BLS 签名和基于 BLS 签名的门限签名

马宇峰 溪塔科技 2019-12-13



在区块链的整个体系中大量使用了密码学算法，密码学是保证区块链安全的基石，而区块链的广泛应用也推进了密码学的发展。2003 年 Boneh 和 Franklin 提出了身份基加密，从此基于双线性映射的密码学算法走向了人们的视野，并且成为了密码学新兴的研究方向。

今天我们要学习的 BLS 签名算法是基于双线性映射构造的，阅读本文将带领你步入基于双线性映射的密码学算法的大门。

如果您对技术感兴趣，欢迎关注我们开源代码库（点个 Star 呗）：
GitHub：https://github.com/citahub

2003 年 Boneh 和 Franklin 提出了身份基加密[1]，从此基于双线性映射(也称为 pairing 运算/对运算)的密码学算法走向了人们的视野，并且成为密码学新兴的研究方向。今天我们要学习的 BLS 签名算法[2]就是基于双线性映射构造的，在给出具体的签名算法之前，我们需要学习一下什么是双线性映射。

双线性映射

定义1 双线性映射:设群 G_1,G_2,G_T 是阶为素数 p 的乘法循环群，如果映射 $e: G_1 \times G_2 \rightarrow G_T$ 满足以下性质，就称映射 e 为双线性映射:

(1) 双线性:任意元素 $u \in G_1$， $v \in G_2$,任意元素 $a,b \in Z_p$ （Z_p 指的是{0,1,...,p-1}),都有 $e(u^a,v^b)=e(u,v)^{ab}$;

(2) 非退化性:存在元素 $m \in G_1$， $n \in G_2$ 使得 e(m,n)≠1;

(3) 可计算性:任意元素 $u,v \in G_1$，存在有效算法计算 e(u,v)。

可能有些读者还不知道群，循环群，乘法循环群，群的阶是什么意思，因此我下面介绍一下这些术语的定义，已经熟悉这些知识的读者可以直接跳过定义 2，定义 3 直接阅读后续签名算法部分。

定义 2 群[3]:群是一个集合 G 和其上的二元运算 ，满 以下条件:

(1)封闭性:对于任意 x,y∈G ,有 ;

(2)结合率:对于任意的 x,y,z∈G ， 恒有 ;

(3)存在单位元:存在 e∈G ， 对于 G 中任意元素 x ,满足 ，e称为单位元，也称为幺元;

(4)存在逆元:对于任意 x∈G ， 都存在一个元素 h∈G ,满足 ， 这样的元素 h 称为 x 的逆元。

当二元运算为已知时，简单的将集合 G 称为群。

定义 3 循环群[3]:若群 G 中存在元素 g 满足对群 G 中任意元素 f，都存在整数 t 使得 ，则群 G 是循环群，g 是群 G 的**生成元**，若群 G 的阶为素数，则 G 中每一个非单位元元素都是生成元。群元素的个数称为**阶**。二元运算为模乘法的循环群，称为**乘法循环群**。

有了以上知识，应该可以理解 BLS 签名算法了，这个算法包括:初始化，密钥生成，签名，验证，四个部分，具体构造如下:

1.初始化G_1, G_2 是阶为 p 的乘法循环群，生成元分别是 g_1, g_2 , e 是双线性映射:$G_1 * G_2 \rightarrow G_T$, 安全 hash 函数:h:{0,1}*-> G_1 , 公开参数是(G_1, G_2, G_T, e, g_1, g_2,p,h)

2.密钥生成

选择一个随机数 $x \in Z_P$,计算 ,私钥 SK=x,公钥 PK=v

3.签名

对消息 M 签名

$$\sigma = h(M)^x$$

4.验证

$$e\left(\sigma, g_2\right) ? = e(h(M), v)$$

门限签名的通俗理解是:在一个签名者群体中，有超过 t(门限)个签名者对一条消息进行签名就可以得到这个群体对这条消息的签名，这个签名是唯一的(即超过 t 个签名者的不同子集生成的签名是一样的)且任何人都可以验证。由门限签名的性质可知，少于 t 个签名者的群体是得不到群体签名的，因此在一个由 n 个节点组成的区块链系统中当恶意节点的数量小于 t 时，可以利用门限签名来做**可验证随机函数(Verifiable Random Functions)**。

