CACM'15 - Bonneau - Passwords and the Evolution of Imperfect Authentication

概述：(S)吐槽理论落后于现实，random user 和 offline attack 两个过时的model使得许多建议不符合实践。(T)本文旨在回顾认证本身并深入理解吐在其中的重要作用；(A)首先回顾吐的历史发展，强调两个过时model带来的理论落后于实践的后果；提到吐不可被替代的原因，并指出许多建议不实际且互不兼容。(R)提出渐近式部署，随安全性需求提高逐渐部署新的认证方案。

优点：①指出3个过时的模型，指出根据此模型提出的建议不符合用户的实际需求

②介绍渐近式认证方式，以及基于机器学习的classifier的认证方式 ③指出用户行为改变代价最高

问题：①文章在考虑猜测攻击时，只考虑了漫步攻击，例如 block 常见的吐对定向攻击作用较小，在定向猜测攻击中，在线猜测所需吐强度不一定弱于离线猜测。可引入定向猜测攻击进一步讨论

②论述离线攻击只是众多攻击的一种时，最好给出实际攻击事件的比例和攻击难度

③值得进一步研究的：更精确地评估吐强度的方法；更好地nudge用户采用强吐。

▲吐的历史（吐的目标不是达到坚不可摧的安全，而是以可接受的代价减少危害）

■ 1）之前对吐的研究通常集中于特定的问题，容易形式化，但不会对现实世界的设计目标产生影响，即保护用户帐户和隐私数据，而不是认证本身。

→例：学术界建议严格的password-composition policies，但没有证据表明它可减少危害

2）吐的相关实践很发生变化，表现学术界没有提供比当前实践更让人信服的方案。

3）标识出两种过时的模型（但仍广泛使用）

① random user：从某个集合中均匀且独立的选取吐(uniformly and independently)。

② offline attack：与其它威胁(如钓鱼攻击)相比，系统猜测的关注过多，其它的攻击更加难以防御

4）现有研究未认识到现实认证的复杂性，未认识到 security, usability, deployability 的相互依存关系，这导致了许多不兼容的吐要求，没有一种方案可解决所有问题，需协同合作

5）工业界的一些方案提出互补认证机制(多通过机器学习)，但此类方案通常适用于对用户习惯有了解的大型企业，此类技术可能侵犯用户隐私且难以被用户和开发者理解

▲吐角色的变更

1）最初为安全辅助手段。1960s 用于分时操作系统，防止恶作剧和未授权的访问，MIT 1961年部署的兼容分时操作系统可能是首个应用吐管理授权的系统。

2）1970s MULTICS和Unix的访问控制中用于保护隐私数据和计算资源，MULTICS Host存储1974年Morris-Thompson的首个针对吐安全的研究描述了hash+salt的方法，并首次分析了dictionary attack和brute-force guessing.

3）1988年，Morris Internet worm 提出许多系统普遍吐猜测攻击漏洞。系统管理员可采用shadow吐文件的方式存储吐，并可以主动检查用户吐是否容易被猜测

4）mid-1990s，SSL和SET协议未能替代文本吐，后者成为最主要的身份认证方式

5）web服务的兴起，导致产生了许多可用性问题。

①采用由附件自动重置吐使其变成 central-point-of-failure.

②增加的帐户数且使得一个帐户一个强吐难以实现，吐重用变得常见

③钓鱼攻击成为广泛的威胁，其中包含黑名单和机器学习类的方法逐渐被提出

6）将石硬件作为第一个因素的方案也没有得到广泛应用(因为价格昂贵)；SSO 也未得到广泛应用

智能手机的广泛应用(2010s)可能被用来作为第二个因素(ToTP除SMS)

在对安全性要求较高的场景中，服务可能提供专门的 token 用于认证。

■ Random models for user behavior

① 用户不仅是吐系统中最弱的一环，而且是最难建模的组件。(用户真实行为与理想行为差距很大)

