密码学多签系列

第 6 课：GG18 门限签名

lynndell 博士

新火科技 密码学专家 lynndell2010@gmail.com

目录

zk 系列

11. **Groth16 证明系统**
12. **Plonk 证明系统**
13. **UltraPlonk 证明系统**
14. **SHA256 查找表技术**
15. **Halo2 证明系统**
16. **zkSTARK 证明系统**

1. 预备知识

1.1 Paillier 同态加密

密钥生成：生成两个长度相同的大素数 p,q，满足 $\gcd\left(pq,(p-1)(q-1)\right)=1$；计算 $n:=p\cdot q$，$\lambda:=lcm(p-1,q-1)$；分式除法函数 $L(y)=(y-1)/n$；

$g=n+1\in Z_{n^2}^*$，使得 $\mu=\left(L(\Gamma^\lambda \bmod n^2)\right)^{-1} \bmod n$ 存在。公钥为 n，私钥为 p,q 或 λ。

加密：消息 $m\in Z_n$，选择随机数 $r\in Z_n^*$，计算密文 $c:=g^m\cdot r^n \bmod n^2$。

解密：输入密文 $c\in Z_{n^2}$，如下计算解密 $m:=\dfrac{L(c^\lambda \bmod n^2)}{L(g^\lambda \bmod n^2)} \bmod n$。

给定两个密文 $c_1, c_2 \in Z_{n^2}$， $c_1 = Enc_{pk}(m_1), c_2 = Enc_{pk}(m_2)$

密文加法同态 $\oplus$： $c_1 \oplus c_2 = c_1 c_2 \bmod n^2$，则 $c_1 \oplus c_2 = c_1 c_2 \bmod n^2 = Enc_{pk}(m_1 + m_2 \bmod n)$；

随机数与密文乘法同态 $\otimes$： $a \in Z_n, c = Enc_{pk}(m)$，则 $a \otimes c = c^a \bmod n^2 = Enc_{pk}(a \cdot m \bmod n)$。

1.2 份额转换协议 MtA

协议描述

输入：Alice 输入保密数据 a，Bob 输入保密数据 b；

输出：Alice 获得保密数据 α，Bob 获得保密数据 β；

功能：不知道对方的保密数据，且 $ab = \alpha + \beta$。

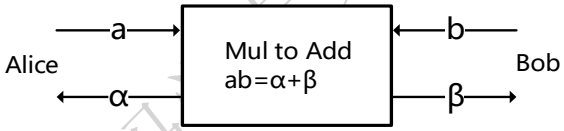

图 1. 份额转换协议

	Alice	Bob
1	Paillier 公钥为 pk，选择随机数 $a \in Z_n$，计算 $c_1 := Enc_{pk}(a)$。 **发送** c_1 与范围证明 $RangeProof\{a \mid a < q^3, c_1 = Enc_{pk}(a)\}$	
2		**接收** c_1 与范围证明 $RangeProof\{a \mid a < q^3, c_1 = Enc_{pk}(a)\}$ **校验**范围证明；

		选择2个随机数 $b,\beta' \in Z_n$，**同态计算** $c_2 := (b \otimes c_1) \oplus Enc_{pk}(\beta'), B := g^b$，则 $c_2 = Enc_{pk}(ab + \beta' \bmod n)$。 获得加性份额为 β，其中 $\beta = -\beta' \bmod n$。 发送 c_2 和范围证明* $RangeProof\left\{b,\beta' \middle| b < q^3, \beta' < q^7, c_2 = (b \otimes c_1) \oplus Enc_{pk}(\beta'), B = g^b\right\}$
3	**接收** c_2 和范围证明* **校验**范围证明*; 解密 c_2 获得 α，其中 $\alpha = ab + \beta' \bmod n$	
	分析：Alice 与 Bob 不知道对方的保密数据，但是保密数据满足等式关系 $\alpha + \beta = (ab + \beta') + (-\beta') = ab$	

1.3 零知识证明

1.3.1 zk-Sigma 对 NP 的零知识证明

初始化：椭圆曲线生成元为 G，群的阶为 $|F_r|$；标量域为 F_r，基域为 F_q；

用户秘密为 ω，公开输入为 H，满足离散对数关系 $H = \omega \cdot G$。

1：（**承诺**）选择随机数 $r \in F_r$，计算 $R := r \cdot G$

2：（**挑战**）计算随机数 $e := Hash(H,R) \bmod |F_r|$

3：（**响应**）计算 $z := r + e \cdot \omega \bmod |F_r|$，发送 (R,z)

4：（**验证**）计算 $e := Hash(H,R) \bmod |F_r|$，如果等式 $z \cdot G == R + e \cdot H$ 成立，则接受，否则拒绝。

1.3.2 zk-Sigma*证明知道 2 个随机数 *s,l*

对协议中第 **5B** 步的零知识证明 $ZK\{s,l | V = s \cdot R + l \cdot G\}$ 补充。

初始化：椭圆曲线生成元为 G，标量域为 F_r，基域为 F_q；

用户秘密为 s,l,ρ，公开输入为 G,V,R，满足离散对数关系 $V = s \cdot R + l \cdot G$。

1：（**承诺**）选择随机数 $a,b \in F_r$，计算 $H := a \cdot R + b \cdot G$

2：（**挑战**）计算随机数 $c := Hash(G,V,R,H) \bmod |F_r|$

3：（**响应**）计算 $t := a + c \cdot s \bmod |F_r|, u = b + cl \bmod |F_r|$，发送 (H,t,u)

4：（**验证**）计算随机数 $c := Hash(G,V,R,H) \bmod |F_r|$,如果等式 $t \cdot R + u \cdot G == H + c \cdot V$ 成立，则接受，否则拒绝。

一致性原理如下：

$$t \cdot R + u \cdot G = (a + c \cdot s) \cdot R + (b + cl) \cdot G = H + c \cdot V$$

