信息安全技术

1. 秘密共享

Secret-Sharing

消息分享游戏

一个 “分发者” 和两个 “参与者” Alice 和 Bob

分发者有一个消息 m

她想要将消息“分享”给两个参与者，且分享过程满足如下条件：

- 任意一个参与者都不能单独得到关于消息的任何信息
- 但是，两个参与者合作可以得到消息m

- **Bad idea:** 如果 m 是一个长度为2个比特的消息 m_1 m_2，将 m_1 分发给 Alice ，m_2 分发给Bob。

- 有没有其它的方法?

分享1个比特

为了分享一个比特 m, 分发者选取一个 服从**均匀随机分布（uniformly random）**的比特 b 并且将$a := m \oplus b$ 分发给Alice，将 b 分发给 Bob

- Bob 没有获得任何信息 (b 是一个随机比特)
- Alice也没有获得任何信息: 对于任意一个可能的m 的值 (0 或 1), a 是一个随机比特(0 w.p. ½, 1 w.p. ½)
  - 其视角（View）与消息m 独立

$m = 0 \rightarrow (a, b) = (0,0)\ or\ (1,1)$
$m = 1 \rightarrow (a, b) = (1,0)\ or\ (0,1)$

- 两人合作容易恢复出消息 $m = a \oplus b$

多个比特可以被独立分享: e.g., $m_1 m_2 = a_1 a_2 \oplus b_1 b_2$

Note: 任意一个share 都可以在得知消息之前被选定 [why?]

秘密vs保密

- 消息 m 真的是一个秘密吗?
- Alice 或者 Bob 能够以½的概率猜到比特m的值
  - 如果他们实现已经获得一些关于m的信息，那么他他们猜对的概率会更大（这里并没有要求消息m服从均匀随机分布）
- 如何定义保密性（ preserving secrecy ）?
  - Share没有向任意一方泄露任何额外的信息
- 保密性是密码学中的一个典型设计目标

如何定义保密性

目标: 对Alice和Bob来说看到share之前和看到share之后，他们关于消息m所知道的信息没有发生变化

- 在得到share之前所知道的信息:
  - 消息的概率分布
  - 即，对于任意一个消息m, $\Pr[msg = m]$
- 在得到share之后所知道的消息($a.k.a.$ 视角，view)
  - 我们使用v表示视角. 那么概率分布为: $\Pr[msg = m \mid view = v]$

- 形式化的定义: $\forall$ possible v, $\forall$ m, $\Pr[ms = m|view = v] = \Pr[msg = m]$
  - i.e., 视角和消息是独立的
  - $\forall\ v, \forall\ m, \Pr[view = v, msg = m] = \Pr[view = v] \cdot \Pr[msg = m]$

如何定义保密性

对Alice和Bob来说看到share之前和看到share之后，他们关于消息m所知道的信息没有发生变化:

- $\forall$ possible v, $\forall$ m, $\Pr[msg = m \mid view = v] = \Pr[msg = m]$
- $\forall$ v, $\forall$ m, $\Pr[view = v, msg = m] = \Pr[view = v] \cdot \Pr[msg = m]$
- $\forall$ v, $\forall$ possible m, $\Pr[view = v \mid msg = m] = \Pr[view = v]$

因方案而异

$\forall$ v, $\forall$ possible m, m', $\Pr[view = v \mid msg = m] = \Pr[view = v \mid msg = m']$

与消息的分布无关

- i.e., 对于任意的消息m, 其视角（view）分布是一致的
- 这一视角可以在没有得到消息之前被模拟（simulate）

重要: $\Pr[msg = m \mid view = v] = \Pr[msg = m' \mid view = v]$ 不一定成立

为什么?

秘密共享

更一般化的秘密共享

- 允许多个参与者，如何实现？
- 具有权限的参与者的子集能够重构出秘密(not necessarily just the entire set of parties)

应用广泛：

- 直接应用(distributed storage of data or keys)

- 其它密码算法构造的重要组件(component)
  - 增强多种密码原语的机密性
  - 安全多方计算
  - 属性基加密
  - 抗泄露技术...

