移动社交网络中的定位隐私保护研究

摘要

本文提出两个关于移动社交网络中定位隐私保护的模型，分别为*UTS*模型和*US*模型．模型都应用*BV*同态加密算法，该算法使得用户与好友之间的定位信息保持同态性，且支持社交网络服务器能够在基于定位密文的条件下完成位置查询服务．模型的主要特点为：
(1)支持用户分享当前的定位信息给社交网络好友，但没有共享定位明文信息给社交网络服务器；(2)为用户提供可验证的查询服务，帮助用户识别查询结果的正确性和判断查询结果是否被敌手恶意篡改过；(3) 支持用户对社交网络好友进行分类，为用户提供个性化的查询服务；(4) 提供灵活化的定位查询服务，使得满足用户查询范围内的好友可以随时随地地访问用户的定位，且好友的查询过程无需要求用户时刻在线来协助其完成；(5)与*UTS* 模型不同，*US*模型不仅拥有(1–4)的特点，同时没有依赖可信第三方服务器来协助用户完成任何计算和查询服务．

基于位置服务（LBS）

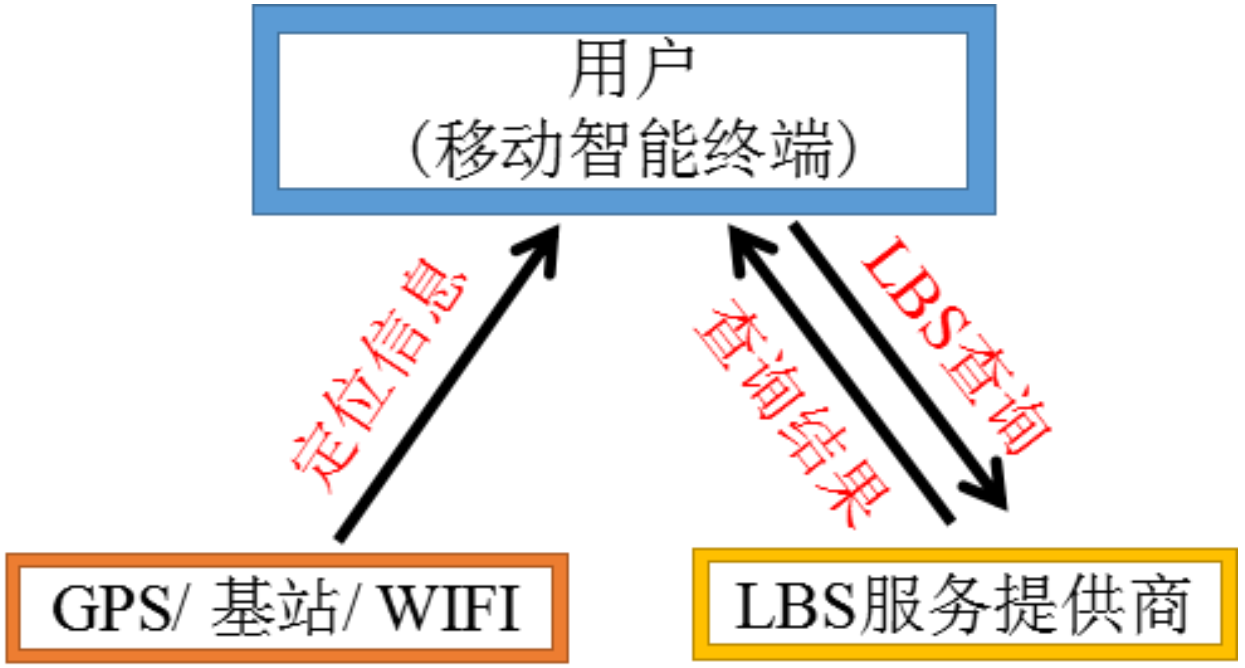
概念

在地理信息系统平台的支持下，帮助用户获得定位信息并基于该定位信息为用户提供与定位相关的服务.

