计算机工程
Computer Engineering
ISSN 1000-3428,CN 31-1289/TP

《计算机工程》网络首发论文

题目：　基于改进 PCFG 算法的口令猜测方法
作者：　李静雯，赵奎
DOI：　10.19678/j.issn.1000-3428.0064678
网络首发日期：　2022-11-30
引用格式：　李静雯，赵奎．基于改进 PCFG 算法的口令猜测方法[J/OL]．计算机工程.
https://doi.org/10.19678/j.issn.1000-3428.0064678

网络首发：在编辑部工作流程中，稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定，且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式（包括网络呈现版式）排版后的稿件，可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定；学术研究成果具有创新性、科学性和先进性，符合编辑部对刊文的录用要求，不存在学术不端行为及其他侵权行为；稿件内容应基本符合国家有关书刊编辑、出版的技术标准，正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性，录用定稿一经发布，不得修改论文题目、作者、机构名称和学术内容，只可基于编辑规范进行少量文字的修改。

出版确认：纸质期刊编辑部通过与《中国学术期刊（光盘版）》电子杂志社有限公司签约，在《中国学术期刊（网络版）》出版传播平台上创办与纸质期刊内容一致的网络版，以单篇或整期出版形式，在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊（网络版）》是国家新闻出版广电总局批准的网络连续型出版物（ISSN 2096-4188，CN 11-6037/Z），所以签约期刊的网络版上网络首发论文视为正式出版。

网络首发时间：2022-11-30 14:39:16
网络首发地址：https://kns.cnki.net/kcms/detail/31.1289.TP.20221129.1550.001.html

计 算 机 工 程
Computer Engineering

基于改进 PCFG 算法的口令猜测方法

李静雯，赵奎

（四川大学，网络空间安全学院，成都 610065）

摘　要：近年来口令泄露事件频出，有效的口令猜测方法是保障口令安全的重要手段，其中基于概率上下文无关文法（Probabilistic Context-Free Grammar, PCFG）的方法效果显著，然而，此方法目前仍存在无法生成新的口令字符子段、对生成口令的概率估计不准确等问题。为此，以基于 PCFG 的口令猜测方法为研究对象，对其在口令构造过程中关键阶段的命中率进行分析，并提出了改进的基于 Backoff-RNN 及概率平衡的 PCFG 口令猜测方法。在口令结构划分阶段，通过分析用户在构造口令时的行为及偏好，将口令从中文拼音和英文单词两方面进行更细粒度的结构划分，提取口令更深层次的结构信息；在口令填充阶段，将 Backoff 的思想应用于字符级 RNN 模型中，用以生成子结构中长序列字符子段，提高了模型的准确性和泛化能力；在口令概率计算阶段，改进口令生成概率的计算方法，解决使用传统计算规则时，因口令结构长度不一致所带来的概率不平衡问题。实验结果表明，所提方法在漫步口令猜测攻击中，针对中英文两种语言环境的交叉数据集，命中率较传统 PCFG 方法分别提升了 20.6%及 22.4%，在定向口令攻击中，较 TarGuess-I提升了 2.8%，验证了所提方法的可行性与有效性。

关键词：口令猜测攻击；自然语言处理；概率上下文无关文法；深度学习；口令安全

开放科学（资源服务）标志码（OSID）：

Password Guessing Method Based on improved PCFG

LI Jingwen, ZHAO Kui

(School of Cyber Science and Engineering, Sichuan University, Chengdu 610065, China)

【Abstract】In recent years, password leaks occur frequently, effective password guessing methods are an important means of securing passwords, among them, the method based on Probabilistic Context-Free Grammar (PCFG) is extremely effective, however, this method still has problems such as the inability to generate new substrings of password and the inaccurate estimation of the probability of generating passwords, etc. Thus, taking the PCFG-based password guessing method as the research object, this paper analyzes its hit rate in the key stage of the password generation process and proposes an improved PCFG password guessing method based on Backoff-RNN model and probability balance. In the password structure division stage, by analyzing the user's behavior and preference when constructing the password, the password is more finely divided into Chinese pinyin and English words to extract the deeper structure information; In the password filling stage, the idea of Backoff is applied to the char-RNN model to generate long sequence substrings in the substructures to improve the accuracy and generation ability of the model; In the password probability calculation stage, the calculation method of password generation probability is improved to solve the problem of probability imbalanced caused by inconsistent length of password structure when using the traditional calculation rules. The experimental results demonstrate that the hit rate of the proposed method is 20.6% and 22.4% higher than PCFG in the cross-dataset of Chinese and English language environments in the Trawling Attacking, and 2.8% higher than TarGuess-I in the Targeted Attacking, which verifies the feasibility and effectiveness of the proposed method.

