

密码学协议动物园快速游览

A Quick Tour of Cryptographic Protocols Zoo

哈尔滨工业大学

张宇

2024春

动物园地图



协议（动物是什么？）

- ❑ 通信协议是为了一个特定目的的数字消息格式与交换规则的形式化描述
 - ❑ 协议之于通信，如算法之于计算
 - ❑ 每个人必须知道并同意服从协议
- ❑ 无歧义：每个步骤必须被明确定义且无误解的可能
- ❑ 完备性：对每个可能的情况都必须有一个明确的行为
- ❑ 密码学协议：除了上述属性，还不可能比协议中说明的做的更多或者知道的更多

协议类型与攻击

- ❑ 仲裁协议：一个仲裁者是一个公正的可信第三方，帮助完成协议
- ❑ 审判协议：一个法官是也是一个公正的可信第三方。与仲裁者不同，其不直接参与协议，而是来审判协议是否正确执行
- ❑ 自强制协议：最佳的协议类型。协议本身保证公平性。
 - ❑ 例子：两人平分蛋糕协议。先分蛋糕的人后选。
- ❑ 对协议的攻击
 - ❑ 被动攻击：攻击者不影响协议，例如窃听
 - ❑ 主动攻击：攻击者更改协议以获得优势
 - ❑ 作弊者：攻击者是协议中的一方
 - ❑ 被动作弊者：按照协议执行，但试图获得比协议所设定的更多的信息
 - ❑ 主动作弊者：在协议进程中干扰协议来作弊

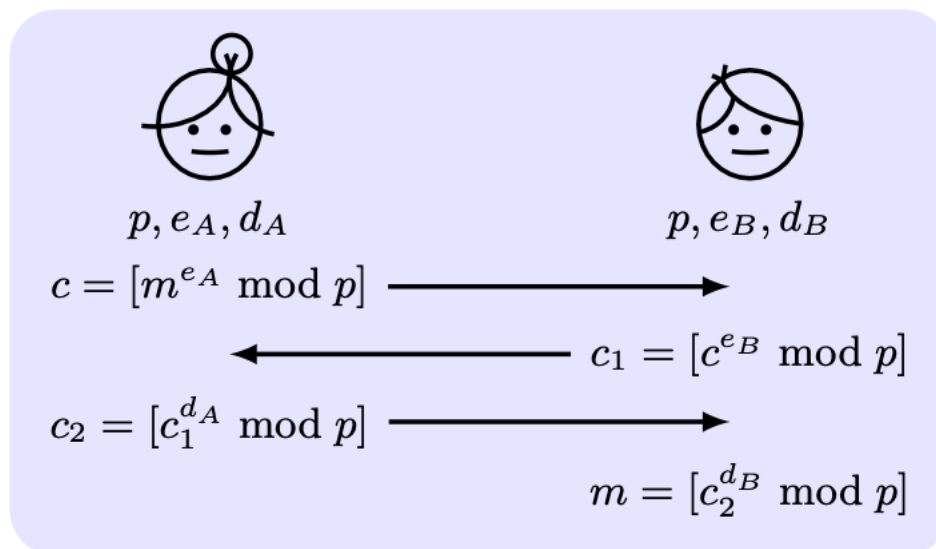
三次传递协议 (Three-Pass)

❑ 目的：两方之间无共享密钥下的保密通信

❑ 类比：两人同一个箱子来传递一个秘密，该箱子可以上锁

Requirement: $\text{Dec}_{k_1}(\text{Enc}_{k_2}(\text{Enc}_{k_1}(m))) = \text{Enc}_{k_2}(m)$

Shamir Protocol: p is a prime, find e, d with $\gcd(e, p-1) = 1$
and $ed \equiv 1 \pmod{p-1}$