随机数在共识领域有着重要的应用，**从本质上讲，共识问题就是随机选择出块人问题**，并且这个随机性不能被提前预测，不能被操纵，并且需要可以被全网验证。PoW 方案通过计算满足一定条件的 Hash 来选取出块人，算力大的节点被选中的概率高，为了获取记账权，需要投入更多的算力。为了解决 PoW 方案算力浪费的问题，PoS 方案根据每个节点所拥有份额多少来随机选择出块人，份额越多的节点被选中的概率越大，门限签名可以作为 PoS 方案的随机性来源。

拉格朗日差值法

现在可以在 BLS 签名方案的基础上通过使 Shamir 秘密分享方法[4]来构造门限签名算法，为了便于理解，在给出具体构造算法之前，我先介绍一下拉格朗日差值法，这是 Shamir 秘密分享方法的核心。

已知一条 n-1 次曲线上的 n 个点 (x_1,y_1), (x_2,y_2), ..., (x_n,y_n) , 我们可以使用拉格朗日插值法来恢复这条曲线方程，具体如下:

现在改变一下问题，想要实现 t-of-n 的秘密分享。t=3,n=4 时，先任意选择一条 t-1=2 次曲线，以这条曲线在 x=0 处的值作为秘密。然后任意取出曲线上的n=4个点 (1,2),(2,5), (3,10), (4,17) 后，任选其中三个点后恢复出来 t-1=2 次曲线是一样的。因此，我可以将这四个点（横坐标分别是 1、2、3、4）发送给四个不同的人，这样就是一个 3-of-4 的秘密分享了。

有了以上知识，应该可以理解基于 BLS 签名的门限签名算法 ，这个算法包括:初始化，密钥生成，签名，验证，四个部分，具体构造如下:

1.初始化

G_1, G_2 是阶为 p 的乘法循环群，生成元分别是 g_1, g_2 , e 是双线性映射:$G_1 * G_2 \to G_T$, 安全 hash 函数:h:{0,1} -> G_1 , 公开参数是(G1, G2, GT, e, g1, g2,p,h)

2.密钥生成

这步由密钥生成中心完成

a.密钥生成中心选择随机数 $x \in Z_P$, 计算 算, 系统主私钥 MSK = x, 系统主公钥 MPK = v

b.随机选择一个 Z_P 上的 t-1 阶多项式 P, 满足 P(0)=x, 计算 $x_i = P(i)$, , 签名者 u_i 的私钥为 x_i, 公钥为 v_i

c.公开 MPK 及所有用户的公钥

3.签名

设对消息 M 签名

a.用户 u_i 计算自己的签名 ，然后将 σ_i 广播出去

b.用户 u_i 收到了来自 u_j 的签名 σ_j 后，首先验证签名的正确性:

, 如果签名正确则记录下来

c.待收集到 t 个不同用户的正确签名后，用户计算完整的签名:

由拉格朗日差值公式可知，任意 t 个用户所产生的完整签名相同

4.验证

$$e\left(\sigma, g_2\right) ? = e(h(M), MPK)$$

然而这个算法由于使用了密钥生成中心这样一个中心化机构，因此并不能直接 用来做可验证随机函数，我们需要构造一个分布式密钥生成的门限签名算法，具体算法我会在下篇文章给出，感兴趣的读者可以先想想怎么做。(tips: 多个多项式之和作为秘密多项式)

ref

[1] Boneh D, Franklin M. Identity based encryption from the Weil pairing[J]. Siam Journal on Computing, 2003, 32(3):213-229.

[2] Dan B, Lynn B, Shacham H. Short Signatures from the Weil Pairing[C]// International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2001:514-532.

[3] 陈恭亮. 信息安全数学基础[M]. 清华 学出版社, 2004.