【Key words】password guessing attack; natural language processing; probabilistic context-free grammar; deep learning; password security

DOI: 10.19678/j.issn.1000-3428.0064678

0　概述

口令认证凭借其实现简单、低成本、高效率的特点在众多身份认证方式中占据主流地位。然而近

基金项目：国家自然科学基金(U19A2068,61872254)

作者简介：李静雯（1998—）、女、硕士研究生、主研方向为大数据安全、口令安全；赵奎(通讯作者)，教授、博士。

E-mail：zhaokui@scu.edu.cn

年来随着口令泄露事件的发生，口令的安全性受到冲击。因此，从攻击者的角度进行口令猜测攻击，对保障用户口令安全具有重要意义，其研究难点在于如何从已泄露的大规模口令样本中挖掘出用户普遍的口令构造方式，并在此基础上构造口令猜测方法。

口令生成虽可看作文本生成任务，但口令具有结构化明显，语义语法弱的特点。基于概率上下文无关文法 (Probabilistic Context-Free Grammars, PCFG)[1] 的口令猜测方法正是利用该特点，在对真实口令数据进行统计分析的基础上，对口令结构及各结构子段进行频次统计，以此生成新的口令，此方法命中率较高，目前应用广泛。

因此，本文将以 PCFG 方法为研究对象，进一步探索如何更好地抽象口令的基础语法结构、提高算法对口令子段的命中率，这将成为提高口令猜测算法攻击命中率的关键。目前基于 PCFG 的口令猜测算法仍存在以下问题：

在口令结构划分阶段，传统 PCFG 方法仅从字符类型的角度对口令进行分割，忽略了字符串中更细粒度的信息。中文和英文作为全球使用最广泛的两种语言，未对两者进行对比分析与提取。

在口令填充阶段，传统 PCFG 方法无法生成结构中新的口令子段，虽已有研究者结合 Markov 模型[2][8] 或循环神经网络(Recurrent Neural Network, RNN)[10]来解决这一问题，但这两种方法目前仍面临模型训练时序列长度难以确定的问题：首先，长度小于所选定阶数的子段无法作为训练数据参与模型训练；其次，序列长度的选择不一定适用于所有的口令数据样本，使用高阶模型会带来数据稀疏问题，而选定低阶模型则可能因使用的历史字符信息太少，导致对当前位置输出结果的概率估计不够精确，模型泛化能力偏弱。

在口令概率计算阶段，传统 PCFG 方法将一条口令中各子段的概率累积作为其生成概率，然而这种计算方法所生成的概率受口令结构长度的影响较大，存在概率计算不平衡的问题。

针对这些问题，本文对传统 PCFG 方法进行改进，提出了基于 Backoff-RNN 及概率平衡的 PCFG 口令猜测方法。首先，在对四个大规模口令集（2 个中文背景口令、2 个英文背景口令）分析的基础上，挖掘出中英文语言背景下口令构造的差异，对口令从汉语拼音、英文单词上进行更细粒度划分，提高模型的准确性；同时，将 Backoff[3]的思想引入 char-RNN 模型中，在生成长序列口令子段时，根据已生成的子串动态选择适合长度的 RNN 模型，动态平衡模型拟合问题，使得生成的子段更符合真实训练样本中的序列关系。最后，将困惑度(perplexity)的计算方法引入口令生成概率的计算规则中，使得改进后的概率计算方法更能体现出口令在口令集中的真实分布规律。

1 相关工作及研究背景

口令猜测算法根据攻击过程中是否利用用户个人信息可以被分为漫步攻击（Trawling Attack）以及定向攻击 (Targeted Attack)。前者不针对特定的攻击对象，唯一目标是在允许的猜测次数下，提高模型对攻击样本的命中率。后者则是在给定目标用户个人信息的前提下，以更少的猜测次数（$\leqslant 10^4$），有针对性地猜测该用户的真实口令。