一般架构

- 定位系统：如基站定位、卫星定位、WIFI定位.
- 移动智能终端（用户）：如智能手机、运动手环、平板电脑等
- 位置服务提供商：如探探、携程旅游、滴滴打车、谷歌地图等

LBS一般架构



基于位置服务（LBS）

分类

1. 社交服务(如查询好友定位)，如探探，微信，陌陌等
2. 导航与叫车服务，如百度、谷歌地图，滴滴打车等
3. 监管与追踪服务，如共享汽车，智能手环等
4. 附近娱乐美食推荐服务，如快手，大众点评，美团等

基于位置服务（LBS）

分类

1. 社交服务(如查询好友定位)，如探探，微信，陌陌等
2. 导航与叫车服务，如百度、谷歌地图，滴滴打车等
3. 监管与追踪服务，如共享汽车，智能手环等
4. 附近娱乐美食推荐服务，如快手，大众点评，美团等

查询社交网络好友的定位

无保护情况下

用户发送定位明文信息给服务器

隐私泄露问题

用户的住宅地址了解到用户的居住环境和经济条件

用户去医院的次数判断其健康状况

用户常去的游玩和娱乐地点推理出用户的个人兴趣爱好

广告推送，推测行踪，盗窃

解决方法 —— 隐私保护的定位查询方案 [1-18]

主要技术 —— k-匿名，假名，虚假定位，paillier同态加密算法，Geographic data transformation系统，基于CP-ABE算法等

主要工作

One — All

基于 [1-13] 的单次查询，用户获得所有满足条件的好友

One — One

基于[14-18]的单次查询，用户只获得一位好友的定位信息

Case

在酒吧且拥有大量好友，想查询附近有哪些大学同学在，而不是所有的好友，if用以上两种方式，则需对结果筛选/查询次数多.

工作 1

提出 One — Somewhat 的查询方式

方法

对好友进行分类，但无发送定位明文信息和真实社交关系给SNS；用户可根据好友的数量进行选择

主要工作

文[1-2] 等需要假设移动社交网络服务器不会与定位服务器勾结，而且敌手不能同时控制两个服务器， 但实际上一个敌手可控制服务器中的任意一个，进而篡改查询结果，而且多个敌手之间也可互相勾结.

工作 2

为用户提供可验证的定位查询服务，帮助用户验证查询结果

当方案运用同态加密算法时，有的需要TTP帮助用户初始化方案和运行算法，无TTP时，需要查询者与被查询者同时在线互相协助来完成查询过程(尤其是运用Paillier同态算法时).

工作 3

提出US模型，无TTP，但用户可以随时随地且独立地完成查询服务.