门限秘密共享

(n,t)-秘密共享

- 将一个消息 m 分成 n 个share $s_1,..,s_n$，使得对于任意的 t 个 shares 都可以重构出消息m
- 任意的t-1个或更少 shares 不能获得关于消息m 的任何信息

e.g., 对于消息空间中任意的 m 来说$(s_1,\dots,s_{t-1})$ 的分布是一致的

之前的例子是一个 (2,2)-秘密共享

门限秘密共享

- **构造: (n,n)-秘密共享**

additive secret sharing

- Message-space = share-space = ***G***, 是一个有限域

  e.g. $G = Z_2$ (group of bits, with xor as the group operation)

  or, $G = Z_2^d$ (group of d-bit strings)

  or, $G = Z_p$ (group of integers *mod p*)

- Share(m):

  利用均匀采样随机选取 $(s_1, \ldots, s_{n-1}) \in G^{n-1}$

  令 $s_n = -(s_1 + .. + s_{n-1}) + m$

- Reconstruct$(s_1, .., s_n)$: $m = s_1 + .. + s_n$

Claim: 这是一个(n,n) 秘密共享方案 [Why?]

PROOF

Additive Secret-Sharing

Share(m):

从$|G|^{n-1}$中服从均匀分布随机选取$(s_1, \ldots, s_{n-1})$

令 $m = s_1 + .. + s_n$

Claim: n-1 shares 不能获得关于m的任何信息

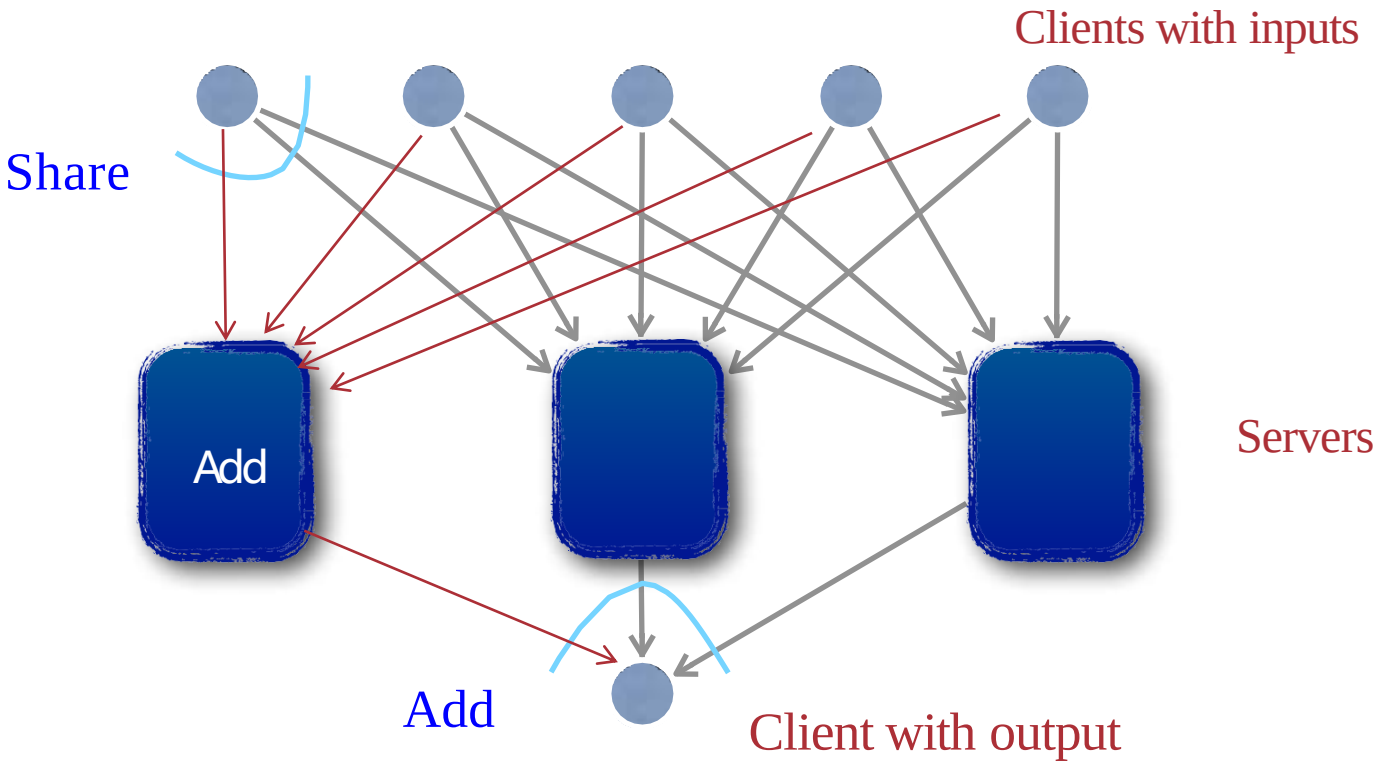
Proof: 令 $T \subseteq \{1, .., n\}, |T| = n - 1$. 需要证明无论m是什么，$\{s_i\}_{i \in T}$ 的分布是一致的 (实时上是均匀分布).

