超越凭证填充：使用神经网络的密码相似性模型

摘要：

使用从一个网站泄漏的密码来破坏其他网站上关联账户的情况越来越常见，攻击者之所以能够进行这种有针对性的攻击，是因为用户在不同的网站上重复使用相同或者相似的密码。我们将针对性猜测攻击中的核心技术挑战重新定义为密码相似性建模任务。我们展示了如何使用泄漏了的14亿邮箱账户和密码来学习密码相似性模型。

在使用我们训练以后的密码相似性模型的情况下，我们对此展开了目前为止最具破坏性的针对性攻击。模拟表明，即使使用了最新的对策，只要攻击者知道他们的其他密码之一，我们的攻击就会在不到一千个猜测中入侵超过16％的用户的账户。我们通过大型大学认证服务的案例研究表明，这种攻击在实践中也是有效的。我们将继续通过个性化密码强度计（PPSMs）提出针对这种定向攻击的首次防御。这种密​​码强度值可以在用户选择容易受到攻击的密码时向用户发出警告，包括利用用户先前被盗用病被针对的密码。我们设计并构建了一个PPSM，可以将其压缩到小于3 MB使其易于部署，以便针对所有已知的猜测攻击来准确估计密码的强度。

1. 介绍

尽管反复呼吁用完全不同的身份验证机制替换密码，但今天密码仍然广泛使用，并有很大可能性在将来继续使用，尽管它们易于猜测，难以记住和难以正确键入。因为密码难以记住和键入，所以人们才会在不同网站重复使用相似的密码：将近40％的用户重复使用相同或相似的密码。密码重用和不断增加的密码泄漏使有针对性的猜测攻击成为越来越严重的威胁。有针对性攻击的最普遍形式为凭据填充，其中攻击者只是尝试使用在泄漏中发现的与该用户相关联的密码来登录用户的账户。到2017年底，互联网上已经分发了超过50亿个泄漏的账户；机器人驱动的凭证填充攻击占全球一些最大网站的登录流量的90％；这些攻击是账户接管的最大来源。

网站运营商有时会借助第三方服务（例如HIBP）帮助重置用户密码（如果发现其用户名或密码遭到破坏）。NIST现在正积极推行这种保护措施，即用户可以选择一些较小的被破坏密码作为其密码，但它只能防止凭据填充。少数学术著作研究了凭证填充的一般性，根据处理规则或概率性上下文无关文法（PCFG）选择泄漏的密码变体。研究表明，即使在用户重置密码后，这种有针对性的攻击仍可能会造成破坏，因为用户仍具有倾向于选择相似密码的趋势。我们使用凭据调整来应对提交泄露密码变体的攻击。

在这项工作中，我们将从人们选择的密码之间所将具有的相似性角度研究凭证填充攻击。我们探索使用现代机器学习技术对相似性进行建模的数据驱动方法。这样就产生了一种新的有针对性的密码猜测攻击，其性能优于以前的所有攻击，以及一种新型的密码强度计的设计，该强度值在强度估算中包括了针对目标攻击的脆弱性。简而言之，我们通过学习估计的模型来处理相似性，其中是一个站点泄漏的密码，表示用户在另一网站选择的口令。然后，我们将估计这组条件概率分布（每个对应一个）作为学习任务，在这里我们使用包含14亿泄漏邮箱账户和密码的汇编。我们探索了各种启发式方法来识别数据集中单个用户使用的密码，最后将获取大量有关密码相似性的数据。我们首先使用该数据集来学习一种紧凑的生成模型，该模型使用序列到序列（seq2seq）算法捕获所有的。这些在自然语言处理文献中广泛用于语言翻译和其他任务。在这里，我们将输入的“源”密码视为，并且模型学习如何生成新密码，以反映数据中看到的相似模式。但是，以这种方式使用seq2seq会导致结果不优于以前的攻击。因此我们采用了不同的方法，训练模型将其转换为以预测所需的的修改。虽然看似等效，但事实证明这种方法更为有效。直观地，它使模型更好地集中于学习数据中发现的常见转换。我们将生成的算法称为密码路径（pass2path），该路径表示转换的顺序。

