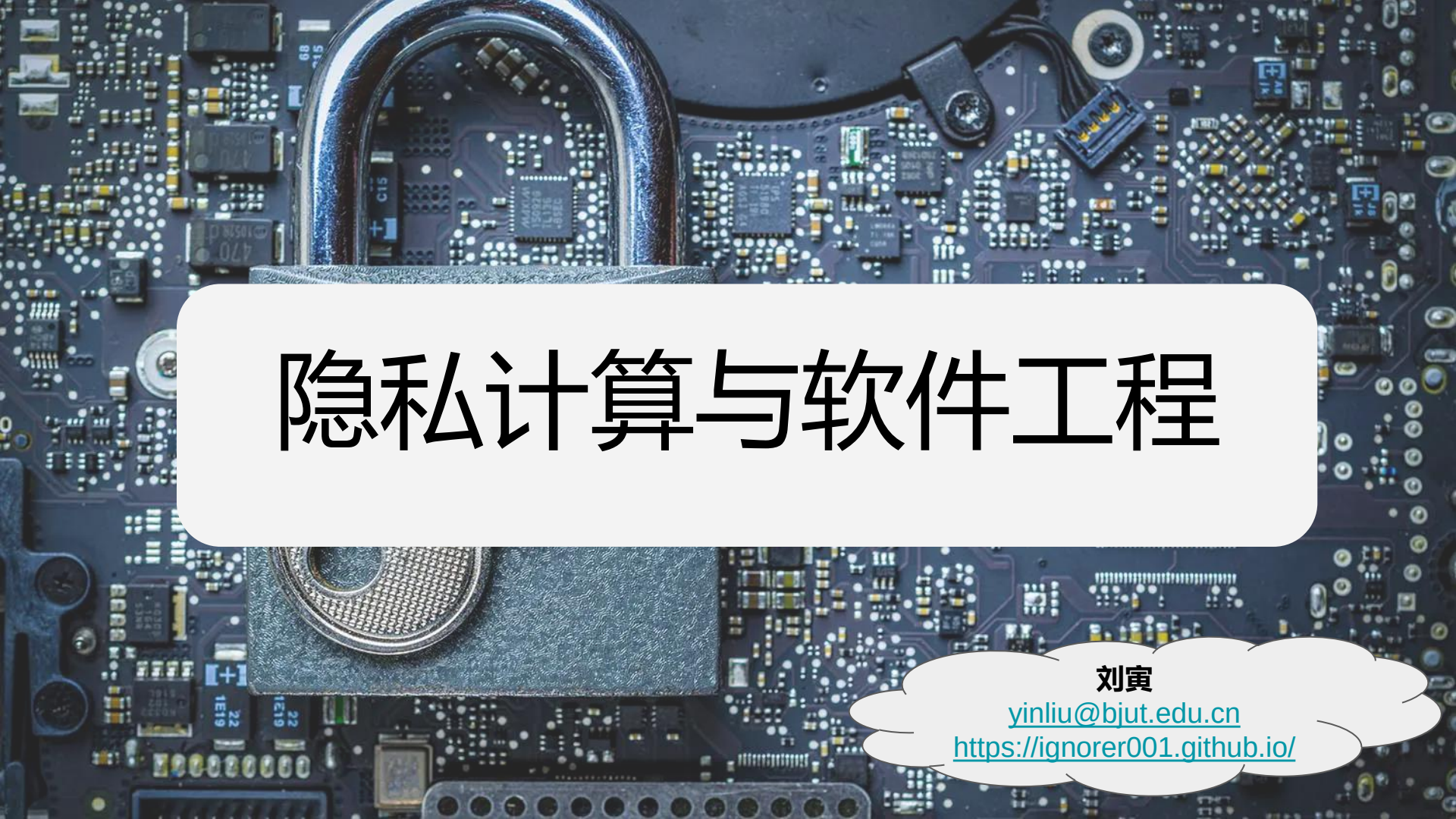


隐私计算与软件工程

刘寅
yinliu@bjut.edu.cn
https://ignorer001.github.io/

Contact

刘寅
北京工业大学 平乐园校区 软件楼-319
yinliu@bjut.edu.cn
https://ignorer001.github.io/

2024.10.21 updated

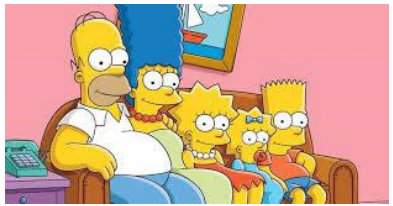
Agenda

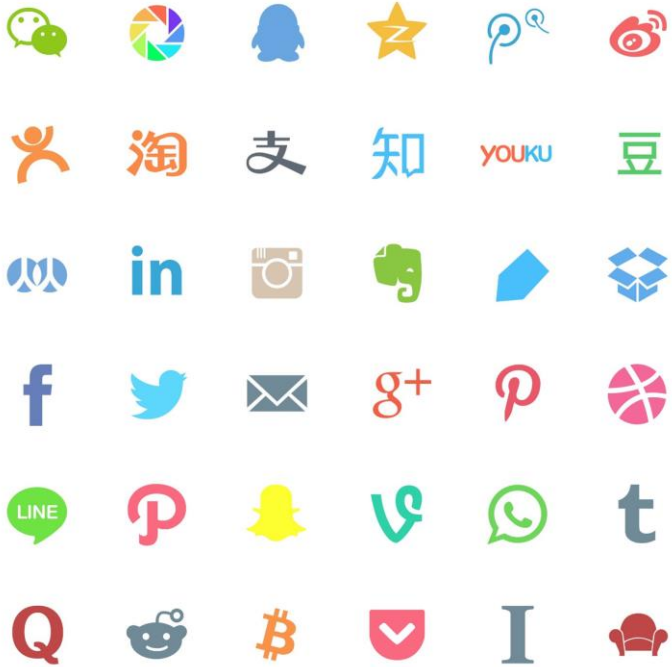
1. 什么是隐私
2. 什么是隐私计算
3. 隐私计算技术
4. 如何与软件工程结合

何谓隐私 — “隐”+“私”

隐私是自然人的**私人生活安宁和不愿为他人知晓**的**私密空间、私密活动、私密信息**（中华人民共和国民法典-第六章-第一千零三十二条）

个人信息





个人信息是以**电子或者其他方式**记录的与**已识别**或者**可识别的自然人有关的各种信息**，不包括匿名化处理后的信息。**个人信息的处理**包括个人信息的**收集、存储、使用、加工、传输、提供、公开、删除等。**（中华人民共和国个人信息保护法-第一章-第四条）

隐私保护重要吗?

Baptist Medical Center（美国德克萨斯州圣安东尼奥）

2022- 恶意软件攻击造成数据泄露。该事件是美国卫生与公众服务

HOME › TECH

533 million Facebook users' phone numbers and personal data have been leaked online

Aaron Holmes Apr 3, 2021, 10:41 PM

位于密歇根州特洛伊的美国星旗银行，发生了一次重大数据泄露事件，受影响人数达到**154万人**。

NEW YORK/BOSTON（Reuters）- Sony suffered a massive breach in its video game online network that led to the theft of names, addresses and possibly credit card data belonging to 77 million user accounts in what is one of the largest-ever Internet security break-ins.

为什么需要收集/处理数据

图表2：大数据产业全景图谱

资料来源：前瞻产业研究院整理

隐私计算

保护

使用

隐私计算

核心思想

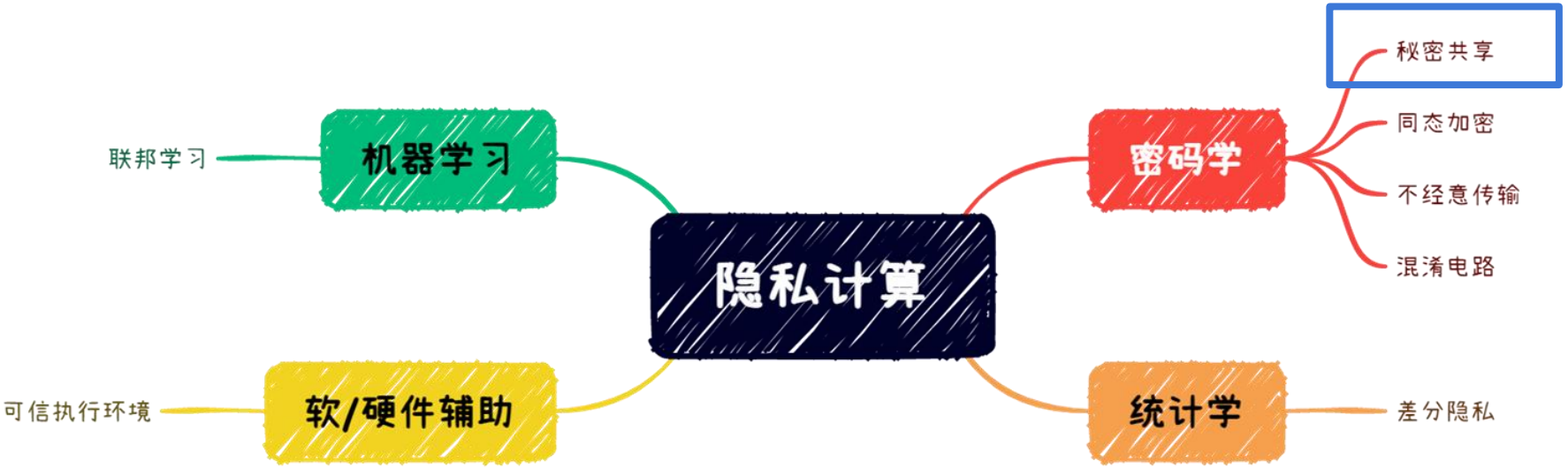
可用但不可见

4个领域，7个技术

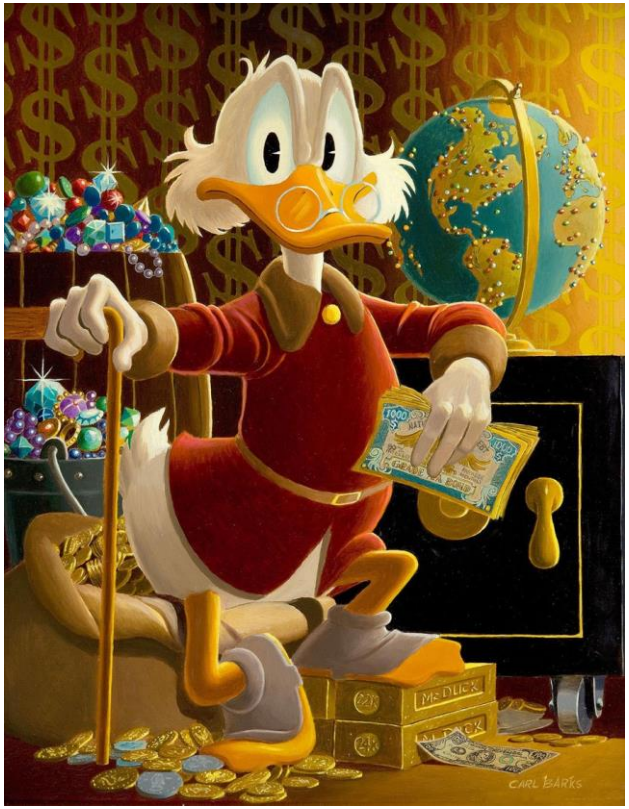
隐私计算

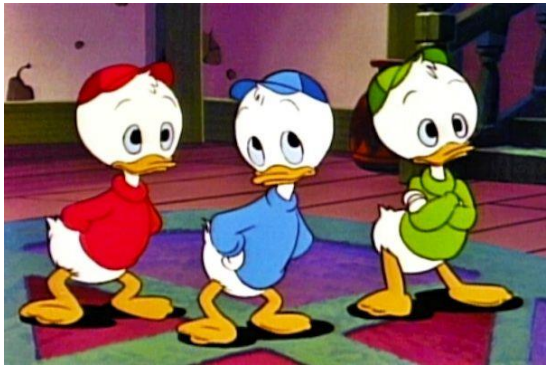

隐私计算
机器学习
联邦学习
密码学
秘密共享
同态加密
不经意传输
混淆电路
软/硬件辅助
可信执行环境
统计学
差分隐私

隐私计算


隐私计算
机器学习
联邦学习
密码学
秘密共享
同态加密
不经意传输
混淆电路
软/硬件辅助
可信执行环境
统计学
差分隐私

秘密共享(Secret Sharing) - 问题描述





如何既保护好金库，又使其可用?