② 1985 Green Book 提出了一系列缓解吐猜测风险的方案：(i) rate-limiting (ii) hashing passwords at rest (iii) limiting the lifetime of passwords

认为用户创建的吐是易猜测的，因此所有的吐均应是机器生成的

③ 同年 FIPS 提出除了推荐机器生成吐，允许用户自行选择吐，但该吐应为随机的，且与个人实体，历史或环境无关

认为用户也很难吐与随机吐相似

仅管几乎所有非军队网站均使用用户创建的吐，但 FIPS 仍保持原有的建议，导致不切实际的假设

1）通过entropy估计吐强度。

①香农熵可以建模：攻击者猜测任一大小的吐集的猜测次数。(H_1)

②另一个猜测指标 guesswork(G)，致力按顺序猜测各个吐直到找到所有吐的预期猜测数目 H_1 代表G的下界。但G会在罕见、难以猜测的吐集中出现高度偏态。

③为了减小这个bias，采用了partial (or marginal) 的评估方法，partial guesswork (G_α) 代表攻击者猜测到 α 成功率需要的猜测次数。(α 值较小时反而可标识真实的攻击者)。

↳不足之处是，需较大的样本才能保证估计的数且准确(例百万条吐)，故应用较小。

④对于较小的数据集通过需要启发式的方法

N16T引入了在不同构造策略下得到的吐分布的熵的估计方法

更受欢迎的方法是，采用开源的吐破解工具评估破解一定比例的吐所需要的猜测数

但可能由于攻击者采用了其它工具造成高估或低估。

2）改善吐强度

①当吐为随机时，要求吐长度和字符数目看起来可使吐强度增加，但很可能高估吐安全性。

② 由于使用这样的吐策略会引入可用性开销，有竞争压力的网站通常不会采用。

③研究建议采用黑名单抵抗在线猜测攻击(即排除流行吐)。

④通过给用户反馈 nudge 用户采用安全的吐，但要求用户能意识到 nudge 的存在

3）选择吐时的独立性

①random user model假设用户每一次独立选择吐，但实际的研究表明，用户通常会在几个网站之间重用吐或简单的修改吐并继续使用 (人可预测的方式更改)

②在强制更新场景中，也存在类似的场景，攻击者若拿到之间的吐，则攻击成功率大大提高

■ Offline vs. online threats.

离线猜测攻击者只受自身计算资源的限制，而在线猜测攻击者则需与合法实体交互认证，此外在线攻击者可能有多种攻击方式，包括猜测，恶意软件，窃听等，但除猜测外，其它攻击与吐强度无关

1）Offline guessing (cracking)

此类攻击只是现实攻击场景的一小部分。

① 若攻击者无法访问口令文件，则只能进行在线猜测

② 若泄漏口令文件为明文存储，则无论口令强度如何均可直接获取。

③ 若文件仅哈希但未加盐，则可采用彩虹表攻击。

④ 若文件加盐哈希，攻击者获取口令文件后可执行离线猜测攻击，口令破解效率与口令强度有意义

→ 有研究表明 90% 网站采用加盐哈希存储，其余包括明文，哈希未加盐，不恰当的哈希，可逆加密

→ 若口令泄漏被检测到，管理员可使口令重置。但泄漏的网站通常不会直接重置口令，担心失去用户；甚至当重用口令的一个服务被攻击后，用户也不会强制被要求重置口令。

2) Online guessing

猜测的口令数是可以被限制的，但 three-strikes Law 在实践中不常见，可能是为了避免 DoS 攻击

在线猜测一次付出的代价比离线猜测大：① 为防止 IP 被 block，需更换 host ② HTTP POST 的开销比 hash 的开销更大 ③ load 可能超出合法流量。

→ 认为：选择抵抗离线攻击的口令比抵抗在线攻击的更困难。

■ 口令替代方案

1) 口令具有经济优势，也具有 deployability 优势

2) 口令替代问题是：

① under-specified：不足之处，没有一个涵盖各种环境、技术平台、文化和应用等方面要求的方案。

② over-constrained：过于严格，没有一个方案能够解决从经济到隐私保护的所有要求，usability, security 和 deployability 的要求过长。

