联　　盟　　规　　范

PPCA 9 — 2023

隐私计算 跨平台互联互通 开放协议 第 1 部分：ECDH-PSI

Privacy-preserving computation cross-platform interconnection—

Open protocol Part1: ECDH-PSI

2023-07-26 发布　　2023-07-26 实施

隐私计算联盟 发 布

目　次

前　言

本文件是《隐私计算 跨平台互联互通》系列文件之一，该系列文件名称如下：

——开放协议 第1部分：ECDH-PSI；

——开放协议 第2部分：SS-LR；

——开放协议 第3部分：PHE-FLR。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本文件由隐私计算联盟提出并归口。

本文件起草单位：中国信息通信研究院、蚂蚁科技集团股份有限公司、中移动信息技术有限公司、天翼电子商务有限公司、中国工商银行股份有限公司、联通数字科技有限公司、深圳市洞见智慧科技有限公司、华控清交信息科技（北京）有限公司、深圳前海微众银行股份有限公司。

本文件主要起草人：陆宇飞、陈卓、贺伟、孙林、夏知渊、靳新、白玉真、黄熹之、昌文婷、邵健、赵原、袁鹏程、余超凡、郭相林、章庆、茹志强、毕剑锋、魏博言、袁博、王思源、张鸣皓、赵永坤、何浩、姚明、王磊、彭晋、张晓蒙、李漓春、殷山、张启超、董佳佳、苏亮、徐文静、王朝阳、马晨、徐长通。

引　言

当前多方安全计算、联邦学习等隐私计算技术快速发展，越来越多的产品从试点部署阶段转入落地应用，市场竞争火热。但是，不同技术厂商提供的产品和解决方案在设计原理和功能实现之间存在较大差异，使得部署于不同平台的隐私计算参与方之间无法跨平台完成同一计算任务，为实现与部署于不同平台的多个合作方之间的数据融合，用户往往不得不部署多套产品以逐一适配。作为促进跨机构间数据共享融合的关键技术，隐私计算有望成为支撑数据流通产业的基础设施，但高额的应用成本不利于隐私计算技术的推广应用。因此，解决不同产品之间的技术壁垒，实现计算任务在跨平台间的互联互通已成为产业内的迫切需求。

版权声明

邮箱：ppca@caictyds.cn

隐私计算 跨平台互联互通
开放协议 第1部分：ECDH-PSI

1 范围

本文件规定了异构隐私计算平台进行跨平台的ECDH-PSI的互联互通的算法协议和传输层实现参考。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 25069-2022 信息安全技术 术语

GB/T 32918-2017 信息安全技术 SM2 椭圆曲线公钥密码算法

3 术语和定义

GB/T 25069-2022界定的以及下列术语和定义适用于本文件。

3.1

隐私计算 privacy-preserving computation

在保证数据提供方不泄露原始数据的前提下，对数据进行分析计算的一系列信息技术，保障数据在流通与融合过程中的“可用不可见”。

3.2

开放协议 open protocol

通过定义算法执行流程中交互信息，各平台独立开发算法来实现平台间算法的互通。

3.3

节点 node

隐私计算生态中的抽象功能单元，用来指代由机构或组织部署的隐私计算平台。

3.4

算法 algorithm

为解决问题严格定义的有限的有序规则集。

[来源：GB/T 25069—2022, 3.581]