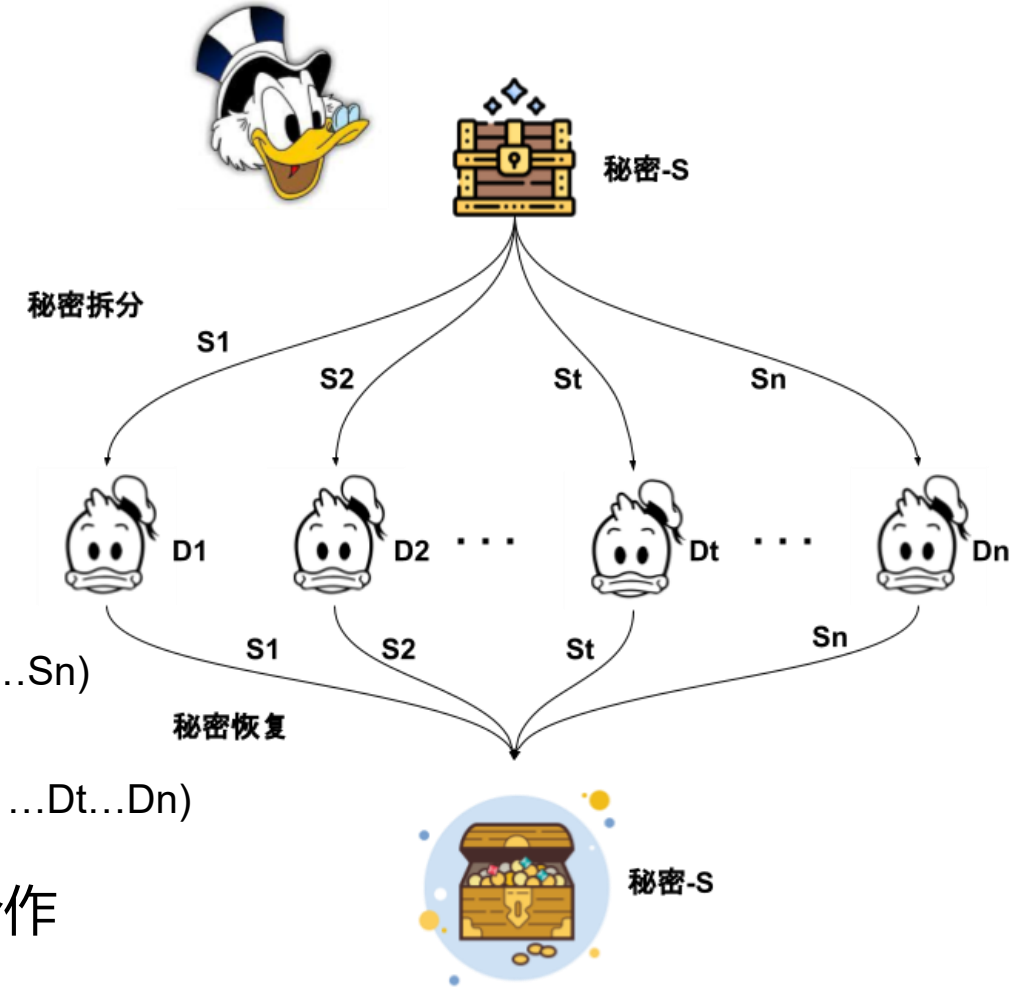
方案一：锁起来，***钥匙一个人保管***

方案二：三把钥匙，**三人都能打开**

方案三：三把钥匙，单独不能打开，
任意两人在场，就可以打开

秘密共享 - 核心思想

拆分 → 分发 → 恢复



秘密拆分：秘密S拆分成n份(S1…St…Sn)

秘密分发：给不同的参与方管理(D1…Dt…Dn)

秘密恢复：需要 **至少 t** 个参与方合作

秘密共享 - 实现思路

-2x+y=0 (1)

x-y=-1 (2)

x+y=3 (3)

x=?

y=?

秘密共享 - 实现思路

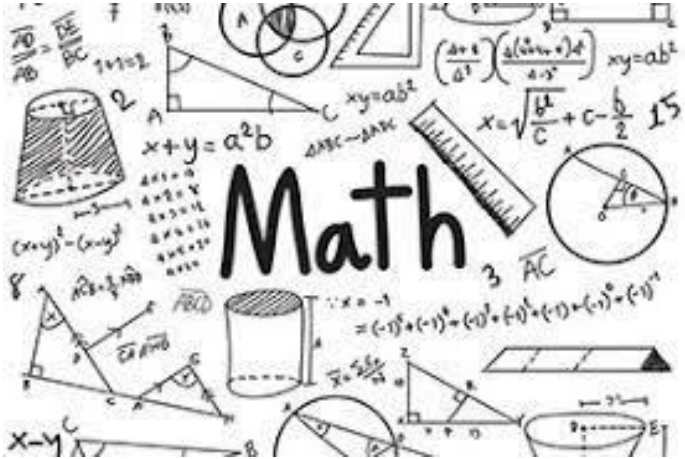
任意两条直线方程 — 求交点

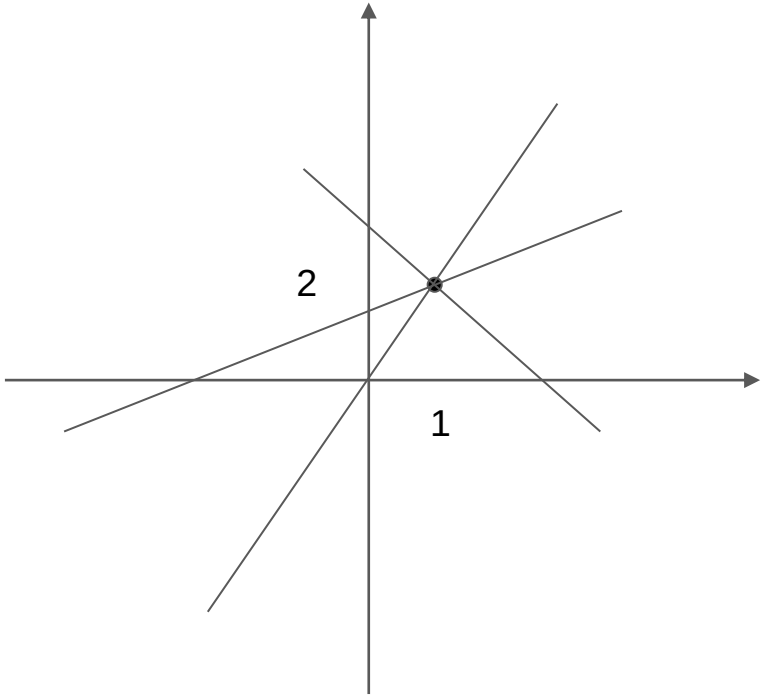
1. 确定**n=3, t=2**
2. 假设**秘密S=(1,2)**, 为 t 空间中的一个**点**
3. 构造经过这个点的n=3条直线：

-2x+y=0

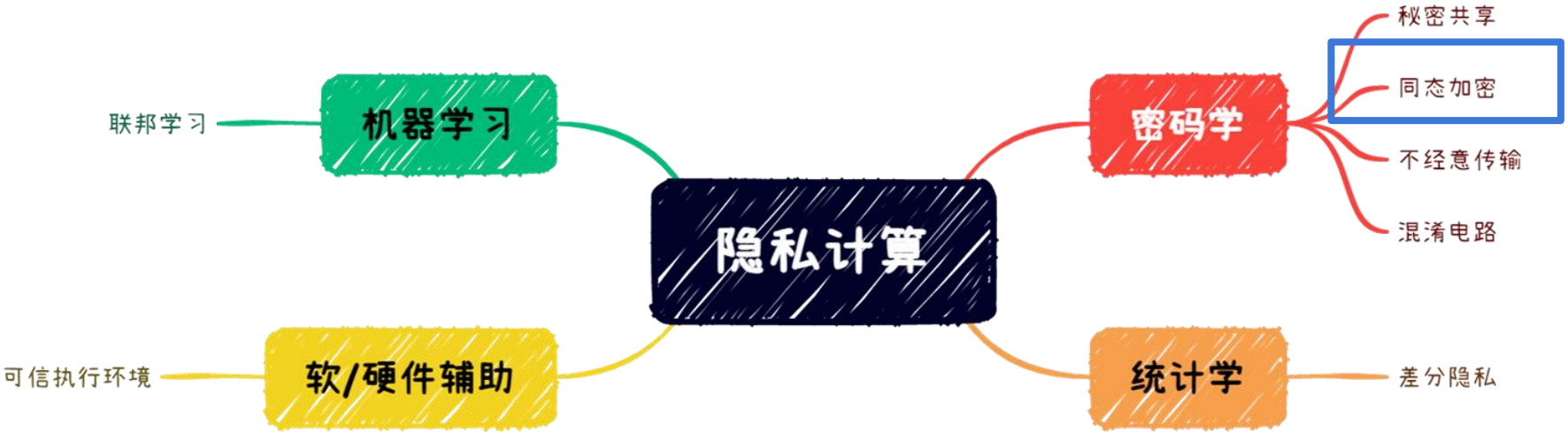
x-y=-1

x+y=3

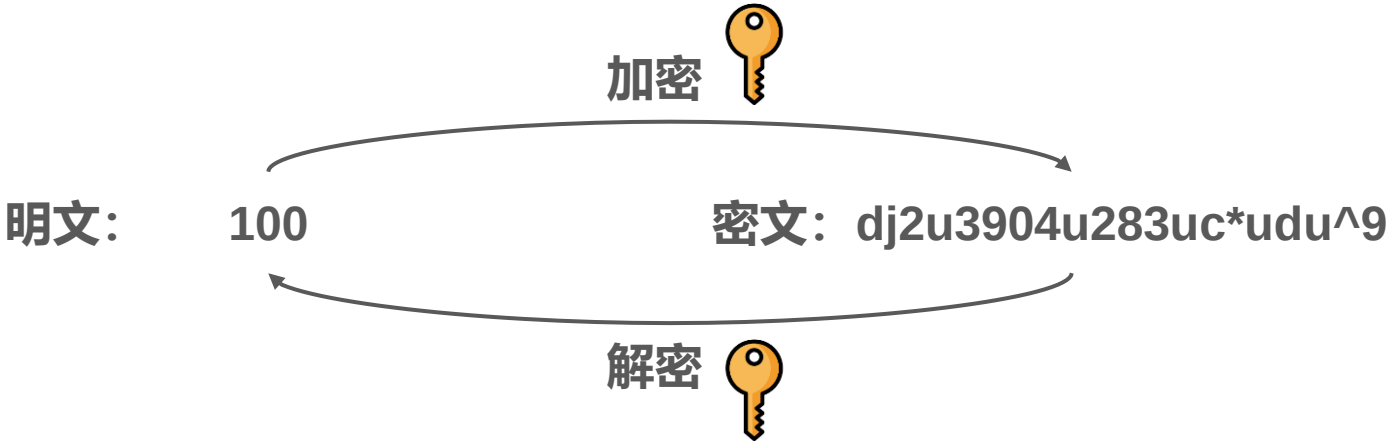




隐私计算


隐私计算
机器学习
联邦学习
密码学
秘密共享
同态加密
不经意传输
混淆电路
软/硬件辅助
可信执行环境
统计学
差分隐私

同态加密(Homomorphic Encryption) – 密码学基本概念



密钥 – 用来完成加密、解密、完整性验证等密码学应用的秘密信息

密码学基本概念

对称加密 – 加密解密用**同一个密钥**

非对称加密 – **公钥**加密、**私钥**解密、**公钥可公开**

加密的核心思想！！！

构造困难问题和解决困难问题的难度不对称！！！

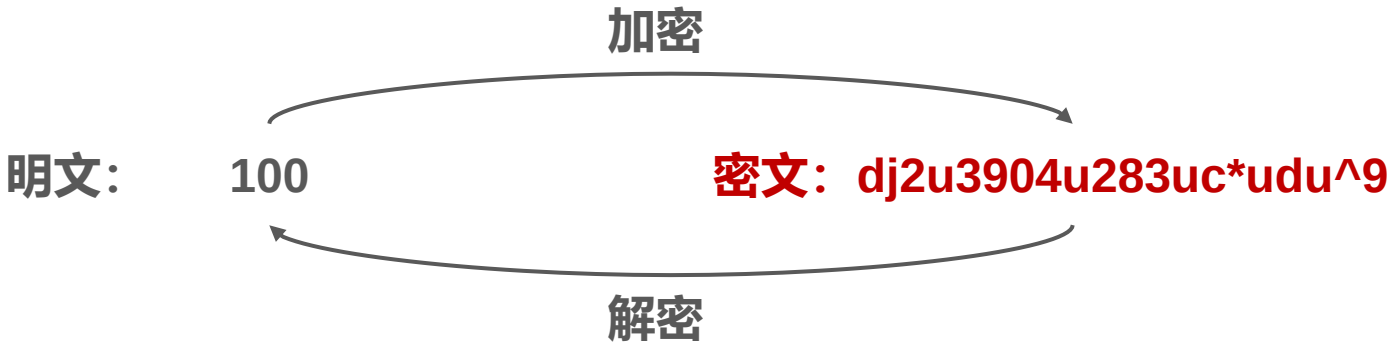
构造困难问题---简单，解决困难问题---困难

RSA：给定两个超大素数p,q,
计算n=p*q很简单，给出n反推p,q极难

密码的问题

不可见也不可用！

同态加密
can help！



同态加密

同态加密

不可见但可用！

明文：100

密文：cfkljsadjal!@#n1nkjnkjn

+1

密文：asduu1892u312nk1ne

明文：101

解密

同态加密

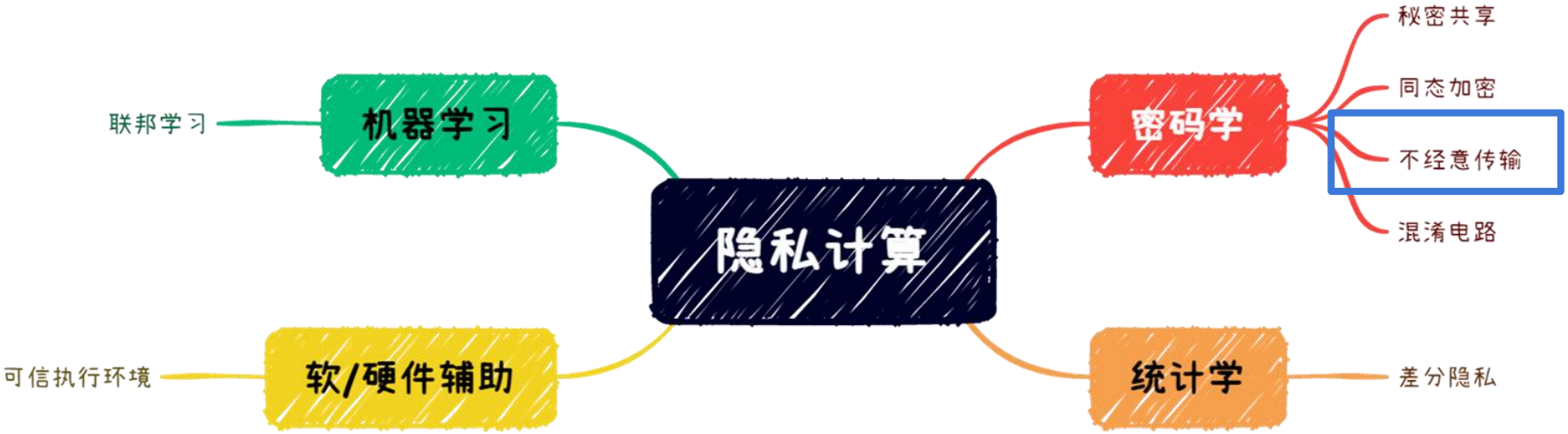
部分同态加密 — 只支持单一同态操作(e.g, 加法)