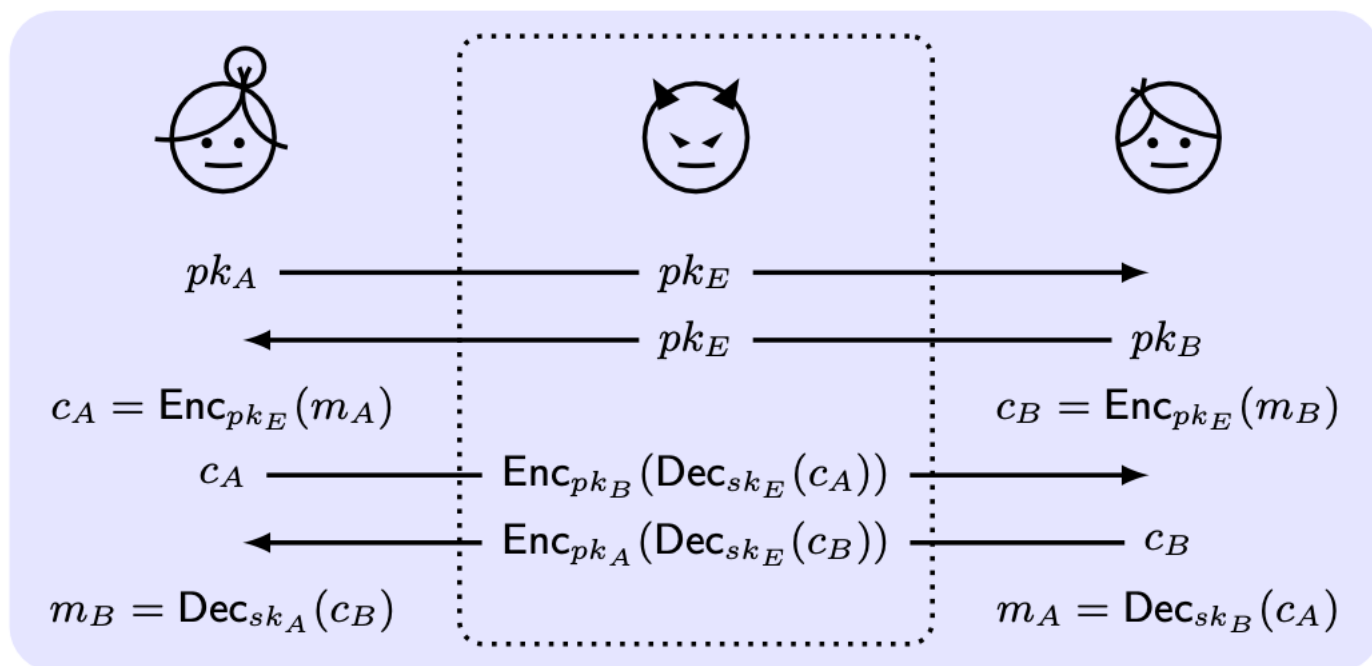
$$c_2^{d_B} = c_1^{d_A \cdot d_B} = c^{e_B \cdot d_A \cdot d_B} = m^{e_A \cdot e_B \cdot d_A \cdot d_B} = m^{e_A d_A \cdot e_B d_B} = m$$

Weakness: insecurity under the man-in-the-middle attack

中间人攻击 (The Man-In-The-Middle Attack)

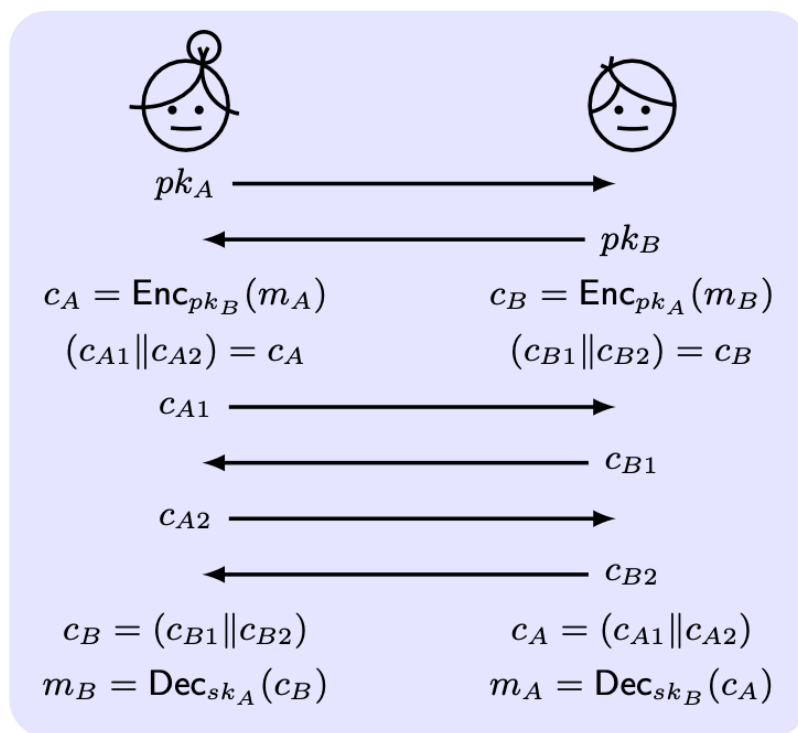
❑ 中间人攻击，也叫水桶小队攻击。

❑ Alice并不能确定和其通信的真的是Bob本人，Bob也不能确定对方是Alice。中间人攻击可以伪装成双方，与双方分别进行协议，与双方分别传递一个秘密，而双方并不知情。中间人可以获得Alice发给Bob的秘密，也可以伪造一个秘密发给Bob。