模型的基本定义

身份 $U_A^* = E_{Hash}(U_A).$

定位 $\boldsymbol{L_A^* = E_{BV}(L_A)}.$

阈值距离 $|\boldsymbol{td_A^*}| = E_{BV}(|\boldsymbol{td_A}|)$

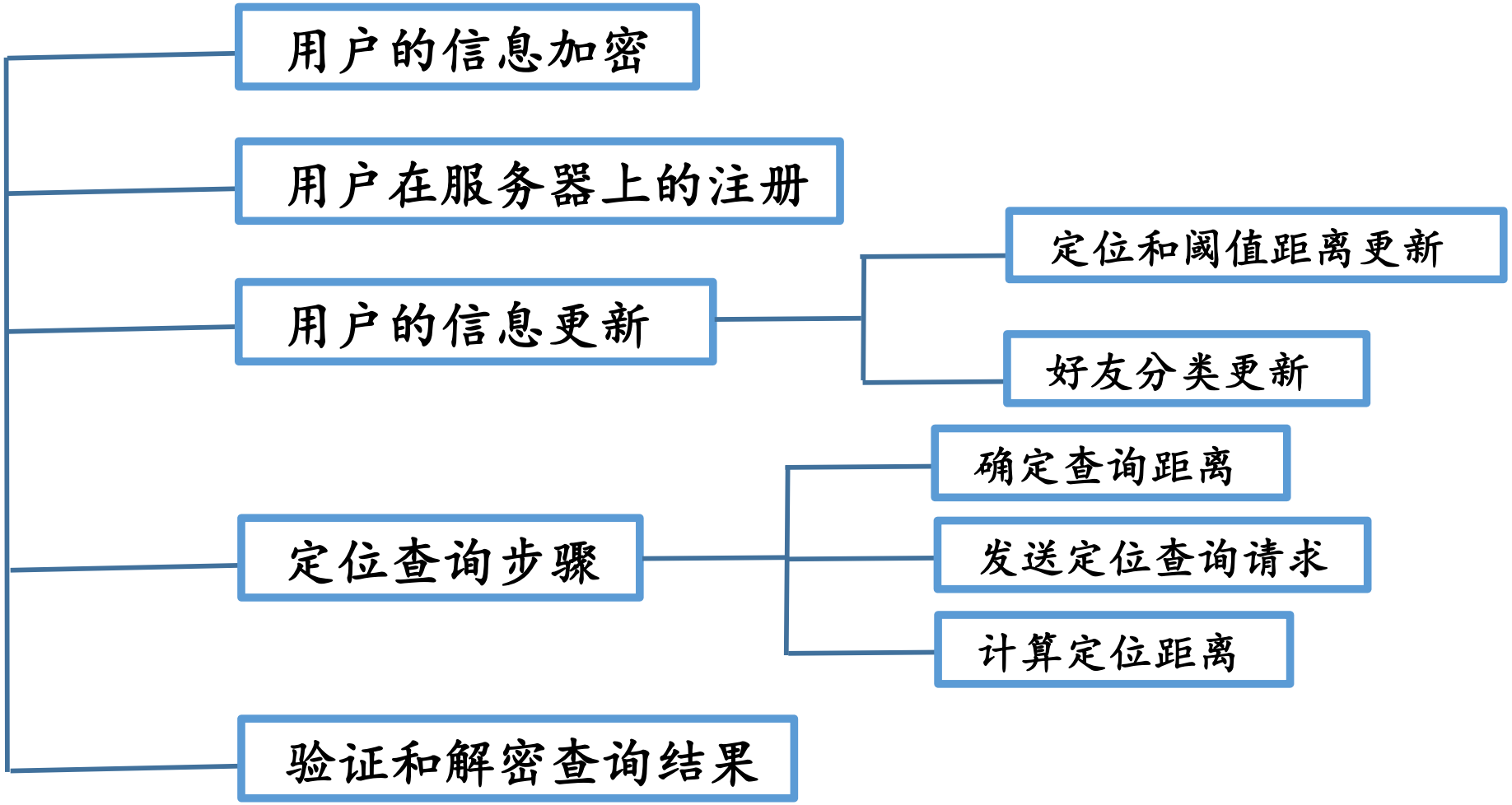
查询距离 $|\boldsymbol{qd_A^*}| = E_{BV}(|\boldsymbol{qd_A}|),$

