3.3.2.1. 对多项式进行加密

将多项式系数进行加密即可。

$Enc_{pk}(f(x)) = (Enc_{pk}(f[0]), Enc_{pk}(f[1]), \cdots, Enc_{pk}(f[deg(f)]))$

3.3.2.2. 多项式求值

给定一个加密后的多项式$Enc_{pk}(f(x))$, 取$x = a$, 求$Enc_{pk}(f(a))$的值。

$Enc_{pk}(f(a))$

$= Enc_{pk}(\sum_{i=0}^{deg(f)} f[i] \cdot a^i)$

$= \Pi_{i=0}^{deg(f)} Enc_{pk}(f[i] \cdot a^i)$

$= \Pi_{i=0}^{deg(f)} [Enc_{pk}(f[i])]^{a^i}$

3.3.2.3. 多项式加法

给定两个加密后的多项式$Enc_{pk}(f(x))$和$Enc_{pk}(g(x))$，计算$Enc_{pk}(h(x)) = Enc_{pk}(f(x) + g(x))$.

考虑$h(x)$第i项系数,

$Enc_{pk}(h[i]) = Enc_{pk}(f[i] + g[i]) = Enc_{pk}(f[i]) \cdot Enc_{pk}(g[i])$

3.3.2.4. 多项式乘法

给一个未加密的多项式$g(x)$，以及加密后的多项式$Enc_{pk}(f(x))$，计算g, f相乘后的密文，即计算$Enc_{pk}(h(x)) = Enc_{pk}(f(x) \cdot g(x))$

考虑$h(x)$的第i项系数

$Enc_{pk}(h[i])$

$= Enc_{pk}(g[0] \cdot f[i] + g[1] \cdot f[i-1] + g[2] \cdot f[i-2] + \cdots + g[i] \cdot f[0])$

$= Enc_{pk}(g[0] \cdot f[i]) \cdot Enc_{pk}(g[1] \cdot f[i-1]) \cdot Enc_{pk}(g[2] \cdot f[i-2]) \cdots Enc_{pk}(g[i] \cdot f[0])$

$= (Enc_{pk}(f[i]))^{g[0]} \cdot (Enc_{pk}(f[i-1]))^{g[1]} \cdot (Enc_{pk}(f[i-2])^{g[2]} \cdots (Enc_{pk}(f[0]))^{g[i]}$

3.3.3. 用多项式来表示一个集合

用户i的集合记为$\{(S_i)_j\}_{1 \le j \le k}$, $(S_i)_j$表示集合S_i的第j个元素。则可构造其多项式为$f_i(x) = \Pi_{1 \le j \le k}(x - (S_i)_j)$, 即集合$S_i$中的元素为多项式$f_i$的根。

令集合S的多项式为f, 集合T的多项式为g, 则:

1. 并集$S \cup T$对应的多项式为$f \cdot g$.
2. 交集$S \cap T$对应的多项式为$f \cdot r + g \cdot s$，其中r, s是次数不大于k次的随机多项式，即$r, s \leftarrow R^k[x]$, 当R足够大，该多项式的根有非常大的概率为$S \cap T$中的元素。

4. 协议

输入：有$n \geq 2$个用户，其中有$c < n$个可能的共谋者，每个人的私有输入为S_i，$|S_i| = k$。用秘密共享方案来分享秘密sk, (sk, pk)为Pallier加密方案的私钥和公钥。（未解决问题：如何在无中心节点情况下实施秘密共享方案？）

输出：交集$S_1 \cap S_2 \cap \cdots \cap S_n$.

1. 对于用户$i = 1, 2, 3, .., n$:
   (a). 计算集合S_i对应的多项式$f_i(x) = (x - (S_i)_1)(x - (S_i)_2)\cdots(x - (S_i)_k)$.
   (b). 将加密后的多项式$Enc_{pk}(f_i(x))$发送给用户$i + 1, \cdots, i + c$.
   (c). 随机选择$c + 1$个多项式$r_{i,0}. r_{i,1}, \cdots, r_{i,c} \leftarrow R^k[x]$.
   (d). 计算加密后的多项式
   $Enc_{pk}(\phi_i) = Enc_{pk}(r_{i,c} \times f_{i-c} + \cdots + r_{i,1} \times f_{i-1} + r_{i,0} \times f_i)$
   $= r_{i,c} \times_h Enc_{pk}(f_{i-c}) +_h \cdots +_h r_{i,1} \times_h Enc_{pk}(f_{i-1}) +_h r_{i,0} \times_h Enc_{pk}(f_i)$.
   （再次注意$+_h, \times_h$这两个运算是3.3.2节中的运算！）
2. 用户1将加密后的多项式$Enc_{pk}(\lambda_1) = Enc_{pk}(\phi_1)$给用户2。
3. 对于用户$i = 2, 3, \cdots, n$:
   (a). 从用户$i - 1$处获得加密多项式$Enc_{pk}(\lambda_{i-1})$.
   (b). 计算加密多项式$Enc_{pk}(\lambda_i) = Enc_{pk}(\lambda_{i-1} + \phi_i) = Enc_{pk}(\lambda_{i-1}) +_h Enc_{pk}(\phi_i)$.
   （再再次注意$+_h$符号！！）
   (c). 将$Enc_{pk}(\lambda_i)$交给第$i + 1 \ mod \ n$ 个用户。
4. 用户1将自己收到的加密多项式给其他所有人，用户1从用户n手中获得
   $Enc_{pk}(p) = Enc_{pk}(\lambda_n) = Enc_{pk}(\sum_{i=1}^{n} \phi_i)$
   $= Enc_{pk}(\sum_{i=1}^{n}(\sum_{j=0}^{c}(r_{i,j} \times f_{i-j})))$
   $= Enc_{pk}(\sum_{i=1}^{n} f_i \times (\sum_{j=0}^{c} r_{i+j,j}))$
   所以p是$f_1, f_2, \cdots, f_n$的线性组合，即p是$S_1 \cap S_2 \cap \cdots \cap S_n$对应的多项式。所有用户都得到多项式$p$.
5. 所有用户用秘密共享方案还原出sk对$Enc_{pk}(p)$进行解密得到p.
6. 对于每个用户i, 检查集合中每个元素$(S_i)_j$是否为p的根，即$p((S_i)_j) = 0$是否成立，若是，则该元素为交集中元素。检查完所有元素后得到交集。