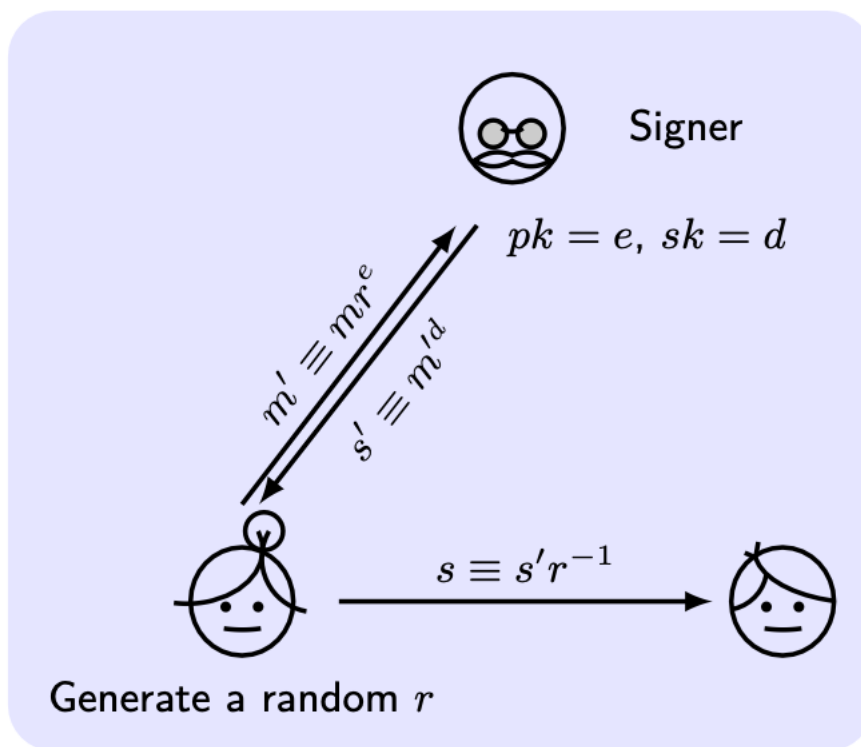
互锁协议 (Interlock)

- ❑ 一种抵御中间人攻击的方法，并不需要对双方身份进行鉴别。
- ❑ 这是由Ron Rivest和Adi Shamir提出的，思路是将两个要交换的密文分成两部分，分别先交换密文的一半，然后再交换另一半。
- ❑ 敌手收到一半密文后，因为没有得到整个密文，无法用自己的密钥解密原本的明文，无法传递密文的一半。如果敌手自己产生一个明文，加密并发送，那么就无法最终令诚实方收到原本的消息。与其他身份鉴别方案结合，可以发现攻击者。



盲签名 (Blind Signature)

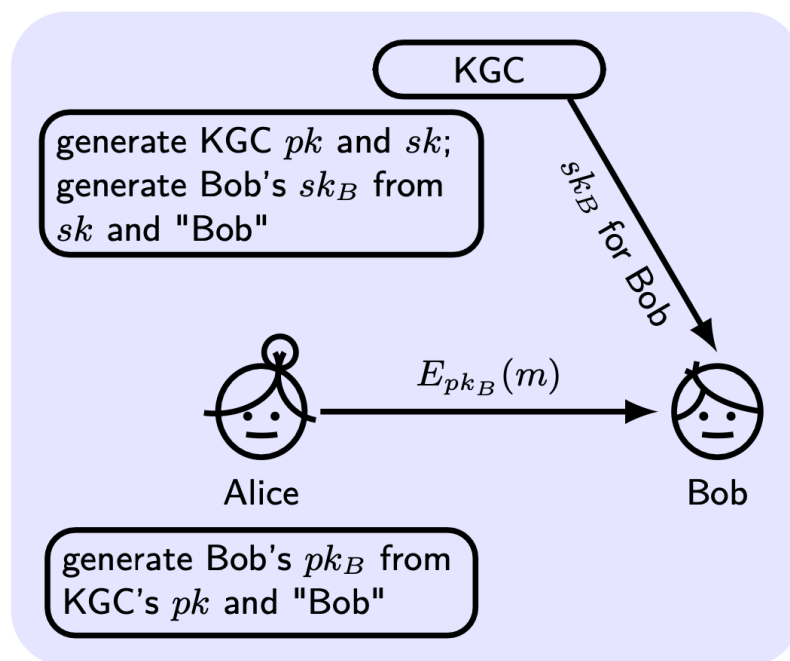
- 签名者在看不见消息的情况下对消息签名；
- 类比隔着一个信封对一个文件盖一个钢印，然后打开信封，文件上有钢印；
- Chaum的盲签名方案：



$$s \equiv s'r^{-1} \equiv m'^d r^{-1} \equiv (mr^e)^d r^{-1} \equiv m^d.$$

基于身份的加密 (IBE)