近似同态加密 — 支持多种同态操作，但在密文上执行次数有限

层级同态加密 — 支持多种同态操作，可自定义操作次数上限

全同态加密 — 无限制各类同态操作

隐私计算


隐私计算
机器学习
联邦学习
密码学
秘密共享
同态加密
不经意传输
混淆电路
软/硬件辅助
可信执行环境
统计学
差分隐私

不经意传输(Oblivious Transfer) – 基本概念

具体场景 — 老师有两份批改好的试卷，其中有一份是小刘的（30分）
假设老师不认识小刘

目的：要告诉小刘的得分，且保证隐私

方案1：上课时点名叫小刘过来，把对应的试卷给他

方案2：下课后把两张卷子放桌子上，让小刘自己看

Teacher

Liu

不经意传输 – 基本概念

方案3：下课后把两张卷子扣着放桌子上，卷子背后写上学生的名字，让小刘只看他的那张

不经意传输

Teacher

Liu

不经意传输 – 基本概念

“2选1” 模型：

发送方Teacher，接收方Liu

Teacher有两条隐私数据X0,X1

Liu有一个选择位 a ∈ {0,1}

最终接收方Liu得到Xa

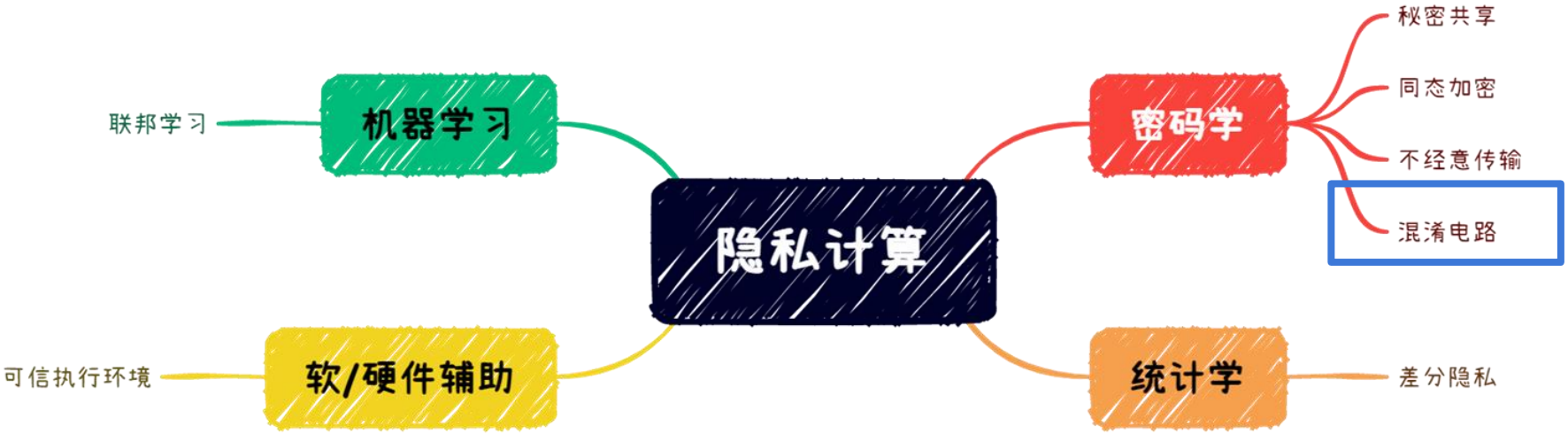
达到的目的：

发送方Teacher：只知道 Liu接收了两条信息中的一条，**不知道**Liu接收了哪条信息。

接收方Liu：只知道自己接收到的信息的内容，**不知道**另一条信息的内容。

推广：K选1模型

隐私计算


隐私计算
机器学习
联邦学习
密码学
秘密共享
同态加密
不经意传输
混淆电路
软/硬件辅助
可信执行环境
统计学
差分隐私

混淆电路（Garbled Circuit）

核心思想：对“用电路描述的计算任务”进行“混淆操作”

电路：输入信号、逻辑门 (与或非)、输出信号

计算任务：输入 –> 计算 –>输出

加密

混淆电路（Garbled Circuit）

核心思想：对“用电路描述的计算任务”进行“混淆操作”

“与门”电路真值表

a	b	输出
0	0	0
0	1	0
1	0	0
1	1	1

混淆（加密）

混淆后的“与门”电路真值表

a	b	输出
X_{a0}	X_{b0}	X_{c0}
X_{a0}	X_{b1}	X_{c0}
X_{a1}	X_{b0}	X_{c0}
X_{a1}	X_{b1}	X_{c1}

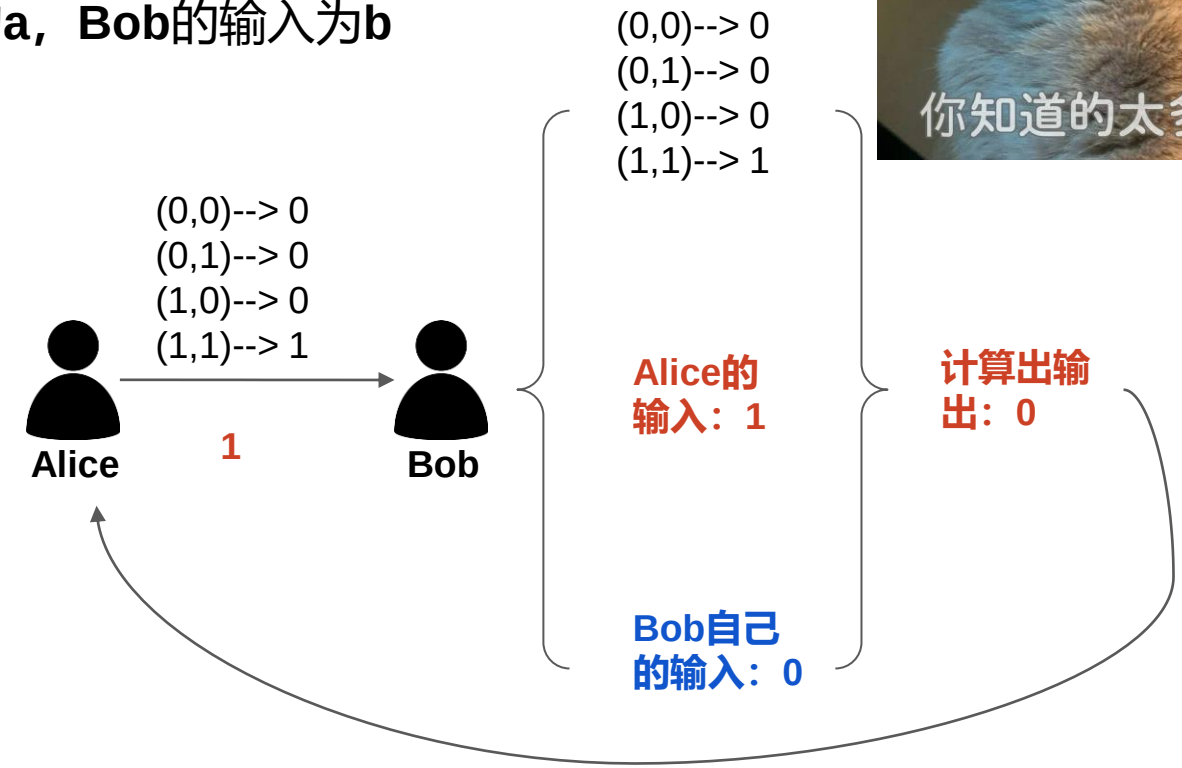
真值表 == 输入-->输出的映射

混淆电路（Garbled Circuit）

Alice的输入为真值表中的a，Bob的输入为b

不使用混淆电路的情况：

a	b	输出
0	0	0
0	1	0
1	0	0
1	1	1

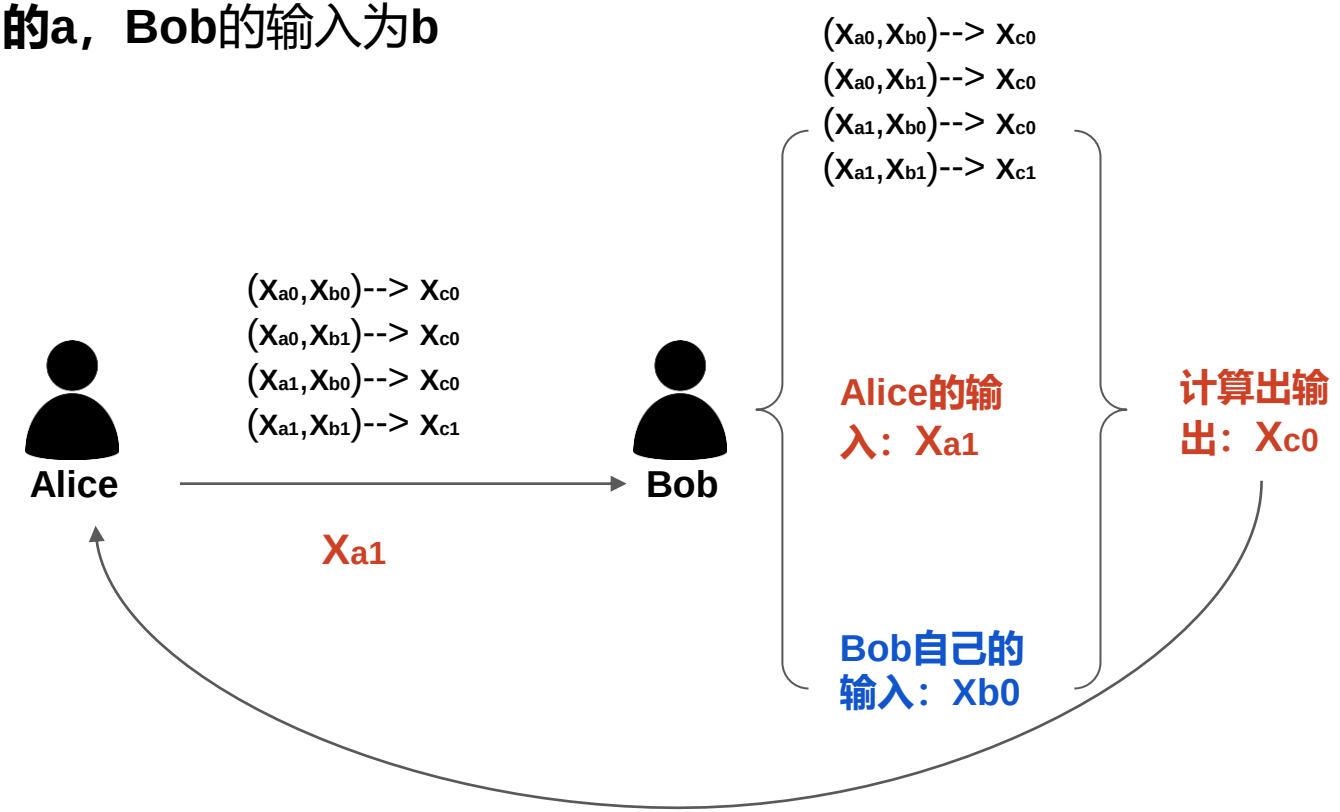


混淆电路（Garbled Circuit）

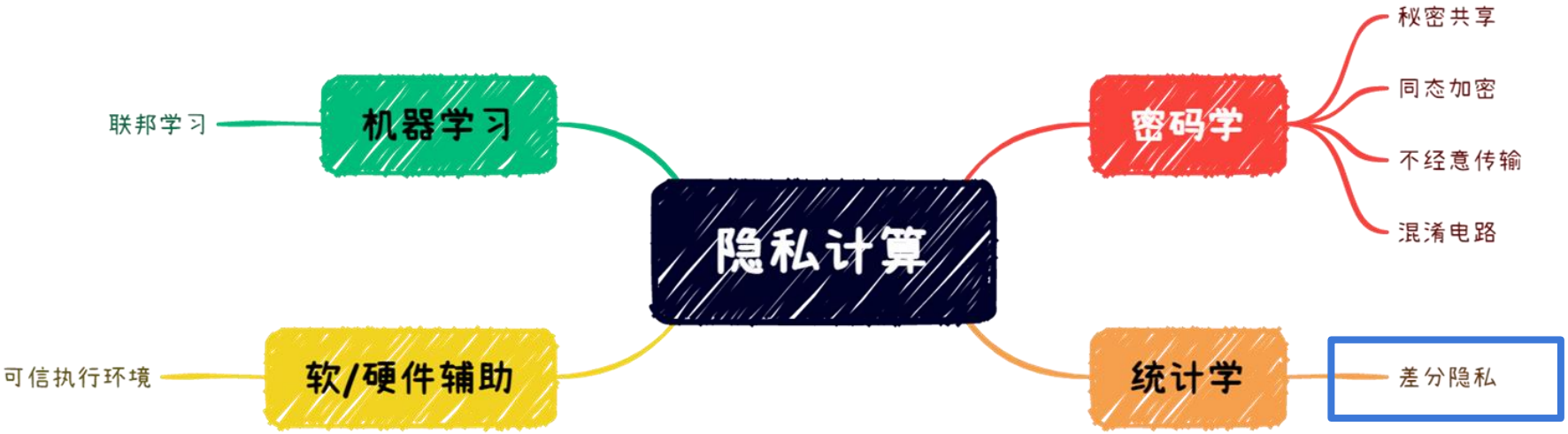
Alice的输入为真值表中的a，Bob的输入为b

使用混淆电路：

a	b	输出
X_{a0}	X_{b0}	X_{c0}
X_{a0}	X_{b1}	X_{c0}
X_{a1}	X_{b0}	X_{c0}
X_{a1}	X_{b1}	X_{c1}