1.3.3 zk-Schnoor 证明知道私钥

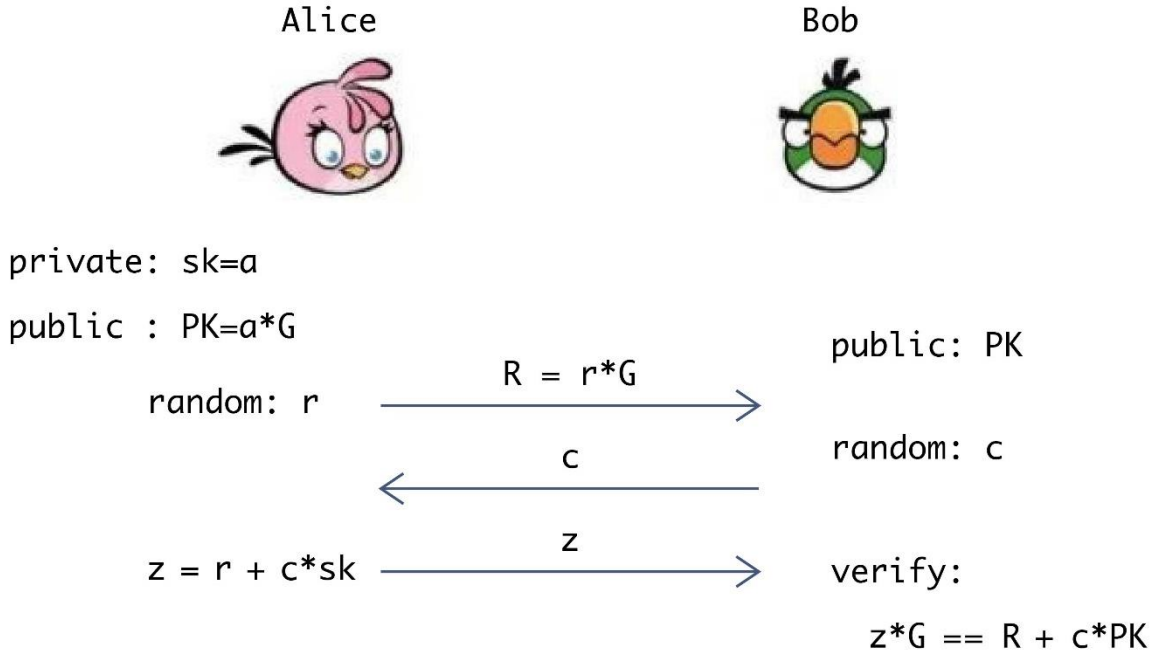

图 2. 交互式 Schnorr 协议

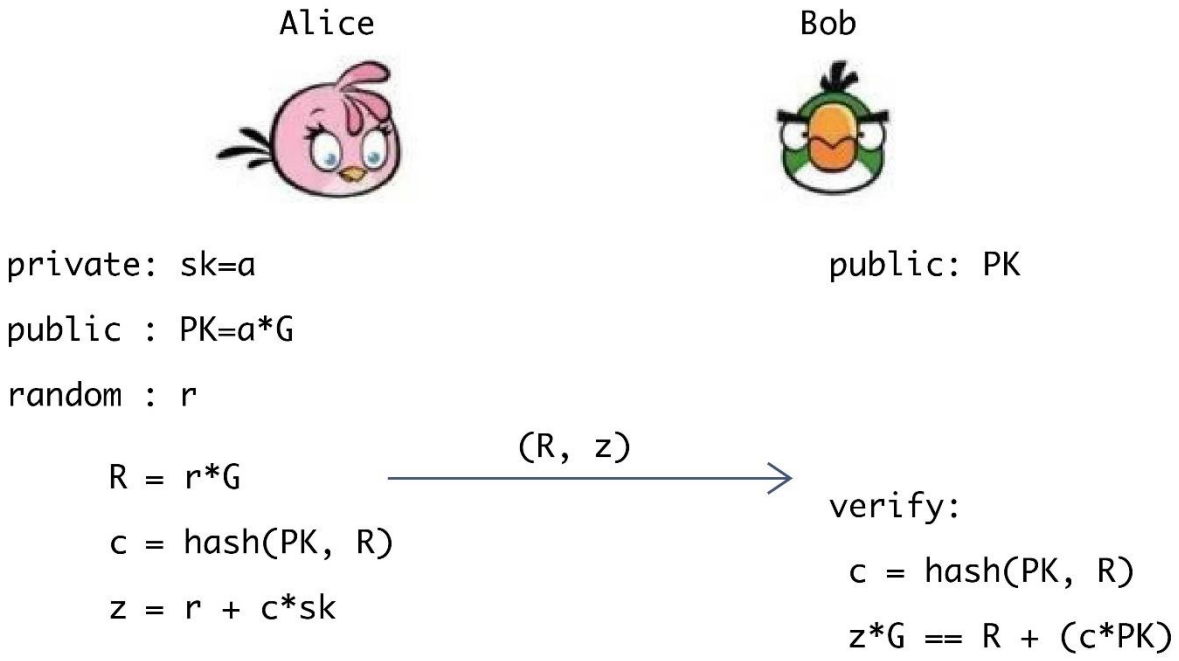

图 3. 非交互式 Schnorr 协议 A 版

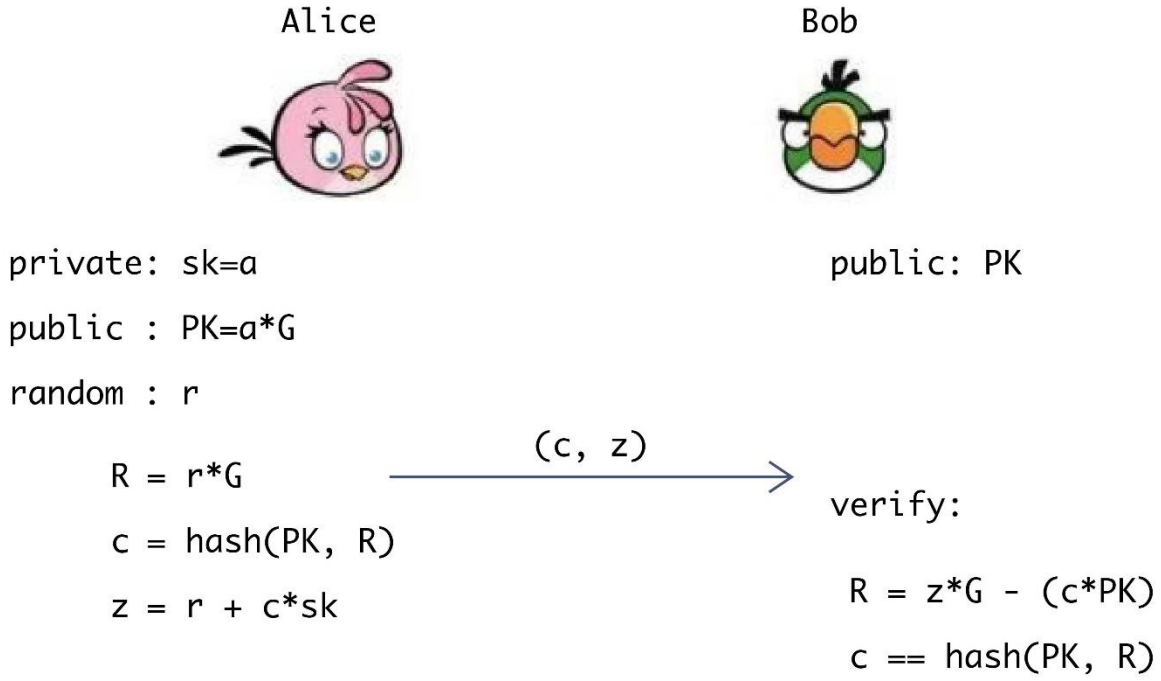

图 3. 非交互式 Schnorr 协议 B 版

1.3.4 zk-RangeProof 范围证明

BulletProof 是 Pedersen 承诺数据的范围证明。这个范围证明是 Paillier 密文中数据的范围证明。

范围证明 $RangeProof\{a \mid a < q^3, c_1 = Enc_{pk}(a)\}$；

证明方 Paillier 公钥为 N，私钥为 p,q，其中 $N = p \cdot q$。DSA 群的阶为 q。消息 $m \in \mathbb{Z}_q$ 和随机数 $r \in \mathbb{Z}_N^*$，密文为 $c = g^m r^N \bmod N^2$。

需要证明：$m \in [-q^3, q^3]$，且满足运算关系$c = g^m r^N \bmod N^2$。

证明方	验证方
选择随机数$\alpha \in \mathbb{Z}_{q^3}, \beta \in \mathbb{Z}_N^*, \gamma \in \mathbb{Z}_{q^3 N}, \rho \in \mathbb{Z}_{qN}$，计算 $z = h_1^m h_2^\rho \bmod N, u = g^\alpha \beta^N \bmod N^2, w = h_1^\alpha h_2^\gamma \bmod N$ 发送z, u, w；	
	发送随机数$e \in \mathbb{Z}_q$；
计算$s = r^e \beta \bmod N, s_1 = em + \alpha, s_2 = e\rho + \gamma$ 发送s, s_1, s_2；确保知道秘密$r, \beta, m, \alpha, \gamma$，随机数起随机化作用。	
	进行以下 3 个校验： $s_1 < q^3, u == g^{s_1} s^N c^{-e} \bmod N^2, h_1^{s_1} h_2^{s_2} z^{-e} == w \bmod N$

分析：（1）范围校验：$\alpha \in \mathbb{Z}_{q^3}, e, m \in \mathbb{Z}_q, s_1 = em + \alpha \in [0, q^2 + q^3] = [0, q^3] \cup (q^3, q^2 + q^3]$，$s_1$在区块$[0, q^3]$的概率为$\frac{q^3}{q^3 + q^2} = \frac{1}{1 + 1/q}$接近 1，在区块$(q^3, q^2 + q^3]$的概率$\frac{q^2}{q^3 + q^2} = \frac{1}{q+1}$可忽略。$s_1 < q^3$确保$m \in [-q^3, q^3]$。**后面 2 个校验确保**响应正确。

（2）Paillier 校验：确保 c 中的 m **等于** s1 中的 m，满足 paillier 运算关系：$g^{s_1} s^N c^{-e} \bmod N^2 = g^{em+\alpha} (r^{eN} \beta^N)(g^m r^N)^{-e} = g^\alpha \beta^N \bmod N^2 = u$

（3）双重 sigma 与 s1/s2 线性关系校验：确保 z 中的 m 等于 s1 中的 m，$h_1^{s_1}h_2^{s_2}z^{-e}=h_1^{em+\alpha}h_2^{e\rho+\gamma}(h_1^m h_2^\rho)^{-e}=h_1^\alpha h_2^\gamma \bmod N=w \bmod N$

如果缺少这个 zk，则 Paillier 同态加密不满足安全性，能够攻破 MtA 协议，获得对方的私钥。

1.3.5 zk-RangeProof*范围证明

Bob 进行范围证明 $RangeProof\left\{b,\beta' \middle| b<q^3,\beta'<q^7,c_2=(b\otimes c_1)\oplus Enc_{pk}(\beta'),B=g^b\right\}$

证明方 Paillier 公钥为 N，私钥为 p,q，其中 $N=p\cdot q$，q 是 DSA 群的阶，生成元为 g。消息 $m\in\mathbb{Z}_q$ 对应的 paillier 密文为 $c_1=g^m r^N \bmod N^2$。选择随机数 $x\in\mathbb{Z}_q, y\in\mathbb{Z}_{q^5}, r\in\mathbb{Z}_N^*$，计算 $c_2=c_1^x\cdot g^y r^N \bmod N^2$，$X=g^x \bmod q$。

需要证明：$x\in[-q^3,q^3], y\in[-q^7,q^7]$，且满足运算关系 $c_2=(b\otimes c_1)\oplus Enc_{pk}(\beta'),B=g^b$。

证明方	验证方
选择随机数 $\alpha\in\mathbb{Z}_{q^3},\rho\in\mathbb{Z}_{qN},\rho'\in\mathbb{Z}_{q^3N},\sigma\in\mathbb{Z}_{qN},\beta\in\mathbb{Z}_N^*,\gamma\in\mathbb{Z}_{q^7},\tau\in\mathbb{Z}_{q^3N}$，计算 $u=g^\alpha,$ $z=h_1^x h_2^\rho \bmod N,$ $z'=h_1^\alpha h_2^{\rho'} \bmod N,$ $t=h_1^y h_2^\sigma \bmod N,$ $v=c_1^\alpha\cdot g^\gamma \beta^N \bmod N^2,$ $w=h_1^\gamma h_2^\tau \bmod N$	

发送 u, z, z', t, v, w；	
	发送随机数 $e \in \mathbb{Z}_q$；
计算 $s = r^e \beta \bmod N, s_1 = ex + \alpha, s_2 = e\rho + \rho', t_1 = ey + \gamma, t_2 = e\sigma + \tau$ 发送 s, s_1, s_2, t_1, t_2；	
	进行以下 5 个校验： $s_1 < q^3, t_1 < q^7,$ $g^{s_1} == X^e u,$ $h_1^{s_1} h_2^{s_2} == z^e z' \bmod N,$ $h_1^{t_1} h_2^{t_2} == t^e w \bmod N,$ $c_2^e v == c_1^{s_1} s^N g^{t_1} \bmod N^2$

分析：（1）**范围校验：** $e \in \mathbb{Z}_q, y \in \mathbb{Z}_{q^5}, \gamma \in \mathbb{Z}_{q^7}, t_1 = ey + \gamma \in [0, q^6 + q^7] = [0, q^7] \cup (q^7, q^6 + q^7]$， t_1 在区块 $[0, q^7]$ 的概率为 $\frac{q^7}{q^7 + q^6} = \frac{1}{1 + 1/q}$ 接近 1，

在区块 $(q^7, q^6 + q^7]$ 的概率 $\frac{q^6}{q^7 + q^6} = \frac{1}{q+1}$ 可忽略。$s_1 < q^3, t_1 < q^7$ 确保 $x \in [-q^3, q^3], y \in [-q^7, q^7]$。**后面 4 个校验确保响应正确**

（2）**sigma 协议离散对数校验：** $X^e u = g^{xe+\alpha} = g^{s_1}$；

（3）**双重 sigma 与 s1/s2 线性关系校验：** $h_1^{s_1} h_2^{s_2} \bmod N = h_1^{ex+\alpha} h_2^{e\rho+\rho'} \bmod N = (h_1^x h_2^\rho)^e (h_1^\alpha h_2^{\rho'}) \bmod N = z^e z' \bmod N$；

（4）**双重 sigma 与 t1/t2 线性关系校验：** $h_1^{t_1}h_2^{t_2} \bmod N = h_1^{ey+\gamma}h_2^{e\sigma+\tau} \bmod N = (h_1^y h_2^\sigma)^e (h_1^\gamma h_2^\tau) \bmod N = t^e w \bmod N$；