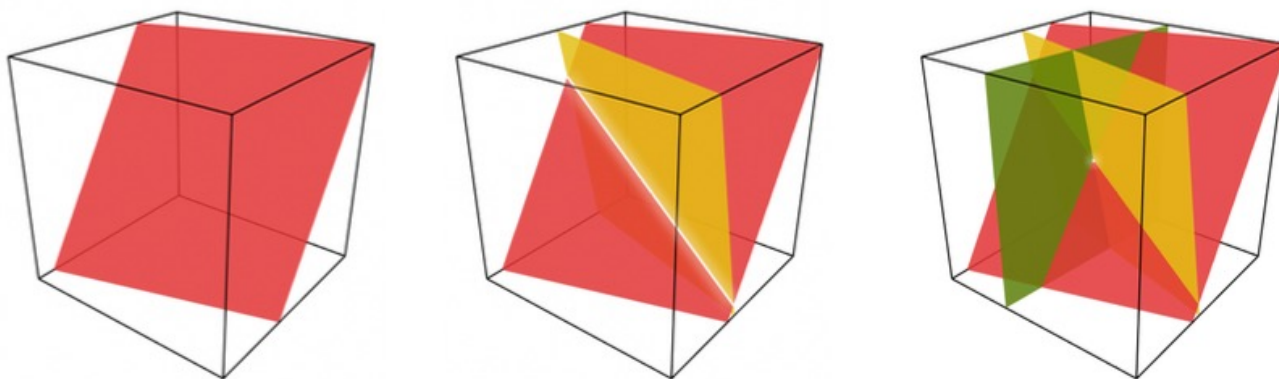
- ❑ IBE: 不使用数字证书来实现公钥分发, 直接用接收方的ID作为公钥, 例如, 直接用接收方的email地址作为其公钥。需要一个可信第三方来协助, 即密钥生成中心KGC;
- ❑ 接收方从KGC获得自己私钥; 发送方需要预先获得KGC的公钥, 但不再需要接收方的数字证书。
- ❑ 优点: KGC在生成用户的私钥后可以被去掉, 不需要PKI来分发密钥
- ❑ 弱点: 单点失效, 隐式的密钥托管



秘密分享 (Secret Sharing)

- 一个秘密在一组人中共享，每个人持有秘密的一部分，但当手里的秘密的份数没有达到某个阈值的时候，没有人能还原秘密；而当秘密的份数达到了某个阈值时，可以还原出秘密。
- 例子，一个三维空间中的一个点，可以被分解为三个面；

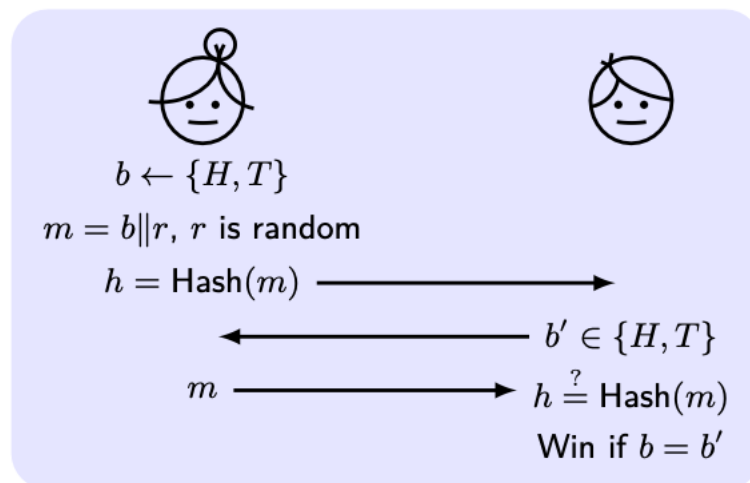
Blakley's scheme: any n nonparallel n -dimensional hyperplanes intersect at a specific point.