隐私计算


隐私计算
机器学习
联邦学习
密码学
秘密共享
同态加密
不经意传输
混淆电路
软/硬件辅助
可信执行环境
统计学
差分隐私

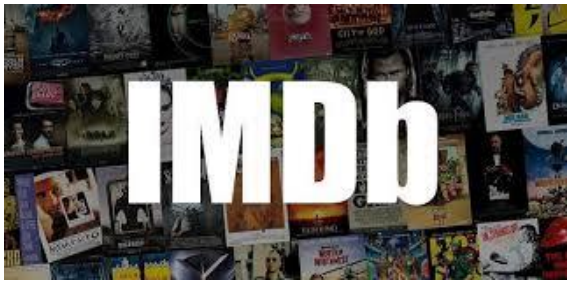
差分隐私(Differential Privacy) — 问题的提出

隐私 — 匿名数据 — 万事大吉？？？

Netflix 开放了一个数据集，~50万个**随机**选择的**匿名**用户的电影评分。

用户 ID（随机分配，无法获取真实身份）、电影信息（年份、标题、评分等）

公开数据库



推断Netflix 匿名用户的身份

差分隐私 — 问题的提出

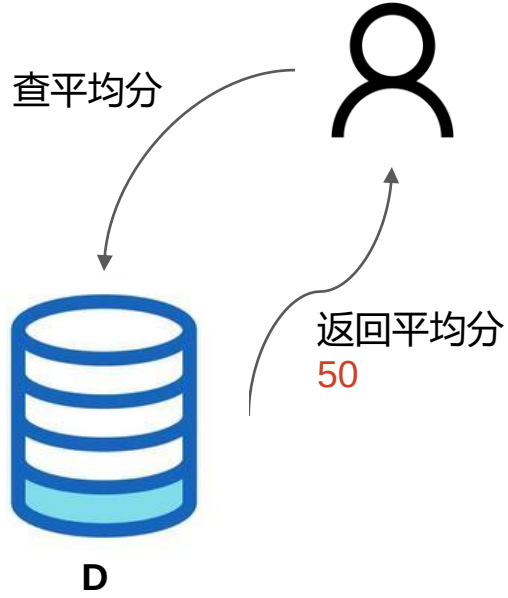
隐私 — 匿名数据

学生	成绩
Alice	100
Bob	30
Steve	70
Bill	0

匿名化

学生	成绩
X1	100
X2	30
X3	70
X4	0

不可见



差分隐私 — 问题的提出

隐私 — 匿名数据

学生	成绩
Alice	100
~~Bob~~	~~30~~
Steve	70
Bill	0

匿名化

学生	成绩
X1	100
~~X2~~	~~30~~
X3	70
X4	0

不可见

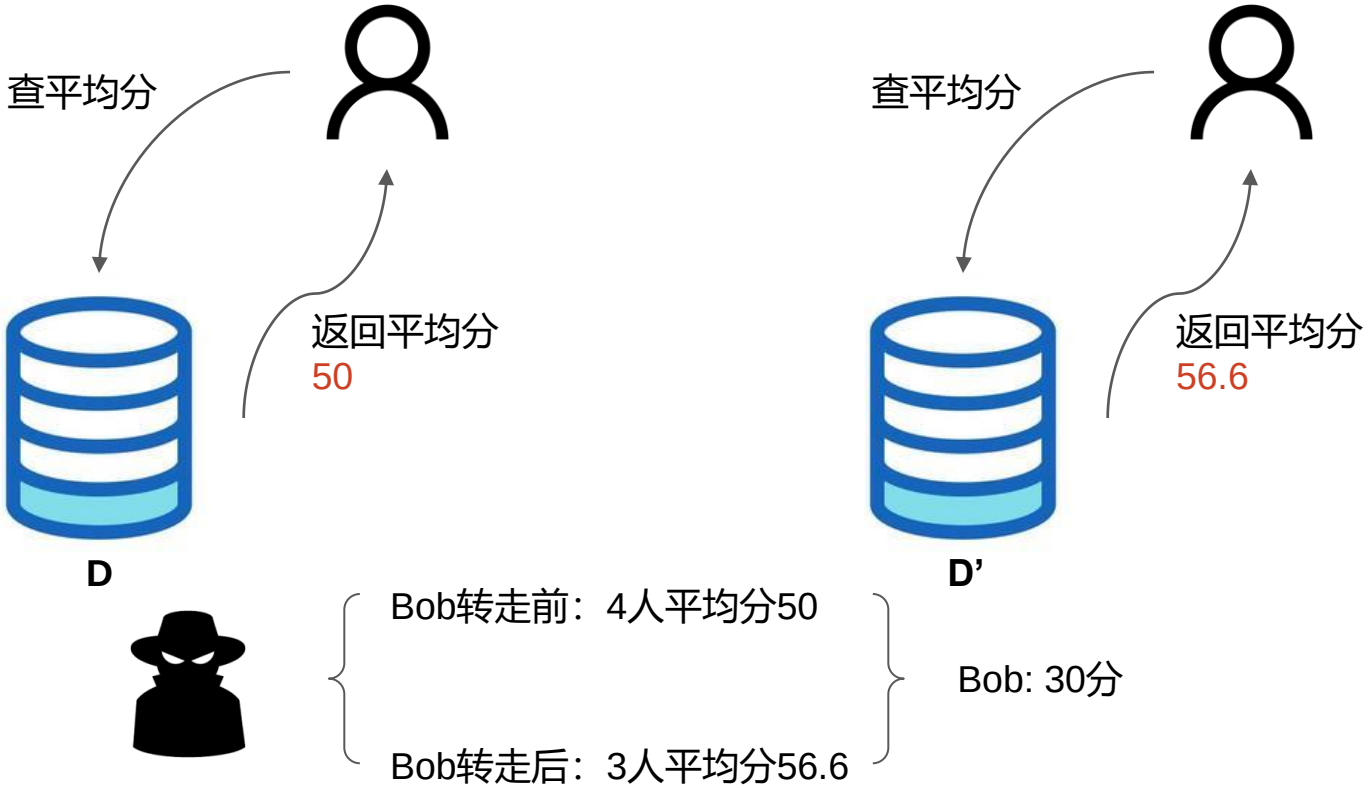
D’

查平均分

返回平均分
56.6

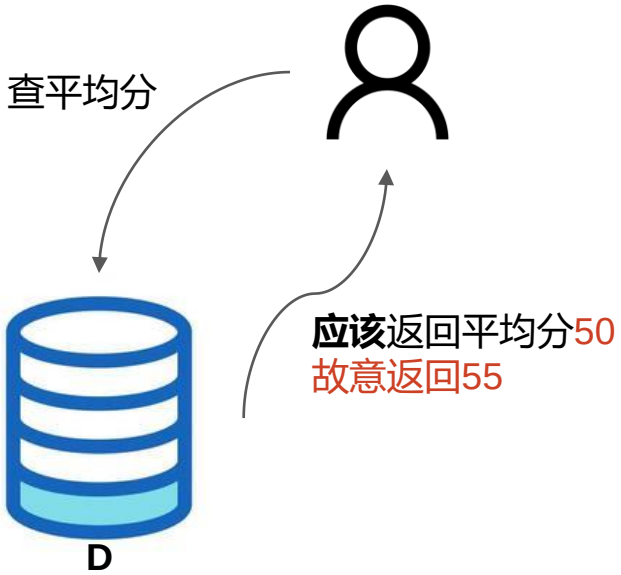
转走了

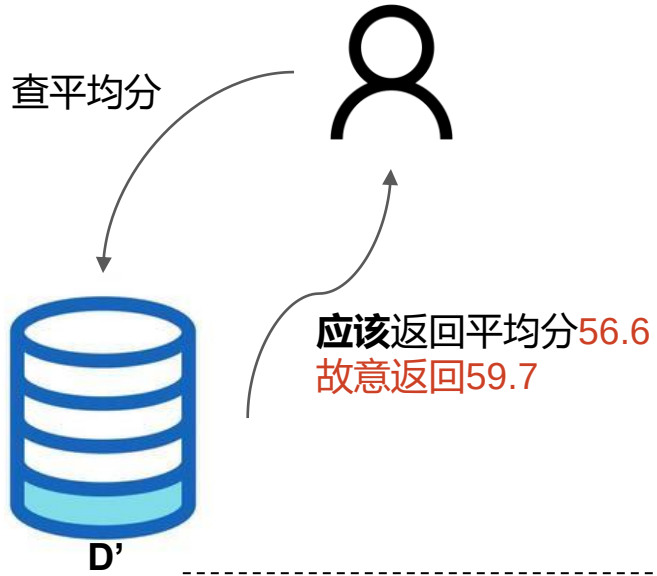
差分隐私 — 问题的提出


查平均分
返回平均分
50
D
查平均分
返回平均分
56.6
D’
Bob转走前：4人平均分50
Bob转走后：3人平均分56.6
Bob: 30分

差分隐私 — 基本概念

可用：能知道全班平均分还是没及格
不可见：真实平均分不可见





核心：加入合适的噪声

Bob转走前：4人平均分**55**

Bob转走后：3人平均分**59.7**

Bob: 实际是30分，但是计算所得**40.9**

差分隐私 — 数学定义

M是一个算法或操作
例如：假设M是取平均数的算法(操作)，则M(D)为M在数据库D中求取平均数的结果。
R是M的值域

Definition 1: ε-differentially private mechanism. Given $\varepsilon \geq 0$, M is ε-differentially private, iff for all neighboring databases (D, D'), and for any sets of outputs $S \subseteq R$: $Pr[M(D) \in S] \leq e^{\varepsilon} Pr[M(D') \in S]$

相邻数据集(neighboring databases)：
D, D' 相差最多一条数据

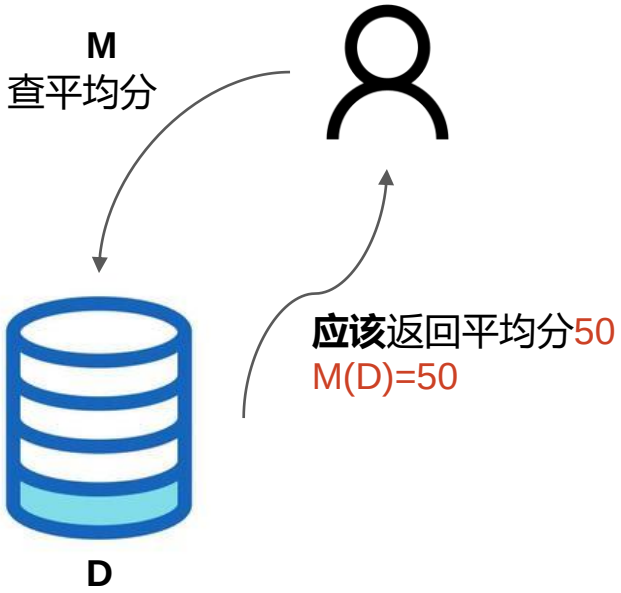
M(D)和M(D')的概率差不多

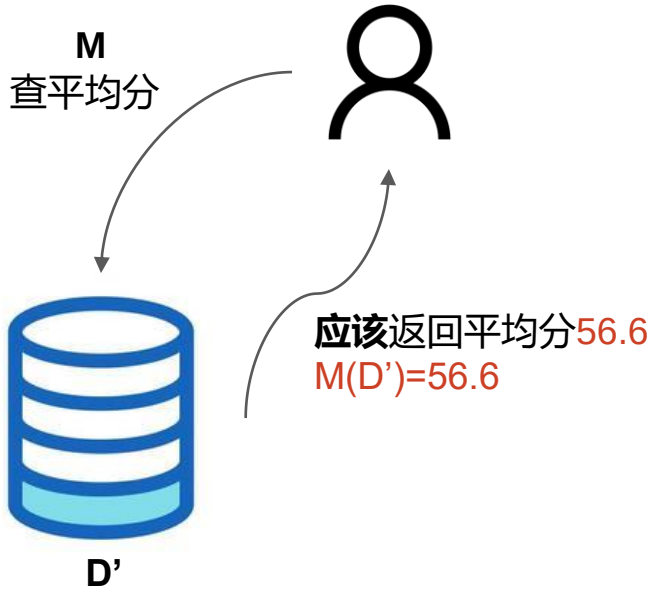
$$\frac{Pr[M(D) \in S]}{Pr[M(D') \in S]} \leq e^{\varepsilon}$$

其中ε称为隐私预算(privacy budget)

