区块链基础及应用 2022

问题 1：多元 Merkle 树。Alice 可以使用二叉 Merkle 树来提交的一组元素 S =（T_1, ..., T_n），之后她可以向 Bob 证明 S [i] = T_i，每个证明最多包含$\lceil \log_2 n \rceil$ 个哈希值。对 S 的承诺是单一的哈希值。在这个问题中，请你解释如何使用 k 叉树来做同样的事情，也就是说，每个非叶节点最多可以有 k 个子节点。每个非叶节点的哈希值是其所有子结点的值的哈希值。

a.假设 S ＝（T_1,...，T_9）。解释 Alice 如何使用三叉 Merkle 树计算对 S 的承诺（即 k = 3）。Alice 如何向 Bob 证明 T_4 在 S 中，即哪些值被包含在证明中？

b.假设 S 包含 n 个元素。S [i] = T_i 的证明长度是多大？用 n 和 k 的函数表示。

c.对于较大的 n 值，如果我们想最小化证明的大小，最好使用二叉 Merkle 树还是三叉 Merkle 树？为什么？

问题 2：

考虑到 Bitcoin 的 Multisig 方法中指令有一些 Bug, 以及为了节省计算、存储开销，我们考虑替代性的方案。假设$G = \{1, g, g^2, \dots, g^{q-1}\}$是一个有限循环群，阶数为素数q, 生成元为g. 设H是一个哈希函数， $H: M \to G$. 在一个 BLS 型的签名方案中，随机选取私钥$x \in 0,1,\dots,q-1$ 相应的公钥为 $pk = g^x \in G$. 消息$m \in M$ 的签名记为$\sigma = H(m)^x \in G$.

(1) 考虑 3-3 的签名方案。为了保护签名私钥, Bob 把x分成 3 份按照如下的方法分发出去： 选择 3 个随机数$x_1, x_2, x_3 \in 0,1,\dots,q-1$使得$x = x_1 + x_2 + x_3 \ mod\ q$. 接下来 Bob 销毁 x 的记录。为了对消息m签名, 每一个参与者计算 $\sigma_i = H(m)^{x_i}$ ，i = 1,2,3; 然后把部分签名送给 Bob. 请你解释 Bob 如何能够从这三个部分签名中获取对消息 m 的签名。

(2) 现在我们推广到 3-5 的签名方案（也就是 5 个人中有 3 个人同意即可）设这 5 方为 P1,P2,P3,P4,P5. 就像 Bitcoin 的 Multisig 方法，我们希望签名能够确认是哪三个参与者签署了该交易。Bob 采用如下的流程，他生成了$\binom{5}{3} = 10$个公钥$pk_i = g^{x_i}, i = 1,2,\dots,10$, 其中一个公钥对应一个三方参与者的集合。比如$pk_1$对应子集合$\{P_1, P_2, P_3\}$, 又比如$pk_9$对应子集合$\{P_2, P_4, P_5\}$. 接下来，Bob 与(1)中一样，对于$i = 1,2,\dots,10$, 他把私钥$x_i$分成$x_i = x_{i,1} + x_{i,2} + x_{i,3}\ mod\ q$, 然后， Bob 把分片$x_{i,j}$分配给相应的参与方。比如对于$pk_9$, P2 得到了$x_{9,1}$, P4

得到了$x_{9,2}$, P5 得到了$x_{9,3}$. 当 3 方想要签名消息 m 时， 他们采用分配得到的私钥分片来签名消息。比如P_2, P_4, P_5将分别采用获得的x_9分片$x_{9,1}, x_{9,2}, x_{9,3}$来签名。为了取代 Bitcoin 中的 Multisig 相关指令，资金交易中将会创建一个没有花费的交易输出(UTXO), 其中包含 10 个叶子节点$pk_1, pk_2, \ldots, pk_{10}$的 Merkle tree 的 Merkle root。为了花费该交易输出，请问需要包含哪些签名数据信息？

(3) 设 Merkle tree 中采用了 SHA256 (32 字节 哈希)，每一个 ECDSA 公钥是 32 字节，而设每一个传统的 ECDSA 数字签名为 64 字节，BLS 数字签名、BLS 公钥是 48 字节。请你考虑，相比 3-5 Multisig 赎回交易的输入数据大小，(2) 中方案是否会耗费更少的字节数？ 另外，结论是否对一般的 t-n (t<n)的签名方案仍然成立？请给出你的计算分析过程。