- 例子，中国剩余定理中将秘密分解为各个素数的群中元素；

承诺方案 (Commitment Scheme)

- ❑ 互联网上掷硬币：利用哈希函数实现对承诺的绑定 (binding)，即信息和承诺一一对应，承诺后不能改变信息；和隐藏 (hiding)，即承诺本身不泄漏信息；
- ❑ 掷硬币并对结果做出承诺：随机选择一个比特 b 为掷硬币结果，将 $h = \text{Hash}(b \parallel r)$ 作为承诺发送给对方；其中， r 为随机串；这个承诺具有绑定和隐藏的功能；
- ❑ 收到承诺的一方给出自己猜测的结果；此时，仍不知道实际结果，但条件是哈希函数需要隐藏信息；
- ❑ 掷硬币一方揭示结果，由于抗碰撞性质，只能揭示 $b \parallel r$ ，否则会被对方利用收到的承诺来识破；



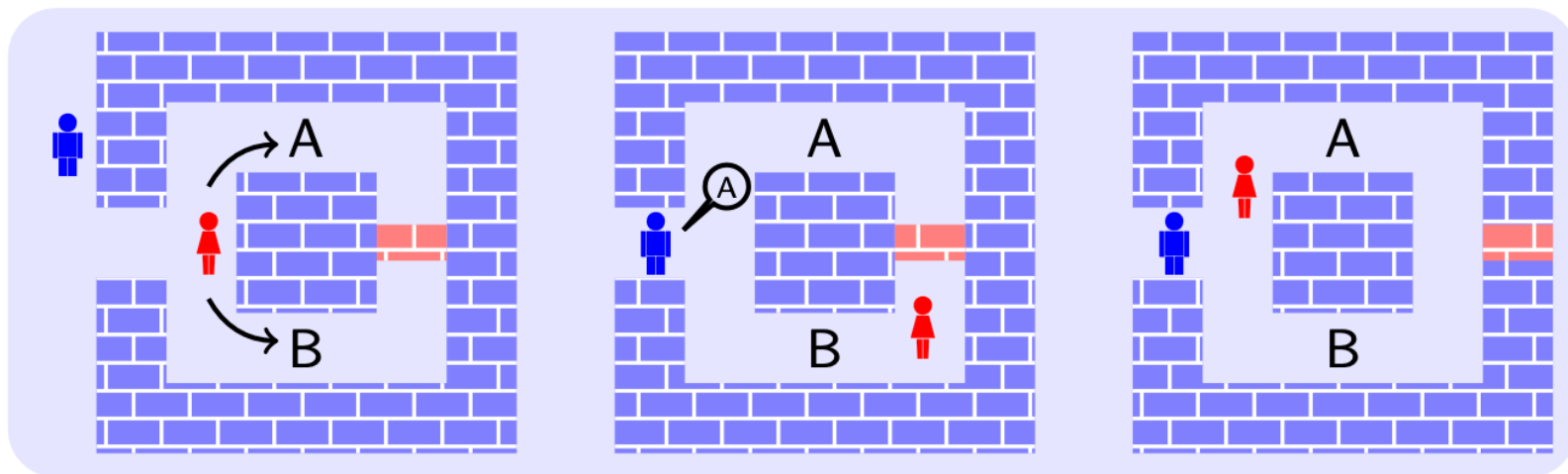
哈希函数H只抗碰撞满足安全要求吗?

零知识证明 (Zero-Knowledge Proof)

- 一种交互式证明，其中证明方成功说服验证方：证明方知道某事，但同时除了该陈述外，不泄漏任何其他信息
- 完备性：如果陈述是真的，那么诚实的验证方可以被诚实的证明方说服
- 有效性：如果陈述是假的，那么没有作弊的证明方可以说服诚实的验证方
- 存在性：如果单项函数存在，则存在对任意NP问题的零知识证明
- 西格玛协议：分三轮：声明（承诺），挑战，响应

ZKP的玩具例子

- 有一个环形山洞，山洞有一个入口，从A和B两条路可以进入洞内；在内部有一个魔法门，魔法门可以用一个咒语开启



- 为什么这是零知识证明？

- 表面的原因：这个游戏中对Bob有意义的唯一知识——魔法门咒语，Bob始终不知道
- 更本质的原因：知道魔法门咒语和读心术（Alice预知验证者Bob给出挑战）之间不可区分

