Semantic Security Invariance under Variant Computational Assumptions

Eftychios Theodorakis[⋆] and John C. Mitchell[⋆⋆]

Abstract. A game-based cryptographic proof is a relation that establishes equivalence between probabilistic sequences of actions by real and ideal world players [1]. The author of a proof selects a *hardness assumption system* for their proof upon which to base their subsequent statements. In this paper, we prove the existence of proof-invariant transformations for varying hardness assumptions. We show that for two systems satisfying certain algebraic properties any proof in one system has an equivalent valid proof in the other. This validates Kurosawa's remark [2] about the existence of proof similarities.
Our result implies a correspondence between the Learning With Errors (LWE) problems and both the Elliptic Curve Discrete Log problem (ECDLP) and the Discrete Logarithm (DLOG) problem. To illustrate this result, we provide a series of example transformations in the appendix. The concrete result of this paper is a prototype proof translation tool.

Keywords: semantic security proof, verification, automation, computational assumption, symmetry, invariance

1 Introduction

The foundation of a cryptographic protocol lies in its hardness assumptions. A plethora of such assumptions continues to be proposed [3][4][5][6][7] coinciding with the introduction of innovative new security models. It is common for authors to propose a hardness assumption in order to tackle an open cryptography problem [8][9] or to introduce a new security model [10]. The community later slowly migrates existing protocols to the new proposed assumption in an effort to exploit it further and explore its boundaries beyond the authors' original use. In the process community needs to assess how i) plausible, ii) strong and iii) expressive the assumption is.

In this work we tackle the following question: Knowing a proof of security under assumption $\mathbf{X}$ for functionality ϕ, can one provide a proof of ϕ under a different assumption $\mathbf{Y}$? For instance, if one proves a HIBE construction is fully (or selectively)-secure supposing a bilinear diffie hellman assumption, can they argue about the existence of a construction (or even better derive one) supposing only factoring is hard instead? In fact, recently Döttling and Garg, while introducing an IBE under the CDH assumption in [11], asked if one can

[⋆] eftychios.theodorakis@gmail.com
[⋆⋆] mitchell@cs.stanford.edu

realize other "heavyweight" functionalities under the DH assumption. We answer their question affirmatively.

Specifically, we introduce a framework for reducing relations between two algebras to a semantic security correspondence between two hardness assumptions. We show in certain cases one can create a correspondence between a new hardness assumption and an existing one, such that any existing protocol with a game-based proof of security can be migrated to utilize the new hardness assumption instead, in a manner that preserves its semantic security proof. That is, there exist proof transformations that preserve its soundness and security guarantees. We show this proof invariance is contingent only on the algebraic properties of the two hardness assumption systems. By the term hardness assumption system we denote the couple of hardness (computational) assumption and algebra $\mathcal{L} = (U, \{+, \dots\})$ over a set U (universe) with a set of operands $\{+, \dots\}$. Common examples are the learning with errors (LWE) [4] and $\mathbb{Z}_q^n$, discrete linear (DLIN) and a bilinear group $(G,\ \hat{G},\ G_T,\ e : G \times \hat{G} \to G_T)$ (e.g. over supersingular elliptic curves).

We show that a relation between hardness assumptions A and B and their respective underlying algebras can imply the existence of a transformation between game-based proofs utilizing assumption A to proofs utilizing B (diagram 1). The strength of this method lies in one's ability to prove correctness and soundness in the simpler or pre-existing framework and implement it using a more practical, novel or robust assumption of that family, congruent to the original via these correspondences. This boosts the currently slow exploration process.

$$\begin{tikzcd}
\boxed{\text{Proof } \phi \text{ in } \mathcal{S}} \arrow[r, "\tau"] \arrow[dr, phantom, "(\textbf{IF } \psi \textbf{ THEN } \tau)"] & \boxed{\text{Proof } \phi \text{ in } \mathcal{S}'} \\
\boxed{\mathcal{S} = (A,\, \mathcal{L}_A)} \arrow[u, dashed] \arrow[r, "\psi"] & \boxed{\mathcal{S}' = (B,\, \mathcal{L}_B)} \arrow[u, dashed]
\end{tikzcd}$$

Diagram 1: We reduce any proof transformation (τ) to a hardness assumption system correspondence (ψ). In this case (Hardness Assumption A, Algebra $\mathcal{L}_A$) to (Hardness Assumption B, Algebra $\mathcal{L}_B$).

1.1 Contributions

We obtain proof transformations, i.e. correspondences between semantic security proofs, which differ in their hardness assumption premise. The transformations' soundness is based completely on the relation between the original and target algebra provided a composition preserving transformation between the hardness assumptions. One is able to apply current or new algebraic correspondences to establish equivalences or improve this work regardless of application. We prove these correspondences preserve indistinguishability; specifically one can use a transformation on the original proof to construct a similar one with different assertions. We show the derived proof is sound. If such a transformation is invertible we call the two systems, i.e. couple of algebra and hardness assumptions, proof symmetric. Specifically, we construct such correspondences for surjective homomorphic algebras. Compared to past work we transform the cryptographic proofs, not the protocol. This allows us to utilize and preserve the proof's logic and structure; we preserve soundness rather than validate it as a last step.

Our approach can thus summarize a new lengthy proof into a short list of algebraic transformations, easier to formalize. That is, we derive a certificate of soundness for the transformed proof and do not generate a new one. Existing synthesis and refinement methods are transferable via this invariance – an old refinement method can be used via the transformation in the end system. Concluding, we gain i) an already rich toolkit of protocols ii) an idea of the expressiveness of the new assumption in comparison to existing ones – the "space" of protocols one can prove semantic security for under the new assumption compared to a previous stronger one.

We apply our result to learning with errors and elliptic curve discrete log assumptions as well as the decision linear and bilinear assumptions. We construct a correspondence from the Decision Learning With Errors (D-LWE) assumption to the Elliptic Curve Discrete Log assumption (ECDLP) and the Discrete Logarithm (DLOG). This extends between the decision linear assumption (DLIN) and LIN-LWE and similarly between DDH and DH-LWE.

Current and Future Work Our work is not restricted to the above assumptions. Pairings (and multi-linear maps) appear to be a fertile ground. Furthermore, a rich amount of work exists on algebraic correspondences. We think, the algebraic nature of the lift, indicates the existence and feasibility of establishing correspondences between interesting new assumptions and existing work. Restrictions to our main theorem (71) are also of interest: For instance the $\mathbb{Z}_q^n$ ring is non-abelian; thus there are no non-trivial homomorphisms to abelian (prime) groups. A possible path is to circumvent this impossibility (e.g. via a garbled circuits construction as shown in the appendix or similar to Döttling and Garg [12]) and apply our result to the resulting protocol.

1.2 Building on Prior Work

The literature is inundated with elaborate hardness assumptions. Pairing-based assumptions [10][13] utilizing bilinear maps have a richer repertoire of trapdoors compared to the discrete log assumption, allowing for innovative cryptographic schemes. The advent of quantum computing shed the spotlight to quantum-resistant suspected computational assumptions such as the Learning With Errors (LWE) [4], R-LWE [14] (multiple protocols proposed [15][16]) and to more intricate varieties, such as Strong Isogenes Elliptical Curves [17][18]. The fragility of security protocol design also manifested itself (see [19]). Several assumptions remain impractical; for instance, the recently suggested multilinear subgroup elimination assumption [20] does not hold for multilinear groups [21][22].

In response to the introduction of such complicated assumptions, there has been a focus to i) simplify the analysis of hardness assumptions, ii) mechanize proof generation, iii) automatically synthesize protocols. One tool for vetting the plausibility of an assumption is the generic group model, introduced by Nechaev [23] and Shoup [24]. The generic group model, like the random oracle model, exposes group operations only, hiding intrinsic group structure – the adversary thus can not exploit any properties of the particular group. It is inadequate, however, for evaluating and comparing a new hardness assumption with past work; we miss the tangible link between the different group structures.

Naor [25] suggested the notion of falsifiability, the ability to efficiently verify an adversary's success of breaking the hardness assumption. This allows one to validate whether a proposed computational assumption encompasses the security proof in question. One can then reason about the protocol's provable security. This notion was later extended and simplified by Gentry and Wichs [26]. Goldwasser and Kalai [27] took a step further classifying assumptions into general and concrete, search and decision ones. In this paper we focus primarily on concrete computational hardness assumptions. Our results though naturally extend to generic ones. Boneh et al. introduced a master theorem to vet and associate bilinear pairing assumptions – utilized initially in [28] and detailed in [29]. Barthe et al. in [30] took a step further and introduced a mechanized algorithm for reducing and falsifying hardness assumptions under the generic group model to well-established basic assumptions.

This semi-automated verification is inspired by [31], where Halevi envisioned and argued the need for computer-assisted cryptographic proofs. This vision has seen other recent advances – Easycrypt [32][33][34][35] provides proof assistance by reducing game equivalence of probabilistic Hoare logic to SMT statements. Building on Halevi's dream, one could imagine the scenario where one provides a proof based on an incorrect premise or asking for a protocol alteration or refinement: the proof assistant would derive the new proofs of security and correctness based on the original proof.

These aspects remain unattainable, however, as the nature of formal proof generation is still laborious. Currently, the user first needs to provide a sequence of games and a series of asserted lemmas. The tool reduces game equivalence to a series of subgoal lemmas solved via SMT solver and user collaboration. Altering a hardness assumption or an assertion implies re-validation and rewriting a substantial subset of the goals. As a result, formal verification is an afterthought for completeness for the cryptographer, instead of an integral design step of the thought process.

Automated protocol synthesis algorithms like recent work of Hoang, Katz and Malozemoff [36] aim to solve this problem via automated synthesis. However, the synthesis is based on an authenticated encryption template, which entails a final validation step which ensures soundness along with other desired properties.

Another approach is translation between systems. Akineyele, Garman and Hohenberger [37] provide a translation tool between Type I ($G = \hat{G}$) to Type III (no tractable homomorphism between $G, \hat{G}$ and vice versa) pairing schemes.

In comparison, we reduce transformations of proofs between different computational assumptions down to algebraic requirements. As a result our work is not restricted to a particular protocol type or class of assumptions, as in the above work focused only on pairing schemes. Proof similarities between different systems have been noted before. Our inspiration stems from Kurosawa et al. [2] – they produced an IBE protocol under the DLIN intractability assumption similar in structure to previous work of Agrawal et al. [38]. They note certain similarities may be the result of a connection of DLIN and LWE systems.

The general idea of establishing protocol existence via algebraic properties is not new either; earlier Ostrovsky and Skeith III [39][40] derived constraints on fully homomorphic encryption based on cardinality bounds between maps to provide an impossibility result. Barto in his work [41][42] uses the symmetry provided by the existence of polymorphisms (e.g. Taylor) for CSP instances to prove they can be solved in polynomial time. In this work we take a step further, establishing classes of hardness assumptions.

