the 28th Annual International Cryptology Conference*, CRYPTO '08, pages 108–125, Berlin, Heidelberg, 2008. Springer-Verlag. `http://crypto.stanford.edu/~dabo/abstracts/circular.html`.

[DGK+10] Yevgeniy Dodis, Shafi Goldwasser, Yael Tauman Kalai, Chris Peikert, and Vinod Vaikuntanathan. Public-key encryption schemes with auxiliary inputs. In *TCC*, pages 361–381, 2010.

[DKL09] Yevgeniy Dodis, Yael Tauman Kalai, and Shachar Lovett. On cryptography with auxiliary input. 2009.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th annual ACM symposium on Theory of computing*, STOC '08, pages 197–206, New York, NY, USA, 2008. ACM.

[HSH+09] J. Alex Halderman, Seth D. Schoen, Nadia Heninger, William Clarkson, William Paul, Joseph A. Calandrino, Ariel J. Feldman, Jacob Appelbaum, and Edward W. Felten. Lest we remember: cold-boot attacks on encryption keys. *Commun. ACM*, 52(5):91–98, 2009.

[NR04] Moni Naor and Omer Reingold. Number-theoretic constructions of efficient pseudorandom functions. *J. ACM*, pages 231–262, 2004.

[NS09] Moni Naor and Gil Segev. Public-key cryptosystems resilient to key leakage. In *CRYPTO*, pages 18–35, 2009.