针对上述两种攻击方式，目前主流的口令猜测方法可分为以下三类，分别是基于马尔可夫链（Markov）[2]的猜测方法、基于 PCFG[1]的口令猜测方法以及基于神经网络的口令猜测方法。

1.1 基于 Markov 的口令生成算法

Narayanan 等人于 2005 年首次提出了一种基于 Markov 模型的口令猜测方法[2]，该方法根据字符序列之间的转移概率来指导口令的生成。

在 n 阶 Markov 中，下一个字符出现的概率基于它前面长度为 n 的子串。以 4 阶 Markov 为例，在训练阶段，口令“Li1234”需要统计出：首字符是“L”的频数、“L”后是字符“i”的频数、“Li”后是字符“1”的频数、“Li1”后是字符“2”的频数、“Li12”后是“3”的频数、“i123”后是“4”的频数。在遍历训练集中的每个口令之后，便可得到各子字符串之间的转移概率矩阵。在生成阶段将各部分概率累乘便可得到目标口令的生成概率，以“Li1234”为例，其生成概率的计算公式如式（1）。最后将生成的口令按概率大小进行排序，生成口令攻击字典。

$$P(Li1234)=P(L)\times P(i\mid L)\times P(1\mid Li)\times P(2\mid Li1)\times P(3\mid Li12)\times P(4\mid i123) \tag{1}$$

基于 Markov 的口令生成方法可以反映出字符之间的序列关系，但其也只是简单地对已有样本进行概率统计，无法学习字符序列间的高阶特征，模型的生成效果易受训练数据的影响，容易过拟合，同时该方法也忽略了口令“重结构轻语义”的特征。

1.2 基于 PCFG 的口令生成算法

1.2.1 实现思路

PCFG 算法[1]将整个口令段划分为字符子段（L_n）、特殊字符子段（S_n）及数字子段（D_n），其中 n 表示该子段的长度，如口令“li#123456”可被划分为基本结构 $L_2S_1D_6$ 及子段“li”、“#”、“123456”。

如图 1 所示，在训练阶段，首先提取出训练集中所有口令的基本结构，以及被分割出的子段，在此基础上构建出口令结构及各子段的频率字典。

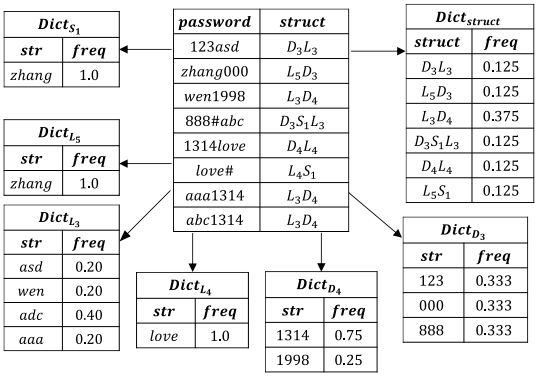

图 1 基于 PCFG 的口令统计过程

Fig.1 Password Statistics Process Based on PCFG

在口令生成阶段，根据上述频率字典，按照频率大小依次取出口令结构，对其进行填充。例如，首先从口令结构频率字典中取出当前频率最高的结构 L_3D_4，再分别取出 L_3 及 D_4 子段表中当前最高频率的子段“abc”和“1314”进行填充，口令“abc1314”的生成概率计算规则如式（2），以此构建出按生成概率递减的猜测口令集。

$$P(abc1314)=P(S=L_3D_4)\times P(L_3=abc)\times P(D_4=1314) \tag{2}$$

1.2.2 现有基于 PCFG 的改进方法

2014 年，Veras 等[4]在字母子段上使用分词、词性标注和归一化等自然语言处理工具分析字母子段的语义特征。2015 年，Houshmand 等[5]在 PCFG 方法的基础上加入键盘词模式和多字模式以提升方法的命中率。2016 年，Li 等[6]和 Wang 等[7]结合用户姓名、生日、邮箱等个人信息，分别提出了 personal-PCFG 和 TarGuess-Ⅰ模型，实现定向猜测用户口令。2019 年，章等[8]考虑到传统 PCFG 无法生成结构中新的子段的弊端，因此将 PCFG 方法的结构划分优势和 Markov 模型在表示前后字符依赖关系中的优势相结合，提出了 SPSR 模型，有效地提升了猜测模型的准确性和泛化能力。

