私有交集求和协议实现

**概述**

本文档描述了基于密码学的私有交集求和协议(DDH-based Private Intersection-Sum Protocol)的Python实现。该实现允许两方(P1和P2)在不泄露各自私有数据的前提下，计算它们的标识符集合的交集大小，以及交集中与P2数据关联的整数值之和。

协议原理

1. 使用椭圆曲线密码学(ECC)实现标识符盲化

2. 使用Paillier同态加密系统进行安全数值计算

3. 通过双重指数盲化和集合重排实现隐私保护

API参考

`H(s: str) -> ecdsa.ellipticcurve.Point`

功能：将任意字符串映射到椭圆曲线上的点

参数：

- `s`：输入的字符串标识符

返回：椭圆曲线点 (x,y)

P1步骤

功能：P1初始化协议并生成第一轮消息

参数：

- `V`：P1的标识符集合（字符串列表）

返回：

- 元组`(pk, blinded\_hashes)`：

- `pk`：Paillier公钥

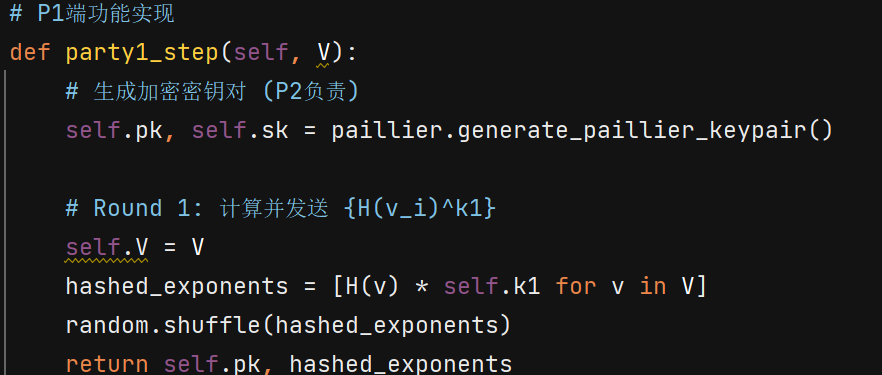
- `blinded\_hashes`：P1盲化后的哈希点列表 {H(v\_i)^k1}

说明：

- 生成Paillier密钥对

- 使用随机指数k1盲化所有标识符

- 重排返回列表顺序以增强隐私



P2步骤

功能：P2处理第一轮消息并生成第二轮消息

参数：

- `pk`：Paillier公钥

- `received\_from\_p1`：P1发送的盲化哈希点集合

- `W`：P2的数据集，元组列表 [(标识符, 整数值)]

返回：

- 元组`(double\_blinded, encrypted\_tuples)`：

- `double\_blinded`：双重盲化点列表 {H(v\_i)^k1k2}

- `encrypted\_tuples`：元组列表 (H(w\_j)^k2, Enc(t\_j))

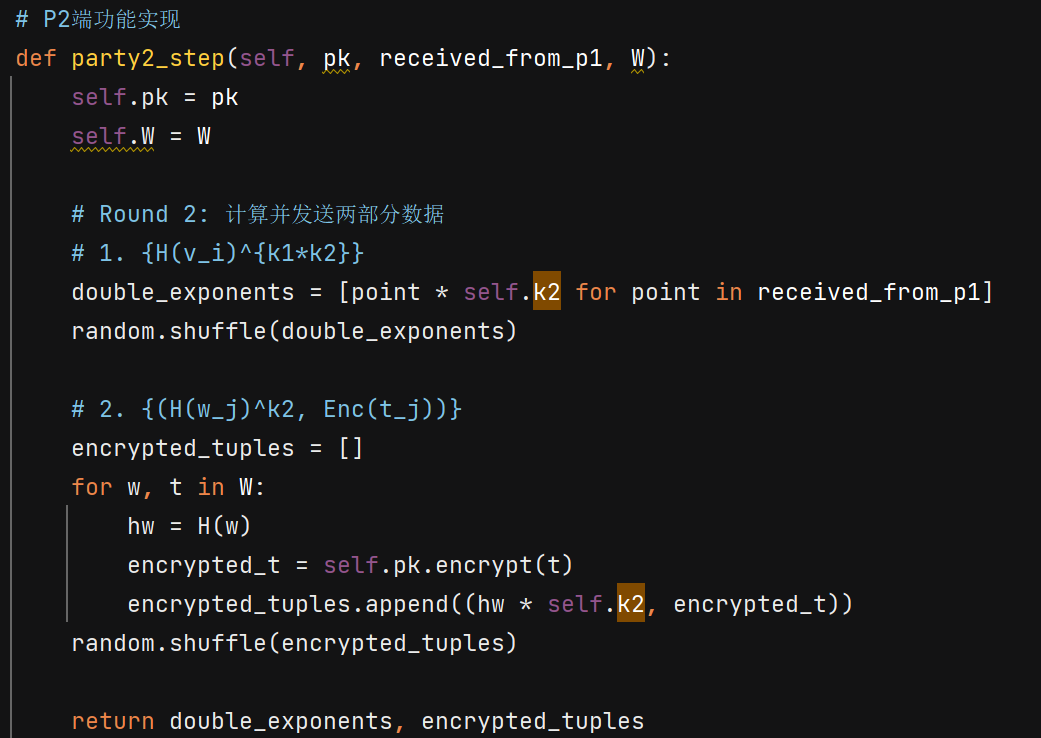
说明：

- 生成P2的随机指数k2

- 双重盲化P1的点集

- 盲化P2的标识符并加密关联值

- 重排所有输出列表顺序



P1最终计算

功能：P1完成交集检测和加密求和

参数：

- `double\_exponents`：P2返回的双重盲化点集

- `encrypted\_tuples`：P2返回的盲化标识符和加密值元组

返回：

- 元组`(count, encrypted\_sum)`：

- `count`：交集大小

- `encrypted\_sum`：交集中值的加密和(已重随机化)

P2解密结果

功能：P2解密最终结果

参数：

- `encrypted\_sum`：P1返回的加密求和值

返回：

- `int`：交集中值的实际和

使用示例



安全特性

1. 标识符隐私：

- 所有标识符以H(·)^k形式传输

- 双重盲化H(v\_i)^k1k2有效隐藏原始标识符

- 集合传输前重排顺序防止位置关联

2. 数值隐私：

- 关联值在传输过程中始终处于加密状态

- Paillier加密保证数值机密性

- 最终结果重随机化防止泄露中间状态

3. 协议安全：

- 基于椭圆曲线DDH假设

- Paillier提供语义安全性

- 新鲜随机指数确保每次执行独立

性能考虑

1. 计算密集操作：

- 椭圆曲线点标量乘法

- Paillier加密/解密操作

2. 优化建议：

- 预计算常用标识符的H(·)

- 使用更快的椭圆曲线(如Curve25519)

- 对于大数据集，使用批处理优化

3. 网络开销：

- 点集和加密值会导致较大消息体

- 可使用点压缩减小传输量