Trustcom'18 - Zeidler - AuthStore: Password-based Authentication and Encrypted Data Storage in Untrusted Environment

概述：口令可用于认证，也可用于加密服务器存储的数据。但若两者使用相同的口令，可能导致恶意的服务器拿到口令解密存储的数据。而使用不同口令或使用访问授权方案会引入新的问题。(T)本文旨在解决上述问题，给出在使用1个口令时仍安全的认证方案。(A)首先提出一个认证框架AuthStore，使用key stretching技术保护口令，在一个或多个dP的情景中完成设计目标；指出许多基于key stretching的协议易遭受parameter attack，提出一个compact PAKE协议可抵抗此攻击，并基于该框架给出了一个口令管理器的实现。(R)对提出的协议与其它协议进行对比，表明其只需更少的消息数时可提供足够的安全性。

优点：① 提出了一种认证框架AuthStore，可使用同一口令进行认证和数据加密。
② 基于AuthStore给出了一种口令管理器的实现方式。
③ 指出许多key stretching协议易遭受parameter attack.

问题：① 协议的安全性分析有些简单；无形式化分析；提出的协议声称可以抵抗parameter attack，但没有具体分析。
② 本文中未涉及ksym的更新，且一旦ksym更新，加密数据应如何更新？应简要讨论。
③ 口令管理器部分应给出更详细的实现思路，另外应提供可用性和安全性分析。

1. 系统需求

1> Simplicity：用户只需用户名和口令即可完成认证。

2> Secure Password Storage：用户可以配置口令的存储方式，以抵抗离线字典攻击

3> Secure Authentication：认证协议不会泄漏口令的信息，否则破坏了服务提供商的攻击者不能伪装成用户使用存储的信息。

4> Data Storage：用户可以使用认证口令加密存储的数据，服务提供商无法访问数据

2. 系统模型.

1> Threat Model:
① 用户通过可信的客户端软件与服务提供商通信，服务提供商可以为认证用户提供数据存储服务
② 服务提供商为恶意的，希望获取用户口令来解密数据或伪装成用户用于认证
③ 假设用户选择合理强度的口令和较强的key stretching参数，抵抗离线字典攻击.
④ 外部敌手可以执行在线猜测攻击，中间人攻击，也可以作为服务提供商参与协议。

2> 本文方案：使用KDF生成1个base key，用户可以使用base key生成一系列user key，其中使用的参数存储在服务提供商中。

▲本文采用Asymmetric PAKE，需满足：① 服务提供商和外部攻击者无法获知口令的额外信息 ② 中间人敌手无法通过截取的信息进行离线字典攻击 ③ 即使认证数据丢失，也不能伪装成用户.

3. 方案细节

1> 生成base key：输入口令pw，盐值和成本参数P_K，生成base key：$K = KDF(P_K, pw)$

2> 生成user key，输入base key和随机salt，user key参数为$P_u = [P_K, salt]$

$k_u = H(k, salt) = H(KDF(P_K, pw), salt) = U(P_u, pw)$.

3> 注册和认证（kauth记为π）

用户在注册时使用密钥参数P_π生成π，和验证值V_π，P_π和V_π提交给服务提供商

在认证时用户请求P_π并生成π.

4> 安全数据存储（用户生成参数$P_u, data$，生成k_{data}）.

采用key chain方法，其中口令更改不需要重新加密数据.

生成一个随机的加密密钥k_{sym}，使用k_{data}加密.

▲数据参数：$P_{data} = [enc(k_{sym}), P_u, data]$，存储在服务提供商中.

5> 口令更改和密钥更新.

更改口令时，同时更改参数并生成新的验证值（也可只更新参数）

存储的数据仍由k_{sym}加密，而新生成的P_{data}可用于加密k_{sym}.

6> 帐户重置.

服务提供商通过可信通道传递认证数据给用户，用户可通过认证并重置自己的口令.

7> 安全分析

① 当P_π与V_π泄漏或敌手能破坏服务提供商时，只能执行离线字典攻击.

② 加密数据应由高熵口令加密，使用足够强的参数，否则随着算力提高，帐户易受攻击

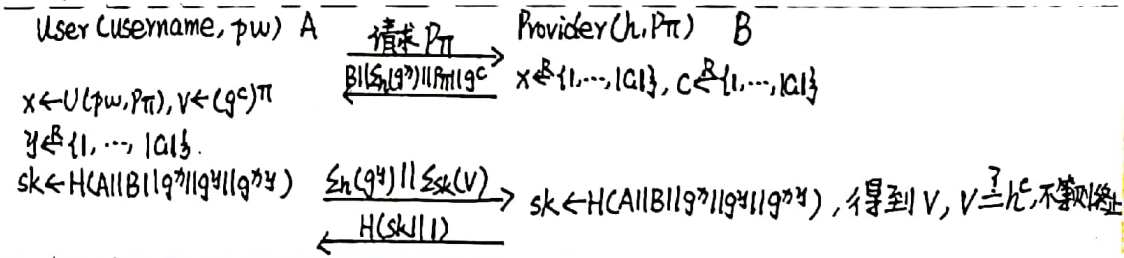
③ parameter attack：服务提供商在用户请求时给定弱的参数P_K'，因此用户使用弱的认证密钥π'认证，若敌手可以在认证协议中获取π'的信息，则可以进行字典攻击.

此类攻击可以通过设计低熵认证密钥协议来避免.

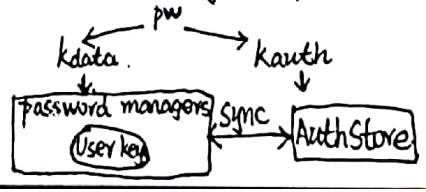
4. Compact PAKE协议（4 messages，基于EKE2和用于B-Speke协议的认证方案

1> 注册：用户从认证参数P_π中生成认证key π，选取生成元g，计算$h = g^\pi$，作为验证值V_π.

2> 认证



5. Password Manager（可在AuthStore供应商处存储任意凭证）



① 需要可信的客户端实现

② 避免了昂贵的key derivation

③ 无需用户选择认证口令即可完成注册

④ 不使用口令管理器仍可以完成认证