MuSig2：简洁的 2 轮 Schnorr 多签名

sammyne

2021 年 2 月 10 日

1 背景

1.1 对旧版 MuSig 的攻击

2019 年 Drijvers 等人 [DEF+19] 发现了一种对旧版 MuSig [MPSW18a] 等两轮模式多签名的攻击方法。攻击的底层原理是解决泛化生日问题的 Wagner 算法 [Wag02]。

定义 1.1 (泛化生日问题)**.** *给定常量 $t \in \mathbb{Z}_p$，整数 k_m 以及一个随机预言机 $H : \mathbb{Z}_p \to \{0,1\}^n$，通过 k_m 次查询找出满足 $\sum_{k=1}^{k_m} H(q_k) = t$ 的集合 $\{q_1, \ldots, q_{k_m}\}$。问题的解决复杂度，$k_m = 1$ 时等价于找出哈希原像，$k_m = 2$ 时等价于找出哈希碰撞，k_m 增大时，问题难度会神奇地变得更低。Wagner 等人 [Wag02] 给出一个在不限定 k_m 条件下的次指数级别算法。*

Wagner 算法的具体攻击流程如下：敌手同时开启 k_m 个签名会话，会话过程敌手扮演公钥为 $X_2 = g^{x_2}$ 的签名方，从公钥为 $X_1 = g^{x_1}$ 的诚实签名方获得共 k_m 个 nonces $R_1^1, \ldots, R_1^{k_m}$。令 $\tilde{X} = X_1^{a_1} X_2^{a_2}$（其中 $a_i = H(\langle X_1, X_2 \rangle, X_i)$ [MPSW18a]）表示聚合所得公钥。给定伪造的消息 m^*，敌手计算 $R^* = \prod_{k=1}^{k_m} R_1^{(k)}$，然后借助 Wagner 算法找出满足以下条件的 $R_2^{(k)}$

$$\sum_{k=1}^{k_m} \underbrace{H_{sig}(\tilde{X}, R_1^{(k)} R_2^{(k)}, m^{(k)})}_{c^{(k)}} = \underbrace{H_{sig}(\tilde{X}, R^*, m^*)}_{c^*} \tag{1}$$

诚实签名方收到 $R_2^{(k)}$ 后会反馈 $s_1^{(k)} = r_1^{(k)} + c^{(k)} \cdot a_1 x_1$。令 $r^* = \sum_{k=1}^{k_m} r_1^{(k)} = DL(R^*)$，敌手借此可得

$$s_1^* = \sum_{k=1}^{k_m} s_1^{(k)} = \sum_{k=1}^{k_m} r_1^{(k)} + \left(\sum_{k=1}^{k_m} c^{(k)} \right) \cdot a_1 x_1 = r^* + c^* \cdot a_1 x_1$$

然后，敌手就可以进一步基于 s_1^* 构造

$$s^* = s_1^* + c^* \cdot a_2 x_2 = r^* + c^* \cdot (a_1 x_1 + a_2 \cdot x_2)$$

因此，(R^*, s^*) 即为 m^* 的合法签名，其中签名哈希为 $c^* = H_{sig}(\tilde{X}, R^*, m^*)$。这里伪造的消息只对 X_1 和 X_2 聚合所得公钥 $\tilde{X}$ 合法。显然，只要是把诚实签名方的公钥 X_1 和敌手的公钥集合聚合，攻击只需稍作调整即可伪造出合法的消息。

Wagner 算法的 **攻击复杂度**为 $O(k_m 2^{\log_2(p)/(1+\lfloor \log_2(k_m) \rfloor)})$。虽然是次指数级别（非多项式级别），攻击对于常用参数和足够大的 k_m 是可操作的。例如，对于椭圆曲线常用的素数 $p \approx 2^{256}$，$k_m = 128$ 能够将攻击复杂度降低到约 2^{39} 次操作，普通硬件都能实施此攻击。如果敌手能够开启更多会话，Benhamouda 等人 [BLL+20] 改进的多项式时间级别的攻击使用 $k_m > \log_2 p$ 就能实现 $O(k_m \log_2 p)$ 攻击复杂度，实际操作性贼强。

2 MuSig2 方案

相关论文参见 [NRS20,jon20]。

2.1 背景

Drijvers [DEF+19] 或 Benhamouda [BLL+20] 等人的攻击方式均是通过控制聚合的 nonce $R_1^{(k)}R_2^{(k)}$（等式 1 的左手边）来控制签名的哈希。由于所有签名方都在第一轮交互结束时知道聚合的 nonce，不像 [MPSW18b] 添加额外承诺轮的情况下防止敌手控制左手边的聚合 nonce 有点难。