好友分类 $m_A = C_1 \| \ldots \| C_t \ , \ C_i = (f_1, f_2, \ldots, f_{k_i}).$

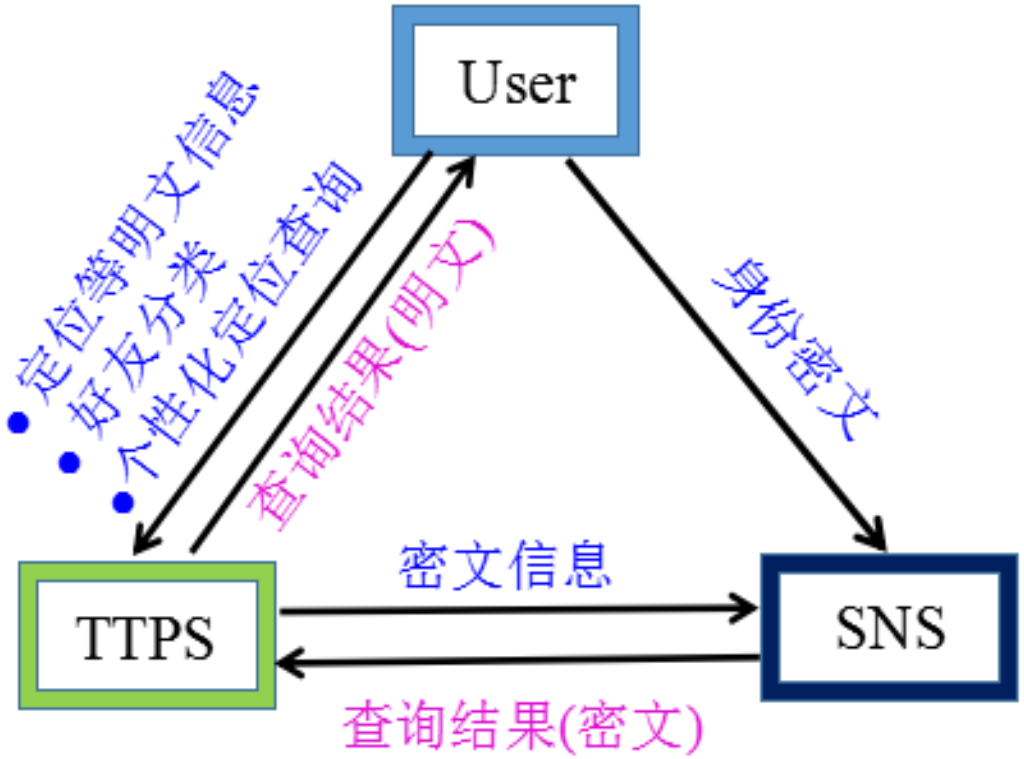
查询区域 敏感区域和非敏感区域

加密 BV同态加密算法

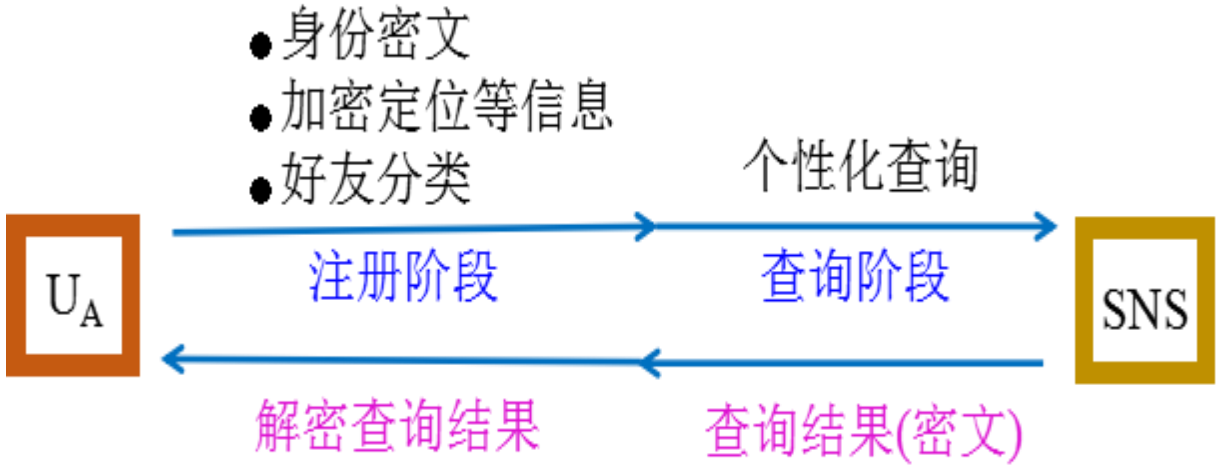
模型的运行框架


用户的信息加密
用户在服务器上的注册
用户的信息更新
定位和阈值距离更新
好友分类更新
定位查询步骤
确定查询距离
发送定位查询请求
计算定位距离
验证和解密查询结果