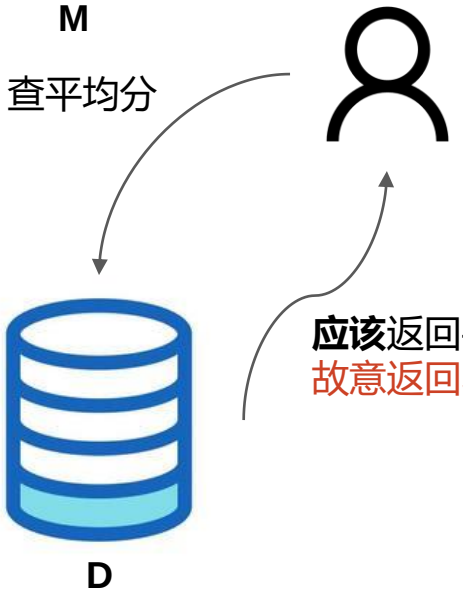
差分隐私

此时的**M**算法返回**确定值(**50或56.6), **不满足差分隐私**





差分隐私



应该返回平均分50
故意返回55

需要巧妙的加一个符合某种概率分布的噪声，使M满足ε-差分隐私

- 拉普拉斯噪声法
- 高斯噪声法

55 = 50 + **5(此次查询的噪声)**

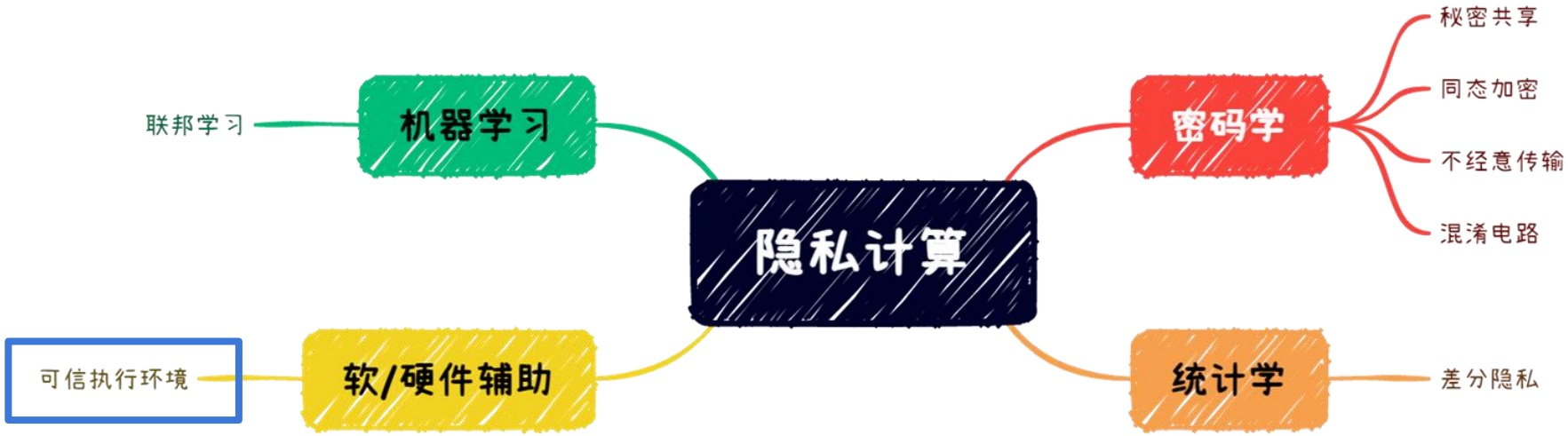
M满足ε-差分隐私

$$\frac{Pr[M(D) \in 值域]}{Pr[M(D') \in 值域]} \leq e^{\varepsilon}$$

给M算法增加随机性！！！

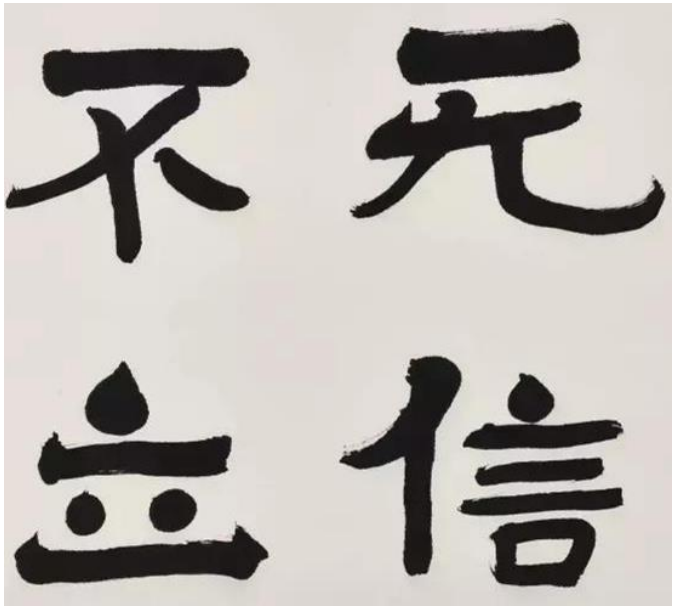
差分隐私---依赖于随机性！！！

隐私计算


隐私计算
机器学习
联邦学习
密码学
秘密共享
同态加密
不经意传输
混淆电路
软/硬件辅助
可信执行环境
统计学
差分隐私

什么是“**可信**”

如果针对某个特定的目的，*实体的行为与预期的行为相符*，则称针对这个目的，该实体是可信的。



什么是“**可信计算**”

欲得黄金，必有所信

- **商鞅**
- **命令**
- 秦国的**社会环境**

南门立木:
商鞅 发布命令
从南门搬到北门→ 赏金千两

可信计算 --- ***三个方面***

- **信任 “商鞅”** *程序的运行者(使用者)*
- **信任 “命令”** *运行的程序*
- **信任**秦国的 **“社会环境”** *平台运行环境*

商鞅 --- 运行一个程序

```
void moveTimber(Person p, Timber t) {
        if (canMoveTimber(p,t)) {
                p.money += 1000;
        }
}
```

可信计算 --- ***包含的技术***

信任*程序的使用者* **身份认证** **软件方法：密码学**

信任*运行的程序* **完整性（integrity）、私密性（confidentiality）**

信任*平台运行环境* **即使被攻破，也依然可信** **硬件方法：隔离**

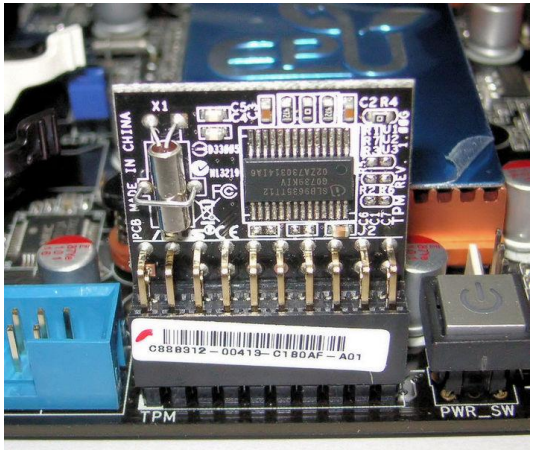
可信计算技术 == 软件（密码学）+ 硬件（隔离）

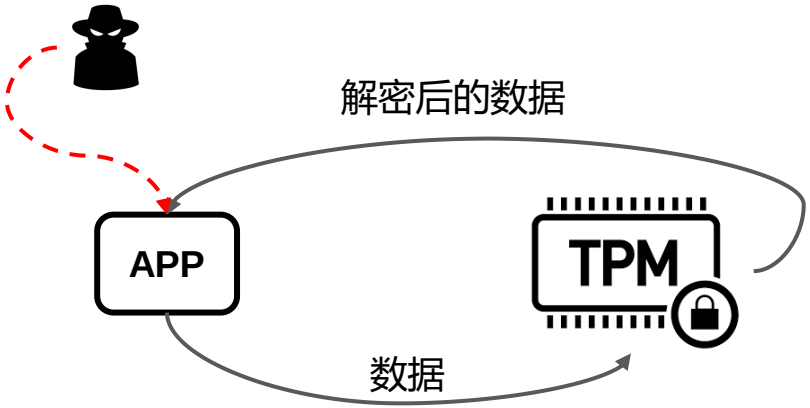
可信平台模块 --- TPM (Trusted Platform Module)

是什么 --- 独立于CPU的计算模块

做什么 --- 加/解密，存储，验证

问题 --- 无法防御**运行时**攻击

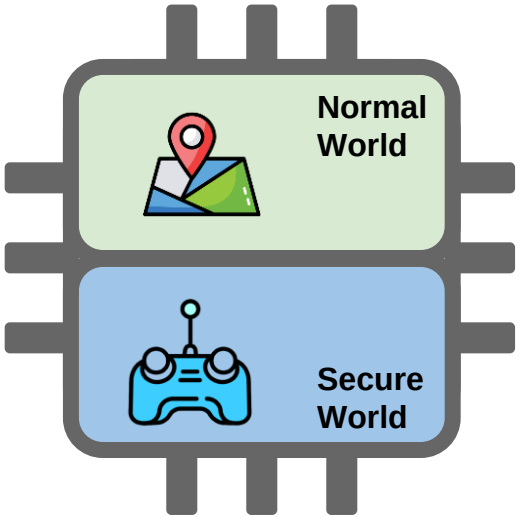




可信执行环境 --- TEE (Trusted Execution Environment)

不仅是加密、验证 --- 支持各种操作

CPU的自我隔离 --- 划分出两个区域（normal and secure world）



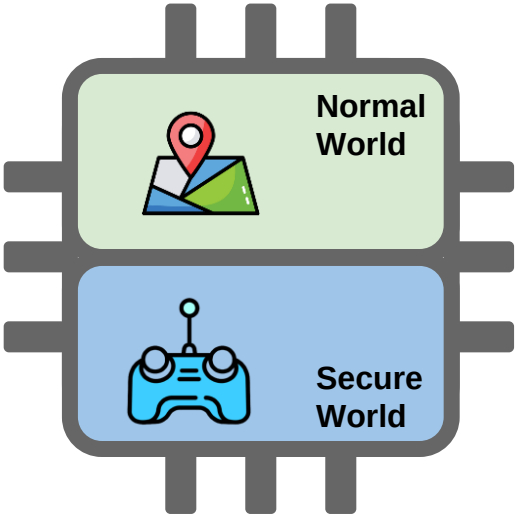
OP-TEE

Amazon, Wing, FedEx, Walgreens

无人机送货计划

无人机的导航算法---**被攻破**

TEE can help!



隐私计算

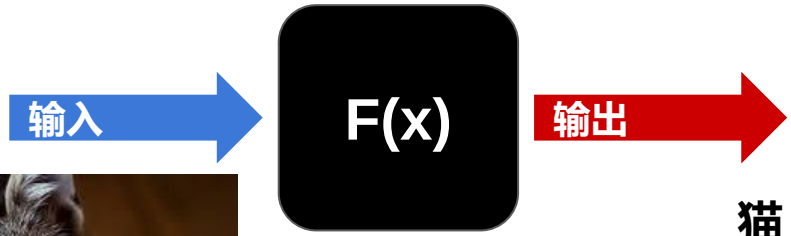

隐私计算
机器学习
联邦学习
密码学
秘密共享
同态加密
不经意传输
混淆电路
软/硬件辅助
可信执行环境
统计学
差分隐私

联邦学习(Federated learning)

机器学习，深度学习，联邦学习？？？

机器学习是什么?