（5）**Paillier 校验：** $c_2^e v = (c_1^x g^y r^N)^e (c_1^\alpha g^\gamma \beta^N) \bmod N^2 = c_1^{ex+\alpha} s^N g^{ey+\gamma} \bmod N^2 = c_1^{s_1} s^N g^{t_1} \bmod N^2$；

如果缺少这个 zk，则 Paillier 同态加密不满足安全性，能够攻破 MtA 协议，获得对方的私钥。

1.3.6 zk-Paillier-N 非平方证明

证明生成正确的 Paillier 密钥对。等价于 Li17 两方签名协议中的证明 $\gcd(N, \varphi(N)) = 1$。

p, q 为两个**不同的大素数** $p \neq q$，令 $N = p \cdot q$。**证明** N 是两个不同的素数之积，而不是两个相同数的乘积。

证明方	验证方
发送 N	
	选择随机数 $x \in \mathbb{Z}_N^*$，发送 x
计算 N 的模 $\psi(N)$ 逆元 $A := N^{-1} \bmod \psi(N)$，其中 $\psi(N)$ 为欧拉函数； 然后计算 $y := x^A \bmod N$；发送 y	
	校验 $y^N == x \bmod N$

分析：

预备知识：**模 n 逆元存在性**

如果两个正整数 a 和 n 互素，则存在整数 b，使得 $ab \equiv 1 \bmod n$，则称 b 是 a 的模 n 逆元。

证明：使用欧拉定理 $a^{\psi(n)} \equiv 1 \bmod n$，则 $a^{1+\psi(n)-1} \equiv 1 \bmod n$，则 $a \times a^{\psi(n)-1} \equiv 1 \bmod n$，则 $b = a^{\psi(n)-1}$。

完备性：如果 $N = p \cdot q$，则 $\psi(N) = (p-1)(q-1)$，且 $\gcd(N, \psi(N)) = 1$，则使用**模 n 逆元**存在性定理，令 $a = N, n = \psi(N)$，带入 $b = a^{\psi(n)-1}$，则有 $b = N^{\psi(\psi(n))-1}$，令 $A = N^{\psi(\psi(N))-1}$ 就是对应的模 $\psi(N)$ 逆元。

因此，$y^N \bmod N = x^{AN} \bmod N = x \bmod N$。

健壮性：如果 $N = k^2$ 或 $N = k_1 \cdot k_2$，其中 k_1, k_2 不是均为大素数，则 $\gcd(N, \psi(N)) = d > 1$，因此 $\left\{x^N \mid x \in \mathbb{Z}_N^*\right\} = \left|\mathbb{Z}_N^*\right| / d$，因此 $y^N == 1 \bmod N$ 的概率为 $1/d$。

零知识：模拟器选择随机数 $y \in \mathbb{Z}_N^*$，输出 $y^N \bmod N$，则 $x = y^N \bmod N$ 是随机分布的。

如果缺少这个 zk，则 Paillier 同态加密不满足安全性，能够攻破 MtA 协议，获得对方的私钥。

1.4 Feldman 可验证秘密共享协议

1.4.1 中心化 shamir 秘密共享协议

秘密分发：用户 i' 的秘密为 $sk \in [1, n-1]$，选择随机数 $a_1, ..., a_{t-1} \in [1, n-1]$ 构造 $t-1$ 阶多项式

$$p(x) = sk + a_1 \cdot x^1 + ... + a_{t-1} \cdot x^{t-1}$$

构造拉格朗日冗余：计算多项式的值 $p(i) := sk + a_1 i^1 + ... + a_{t-1} i^{t-1}, i = 1, ..., m$，将 $p(i)$ 保密发送给对应的参与方 $i, i = 1, ..., m$。自己的排序为 i'，则自己保存 $p(i')$。

秘密重构： t 个参与方广播多项式的值 $p(1), ..., p(t)$，则能够解 t 元方程组**或拉格朗日插值法**，解出 sk。如果出现错误，则不知道谁错了。

shamir 秘密共享协议缺点：缺少验证过程。密码协议中全是随机数，根本不知道接收或计算的随机数是否正确，因此需要校验，确保正确。

1.4.2 中心化可验证秘密共享协议

秘密分发：用户 i' 的秘密为 $sk \in [1, n-1]$，选择随机数 $a_1, ..., a_{t-1} \in [1, n-1]$ 构造 $t-1$ 阶多项式

$$p(x) = sk + a_1 \cdot x^1 + ... + a_{t-1} \cdot x^{t-1}$$

构造拉格朗日冗余：计算多项式的值 $p(i) := sk + a_1 i^1 + ... + a_{t-1} i^{t-1}, i = 1, ..., m$，将 $p(i)$ 保密发送给对应的参与方 $i, i = 1, ..., m$。自己的排序为 i'，则自己保存 $p(i')$。

计算 Feldman 校验元组： $A_0 := sk \cdot G, A_i := a_i \cdot G, i = 1, ..., t-1$，**广播** $\{A_i\}_{i=0,...,t-1}$。

校验：参与方 j 接收到的多项式值为 $p(j) := sk + a_1 j^1 + ... + a_{t-1} j^{t-1}$，进行以下 **Feldman 校验**

$$p(j) \cdot G == \sum_{j=0}^{t} i^j A_j$$

一致性过程如下：

$$\begin{aligned} p(j) \cdot G &= (sk + a_1 j^1 + ... + a_{t-1} j^{t-1}) \cdot G \\ &= A_0 + j^1 \cdot A_1 + ... + j^t \cdot A_{t-1} \\ &= \sum_{j=0}^{t-1} i^j A_i \end{aligned}$$

秘密重构： t 个参与方广播多项式的值 $p(1), ..., p(t)$，则能够解 t 元方程组**或拉格朗日插值法**，解出 sk。

有校验过程，参与方能够确定秘密信息是正确的。因此，应该尽可能多使用有校验的协议。

1.4.3 中心化分片私钥刷新

安全需求：提高分片私钥的安全性。

份额刷新方法 1：可信第三方选择**新的**随机数 $a_1',...,a_{t-1}' \in F_r$ 构造 $t-1$ 阶多项式

$$p'(x) = a_1' x^1 + ... + a_{t-1}' x^{t-1}，\text{常数项为零}$$

构造拉格朗日冗余：计算 $p'(i) := a_1' i^1 + ... + a_{t-1}' i^{t-1}, i = 1,...,n$，将 $p'(i)$ 保密发送给对应的参与方 $i, i = 1,...,n$。

计算 Feldman 校验元组： $A_i' := a_i' \cdot G, i = 1,...,t-1$，广播 $\{A_i'\}_{i=0,...,t-1}$。

校验：参与方 j 接收到的**分片私钥**为 $p'(j) := a_1' j^1 + ... + a_{t-1}' j^{t-1}$，进行以下 **Feldman 校验**

$$P'(j) \cdot G == \sum_{j=1}^{t} i^j A_j'$$

校验过程如下

$$\begin{aligned} P'(j) \cdot G &= (a_1' j^1 + ... + a_{t-1}' j^{t-1}) \cdot G \\ &= j^1 \cdot A_1' + ... + j^t \cdot A_{t-1}' \\ &= \sum_{j=1}^{t-1} i^j A_i' \end{aligned}$$

参与方 j 将 2 次的多项式值相加：$p(j) + p'(j) := sk + (a_1 + a_1') j^1 + ... + (a_{t-1} + a_{t-1}') j^{t-1}$

秘密重构：t 个参与方广播份额 $p(1) + p'(1), ..., p(t) + p'(t)$，则能够解 t 元方程组**或拉格朗日插值法**，解出 sk。

份额刷新方法 2：更新过程本质上等于可信第三方的多项式为 $f(x) = sk + (a_1 + a_1')x^1 + ... + (a_{t-1} + a_{t-1}')x^{t-1}$

将多项式的值发送给各个参与方，并广播这些随机数的离散对数。

1.4.4 分布式可验证秘密共享协议

	P_1	P_2	P_3
1	选择**原始**随机数 u_1 计算 $U_1 := u_1 \cdot G$ $(KGC_1, KGD_1) = Com(U_1)$	选择**原始**随机数 u_2 计算 $U_2 := u_2 \cdot G$ $(KGC_2, KGD_2) = Com(U_2)$	选择**原始**随机数 u_3 计算 $U_3 := u_3 \cdot G$ $(KGC_3, KGD_3) = Com(U_3)$
	广播 KGC_1, E_1	广播 KGC_2, E_2	广播 KGC_3, E_3
2	广播 KGD_1	广播 KGD_2	广播 KGD_3
3	校验承诺正确性，然后计算公共公钥：$PK = U_1 + U_2 + U_3$		
4	选择随机数 $a_1, b_1 \in F_r$， 构造 2 阶多项式 $p_1(x) = u_1 + x \cdot a_1 + x^2 \cdot b_1$ 存储 $p_1(1)$， **门限为 3** **构造拉格朗日冗余：**将 $p_1(2), p_1(3)$ 保密发给对应用户 P_2, P_3	选择随机数 $a_2, b_2 \in F_r$， 构造 2 阶多项式 $p_2(x) = u_2 + x \cdot a_2 + x^2 \cdot b_2$ 存储 $p_2(2)$， **门限为 3** **构造拉格朗日冗余：**将 $p_2(1), p_2(3)$ 保密发给对应用户 P_1, P_3	选择随机数 $a_3, b_3 \in F_r$ 构造 2 阶多项式 $p_3(x) = u_3 + x \cdot a_3 + x^2 \cdot b_3$ 存储 $p_3(3)$ **门限为 3** **构造拉格朗日冗余：**将 $p_3(1), p_3(2)$ 保密发给对应用户 P_1, P_2