UTS 模型



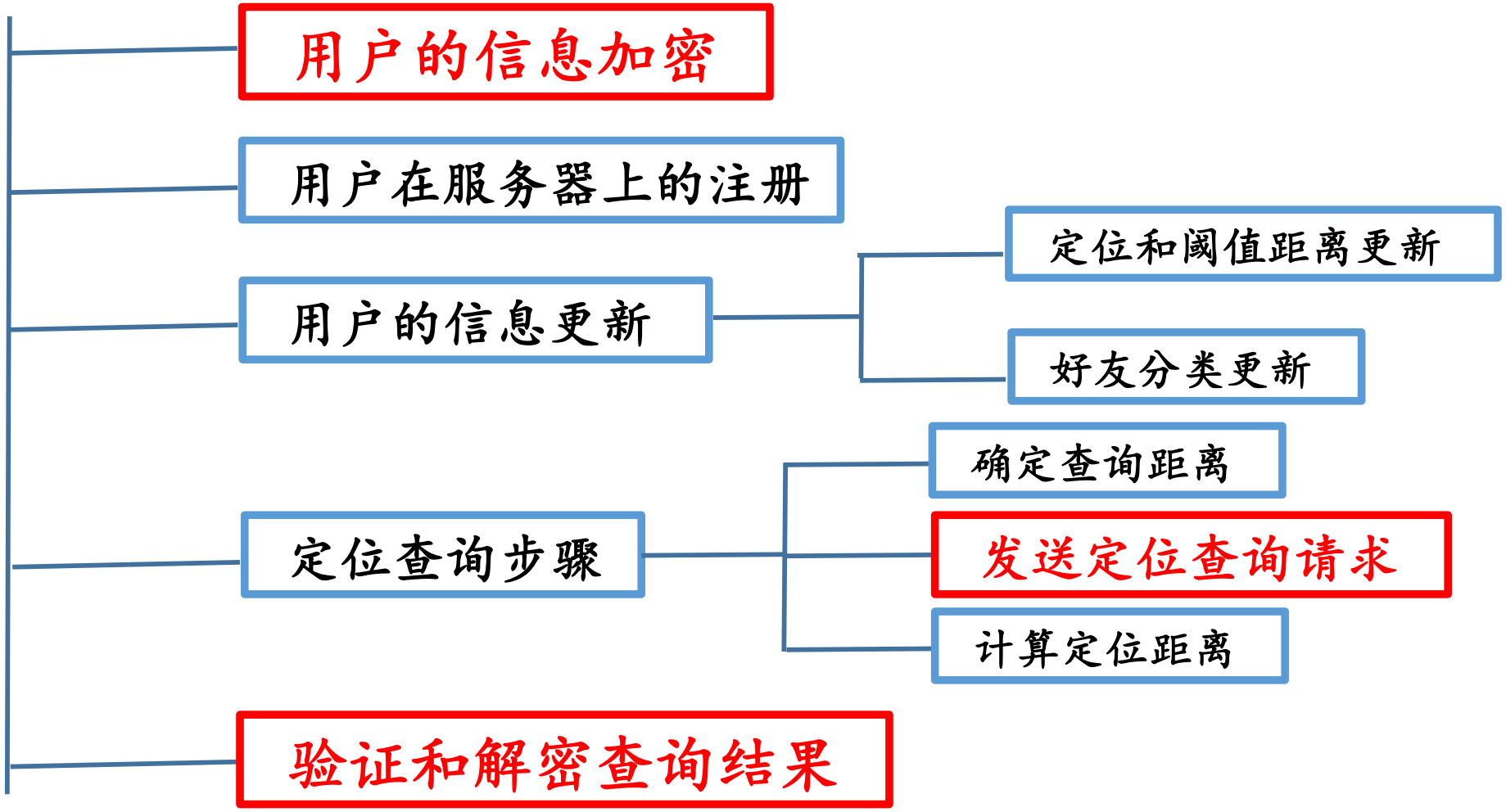
US 模型



TTPS： Trust Third Party Server

SNS： Social Network Server

模型的运行框架


用户的信息加密
用户在服务器上的注册
用户的信息更新
定位和阈值距离更新
好友分类更新
定位查询步骤
确定查询距离
发送定位查询请求
计算定位距离
验证和解密查询结果