学到了一个**函数**

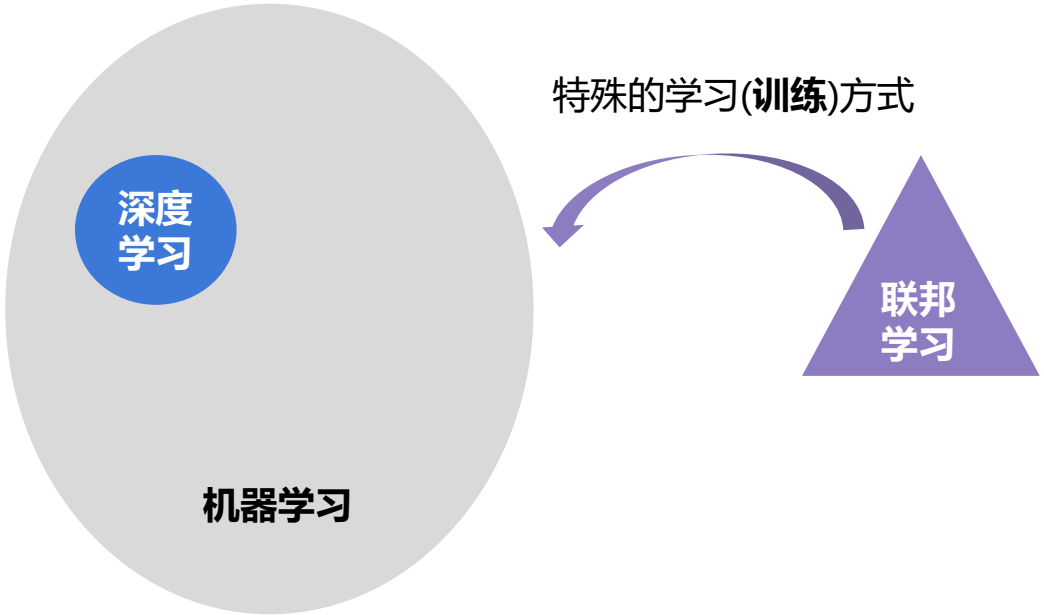


猫　　**分类**

房屋特征(位置\大小\朝向, etc.)　　**房屋价格**　　**回归**

联邦学习

机器学习，深度学习，联邦学习？？？



联邦学习

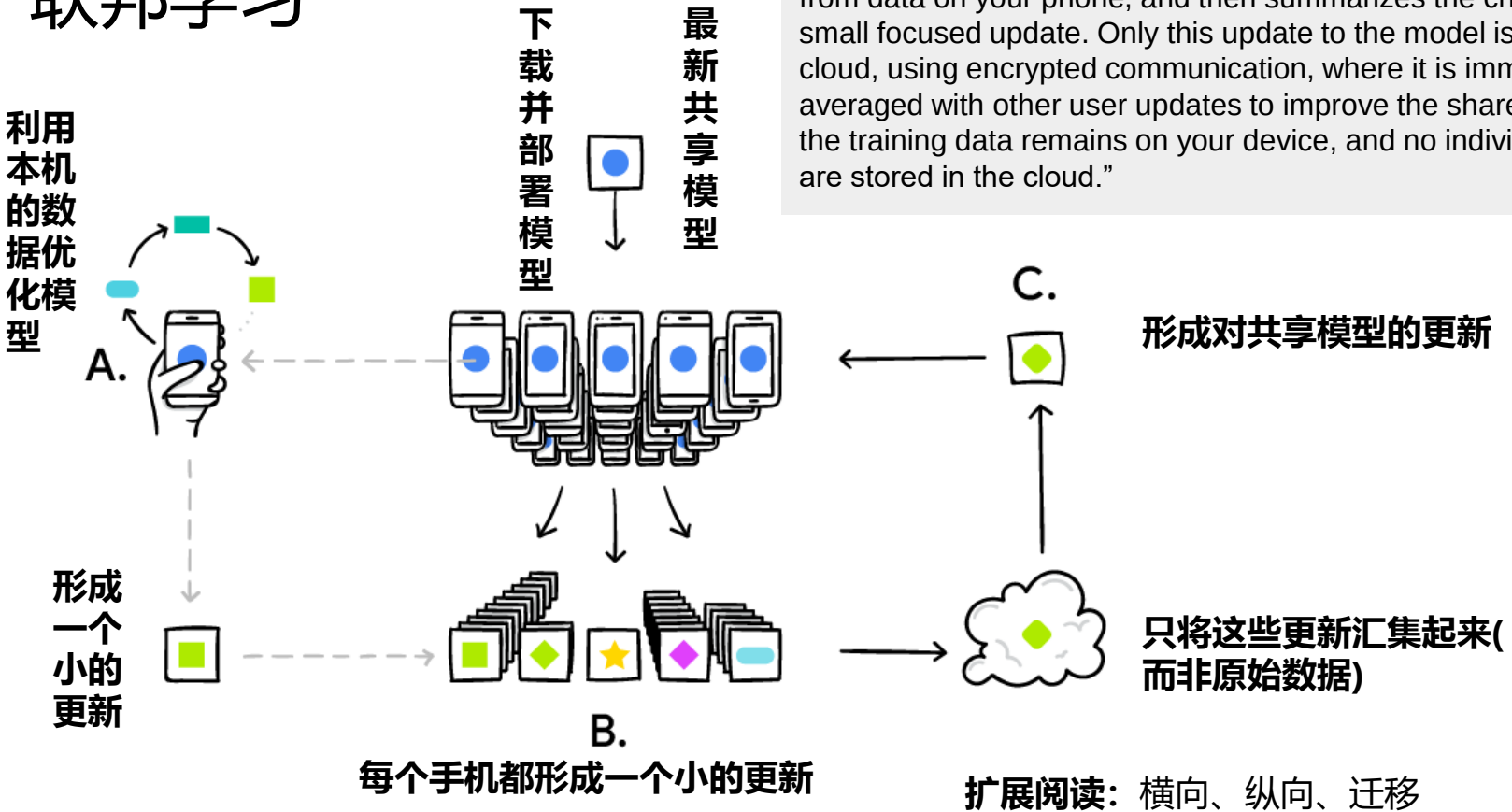
一种模型训练方式，在**训练过程**中**保护数据隐私**

一种具有以下特征的机器学习**算法框架:**

- **多方参与** — 2个以上，每个参与方都拥有(一部分)数据
- **原始数据不交换** — 不交换或收集任意参与方的**原始**数据
- **传输"已保护信息"** — 训练所必须的信息需**先经保护处理，再进行传输**
- **近似无损** — **性能**充分接近**原有模型** (i.e., 未使用联邦学习，各参与方直接共享原始数据训练所得的模型)

联邦学习

It works like this:
“your device downloads the current model, improves it by learning from data on your phone, and then summarizes the changes as a small focused update. Only this update to the model is sent to the cloud, using encrypted communication, where it is immediately averaged with other user updates to improve the shared model. All the training data remains on your device, and no individual updates are stored in the cloud.”



扩展阅读：横向、纵向、迁移

隐私计算


隐私计算
机器学习
联邦学习
密码学
秘密共享
同态加密
不经意传输
混淆电路
软/硬件辅助
可信执行环境
统计学
差分隐私

隐私计算的挑战 — 世上没有免费的午餐

保护

使用

计算效率

传输效率

准确性

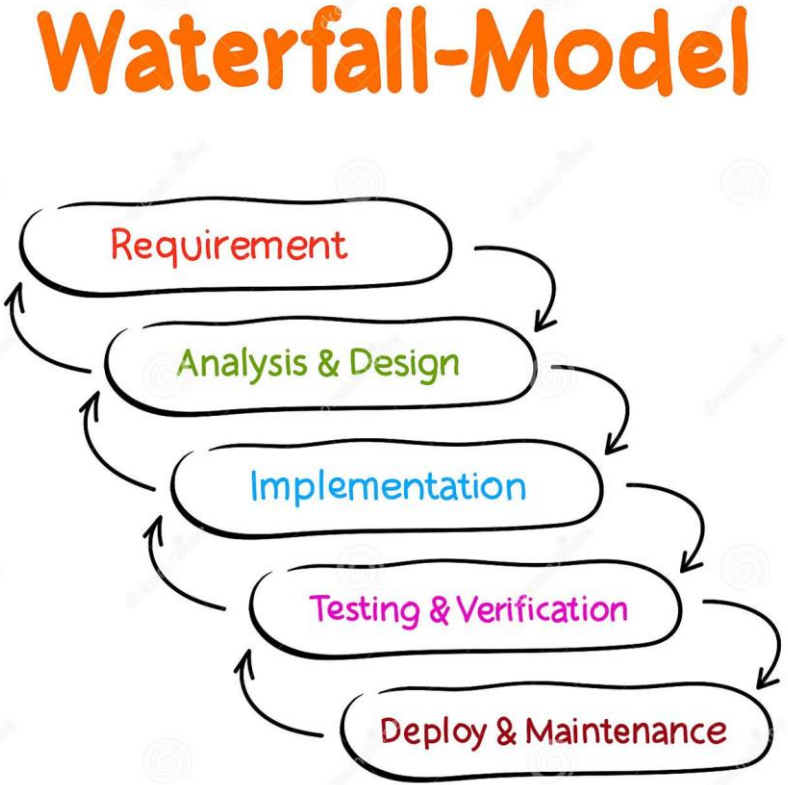
不可信的
参与方

隐私计算与软件工程

什么是软件工程?

软件工程

瀑布模型


Waterfall-Model
Requirement
Analysis & Design
Implementation
Testing & Verification
Deploy & Maintenance

软件工程与隐私计算

目标：

保护敏感程序信息

并使其可用

问题引入

1. 如何识别?

2. 如何保护?

问题引入

1. 如何识别？

2. 如何保护？

VarSem (GPCE'20)

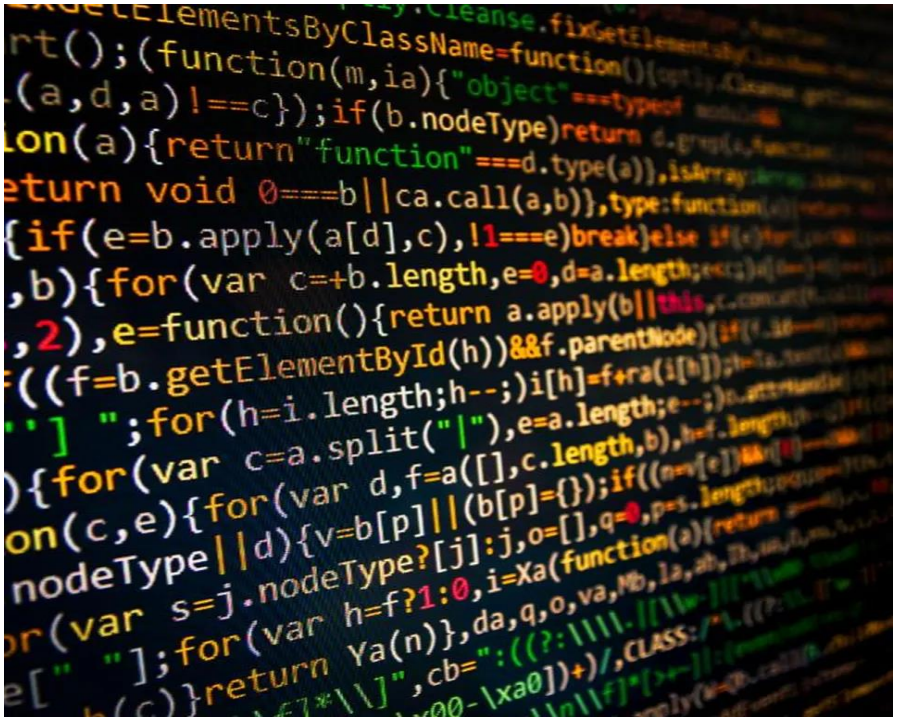
方法&贡献

- 一个新的程序分析类别: ***变量语义分析 (VUSA)***
  - 识别有特定目的的**变量**（e.g., 密码，文件句柄，GPS坐标）

- 一个新的领域特定语言: ***VarSem***

场景 - 识别 Passwords

- **"passwd"** -- 11M
- **"p_wd"** -- 9,258
- **"pass_wd"** -- 280
- **"p_w_d"** -- 164
- single letter **"p"**
- irrelevant string **"abc"**



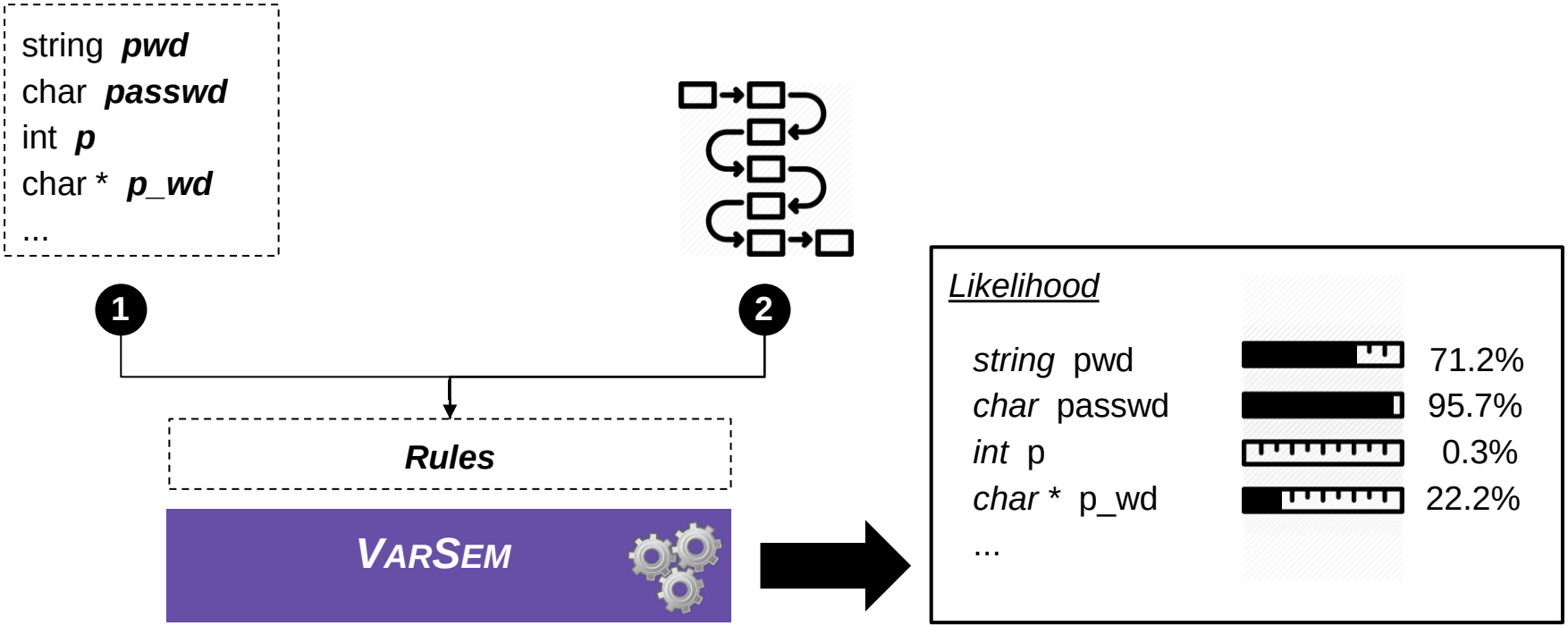
Ludwig Wittgenstein

“The meaning of a word is its use in the language”

(1) 变量名称 (i.e., 语言分析)

(2) 在程序中的用法 (i.e.,语境分析)