3.5

组件 component

独立执行隐私计算任务的模块单元，其经过封装、符合开放接口规范、可以完成某个特定计算或算法，可独立部署。

3.6

任务 task

组件运行实例的载体。

4 缩略语

下列缩略语适用于本文件。

ECDH：椭圆曲线迪菲-赫尔曼（Elliptic Curve Diffie–Hellman）

PSI：隐私集合求交（Private Set Intersection）

5 算法协议

5.1 ECDH-PSI 算法概述

ECDH-PSI的算法流程如图1所示，包括5个步骤：

假设有参与方A和参与方B：

第一步：参与方在本地计算原始数据（如a_i）的杂凑值，并将杂凑值映射为椭圆曲线上的点，然后加密[注1]得到加密后的数据（如$P1_i$）；

第二步：每个参与方将加密后的数据传输给其它参与方，如参与方A将$P1_i$传输给参与方B；

第三步：每个参与方在本地使用自己的私钥对步骤二中接收到的数据进行二次加密[注1]；

第四步：如果结果对另一个参与方可见，将步骤三中加密后的数据传输给另外一个参与方；

第五步：拿结果的参与方基于步骤三和步骤四的两方的二次密文在本地计算集合求交的结果。

注1：加密指基于椭圆曲线的点乘算法和本地的密钥（如key_A），对数据完成加密。

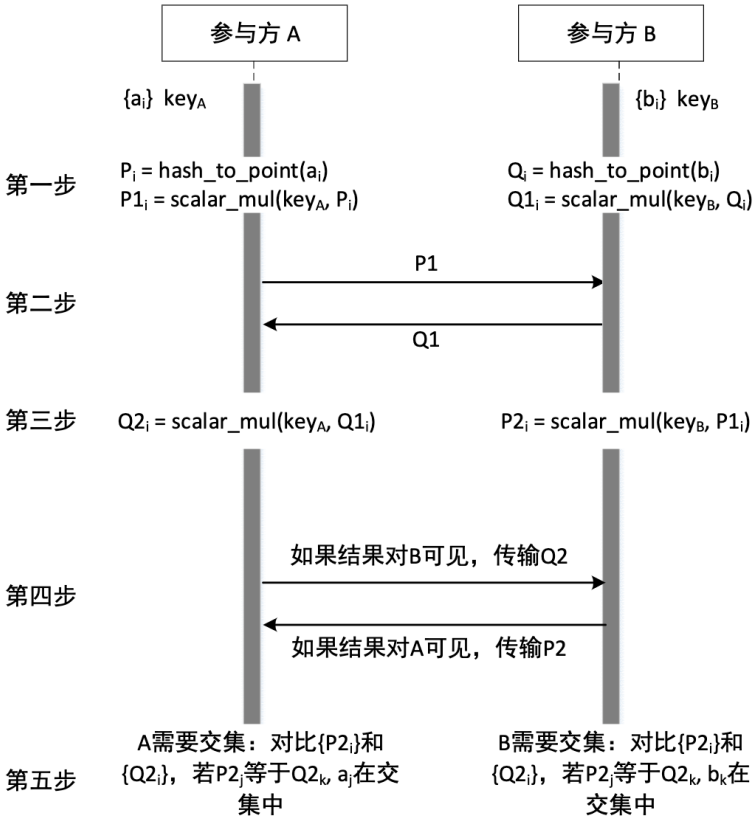

图1 ECDH-PSI算法

5.2 算法流程

算法流程包含两个阶段，第一阶段为算法协商握手阶段，第二阶段为算法主体运行阶段：

a) 算法协商握手阶段，确定算法版本、PSI算法类型、PSI算法参数、待求交集合的大小等算法运行所需的信息；

b) 算法主体运行阶段，实现隐私集合求交。

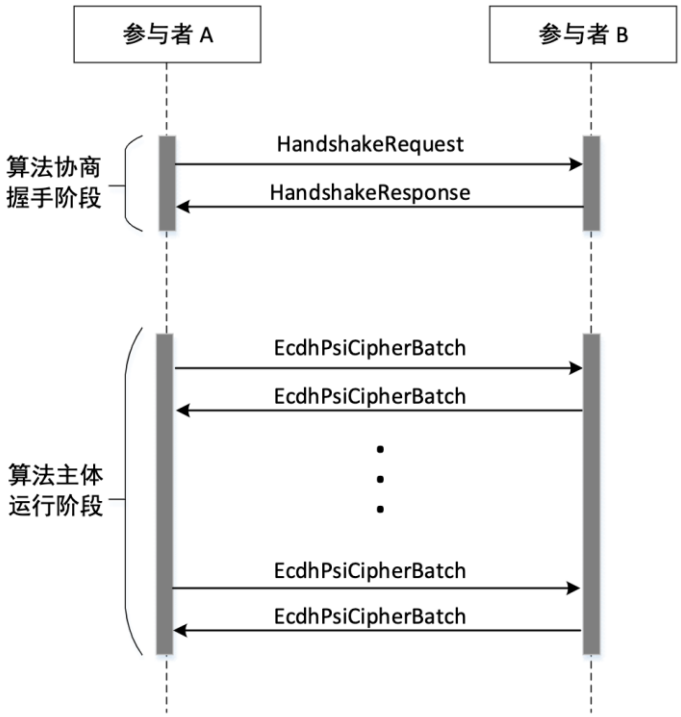

图2 ECDH-PSI协议流程

6 协议配置

6.1 安全参数

计算安全强度应至少128比特级别。

6.2 密码算法

6.2.1 椭圆曲线

椭圆曲线应选择安全强度达到计算安全等级的椭圆曲线，至少包括SM2和Curve25519。

6.2.2 映射到椭圆曲线

映射到椭圆曲线指将任意长度的字节数组映射到椭圆曲线上的一个点。针对不同的椭圆曲线，有高性能和高安全的映射方案。

高性能映射到椭圆曲线方案指计算速度快的映射方案，包括：

a) SM2等曲线的高性能映射方案Try-and-Increment[5]；

b) Curve25519等曲线的高性能映射方案Direct-Hash，即哈希值直接作为X坐标。

高安全映射到椭圆曲线方案指均匀的抗侧信道攻击的映射方案，遵循IETF Hashing to Elliptic Curves[4]，包括：