2 Preliminaries

2.1 Protocol

Security protocols are a concurrent execution of actions executed by finite number of parties. A realized security protocol simulates an ideal protocol where parties can query trusted realizable functionalities. Here we describe protocols in a similar fashion to Hoare's CSP [43] and descendant approaches – like spi calculus [44]. This composability of functionalities view was presented by Canetti in [45]. The differentiating factor of security protocol languages and semantics with communicating protocols in general is the existence of an adversarial environment or participant.

Protocol Language

Definition 1. *We define a protocol as a member of language*

$\langle \mathit{\Pi} \rangle ::= \text{‘}\texttt{seq}\text{’}\ [\langle \mathit{Party} \rangle]\ [\langle \mathit{Func} \rangle]$

$\langle \mathit{Func} \rangle ::= \text{‘}\texttt{func}\text{’}\ [\langle \mathit{Expression} \rangle]$

$$\begin{aligned}
&\langle \mathit{Expression} \rangle ::= \text{‘}\texttt{while}\text{’}\ \langle \mathit{Expression} \rangle \\
&\quad|\quad \langle \mathit{AlgebraicExpression} \rangle \\
&\quad|\quad \langle \mathit{Variable} \rangle\ \text{‘}\texttt{=}\text{’}\ \langle \mathit{AlgebraicExpression} \rangle \\
&\quad|\quad \langle \mathit{Variable} \rangle\ \text{‘}\texttt{=}\text{’}\ \langle \mathit{Distribution} \rangle \\
&\quad|\quad \langle \mathit{Expression} \rangle\ \text{‘}\texttt{==}\text{’}\ \langle \mathit{Expression} \rangle \\
&\quad|\quad \langle \mathit{Expression} \rangle\ \text{‘}\texttt{>}\text{’}\ \langle \mathit{Expression} \rangle \\
&\quad|\quad \langle \mathit{Expression} \rangle\ \text{‘}\texttt{!=}\text{’}\ \langle \mathit{Expression} \rangle
\end{aligned}$$

| ‘`out`’ ⟨*Party*⟩ ⟨*Variable*⟩
| ⟨*Expression*⟩ ⟨*Term*⟩ ⟨*Expression*⟩

⟨*Party*⟩ *::=* ‘`Party`’ ⟨*Natural*⟩

⟨*Natural*⟩ *::=* *.*

⟨*Distribution*⟩ *::=* *.*

⟨*Term*⟩ *::* *=* ‘`return`’
| $\overset{R}{\leftarrow}$

This approach easily generalizes and applies to other languages. We are directing the user to the particular literature for the intricacies of aforementioned languages. Similarly to spi calculus, assume a basic algebra $\mathcal{L}$ and a series of auxiliary terms. The basic building blocks are then the algebraic operations, send (**out**) and receive (**in**) operations, $\overset{R}{\leftarrow}$ sampling oracle and **return** returning the probability the argument boolean expression is true.

Games Proving the security of a protocol is typically modeled ([1][46][35]) as an exchange of moves between at least two probabilistic processes – an adversary and one or more challengers. Our goal is to show the adversary's strategy is negligibly better than arbitrary choice; a strategy is defined as a weighted list of moves dependent on the current state of the system. A game-based proof steps through a particular probabilistic strategy of the challengers; the cryptographer proceeds to show this strategy is equivalent with an ideal construction.

We follow Shoup's approach which models games as probability space functions [1]; each player's move is a transition to a new probability space. A typical game would start with a set of initial values with probability 1. Then the challenger would sample a variable $x \overset{R}{\leftarrow} \mathbb{Z}$, compute $f(x)$ for some function $f(\cdot)$ and give the result to the adversary. In the end the adversary would try to guess the result returning true on success. In the end the game is a function between probability spaces $\omega,\ \omega'$. We start with a null value with probability 1 and map it to a true event – adversary guessed right with a certain probability $[0, 1]$. This has two benefits i) it is easier to show observational equivalence for any two games [35] for an adversary ii) makes it easier to reason about game transformations and composition - as seen in [46].

Example 1. $y = a$ with probability 1 (a constant)
$k \overset{R}{\leftarrow} \mathbb{Z}$
$x = k + y$
Guess:
$g = A(y)$
$return\ x == g$

Formally,

Definition 2. *A game P_l, for a protocol $\mathcal{P}$, is modeled as a function $P_l \in \mathcal{P} : \mathcal{D}_{init} \to \mathcal{D}_{final}$, from an initial distribution $\mathcal{D}_{init}$ over some probability space $X = (\Omega, \mathcal{F}, P)$ to $\mathcal{D}_{final}$ over $X' = (\Omega', \mathcal{F}', P')$, X, X' representing the state of the system.*

For convenience a game,

$$\begin{gathered}\textbf{Party 1:}\\ x_1 \overset{R}{\leftarrow} \{0,1\}\\ \textit{out}\text{ Party 2 } x_1\\ \textbf{Party 2:}\\ x_2 = x_1 + 1\end{gathered}$$

can be written in a more functional fashion as

$$(+)(1,\ \text{Party 2 (out Party 2 }.x_1 = \overset{R}{\leftarrow} \{0,1\}(\text{ Party 1 })(\mathit{1}))) \tag{1}$$

where *Party* $1(\mathit{1})$ initializes party 1 with an initial distribution $\mathit{1}-$ a no operation, out exposes the result to Party 2 . The final distribution $\mathcal{D}_{\text{final}}$ is $\Pr[x_1 = 1] = \Pr[x_1 = 0] = 1/2, \Pr[x_2 = 1] = \Pr[x_2 = 2] = 1/2$. Terms Party and out ensure type soundness and access – which party can modify and access which variables; = provides us with a naming directive.

2.2 Game-based Proofs

We consider game sequence based proofs [1]. A proof then is a sequence of games, starting from an ideal to a realizable game (Real). The real game makes use of no trusted parties or any ideal constructions. We use the notation G_I, G_R for ideal and real game respectively. A complete proof is a game transition from an ideal to a real game. Recall game transitions are an equivalence relation.

Defining hereafter:

$$\text{Adv}_{\mathcal{A}}(\beta, \gamma) \overset{\text{def}}{=} \| Pr\left(\mathcal{A}(\beta) = 1\right) - Pr\left(\mathcal{A}(\gamma) = 1\right) \|$$

($\mathcal{A}$ a probabilistic polynomial time (PPT) algorithm) and also:

$$\text{Negl.} \overset{\text{def}}{=} 1/n^{O(1)} \tag{2}$$

Hence

$$[equiv] \frac{\forall \mathcal{A},\ \ \text{Adv}_{\mathcal{A}}(G_i, G_j) < \text{Negl.}}{G_i \sim G_j} \tag{3}$$

e the identity element in $\mathcal{L}$ and $\sim$ an equivalence relation.

$$[proves] \frac{G_I^{\phi} \sim G_R^{\phi}}{\text{Proof}^{\phi}} \tag{4}$$

See Shoup [1], Nowak [46] and Barthe et al. [35] for detailed analysis and examples. For convenience, we will use derivation notation to describe proofs (assuming always a true premise); a proof Π of R in system $\mathcal{T}$ is the deduction denoted as

$$\underset{R}{\Pi \parallel \mathcal{T}} \tag{5}$$

2.3 The Diffie-Hellman Family of Assumptions

The Diffie Hellman problem has become the foundation of modern cryptography [47]. In this section we introduce the family of problems succinctly. For more information the reader is directed to [48].

Computational and Decisional Diffie-Hellman Problems Let group G_p of order $\|p\|$, generator g, $n = \log_2 \|p\|$ and $a, b \in \mathbb{Z}$. Then

Definition 3. *Suppose a group $G = (g, p)$ with a randomly chosen generator g. The Computational Diffie-Hellman Assumption states that there is no probablistic polynomial-time algorithm $\mathcal{A}$ able to efficiently compute the value g^{ab} provided with (g, g^a, g^b) with non-negligible probability. Specifically,*

$$\forall \mathcal{A} : \; Pr(\mathcal{A}(G = (g, p), g^a, g^b) = g^{ab}) < \frac{1}{n^{O(1)}} \tag{6}$$

Definition 4. *The Decisional Diffie-Hellman Assumption states that there is no algorithm $\mathcal{A}_D$ solving the Decisional Diffie-Hellman problem with non-negligible probability. Specifically,*

$$\mathrm{Adv}_{\mathcal{A}}((G, g^a, g^b, g^c), (G, g^a, g^b, g^{ab})) < \frac{1}{n^{O(1)}} \tag{7}$$

with g^c uniformly sampled.

2.4 Decision Linear (DLIN)

Definition 5. *The Decision Linear (DLIN Assumption states that there is no PPT algorithm $\mathcal{A}_D$ able to distinguish g^{a+b} from a uniformly sampled g^c with non-negligible probability. Specifically*

$$\mathrm{Adv}_{\mathcal{A}}((G, g^a, g^b, g^c), (G, g^a, g^b, g^{a+b})) < \frac{1}{n^{O(1)}} \tag{8}$$

2.5 LWE and R-LWE Assumptions

The learning with errors (LWE) assumption is based on the hardness of the shortest vector approximation problem ($\gamma - SVP$)[4].

Definition 6. *Learning with Errors assumption implies that any PPT adversary $\mathcal{A}$ is unable to distinguish between $As + e \mod q$ and $u \mod q$ where u is randomly chosen from $\mathbb{Z}^n$ and $A \in \mathbb{Z}^{m \times n}$, $s \in \mathbb{Z}^n$ and e follows a noise distribution $\mathcal{D}$. Hence*

$$Adv_{\mathcal{A}}((A, As + e), (A, u)) < \frac{1}{n^{O(1)}} \tag{9}$$

3 Main Result and Overview

In this paper we consider computational assumptions and the underlying algebra as proof variables. For example, consider a proof of CCA2-IND semantic security of a new primitive under LWE. One can substitute the ECDH assumption with Elliptic Curve Diffie-Hellman assumption (ECDH) and transform the original proof into a new one. We coin the term *translation* and work on introducing a *translator* τ. Hence, this paper studies the following type of transformations (τ):

$$\tau : \mathit{\Pi} \underset{R}{\parallel} \mathcal{T}, \mathcal{A} \rightarrow \mathit{\Pi}' \underset{\tau \circ R}{\parallel} \mathcal{T}', \mathcal{A}' \qquad (10)$$