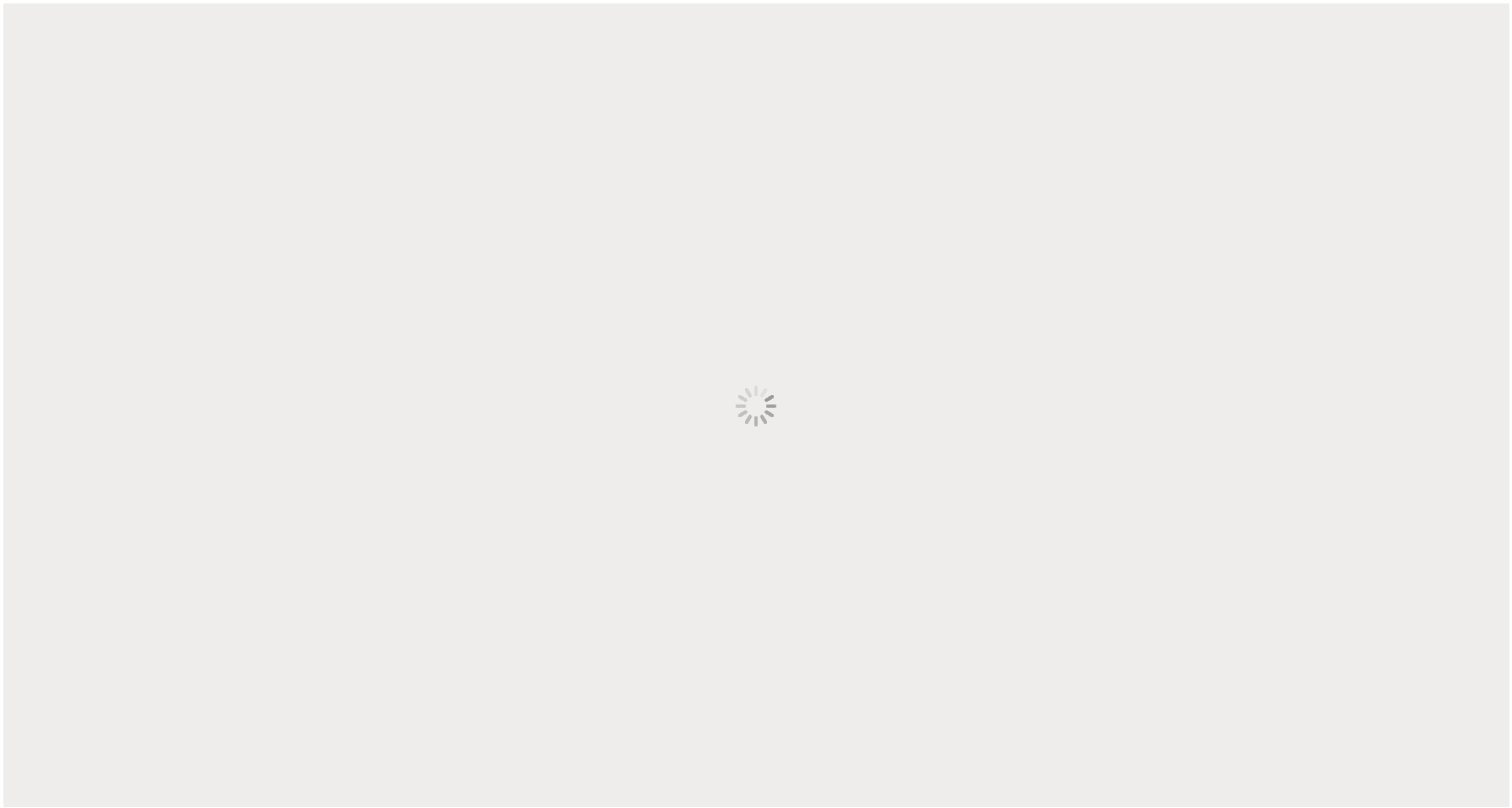
[4] Shamir A. How to share a secret[J]. Communications of the ACM, 1979, 22(11): 612-613.

推荐阅读

- *拜占庭容错的代表 PBFT*
- *BLS 签名和基于 BLS 签名的门限签名*
- *密码学入门系列（一）*
- *CITA架构与云计算服务*

- *白话零知识证明（一）*
- *白话零知识证明（二）*
- *Authenticated Dynamic Dictionaries in Ethereum*
- *区块链与共同知识库*
- *P2P 网络思考*
- *智能合约的一种设计结构*

如果您对技术感兴趣，欢迎关注我们开源代码库（点个 Star 呗）：
GitHub：https://github.com/citahub



阅读原文