a) SM2等曲线的高安全映射方案：Simplified Shallue-van de Woestijne-Ulas method[4]（6.6.2节）；

b) Curve25519等曲线的高安全映射方案: Elligator 2 method[4]（6.7.1节）。

6.2.3 哈希算法

哈希算法用于实现映射到椭圆曲线的方案，应选择安全强度达到计算安全强度的哈希算法，至少包括SM3和SHA256。

6.3 协议参数

6.3.1 算法套件配置

算法套件定义了<Curve, Hash, HashToCurveStrategy>的三元组：

a) Curve表示所选的椭圆曲线；
b) Hash表示所选的映射到椭圆曲线方案的哈希算法；
c) HashToCurveStrategy表示所选椭圆曲线的映射到椭圆曲线方案，即该椭圆曲线的高性能映射到椭圆曲线方案或高安全映射到椭圆曲线方案。

算法套件举例：

a) <SM2,SM3,Try-and-Increment> 高性能映射到椭圆曲线方案的国密算法套件；
b) <SM2,SM3,Simplified Shallue-van de Woestijne-Ulas method[4]> 高安全映射到椭圆曲线方案的国密算法套件；
c) <Curve25519,SHA256,Direct-Hash> 高性能映射到椭圆曲线方案的国际算法套件；
d) <Curve25519,SHA256,Elligator 2 method[4] > 高安全映射到椭圆曲线方案的国际算法套件。

以上4个组件中，2个高性能算法套件<SM2,SM3,Try-and-Increment>,<Curve25519, SHA256, Direct-Hash> 是至少要必选支持的。

6.3.2 一次密文点表示配置

一次密文点表示规定了一次点乘后生成的密文在椭圆曲线上的表示形式。双方在互联互通之前需要对齐密文点表示的配置。

对一次密文的椭圆曲线点表示的配置如下：

a) 对于双坐标(X,Y)表示椭圆曲线点计算点乘的曲线，如SM2，遵循GB/T 32918：
   - 非压缩表示配置为：0x04||X||Y[注2]；
   - 压缩表示配置为：若Y的最低比特是0，0x02||X；若Y的最低比特是1，0x03||X。
b) 单X坐标表示椭圆曲线点可计算点乘的曲线，如Curve25519等，点表示配置为X坐标。

注2：||表示字符串连接。

6.3.3 二次密文截断配置

假阳等级：30比特。即PSI交集中存在至少一个ID假阳（假命中）的概率不超过$\frac{1}{2^{30}}$。对二次密文的椭圆曲线点可截断，截断比特长度要满足假阳等级的要求，即：