3.1 Example: El Gamal Encryption in Elliptic Curves from LWE public key encryption

Let us try to derive the El Gamal Encryption in Elliptic Curves proof (fig. 3a, fig. 3b) from the following simple public key encryption primitive proof under LWE (see [14]).

```
Init Phase:
$L = L(\mathbb{Z}_q^n)$,
$(M_0, M_1) =\overset{R}{\leftarrow} \mathbb{Z}_q^{m\times 1})$
$s \overset{R}{\leftarrow} \mathbb{Z}_q^{n\times 1}$ (Bob's secret)
$A \overset{R}{\leftarrow} \mathbb{Z}_q^{m\times n}, B = As + e,$
(Bob's public key, $e$  noise)
Alice:
$k \overset{R}{\leftarrow} \mathbb{Z}_q^{m\times m}$
$a =\overset{R}{\leftarrow} \{0,1\})$
$y = kA + e, x = kB + M_a + e$
$out$ Bob $.(x, y)$
Guess:
$b = A(x, y)$
return b==a
```

(a) Real Game for CPA proof of public key encryption from LWE

```
Init Phase:
$(M_0, M_1) =\overset{R}{\leftarrow} \{0,1\})$
Alice:
$a =\overset{R}{\leftarrow} \{0,1\})$
$(x, y)$ :
$x, y = \mathcal{O}(M_a)$
$out$ Bob $.(x, y)$
Guess:
$b = A(x, y)$
return b==a

with $\mathcal{O} : M \rightarrow X \times Y$ and
$\mathcal{O}_D : X \times Y \rightarrow M$ s.t.
$\mathcal{O}_D(\mathcal{O}(m)) = m$ and
$\mathcal{O}_D(x', y') =\perp$ otherwise.
```

(b) Ideal Game for CPA proof of public key encryption from LWE

Diagram 2: LWE Public key encryption CPA proof [14]

We can write the real game:

$$\begin{aligned}
&\text{El Gamal Public key CPA Game} = \text{Guess } \circ \text{ Alice round } \circ \text{ Party (Bob, Alice) init}\\
&\text{Guess} = \text{func return } b == a, b = A(x, y)\\
&\text{Alice\_Round} = \text{func out Party Bob }.(x, y)(y = kA + e,\\
&x = kB + M_a + e, k = \xleftarrow{R} \mathbb{Z}_q^{m\times m}, a = \xleftarrow{R} \{0, 1\})\\
&\text{init} = \text{func (Party Bob } (s \xleftarrow{R} \mathbb{Z}_q^{n\times 1}), L = \mathbb{Z}_q^n), A \xleftarrow{R} \mathbb{Z}_q^{m\times n},\\
&, B = As + e, (M_0, M_1) = \xleftarrow{R} \mathbb{Z}_q^{1\times m})
\end{aligned}$$

and respectively the ideal one:

$$\begin{aligned}
&\text{El Gamal Public key CPA Game} = \text{Guess } \circ \text{ Alice round } \circ \text{ Party (Bob, Alice) init}\\
&\text{Guess} = \text{func return } b == a, b = A(x, y)\\
&\text{Alice\_Round} = \text{func out Party Bob }.(x, y)(y = \mathcal{O}(k), x = \mathcal{O}_{\text{Encrypt}}(\mathcal{O}_{\text{KE}}(k, B), M_a),\\
&k = \xleftarrow{R} \mathbb{Z}, a = \xleftarrow{R} \{0, 1\})\\
&\text{init} = \text{func (Party Bob } (s \xleftarrow{R} \mathbb{Z}), B = \mathcal{O}(s), (M_0, M_1) = \xleftarrow{R} \mathcal{M})
\end{aligned}$$

with $\mathcal{M}$ an arbitrary large set.

The proofs consists of two game transitions. First we substitute oracle $\mathcal{O}(\cdot)$ sampling queries for public and private parameters with sampling from $\mathbb{Z}_q^n$ (s_a). Afterwards we substitute the encryption and private key retrieval queries with point multiplication operations ($s_{\mathbf{LWE}}$). We can express the real game as the composition of the above game transitions on the ideal game:

$$\text{Real Game} = s_{\mathbf{LWE}} \circ s_a \circ \text{Ideal Game}$$

We write the elliptic curve equivalent of the above El gamal public key encryption as follows (see Koblitz original paper [49]).

$$\begin{aligned}
&\text{El Gamal Public key CPA Game} = \text{Guess } \circ \text{ Alice round } \circ \text{ Party (Bob, Alice) init}\\
&\text{Guess} = \text{func return } b == a, b = A(x, y)\\
&\text{Alice\_Round} = \text{func out Party Bob }.(x, y)(y = kP, x = kB + M_a,\\
&k = \xleftarrow{R} \mathbb{Z}, a = \xleftarrow{R} \{0, 1\})\\
&\text{init} = \text{func (Party Bob } (s \xleftarrow{R} \mathbb{Z}), E = E(\mathbb{F}_q), P \in E, B = sP, (M_0, M_1) = \xleftarrow{R} E)
\end{aligned}$$

We first define the parameters in init: we pick an elliptic curve is E, sample secret s under Bob's context, and pick a point P in E as public parameter. The ideal game is written similarly to the original CPA proof under LWE. Similarly,

$$\text{Real Game El Gamal ECDLP} = s_{\mathbf{ECDLP}} \circ s'_a \circ \text{Ideal Game}$$

To derive the CPA proof for the El Gamal public key encryption (fig. 3) under ECDH from the LWE proof we define the following transformation:

$$\tau = \begin{cases} h : \mathbb{Z}_q^n \twoheadrightarrow E(\mathbb{F}_p) \\ s_{\mathbf{LWE}} \mapsto s_{\mathbf{ECDLP}} \end{cases}$$

and

$$\frac{\tau(f \circ g)}{\tau(f) \circ \tau(g)}, \quad \frac{\tau(x, y)}{\tau(x), \tau(y)}$$

s.t.

$$\begin{aligned} \tau(\text{Real Game LWE}) &= \tau(s_{\mathbf{LWE}}) \circ \tau(s_a) \circ \tau(\text{Ideal Game}) \\ &= s_{\mathbf{ECDLP}} \circ s'_a \circ \tau(\text{Ideal Game}) \\ &= \text{Real Game El Gamal ECDLP} \end{aligned}$$

We derived the proof[1] as a transformation of the original LWE proof.

```
Init Phase:
$E = E(\mathbb{F}_q)$, $(M_0, M_1) =\overset{R}{\leftarrow} E)$
$s \overset{R}{\leftarrow} \mathbb{Z}$ (Bob's secret)
$P \in E, B = sP$, (Bob's public key)
Alice:
$k \overset{R}{\leftarrow} \mathbb{Z}$
$a =\overset{R}{\leftarrow} \{0, 1\})$
$(x, y)$ :
    $y = kP, x = kB + M_a$
    out Bob $.(x, y)$
Guess:
$b = A(x, y)$
return b==a
```

(a) Real Game for El Gamal Public Key Encryption

```
Init Phase:
$(M_0, M_1) =\overset{R}{\leftarrow} \{0, 1\})$
Alice:
$a =\overset{R}{\leftarrow} \{0, 1\})$
$(x, y)$ :
    $x, y = \mathcal{O}(M_a)$
    out Bob $.(x, y)$
Guess:
$b = A(x, y)$
return b==a

with $\mathcal{O} : M \to X \times Y$ and
$\mathcal{O}_D : X \times Y \to M$ s.t.
$\mathcal{O}_D(\mathcal{O}(m)) = m$ and
$\mathcal{O}_D(x', y') = \perp$ otherwise.
```

(b) Ideal Game for El Gamal Public Key Encryption

Diagram 3: El Gamal CPA proof

[1] We have to consider the soundness proof too. This is simpler as it is a pair of equalities $(x - sy = M)$

3.2 Outline of approach

In the rest of the paper we shall prove the existence and properties of game transition transformations. We divide the problem as follows:

1. Prove the existence of an equivalent ideal game (lemma 2)
2. Prove every mirror transformation exists – $G_{ij} \Rightarrow G'_{ij}$ in the new system (definition & lemma 5)
3. Prove every mirror transformation is a game transition, i.e. an adversary has negligible advantage to distinguish between the two games (lemma 4, lemma 8)

Main Theorems Consequently, we prove (section 7) that:

Theorem 71. *Suppose security parameter s, an algebraic surjective homomorphism $h_s : \mathcal{L}_s \to \mathcal{L}'_s$, with $\{\mathcal{L}_s\}$, $\{\mathcal{L}'_s\}$ families of finite algebras in the standard sense (first isomorphism theorem holds and equipped with an identity element), an S_0 s.t. for all $s > S_0$: $\frac{|\mathcal{L}_s|}{|\mathcal{L}'_s|} s^c$ negligible in s for any $c < 0$. Also let theories $T = T_\star \cup \{A\}$ and $T' = T_\star \cup \{A'\}$, such that $\eta : A \to A'$ natural. Then a proof correspondence exists from system $(T, \mathcal{L})$ to $(T', \mathcal{L}')$. Namely, for any functionality ϕ with proof $\varPi_\phi$ using security parameter s in $(T, \mathcal{L})$ there is a proof $\varPi'_\phi$ in $(T', \mathcal{L}')$ satisfying the same security model.*

Theorem 72. *Suppose algebras $\mathcal{L}$, $\mathcal{L}'$ and consistent theories with $T, T' – T = T_\star \cup \{A\}$ and $T' = T_\star \cup \{A'\}$, with A, A' natural. If there is a weak equivalence between $\mathcal{L}$ and $\mathcal{L}'$ then there exists a proof symmetry between proofs in system $(\mathcal{L}, T)$ and $(\mathcal{L}', T')$. For every proof $\varPi^\phi$ of $(\mathcal{L}, T)$ there exists $\varPi^\phi$ of $(\mathcal{L}, T)$ and conversely.*

4 Model

We model games as categories. This allows us to focus on the properties and composability of game transitions as an algebraic consequence. Then a proof being a game transition naturally forms a 2-category itself. Specifically, let protocol $\varLambda$ contain all valid distribution functions, and a game $\lambda(D_i)$ for some distribution D_i and $\lambda \in \varLambda$. A game transition is then

$$\begin{aligned} \gamma : \text{Game} &\to \text{Game}' \\ \text{s.t. } \text{Adv}(\text{Game}, \text{Game}') &< \text{Negl.} \end{aligned}$$

And a proof $\varPi$ (see diagram 4) is:

$$\begin{gathered} \varPi = (\gamma, G_I, G_R) \\ \text{s.t. } G_R = \gamma(G_I) \end{gathered}$$

Hence here we argue about the existence of game transition transformations, that allow one to substitute a hardness assumption with a different one. First let us formalize the above notions.

.

4.1 Category of Games

Lemma 1. *A game sequence of parties $\mathcal{P}$ expressed in a protocol language Λ over an algebra L is a category:*

- *class of subdistributions $(\mathcal{D}_0, \mathcal{D}_1, \dots)$ as its objects*
- *games, functions of subdistributions $\mathcal{D}_x \to \mathcal{D}_y$, as morphisms*
- *$x \mapsto x$ as the identity mapping*
- *game composition as the morphism composition*

We denote the category containing all games realizing functionality ϕ $\mathfrak{G}_\phi$ and the category containing all games $\mathfrak{G}$.