	计算并广播 **Feldman 校验元组** $A_1 := a_1 \cdot G$ $B_1 := b_1 \cdot G$	计算并广播 **Feldman 校验元组** $A_2 := a_2 \cdot G$ $B_2 := b_2 \cdot G$	计算并广播 **Feldman 校验元组** $A_3 := a_3 \cdot G$ $B_3 := b_3 \cdot G$
5	拥有保密数据为 $p_1(1) = u_1 + a_1 + b_1,$ $p_2(1) = u_2 + a_2 + b_2,$ $p_3(1) = u_3 + a_3 + b_3$	拥有保密数据为 $p_1(2) = u_1 + 2a_1 + 4b_1,$ $p_2(2) = u_2 + 2a_2 + 4b_2,$ $p_3(2) = u_3 + 2a_3 + 4b_3$	拥有保密数据为 $p_1(3) = u_1 + 3a_1 + 9b_1,$ $p_2(3) = u_2 + 3a_2 + 9b_2,$ $p_3(3) = u_3 + 3a_3 + 9b_3$
6	**Feldman 校验** $p_1(1) \cdot G == U_1 + A_1 + B_1,$ $p_2(1) \cdot G == U_2 + A_2 + B_2,$ $p_3(1) \cdot G == U_3 + A_3 + B_3$	**Feldman 校验** $p_1(2) \cdot G == U_1 + 2A_1 + 4B_1,$ $p_2(2) \cdot G == U_2 + 2A_2 + 4B_2,$ $p_3(2) \cdot G == U_3 + 2A_3 + 4B_3$	**Feldman 校验** $p_1(3) \cdot G == U_1 + 3A_1 + 9B_1,$ $p_2(3) \cdot G == U_2 + 3A_2 + 9B_2,$ $p_3(3) \cdot G == U_3 + 3A_3 + 9B_3$
7	计算分片私钥 $x_1 := \sum_{i=1}^{3} p_i(1)$ $= \sum_{i=1}^{3} (u_i + a_i + b_i)$ $= sk + \sum_{i=1}^{3} (a_i + b_i)$	计算分片私钥 $x_2 := \sum_{i=1}^{3} p_i(2)$ $= \sum_{i=1}^{3} (u_i + 2a_i + 4b_i)$ $= sk + \cdot \sum_{i=1}^{3} (2a_i + 4b_i)$	计算分片私钥 $x_3 := \sum_{i=1}^{3} p_i(3)$ $= \sum_{i=1}^{3} (u_i + 3a_i + 9b_i)$ $= sk + \sum_{i=1}^{3} (3a_i + 9b_i)$
8	计算并广播**分片公钥** $X_1 := PK + \left(\sum_{i=1}^{3} (a_i + b_i) \right) \cdot G$	计算并广播**分片公钥** $X_2 := PK + \left(\sum_{i=1}^{3} (2a_i + 4b_i) \right) \cdot G$	计算并广播**分片公钥** $X_3 := PK + \left(\sum_{i=1}^{3} (3a_i + 9b_i) \right) \cdot G$

1.4.5 分布式分片私钥刷新

安全需求：提高分片私钥安全性。

1	公共公钥 $PK = U_1 + U_2 + U_3$ 不变		
	选择**新**随机数 $a_1', b_1' \in F_r$， 构造**新** 2 阶多项式 $p_1'(x) = x \cdot a_1' + x^2 \cdot b_1'$ **注意常数项为 0** 存储 $p_1'(1)$， **门限为 3** **构造拉格朗日冗余：** 将 $p_1'(2), p_1'(3)$ 保密发给对应用户 P_2, P_3 计算并广播 **Feldman 校验元组** $A_1' := a_1' \cdot G$，$B_1' := b_1' \cdot G$	选择**新**随机数 $a_2', b_2' \in F_r$， 构造**新** 2 阶多项式 $p_2'(x) = x \cdot a_2' + x^2 \cdot b_2'$ **注意常数项为 0** 存储 $p_2'(2)$， **门限为 3** **构造拉格朗日冗余：** 将 $p_2'(1), p_2'(3)$ 保密发给对应用户 P_1, P_3 计算并广播 **Feldman 校验元组** $A_2' := a_2' \cdot G$，$B_2' := b_2' \cdot G$	选择**新**随机数 $a_3', b_3' \in F_r$ 构造**新** 2 阶多项式 $p_3'(x) = x \cdot a_3' + x^2 \cdot b_3'$ **注意常数项为 0** 存储 $p_3'(3)$ **门限为 3** **构造拉格朗日冗余：** 将 $p_3'(1), p_3'(2)$ 保密发给对应用户 P_1, P_2 计算并广播 **Feldman 校验元组** $A_3' := a_3' \cdot G$，$B_3' := b_3' \cdot G$
2	拥有保密数据为 $p_1'(1) = a_1' + b_1'$， $p_2'(1) = a_2' + b_2'$， $p_3'(1) = a_3' + b_3'$	拥有保密数据为 $p_1'(2) = 2a_1' + 4b_1'$， $p_2'(2) = 2a_2' + 4b_2'$， $p_3'(2) = 2a_3' + 4b_3'$	拥有保密数据为 $p_1'(3) = 3a_1' + 9b_1'$， $p_2'(3) = 3a_2' + 9b_2'$， $p_3'(3) = 3a_3' + 9b_3'$
3	进行 **Feldman 校验**	进行 **Feldman 校验**	进行 **Feldman 校验**

	$p_1'(1)\cdot G == A_1'+B_1'$, $p_2'(1)\cdot G == A_2'+B_2'$, $p_3'(1)\cdot G == A_3'+B_3'$	$p_1'(2)\cdot G == 2A_1'+4B_1'$, $p_2'(2)\cdot G == 2A_2'+4B_2'$, $p_3'(2)\cdot G == 2A_3'+4B_3'$	$p_1'(3)\cdot G == 3A_1'+9B_1'$, $p_2'(3)\cdot G == 3A_2'+9B_2'$, $p_3'(3)\cdot G == 3A_3'+9B_3'$
4	计算**新分片私钥** $x_1 := \sum_{i=1}^{3} p_i(1)+\sum_{i=1}^{3} p_i'(1)$ $=\sum_{i=1}^{3}(u_i+a_i+b_i)+\sum_{i=1}^{3}(a_i'+b_i')$ $=sk+\sum_{i=1}^{3}(a_i+a_i'+b_i+b_i')$	计算**新分片私钥** $x_2 := \sum_{i=1}^{3} p_i(2)+\sum_{i=1}^{3} p_i'(2)$ $=\sum_{i=1}^{3}(u_i+2a_i+4b_i)+\sum_{i=1}^{3}(2a_i'+4b_i')$ $=sk+\cdot\sum_{i=1}^{3}\left(2(a_i+a_i')+4(b_i+b_i')\right)$	计算**新分片私钥** $x_3 := \sum_{i=1}^{3} p_i(3)+\sum_{i=1}^{3} p_i'(3)$ $=\sum_{i=1}^{3}(u_i+3a_i+9b_i)+\sum_{i=1}^{3}(3a_i'+9b_i')$ $=sk+\sum_{i=1}^{3}\left(3(a_i+a'_i)+9(b_i+b_i')\right)$
5	计算并广播**新分片公钥** $X_1 := PK+\left(\sum_{i=1}^{3}(a_i+a_i'+b_i+b_i')\right)\cdot G$	计算并广播**新分片公钥** $X_2 := PK+\left(\sum_{i=1}^{3}\left(2(a_i+a_i')+4(b_i+b_i')\right)\right)\cdot G$	计算并广播**新分片公钥** $X_3 := PK+\left(\sum_{i=1}^{3}\left(3(a_i+a_i')+9(b_i+b_i')\right)\right)\cdot G$

1.5 承诺

承诺三个步骤：密钥生成、承诺、打开验证；
密钥生成：生成求值密钥 pk；

承诺：生成承诺与打开信息 $[KGC(M),KGD(D)] := Com(pk,M,R)$，其中 $R := r\cdot G$

打开与验证： $Valid/Invalid \leftarrow Ver(pk,KGC(M),KGD(M))$，如果验证成功，则输出 M，否则拒绝。

1.6 ECDSA

初始化：椭圆曲线生成元为G，标量域为F_r，基域为F_q。

密钥生成：输入安全参数，输出私钥$x \in F_r$和公钥PK，满足离散对数关系

$$PK = x \cdot G$$

签名：输入任意消息M，计算$m := Hash(M)$；选择随机数$k \in F_r$，计算$R := k^{-1} \cdot G$，取R横坐标为$r := R_x \bmod |F_r|$；计算$s := k(m + xr)$，则签名为(r, s)。

验证：输入消息M，计算$m := Hash(M)$；校验$r, s \in F_r$，计算$R' := (s^{-1}m) \cdot G + (s^{-1}r) \cdot PK$，取$R'$横坐标为$r' := R'_x \bmod |F_r|$；校验$r == r'$。如果相等，则接受，否则拒绝。

公式推导过程如下：

$$\begin{aligned} R' &= (s^{-1}m) \cdot G + (s^{-1}r) \cdot PK \\ &= (s^{-1}m) \cdot G + (s^{-1}rx) \cdot G \\ &= \left(s^{-1}(m + rx)\right) \cdot G \\ &= k^{-1} \cdot G \end{aligned}$$

ECDSA 的验证本质：

$$\begin{aligned} s &= k(m + xr) \\ k^{-1} &= s^{-1}(m + xr) \\ k^{-1} \cdot G &= s^{-1}m \cdot G + s^{-1}xr \cdot G \\ R &= s^{-1}m \cdot G + s^{-1}r \cdot PK \end{aligned}$$

检测 $(r, F_r - s)$ 是否为合法的签名：

$$
\begin{aligned}
F_r - s &= k^{-1}(m + xr) \\
k(F_r - s) &= (m + xr) \\
k(F_r - s) \cdot G &= m \cdot G + xr \cdot G \\
-ks \cdot G &= m \cdot G + r \cdot PK \\
-R &= s^{-1} m \cdot G + s^{-1} r \cdot PK
\end{aligned}
$$

计算出 $-R$，纵坐标是负的无所谓，取横坐标得到的就是 $r' := R'\_x \bmod |F_r|$， 校验 $r == r'$。如果相等，则接受，否则拒绝。因此，$(r, F_r - s)$ 是合法签名。既然有 2 个合法签名，所以 Li17 里面计算出 $s = \min\{s', |F_r| - s'\}$，两种签名，确定一个小的。

1.7 复杂度假设

判决性 Diffie-Hellman 困难假设（DDH）： $\mathcal{G}$ 为循环群，阶为 q，生成元为 g；选择 3 个随机数 $a, b, c \in \mathbb{Z}_q$，以下两个集合分布不可区分

$$\{g^a, g^b, g^{ab}\} \approx \{g^a, g^b, g^c\}$$

RSA 密码系统： e 为 RSA 公钥，d 为 RSA 私钥。加密为 $s := x^e$；解密为 $x := s^d = x^{ed}$；
强 RSA 假设： 选择两个不同的大素数 p', q'，计算 $p = 2p' + 1, q = 2q' + 1$，计算 $N = pq$。欧拉函数 $\psi(N) = (p-1)(q-1) = p'q'$。