使用pass2path模型，我们建立了一个凭证调整攻击，通过模拟可以证明，如果他们另一个账户的密码出现了漏洞，该攻击可以在不到一千个猜测中破坏超过48％的用户账户。用于凭证调整攻击的基准算法可以猜测泄漏的密码，大约有40％是由于密码重用造成的。因此需要关注的是在凭据填充对策的情况下，我们的攻击效果如何。我们针对该情况执行（分离）模拟，该模拟表明我们的攻击可能会破坏16％的用户账户。这是以前最佳针对性攻击的1.2倍，是最佳非针对性攻击的3倍。

模拟可能无法准确地代表真实的效果，因此，我们通过与康奈尔大学（Cornell University）IT安全办公室（ITSO）合作，评估了对真实系统的凭据调整攻击。ITSO部署了凭证填充对策以及其他最新的防御措施。但是，基于pass2path的凭据调整攻击成功地在1000次尝试中猜中了泄露的15665个活跃的Cornell用户账户中超过8.4％的密码。我们在这里进行的实验不仅证实了在实践中进行凭据调整攻击的威胁性，而且还帮助我们领先于攻击者一步，并确定了成千上万个存在危险的Cornell账户并进行特殊监控。然而，强迫这些用户选择新密码并不一定会阻止攻击，因为他们最终可能会选择以前的密码的变体。

因此，我们引入了个性化密码强度计（PPSM）。这些参数会考虑用户的其他（泄露的）密码来估算密码的强度（不可猜测性）。我们使用基于神经网络的词嵌入技术创建了一个称为vec-ppsm的PPSM，它代表了另一种建模密码相似性的方法，该方法比pass2path更适合用作强度表。我们的PPSM可以在面对有针对性的猜测攻击时识别不安全的密码，并且可以与现有的密码强度表结合使用，以针对所有已知攻击提供准确的密码强度估计。同时在主体中我们讨论了vec-ppsm的各种部署设置。

总而言之，我们的贡献包括以下内容：

•我们将针对性猜测攻击中的核心技术挑战重新定义为密码相似性建模任务。这种观点使我们能够适应最先进的机器学习工具，并将其应用在公开可用的数十亿泄漏凭证。我们设计了一个模型pass2path，它可以准确地生成给定泄露密码的用户可能选择的转换。

•使用pass2path，我们构建了迄今为止最有效的针对性密码猜测攻击。 仅在1000次猜测中，它就可以保护16％的用户账户免受凭据填充的侵害。

•我们在实践中首次衡量了有针对性的攻击，结果显示，尽管康奈尔大学(Cornell University)采取了应对凭证填充的对策，但仍有1316个正在使用的账户可能被我们的证书篡改攻击攻破。

•我们介绍了个性化密码强度计（PPSM）的概念。 我们使用词嵌入技术来构建PPSM，并展示如何将其用于帮助防止凭证篡改攻击。

1. 背景

**密码模型。**人类选择的密码之前已经使用自然语言处理(NLP)工具进行了分析。早期的例子包括使用马尔可夫模型来帮助改进基于字典的破解工具。随后提出了许多数据驱动的方法来使用密码泄漏来学习密码的语言模型。此外，还提出了许多利用密码泄漏来学习密码语言模型的数据驱动方法，其中Weir等使用了概率上下文无关语法(PCFGs)，后来被Komanduri等改进，在“通过有限的样本对对手进行建模以评估密码强度”中估计人为选择的密码的分布情况。 Ma等使用一些精心选择的参数对基于Markov模型的技术进行了改进，表明在用于生成大量密码时优于基于PCFG的模型。在2016年，Melicher等使用递归神经网络（RNNs）和Hitaj等提出使用深度生成对抗网络（GAN）对密码进行建模。

**密码猜测攻击。**密码模型的主要应用是训练暴力猜测攻击。此类攻击分为两大类：离线和在线。当攻击者获得某些用户密码的加密哈希值并尝试通过猜测和检查数十亿（甚至数万亿）的密码来尝试恢复用户密码时，就会发生脱机攻击。攻击者面临的主要挑战是生成密码猜测值，...的有序列表，其真实用户密码很可能会提前出现。该列表中密码的索引称为密码的猜测等级(β)。