用户的信息加密——UTS 模型——由TTPS完成

$$\mathbf{M_T} = \begin{pmatrix} U_1^* & U_2^* & U_A^* & \dots & U_{n-1}^* & U_n^* \\ |td_1| & |td_2| & |td_A| & \dots & |td_{n-1}| & |td_n| \\ |td_1^*| & |td_2^*| & |td_A^*| & \dots & |td_{n-1}^*| & |td_n^*| \\ L_1 & L_2 & L_A & \dots & L_{n-1} & L_n \\ L_1^* & L_2^* & L_A^* & \dots & L_{n-1}^* & L_n^* \\ |qd_1| & |qd_2| & |qd_A| & \dots & |qd_{n-1}| & |qd_n| \\ |qd_1^*| & |qd_2^*| & |qd_A^*| & \dots & |qd_{n-1}^*| & |qd_n^*| \end{pmatrix}.$$

—— 用户的身份密文

—— 用户的阈值距离明文/密文

—— 用户的定位明文/密文

—— 用户的查询距离明文/密文

UTS

SNS

储存和计算

$$\mathbf{M_S} = \begin{pmatrix} U_1^* & U_2^* & U_A^* & \dots & U_{n-1}^* & U_n^* \\ |\boldsymbol{td}_1^*| & |\boldsymbol{td}_2^*| & |\boldsymbol{td}_A^*| & \dots & |\boldsymbol{td}_{n-1}^*| & |\boldsymbol{td}_n^*| \\ \boldsymbol{L}_1^* & \boldsymbol{L}_2^* & \boldsymbol{L}_A^* & \dots & \boldsymbol{L}_{n-1}^* & \boldsymbol{L}_n^* \end{pmatrix}$$ ——储存

矩阵M_S 由SNS 创建, M_S 中的数据由$TTPS$ 发送给SNS.

根据$U_A^*, U_{11}^*, U_{12}^*, \dots, U_{1k}^*$,

SNS 从矩阵M_S 中提取一个$3 \times (1k+1)$ 的子矩阵,

$$\mathbf{M_C} = \begin{pmatrix} U_A^* & U_{11}^* & U_{12}^* & \dots & U_{1(k-1)}^* & U_{1k}^* \\ |\boldsymbol{td}_A^*| & |\boldsymbol{td}_{11}^*| & |\boldsymbol{td}_{12}^*| & \dots & |\boldsymbol{td}_{1(k-1)}^*| & |\boldsymbol{td}_{1k}^*| \\ \boldsymbol{L}_A^* & \boldsymbol{L}_{11}^* & \boldsymbol{L}_{12}^* & \dots & \boldsymbol{L}_{1(k-1)}^* & \boldsymbol{L}_{1k}^* \end{pmatrix}$$ ——计算

用户的信息加密——US 模型——由用户完成

定位加密.

$$\boldsymbol{L}_{\boldsymbol{A}}^{*} = E_{pk_B^A}(\boldsymbol{L}_{\boldsymbol{A}}) = (E_{pk_B^A}(x_A), E_{pk_B^A}(y_A)).$$

$\boldsymbol{L}_{\boldsymbol{A}|\boldsymbol{A}_i}^{*}$: $Alice$ 用$U_{A_i}^{*}$ 的$(u, f, g)_{A_i}$和$pk_B^{A_i}$ 加密$\boldsymbol{L}_{\boldsymbol{A}}$

$\boldsymbol{L}_{\boldsymbol{A}_i|\boldsymbol{A}}^{*}$: $U_{A_i}^{*}$ 用$Alice$ 的$(u, f, g)_A$和pk_B^A 加密$\boldsymbol{L}_{\boldsymbol{A}_i}$

} $\boldsymbol{L}_{\boldsymbol{A}}^{*}$ 与$\boldsymbol{L}_{\boldsymbol{A}|\boldsymbol{A}_i}^{*}$ 不同态，但$\boldsymbol{L}_{\boldsymbol{A}}^{*}$ 与$\boldsymbol{L}_{\boldsymbol{A}_i|\boldsymbol{A}}^{*}$ 同态.