Example 2. Consider the object to be the variable $V \in \mathbb{Z}_q$ with morphisms operations *in*, *out* and addition in the $< \mathbb{Z}_q, + >$. Two parties x and y can construct a simple messaging protocol for instance - or a mutual exclusion schema if one assumes atomicity.

4.2 Category of Proofs

We defined a proof as a sequence of, equivalent for an adversary, games. This then forms a category of games. Naturally the 2-category of equivalent game transitions encapsulates the notion of proofs, which is a chain of game equivalent game transformations.

Definition 7. $\mathrm{Morph}\,(\mathcal{C})$ *is the set containing all morphisms of category $\mathcal{C}$.*

Definition 8. $\mathrm{Obj}(\mathcal{C})$ *is the set containing all objects of category $\mathcal{C}$.*

Definition 9. *We define Proof category $\mathcal{P}$ for functionality ϕ*

- *the set of games $\{G\}_{\mathrm{Morph}\,\mathcal{G}_\phi}$ as its objects*
- *set game transitions as morphisms*
- *$\mathrm{id} : G \mapsto G$ as the identity mapping*
- *game transition composition as the composition*

Proof. Note the composition is sequential application of game transitions (rule application). In that sense we define the identity morphism as being the null substitution - unique up to isomorphism $\left(\dfrac{\Gamma \vdash \Delta, A \;\; Z \vdash E}{\Gamma \vdash \Delta, A} \overset{\sim}{\to} \dfrac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A}\right)$. Associativity is a direct result of logic application. □

Definition 10. *A functor $F : C \to D$ is a transformation from category C to category D that preserves structure, i.e.*

$$\forall A, B \in \mathrm{Obj}(C) \exists F(A),\; F(B) \text{ and } \forall f, g \in \mathrm{Morph}\, C : B = g(A) \Rightarrow$$
$$F(f \circ g) = F(f) \circ F(g) \text{ and } F(\mathrm{id}_A) = id_{F(A)}$$

We can reword our problem as follows now:

Definition 11. *A translator τ_D^C is a functor from proof category C to proof category D.*

Namely, if there is a proof $\text{Proof}_{\mathcal{F}}^C$ under assumption A then, given a functor τ_D^C, there is a proof $\text{Proof}_{\mathcal{F}}^D = \tau_D^C(\text{Proof}_{\mathcal{F}}^C)$ under assumption B. Recall, it is necessary one i) preserves composition, ii) maps game transitions in C to game transitions in D.

4.3 Computational Assumptions

A computational hardness assumption is an assertion of a proof, based on the (conjectured) computational infeasibility of a particular problem. In particular,

Definition 12. *[27] A decisional computational assumption $A(\mathcal{D}_0, \mathcal{D}_1)$ is an assertion that a pair of distributions $\mathcal{D}_0, \mathcal{D}_1$ are equivalent for every Probabilistic Polynomial Time (PPT) adversary $\mathcal{A}$. Namely,*

$$\Pr_{b \xleftarrow{R} \{0,1\}, x \xleftarrow{R} \mathcal{D}_b} [\mathcal{A}(x) = b] < 1/2 + \frac{1}{poly(n)} \tag{11}$$

with n input size of the security parameter.

Assumption $A(\mathcal{D}_0, \mathcal{D}_1)$ is thus written as an equivalence relation $\dfrac{R}{T}$ – R being the premise and T the conclusion.

Examples For instance, one can write **DDH** as the pair

$$[\alpha]\frac{g^{ab} \quad a, b \in \mathbb{Z}}{u \quad u \leftarrow \text{sample}\, G} \qquad [\rho]\frac{u \quad u \leftarrow \text{sample}\ G}{g^{ab} \quad a, b \in \mathbb{Z}} \tag{12}$$

implying both $\alpha \overset{T}{\underset{R}{\|}} \mathcal{A}$ and $\rho \overset{R}{\underset{T}{\|}} \mathcal{A}$. Equivalence of premise and conclusion can be expressed by a pair of game transitions, as the following diagram exhibits.

$$\begin{tikzcd}
X = g, g^a, g^b \arrow[r, shift left, "\alpha"] \arrow[d, "g^{ab}"] & X_I \arrow[l, shift left, "\rho"] \arrow[d, "u \leftarrow \text{sample}\ G"] \\
Y \arrow[r, shift left, "\alpha"] & Y_I \arrow[l, shift left, "\rho"]
\end{tikzcd}$$

Definition 13. *Similarly, a search computational (search) assumption is an assertion that for every efficient (PPT) algorithm/adversary $\mathcal{A} : X \to Y$, given a pair of polynomial time algorithms $(\mathcal{D}, \mathcal{V})$ (instance sampler and verifier):*

$$\Pr_{r \xleftarrow{R} \{0,1\}^n,\ x = \mathcal{D}(r)} [\mathcal{A}(x) = y \ s.t. \ \mathcal{V}(x, y, r) = 1] < \frac{1}{poly(n)} \tag{13}$$

Here we use the most liberal definition of a privately-verifiable search hardness assumption in [27]. In this case we provide the randomness to the verifier function $\mathcal{R}$. If we restrict access of the randomness for the verifier we define a "classical" search computational assumption. We can also extend the verifier to a t-search problem by bounding the probability of counting $t(n)$ witnesses (for more details see [27]). Hence, a search assumption is an equivalence with a simulated function $S : \_ \rightarrow Y$, with no access to x, uniformly sampling Y.

Example 3. **CDH** implies for an adversary $\mathcal{A}$ any function $f : g^a \times g^b \mapsto g^{ab}$ is equivalent to a function $r : g^a \times g^b \mapsto u \overset{R}{\leftarrow} G$.

One may then represent a hardness assumption A a a pair of functors

$$F^A : C \rightleftarrows D : G^A \tag{14}$$

Example 4. **DDH** can be written as the following pair of game transitions. Recall these are endofuncttors – they map a category to itself:

$$\overrightarrow{\mathcal{DDH}} = \begin{cases} (g^a \times g^b \mapsto g^a \times g^b \times g^{ab}),\ (g^a \times g^b \mapsto g^a \times g^b \times \mathcal{U}_{\mathbb{Z}_p}) \\ \text{id}, \quad \text{otherwise} \end{cases}$$

$$\overleftarrow{\mathcal{DDH}} = \begin{cases} (g^a \times g^b \mapsto g^a \times g^b \times \mathcal{U}_{\mathbb{Z}_p}),\ (g^a \times g^b \mapsto g^a \times g^b \times g^{ab}) \\ \text{id}, \quad \text{otherwise} \end{cases}$$

We denote with $\mathcal{U}_X$ a type with the properties of a random sample from set X:

$$\begin{aligned} \mathcal{U} + v &\approx_c \mathcal{U} \\ v\mathcal{U} &\approx_c \mathcal{U} \\ \forall v &\in X \end{aligned}$$

For instance,

$$(g^a, b, M) \mapsto g^{ab} + M$$

can be written as

$$(g^a, b, M) \mapsto \overrightarrow{\mathcal{DDH}}((g^a, b) \mapsto g^b, g^a, \mathcal{U}) + M$$

Observe a hardness assumptions can be considered as a parametric type proof: given a specific data type one could substitute it with another type with similar operators. We argue this allows us define transformations between hardness assumptions.

Definition 14. *Let functors $F, G : C \rightarrow D$. A natural tranformation η is a morphism such that $\eta_X : F(X) \rightarrow G(X)$ for every object X of C and for all $f : X \rightarrow Y$ holds $F(f) \circ \eta_Y = \eta_X \circ G(f)$.*

Essentially a natural morphism between two hardness assumptions ensures we can write any game transition or functionality in the former assumption using the latter assumption. The converse also holds: if one can transform any functionality from one assumption to a different assumption then that transformation is natural. However, this does not imply game transitions are preserved! Note the natural transformation ensures the preservation of composition.

$$\begin{tikzcd}
G_I : X \arrow[r, "F_{I1}"] \arrow[d, "\mu_0^I"] & G_1 : X \arrow[r, "F_{12}"] \arrow[d, "\mu^1_{XY}"] & \cdots \arrow[r, "F_{..R}"] & G_R : X \arrow[d, "\mu^R_{XY}"] \\
G_I : Z \arrow[r, "F_{I1}"] \arrow[d, "\mu^I_{YZ}"] & G_1 : Z \arrow[r, "F_{12}"] \arrow[d, "\mu^1_{YZ}"] & \cdots \arrow[r, "F_{..R}"] & G_R : Z \arrow[d, "\mu^R_{YZ}"] \\
G_I : Y \arrow[r, "F_{I1}"] \arrow[d, "\mu^I_{Z..}"] & G_1 : Y \arrow[r, "F_{12}"] \arrow[d, "\mu^1_{Z..}"] & \cdots \arrow[r, "F_{..R}"] & G_R : Y \arrow[d, "\mu^R_{Z..}"] \\
G_I : \cdots \arrow[r, "F_{I1}"] & G_1 : \cdots \arrow[r, "F_{12}"] & \cdots \arrow[r, "F_{..R}"] & G_R : \cdots
\end{tikzcd}$$

Diagram 4: A proof is a sequence of game transitions F_{ij} from an ideal to a real game. (In particular, $F_{IR} = F_{Ix} \circ \cdots \circ F_{yR}$. A game $G : X \to Y$ itself is a composition of games; for instance hybrid game $G^A_{XY} = \mu^A_{ZY}(\mu^A_{XZ})G^A_X$. The advantage for $\mathcal{A}$ is $\mathrm{Adv}_{\mathcal{A}}(G_I, G_R) = \sum\limits_{A \in [I,\ldots,R-1]} \mathrm{Adv}_{\mathcal{A}}(\mu^A_{ZY}(\mu^A_{XZ})G^A_X, \mu^{A+1}_{ZY}(\mu^{A+1}_{XZ})G^{A+1}_X)$.

5 Constructing a symmetric proof

5.1 Algorithm Outline

In this section we construct an equivalent ideal game (lemma 2). An ideal game describes the functionality as a sequence of message exchanges, trusted party queries and basic algebraic operations. We argue a functor preserves each component and their composition. We show in the next section (6) there are correspondences that preserve computational indistinguishability of a distribution for all PPT algorithms. Specifically, we show the minimal set of properties for such a correspondence to exist and show that a surjective homomorphism between two families of algebras is one. We also show a weak equivalance between two algebras forms such a correspondence. We combine the above to prove our main statement 71, 72 in section 7. We assume we are given an existing proof implied by an asserted assumption (fig. 5a). To ensure a new hardness assumption (fig. 5b) can construct protocols simulating at least the same trusted parties we need a natural mapping between the two hardness assumptions (fig. 5c). Now we can construct all game transitions (fig. 4) leading to aforementioned theorems 71 and 72.