截断后密文的比特长度 $>= \lceil log_2(A方样本数) \rceil + \lceil log_2(B方样本数) \rceil +$ 假阳等级

对二次密文椭圆曲线点的X坐标从低位到高位按截断比特长度截取比特用于求交，Y坐标舍弃。

如果不使用6.3.3的密文截断配置，二次密文可使用6.3.2所示的密文点表示配置。

7 算法协商握手

7.1 HandshakeRequest 消息

7.1.1 HandshakeRequest 数据结构

HandshakeRequest包括算法协商握手请求的基本信息，其数据结构如表1所示。

表 1 HandshakeRequest 数据结构

属性名称	数据类型	数据说明	示例	数据备注
version	int32	握手请求版本号	2	必选
requester_rank	int32	发送方 rank 值	1	必选
supported_algos	int32 list	支持的 PSI 算法的 enum 值，如 ecdh-psi 算法	[1]	必选
protocol_families	int32 list	支持的协议族的 enum 值，如 ECC 协议族	[1]	必选
protocol_family_params	google.protobuf.Any list	相应的协议族详细握手参数，与 protocol_family 对应。 实际类型随协议族类型而变，ECC 协议族的类型是 EccProtocolProposal	见表 2	必选
io_param	google.protobuf.Any	PSI 算法的输入和结果输出格式参数，与 algo 对应，其数据结构如表 7 所示。 实际类型随算法类型而变，ECDH-PSI 协议的类型是 PsiDataInfoProposal	见表 7	必选

其中：

- EccProtocolProposal包括ECDH-PSI算法参数协商的基本信息，如表2：

表 2 EccProtocolProposal 数据结构

属性名称	数据类型	数据说明	示例	数据备注
supported_versions	int32 list	支持的版本列表	[1]	必选
ec_suits	tuple <int32,int32,int32> list	算法套件编号推荐列表，每个元素是一个 tuple: <Curve 编号, Hash 编号, HashToCurveStrategy 编号>。算法套件按推荐优先级进行先后顺序。	Curve 编号见表 3 Hash 编号见表 4 HashToCurveStrategy 编号见表 5	必选
point_octet_formats	int32 list	支持的点的序列化格式的 enum 值，如国密 SM2 标准 GB/T 32918 压缩、不压缩，按顺序从先向后推荐	见表 6	必选
support_point_truncation	bool	协商要不要启用二次密文截断。true 表示 HandshakeRequest 发送方支持截断。false 表示 HandshakeRequest 发送方不支持截断。	false	必选

- Curve取值和说明如表3表13所示：

表 3 Curve 取值说明

数值	数据说明
0	未定义
1	Curve25519
2	SM2

- Hash取值和说明如表4表13所示：

表 4 Hash 取值说明

数值	数据说明
0	未定义
1	SM3
10	SHA-224
11	SHA-256
12	SHA-384
13	SHA-512
20	SHA3-224
21	SHA3-256
22	SHA3-384
23	SHA3-512
30	SHAKE-256

- HashToCurveStrategy取值和说明如表5所示：

表 5 HashToCurveStrategy 取值说明

数值	数据说明
0	未定义
1	TRY_AND_INCREMENT
2	TRY_AND_REHASH
3	DIRECT_HASH
10	IRTF_SSWU_RO
11	IRTF_SSWU_NU
12	IRTF_ELL2_RO
13	IRTF_ELL2_NU

- point_octet_format取值和说明如表6所示：

表 6 point_octet_format 取值说明

数值	数据说明
0	未定义
1	UNCOMPRESSED
2	X962_COMPRESSED
3	X962_UNCOMPRESSED
4	X962_HYBRID

- PsiDataInfoProposal包括输入输出参数协商的基本信息，如表7：

表 7 PsiDataInfoProposal 数据结构

属性名称	数据类型	数据说明	示例	数据备注
supported_versions	int32 list	支持的版本列表	[1]	必选
item_num	int64	待求交的 PSI 数据总量	10000	必选
result_to_rank	int32	确定 PSI 结果获取方，其值域表如表 8 所示。	-1	必选

- result_to_rank字段用来确定 PSI 结果获取方，其值域如表8所示。

表 8 result_to_rank 的值域表

数值	数据说明
-1	所有机构都可以拿到交集结果
rank	指定机构 rank 拿到交集结果。Rank 是参与方在传输层的编号。

7. 2 HandshakeResponse 消息

HandshakeResponse消息结构包括算法协商握手响应的基本信息，其数据结构如表9所示。

表 9 HandshakeResponse 数据结构

属性名称	数据类型	数据说明	示例	数据备注
header	ResponseHeader	握手请求响应头	见表 12	必选
algo	int32	决策下来的 PSI 算法	1	必选
protocol_families	int32 list	决策下来的协议族	[1]	必选
protocol_family_params	google.protobuf.Any list	决策下来的协议族详细参数，实际类型随协议族而变，ECC 的类型是 EccProtocolResult，见表 10	见表 10	必选