阈值距离加密.

$|\boldsymbol{td}_{\boldsymbol{A}}^{*}|$ 与$|\boldsymbol{td}_{\boldsymbol{A}|\boldsymbol{A}_i}^{*}|$ 不同态，$|\boldsymbol{td}_{\boldsymbol{A}}^{*}|$ 与$|\boldsymbol{td}_{\boldsymbol{A}_i|\boldsymbol{A}}^{*}|$保持同态.

US

SNS

储存和计算

Alice 发送$(|\boldsymbol{td}^*_{\boldsymbol{A}}|, \boldsymbol{L}^*_{\boldsymbol{A}})$, $(U^*_{A_i}, |\boldsymbol{td}^*_{\boldsymbol{A}|\boldsymbol{A_i}}|, \boldsymbol{L}^*_{\boldsymbol{A}|\boldsymbol{A_i}})$给$SNS$,

而$\boldsymbol{L}^*_{\boldsymbol{A_i}|\boldsymbol{A}}$与$|\boldsymbol{td}^*_{\boldsymbol{A_i}|\boldsymbol{A}}|$ 分别由 m 位好友发送给SNS.

$$\mathbf{M_{A_i|A}} = \begin{pmatrix} U^*_A & U^*_{A_1} & U^*_{A_2} & \cdots & U^*_{A_{m-1}} & U^*_{A_m} \\ |\boldsymbol{td}^*_{\boldsymbol{A}}| & |\boldsymbol{td}^*_{\boldsymbol{A_1}|\boldsymbol{A}}| & |\boldsymbol{td}^*_{\boldsymbol{A_2}|\boldsymbol{A}}| & \cdots & |\boldsymbol{td}^*_{\boldsymbol{A_{m-1}}|\boldsymbol{A}}| & |\boldsymbol{td}^*_{\boldsymbol{A_m}|\boldsymbol{A}}| \\ \boldsymbol{L}^*_{\boldsymbol{A}} & \boldsymbol{L}^*_{\boldsymbol{A_1}|\boldsymbol{A}} & \boldsymbol{L}^*_{\boldsymbol{A_2}|\boldsymbol{A}} & \cdots & \boldsymbol{L}^*_{\boldsymbol{A_{m-1}}|\boldsymbol{A}} & \boldsymbol{L}^*_{\boldsymbol{A_m}|\boldsymbol{A}} \end{pmatrix}$$ ——储存

$$\mathbf{M_{A|C}} = \begin{pmatrix} U^*_A & U^*_{11} & U^*_{12} & \cdots & U^*_{1(k-1)} & U^*_{1k} \\ |\boldsymbol{td}^*_{\boldsymbol{A}}| & |\boldsymbol{td}^*_{\boldsymbol{11}|\boldsymbol{A}}| & |\boldsymbol{td}^*_{\boldsymbol{12}|\boldsymbol{A}}| & \cdots & |\boldsymbol{td}^*_{\boldsymbol{1(k-1)}|\boldsymbol{A}}| & |\boldsymbol{td}^*_{\boldsymbol{1k}|\boldsymbol{A}}| \\ \boldsymbol{L}^*_{\boldsymbol{A}} & \boldsymbol{L}^*_{\boldsymbol{11}|\boldsymbol{A}} & \boldsymbol{L}^*_{\boldsymbol{12}|\boldsymbol{A}} & \cdots & \boldsymbol{L}^*_{\boldsymbol{1(k-1)}|\boldsymbol{A}} & \boldsymbol{L}^*_{\boldsymbol{1k}|\boldsymbol{A}} \end{pmatrix}$$ ——计算