当攻击者使用登录界面或其他API针对某个账户提交猜测的密码时，就会发生在线攻击。由于现代身份验证系统会在几次失败尝试后锁定账户（例如10次），因此就攻击者可能做出的猜测而言，在线攻击比离线攻击的限制更大。但是，主要挑战是相同的。给定若干猜测或查询的q，攻击的成功概率就是我们所说的q成功率，记为λq。对于本研究，我们将重点放在在线设置上，将查询预算限制为1,000或更少。

大多数密码猜测文献都集中于非目标攻击，这些攻击以与被攻击账户无关的方式生成密码猜测序列。相反，针对性攻击则尝试利用有关被攻击账户的额外信息，凭据填充攻击将一个账户的泄露密码提交给另一个网站上的关联账户。这些问题引起了越来越多的关注，这在很大程度上是由于大量的密码泄漏：用户账户很可能与从另一个来源泄漏的至少一个账户相关联。

Das等利用此类辅助信息进行了针对性攻击的第一项学术工作。他们表明，大约43％的用户在不同的网站上重复使用相同的密码。他们还手动开发了一种基于规则的算法，以通过用户其他密码之一的信息来猜测用户的密码。我们将这种凭证填充的一般化称为凭证调整，因为攻击者还会对泄露的密码进行修改。后来，Wang等构建了个性化的PCFG模型，以基于个人信息（包括泄露的密码）来指导凭证调整。对于与在线猜测有关的少量查询预算，这些有针对性的攻击要胜过无目标的攻击。但是，这些现有技术不适合更好地利用现在可用的大量泄漏数据。我们将转向更现代的机器学习技术。

**密码强度计。**密码模型还用于开发强度值，它们最常被用作“助推”以帮助引导用户选择更强的密码。最初使用各种统计方法（例如Shannon熵）进行密码强度估计。这种方法有很多缺陷，请参见[28]，[29]。最近，可以通过在某种密码模型下计算密码的猜测等级来估算密码强度。给定密码模型，可以使用Dell和Filippone引入的蒙特卡洛技术有效地估计猜测等级。

1. 预备知识

用户为不同的账户选择相似和相关的密码，因此可以利用用户的一个密码的信息来更有效地猜测他们的其他密码。虽然可能有许多潜在因素会影响用户对密码的选择，例如其人口统计信息，网站内容的敏感性以及网站的密码策略，但先前的研究表明，用户的先前密码是用户选择密码的最主要因素。因此要了解密码之间的相似性，我们将只关注用户的密码，而不考虑选择密码的用户和选择密码的网站。如果用户经常选择两个密码，我们认为这两个密码是“相似的”。

在形式上，令Σ表示密码中允许的字符集（例如，所有ASCII字符），表示密码的最大允许长度（例如50）。令p表示用户为账户选择密码w∈∑∗的可能性。我们用W表示该分布的支持度。我们将两个密码和之间的相似性建模为条件概率，即假设用户选择密码∈W的情况下，则用户选择密码∈W。我们可以扩展相似性的定义，以考虑用户过去的多个密码，，...，并计算用户选择的概率。在这种情况下，我们可以将密码的条件概率分布建模为

先前的研究隐式地尝试使用人工策划的规则或使用概率上下文无关文法（PCFG）来理解人为选择的密码的相似性。近年来，神经网络已被证明对于许多自然语言任务非常有效，例如理解单词的相似性或将自然语言文本从一种语言翻译为另一种语言。我们采用基于神经网络的NLP工具来对密码相似性进行建模。使用这些工具，我们可以建立更有效的攻击和针对目标攻击的有效防御。

**密码相似模型的应用。**如果攻击者可以从其他网站访问用户的密码，则可以使用良好的密码相似度模型对用户进行有针对性的攻击。这种模型对于创建针对最新目标攻击的防御措施也很有用。客户端应用程序可以通过查看与用户其他各种密码之间的相似性来警告/阻止用户选择密码，从而在面对有针对性的攻击时减少危险。密码相似性的另一种应用可以是纠正密码输入错误，因为输入错误通常包含相似的密码。