基于 PCFG 的口令猜测方法能够准确地抽象出口令的结构，且生成的口令可按照口令概率排序。但现有的概率计算方法使得口令的生成概率受口令结构长度的影响较大，导致概率计算不平衡。

另外，本文经实验发现，PCFG 方法对于字符序列长度在 4 以内的短序列命中率较高，但随着字符序列长度的增加，其命中率会不断降低，如何运用此特征指导口令生成将是本文研究的重点。

1.3 基于神经网络的口令生成算法

近年来，生成式对抗网络（Generative Adversarial Networks，GAN）、循环神经网络（Recurrent Neural Network, RNN）在文本生成领域的不断发展为口令生成提供了新思路。

2019 年，Hitaj 等人[9]提出 PassGAN 模型，首次利用 GAN 来学习真实口令的构造规则，模拟生成猜测口令集。但 GAN 模型无法给出生成口令的概率并且针对文本这类离散数据无法进行反向传播，这两个弊端使得基于 GAN 的口令猜测方法在命中率上低于传统方法。

2016 年，Melicher 等人[10]首次使用 RNN 进行口令破解，Zhou 等人[11]将个人信息与 RNN 相结合，提出了定向口令猜测模型 TPGXNN。另外，长短期记忆网络(Long Short-Term Memory, LSTM)作为 RNN 的一个变种，具备长期记忆能力，因此，Xu 等人[12]使用 LSTM 进行口令破解，2018 年 Liu 等人[13]提出 PL 模型，将 PCFG 与 LSTM 结合进行口令猜测。

上述基于循环神经网络的口令猜测算法和流程与 Markov 模型类似，但通过前者得到的概率分布要比基于概率统计的 Markov 模型更加合理，可提取字符间的隐含特征，所以上述方法相较于传统模型均有效地提高了猜测模型的命中率。

LSTM 通过引入多种门操作解决了长期依赖问题，但与此同时也增加了模型结构的复杂度及计算量。对于口令数据而言，一条口令的长度通常被限制在 8-20，再经过 PCFG 方法的结构拆分后，子串的长度只会更短，在训练和生成的过程中并不会存在明显的长期依赖问题。2021 年，Wang 等人[14]在 PL 模型[13]的基础上，将其中的 LSTM 模型替换为 RNN，提出 PR 模型。实验结果显示，后者的命中率普遍略高于前者，同时训练效率远高于前者。因此，在结合 PCFG 方法时，RNN 模型相较于 LSTM 在口令生成中更具有优势。

现阶段无论是 Markov、RNN 还是 LSTM 模型，它们都是在选定模型阶数后进行口令数据的训练与生成，均存在引言中所述的弊端。

2 用户口令行为分析

2.1 用户口令数据集

口令猜测方法建立在对大规模真实用户口令集的分析工作的基础上。因此，本文选取了四个国内外泄露的大规模口令样本进行统计分析，综合对比中英文背景用户在构造口令时的习惯差异，为提出本文的拼音和单词的划分策略提供依据，本文采用的四个口令数据集的基本信息如表 1 所示。

表 1 数据集中的口令信息

Table 1 Password information in the dataset

口令来源	类型	口令数量	用户语言	包含个人信息
12306	铁路系统	129303	中文	是
CSDN	程序员论坛	6428632	中文	否
rockyou	社交网络	32581870	英文	否
yahoo	门户网站	5626485	英文	否

2.2 中英文语言背景下常用字符串统计

本节对中英文流行口令进行具体分析。首先，分别对数据集中的每一个口令提取长度为 3 至 10 的口令子串序列，分别构造出如下表 2 及表 3 所示的中英文常用口令子串。