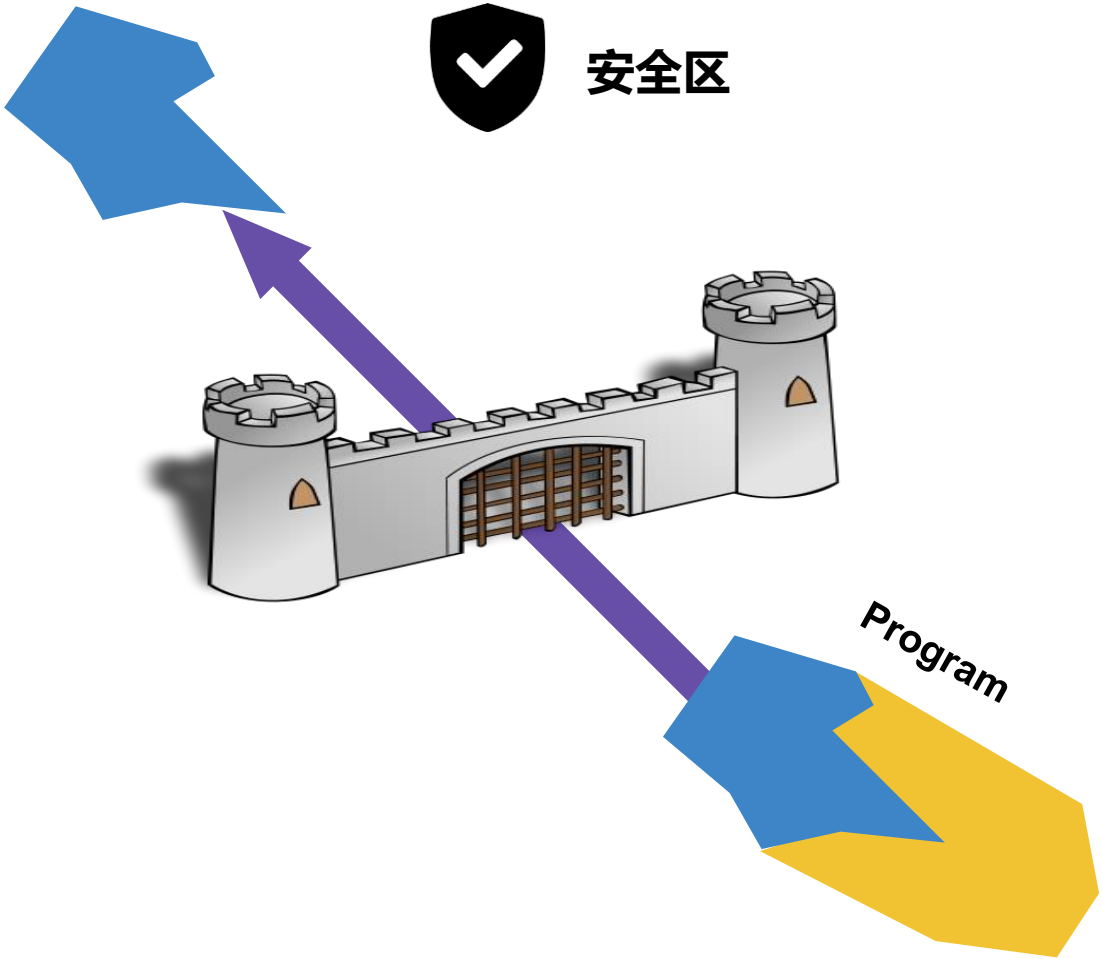
场景 - 识别 Passwords



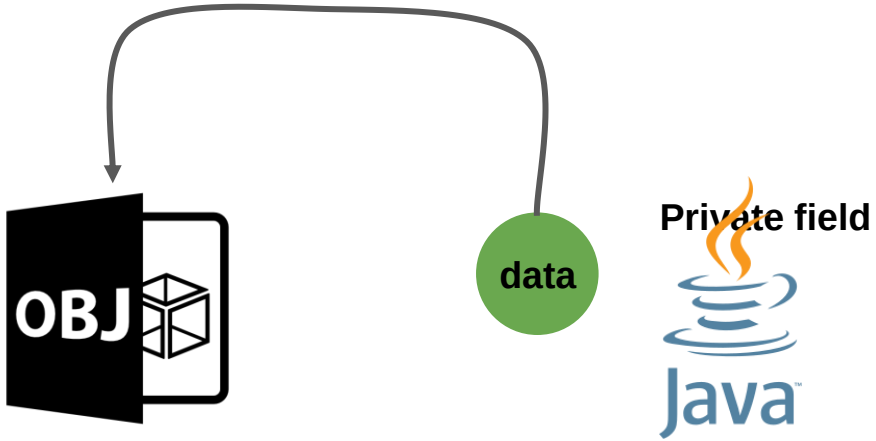
问题引入

1. 如何识别？

2. 如何保护？


安全区
Program

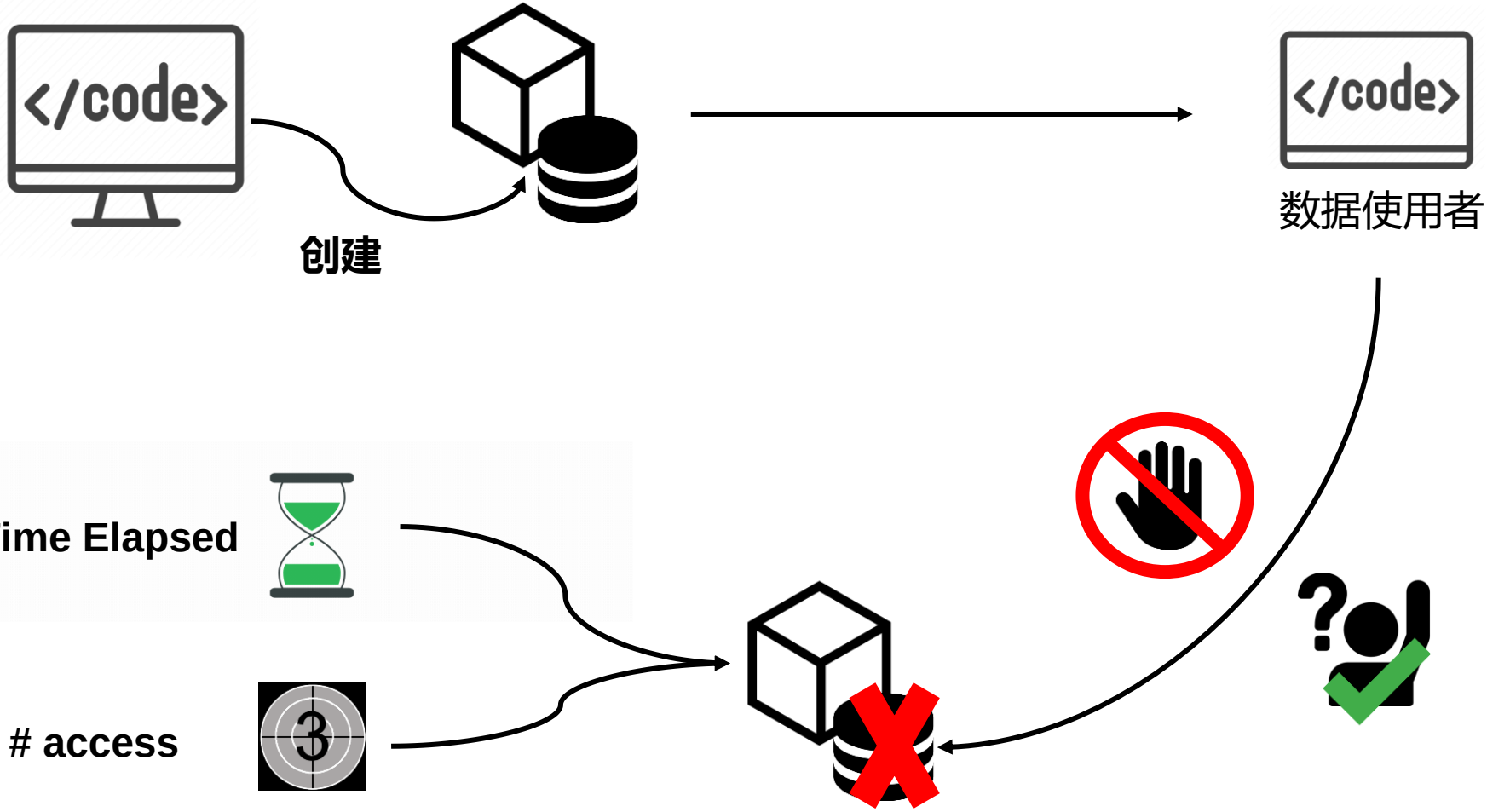
(a) 对象(Object)级别的隔离



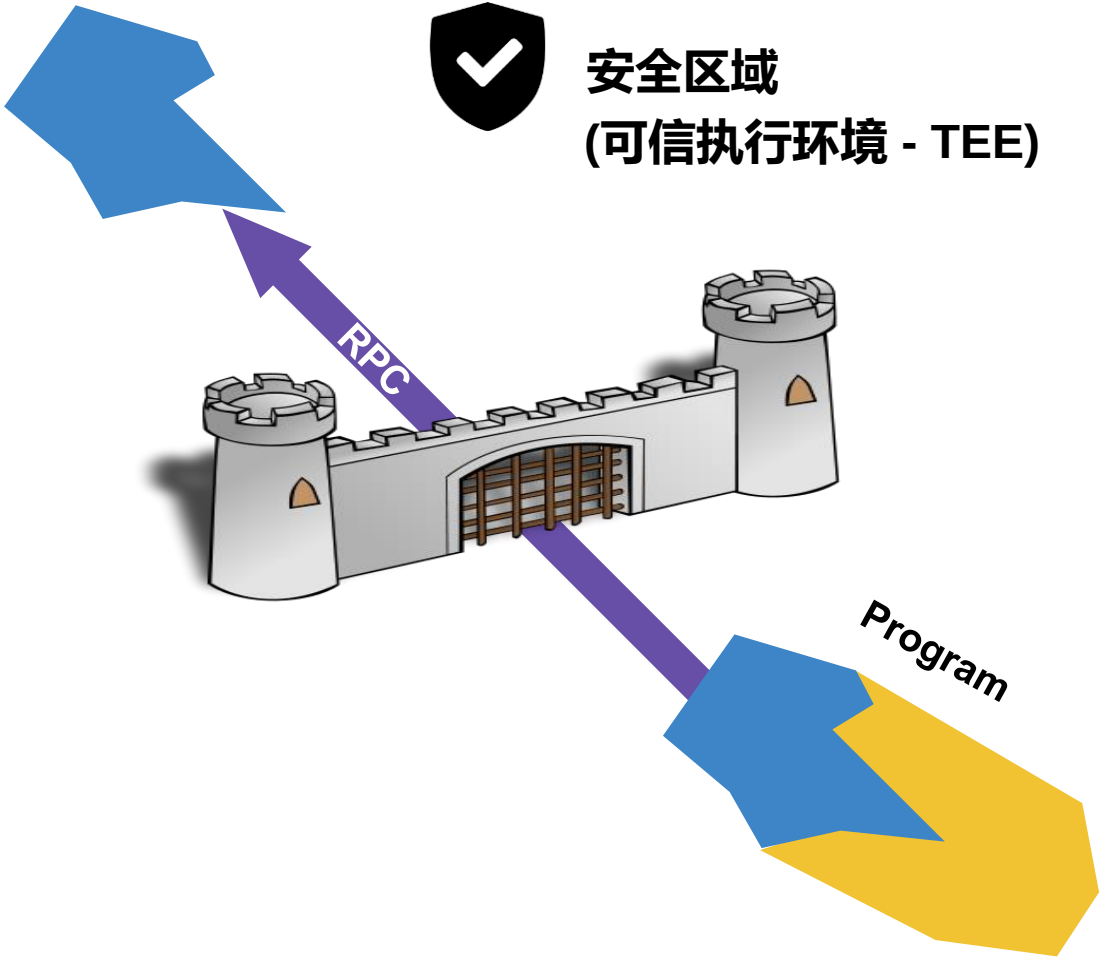
ObEx (ManLang'17)

方法 & 贡献

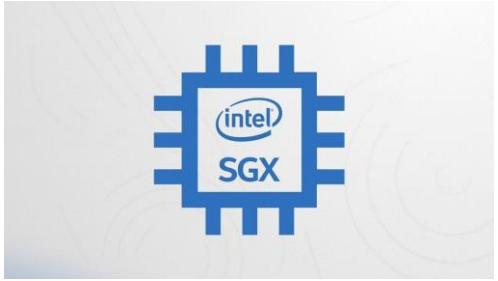
- **不可见** 但 **可查询**
  - 无法直接访问原始数据
  - 查询接口: compare, statistics (sum, average, etc..)
- **可配置的"自我清理"策略**
  - 最大访问次数Max number of accesses
  - 到期时间Time-to-expiration
- **可控的生命周期**


</code>
创建
</code>
数据使用者
Time Elapsed
access
3

(b) 函数级别的隔离



OP-TEE



RT-Trust (GPCE’18 & COLA’19)

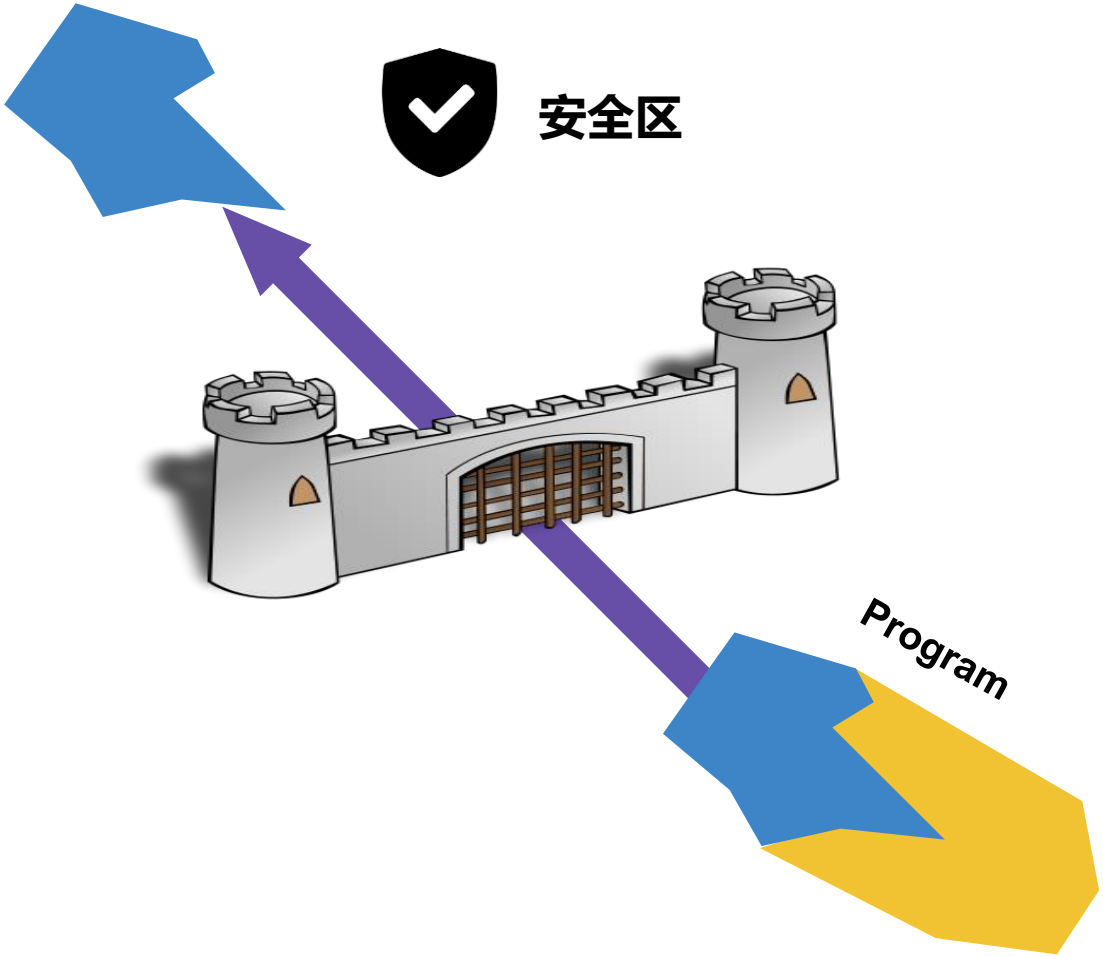
方法 & 贡献

- **针对TEE的代码自动重构**
  - 声明式的编程模型
  - 程序分析 --- 确保重构后的代码仍满足执行约束 (e.g., real-time constraints)
  - 支持C语言的重构工具
  - RPC代码生成工具