尽管密码相似性的所有这些应用都需要学习条件概率分布，但是它们需要的是与训练模型不同的接口。例如，要构建有针对性的攻击，必须能够有效地枚举条件概率分布以生成猜测。但是，在使用密码强度计的情况下，我们不需要有效的枚举功能。因此，我们针对两种不同的密码相似性模型。

第一个模型是生成模型，使用之前提出的用于语言翻译的序列到序列样式模型（seq2seq）构建。给定密码，这个模型可以用于按条件概率降序的列举类似的密码。

我们训练的第二个模型基于词嵌入技术，通常被用来理解词之间的相似性。此模型对于获得一对密码之间的相似性评分（代表条件概率）很有用，但是对于仅给定一个输入密码的情况无法枚举相似密码。尽管生成模型也可以用于获得相似性评分，但嵌入模型足以构建强度计。正如我们在第VII节中所示，嵌入模型比生成模型更容易训练，并且更有效地计算密码之间的相似性评分。

**密码泄露数据集。** 我们用于学习密码相似性的数据集是随时间推移泄露的各种密码集合。 该数据集最早是由4iQ在Dark Web中发现的。数据集包括14亿个邮箱账户密码对，11亿个唯一电子邮件和4.63亿个唯一密码。 数据集的管理者（未知）删除了重复的电子邮箱账户密码对。

尽管我们不知道用于编译此数据集的确切泄漏情况，但该文件夹包含名为“ imported.log”的文件，该文件指示所有泄漏主要是在2017年12月5日之前。 列出的泄漏包括Linkedin，Myspace，Badoo，Yahoo，Twitter，Zoosk，Neopet等。尽管没有官方方法可以保证泄漏的真实性，但一些研究人员已验证了一部分密码是合法的。 （令人担忧的是，泄露中出现了两位作者的密码）

**数据集清洗**。数据集中的几个密码是完整的哈希值。为了清理数据集，我们删除了任何包含20个或更多字符的子字符串(只包含十六进制字符)。这种方法删除了150万个密码。我们还删除了包含非ASCII字符的密码以及长度超过30个字符或少于4个字符的密码。总体而言，我们删除了260万个密码（占0.6％），将有效密码的数量减少到4.604亿。我们还发现4,528个用户与数千个密码相关联。这些不太可能是真实用户的密码，因此我们删除了这些账户。

|  |  |  |
| --- | --- | --- |
| 属性 | 值 | 百分比 |
| 长度 | 3-5  6-8  9-12  13-50 | 2  48  40  10 |
| 构成 | 仅含小写  仅含大写  仅含字符  仅含数字  仅含特殊字符  仅含字母和数字  至少包含一个字母、一个数字和一个特殊字符 | 80  3  38  8  <0.1  55  5 |

图1：清洗后数据中密码长度和组成的分布。

干净数据集中最受欢迎的密码（123456）被所有用户的0.9％使用。因此，密码分配的最小熵为6.68bit。 q-成功率定义为攻击者对每个账户可以进行q次猜测的预期成功概率，它的上限是q个最可能密码的概率之和。对于我们的数据集 = 0.11。这些值与先前工作报告的密码分配一致。图1显示了清理后的数据集中有关密码的组成和长度的统计信息。超过88％的密码长度在6和12之间，而80％的密码仅包含小写字母。

**加入账户。**泄漏数据集包含电子邮件密码对形式的账户凭据，并删除了重复的对。我们要合并账户以查找属于单个用户的账户集,这将为我们提供与用户对应的密码列表。我们探索了三种启发式方法来合并账户，如下所述

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 用户数量（百万）  密码数量（百万） |  | 146  183 | 195  210 | 174  190 |
| 每位用户的密码 | 2  3  ≥4 | 77.0  15.5  7.5 | 57.1  19.1  23.8 | 74.9  16.3  8.8 |
| 密码重用率 |  | 0.0 | 30.3 | 39.7 |
| 编辑距离 | 1  2  3  ≥4 | 9.4  5.2  3.3  82.1 | 6.8  3.9  2.4  86.9 | 9.1  5.9  3.2  81.8 |

图2：三种账户加入技术下的数据集比较

注：电子邮件地址（），用户名（）和电子邮件和用户名的组合（）。 我们仅考虑具有至少两个泄露密码的用户。 最后一组行给出了在指定的编辑距离内来自同一用户的不同密码的分数。 除前两行外，所有值均为百分比（％）。