属性名称	数据类型	数据说明	示例	数据备注
io_param	google.protobuf.Any	决策下来的 PSI 算法输入输出详细参数。实际类型随 PSI 类型而变，ECDH-PSI 的类型是 PsiDataInfoResult	见表 11	必选

其中：

- EccProtocolResult包括ECDH-PSI算法参数协商的基本信息，如表10：

表 10 EccProtocolResult 数据结构

属性名称	数据类型	数据说明	示例	数据备注
version	int32	支持的版本列表	1	必选
result_to_rank	int32	确定 PSI 结果获取方，其值域表如表 8 所示。	-1	必选
ec_suit	tuple<int32,int32,int32>	算法套件，即 < Curve 编号, Hash 编号, HashToCurveStrategy 编号>	<SM2 编号, SM3 编号,Try-and-Increment[5] 编号>	必选
point_octet_format	int32	支持的点的序列化格式的 enum 值，如国密 SM2 标准 GB/T 32918 压缩、不压缩	1	必选
bit_length_after_truncated	int32	二次密文截断后比特长度。长度一般为 8 的倍数，例如 96，128，256 等。-1 表示不截断。	96 （对于 10 亿匹配 10 亿，30 比特假阳等级，推荐 96 比特截断长度）	必选

- PsiDataInfoResult包括输入输出参数协商的基本信息，如表11：

表 11 PsiDataInfoResult 数据结构

属性名称	数据类型	数据说明	示例
version	int32	版本号	1
result_to_rank	int32	确定 PSI 结果获取方，其值域表如表 8 所示。	-1

- ResponseHeader消息结构包括的基本信息如表12所示

表 12 ResponseHeader 数据结构

属性名称	数据类型	数据说明	数据备注
error_code	int32	握手响应的结果，其值域如表 13	必选

属性名称	数据类型	数据说明	数据备注
		所示。	
error_msg	string	用户自定的消息字符串	可选

- error_code的取值和说明如表13所示。

表 13 error_code 的值域表

数值	数据说明
0	成功
31100000	GENERIC_ERROR，通用错误
31100001	UNEXPECTED_ERROR，状态不符合预期错误
31100002	NETWORK_ERROR，网络通信错误
31100100	INVALID_REQUEST，非法请求
31100101	OUT_OF_RESOURCE，运行资源不满足
31100200	HANDSHAKE_REFUSED，握手拒绝
31100201	UNSUPPORTED_VERSION，不支持的版本
31100202	UNSUPPORTED_ALGO，不支持的算法
31100203	UNSUPPORTED_PARAMS，不支持的算法参数

8 算法主体运行

8.1 EcdhPsiCipherBatch 消息

EcdhPsiCipherBatch的基本信息如表14所示。

表 14 EcdhPsiCipherBatch 数据结构

属性名称	数据类型	数据说明	示例	数据备注
type	string	标识密文的类型，取值"enc"和“dual.enc”。 "enc":图 1 中算法第二步中交换的密文信息。" dual.enc ":图 1 中算法第四步中交换的密文信息	“enc”	必选
batch_index	int32	传输批次的编号。当待求交集合比较大的时候，发送方可以选择分多个批次发送密文	0	必选
is_last_batch	bool	是否为最后一个传输批次	false	必选
count	int32	当前批次包含的密文数量	1000	必选
ciphertext	bytes	当前批次包含的密文	-	必选

9 传输层实现参考

9.1 通信框架

异构隐私计算技术平台间进行互联互通时，通信框架应能满足兼容性、通用性以及安全性的要求，如表15所示。

表 15 通信框架范围

RPC框架	GRPC
编码方式	ProtoBuf

9.2 初始化通信协议

初始化通信协议在 PSI 任务开始前执行一次。

每个参与者向其它参与者通知自己的存在性，即向他人发送 connect_{self_rank}：

```
For i in 0..word_size [注3]:
  if i == self_rank:
    continue
  P2P send to rank i: {key: connect_{self_rank}, value: ""}
```