从零到 $N-1$ 之间，与 N 互素的元素集合记为集合 $\mathbb{Z}_N^*$。e 是与 $\psi(N)$ 互素的整数。对于随机元素 $s \in \mathbb{Z}_N^*$，寻找 x 和 e，满足计算关系 $x^e = s$ 是困难的。

区别：RSA 密码系统的 e 是确定的，强 RSA 密码系统的 e 是不确定的。

2. GG18 (3-3)概述

为直观理解，以下描述引入可信第三方。GG18 使用三方协议代替可信第三方。

2.1 分布式密钥生成

1. 三个用户 P_i 各自选择随机数 $u_i \in F_r$，计算 $U_i := u_i \cdot G$，广播 U_i；将 u_i 发送给可信第三方。

2. 三个用户均能获得 U_1, U_2, U_3。三个用户的公共公钥为 $PK = U_1 + U_2 + U_3$。对应的公共私钥为 $sk = x = u_1 + u_2 + u_3$，三方均不知道私钥 sk，可信第三方知道私钥 sk。

3. 门限为 3，可信第三方基于私钥 sk 构造 2 阶多项式，选择两个随机数 $1, 2$，假设 $sk = 3$

$$f(x) = 3 + x + 2x^2$$

可信第三方将 $f(1) = 6$ 保密发送给用户 1，$f(2) = 13$ 保密发送给用户 2，$f(3) = 24$ 保密发送给用户 3。

4. **私钥恢复：**拉格朗日插值多项式 $\lambda_i(x) = \prod_{j=1, j\neq i}^{t} \frac{x-j}{i-j}$。令 $x = 0$，则 $\lambda_i(0) = \prod_{j=1, j\neq i}^{t} \frac{-j}{i-j}$ 称为拉格朗日插值系数。

三个用户能够如下恢复私钥 sk

$$f(x)=f(1)\cdot\frac{x-2}{1-2}\frac{x-3}{1-3}+f(2)\cdot\frac{x-1}{2-1}\frac{x-3}{2-3}+f(3)\cdot\frac{x-1}{3-1}\frac{x-2}{3-2}$$
$$=6\cdot\frac{x^2-5x+6}{2}+13\cdot\frac{x^2-4x+3}{-1}+24\cdot\frac{x^2-3x+2}{2}$$
$$=3(x^2-5x+6)-13(x^2-4x+3)+12(x^2-3x+2)$$
$$=3+x+2x^2$$
$$w_1(x)=f(1)\cdot\frac{x-2}{1-2}\frac{x-3}{1-3}$$
$$w_2(x)=f(2)\cdot\frac{x-1}{2-1}\frac{x-3}{2-3}$$
$$w_3(x)=f(3)\cdot\frac{x-1}{3-1}\frac{x-2}{3-2}$$
$$sk=f(0)=w_1(0)+w_2(0)+w_3(0)=3$$

将 w_1,w_2,w_3 称为**私钥加性份额**。注意：**私钥加性份额：多项式的值*拉格朗日插值系数，而不是原来选择的随机数。**

三个用户不重构私钥 sk ， 而是基于私钥加性份额 w_1,w_2,w_3 生成**签名加性份额** sig_1,sig_2,sig_3 ， 累加后得到完整的签名

$$sig=sig_1+sig_2+sig_3$$

2.2 三个用户签名

单方 ECDSA 签名：输入任意消息 M ，计算 $m:=Hash(M)$ ；选择随机数 $k\in F_r$ ，计算 $R:=k^{-1}\cdot G$ ，取 R 横坐标为 $r:=R\_x \bmod |F_r|$ ；计算 $s:=k(m+xr)$ ，则签名为 (r,s) 。

分析：用户 P_1, P_2, P_3 各自选择 2 个随机数 $(k_1, \gamma_1), (k_2, \gamma_2), (k_3, \gamma_3)$，计算目标为 (R, s)，如下展开

$$
\begin{aligned}
& R = k^{-1} \cdot G \\
& = (k\gamma)^{-1} \cdot (\gamma \cdot G) \\
& = \left((k_1 + k_2 + k_3)(\gamma_1 + \gamma_2 + \gamma_3)\right)^{-1} \cdot (\gamma_1 + \gamma_2 + \gamma_3) \cdot G \\
& = \begin{pmatrix} k_1\gamma_1 + (k_1\gamma_2) + (k_1\gamma_3) + \\ (k_2\gamma_1) + k_2\gamma_2 + (k_2\gamma_3) + \\ (k_3\gamma_1) + (k_3\gamma_2) + k_3\gamma_3 \end{pmatrix}^{-1} \cdot (\gamma_1 \cdot G + \gamma_2 \cdot G + \gamma_3 \cdot G) \\
& = \begin{pmatrix} k_1\gamma_1 + (\alpha_{1,2} + \beta_{2,1}) + (\alpha_{1,3} + \beta_{3,1}) + \\ (\alpha_{2,1} + \beta_{1,2}) + k_2\gamma_2 + (\alpha_{2,3} + \beta_{3,2}) + \\ (\alpha_{3,1} + \beta_{1,3}) + (\alpha_{3,2} + \beta_{2,3}) + k_3\gamma_3 \end{pmatrix}^{-1} \cdot (\Gamma_1 + \Gamma_2 + \Gamma_3) \\
& = \begin{pmatrix} k_1\gamma_1 + \alpha_{1,2} + \alpha_{1,3} + \beta_{1,2} + \beta_{1,3} \\ k_2\gamma_2 + \beta_{2,1} + \alpha_{2,1} + \alpha_{2,3} + \beta_{2,3} \\ k_3\gamma_3 + \beta_{3,1} + \beta_{3,2} + \alpha_{3,1} + \alpha_{3,2} \end{pmatrix}^{-1} \cdot (\Gamma_1 + \Gamma_2 + \Gamma_3) \\
& = (\delta_1 + \delta_2 + \delta_3)^{-1} \cdot (\Gamma_1 + \Gamma_2 + \Gamma_3)
\end{aligned}
$$

用户 P_1, P_2, P_3 各自广播 $(\delta_1, \Gamma_1), (\delta_2, \Gamma_2), (\delta_3, \Gamma_3)$ 而不泄露 $(k_1, \gamma_1), (k_2, \gamma_2), (k_3, \gamma_3)$。

$$
\begin{aligned}
&s = k(m + xr) \\
&= mk + rkx \\
&= m(k_1 + k_2 + k_3) + r(k_1 + k_2 + k_3)(w_1 + w_2 + w_3) \\
&= m(k_1 + k_2 + k_3) + r\begin{pmatrix} k_1w_1 + k_1w_2 + k_1w_3 + \\ k_2w_1 + k_2w_2 + k_2w_3 + \\ k_3w_1 + k_3w_2 + k_3w_3 \end{pmatrix} \\
&= m(k_1 + k_2 + k_3) + r\begin{pmatrix} k_1w_1 + (u_{1,2} + v_{2,1}) + (u_{1,3} + v_{3,1}) + \\ (u_{2,1} + v_{1,2}) + k_2w_2 + (u_{2,3} + v_{3,2}) + \\ (u_{3,1} + v_{1,3}) + (u_{3,2} + v_{2,3}) + k_3w_3 \end{pmatrix} \\
&= m(k_1 + k_2 + k_3) + r\begin{pmatrix} k_1w_1 + u_{1,2} + v_{1,2} + v_{1,3} + u_{1,3} + \\ u_{2,1} + v_{2,1} + k_2w_2 + u_{2,3} + v_{2,3} + \\ v_{3,1} + u_{3,1} + u_{3,2} + v_{3,2} + k_3w_3 \end{pmatrix} \\
&= m(k_1 + k_2 + k_3) + r(\sigma_1 + \sigma_2 + \sigma_3) \\
&= (mk_1 + r\sigma_1) + (mk_2 + r\sigma_2) + (mk_3 + r\sigma_3) \\
&= s_1 + s_2 + s_3
\end{aligned}
$$

用户 P_1, P_2, P_3 各自广播 s_1, s_2, s_3 而不泄露 $(k_1, w_1), (k_2, w_2), (k_3, w_3)$。

因此，用户 P_1, P_2, P_3 各自的**签名加性份额**为 sig_1, sig_2, sig_3。

如果用户 P_1, P_2, P_3 是诚实用户，则能够直接广播**签名加性份额** sig_1, sig_2, sig_3，生成正确的签名；如果不确定对方是否被黑客操控，则需要先广播**签名加性份额的承诺**，密态校验，再打开与验证承诺，再广播**签名加性份额**，累加后得到完整签名再校验完整签名的正确性。

	计算 R 概述		
	P_1	P_2	P_3
	开始计算公共随机点 R		
1	保密输入(k_1,γ_1)	保密输入(k_2,γ_2)	保密输入(k_3,γ_3)
	以下进行 6 个份额转换协议 MtA		
2.1	$\alpha_{1,2}\leftarrow\left\{P_1(k_1)\overset{\text{MtA}}{\rightleftarrows}P_2(\gamma_2)\right\}(k_1\gamma_2)$	$\beta_{2,1}\leftarrow\left\{P_1(k_1)\overset{\text{MtA}}{\rightleftarrows}P_2(\gamma_2)\right\}(k_1\gamma_2)$	
2.2	$\alpha_{1,3}\leftarrow\left\{P_1(k_1)\overset{\text{MtA}}{\rightleftarrows}P_3(\gamma_3)\right\}(k_1\gamma_3)$		$\beta_{3,1}\leftarrow\left\{P_1(k_1)\overset{\text{MtA}}{\rightleftarrows}P_3(\gamma_3)\right\}(k_1\gamma_3)$
2.3	$\beta_{1,2}\leftarrow\left\{P_2(k_2)\overset{\text{MtA}}{\rightleftarrows}P_1(\gamma_1)\right\}(k_2\gamma_1)$	$\alpha_{2,1}\leftarrow\left\{P_2(k_2)\overset{\text{MtA}}{\rightleftarrows}P_1(\gamma_1)\right\}(k_2\gamma_1)$	
2.4		$\alpha_{2,3}\leftarrow\left\{P_2(k_2)\overset{\text{MtA}}{\rightleftarrows}P_3(\gamma_3)\right\}(k_2\gamma_3)$	$\beta_{3,2}\leftarrow\left\{P_2(k_2)\overset{\text{MtA}}{\rightleftarrows}P_3(\gamma_3)\right\}(k_2\gamma_3)$
2.5	$\beta_{1,3}\leftarrow\left\{P_3(k_3)\overset{\text{MtA}}{\rightleftarrows}P_1(\gamma_1)\right\}(k_3\gamma_1)$		$\alpha_{3,1}\leftarrow\left\{P_3(k_3)\overset{\text{MtA}}{\rightleftarrows}P_1(\gamma_1)\right\}(k_3\gamma_1)$
2.6		$\beta_{2,3}\leftarrow\left\{P_3(k_3)\overset{\text{MtA}}{\rightleftarrows}P_2(\gamma_2)\right\}(k_3\gamma_2)$	$\alpha_{3,2}\leftarrow\left\{P_3(k_3)\overset{\text{MtA}}{\rightleftarrows}P_2(\gamma_2)\right\}(k_3\gamma_2)$