左边不好弄的话，不妨换个思路，我们允许敌手控制等式左手边，但是防止他们控制等式的右手边。

MuSig2 方案的巧妙点在于让每个签名方 i 发送一个 nonces 列表 $R_{i,1},\ldots,R_{i,\nu}$ $(\nu \geq 2)$，以它们的线性组合 $\hat{R}_i = \prod_{j=1}^{\nu} R_{i,j}^{b_j}$ 作为自己最终的 nonce，而不是之前的单个 nonce R_i，其中 $b_j = H_{non}(j, \tilde{X}, (\prod_{i=1}^{n} R_{i,1}, \ldots, \prod_{i=1}^{n} R_{i,\nu}), m)$，$H_{non} : \{0,1\}^* \to \mathbb{Z}_p$ 是一个可看做随机预言机的哈希函数。

这样一来，每次敌手尝试不同的 $R_2^{(k)}$，系数 $b_1^{(k)},\ldots,b_\nu^{(k)}$ 都会随之变化，进而改变诚实签名方的 $\hat{R}_1 = \prod_{j=1}^{\nu} R_{1,j}^{b_j}$，最终改变等式 (1) 右手边的 $R^* = \prod_{k=1}^{k_m} \hat{R}_1^{(k)}$。这也就确保了等式右手边不再是常量，破坏掉泛化生日问题的必要前提条件，Wagner 算法也就不再适用。

关于 $\nu = 1$ 的情形不可行的具体原因分析如下（尚未搞懂）。

既然这样，是否可以回退到单个 nonce 的情况呢（$\nu = 1$）–只依赖系数 b_1？然而，敌手还是可以借助计算以下等式抵消这个变换的效果

$$\sum_{k=1}^{k_m} \underbrace{\frac{H_{sig}(\tilde{X}, (R_1^{(k)})^{b_1^{(k)}}(R_2^{(k)})^{b_1^{(k)}}, m^{(k)})}{b_1^{(k)}}}_{c^{(k)}} = \underbrace{H_{sig}(\tilde{X}, R^*, m^*)}_{c^*} \tag{2}$$

Given

- the fake nonce $R^* = \prod_{k=1}^{k_{max}} R_1^{(k)}$, w.r.t $r^* = \sum_{k=1}^{k_{max}} r_1^{(k)} = DL(R^*)$
- the co-signature by the honest signer as $s_1^{(k)} = r_1^{(k)} b_1^{(k)} + c^{(k)} a_1 x_1$

We have

$$\begin{aligned}
s_1^{(k)} &= r_1^{(k)} b_1^{(k)} + c^{(k)} a_1 x_1 && (3)\\
\Rightarrow s_1^* &= \sum_{k=1}^{k_{max}} \frac{s_1^{(k)}}{b_1^{(k)}} && (4)\\
&= \sum_{k=1}^{k_{max}} \frac{r_1^{(k)} b_1^{(k)} + c^{(k)} a_1 x_1}{b_1^{(k)}} && (5)\\
&= \sum_{k=1}^{k_{max}} \frac{r_1^{(k)} b_1^{(k)}}{b_1^{(k)}} + \sum_{k=1}^{k_{max}} \frac{c^{(k)} a_1 x_1}{b_1^{(k)}} && (6)\\
&= \sum_{k=1}^{k_{max}} r_1^{(k)} + a_1 x_1 \sum_{k=1}^{k_{max}} \frac{c^{(k)} a_1 x_1}{b_1^{(k)}} && (7)\\
&= r^* + c^* a_1 x_1 && (8)\\
\Rightarrow s^* &= s_1^* + c^* a_2 x_2 && (9)\\
&= r^* + c^*(a_1 x_1 + a_2 x_2) && (10)
\end{aligned}$$

Therefore,

$$g^{s^*} = g^{r^*} + g^{c^*(a_1x_1 + a_2x_2)} = R^* + (g^{a_1x_1} \cdot g^{a_2x_2})^{c^*} = R^* + ({}^{a_1x_1} \cdot g^{a_2x_2})^{c^*} \tag{11}$$

renders (R^*, s^*) a valid but forgery signature.