每个参与者检查他人的存在性，即依次检查 connect_{rank} 消息已经收到：

```
For i in 0..word_size:
  if i == self_rank:
    continue
  P2P receive on key connect_{i}
```

注 3：word_size 表示参与者数量

9.3 Protobuf 消息

不同隐私计算的参与者之间使用 Protobuf 协议传递信息。

```
service ReceiverService
{
    rpc Push(PushRequest) returns (PushResponse);
}
```

9.3.1 PushRequest 消息结构

PushRequest包括传输的基本信息，其数据结构如表16所示。

表 16 PushRequest 数据结构

属性名称	数据类型	数据说明	示例	数据备注
sender_rank	uint64	发送者的编号，如 0 指 rank	0	必选
key	string	消息唯一 ID，生成规则见 9.4 节	“root:P2P-0:0->1”	必选
value	bytes	消息体，ECDH-PSI 中的 protobuf 序列化成二进制 string，	需要传输的实际信	必选

属性名称	数据类型	数据说明	示例	数据备注
		然后把整个 string 放到 value 中	息	
trans_type	TransType	传输模式，如全量传输、分块传输，其值域如表 17 所示	见表 17	必选
chunk_info	ChunkInfo	消息大小,其数据结构如表 18 所示	见表 18	必选

其中：

- TransType 的值域如表 17 所示：

表 17 TransType 的值域表

数值	数据说明
MONO	全量传送模式
CHUNKED	分块传送模式

- ChunkInfo 的值域如表 18 所示。

表 18 ChunkInfo 的数据结构

属性名称	数据类型	数据说明		数据备注
message_length	uint64	数据总大小，单位是字节 Byte	1048576	必选
chunk_offset	uint64	当前分块的偏移量	0	必选

9.3.2 PushResponse 消息结构

PushRequest包括传输的基本信息，其数据结构如表19所示。

表 19 PushResponse 消息的数据结构

属性名称	数据类型	数据说明	数据备注
header	ResponseHeader	返回消息，如表 12 所示	必选

9.4 通信模式

9.4.1 信道

信道是一个逻辑概念，用于区分通信的上下文。每一个信道有一个全局唯一名称，命名规则为：\w+，即信号名称由字母、数字、下划线组成。信道的名字由通信组双方约定，在初始化阶段由用户传入。

信道唯一的作用就是影响 message key 的生成，信道名称会作为 message key 一部分，因此，不同信道中的消息一定不会有相同的 key，因此不同信道的消息在逻辑上不会混淆。

当上层算法需要多个信道时，第一个信道称为主信道，其它信道称为子信道。子信道的命名规则为：主信道名称-子信道编号。

举例：假设主信道名称为 root，则 0 号子信道名称为 root-0，1 号子信道名称为 root-1，以此类推。

9.4.2 P2P 通信

P2P 通信允许在任意两个参与者之间发送信息。P2P 通信 key 的命名规则为：{信道名称}:P2P-{计数器}:{发送者 RANK}->{接收者 RANK}，其中每一个信道、每一对 <sender, receiver> 都有一个独立的计数器。

举例，假设信道名称为 root，以下消息依次发送：

Rank 0 → 1 发送消息，key 为：root:P2P-0:0->1

Rank 1 → 0 发送消息，key 为：root:P2P-0:1->0

Rank 0 → 2 发送消息，key 为：root:P2P-0:0->2

Rank 0 → 1 发送消息，key 为：root:P2P-1:0->1

参 考 文 献

[1] NIST SP 800-57 Part1 Rev. 5 Recommendation for Key Management: Part 1 – General

[2] RFC 7748: Elliptic Curves for Security

[3] GM/T 32905 - 2016 信息安全技术 SM3 密码杂凑算法

[4] IETF Hashing to Elliptic Curves https://datatracker.ietf.org/doc/draft-irtf-cfrg-hash-to-curve/

[5] Icart T. How to hash into elliptic curves[C]//Advances in Cryptology-CRYPTO 2009: 29th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings. Springer Berlin Heidelberg, 2009: 303-316