	份额转换协议 MtA 结束		
3	计算 $\delta_1 := k_1\gamma_1 + \alpha_{1,2} + \alpha_{1,3} + \beta_{1,2} + \beta_{1,3}$ $\Gamma_1 := \gamma_1 \cdot G$	计算 $\delta_2 := k_2\gamma_2 + \beta_{2,1} + \alpha_{2,1} + \alpha_{2,3} + \beta_{2,3}$ $\Gamma_2 := \gamma_2 \cdot G$	计算 $\delta_3 := k_3\gamma_3 + \beta_{3,1} + \beta_{3,2} + \alpha_{3,1} + \alpha_{3,2}$ $\Gamma_3 := \gamma_3 \cdot G$
4	广播 (δ_1, Γ_1)	广播 (δ_2, Γ_2)	广播 (δ_3, Γ_3)
5	计算 $R := (\delta_1 + \delta_2 + \delta_3)^{-1} \cdot (\Gamma_1 + \Gamma_2 + \Gamma_3)$ 一致性原理如下：		

	$$(\delta_1+\delta_2+\delta_3)^{-1}\cdot(\Gamma_1+\Gamma_2+\Gamma_3)$$ $$=\begin{pmatrix}k_1\gamma_1+\alpha_{1,2}+\alpha_{1,3}+\beta_{1,2}+\beta_{1,3}\\ k_2\gamma_2+\beta_{2,1}+\alpha_{2,1}+\alpha_{2,3}+\beta_{2,3}\\ k_3\gamma_3+\beta_{3,1}+\beta_{3,2}+\alpha_{3,1}+\alpha_{3,2}\end{pmatrix}^{-1}\cdot(\Gamma_1+\Gamma_2+\Gamma_3)$$ $$=\begin{pmatrix}k_1\gamma_1+(\alpha_{1,2}+\beta_{2,1})+(\alpha_{1,3}+\beta_{3,1})+\\ (\alpha_{2,1}+\beta_{1,2})+k_2\gamma_2+(\alpha_{2,3}+\beta_{3,2})+\\ (\alpha_{3,1}+\beta_{1,3})+(\alpha_{3,2}+\beta_{2,3})+k_3\gamma_3\end{pmatrix}^{-1}\cdot(\Gamma_1+\Gamma_2+\Gamma_3)$$ $$=\begin{pmatrix}k_1\gamma_1+(k_1\gamma_2)+(k_1\gamma_3)+\\ (k_2\gamma_1)+k_2\gamma_2+(k_2\gamma_3)+\\ (k_3\gamma_1)+(k_3\gamma_2)+k_3\gamma_3\end{pmatrix}^{-1}\cdot(\gamma_1+\gamma_2+\gamma_3)\cdot G$$ $$=\left((k_1+k_2+k_3)(\gamma_1+\gamma_2+\gamma_3)\right)^{-1}\cdot(\gamma_1\cdot G+\gamma_2\cdot G+\gamma_3\cdot G)$$ $$=(k\gamma)^{-1}\cdot(\gamma\cdot G)$$ $$=k^{-1}\cdot G$$ $$=R$$

	计算 s 概述		
	P_1	P_2	P_3
1	保密输入(k_1,w_1)	保密输入(k_2,w_2)	保密输入(k_3,w_3)
	进行 6 个份额转换协议 MtA		

2.1	$u_{1,2} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_2(w_2)\right\}(k_1 w_2)$	$v_{2,1} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_2(w_2)\right\}(k_1 w_2)$	
2.2	$u_{1,3} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_3(w_3)\right\}(k_1 w_3)$		$v_{3,1} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_3(w_3)\right\}(k_1 w_3)$
2.3	$v_{1,2} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_1(w_1)\right\}(k_2 w_1)$	$u_{2,1} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_1(w_1)\right\}(k_2 w_1)$	
2.4		$u_{2,3} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_3(w_3)\right\}(k_2 w_3)$	$v_{3,2} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_3(w_3)\right\}(k_2 w_3)$
2.5	$v_{1,3} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_1(w_1)\right\}(k_3 w_1)$		$u_{3,1} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_1(w_1)\right\}(k_3 w_1)$
2.6		$v_{2,3} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_2(w_2)\right\}(k_3 w_2)$	$u_{3,2} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_2(w_2)\right\}(k_3 w_2)$
	份额转换协议 MtA 结束		
3	计算 $\sigma_1 := k_1 w_1 + u_{1,2} + v_{1,2} + v_{1,3} + u_{1,3}$	计算 $\sigma_2 := u_{2,1} + v_{2,1} + k_2 w_2 + + u_{2,3} + v_{2,3}$	计算 $\sigma_2 := v_{3,1} + u_{3,1} + u_{3,2} + v_{3,2} + k_3 w_3$
4	计算 $s_1 := mk_1 + r\sigma_1$	计算 $s_2 := mk_2 + r\sigma_2$	计算 $s_3 := mk_3 + r\sigma_3$
5	广播 s_1	广播 s_2	广播 s_3

6

计算 $s := s_1 + s_2 + s_3$

一致性原理如下：

$$
\begin{aligned}
& s_1 + s_2 + s_3 \\
&= (mk_1 + r\sigma_1) + (mk_2 + r\sigma_2) + (mk_3 + r\sigma_3) \\
&= m(k_1 + k_2 + k_3) + r(\sigma_1 + \sigma_2 + \sigma_3) \\
&= m(k_1 + k_2 + k_3) + r\begin{pmatrix} k_1w_1 + u_{1,2} + v_{1,2} + v_{1,3} + u_{1,3} + \\ u_{2,1} + v_{2,1} + k_2w_2 + u_{2,3} + v_{2,3} + \\ v_{3,1} + u_{3,1} + u_{3,2} + v_{3,2} + k_3w_3 \end{pmatrix} \\
&= m(k_1 + k_2 + k_3) + r\begin{pmatrix} k_1w_1 + (u_{1,2} + v_{2,1}) + (u_{1,3} + v_{3,1}) + \\ (u_{2,1} + v_{1,2}) + k_2w_2 + (u_{2,3} + v_{3,2}) + \\ (u_{3,1} + v_{1,3}) + (u_{3,2} + v_{2,3}) + k_3w_3 \end{pmatrix} \\
&= m(k_1 + k_2 + k_3) + r\begin{pmatrix} k_1w_1 + k_1w_2 + k_1w_3 + \\ k_2w_1 + k_2w_2 + k_2w_3 + \\ k_3w_1 + k_3w_2 + k_3w_3 \end{pmatrix} \\
&= m(k_1 + k_2 + k_3) + r(k_1 + k_2 + k_3)(w_1 + w_2 + w_3) \\
&= mk + rkx \\
&= k(m + xr) \\
&= s
\end{aligned}
$$

3.GG18 (3-3)实例

3.1 分布式密钥生成

	P_1	P_2	P_3
1	生成 Paillier 密钥对 (N_1, p_1, q_1) 选择原始随机数 $u_1 \in [1, n-1]$ 计算椭圆曲线随机点、承诺与打开承诺 $U_1 := u_1 \cdot G$ $(KGC_1, KGD_1) = Com(U_1)$	生成 Paillier 密钥对 (N_2, p_2, q_2) 选择原始随机数 $u_2 \in [1, n-1]$ 计算椭圆曲线随机点、承诺与打开承诺 $U_2 := u_2 \cdot G$ $(KGC_2, KGD_2) = Com(U_2)$	生成 Paillier 密钥对 (N_3, p_3, q_3) 选择原始随机数 $u_3 \in [1, n-1]$ 计算椭圆曲线随机点、承诺与打开承诺 $U_3 := u_3 \cdot G$ $(KGC_3, KGD_3) = Com(U_3)$
	广播承诺 KGC_1 与 Paillier 公钥 N_1	广播承诺 KGC_2 与 Paillier 公钥 N_2	广播承诺 KGC_3 与 Paillier 公钥 N_3
2	广播打开承诺 KGD_1	广播打开承诺 KGD_2	广播打开承诺 KGD_3
3	三方均校验另外两方承诺的正确性， 然后计算公共公钥： $PK = U_1 + U_2 + U_3$		
4	选择随机数 $a_1, b_1 \in [1, n-1]$， 构造 2 阶多项式 $p_1(x) = u_1 + x \cdot a_1 + x^2 \cdot b_1$	选择随机数 $a_2, b_2 \in [1, n-1]$， 构造 2 阶多项式 $p_2(x) = u_2 + x \cdot a_2 + x^2 \cdot b_2$	选择随机数 $a_3, b_3 \in [1, n-1]$ 构造 2 阶多项式 $p_3(x) = u_3 + x \cdot a_3 + x^2 \cdot b_3$