- 出于正确性，考虑 $T = \{2, .., n\}$. 给定G中的任意一个 (n-1)-元组 即 $(g_1, .., g_{n-1}) \in G^{n-1}$. 需要证明Pr[$(s_2, .., s_n) = (g_1, .., g_{n-1})$] 对于所有的m成立.
- 给定任意的 m.
- $(s_2, .., s_n) = (g_1, .., g_{n-1}) \Leftrightarrow (s_2, .., s_{n-1}) = (g_1, .., g_{n-2}) \wedge s_1 = m - (g_1 + \cdots + g_{n-1})$.
- 因此 $\Pr[(s_2, .., s_n) = (g_1, .., g_{n-1})] = \Pr[(s_1, .., s_n) = (a, g_1, .., g_{n-2})]$ 其中 $a := m - (g_1 + \cdots + g_{n-1})$
- 由于 $(s_1, .., s_{n-1})$ 是从$|G|^{n-1}$中服从均匀分布随机选取的， $\Pr[(s_1, .., s_n) = (a, g_1, .., g_{n-2})] = \frac{1}{|G|^{n-1}}$,
- 因此 $\Pr[(s_2, .., s_n) = (g_1, .., g_{n-1})] = \frac{1}{|G|^{n-1}}$, 与 m 无关.

□

一个应用

“私有求和” 协议



只需要有一个服务器不参与合谋，其它的用户和服务器进行合谋都无法获得除了他们本身已经拥有的输入输出以外的任何信息

门限秘密共享

(n, 2)-秘密共享的构造

消息空间= share 空间= **F**, 一个域(e.g.整数对p 取模， p为素数)

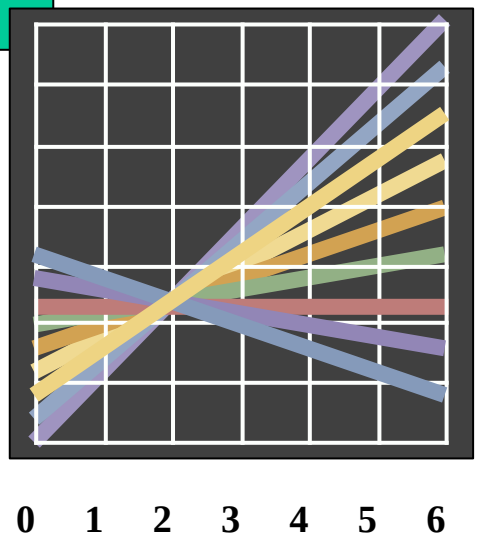
- Share(m): $r \xleftarrow{R} \boldsymbol{F}$. 令 $s_i = r \cdot a_i + m \, (for \, i = 1, \ldots, n < |F|)$
- Reconstruct(s_i, s_j): $r = \frac{s_i - s_j}{a_i - a_j}; m = s_i - r \cdot a_i$
- 每一个 s_i 都服从均匀分布, 与 m 无关 [Why?]

a_i are n distinct, non-zero field elements

Since a_i^{-1}exists, exactly one solution for $r \cdot ai + m = d$, for every value of d

“几何” 解释

- Share的过程随机选取了一条“直线” y = f(x)，满足$f(0) = m$. Shares $s\_i = f(a\_i)$.
- s_i 与 m 独立: 对于任意的m'，都存在一条直线同时经过点 (a_i, s_i)， $(0, m')$
- 但是给定两个点，直线被唯一确定!



