ProvSec'18 - Becerra - Forward Secrecy of SPAKE2

对Abdalla-Pointcheval CT-RSA'05 的SPAKE2 协议，本文证明了它满足弱的完美前向安全性；并进一步集成了key-confirmation 的过程实现了完美前向安全性，并且在Find-then-Guess 的模型中证明了安全性.

优点：①改进SPAKE2，增加key-confirmation使其满足PFS，符合TLS1.3对PFS的要求

②在C-S 场景下从效率、安全性的角度对比了EKE、SPEKE、PPK、PAK、J-PAKE、SPAKE2、PFS-SPAKE2

③在安全定义部分针对PFS和wPFS的freshness进行了说明

问题：①在性能比较部分，未考虑群检测(group test)引入的exp运算.

②CRS M 的引入可能带来密码外开销，如TTP和open source，但文中未分析

③证明采用 DIMACS'02 的 PAK 中的组件，但描述不够清晰，可以增强描述.

④文章假设了x的均匀分布.

PAKE在TLS当中的应用：TLS包含两个阶段 < Handshake：协商会话密钥 / Record：进行会话密钥保护下的通信

PAKE 适用于双方身份认证且需要保护 shared password 的场景(SRP, SPAKE2).

▲ 构建可提供PFS的PAKE协议符合最新TLS协议的要求

PFS与Freshness：PFS保证了长期密钥丢失后会话密钥的安全

▲ Krawczyk 在 CRYPTO'05 中指出使用公钥作为认证机制的 two-flow 协议无法满足PFS，故提出了 wPFS.(在添加显式认证后可以实现PFS)

→ Freshness：保证敌手不会轻易获胜

PFS-Freshness：保护所有corruption之前协商的session key

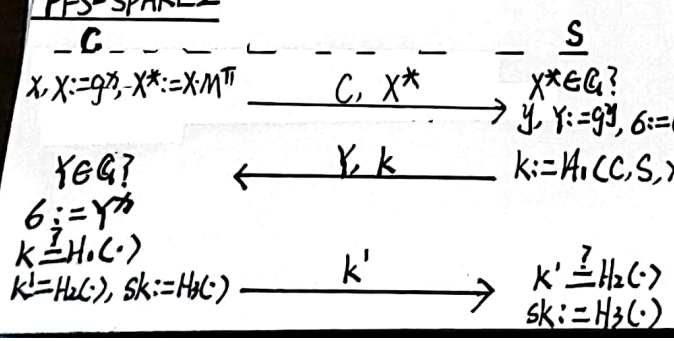
1> 没有Reveal query 指向 Π_U^i 或其partner.

2> 没有敌手参与的Corruption 发生在Test之前(即Test前的Corrupt会石破坏fresh)

wPS-Freshness：保护 corruption之前协商的没有敌手参与的session key.

与上不同的是第2点，只要有敌手的Corrupt参与即破坏fresh.

PFS-SPAKE2



Public: $\mathbb{G}, g, M\in\mathbb{G}$

$H_1, H_2, H_3: \{0,1\}^* \to \{0,1\}^k$

M的离散对数应保密.

安全性证明(Find-to-Guess, CDH, ROM)

G_0：原始协议执行. Execution of original protocol.

G_1：challenger simulate 所有honest instance Uniqueness of honest sessions

排除了所有消息在碰撞的可能(一旦生成了相同的参数，$\mathcal{A}$可以轻易获胜)

由birthday paradox：$Pr[Succ_1]\le Pr[Succ_0]+O(\frac{(n_{se}+n_{ex})(n_{se}+n_{ex}+n_{ro})}{q})$.

G_2：排除$\mathcal{A}$猜测到Hash output的可能 Prevent Lucky Guesses on Hash outputs

k'会决定协议terminate还是accept，更改simulate方式，

若双方pair，$sk\leftarrow\{0,1\}^k$，若$\mathcal{A}$调用了H_2, H_3，则使用生成的值作为sk.

若$\mathcal{A}$未调用H_3，则直接猜测到k'(F_1)：$Pr[F_1]\le\frac{n_{se}}{2^k}$.

未pair. $Pr[Succ_1]\le Pr[Succ_2]+\frac{n_{se}}{2^k}$

G_3：对于Execute的实例不维护$H(\cdot)$的一致性视图. Do not backpatch $H(\cdot)$

只有敌手query了$H(\cdot)$时才可以区分G_2, G_3. queries against Execute queries

通过证明表明敌手仅$H(\cdot)$来解决CDH难题.

$Pr[Succ_2]\le Pr[Succ_3]+n_{ro}\cdot Adv^{CDH}_{\mathbb{G}}(\mathcal{B}^{\mathcal{A}})$.

G_4：$\mathcal{A}$可伪装成C或S来进行在线字典攻击，猜测口令. Check for successful guesses

当敌手未猜到口令或Corrupt未发生时，不会有unpaired的实例terminate

$Pr[Succ_3]\le Pr[Succ_4]$

G_5：simulate了Send指向的instance，当敌手成功得到k并成功计算paired instance的k k'和sk时，游戏终止. Randomized sessions' keys for paired instance.

通过证明出现此类事件的可能性可归约成CDH问题得到

$Pr[Succ_4]\le Pr[Succ_5]+n_{se}\cdot n_{ro}\cdot Adv^{CDH}_{\mathbb{G}}(\mathcal{B}^{\mathcal{A}})$.

G_6：防止$\mathcal{A}$一次性测试两个及以上口令. Prevent testing more two pws per instance

$Pr[Succ_5]\le Pr[Succ_6]+n_{ro}^2\cdot Adv^{CDH}_{\mathbb{G}}(\mathcal{B}^{\mathcal{A}})$.

G_7：使用私有oracle Internal password oracle

$Pr[Succ_6]=Pr[Succ_7]\le\frac{1}{2}+\frac{n_{se}}{2|D|}$，即$\mathcal{A}$猜对口令时才可区分

SPAKE2