5.2 Generating a Corresponding Ideal Game

Definition 15. *A security model of a protocol is the collection of security properties the protocol must satisfy against specific adversarial attacks, expressed in the form of an ideal game.*

Common Examples: IND-CPA, IND-CCA1,NM-CPA

Namely, an ideal game is a protocol execution specifying functionality ϕ via abstract or ideal operations. It describes the relationship between the objects

$$\begin{tikzcd}
X \arrow[r, "u"] \arrow[d, "\alpha_X"] & Y \arrow[d, "\alpha_Y"] \\
X' \arrow[r, "v"] & Y'
\end{tikzcd}$$

(a) The assumption α in our theory $(u \simeq_c v)$ is expressed by the above diagram.

$$\begin{tikzcd}
M \arrow[r, "\mu"] \arrow[d, "\beta_M"] & N \arrow[d, "\beta_N"] \\
M' \arrow[r, "\kappa"] & N'
\end{tikzcd}$$

(b) A new assumption β $(\mu \simeq_c \kappa\ )$ expressed by a similar diagram.

$$\begin{tikzcd}
 & X \arrow[rr, "u"] \arrow[dl, "\alpha_X"] \arrow[dd] & & Y \arrow[dl, "\alpha_Y"] \arrow[dd] \\
M \arrow[rr, "\mu"] \arrow[dd] & & N \arrow[dd] & \\
 & X' \arrow[rr, "v"] \arrow[dl, "\beta_M"] & & Y' \arrow[dl, "\beta_N"] \\
M' \arrow[rr, "\kappa"] & & N' &
\end{tikzcd}$$

(c) There is an algebraic correspondence between assumptions.

of the system, hence highlighting its structure. In that sense it is abstract; one describes how protocol participants interact with each other via ideal trusted functionalities. It is devoid of any hardness assumption constructs and it is consistent. We need then to first provide the specification corresponding to the same security model utilizing the new algebra. Specifically, an ideal game G can be written as

$$G = \mathrm{Game}\, g(\mathcal{F}_0, \mathcal{F}_1, \cdots, \mathcal{F}_k) \tag{15}$$

for ideal trusted functionalities $\boldsymbol{\mathcal{F}} = \{\mathcal{F}_0, \mathcal{F}_1, \cdots, \mathcal{F}_k\}$, a function g, and a type constructor Game.

One can write the derived ideal game G_I^B as the composition of the original G_I^A and a tranformation h that preserves "enough structure" – $G_I^B = h \circ G_I^A$. This is not a necessary condition but a sufficient one.

Lemma 2. *An ideal game transformation into a new system contains only algebra related sequents. Suppose a functor F between the two proof categories; then the game $G'_I = F(G_I)$ is ideal, consistent and satisfies the same security model, provided $F(\mathcal{F}_i)$ is ideal functionality for all $\mathcal{F}_i$.*

Proof. G_I is a game $g(\mathcal{F}_0, \mathcal{F}_1, \cdots, \mathcal{F}_k)$. Rewrite this as $g \circ \mathcal{F}_0 \times \mathcal{F}_1 \times \cdots \times \mathcal{F}_k$. Applying functor F we get $F(g) \circ F(\mathcal{F}_0 \times \mathcal{F}_1 \times \cdots \times \mathcal{F}_k)$. We know the latter term exists because the two theories are consistent (by above assertion). By definition, for any games z, y, w $z \circ G_I \circ y = z \circ g \circ \mathcal{F}_0 \times \mathcal{F}_1 \times \cdots \times \mathcal{F}_k \circ y = w$ it holds that $F(z) \circ F(G_I) \circ F(y) = F(w)$. □

6 Semantic Security Preserving Transformations

6.1 Outline of our argument

We want to show that any map A_H satisfies an indistinguishability property $V(A_H)$, i.e.

$$\mathrm{Adv}_{\mathcal{A}}(A_H) < \kappa \tag{16}$$

with κ a negligible amount for any $\mathcal{A}$ adversary. As you can see in diagram 6 our approach is as follows: we pick an adversary in $\mathcal{L}'$ A_H. A_H is in the image of F. We show if one increases the advantage above a value, we can find an adversary in the original system that contradicts our initial advantage inequality (this is due to reflection r).

$$\begin{tikzcd}
\mathcal{L} \arrow[d, "F"'] \arrow[rd, shift left, "F"] & \\
\mathcal{C} \in \mathrm{Obj}(\mathcal{L}') \arrow[r, "A_H"'] & \mathcal{D} \in \mathrm{Obj}(\mathcal{L}') \arrow[lu, shift left, "r"]
\end{tikzcd}$$

Diagram 6: Let $F : \mathcal{L} \to \mathcal{L}'$. We want to show that there is a correspondence r reflecting all A_H to $\mathcal{L}$. One utilizes the reflection to reason about which properties A_H mappings preserve.

6.2 Master Lemma and Extensions

We generalize Lemma 1 in [50] - group homomorphism preserves uniform distribution - for any algebra. The result follows by applying the fundamental homomophism theorem.

Lemma 3. *Given an (algebraic) surjective homomorphism $h : \mathcal{L}_A \to \mathcal{L}_B$, a uniform random variable X over $\mathcal{L}_A$ then $y \in \mathcal{L}_A$,* $\Pr[h(X) = y] = \frac{\|\ker h\|}{\|G\|}$.

Hence, similarly to [50], Li et al; we can write for the entropy H

$$H(h(X)) = \log \frac{\|\mathcal{L}\|}{\|\ker h\|} \tag{17}$$

Proof.

$$\forall v \in \mathcal{L}_B \Pr[h(X) = v] - \Pr[h(Y) = v] \le \delta' \tag{18}$$

Given $\Pr[X = u] - \Pr[Y = u] \le \delta$ and h surjective homomorphism with $\ker(h) = k$, one can see that in worst case all elements of a subset of size $\|k\|$ will map to a single element. Hence

$$\Pr[h(X) = v] - \Pr[h(Y) = v] \le k\delta = \delta' \tag{19}$$

□

Ideally h has kernel $k = \ker(h) \leq \frac{1}{n^{O(1)}}$. We show in proposition 61 the kernel cardinality is not related to the size of the security variable, but is an intrinsic value of the homomorphism. This result is not surprising though - any special attributes and characteristics of the algebra used to solve the problem are transplanted into the new algebra via the homomorphism.

$$\begin{tikzcd}
\mathcal{L}_A \arrow[d, "h"'] \arrow[dr, shift left] & \\
\mathcal{L}_B \arrow[r, "A_H"'] & \mathcal{L}_B \arrow[ul, shift left]
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
\mathcal{L}_A \arrow[d, "h"'] \arrow[dr] & \\
\mathcal{L}_B \arrow[r, "A_H"'] & \mathcal{L}_B \subseteq \mathcal{S} \simeq \mathcal{L}_A
\end{tikzcd}$$

Diagram 7: One way to preserve the original statistical indistinguishability property of $\mathcal{L}_A$ algebra is to find a relation between $\mathcal{L}_A$ and $\mathcal{L}_B$, a mapping $\mathcal{L}_A$ to $\mathcal{L}_B$. For example, showing $\mathcal{L}_B$ is a subset of $\mathcal{L}_A$.

Let us consider a toy case. If $\mathcal{L}_B \subseteq \mathcal{L}_A$ we have an arrow $\mathcal{L}_B \to \mathcal{L}_A$ connecting any morphism between domain and codomain. Particularly we have that $A = A_H \circ h$ and hence $\delta' = \delta$.

Example 5. Applying the same reasoning solving the roots of a polynomial bounded via the Schwartz-Lippel lemma (as usually applied in the generic group model):

$$\Pr\left[P(r) = 0\right] \leq \frac{d}{\|S\|},\ P \in \mathbb{F}[x_0, \ldots, x_n],\ r \in \mathbb{R}^n \tag{20}$$

for maximum degree d and set S, can be bounded by

$$\Pr[h(P(r)) = 0_h] \leq \frac{d|\ker(h)|}{\|S\|} \tag{21}$$

Lemma 3 gives us $\Pr\left[h(P(r)) = 0_h\right] \leq \frac{d}{\|h(S)\|} = \frac{d|\ker(h)|}{\|S\|}$ – due to following proposition 61.

The above result is a statistical bound for uniform distributions. In the following section we extend our argument for any distribution sampled by a PPT adversary.

6.3 Preserving Distributions

Preserving Distribution Indistinguishability We want to derive an upper bound for the adversarial advantage of any game transition between two games G_0, G_1 in the target system $(T', \mathcal{L}')$. We argue that given two distributions $\mathcal{D}, \mathcal{E}$ in $\mathcal{L}$ with $\mathrm{Adv}_A(\mathcal{D}, \mathcal{E}) \leq \delta$ for all adversaries $A \in PPT(\mathcal{L} \to \mathcal{L})$, we can derive a bound $\mathrm{Adv}_B(\mathcal{D}', \mathcal{E}') \leq \delta'$ for $\mathcal{E}' = h \circ \mathcal{E}, \mathcal{D}' = h \circ \mathcal{D}$ for all adversaries B.

Specifically we need δ' to be negligible ($\delta' \leq \delta + \epsilon = \delta + \frac{1}{n^{O(1)}}$). Consider ϵ as the deviation due to incomplete information.

We first argue at this point the cardinality of the kernel of a surjective homomorphism is a constant dependent only on the cardinality of the universes. We will use that to bound the advantage deviation only in terms of the underlying algebras.

Proposition 61. *Suppose $h : \mathcal{L} \to \mathcal{L}'$ as surjective homomorphism. Then its kernel $k = \frac{\|\mathcal{L}\|}{\|\mathcal{L}'\|}$.*

Proof. From the first (generalized) isomorphism theorem of algebra we have that

$$\mathcal{L}/\ker h \simeq Im(h) \tag{22}$$

Because h is onto, it holds

$$\mathcal{L}/\ker h \simeq Im(h) = \mathcal{L}' \tag{23}$$

From the Langrange theorem it follows that

$$\begin{align} |\mathcal{L}| = |\mathcal{L}/\ker h||\ker h| \Rightarrow \tag{24} \\ |\mathcal{L}| = |\mathcal{L}'||\ker h| \Rightarrow \tag{25} \\ |\ker h| = \frac{|\mathcal{L}|}{|\mathcal{L}'|} \tag{26} \end{align}$$

□

The following lemma and its corollaries follow.