汉弥尔顿环路的零知识证明

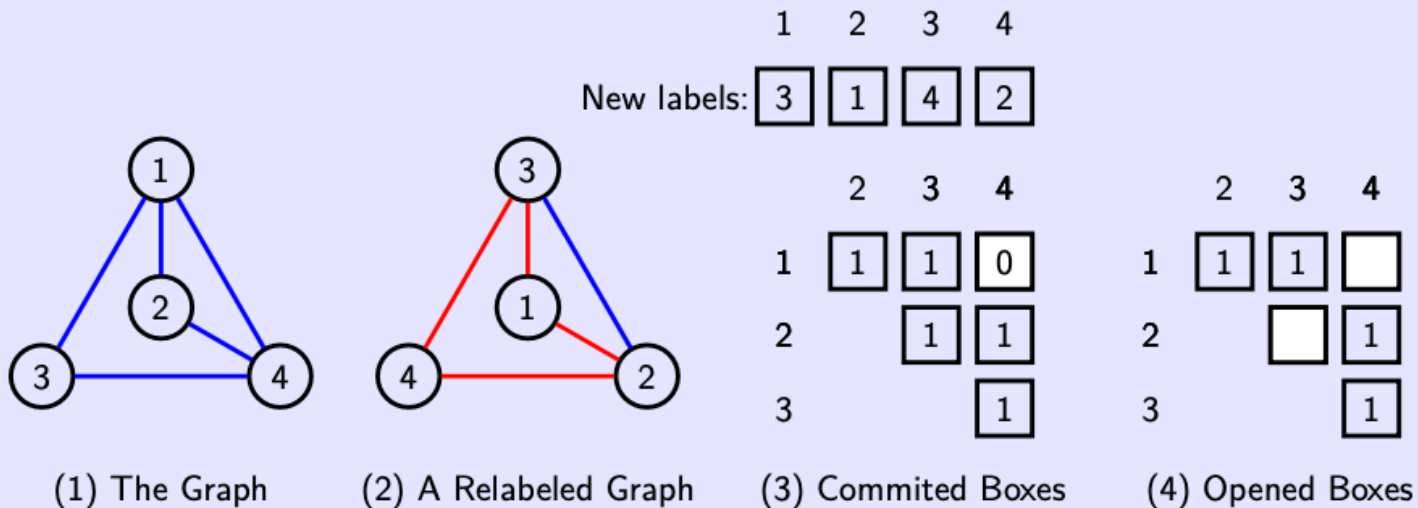
- ❑ 汉弥尔顿环路是一个NPC问题：给定一个图，给出一个经过所有节点一次的环路。证明者知道一个图的汉弥尔顿环路；
- ❑ 声明（承诺）：首先，证明者将图重新做标记：将节点重新编号，并构造邻接矩阵（行列表示节点，两点之间有连接时置1，否则置0）；将原节点编号和新编号对应关系（ N 个箱子， N 为图中节点数量）以及新邻接矩阵（ $N*(N-1)/2$ 个箱子）加密，全部发送给验证者
- ❑ 挑战：验证者从两个挑战问题中随机选择一个，一是打开被加密消息中所有箱子，以揭示其与原图是同一个图；二是打开被加密的邻接矩阵中一个汉弥尔顿环路，但不打开原节点编号和新编号对应关系的箱子
 - ❑ 第一个挑战可以令验证者确认证明者的确对图做重新标记
 - ❑ 第二个挑战可以令验证者确认证明者的确知道一个环路，但泄漏给验证者答案
- ❑ 响应：证明者根据挑战，或者揭示所有箱子，或者揭示一个汉弥尔顿环路
- ❑ 零知识：知道汉弥尔顿环路和预知挑战问题之间不可区分

汉弥尔顿环路的零知识证明 (英文)

ZKP for a solution of Hamilton Cycle (NPC). [Blum (1986)]

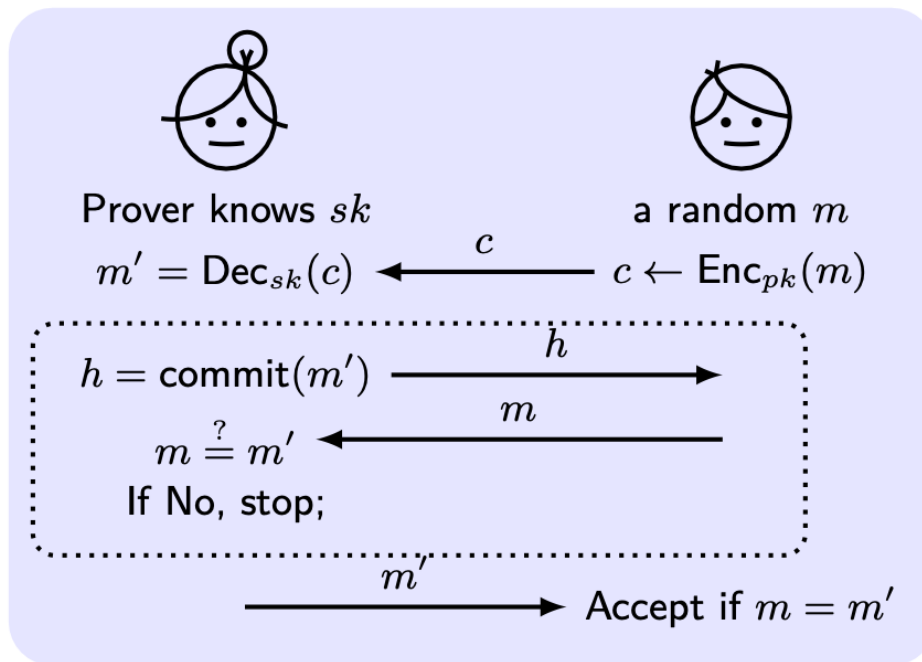
Prover relabels the graph (1) randomly, encrypts the randomly relabeld graph (2) with $N + N * (N - 1)/2$ boxes (3), and sends them to verifier.