	门限为 3	**门限为 3**	**门限为 3**
	构造拉格朗日冗余： 存储 $p_1(1)$，将 $p_1(2), p_1(3)$ 保密发给对应用户 P_2, P_3 计算 **Feldman 校验元组** $A_1 := a_1 \cdot G$ $B_1 := b_1 \cdot G$ 广播 $\{A_1, B_1\}$	**构造拉格朗日冗余：** 存储 $p_2(2)$，将 $p_2(1), p_2(3)$ 保密发给对应用户 P_1, P_3 计算 **Feldman 校验元组** $A_2 := a_2 \cdot G$ $B_2 := b_2 \cdot G$ 广播 $\{A_2, B_2\}$	**构造拉格朗日冗余：** 存储 $p_3(3)$，将 $p_3(1), p_3(2)$ 保密发给对应用户 P_1, P_2 计算 **Feldman 校验元组** $A_3 := a_3 \cdot G$ $B_3 := b_3 \cdot G$ 广播 $\{A_3, B_3\}$
5	拥有保密数据为 $p_1(1) = u_1 + a_1 + b_1$ $p_2(1) = u_2 + a_2 + b_2$ $p_3(1) = u_3 + a_3 + b_3$	拥有保密数据 $p_1(2) = u_1 + 2a_1 + 4b_1$ $p_2(2) = u_2 + 2a_2 + 4b_2$ $p_3(2) = u_3 + 2a_3 + 4b_3$	拥有保密数据 $p_1(3) = u_1 + 3a_1 + 9b_1$ $p_2(3) = u_2 + 3a_2 + 9b_2$ $p_3(3) = u_3 + 3a_3 + 9b_3$
6	进行 **Feldman 校验** $p_1(1) \cdot G == U_1 + A_1 + B_1$ $p_2(1) \cdot G == U_2 + A_2 + B_2$ $p_3(1) \cdot G == U_3 + A_3 + B_3$	进行 **Feldman 校验** $p_1(2) \cdot G == U_1 + 2A_1 + 4B_1$ $p_2(2) \cdot G == U_2 + 2A_2 + 4B_2$ $p_3(2) \cdot G == U_3 + 2A_3 + 4B_3$	进行 **Feldman 校验** $p_1(3) \cdot G == U_1 + 3A_1 + 9B_1$ $p_2(3) \cdot G == U_2 + 3A_2 + 9B_2$ $p_3(3) \cdot G == U_3 + 3A_3 + 9B_3$
7	计算**分片私钥** x_1	计算**分片私钥** x_2	计算**分片私钥** x_3

	$x_1 \coloneqq \sum_{i=1}^{3} p_i(1) \bmod n$ $= \sum_{i=1}^{3} (u_i + a_i + b_i) \bmod n$ $= sk + \sum_{i=1}^{3} (a_i + b_i) \bmod n$	$x_2 \coloneqq \sum_{i=1}^{3} p_i(2) \bmod n$ $= \sum_{i=1}^{3} (u_i + 2a_i + 4b_i) \bmod n$ $= sk + \cdot \sum_{i=1}^{3} (2a_i + 4b_i) \bmod n$	$x_3 \coloneqq \sum_{i=1}^{3} p_i(3) \bmod n$ $= \sum_{i=1}^{3} (u_i + 3a_i + 9b_i) \bmod n$ $= sk + \sum_{i=1}^{3} (3a_i + 9b_i) \bmod n$
8	计算**分片公钥** $X_1 \coloneqq PK + \left(\sum_{i=1}^{3} (a_i + b_i) \right) \cdot G$ 广播分片公钥 X_1	计算**分片公钥** $X_2 \coloneqq PK + \left(\sum_{i=1}^{3} (2a_i + 4b_i) \right) \cdot G$ 广播分片公钥 X_2	计算**分片公钥** $X_3 \coloneqq PK + \left(\sum_{i=1}^{3} (3a_i + 9b_i) \right) \cdot G$ 广播分片公钥 X_3
9	公共公钥 PK 与分片公钥 X_1, X_2, X_3 之间满足**拉格朗日插值校验**（三方均执行） $PK == \lambda_1 \cdot X_1 + \lambda_2 \cdot X_2 + \lambda_3 \cdot X_3$ 根据三方的 Party ID 为 1,2,3，则对应的**拉格朗日插值系数**为 $\lambda_1 = \frac{0-2}{1-2}\frac{0-3}{1-3} = 3, \lambda_2 = \frac{0-1}{2-1}\frac{0-3}{2-3} = -3, \lambda_3 = \frac{0-1}{3-1}\frac{0-2}{3-2} = 1$ 一致性原理如下：		

	$$\begin{aligned}&\lambda_1 \cdot X_1 + \lambda_2 \cdot X_2 + \lambda_3 \cdot X_3 \\ &= 3X_1 - 3X_2 + X_3 \\ &= 3\left(PK + \left(\sum_{i=1}^{3}(a_i + b_i)\right)\cdot G\right) - 3\left(PK + \left(\sum_{i=1}^{3}(2a_i + 4b_i)\right)\cdot G\right) + \left(PK + \left(\sum_{i=1}^{3}(3a_i + 9b_i)\right)\cdot G\right) \\ &= PK\end{aligned}$$ **注释 1：**三方的分片私钥为 x_1, x_2, x_3，能够通过拉格朗日插值计算公共私钥 sk（但是不计算），而是计算**分片签名**。 $$sk == \lambda_1 \cdot x_1 + \lambda_2 \cdot x_2 + \lambda_3 \cdot x_3$$ **注释 2：**上述过程是分布式密钥生成。可以假设存在一个**可信第三方**，**可信第三方**运行一个**公共多项式** $$p(x) = (u_1 + u_2 + u_3) + (a_1 + a_2 + a_3)\cdot x^1 + (b_1 + b_2 + b_3)\cdot x^2 = sk + (a_1 + a_2 + a_3)\cdot x^1 + (b_1 + b_2 + b_3)\cdot x^2$$ 当 $x = 1,2,3$，则计算出**分片私钥** x_1, x_2, x_3，保密发送给各个参与方。

10	zk-Paillier-N 证明私钥不等 $p_1 \neq q_1$ zk-Schnorr 证明知道分片私钥 x_1	zk-Paillier-N 证明私钥不等 $p_2 \neq q_2$ zk-Schnorr 证明知道分片私钥 x_2	zk-Paillier-N 证明私钥不等 $p_3 \neq q_3$ zk-Schnorr 证明知道分片私钥 x_3
11	三方均**校验**其他两方广播过来的 2 个 zk **分析：**①如果少了 zk-Schnorr，则参与方可能不知道分片私钥，不能正确签名； ②如果少了 zk-Paillier-N，则同态加密不安全，参与方能够解密获得其他参与方的分片私钥。		
	最终结果：三方均拥有：公共公钥 PK 、3 个分片公钥 X_1, X_2, X_3 、3 个 Pailler 公钥 N_1, N_2, N_3		
	拥有以下 3 项保密信息：	拥有以下 3 项保密信息：	拥有以下 3 项保密信息：

	1. Paillier 私钥 (p_1, q_1) 2. 原始随机数 u_1（可删除） 3. 分片私钥 x_1	1. Paillier 私钥 (p_2, q_2) 2. 原始随机数 u_2（可删除） 3. 分片私钥 x_2	1. Paillier 私钥 (p_3, q_3) 2. 原始随机数 u_3（可删除） 3. 分片私钥 x_3
	令 $w_1 = \lambda_1 \cdot x_1$	令 $w_2 = \lambda_2 \cdot x_2$	令 $w_3 = \lambda_3 \cdot x_3$
	注释：公共私钥 sk 不出现，满足以下**拉格朗日插值校验** $$sk = \lambda_1 \cdot x_1 + \lambda_2 \cdot x_2 + \lambda_3 \cdot x_3$$ $$= w_1 + w_2 + w_3$$ 其中 w_1, w_2, w_3 称为**分片私钥加性份额。**三个各自基于分片私钥和拉格朗日插值系数计算**分片私钥加性份额**。 3 个用户不重构公共私钥 sk，而是使用私钥加性份额与随机数 k，计算签名加性份额 sig_1, sig_2, sig_3，广播签名加性份额，累加后得到完整签名 $sig \coloneqq sig_1 + sig_2 + sig_3$		

3.2 三个用户签名

	P_1	P_2	P_3
	注释：以下开始计算公共随机点 *R* **三方各自输入输入 2 份保密的原始随机数** $(k_1,k_2,k_3),(\gamma_1,\gamma_2,\gamma_3)$		
1	选择两个原始随机数 $(k_1,\gamma_1)\in[1,n-1]$ 计算椭圆曲线随机点、承诺与打开承诺 $\Gamma_1 \coloneqq \gamma_1 \cdot G$ $(C_1,D_1)=Com(\Gamma_1)$	选择两个原始随机数 $(k_2,\gamma_2)\in[1,n-1]$ 计算椭圆曲线随机点、承诺与打开承诺 $\Gamma_2 \coloneqq \gamma_2 \cdot G$ $(C_2,D_2)=Com(\Gamma_2)$	选择两个原始随机数 $(k_3,\gamma_3)\in[1,n-1]$ 计算椭圆曲线随机点、承诺与打开承诺 $\Gamma_3 \coloneqq \gamma_3 \cdot G$ $(C_3,D_3)=Com(\Gamma_3)$
	广播承诺 C_1	广播承诺 C_2	广播承诺 C_3
2	**以下进行 6 个份额转换协议 MtA**		
	$\alpha_{1,2} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_2(\gamma_2)\right\}(k_1\gamma_2)$	$\beta_{2,1} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_2(\gamma_2)\right\}(k_1\gamma_2)$	
	$\alpha_{1,3} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_3(\gamma_3)\right\}(k_1\gamma_3)$		$\beta_{3,1} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_3(\gamma_3)\right\}(k_1\gamma_3)$
	$\beta_{1,2} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_1(\gamma_1)\right\}(k_2\gamma_1)$	$\alpha_{2,1} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_1(\gamma_1)\right\}(k_2\gamma_1)$	

		$\alpha_{2,3} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_3(\gamma_3)\right\}(k_2\gamma_3)$	$\beta_{3,2} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_3(\gamma_3)\right\}(k_2\gamma_3)$
	$\beta_{1,3} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_1(\gamma_1)\right\}(k_3\gamma_1)$		$\alpha_{3,1} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_1(\gamma_1)\right\}(k_3\gamma_1)$
		$\beta_{2,3} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_2(\gamma_2)\right\}(k_3\gamma_2)$	$\alpha_{3,2} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_2(\gamma_2)\right\}(k_3\gamma_2)$
	注释：以下进行 6 个份额转换协议 MtA **三方各自输入输入 1 份保密原始随机数 k_1, k_2, k_3 和 1 份分片私钥加性份额 w_1, w_2, w_3**		
	保密输入为 $(k_1, w_1) \in [1, n-1]$	保密输入为 $(k_2, w_2) \in [1, n-1]$	保密输入为 $(k_3, w_3) \in [1, n-1]$
	$u_{1,2} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_2(w_2)\right\}(k_1w_2)$	$v_{2,1} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_2(w_2)\right\}(k_1w_2)$	
	$u_{1,3} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_3(w_3)\right\}(k_1w_3)$		$v_{3,1} \leftarrow \left\{P_1(k_1) \overset{\text{MtA}}{\rightleftarrows} P_3(w_3)\right\}(k_1w_3)$
	$v_{1,2} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_1(w_1)\right\}(k_2w_1)$	$u_{2,1} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_1(w_1)\right\}(k_2w_1)$	