•基于电子邮件（）。在第一种（也是最明显的）策略中，我们识别用户并根据电子邮件地址加入账户。我们可以说，这种策略将仅合并属于同一用户的账户，因为在大多数情况下，电子邮件地址属于唯一用户。但是，由于从数据集中删除了重复的电子邮件密码，因此我们无法观察到用户使用此方法重复使用密码。对于用户拥有多封电子邮件的情况，此策略无法捕获。

 •基于用户名（）。因此，我们考虑了另一种方法，特别是使用电子邮件地址的用户名字段-域名和'@'符号（也称为local-part）之前的字符串来进一步加入可能属于同一用户的账户。如果用户名相同，我们将合并两个电子邮件。在此过程中，我们发现用户重用了30％的密码（请参见图2），这略低于先前的报告（40％）。同样，如图2所示，每个用户密码数量的分布以及属于一个用户的密码对之间的编辑距离的分布与基于电子邮件的加入后的情况发生了巨大变化。我们预计，这是由于属于不同用户的账户的错误合并造成的。

•混合方法（）。为了减少错误的合并，我们最终考虑了两步方法。我们首先根据电子邮件地址加入账户。然后，如果两封电子邮件的用户名部分相等，则将其视为“已连接”，并且与这些电子邮件关联的两个密码至少具有一个共同的密码。然后，所有连接的电子邮件都被视为属于一个用户。因此，属于用户的两封电子邮件可能没有直接的公用密码，但是它们可能与另一封电子邮件共享公用密码。这种启发式方法导致密码重用率达到40％，同时使编辑距离的分布和每个用户的密码数量与仅基于电子邮件的加入所观察到的非常相似。

另一个可能的更宽松的试探法是在多个账户之间匹配用户名。例如，攻击者可能有理由得出结论，“ Alice.Chang@service1.com”和“ AliceChang@service2.com”是同一个人拥有的账户，我们没有详细探讨这种启发式方法。

加入账户后，我们仅考虑数据集中有至少两个密码泄漏的用户，因为训练和测试我们的目标攻击以及个性化强度计需要用户至少输入两个密码-一个密码用作目标账户密码，另一个作为密码泄露。

由于基于用户名的合并技术不准确，因此我们不再讨论。中密码重用率（40％）和子字符串排列率（18.2％）（参见图2），这与先前的研究一致。虽然我们没有确凿的事实证明混合方法得出的结果是正确的，但我们相信，鉴于我们在数据集中拥有的信息，这是用户选择的密码分布的最佳近似值。

我们将清理后的基于电子邮件的数据集分为两部分：（80％）和（20％）。类似地，混合数据集分为（80％）和（20％）。除非另有说明，否则对于所有训练和验证（训练期间），我们均使用。这是因为和中用户的相似（唯一）密码的分布几乎相同。由于我们在训练期间仅考虑用户的相似密码，因此我们不会单独使用进行训练。为了测试，我们使用来自和的随机样本。

1. 密码相似性的生成模型

在本节中，我们描述如何构建一个生成模型，该模型估计输入密码的条件概率分布，其中。密码可以看作是字符序列。 因此，我们可以对字符序列的条件分布进行建模，如下所示：

 (1)

这种密码相似度的公式与统计机器翻译(SMT)问题非常接近，或者更一般地说，是学习序列到序列的翻译。Sutskever等提供了一个非常有效的通用框架来训练序列到序列（seq2seq）模型，而无需明确指定序列代表什么。他们的seq2seq模型使用基于编码器-解码器的体系结构。编码器功能将输入序列映射到某个超参数化d维的实值向量上。该向量简洁地“总结”了输入序列的细节。解码器获取向量v并输出相关输出序列空间中tokens的条件概率分布。

学习密码相似性的一个建议是将seq2seq方法直接应用于字符序列的密码。我们将此模型称为“密码转密码”或“ pass2pass”，但是与以前的工作相比，该技术并未提高性能。在附录A中，我们提供了有关如何训练此模型的详细信息。下面我们描述使用基于编码器-解码器的体系结构对密码相似性进行建模的另一种（更有效的）方法。

