AsiaCCS'16 - Jarecki - DE-PAKE.

Device-Enhanced Password Protocols with Optimal Online-Offline Protection

introduce a setting called DE-PAKE, 通过 auxiliary device 来增强 PAKE 协议抵抗在线和离线猜测攻击的能力, 引入3模块化的设计 PTR+PAKE, U、D、S三者之一被攻击, 都不会导致 pw 泄漏 (PTR, password to random)

优点: ①模块化设计, PTR+PAKE, 可以更改任一方的实现方式
②方案无需服务器的更改, 服务器也不会察觉用户端使用了DE-PAKE
③无论是否使用PKI, 都可实现较强的安全性 (抗在线、离线字典攻击).

问题: ①引入了用户端的更改, 非用户友好型方案.
②没有方案的具体性能评估, 以及与其它方案的对比.
③在 Theorem 3 的证明中, Case 1 中式子(2)的名义应为优势而非概率.

敌手的能力: 在U-S和U-D可发动主动攻击的敌手; 可破坏S和D

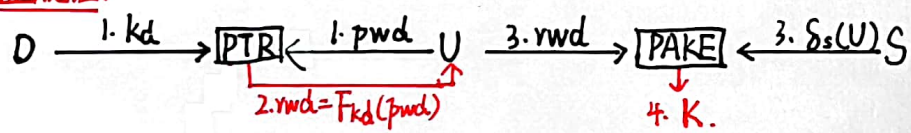
与2FA的联系: 本文基于 NDSS'14 - Shirvanian 的想法, 对其方案进行了改进.

1> C需要有S的公钥, 减少对公钥的需要对认证是需要的 2>6需更改都遭受阻

DE-PAKE的设计与分析. (基于 WET-ICE'00 - Ford&Kaliski 的 password hardening).

注册的 rwd 变体

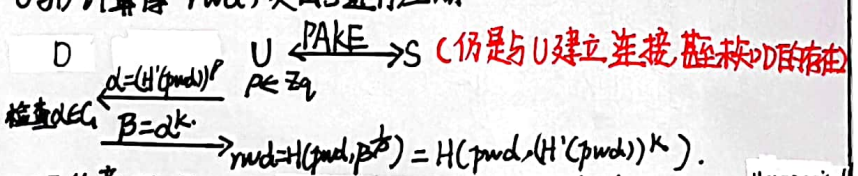
认证流程:



PTR实现过程: (FK-PTR-PAKE) 在循环群G上工作, 素数阶q, 生成元g. H→$\{0,1\}^\tau$, H'→G, $k \leftarrow Z_q$, OPRT: $F_k(x) = H(x, (H'(x))^k)$.

初始化阶段, U与D计算得 rwd, 交由S进行注册

登录阶段:



敌手可做的query (对于任意的 Party (P, 包含 U, D, S).

output $\delta_S(U)$ at S if uncompromised

U-S: send(P, i, P', M), Reveal(P, i) → sk, corrupt(P), compromise(S, U), test(P, i)
用于衡量在线猜测攻击的为敌手生成的消息, → Rogue send queries/activations
当server-compromise, 敌手应做离线猜测攻击来伪装成U (wkc1).

U-D: Reveal 不可作用于D, Corrupt 可得到D中的secret

重新定义fresh: 当没有 corrupt、compromise, reveal query 到已 accept 的匹配会话该会话实例是fresh; 另, 当实例被 corrupt(S) 或 compromise(S, U) 时, 敌手无法获取破坏S前的临时 secret, 即阻止A伪装成U. (Strong KCI resistance)

FK-PTR方案及安全性分析 (Attacker A has oracle access to U and D).

通过4种攻击方式来定义安全性, 其中 RDict = $\{F_k(p): p \in Dict\}$, 其中 k是D secret key

1> Attack avenue 1: Leakage on rwd = F_k(pwd). 确保pwd与rwd在敌手角度是独立的, 通过 distinguishing test 来区分: rwd = F_k(pwd), $r \leftarrow RDict \setminus \{rwd\}$ 给A rwd及r (in random order). A需guess哪一个值等于 F_k(pwd).

2> Learning values in RDict: 敌手可与D交互得到 rwd 并与S交互测试, 使每次与 D交互, 至多可以排除1个值, 即游戏结束时, 输出 RDict 依然猜不起 q_D.

3> Using U to test passwords: 敌手可以通过在U-D中交互使U输出 F_k(pwd'), 而pwd'可为A已知的, 进而A可以不进行 U-S 的中间人攻击 即可测试 pwd' 是否正确.
本方案不允许敌手: use U as an oracle for testing passwords in RDict \ {rwd} (negligible.)

4> Running U on passwords outside RDict: ①A使U在一个不同的RDict中运行, A可伪装成D, 并使用k'构成 RDict*, ②A伪装成S, $F_{k'}$(p) 为注册的信息, 若 p=pwd, 则A成功, 但该过程需 2^d 次对S的伪装才可成功, 但是若 $F_{k'}$ 为一个 t-to-1 的函数, 则可以一次性排除t个口令. ⇒ 故应使 F_k 为 1-to-1 函数.

在 OMG-DH 假设下证明 FK-PTR 方案的安全性

实例化与扩展

1> 使用 PKI-free PAKE + 协议构成的 FK-PTR-PAKE
ASIACRYPT'14 Jarecki + EuroSP'15 Jarecki

2> Server Transparent DE-PAKE., 由于 U-D 之间执行 PTR 生成 rwd, 无论是否使用 TLS, 都无需 server 做更多更改, 它只与U交互
好处 ①用户只需记住低熵 pwd, 但存储在S中的为高熵.
②pwd 可以复用, 但 OPRF key 对于每个service应不同.
③pwd 可以减少更改频率, 而是变更 D 的 key.

3> 抵抗 Client-Side & Phishing Attacks: 即使拿到了 pwd, 需要攻破C才能拿到 rwd; 而且 rwd 若 service 唯一, 或通过 2FA 的 One Time PIN, 来抵抗 client compromise. 通过将 rwd 与 URL 绑定计算, 抵抗钓鱼攻击.