Lemma 4. *Suppose two distributions $\mathcal{D}, \mathcal{E}$ on $\mathcal{L}$ a finite algebra in the standard sense (satisfying the first isomorphism theorem and having an identity element) with $\mathrm{Adv}_A(\mathcal{D}, \mathcal{E}) \leq \delta$ for any PPT adversary $A : \mathcal{L} \to \mathcal{L}$ and a surjective homomorphism $h : \mathcal{L} \to \mathcal{L}'$. Assume we can compute the algebraic operations of $\mathcal{L}$, $\mathcal{L}'$ in polynomial time. It holds that for any adversary $\mathcal{B}$ in $\mathcal{L}'$, $\mathrm{Adv}_{\mathcal{B}}(h(\mathcal{D}), h(\mathcal{E})) \leq \frac{|\mathcal{L}|}{|\mathcal{L}'|}\delta$. The minimum bound δ is achieved only for an isomorphism.*

Proof. We have that

$$\forall B : \mathcal{L}' \to \mathcal{L}', \exists A : \mathcal{L} \to \mathcal{L} \tag{27}$$

s.t.

$$\begin{tikzcd} \mathcal{L} \arrow[r, "A"] \arrow[d, two heads, "h"'] & \mathcal{L} \arrow[d, two heads, "h"] \\ \mathcal{L}' \arrow[r, "B"'] & \mathcal{L}' \end{tikzcd}$$

commutes. Then we have that for distributions D', E' on $\mathcal{L}'$:

$$|Pr[B(D') = e] - Pr[B(E') = e]| = \tag{28}$$

$$= |Pr[BhD = e] - Pr[BhE = e]| \tag{29}$$

$$= |Pr[hAD = e] - Pr[hAE = e]| \tag{30}$$

$$= \left| \sum_{g \in \ker h} (Pr_D[A = g] - Pr_E[A = g]) \right| \tag{31}$$

$$\leq \sum_{g \in \ker h} |(Pr_D[A = g] - Pr_E[A = g])| \tag{32}$$

$$\leq |\ker(h)|\,|(Pr_D[A = g] - Pr_E[A = g])| \text{ for some } g \tag{33}$$

$$\leq |\ker(h)| \max_{g \in \ker(h)} |(Pr_D[A = g] - Pr_E[A = g])| \tag{34}$$

Let there be g' maximizing point then there exists A'

$$A^{-1}(g') = A'^{-1}(e)$$

for instance

$$A'(x) = \begin{cases} A(x), \; g \neq x \overset{R}{\leftarrow} D \\ e, \; g = x \overset{R}{\leftarrow} D \end{cases}$$

If A is a PPT algorithm we can see A' and thus any B are PPT[2]. Conversely, if B is a function computed by a PPT algorithm then A' has to be computed by a PPT algorithm. Then

$$|\ker(h)| \max_{g \in \ker(h)} |(Pr_D[A = g] - Pr_E[A = g])| = \tag{35}$$

$$|\ker(h)|\,|(Pr_D[A' = e] - Pr_E[A' = e])| \tag{36}$$

$$\leq |\ker(h)|\delta = \frac{|\mathcal{L}|}{|\mathcal{L}'|}\delta \tag{37}$$

□

Notice that $\kappa = \frac{|\mathcal{L}|}{|\mathcal{L}'|} \geq 1$, with equality holding only for h isomorphism. This is optimal as we do not gain any further information by applying the transformation. Notice also δ remains negligible in the new security variable as a result.

Corollary 61. *Assume two indistinguishable distributions $\mathcal{D}$, $\mathcal{E}$ with advantage* $\mathrm{Adv}_{\mathcal{A}}(\mathcal{D}, \mathcal{E}) \leq \delta$ *and a surjective homomorphism h with kernel* $\ker(h)$ *s.t.* $\ker(h)\delta \leq 1/n^{O(1)}$*, with n the security variable size. The homomorphism h preserves indistinguishability, i.e. for any adversary* $\mathrm{Adv}(h(\mathcal{D}), h(\mathcal{E}))$ *is negligible.*

Next we show that faithful functors reflect semantic security and a proof equivalence arises in the case of equivalence of two proof categories. This extends our results to arbitrary algebras.

[2] Recall $h(f \circ g) = h(f) \circ h(g)$

6.4 Soundness preserving Functors

Puncturing method Given a concrete category C with functions $u, v : X \to Y$ as morphisms and functor $F : C \to D$ for which holds

$$\begin{aligned} r \leq & \Pr_{x \overset{R}{\leftarrow} X, X \in \mathrm{Obj}(C)} [u(x) = v(x)] = \\ = & \Pr_{\tilde{v}_r, \tilde{x} \in X, X \in \mathrm{Obj}(C)} [u(\tilde{x}) = \tilde{v}_r(\tilde{x})] \end{aligned}$$

We construct P(C) with

$$\tilde{v}_r = \begin{cases} u, & \text{with probability r} \\ v_{\sim u} \ s.t. u \neq v_{\sim u}, & \text{otherwise} \end{cases}$$

Then we have

$$\begin{aligned} \Pr_{x \overset{R}{\leftarrow} X, X \in \mathrm{Obj}(C)} [F(u(x)) = F(v(x))] &= \frac{1}{\|Z\|} \sum_{z \in Z} \mathbb{1}(F(u)(z) = F(\tilde{v}_r)(z)) \\ &= \frac{1}{\|Z\|}(r\|Z\| + (1-r)\mathbb{1}(F(u)(z) = F(\tilde{v}_{\sim u})(z)) \geq r \end{aligned}$$

Lemma 5. *Given a functor $F : \mathcal{C} \to \mathcal{D}$ we can construct functor $F' : P(C) \to D$ that preserves soundness, i.e. for any $u, v \in \mathrm{Morph}\,\mathcal{C}$*

$$\Pr_{x \overset{R}{\leftarrow} X, X \in \mathrm{Obj}(C)} [u(x) = v(x)] \geq 1 - \epsilon \Rightarrow \tag{38}$$

$$\Pr_{z \overset{R}{\leftarrow} F(X)} [F(u)(z) = F(v)(z)] \geq (1 - \epsilon) \tag{39}$$

for arbitrary ϵ.

6.5 Faithful functors reflect indistinguishability

Similarly, we can make a contrapositive statement assuming a reflective transformation. Suppose the assertion A_H is indistinguishable from random for an adversary $\mathcal{A}$ – see diagram 6. If transformation h reflects indistinguishability and $h \circ A = A_H$, then A is also indistinguishable for $\mathcal{A}_{\mathcal{L}}$.

Lemma 6. *Let $\mathcal{A}, D, \mu$ PPT algorithms, X game.* $\mathrm{Adv}_{\mathcal{A}}(u, v) \leq \delta$ *for all $\mathcal{A}$ is equivalent to*

$$\forall D \forall \mu : \Pr_{x \sim X}[D(u \circ \mu(x)) = D(v \circ \mu(x))] \geq 1 - \delta \tag{40}$$

$\mu : X \to X$.

Proof. Proved by contradiction. Assume there is a distinguisher pair for which $\Pr_{x \overset{R}{\leftarrow} X}[D(u \circ \mu(x)) = D(v \circ \mu(x))] < 1 - \delta$ then we can build an adversary with advantage greater than δ. Converse follows similarly.

□

Lemma 7. *Suppose there is a faithful functor $F : C \to D$ and for some $f, g : F(X) \to F(Y) \in \mathrm{Morph}(C)$ it holds: $\mathrm{Adv}_{\mathcal{A}}(F \circ f, F \circ g) \leq \kappa$ for all $\mathcal{A}$ PPT algorithms in D. Also $F(X) \sim X$. Then it holds that $\mathrm{Adv}_{\mathcal{A}_C}(f, g) \leq \kappa' \leq \kappa$, for all $\mathcal{A}_C$ PPT.*

The result follows by applying the puncturing method, lemma 6 and faithful functor definition.

6.6 Weak equivalence preserves indistinguishability

Consider a functor $F : C \to D$ between C, D locally small categories. F implies the $F_{X,Y} : \hom_C(X, Y) \to \hom_D(F(X), F(Y))$ mapping.

Definition 16. *F is a full functor if and only if $F_{X,Y}$ is surjective for each set of mappings $(F(X), F(Y))$.*

Definition 17. *F is a faithful functor if and only if each function $F_{X,Y}$ is injective.*

Definition 18. *Let C, D proof categories. A strong translator is an injective on game transitions (faithful) proof transformation $\tau_C^D : C \to D$. Namely for all games $f, g \in \mathrm{Obj}(C)$ if $\tau_C^D \circ g = \tau_C^D \circ f$ then $f = g$ and*

$$f \simeq_c g \Leftrightarrow \tau_C^D \circ f \simeq_c \tau_C^D \circ g$$

$\simeq_c$ denotes computational indistinguishability.

We remark here that a strong translator also preserves the proof structure, i.e. all subproofs are also valid.

Lemma 8. *A full and faithful functor $F : C \to D$ with a left adjoint $G : D \to C$ is a strong translator τ_D^C.*

The existence of a full and faithful functor with a left adjoint implies weak equivalence of the two categories. Recall then that F is object surjective. We defer the detailed proof to the appendix of the full version of the paper.

7 Main Theorems

Definition 19. *Theories $T = T_\star \cup \{M\}$ and $T' = T_\star \cup \{M'\}$ are consistent if $\nexists \sigma$ s.t. $T_\star,\ M \vDash \sigma,\ T_\star$ and $M' = \neg\sigma$.*

Now we can derive the two main theorems of the paper.

Theorem 71. *Suppose security parameter s, an algebraic surjective homomorphism $h_s : \mathcal{L}_s \to \mathcal{L}'_s$, with $\{\mathcal{L}_s\}$, $\{\mathcal{L}'_s\}$ families of finite algebras in the standard sense (first isomorphism theorem holds and equipped with an identity element), an S_0 s.t. for all $s > S_0$: $\frac{|\mathcal{L}_s|}{|\mathcal{L}'_s|}s^c$ negligible in s for any $c < 0$. Also let theories $T = T_\star \cup \{A\}$ and $T' = T_\star \cup \{A'\}$, such that $\eta : A \to A'$ natural. Then a proof correspondence exists from system $(T, \mathcal{L})$ to $(T', \mathcal{L}')$. Namely, for any functionality ϕ with proof Π_ϕ using security parameter s in $(T, \mathcal{L})$ there is a proof Π'_ϕ in $(T', \mathcal{L}')$ satisfying the same security model.*

Proof. We first apply lemma 2 constructing an ideal game in $(T', \mathcal{L}')$ for $\phi, \mathcal{S}$. We decompose original game transitions comprising of the hardness assumption. For each other game transition s_{ij} ($G_j = s_{ij}G_i$) we apply the master lemma 4, applying $h(s_{ij}^{-1}) = h(s_{ji})$. As $\frac{|\mathcal{L}|}{|\mathcal{L}'|}$ is independent of the security variable s, every new game transition can be bounded by $\frac{1}{\text{poly}(s)}$. Without loss of generality assume a single invocation of the hardness assumption asserted diagram. Then

$$\text{Adv}(G'_I, G'_R) = \text{Adv}(G'_I, G'_X) + \text{Adv}(G'_Y, G'_R) + \text{Adv}(s_{A'}) \qquad (41)$$

with G'_I, G'_R the new ideal and real games respectively. $G'_Y = s_{A'}G_X$ and the advantage of $s_{A'}$ is negligible due to the hardness assumption. Note that at this point we have constructed only games $[G'_I \ldots G'_X]$. Using $A \mapsto A'$ (naturality condition) we can construct a game transition $s_{A'}$ such that $G'_X \simeq_c G'_Y$. Summing the advantages of all new hybrid games we have $\text{Adv}(G'_I, G_R) <$ Negl. □

We call $\mu = \frac{\|\mathcal{L}\|}{\|\mathcal{L}'\|}$ our transition magnification factor. The ratio of the advantage between ideal game and real game of the new proof over the original is the total magnification factor.