Verifier asks only one question: either (a) show the relabelled graph is valid by opening all boxes (3); or (b) show one Hamilton cycle by opening the boxes on the cycle (4).



ZKP与承诺

- ❑ 模拟范式：当一件事Y本来就可以从X得到，那么通过Y并不会从X额外获得什么；这个范式用于保证验证者不会通过证明过程额外知道其他知识；
- ❑ 在关于是否知道RSA私钥的零知识证明中，验证者给一个密文C后，让证明者给出对应明文M，来验证证明者知道私钥
 - ❑ 当没有承诺协议时（无虚线框内协议交互），验证者可能在不知道明文M时直接给出一个密文C，而证明者返回的消息M令验证者额外知道了M；
 - ❑ 当加入承诺后时，证明者在给出M之前，先给出对M的承诺，即不泄漏M，又对后面给出的M作出承诺；在验证者提供M后，证明者知道验证者已经知道M了，根据上面的模拟范式可知，之后验证者获得的M对于验证者也不是新信息。



健忘传输 (Oblivious Transfer)

- ❑ 健忘传输：发送者不知道信息是否被传递
- ❑ 社会学家百万富翁问题：判断两个数（各自的工资）是否相同，但不暴露工资（如果两人相同，则知道对方工资）

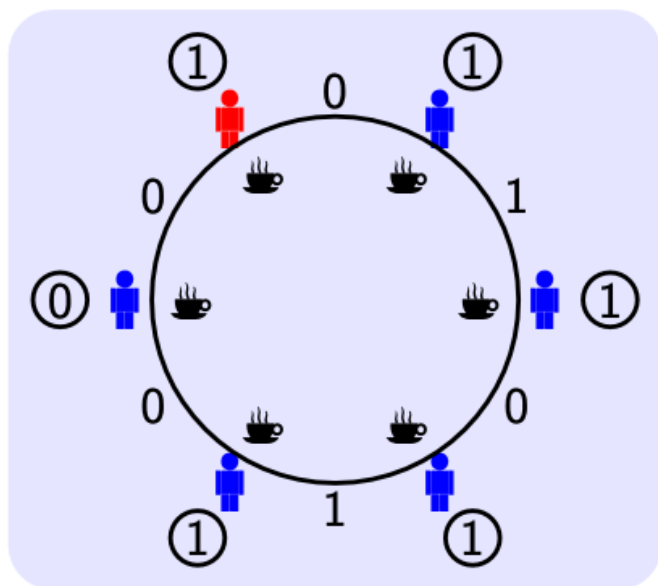





- 1 Bob prepares 4 lockable suggestion boxes marked w/ salaries.
- 2 Bob destroys the keys except for the box marked w/ his salary.
- 3 Alice puts a paper "YES" into the box marked w/ her salary, "NO" for the others.
- 4 Bob open the box and may (or may not) share the paper with Alice.

安全多方计算 (Secure Multi-Party Computation)

- ❑ 一群人用大家的输入共同计算一个函数，但保留各自输入的隐私
- ❑ 密码学家午餐问题：一群密码学家在饭后判断是否有人买单，但不知道每个人是否买单；至多有一个人买单，如果买单，则输入为1，否则为0，这是一个布尔或的多方安全计算问题

Dining Cryptographers Problem: how to perform a secure MPC of the boolean-OR function [David Chaum (1988)]



- at most one  (1), other  (0)
- every two adjacent people establish a shared one-bit secret
- everyone shouts the XOR of two shared secrets and its own bit
- output the XOR of all of what everyone shouts. If 1, there is a , otherwise there is none