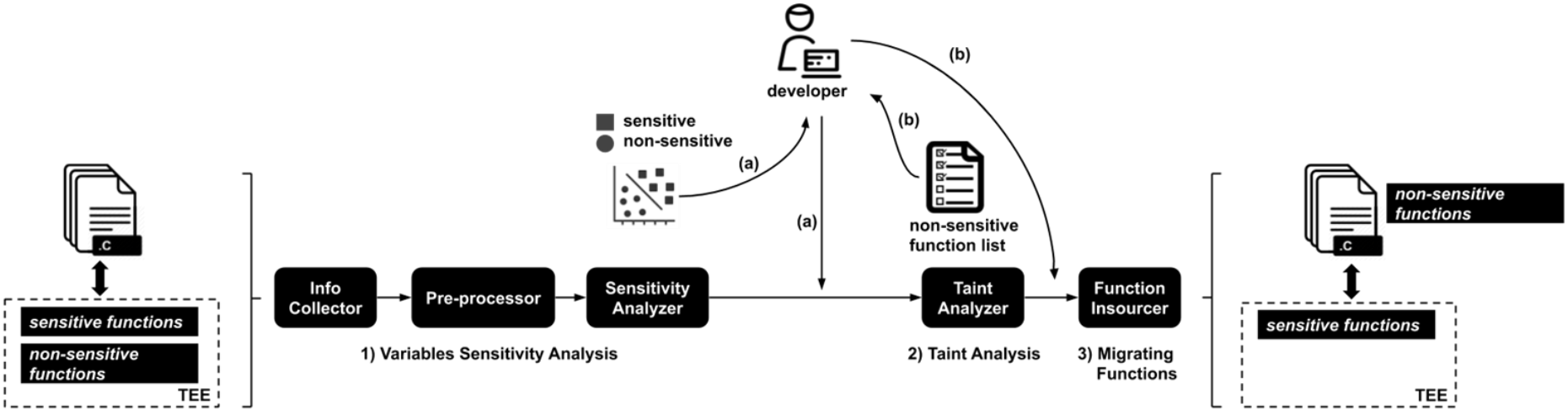
函数级别隔离的问题:

- 现有的方法(PtrSplit Liu et al. 2017; Glamdring Lind et al. 2017; RT-Trust Liu et al. 2018,2020)
  仅关注于把**函数放进安全区**

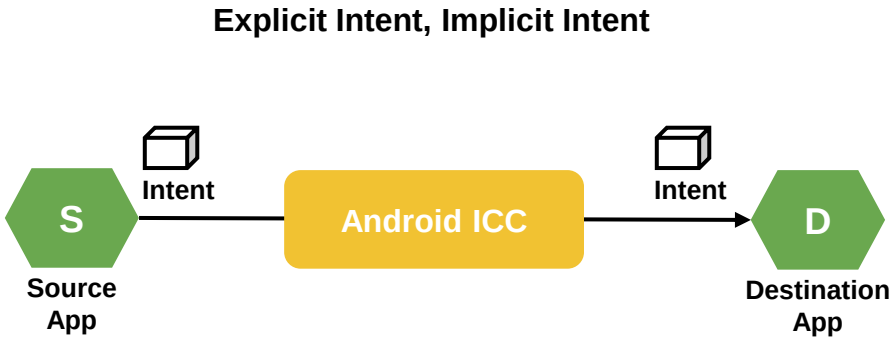
- **不需要被保护的函数**也被隔离
  - 代码执行效率下降
  - 安全隐患


安全区
Program

TEE-insourcing (TrustCom'20, JSS'22)



(c) 数据传输过程中的隐私保护



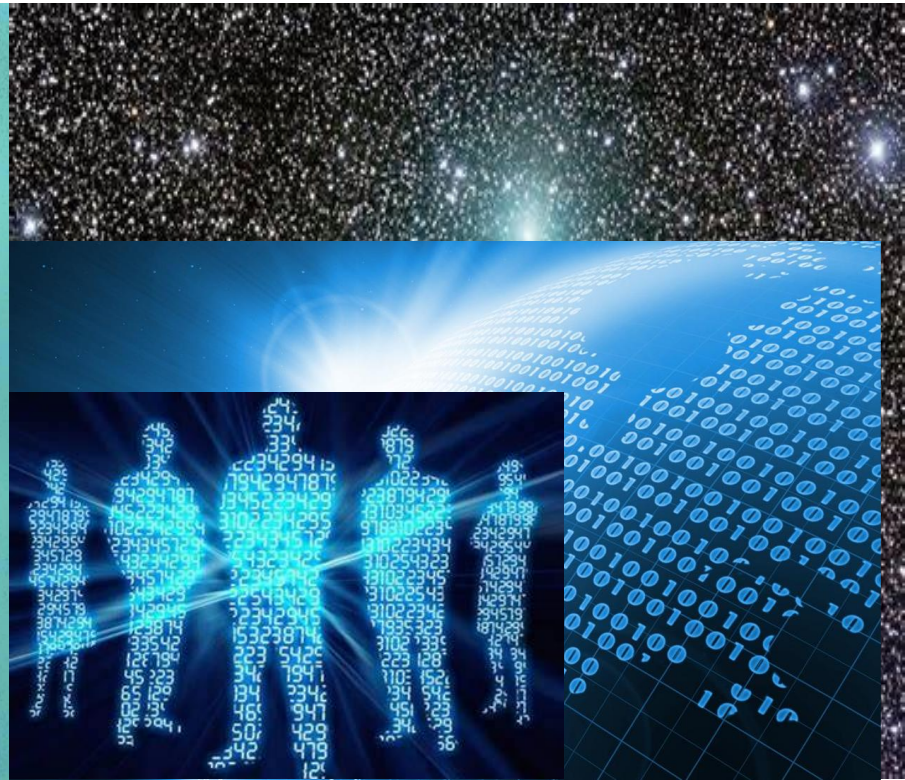
首次将同态加密引入移动计算中间件

HTPD — 一种新的**传输模型**，通过**“隐藏传输”**和**“多态发送”**增强通信的安全性

PoliCC -- HTPD的**具体实现**，取代了原有的Android ICC，用以缓解**拦截、窃听和权限升级攻击 (SecureComm’21, JSS’22)**

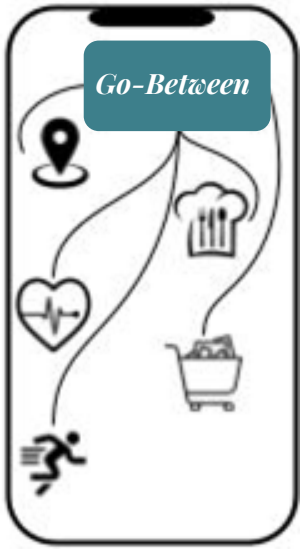
允许“不可信但非恶意”(untrusted-but-not-malicious)的 apps 操作数据

(d) 数据共享过程中的隐私保护



Sharing is NOT always easy ……

Our solution – Go-Between （MobiCASE’21）



- **On-device** 的数据共享框架，作为一个**可信中间件**，聚合各个App提供的传感器数据，对组合的数据集提供**可保护的统计查询。**

- 将**自适应的拉普拉斯噪声**添加到查询结果中，从而保护App的数据隐私。

总结

1. **什么是隐私**
   - **隐私是自然人的私人生活安宁和不愿为他人知晓的私密空间、私密活动、私密信息**
2. **什么是隐私计算**
   - **可用但不可见**
3. **隐私计算技术**
   - **秘密共享，同态加密，不经意传输，混淆电路，差分隐私，可信执行环境，联邦学习**
4. **如何与软件工程结合**
   - **如何识别隐私数据**
   - **如何保护隐私数据**

рахмат
danke
謝謝
ngiyabonga
spas
teşekkür ederim
Баярлалаа
mersi
kia ora
barka
welalin
tack
mahalo
tapadh leat
спасибо
faafetai lava
vinaka
спасибі
blagodaram
dank je
misaotra
matondo
paldies
grazzi
kiitos
dankie
nanni
nandri
dhanyavad
thank you
gracias
хвала
asante
manana
bedankt
hvala
maururu
köszönöm
aciu
dankon
akun
obrigada
tenki
murakoze
enkosi
bayarlalaa
gracie
djiere dieuf
tau
mochchakkeram
дякую
mamnun
chokrane
dziękuję
chnorakaloutioun
gratias ago
gràcies
sulpáy
taiku
go raibh maith agat
sobodi
dėkuji
sagolun
sukriya
kop khun krap
obrigado
mèsi
ありがとう
tanemirt
grazie
arigatô
takk
dakujem
trugarez
didi madloba
najis tuke
terima kasih
rahmet
diolch
dhanyavadagalu
shukriya
merce
мерси
kam sah hamnida
rahmat
ευχαριστώ
merci
তোমাকে ধন্যবাদ
감사합니다
xièxie

参考文献

[1]中华人民共和国民法典 http://www.npc.gov.cn/npc/c30834/202006/75ba6483b8344591abd07917e1d25cc8.shtml
[2]中华人民共和国个人信息保护法 http://www.npc.gov.cn/npc/c30834/202108/a8c4e3672c74491a80b53a172bb753fe.shtml
[3]年中盘点：2022年十大数据泄露事件 https://www.51cto.com/article/713157.html
[4]2021年中国大数据产业市场现状及发展趋势分析 线下场景营销成为大数据应用新机遇 https://bg.qianzhan.com/trends/detail/506/210224-8ccd0b37.html
[5]密钥 https://en.wikipedia.org/wiki/Key_(cryptography)
[6]Arvind Narayanan and Vitaly Shmatikov, Robust De-anonymization of Large Sparse Datasets https://www.cs.utexas.edu/~shmat/shmat_oak08netflix.pdf
[7][联邦学习定义]Yang, Q., Liu, Y., Cheng, Y., Kang, Y., Chen, T. and Yu, H., 2019. Federated learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, 13(3), pp.1-207.
[8][Google]Federated Learning: Collaborative Machine Learning without Centralized Training Data https://ai.googleblog.com/2017/04/federated-learning-collaborative.html
[9] 陈凯,杨强，隐私计算, 电子工业出版社
[10] Yin Liu, Shuangyi li, and Eli Tilevich, "Toward a Better Alignment Between the Research and Practice of Code Search Engines," Proceedings of the 29th Asia-Pacific Software Engineering Conference (APSEC 2022), December, 2022.
[11] Yin Liu, Breno Dantas Cruz, and Eli Tilevich, "Secure and Flexible Message-Based Communication for Mobile Apps Within and Across Devices," Journal of Systems & Software (JSS), In Print, July 2022.
[12] Yin Liu, Siddharth Dhar, and Eli Tilevich, "Only Pay for What You Need: Detecting and Removing Unnecessary TEE-Based Code," Journal of Systems & Software (JSS), In Print, February 2022.
[13] Yin Liu, Breno Dantas Cruz, and Eli Tilevich, "Privacy-Preserving Sharing of Mobile Sensor Data," Proceedings of EAI MobiCASE 2021 - 12th EAI International Conference on Mobile Computing, Applications and Services 2021 (MobiCASE 2021), November 2021.
[14] Yin Liu, Breno Dantas Cruz, and Eli Tilevich, "HTPD: Secure and Flexible Message-Based Communication for Mobile Apps," Proceedings of the 17th EAI International Conference on Security and Privacy in Communication Networks (SecureComm 2021), September 2021.
[15] Yin Liu and Eli Tilevich, "Reducing the Price of Protection: Identifying and Migrating Non-Sensitive Code in TEE," Proceedings of the 19th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom 2020), December 29, 2020 - January 1, 2021.
[16] Yin Liu and Eli Tilevich, "VarSem: Declarative Expression and Automated Inference of Variable Usage Semantics," Proceedings of the 19th International Conference on Generative Programming: Concepts & Experiences (GPCE 2020), November 2020.
[17] Yin Liu, Kijin An, and Eli Tilevich, "Automated Refactoring for Different Trusted Execution Environments under Real-Time Constraints," Journal of Computer Languages (COLA) 2019. Nominated for Best Paper Award
[18] Yin Liu, Kijin An, and Eli Tilevich, "RT-Trust: Automated Refactoring for Trusted Execution Under Real-Time Constraints," Proceedings of the 17th International Conference on Generative Programming: Concepts & Experience (GPCE 2018), November 2018. Slides Poster
[19] Yin Liu, Zheng Song, and Eli Tilevich, "Querying Invisible Objects: Supporting Data-Driven, Privacy-Preserving Distributed Applications," Proceedings of the 14th International Conference on Managed Languages & Runtimes (ManLang 2017, formerly PPPJ, now called MPLR), September 2017.