验证和解密查询结果

（1）验证结果中的用户是否都是Alice的查询好友；

（2）验证结果中的数据是否都满足两个计算公式：

$$|td^*_{1j|A}| - |qd^*_A| \geq 0, |qd^*_A| - dist(L^*_A, L^*_{1j|A}) \geq 0$$

不同点：
- UTS 模型——由TTPS完成
- U S 模型——由用户完成

需要进一步改善的地方：

(1) 由于模型使用BV同态加密算法,所以存在计算效率低的问题.

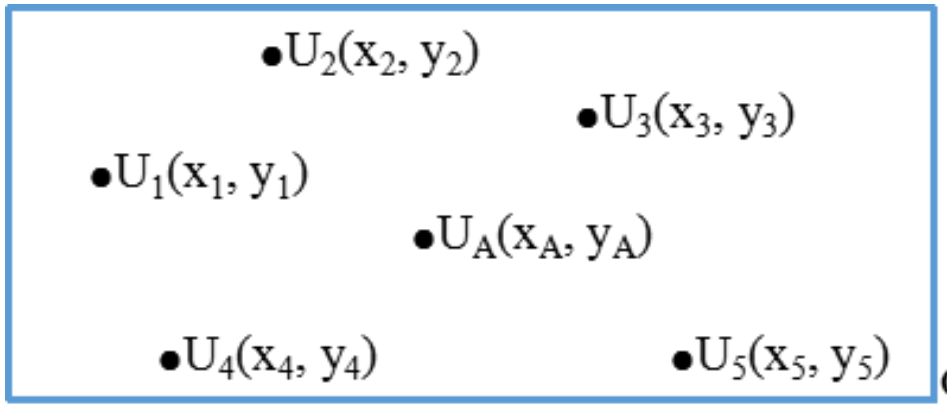
(2) UTS模型引入TTPS帮助用户加密定位信息和验证查询结果,但很多方案提出TTPS的计算瓶颈.

(3) US模型对客户端的储存和计算能力要求较高.

去中心化；高效率；低储存

K 匿名

指对要保护的敏感信息添加$k-1$ 条相同类别的信息, 使攻击者从所有数据中识别出敏感信息的概率为$\frac{1}{k}$.



难以从k 位用户中辨别出当前查询者的精确定位

paillier同态加密算法

加密 $c = g^m \cdot r^n (mod\ \ n^2).$

解密 $m = L(c^\lambda (mod\ \ n^2)) \cdot \mu (mod\ \ n)$

满足的加法同态性质和乘法性质分别为

$$D(E(m_1, pk) \cdot E(m_2, pk)(mod\ n^2)) = m_1 + m_2 (mod\ n)$$

$$D(E(m_1, pk)^{m_2} (mod\ n^2)) = m_1 m_2 (mod\ n)$$

$$D(E(m_1, pk)^k (mod\ n^2)) = k m_1 (mod\ n)$$

BV同态加密算法

密钥生成 公/私钥对 (pk_A, sk_A)

$$pk_A = (b_A, a_A) \quad sk_A = s_A$$

$$b_A = -(a_A s_A + q e_A). \; a_A \in \mathbb{R}_p$$

$$(s_A, e_A), (u_A, f_A, g_A) \; discrete \;\; Gaussian \;\; error \; \chi$$

$$|q| = \lambda. \; \mathbb{R} = Z[x]/(x^2+1), \mathbb{R}_p = Z_p[x]/(x^2+1), \mathbb{R}_q = Z_q[x]/(x^2+1).$$

加密

$$C = E_{pk}(m) \quad C = (c_0, c_1)$$

$$c_0 = b_A u_A + q g_A + m, \; c_1 = a_A u_A + q f_A.$$

解密

$$m = (c_0 + c_1 s_A) \, mod \, q$$