同态加密 (Homomorphic Encryption)

- ❑ 同态加密：两个密文操作后，得到新密文；新密文解密后得到对应两个明文操作后的结果，即
- ❑ Elgamal加密方案是乘法的同态加密方案
- ❑ Paillier加密方案是加法的同态加密方案
- ❑ 应用：投票，计票，但不暴露投票内容
- ❑ 首个支持加法和乘法的完全同态加密方案在2009年由Craig Gentry提出

- **Homomorphic Encryption** with \circ : $\text{Dec}_{sk}(c_1 \circ c_2) = m_1 \circ m_2$.
- Elgamal encryption is homomorphic with \times :
 $\langle g^{y_1}, h^{y_1} \cdot m_1 \rangle \cdot \langle g^{y_2}, h^{y_2} \cdot m_2 \rangle = \langle g^{y_1+y_2}, h^{y_1+y_2} \cdot m_1 m_2 \rangle$
- Paillier scheme is homomorphic with $+$:
 $\text{Enc}_N(m_1) \cdot \text{Enc}_N(m_2) = \text{Enc}_N([m_1 + m_2 \bmod N])$.
- **Application**: voting without learning any individual votes.

$$c_i := [(1 + N)^{v_i} \cdot r^N \bmod N^2], v_i \in \{0, 1\}$$

$$c^* := [\prod_i c_i \bmod N^2], v^* = \sum_i v_i$$

端到端投票 (End-to-End Voting)

□ 端到端投票系统

- 投票 (cast) : 投票者 (Voter) 投票 (ballot) 到投票机 (Voting Machine, VM)
- 张贴 (Post) : 将票公开到公告栏 (Public Bulletin Board, PBB)
- 计票 (Count) : 根据公告栏由选举官 (Election Official, EO) 计票

□ 安全目标

- 端到端可验证性: 任何投票者确信按意愿投票, 按投票来张贴, 按张贴来计票
- 隐私: 没人知道投了什么票, 甚至投票者也无法说服其他人她投了什么票; 隐私意味着抗强迫!

三票投票法 (ThreeBallot [Rivest (2006)] w/o Crypto)

- ❑ 原理：按行选，按列投
- ❑ 每个投票者投三张票，每行是一个候选人，每列是一张票。每行做1或2个标记，选谁就做2个标记，不选谁就做1个标记。不能不做，也不能做3个标记。
- ❑ 每张票有唯一的ID。所有票公布在PBB上。
- ❑ 投票者将任意一张票的拷贝作为收据带回家。收据用对照PBB做完整性检查。

Alice ○	Alice ○	Alice ○
Bob ○	Bob ○	Bob ○
Charlie ○	Charlie ○	Charlie ○
ID3645	ID3371	ID3733

Empty ballots

Alice ○	Alice ○	Alice ●
Bob ●	Bob ●	Bob ○
Charlie ○	Charlie ●	Charlie ○
ID5017	ID5242	ID4583

Vote for Bob

量子密钥分发 (Quantum Key Distribution)

- ❑ BB84 QKD, 由Bennett和Brassard在1984年发明, 利用光子偏振状态来在公开信道上传递消息, 并可以发现窃听者
- ❑ 用与制备基相同的测量基来测量光子, 则得到原始光子偏振方向; 否则, 得到随机的方向
- ❑ 首先, Alice产生随机比特串, 并用随机的一组制备基来产生相应的带偏振光子, 发送给Bob
- ❑ 然后, Bob产生随机测量基来测量光子偏振, 得到一个比特串
- ❑ 最后, Alice和Bob公开自己的制备基和测量基, 将使用了相同基处理和得到的部分比特串作为密钥的一部分; 为了检查是否有人窃听, 也就是在传递信道中对光子偏振测量; Alice和Bob分配公开一段相同基下得到的比特串, 如果相同说明中间没有人窃听; 如果敌手窃听, 则会影响量子传输过程, 光子被其他的基测量后会改变偏振方向, 从而被监测发现

BB84 protocol: C. H. Bennett and G. Brassard (1984)

			Alice's random bits	01101001
			Alice's random sending basis	++x+xxx+
			Photon polarization Alice sends	-\ \\//-
			Bob's random measuring basis	+xxx+x++
			Photon polarization Bob measures	/\ /-/--
			Shared secret key	0 1 0 1
Basis	0	1		
+		-		
x	/	\		

本节小结

- ❑ 克拉克三定律之一：任何足够先进的技术和魔法是不可区分的。
- ❑ One of Clarke's three laws: Any sufficiently advanced technology is indistinguishable from magic.