PROOF

(n, 2) 秘密共享证明

Share(m): r $\xleftarrow{R}$ **F**. 令 $s_i = r \cdot a_i + m\ (for\ i = 1, \ldots, n < |F|)$

Claim: 任意的share 都不包含关于 m 的任何信息

Proof: 对于任意的$i \in \{1, .., n\}$ 需要证明s_i 的分布是一致的 (事实上，都是均匀分布) 与m是什么无关.

考虑任意$g \in \boldsymbol{F}$. 需要证明Pr[$s_i = g$] 与 m 是独立的.

给定任意的m.

对于任意的$g \in F, s_i = g \Leftrightarrow r \cdot a_i + m = g \Leftrightarrow r = (g - m) \cdot a_i^{-1}$ (由于 $a_i \neq 0$)

由于 r 是按照均匀分布随机选取的，因此有

$$\Pr[\, s_i = g] = \Pr[r=(g - m) \cdot a_i^{-1}] = 1/|F| \quad \square$$

门限秘密共享

域F上的（n,t）秘密共享

Shamir Secret-Sharing

- 几何/代数 视角下的一般化（generalize）: 使用多项式代替直线

Share(m): 随机选取一个阶为 t-1 多项式 f(X), 满足f(0) = m.

Share 为 $s_i = f(a_i)$.

- 满足f(0) = m的随机多项式的选取: 令 $c_0 = m$ 并随机选取 $c_1, .., c_{t-1} \overset{R}{\leftarrow} \mathbf{F}$,

令$f(X) = c_0 + c_1X + c_2X^2 + .. + c_{t-1}X^{t-1}$

Reconstruct$(s_1, .., s_t)$:拉格朗日插值找到 $m = c_0$

- 需要 **t** 个点来重构多项式。给定**t-1**个点，$|\boldsymbol{F}|^{t-1}$上的多项式中，恰有一个多项式经过点 $(0, m')$。

拉格朗日插值

- 给定阶为 *t-1* 的多项式(元素个数大于t的域上的单变量多项式),上t个 不同的点重新构造出整个多项式 (i.e., 找到 t 个系数)
  - t 个变量: $c_0, .., c_{t-1}$.
  - t 个等式: $1.c_0 + a_i.c_1 + a_i^2.c_2 + ..a_i^{t-1}.c_{t-1} = s\_i$
  - 一个线性系统: $Wc = s$, 其中 W 是一个 $t \times t$ 矩阵，其第i行为 $W_i = (1\ a_i\ a_i^2\ \ldots\ a_i^{t-1})$
  - W (范德蒙德矩阵) 是可逆的
  - $c = W^{-1}s$

总结

- 保密性: 视角和消息是独立的
  - i.e., $\forall view, \forall msg1, msg2, \Pr[view|msg_1] = \Pr[view|msg_2]$
  - 视角不会带来关于消息m的额外信息，（与事先知道的信息相比）
  - 视角在不知道消息的情况下也可以被模拟
  - 即便攻击者的计算能力是无限的，方案仍然具有机密性
- 实现了加法秘密共享和门限秘密共享
- 对于其它的问题（原语），这样的机密性不一定总是能达到 (e.g., 敌手计算能力不受限的条件下，不存在公钥加密方案)

作业

- Consider the following secret-sharing scheme
  - Message space = { buy, sell, wait }
  - buy → (00,00), (01,01), (10,10) or (11,11) w/ prob 1/4 each
  - sell → (00,01), (01,00), (10,11) or (11,10) w/ prob 1/4 each
  - wait → (00,10), (01,11), (10,00), (11,01), (00,11), (01,10), (10,01) or (11,00) w/ prob 1/8 each
  - Reconstruction: Let $\beta_1\beta_2 = \text{share}_{\text{Alice}} \oplus \text{share}_{\text{Bob}}$. Map $\beta_1\beta_2$ as follows: 00 → buy, 01 → sell, 10 or 11 → wait
- Is it secure?