Comparision queries on encrypted data

Alexandru Ionita

May 17, 2018

Abstract

The abstract text goes here.

1 Introduction

The scheme definition:

$setup_{AI}(n = log(Max\_index))$: It generates $P = p_1 * p_2 * ... * p_n$, where p_1, p_2 ... p_n are random primes.
It creates than a multilinear group G of composite order P.(Having the subgroups G_{p_1}, G_{p_2} ... G_{p_n}), and a random generator $g \in G$.
For each index x, we generate the following: a secret array a: a_1, a_2 ... a_n, some random exponents r_1, r_2, ... $r_n \in \mathbb{Z}_P$ and r'_1, r'_2, ... $r'_n \in \mathbb{Z}_P$, the keys k_1, k_2, .. $k_n \in \mathbb{Z}_P \times \mathbb{Z}_P$ as follows:

$$k_i = \begin{cases} (r_i p_i,\ r_i p_i p_{i+1} ... p_n) & x = 1 \\ (r'_i,\ r_i p_i) & x = 0 \end{cases}$$

This table may help in viewing the values of the keys.

	$b_i = 1$	$b_i = 0$
$x_i = 1$	$r_i p_i$	$r_i p_i p_{i+1} ... p_n$
$x_i = 0$	r_i	$r_i p_i$

$encrypt_{AI}(M, b)$: We want to encrypt element M based on the index b.
We will compute $C = M * e(g, g..., g)^{a_{1,b} a_{2,b} .. a_{n,b}}$, where $a_{i,b}$ is the value a_i for index b. Compute $D_i = e(g^{k_1}, g^{k_2}, ..., g^{k_{i-1}}, g^{k_i} g^{a_{i,b}}, ..., g^{k_n})$.

$decrypt_{AI}(k, (C, D))$: First compute $B_i = e(g^{k_1}, g^{k_2}, ..g^{k_{i-1}}, g, g^{k_{i+1}}, ..., g^{k_n})$.
To decrypt, we need to compute $C * \frac{D_1 D_2 ... D_n}{B_1 B_2 ... B_n}$