**密码到路径模型。**在pass2pass中，我们尝试学习完整密码的条件概率。由于效果不佳，我们决定学习用户可能对其先前密码进行的修改。网站的密码策略可能会影响其中某些修改的选择。但是，我们的相似性模型可以轻松扩展为考虑网站密码策略，为简单起见，我们暂时将忽略密码策略的影响，并考虑所有的密码都是相似的。

我们将对密码的修改视为如下定义的一系列转换。单位转换在指定密码中要应用的编辑。因此，用三元组（）表示，其中e代表要应用的编辑，是字符或空字符串，而是密码中的编辑位置。我们考虑三种类型的编辑：替换（sub），插入（ins）和删除（del）。对于插入和替换编辑，表示要插入或替换的字符；如果表示删除，始终为空字符串⊥。例如，在字符串'password1'上应用转换（sub，'！'，8）意味着用密码'！'替换密码中的第8个（最后一个）字符，那么字符串将变为'password！'。

给定一对具有编辑距离t的密码（，），我们可以找到一系列变换，当以累积方式应用于密码时，它将产生密码。这样的变换就是我们所说的路径。为了计算两个密码之间的路径，我们选择一个最短的密码，在该密码关系中，更倾向使用删除而不是插入和替换来打破联系。路径中的变换按编辑位置排序。（有关更多详细信息，请参见附录C。）例如，从“cats”到“kates”（编辑距离为2）的路径为：{（sub，'k'，0），（ins，'e'，3） }。

在pass2path中，我们在给定另一个密码的情况下定义密码的条件概率，如下所示。



其中t是两个密码和之间的最小编辑距离，并且。

我们使用基于编码器-解码器的模型，其中解码器函数的输出是T变换中的概率分布。 这样，我们可以将上面的等式重写为



其中是编码器的输出，是解码器在输入和上的输出，并且是特殊的路径开始符号。包含来自的信息，因此替换了最终方程中的。 我们将学习此概率模型的任务设置为有监督的学习任务，其训练目标是找到使各个用户选择的密码提示之间正确编辑路径的对数概率最大化的参数θ。 设D为此类密码广播的集合，则训练目标为



pass2path的模型架构类似于用于seq2seq的基于编码器和解码器的架构，该架构使用两个递归神经网络（RNN）实例化。编码器和解码器的RNN一起训练，附录B中给出了模型体系结构的详细信息。下面，我们将介绍pass2path的一些进一步的训练详细信息。 训练后,给出的模型可用于生成类似的密码泄露中的密码。关于如何生成q个最可能的密码的详细信息将在第五节中给出。

**训练pass2path。**我们使用基于电子邮件数据集（）创建的数据来训练密码模型。对于数据集中的每个用户，我们计算所有密码对包括密码对的重新排序，从而产生8.23亿个密码对。

我们将密码表示为美式键盘上的按键序列（按键序列）。例如，“ PASS-WORD！”表示为“<c>password<s>1”，其中<c>和<s>表示键盘上的大写锁定和Shift键。 Chatterjee等表明密码的键序表示法可以有效地改善密码错字校正，我们在这里使用它是因为它比标准编辑距离更好地捕获了与大写有关的变换。

对于训练集中的每对密码，我们使用基于动态编程的算法生成了它们之间的最小路径。该算法是精算算法的扩展，用于计算字符串之间的最小编辑距离。给定一对密码和，我们首先将密码转换为密钥序列，然后找到可以将转换为的转换路径。我们在附录D中详细描述了我们的算法。

少量密码对的手动样本显示，其中很大一部分是完全不同的，没有任何明显的语义或句法相似性。因此，我们决定在训练之前根据路径长度（也等于密码之间的键序编辑距离）过滤密码。给定一个截止值δ，我们只考虑路径长度最大为δ的密码对。我们从δ= 2开始，完成三个时期的训练，然后通过添加更多对δ= 3、δ= 4的对来逐步转移学习网络。我们发现这种方式收敛速度更快，并且具有更高的准确性。总体而言，该模型是针对1.44亿个密码对进行训练的，有关训练pass2path的更多详细信息，请参见附录C。

pass2path模型有240万个参数，占用磁盘上60兆字节的存储空间。在Nvidia GTX 1080 GPU和Intel Core i9处理器上训练批量大小为256的模型大约花了两天时间。然而，这项训练只需要不到2 GB的物理内存。