Theorem 72. *Suppose algebras $\mathcal{L}$, $\mathcal{L}'$ and consistent theories with $T, T' – T = T_\star \cup \{A\}$ and $T' = T_\star \cup \{A'\}$, with A, A' natural. If there is a weak equivalence between $\mathcal{L}$ and $\mathcal{L}'$ then there exists a proof symmetry between proofs in system $(\mathcal{L}, T)$ and $(\mathcal{L}', T')$. For every proof Π^ϕ of $(\mathcal{L}, T)$ there exists Π^ϕ of $(\mathcal{L}, T)$ and conversely.*

We work similarly with theorem 71 instead using lemma 8.

Remarks Notice that the converse also holds, i.e. if there exists a surjective homomorphism or a weak equivalence and the naturality condition does not hold we can construct at least one proof for which the correspondence does not work.

Corollary 71. *Assume surjective homomorphism h between algebras $\mathcal{L}$, $\mathcal{L}'$ and theories $T = T_\star \cup \{\phi\}$ and $T' = T_\star \cup \{\phi'\}$ with T, T' consistent, with $\eta : A \to A'$. If a proof symmetry exists between systems $(T, \mathcal{L})$ and $(T', \mathcal{L}')$, i.e. for every proof Π^ϕ of $(\mathcal{L}, T)$ there exists Π'^ϕ of $(\mathcal{L}', T')$, then η is natural.*

Follows by contradiction: suppose there is a proof symmetry, i.e. for every proof there is a corresponding one in the new system; then we see the naturality diagram commutes. In a similar manner:

Corollary 72. *Suppose theories $T = T_\star \cup \{A\}$ and $T' = T_\star \cup \{A'\}$, such that $\eta : A \to A'$ natural. If there is a proof correspondence from system $(T, \mathcal{L})$ to $(T', \mathcal{L}')$ via a transformation τ, then $\tau = (\tau_T, \tau_L)$ satisfies the following properties:*

- $\tau_L : \mathcal{L} \to \mathcal{L}'$ *surjective*
- $\tau(T, g \circ f) = \tau(T, g) \circ \tau(T, f)$

8 Connections between various Hardness Assumptions

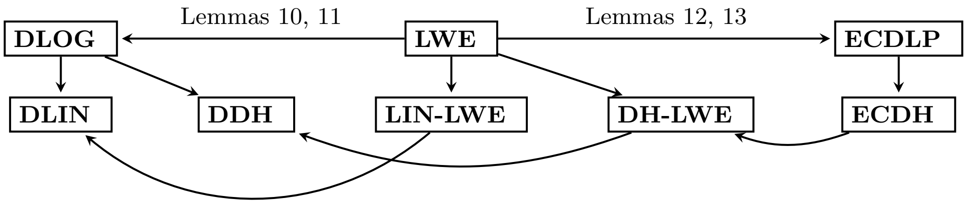

Diagram 8: Proof correspondences between LWE, DLOG, ECDH and their derivations.

In this section we give as examples the relationship between some well known and utilized computational assumptions. We construct a surjective homomorphism from lattices Z_q^n to prime group Z_p (q, p prime) and bound the magnification factor and similarly between $\mathbb{Z}_q^n$ to a curve $E(\mathbb{Z}_p)$. We use this to connect LWE to DLOG problem by showing and a natural tranformation exists. We extend this result to define a correspondences between **DH-LWE** and **DDH** and between the LWE - LIN and DLIN assumptions. Both aforementioned homomorphisms are tractable. We do not derive any efficiency guarantees.

Magninification Factor Bound First we are going to bound the magnification factor for a surjective homomorphism between prime groups.

Lemma 9. *Let surjective homomorphism $\psi : \mathbb{Z}_q^n \twoheadrightarrow \mathbb{Z}_p^m$. We can always find two primes p, q s.t. the magnification factor $\rho = \frac{q^n}{p^m}$ of ψ is subpolynomial (in fact $O(1)$) with the right choices of n, m.*

Proof. Consider a surjective homomorphism $\psi : \mathbb{Z}_q^n \twoheadrightarrow \mathbb{Z}_p^m$. We have $|\ker(\psi)| = \frac{|\mathbb{Z}_q^n|}{|\mathbb{Z}_p^m|}$. For adversaries $\mathcal{A}, \mathcal{B}$ assume

$$\mathrm{Adv}_{\mathcal{A}} < 1/\operatorname{poly}(s) \tag{42}$$

$$\text{Then } \mathrm{Adv}_{\mathcal{B}}(\psi \circ \sigma) = \mathrm{Adv}_{\mathcal{B}}(\psi \circ \sigma^{-1}) \leq \tag{43}$$

$$\leq \frac{q^n}{p^m} 1/\operatorname{poly}(s) = \frac{q^n}{p^m} 1/\operatorname{poly}(n \log q) \tag{44}$$

If the degree of δ polynomial bound is t we want

$$\frac{q^n}{p^m} = o(n^t \log q) \tag{45}$$

Denote the prime gap as $g_\nu = p_{\nu+1} - p_\nu$ with $p_\nu, p_{\nu+1}$ consecutive primes. We will show that we can tweak our parameters n, m to keep the maginification factor, $\frac{q^n}{p^m}$, subpolynomial (sublinear in fact), assuming large enough $p > x_0$, $x_0 \in \mathbb{N}$. Hoheisel [51] showed initially that $g_n < p_n^\theta$, for $\theta < 1$. Baker and Harman [52] with Pintz [53] improved θ to 0.525. We will consider two cases of interest here.

Case I : We can pick two primes q, p with small gap, i.e. $g_n < \max\{q, p\}$. Then $\lim_{s\to\infty} \frac{q^n}{p^m} = \frac{p^{m(1+\theta)}}{p^m} = 1$.

Case II : We pick $p >> q$ with $q^n > p$ $(m = 1)$. Worst case q^n is $p_{\nu+1} - 2$. However we know there exists a prime p s.t. $p_{\nu+1} - p = g < p^\theta$. Then

$$1 \leq \lim_{s\to\infty} \frac{q^n}{p} \leq \lim_{p\to\infty} \frac{(p + p^\theta - 2)}{p} = 1 \tag{46}$$

. □

Next we will define explicitly the aforementioned hardness assumptions.

Definition 20. *The Discrete Log (**DLOG**) hardness assumption system is denoted as the tuple* $((\overrightarrow{\mathcal{DLOG}}, \overleftarrow{\mathcal{DLOG}}), (\mathbb{Z}_p, *))$ *with*

$$\overrightarrow{\mathcal{DLOG}} = \begin{cases} y \mapsto g^y,\ y \mapsto \mathcal{U} \\ \mathrm{id}, \quad \textit{otherwise} \end{cases} \quad \textit{and} \quad \overleftarrow{\mathcal{DLOG}} = \begin{cases} y \mapsto \mathcal{U}, y \mapsto g^y \\ \mathrm{id}, \quad \textit{otherwise} \end{cases}$$

endofunctors (they map the category to itself).

Recall a functor maps game transitions.

Definition 21. *The Elliptic Curve Discrete Log (**ECDLP**) hardness assumption system is denoted as the tuple* $((\mathcal{EC}\overrightarrow{\mathcal{D}}\mathcal{LP}, \mathcal{EC}\overleftarrow{\mathcal{D}}\mathcal{LP}), (E(\mathbb{F}_q), \cdot))$ *with*

$$\mathcal{EC}\overrightarrow{\mathcal{D}}\mathcal{LP} = \begin{cases} s \times P \mapsto s \cdot P,\ s \times P \mapsto \mathcal{U} \\ \mathrm{id}, \quad \textit{otherwise} \end{cases} \quad \textit{and} \quad \mathcal{EC}\overleftarrow{\mathcal{D}}\mathcal{LP} = \begin{cases} s \mapsto \mathcal{U}, s \mapsto s \cdot P \\ \mathrm{id}, \quad \textit{otherwise} \end{cases}$$

endofunctors, $P \in E(\mathbb{F}_p), s \in \mathbb{Z}$.

Definition 22. *The Learning With Errors (**LWE**) hardness assumption system is denoted as the tuple* $((\overset{\rightarrow}{\mathcal{DLWE}}, \overset{\leftarrow}{\mathcal{DLWE}}), (\mathbb{Z}_q^n, +))$ *with*

$$\overset{\rightarrow}{\mathcal{DLWE}} = \begin{cases} A \times s \mapsto A \times As + e, \ s \mapsto \mathcal{U} \\ \text{id}, \quad \textit{otherwise} \end{cases} \quad \textit{and} \quad \overset{\leftarrow}{\mathcal{DLWE}} = \begin{cases} s \mapsto \mathcal{U}, \ A \times s \mapsto A \times As + e \\ \text{id}, \quad \textit{otherwise} \end{cases}$$

endofunctors. $A \in \mathbb{Z}_q^{m\times n}, s \in \mathbb{Z}_q^{n\times 1}, e$ *noise in* $\mathbb{Z}_q^{m\times 1}$.

8.1 LWE To DLOG Proof Correspondence

Relation between $\mathbb{Z}_q^n$ and $\mathbb{Z}_p$

Lemma 10. *There is a surjective homomorphism* $(\mathbb{Z}^n/q\mathbb{Z}, +) \to (\mathbb{Z}_p, *)$ *for some* q, p *co-prime and some* $n > N$ *constant.*

Proof. Let g be generator of Z_p and

$$A = [a_0, \dots, a_n] \in \mathbb{Z}_q^n$$

We define y s.t.

$$h(x, A) = g^{y(x,A)} \mod p = g^{\sum\limits_{i=0}^{n-1} a_i x^{n-i-1}} \mod p$$

h homomorphism for some x. Let us set x=q.

then h is surjective as $q^n\mathbb{Z}_q + q^{n-1}\mathbb{Z}_q + \dots + \mathbb{Z}_q \cap [0, p) = [0, p)$ for some $n > N$ given that $(kp = \alpha q + r)$ for some $k, \alpha, r < q$

We can find p from lemma 9. □

Naturality Condition

Lemma 11. *There is a natural transformation* η *between* $\overset{\rightarrow}{\mathcal{DLWE}}$ *and* $\overset{\rightarrow}{\mathcal{DLOG}}$. *Same for their opposite* $\overset{\leftarrow}{\mathcal{DLWE}}$ *and* $\overset{\leftarrow}{\mathcal{DLOG}}$.