		$u_{2,3} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_3(w_3)\right\}(k_2 w_3)$	$v_{3,2} \leftarrow \left\{P_2(k_2) \overset{\text{MtA}}{\rightleftarrows} P_3(w_3)\right\}(k_2 w_3)$
	$v_{1,3} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_1(w_1)\right\}(k_3 w_1)$		$u_{3,1} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_1(w_1)\right\}(k_3 w_1)$
		$v_{2,3} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_2(w_2)\right\}(k_3 w_2)$	$u_{3,2} \leftarrow \left\{P_3(k_3) \overset{\text{MtA}}{\rightleftarrows} P_2(w_2)\right\}(k_3 w_2)$
	注释：份额转换协议 MtA 结束		
3	计算 $\delta_1 := k_1\gamma_1 + \alpha_{1,2} + \alpha_{1,3} + \beta_{1,2} + \beta_{1,3} \bmod n$ $\sigma_1 := k_1 w_1 + u_{1,2} + v_{1,2} + v_{1,3} + u_{1,3} \bmod n$	计算 $\delta_2 := k_2\gamma_2 + \beta_{2,1} + \alpha_{2,1} + \alpha_{2,3} + \beta_{2,3} \bmod n$ $\sigma_2 := u_{2,1} + v_{2,1} + k_2 w_2 + + u_{2,3} + v_{2,3} \bmod n$	计算 $\delta_3 := k_3\gamma_3 + \beta_{3,1} + \beta_{3,2} + \alpha_{3,1} + \alpha_{3,2} \bmod n$ $\sigma_3 := v_{3,1} + u_{3,1} + u_{3,2} + v_{3,2} + k_3 w_3 \bmod n$
	广播 δ_1	广播 δ_2	广播 δ_3
	三方均**计算** $\delta^{-1} := (k\gamma)^{-1} \bmod n = (\delta_1 + \delta_2 + \delta_3)^{-1} \bmod n$		
4	广播打开承诺 D_1 和 zk-Sigma 证明 $ZK\{\gamma_1 \mid \Gamma_1 := \gamma_1 \cdot G\}$	广播打开承诺 D_2 和 zk-Sigma 证明 $ZK\{\gamma_2 \mid \Gamma_2 := \gamma_2 \cdot G\}$	广播打开承诺 D_3 和 zk-Sigma 证明 $ZK\{\gamma_3 \mid \Gamma_3 := \gamma_3 \cdot G\}$
	三方均校验打开承诺 D_1, D_2, D_3 和 Schnorr 零知识证明的正确性，		

<table>
<tr><td></td><td colspan="3">然后计算公共随机点 $R := \delta^{-1} \cdot (\Gamma_1 + \Gamma_2 + \Gamma_3)$

一致性原理如下：

$(\delta_1 + \delta_2 + \delta_3)^{-1} \cdot (\Gamma_1 + \Gamma_2 + \Gamma_3)$
$= \begin{pmatrix} k_1\gamma_1 + \alpha_{1,2} + \alpha_{1,3} + \beta_{1,2} + \beta_{1,3} \\ k_2\gamma_2 + \beta_{2,1} + \alpha_{2,1} + \alpha_{2,3} + \beta_{2,3} \\ k_3\gamma_3 + \beta_{3,1} + \beta_{3,2} + \alpha_{3,1} + \alpha_{3,2} \end{pmatrix}^{-1} \cdot (\Gamma_1 + \Gamma_2 + \Gamma_3)$
$= \begin{pmatrix} k_1\gamma_1 + k_1\gamma_2 + k_1\gamma_3 + \\ k_2\gamma_1 + k_2\gamma_2 + k_2\gamma_3 + \\ k_3\gamma_1 + k_3\gamma_2 + k_3\gamma_3 \end{pmatrix}^{-1} \cdot (\Gamma_1 + \Gamma_2 + \Gamma_3)$
$= \left((k_1 + k_2 + k_3)(\gamma_1 + \gamma_2 + \gamma_3)\right)^{-1} \cdot (\gamma_1 \cdot G + \gamma_2 \cdot G + \gamma_3 \cdot G)$
$= (k\gamma)^{-1} \cdot (\gamma \cdot G)$
$= k^{-1} \cdot G$
$= R$
取公共随机点 R 的横坐标，然后模 n，则得到签名中的 r</td></tr>
<tr><td></td><td colspan="3">$m := hash(M) \bmod n$ ，以下开始计算 s</td></tr>
<tr><td>5</td><td>计算 $s_1 := mk_1 + r\sigma_1$</td><td>计算 $s_2 := mk_2 + r\sigma_2$</td><td>计算 $s_3 := mk_3 + r\sigma_3$</td></tr>
<tr><td>5A</td><td>选择原始随机数 $l_1, \rho_1 \in [1, n-1]$，

计算 $V_1 := s_1 \cdot R + l_1 \cdot G, \Upsilon_1 := \rho_1 \cdot G$</td><td>选择原始随机数 $l_2, \rho_2 \in [1, n-1]$

计算 $V_2 := s_2 \cdot R + l_2 \cdot G, \Upsilon_2 := \rho_2 \cdot G$</td><td>选择原始随机数 $l_3, \rho_3 \in [1, n-1]$

计算 $V_3 := s_3 \cdot R + l_3 \cdot G, \Upsilon_3 := \rho_3 \cdot G$</td></tr>
</table>

	计算承诺与打开承诺 $(\hat{C}_1, \hat{D}_1) = Com(V_1, \Upsilon_1)$ 广播承诺 $\hat{C}_1$	计算承诺与打开承诺 $(\hat{C}_2, \hat{D}_2) = Com(V_2, \Upsilon_2)$ 广播承诺 $\hat{C}_2$	计算承诺与打开承诺 $(\hat{C}_3, \hat{D}_3) = Com(V_3, \Upsilon_3)$ 广播承诺 $\hat{C}_3$
5B	广播打开承诺 $\hat{D}_1$ 与 zk-Sigma*和 zk-Sigma 证明 $ZK\left\{ s_1, l_1, \rho_1 \middle\| \begin{matrix} V_1 = s_1 \cdot R + l_1 \cdot G, \\ \Upsilon_1 = \rho_1 \cdot G \end{matrix} \right\}$	广播打开承诺 $\hat{D}_2$ 与 zk-Sigma*和 zk-Sigma 证明 $ZK\left\{ s_2, l_2, \rho_2 \middle\| \begin{matrix} V_2 = s_2 \cdot R + l_2 \cdot G, \\ \Upsilon_2 = \rho_2 \cdot G \end{matrix} \right\}$	广播打开承诺 $\hat{D}_3$ 与 zk-Sigma*和 zk-Sigma 证明 $ZK\left\{ s_3, l_3, \rho_3 \middle\| \begin{matrix} V_3 = s_3 \cdot R + l_3 \cdot G, \\ \Upsilon_3 = \rho_3 \cdot G \end{matrix} \right\}$
	三方均校验另外两方的打开承诺 $\hat{D}_1, \hat{D}_2, \hat{D}_3$ 与零知识证明的正确性， 然后计算 $\begin{matrix} V := (-m) \cdot G + (-r) \cdot PK + V_1 + V_2 + V_3 \\ \Upsilon := \Upsilon_1 + \Upsilon_2 + \Upsilon_3 \end{matrix}$		
5C	计算 $\begin{matrix} \Omega_1 := \rho_1 \cdot V, \\ \Psi_1 := l_1 \cdot \Upsilon \end{matrix}$	计算 $\begin{matrix} \Omega_2 := \rho_2 \cdot V, \\ \Psi_2 := l_2 \cdot \Upsilon \end{matrix}$	计算 $\begin{matrix} \Omega_3 := \rho_3 \cdot V, \\ \Psi_3 := l_3 \cdot \Upsilon \end{matrix}$
	计算承诺与打开承诺 $(\tilde{C}_1, \tilde{D}_1) = Com(\Omega_1, \Psi_1)$	计算承诺与打开承诺 $(\tilde{C}_2, \tilde{D}_2) = Com(\Omega_2, \Psi_2)$	计算承诺与打开承诺 $(\tilde{C}_3, \tilde{D}_3) = Com(\Omega_3, \Psi_3)$
	广播承诺 $\tilde{C}_1$	广播承诺 $\tilde{C}_2$	广播承诺 $\tilde{C}_3$

<table>
<tr><td rowspan="2">5D</td><td>广播打开承诺 $\tilde{D}_1$</td><td>广播打开承诺 $\tilde{D}_2$</td><td>广播打开承诺 $\tilde{D}_3$</td></tr>
<tr><td colspan="3">三方均校验另外两方的打开承诺 $\tilde{D}_1,\tilde{D}_2,\tilde{D}_3$ 的正确性，
然后校验 $\Omega_1+\Omega_2+\Omega_3 == \Psi_1+\Psi_2+\Psi_3$
一致性过程如下：
$$\Omega_1+\Omega_2+\Omega_3=(\rho_1+\rho_2+\rho_3)\cdot\big((-m)\cdot G+(-r)\cdot PK+V_1+V_2+V_3\big)$$
$$=(-m\rho)\cdot G+(-r\rho)\cdot PK+\rho(V_1+V_2+V_3)$$
$$=(-m\rho)\cdot G+(-r\rho)\cdot PK+\rho(s_1\cdot R+l_1\cdot G+s_2\cdot R+l_2\cdot G+s_3\cdot R+l_3\cdot G)$$
$$=(-m\rho)\cdot G+(-r\rho)\cdot PK+\rho(s\cdot R+l\cdot G)$$
$$=(-m\rho)\cdot G+(-r\rho)\cdot PK+\rho s\cdot R+l\rho\cdot G$$
$$=-\rho\big(m\cdot G+r\cdot PK-sR\big)+l\rho\cdot G$$
$$\Psi_1+\Psi_2+\Psi_3=(l_1+l_2+l_3)\cdot\Upsilon=l(\Upsilon_1+\Upsilon_2+\Upsilon_3)=lp\cdot G$$
红色部分 $m\cdot G+r\cdot PK-sR=0$，在密文状态下确保了 ECDSA 的验证是正确的。</td></tr>
<tr><td rowspan="3">5E</td><td>广播 s_1</td><td>广播 s_2</td><td>广播 s_3</td></tr>
<tr><td colspan="3">三方均计算 $s:=s_1+s_2+s_3$</td></tr>
<tr><td colspan="3">三方均校验签名一致性 (r,s)</td></tr>
</table>

lynndell 新火科技 密码学专家 lynndell2010@gmail.com