5.使用path2path进行定向攻击

如第二节所述，学习密码相似度的一个主要应用是了解目标猜测攻击的危险，在此情况下的对手根据用户的其他密码生成猜测密码。在本节中，我们将描述如何从我们训练有素的pass2path模型中获得数千个猜测，以建立有效的针对性攻击。我们将通过仿真证明我们的攻击优于所有先前的猜测攻击。

**生成相似密码。**为了将密码相似性模型用于有针对性的攻击，我们需要能够在给定泄露密码的情况下，以似然度递减的顺序生成一系列密码，，…，。即 for i <j。 这里的q是一些猜测，我们将在下面对一个参数进行具体化。给定我们的pass2path模型下生成非常简单。首先，将输入密码转换为固定维向量，并将其与特殊的序列开始符号一起馈送到解码器。解码器在一组变换T上输出概率分布。在每次迭代中选择最可能的输出，并将其用作下一次调用解码器的输入，直到到达序列结束符号。然后将转换的输出序列应用于输入密码以生成新密码。

但是此过程仅输出最可能的密码。为了产生多个输出，我们使用了广度优先光束搜索技术。波束搜索算法使用一组大小为q的波束，该波束在每次解码迭代时都存储到目前为止生成的q个最可能的路径（以及网络状态和概率）。如果路径以结束序列转换结束，我们将其称为完整路径，否则将其称为不完整路径。利用解码器在向量的输入和序列开始符号上输出的q个最可能的变换来初始化波束。接下来，对于当前存储在波束中的每个不完整路径，在路径的最后一个变换上调用解码器，并通过将变换附加到路径来计算新路径。光束中仅保留q个最可能的新建路径，以进行下一次迭代。重复此步骤，直到达到预定义的最大迭代计数或完成光束中的所有路径为止。

光束搜索是一种贪婪算法，不能保证提供q个最可能的路径。但是，它是一种广泛使用的启发式方法，用于在有限的内存和时间的情况下找到前q个猜测。要查找输入密码的q条路径，波束搜索将最多对解码过程进行q·t次调用，其中t是表示模型中允许的输出路径的最大长度的参数。

由于可能存在多个路径，当应用于密码时会输出相同的密码w，因此波束宽度为q的波束搜索可能不会生成q个唯一的密码。 因此，我们生成q’≥q个密码，然后输出前q个唯一密码。 在我们的实验中，我们发现取q’= 2·q足以为超过99.9％的密码找到q个唯一密码。

**评估有针对性的猜测攻击。**我们基于有多少用户的账户会因为另一个泄露的密码而被攻破的情况对有针对性的攻击进行了评估。我们主要关注在线攻击设置，在这种情况下，如果尝试登录失败的次数过多，则应阻止账户登录（即，如果没有带外身份验证，则将不允许登录）。我们称查询预算q为尝试次数，也是攻击者阻止账户之前可供攻击者使用的最大猜测数。为了评估攻击，我们将使用q∈{10,100,1000}的猜测预算。这些是身份验证服务使用的典型值。

我们使用两个测试数据集：（仅使用电子邮件来识别用户），以及（使用混合方法生成）（请参见第III节）。第一个模拟攻击是已部署凭据填充对策的服务，例如，如果用户先前的密码存在泄漏，则强制用户选择新密码。由于不允许在两个账户之间重复使用密码，因此以下将其称为“无重复”设置。第二个模拟攻击没有部署这种对策的服务，因此我们将其称为“重复”设置。

这两个数据集都包含数百万个用户。我们评估的某些目标攻击在计算上非常昂贵。我们需要选择一个较小但有代表性的测试数据样本。我们计算了用户针对不同测试集大小使用相同和相似密码的比率的差异。对于大小≥105的测试集，我们发现方差足够低（<0.5％）。因此，我们为每个数据集随机抽取105个随机用户进行评估。对于每个选定的用户，我们从与该用户关联的多组密码中随机选择两个密码而不进行替换-其中一个（随机选择）被视为泄露密码，另一个被视为目标密码w。