Proof. We want to show there is η s.t. for every game transition $f : X \to Y$, $\eta_X \circ \overset{\rightarrow}{\mathcal{DLWE}} = \overset{\rightarrow}{\mathcal{DLOG}} \circ \eta_Y$. Construct η so it maps $s \mapsto U \to (A \times s \mapsto A \times As + e)$ transitions to $y \mapsto U \to (y \mapsto g^y)$. Utilizing the homomorphism $A \sum s \twoheadrightarrow g^{\sum y}$ we get the above equality for every transition f. □

8.2 LWE to ECDLP Proof Correspondence

Relation between $E(\mathbb{C})$ and $\mathbb{Z}_q^n$

Lemma 12. *There is a surjective homomorphism* $(\mathbb{Z}_q^n, +) \to (E(\mathbb{Z}_p), +)$

Proof. Assume $h_k : \mathbb{Z}_q^n \twoheadrightarrow \mathbb{Z}_k$ surjective homomorphism for $k \in \{w, m\}$, with k co-prime to w and m. We have that $E(\mathbb{Z}_p) \simeq \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/w\mathbb{Z}$.

$$\mathbb{Z}_q^{2n} \xrightarrow{\sim} \mathbb{Z}^{2n}/q\mathbb{Z} \xrightarrow{\sim} \mathbb{Z}^n/q\mathbb{Z} \times \mathbb{Z}^n/q\mathbb{Z} \twoheadrightarrow h_m(\mathbb{Z}/m\mathbb{Z}) \times h_w(\mathbb{Z}/w\mathbb{Z})$$ □

Naturality Condition

Lemma 13. *There is a natural transformation η between $\overrightarrow{\mathcal{DLWE}}$ and $\overrightarrow{\mathcal{ECDLP}}$. Same for their opposite $\overleftarrow{\mathcal{DLWE}}$ and $\overleftarrow{\mathcal{ECDLP}}$.*

Proof. As above. □

Note lemmas 10, 11 and 12, 13 satisfy theorem 71.

References

1. Shoup, V.: Sequences of games: a tool for taming complexity in security proofs. IACR Cryptology ePrint Archive **2004** (2004) 332
2. Kurosawa, K., Trieu Phong, L.: Leakage resilient ibe and ipe under the dlin assumption. In Jacobson, M., Locasto, M., Mohassel, P., Safavi-Naini, R., eds.: Applied Cryptography and Network Security. Volume 7954 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2013) 487–501
3. Pass, R., Seth, K., Telang, S.: Indistinguishability obfuscation from semantically-secure multilinear encodings. In: International Cryptology Conference, Springer (2014) 500–517
4. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. Journal of the ACM (JACM) **56**(6) (2009) 34
5. Peikert, C.: Public-key cryptosystems from the worst-case shortest vector problem. In: Proceedings of the forty-first annual ACM symposium on Theory of computing, ACM (2009) 333–342
6. Brakerski, Z., Langlois, A., Peikert, C., Regev, O., Stehlé, D.: Classical hardness of learning with errors. In: Proceedings of the forty-fifth annual ACM symposium on Theory of computing, ACM (2013) 575–584
7. Blum, A., Kalai, A., Wasserman, H.: Noise-tolerant learning, the parity problem, and the statistical query model. Journal of the ACM (JACM) **50**(4) (2003) 506–519
8. Pandey, O., Pass, R., Vaikuntanathan, V.: Adaptive one-way functions and applications. In: Annual International Cryptology Conference, Springer (2008) 57–74
9. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Advances in Cryptology–CRYPTO 2004, Springer (2004) 41–55
10. Boneh, D., Franklin, M.: Identity-based encryption from the weil pairing. In: Annual International Cryptology Conference, Springer (2001) 213–229
11. Döttling, N., Garg, S.: Identity-based encryption from the diffie-hellman assumption. In: Annual International Cryptology Conference, Springer (2017) 537–569
12. Döttling, N., Garg, S., Hajiabadi, M., Masny, D.: New constructions of identity-based and key-dependent message secure encryption schemes. Cryptology ePrint Archive, Report 2017/978 (2017) `https://eprint.iacr.org/2017/978`.
13. Boldyreva, A., Gentry, C., O'Neill, A., Yum, D.H.: Ordered multisignatures and identity-based sequential aggregate signatures, with applications to secure routing. In: Proceedings of the 14th ACM conference on Computer and communications security, ACM (2007) 276–285
14. Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. Journal of the ACM (JACM) **60**(6) (2013) 43
15. Bos, J.W., Costello, C., Naehrig, M., Stebila, D.: Post-quantum key exchange for the tls protocol from the ring learning with errors problem. In: Security and Privacy (SP), 2015 IEEE Symposium on, IEEE (2015) 553–570

16. Costello, C., Longa, P., Naehrig, M.: Efficient algorithms for supersingular isogeny diffie-hellman
17. Galbraith, S., Stolbunov, A.: Improved algorithm for the isogeny problem for ordinary elliptic curves. Applicable Algebra in Engineering, Communication and Computing **24**(2) (2013) 107–131
18. Childs, A., Jao, D., Soukharev, V.: Constructing elliptic curve isogenies in quantum subexponential time. Journal of Mathematical Cryptology **8**(1) (2014) 1–29
19. Koblitz, N., Menezes, A.: The brave new world of bodacious assumptions in cryptography. Notices of the American Mathematical Society **57**(3) (2010) 357–365
20. Gentry, C., Lewko, A.B., Sahai, A., Waters, B.: Indistinguishability obfuscation from the multilinear subgroup elimination assumption. In: Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on, IEEE (2015) 151–170
21. Minaud, B., Fouque, P.A.: Cryptanalysis of the new multilinear map over the integers. IACR Cryptology ePrint Archive **2015** (2015) 941
22. Cheon, J.H., Fouque, P.A., Lee, C., Minaud, B., Ryu, H.: Cryptanalysis of the new clt multilinear map over the integers. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer (2016) 509–536
23. Nechaev, V.I.: Complexity of a determinate algorithm for the discrete logarithm. Mathematical Notes **55**(2) (Feb 1994) 165–172
24. Shoup, V. In: Lower Bounds for Discrete Logarithms and Related Problems. Springer Berlin Heidelberg, Berlin, Heidelberg (1997) 256–266
25. Naor, M.: On cryptographic assumptions and challenges. In: Annual International Cryptology Conference, Springer (2003) 96–109
26. Gentry, C., Wichs, D.: Separating succinct non-interactive arguments from all falsifiable assumptions. In: Proceedings of the forty-third annual ACM symposium on Theory of computing, ACM (2011) 99–108
27. Goldwasser, S., Kalai, Y.T.: Cryptographic assumptions: A position paper. In: Theory of Cryptography Conference, Springer (2016) 505–522
28. Boneh, D., Boyen, X., Goh, E.J.: Hierarchical identity based encryption with constant size ciphertext. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer (2005) 440–456
29. Boyen, X.: The uber-assumption family. In: International Conference on Pairing-Based Cryptography, Springer (2008) 39–56
30. Barthe, G., Fagerholm, E., Fiore, D., Mitchell, J., Scedrov, A., Schmidt, B.: Automated analysis of cryptographic assumptions in generic group models. In: International Cryptology Conference, Springer (2014) 95–112
31. Halevi, S.: A plausible approach to computer-aided cryptographic proofs. IACR Cryptology ePrint Archive **2005** (2005) 181
32. Barthe, G., Grégoire, B., Heraud, S., Béguelin, S.Z.: Computer-aided security proofs for the working cryptographer. In: Annual Cryptology Conference, Springer (2011) 71–90
33. Barthe, G., Crespo, J.M., Grégoire, B., Kunz, C., Lakhnech, Y., Schmidt, B., Zanella-Béguelin, S.: Fully automated analysis of padding-based encryption in the computational model. In: Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, ACM (2013) 1247–1260
34. Barthe, G., Grégoire, B., Béguelin, S.Z.: Probabilistic relational hoare logics for computer-aided security proofs. In: International Conference on Mathematics of Program Construction, Springer (2012) 1–6
35. Barthe, G., Grégoire, B., Zanella Béguelin, S.: Formal certification of code-based cryptographic proofs. ACM SIGPLAN Notices **44**(1) (2009) 90–101

36. Hoang, V.T., Katz, J., Malozemoff, A.J.: Automated analysis and synthesis of authenticated encryption schemes. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, ACM (2015) 84–95
37. Akinyele, J.A., Garman, C., Hohenberger, S.: Automating fast and secure translations from type-i to type-iii pairing schemes. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, ACM (2015) 1370–1381
38. Agrawal, S., Boneh, D., Boyen, X.: Lattice basis delegation in fixed dimension and shorter ciphertext hierarchical ibe. In: Proc. of Crypto'10. Volume 6223 of LNCS. (2010) 98–115
39. Ostrovsky, R., Skeith III, W.E.: Algebraic lower bounds for computing on encrypted data. IACR Cryptology ePrint Archive **2007** (2007) 64
40. Ostrovsky, R., Skeith Iii, W.E.: Communication complexity in algebraic two-party protocols. In: Annual International Cryptology Conference, Springer (2008) 379–396
41. Barto, L.: The constraint satisfaction problem and universal algebra. The Bulletin of Symbolic Logic (2015) 319–337
42. Barto, L., Kozik, M., Niven, T.: Graphs, polymorphisms and the complexity of homomorphism problems. In: Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing. STOC '08, New York, NY, USA, ACM (2008) 789–796
43. Hoare, C.A.R., et al.: Communicating sequential processes. Volume 178. Prentice-hall Englewood Cliffs (1985)
44. Abadi, M., Gordon, A.D.: A calculus for cryptographic protocols: The spi calculus. In: Proceedings of the 4th ACM conference on Computer and communications security, ACM (1997) 36–47
45. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on, IEEE (2001) 136–145
46. Nowak, D.: A framework for game-based security proofs. In: International Conference on Information and Communications Security, Springer (2007) 319–333
47. Diffie, W., Hellman, M.E.: New directions in cryptography. Information Theory, IEEE Transactions on **22**(6) (1976) 644–654
48. Boneh, D.: The decision diffie-hellman problem. In: Algorithmic number theory. Springer (1998) 48–63
49. Koblitz, N.: Elliptic curve cryptosystems. Mathematics of computation **48**(177) (1987) 203–209
50. Li, H., Chong, E.K.: On connections between group homomorphisms and the ingleton inequality. In: 2007 IEEE International Symposium on Information Theory, IEEE (2007) 1996–1999
51. Hoheisel, G.: Primzahlprobleme in der analysis... Walter de Gruyter. (1930)
52. Baker, R.C., Harman, G.: The difference between consecutive primes. Proceedings of the London Mathematical Society **3**(2) (1996) 261–280
53. Baker, R.C., Harman, G., Pintz, J.: The difference between consecutive primes, ii. Proceedings of the London Mathematical Society **83**(3) (2001) 532–562