这样就解释了 $\nu = 2$ 的必要性，我们后续会证明固定 $b_1 = 1$（随机化其余系数 $b_2,\ldots,b_\nu$）是一个优化技巧，且不会损害安全性。

2.2 具体算法

2.2.1 签名

设定参数 给定群 $(\mathbb{G}, p, g)$，以及三个 $\{0,1\}^*$ 到 $\mathbb{Z}_p$ 的哈希函数。

生成密钥 生成随机私钥 $x \leftarrow_\$ \mathbb{Z}_p$，计算相应公钥 $X = g^x$。

生成 nonce 令 (x_1, X_1) 表示特定签名方的公私钥对。对于 $j \in \{1, \ldots, \nu\}$，签名方生成随机数 $r_{1,j} \leftarrow \mathbb{Z}_p$，计算 $R_{1,j} = g^{r_{1,j}}$，向其他所有签名方广播 $(R_{1,1}, \ldots, R_{1,\nu})$。

生成签名碎片 给定消息 m，其他签名方的公钥为 $X_2, \ldots, X_n$，令 $L = \{X_1, \ldots, X_n\}$ 表示签名过程涉及的所有公钥。对于 $i \in \{1, \ldots, n\}$，签名方计算 $a_i = H_{agg}(L, X_i)$，然后计算聚合公钥 $\tilde{X} = \prod_{i=1}^n X_i^{a_i}$。一旦**收齐**其他签名方的 $(R_{2,1}, \ldots, R_{2,\nu}), \ldots, (R_{n,1}, \ldots, R_{n,\nu})$，计算

$$
\begin{aligned}
R_j &= \prod_{i=1}^n R_{i,j} \quad (j \in \{1, \ldots, \nu\}) \\
(b_1, \ldots, b_\nu) &= \Big(1, , H_{non}(2, \tilde{X}, (R_1, \ldots, R_\nu), \ldots, H_{non}(\nu, \tilde{X}, (R_1, \ldots, R_\nu)\Big) \\
R &= \prod_{j=1}^\nu R_j^{b_j} \Rightarrow c = H_{sig}(\tilde{X}, R, m) \Rightarrow s_1 = c a_1 x_1 + \sum_{j=1}^\nu r_{1,j} b_j \bmod p
\end{aligned}
$$

把 s_1 发送给其他所有签名方。

聚合签名碎片 收齐其他方的 $s_2, \ldots, s_n$ 之后，计算 $s = \sum_{i=1}^n s_i \bmod p$，输出最终签名为 $\sigma = (R, s)$。

2.2.2 验签

给定公钥集合 $L = \{X_1, \ldots, X_n\}$，消息 m 和签名 $\sigma = (R, s)$，验证方计算

$$
a_i = H_{agg}(L, X_i) \quad (i \in \{1, \ldots, n\}) \Rightarrow \tilde{X} = \prod_{i=1}^n X_i^{a_i} \Rightarrow c = H_{sig}(\tilde{X}, R, m)
$$

如果 $g^s = R \prod_{i=1}^n X_i^{a_i c} = R\tilde{X}$，则签名合法。

2.3 与 MuSig 相比

[MPSW18b] 的签名需要三轮，而 MuSig2 只需要两轮。

参考文献

[BLL+20] Fabrice Benhamouda, Tancrède Lepoint, Julian Loss, Michele Orrù, and Mariana Raykova. On the (in)security of ros. Cryptology ePrint Archive, Report 2020/945, 2020. https://eprint.iacr.org/2020/945.

[DEF+19] Manu Drijvers, Kasra Edalatnejad, Bryan Ford, Eike Kiltz, and Igors Stepanovs. On the security of two-round multi-signatures. In *2019 IEEE Symposium on Security and Privacy (SP)*, 2019.

[jon20] jonasnick. Musig2: Simple two-round schnorr multisignatures, 2020.

[MPSW18a] Gregory Maxwell, Andrew Poelstra, Yannick Seurin, and Pieter Wuille. Simple schnorr multi-signatures with applications to bitcoin. Cryptology ePrint Archive, Report 2018/068, 2018. https://eprint.iacr.org/2018/068/20180118:124757.

[MPSW18b] Gregory Maxwell, Andrew Poelstra, Yannick Seurin, and Pieter Wuille. Simple schnorr multi-signatures with applications to bitcoin. Cryptology ePrint Archive, Report 2018/068, 2018. https://eprint.iacr.org/2018/068.

[NRS20] Jonas Nick, Tim Ruffing, and Yannick Seurin. Musig2: Simple two-round schnorr multi-signatures. Cryptology ePrint Archive, Report 2020/1261, 2020. https://eprint.iacr.org/2020/1261.

[Wag02] David Wagner. A generalized birthday problem. In *Annual International Cryptology Conference*, pages 288–304. Springer, 2002.