我们将攻击算法与现有的两种有针对性的猜测攻击进行了比较。Das等创建了一个手动管理的转换列表以生成类似的密码。 Wang等根据有关用户的信息提供了多种攻击，包括他们的人口统计信息，其他密码以及它们的组合。我们将集中讨论[14]中的TarGuess-II攻击，该攻击仅在知道先前密码的情况下进行。Wang等慷慨地提供了Das等的实现、算法及其TarGuess-II算法。后者需要从数据集中进行训练。有关详细信息，请参见附录E。

我们还针对仅基于训练集中密码的经验分布的两种攻击进行了比较。第一个是无目标攻击，它简单地（对于任何泄露的密码）猜测训练数据集中用户的q个最常用的密码。我们发现，在少量猜测（例如q≤104）的情况下，该非目标经验模型的性能优于最新的非目标猜测攻击[24]。

第二种是有针对性的经验攻击，对于给定的泄露密码w，攻击者输出同时使用密码w的用户最常用的q个密码。尽管这种有针对性的经验攻击从概念上讲是简单明了的，但它需要大量的有效访问内存才能实现。因此，我们通过计算在测试数据中泄露的密码的经验分布来模拟这种攻击的有效性。

我们将泄漏的密码用作所有设置中所有定向攻击的第一个猜测。无目标的经验攻击会在所有设置中为所有帐户使用固定的q个猜测列表。

对于更高的查询预算值q，某些攻击无法对某些泄露的密码产生q猜测。在这些情况下，我们只是中止了攻击而没有用完剩余的查询预算。在实践中，人们可能会尝试以某种临时方式扩展猜测的数量，例如，通过添加非目标猜测。放眼未来，这样的修饰无法充分改善攻击以赶上pass2path的速度。

在图3中，我们展示了两种设置下不同攻击的功效。我们依次讨论每个设置的结果。

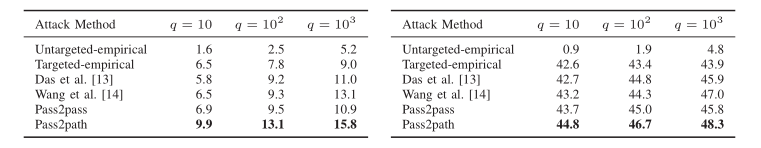


图3：在由无重复设置（左）和有重复设置（右）生成的两个测试集中，q猜测中各种攻击所猜测的密码百分比。 在后一种情况下，猜测性能的显着提高是因为40％的目标密码与泄露的密码相同。

**无重复设置时的攻击效率。**首先，我们讨论结果（左表），其中目标密码与泄露的密码不同。值得注意的是，对于查询预算q = 10，基于经验分布的目标攻击的性能要优于所有先前的目标攻击。但是，它缺乏可概括性，因此无法在较高的查询预算下发挥作用。平均而言，每个密码在训练数据集中只有几个关联的密码，这种攻击只能猜测在训练数据集中观察到的密码。

Das等攻击不需要训练数据，是我们测试的所有目标攻击中执行速度最快的。它的表现相对较好，在不到一千次的猜测中破解了11％的用户帐户。但是，对于许多密码而言，这种有针对性的攻击无法产生1,000次猜测（因为它用尽了处理规则以应用于泄漏的密码）。Wang等[14]算法是我们工作之前最先进的针对性猜测攻击。它可以在不到1,000个目标猜测中破解13％的用户帐户。但是，猜测的生成过程非常缓慢，需要花费三天多的时间才能在具有Core i9 CPU和128 GB内存的机器的一个线程上仅对我们的一个测试集中的所有密码进行猜测。尽管在在线猜测攻击中，计算复杂度并不是特别重要（与试图破解哈希的离线猜测攻击不同），我们之所以提到它是因为它在我们的仿真中证明是一项重大的工程挑战。

最后，pass2path在所有攻击中表现最佳，在100个猜测中破解了约13％的密码（比Wang等破解的密码高40％），在1,000个猜测中破解了15.8％的密码（增长了20％）。 pass2path算法的计算速度也相对较慢，需要四个小时的计算才能评估测试集。这仍然比Wang等的速度要快得多。