口令类似于 Pareto equilibrium，想获取一个属性就必须放弃另一个属性，故难以被替代

例→ Password Managers 的问题是：难以在所有的用户代理中进行配置。

3) 更好的方式是根据组织机构的属性和使用场景确定需求，并逐步完成采用

4) 潜在替代方案的挑战仍有许多

① 若提升安全性和降低可用性可能导致用户流失 ② 替代方案需 S 或 C 的更改，阻碍采用

③ 需用户变更自己的使用习惯 ④ 只解决部分问题 ⑤ 开销较大，有部署性问题。

■ 给用户提供的建议

1) 专家给用户提供的口令建议组合起来给用户带来了巨大的负担，

"compliance budget" model，每个用户遵守个人对应的安全需求的意愿是有限的。

可用的安全建议需成熟的风险管理的态度，以及每种方案的 cost 和 risk。

2) ① 选择强口令。对危害的减少效果未知，但对于重要的帐户应建议使用他人难以猜测，可以承受合理次数的在线口令猜测，鼓励不使用简单字典口令，使用 PSM 及 blacklist 可提供帮助

② 避免口令重用。在重要的帐户之间避免重用口令，不需要特别担心低价值的帐户

③ 若用户可保证写下来的口令文档的安全性，则这种行为可以减少弱口令的使用，是一种 worthwhile trade-off, → Password Manager 是一种更好的 trade-off.

■ 理解为什么大的服务提供商使用坏的技术的关键是：网站不需要完美。帐户泄漏只是

垃圾邮件、钓鱼攻击中的一种，它们也未被技术完全击败，但依然可正常运行。

→ 几种种情况，从技术上解决问题的方法都不如采用统计学的方法。例如，对于垃圾邮件，基于已知的攻击者行为模式将邮件分类的技术比协议应用更普遍，且代价比需要用户变更行为小很多

▲ A multi-dimensional picture. 多维方法进行认证

1) Web authentication as classification.

① 很多网站转型成 risk-based model，不正确的口令无法访问，但正确的口令是 classifier 的 signal

② classifier 可以利用许多 signal，例如 IP 等信息，但这些信息均可能被伪造，不过伪造所有的 signal 是比较困难的

③ 此时的输出不是 binary，而是 real-valued estimated likelihood 来标识该尝试是否为真实用户

任何的 classifier 都可能出现 false accepts 和 false reject.

④ 另外，获取足够的样本进行训练也是一个挑战，特别是 APT 攻击

2) New modes of operation 更灵活的认证方式

① progressive authentication：在 classifier 的 confidence 较低或用户执行敏感的操作时，需要额外的 signal 作认证

② Multi-level authentication，当 classifier 的 confidence 较低时，用户只有受限的访问权，并可以根据认证的 signal 而确定用户权限（例如 session cookie）

③ Opportunistic 2FA：当 2F 存在时可完成认证；若口令正确且额外的 signal 足够，可以提供 fallback security.

④ Continual authentication：在口令认证后，需基于其它 signal 持续认证，最终认证方案与其他检测系统共同使用

3) Changes to the user experience (对用户行为的改变应为正向的)

① 当 classifier 的 confidence 较低时，用户可能遇到更多作的 quest，用户会有更多情况无法完成认证

② 认证系统变得更加复杂，用户可能难以理解而采用

③ 好的 classifier 可能会破坏口令用户原有的习惯，例如难以完成共享口令

④ 减少输入口令可能会降低可用性，用户可能会忘记

4) Advantages of scale.

① 大型服务更可能被依赖方接受作为认证服务，这反过来鼓励用户注册帐户。

② 具有更多用户数据的大型服务可以提供更准确的身份验证。

但这种不平衡的数据占用可能使得研究人员难以对其做探入研究

且收集信息将导致用户在隐私担忧。