表 2 中文背景下用户常用口令子串

Table 2 Common sub-password segments in the Chinese background

序列长度	常用子串
3	123, 520, 111, 000, aaa, abc, asd, qwe, 666, 888
4	aini, 1314, 1234, love, woai, 6666, 8888, a123
5	12345, aaaaa, 66666, 11111, 00000, a1234, woshi, ilove
6	999999, 000000, 321321, 112233, qwerty, 121212
7	1234567, 5201314, zxcvbnm, a123456
8	87654321, aaaaaaaa, 12345678, 11111111, 00000000, 888888888, 66666666, iloveyou, password, 1q2w3e4r
9	123456789, 987654321, a12345678, qwertyuiop
10	woaini1314, a123456789, 0123456789, 1234567890

表 3 英文背景下用户常用口令子串

Table 3 Common sub-password segments in the English background

序列长度	常用子串
3	123, and, all, 000, one, ass, son, aaa, ann, her, 111
4	love, 1234, ball, baby, 1111, ever, rock, life, a123, hell
5	ilove, 12345, angel, hello, 11111, lover, jesus, lucky
6	monkey, qwerty, prince, dragon, christ, jordan, flower
7	1234567, welcome, michael, diamond, charlie, anthony
8	password, princess, sunshine, iloveyou, november
9	123456789, butterfly, chocolate, Elizabeth, beautiful
10	basketball, tinkerbell, strawberry, volleyball, sweetheart

观察两表，可以直观地看出英语背景用户在构造口令时，除了简单数字序列以外，更多的选择常用英文单词，如：baby, girl, sweet 等。而国内用户则会更频繁且集中地使用简单的字母串和数字串。

2.3 中英文语言背景下拼音及英文单词的占比统计

在上一节的统计结果之上，本节对国内口令数据中的拼音占比、国内外口令样本中的英文占比、混合字母串占比进行统计分析。如表 4，在 12306 数据集包含字母的口令中，包含拼音的口令约占 41.5%，在 CSDN 数据集中，这一比例高达 73.1%，英文单词在以上两口令集中的占比均达到 30%至 40%。在 rockyou 和 yahoo 的数据集中英文单词的占比分别达到了 49.2%和 76.3%。这一统计结果说明拼音在国内用户的口令构造以及英文单词在英文背景用户的口令构造中占据重要的地位。基于上述分析结果，本文认为对口令中的拼音和英文单词进行提取和标注将会有针对性地提高口令破解的命中率。

表 4 口令集中拼音、英文单词的占比统计

Table 4 Statistics on the proportion of Pinyin and English words in the password set

口令来源	包含字母的口令占比	在所有包含字母的口令中		
		包含拼音的口令占比	包含单词的口令占比	混合字母串占比
12306	72.5%	41.5%	29.7%	47.6%
CSDN	56.8%	73.1%	43.6%	49.2%
rockyou	92.3%	-	49.2%	49.4%
yahoo	94.3%	-	76.3%	47.1%

本节还统计了混合字母子串在包含字母的口令集中的占比，混合字母串表示同时包含拼音子串、英文单词子串、普通字母子串这三种子串中任意两种及以上的一段连续字母子串。在四个数据集中，占比均在 50%左右，在这种情况下，传统 PCFG 方法仅将字母段提取为以长度为单位分割的字母子段 L_n 则无法体现出用户口令构造时的深层结构特征，所以本文在此统计结果的基础上，对字母子段从拼音和英文单词两方面进行更细粒度的标记与提取。

3 基于 Backoff-RNN 及概率平衡的 PCFG 口令猜测方法

本文所提方法由口令结构划分、多阶 RNN 模型训练和口令生成三个模块构成，图 2 为方法框架图。首先，将数据集中的所有口令按照个人信息、拼音、英文单词、字母子段、数字子段和特殊字符子段进行细粒度分割，得到口令结构及各子段的频率字典，并将各子段作为多阶 RNN 模型的训练数据。接着，利用 Backoff-RNN 模型生成长度大于 4 的字母子段、数字子段、特殊字符子段。最后，依次对口令结构字典中的口令结构进行填充，利用改进的概率计算规则对生成口令进行概率计算，并按照概率从高到低的顺序对口令进行排序，生成口令猜测字典。

本文所提方法保留了 PCFG 方法在抽象口令结

构、对短序列字符串命中率高的优势的同时，对其在口令结构划分、生成长序列子段及概率计算上存在的不足进行了改进。

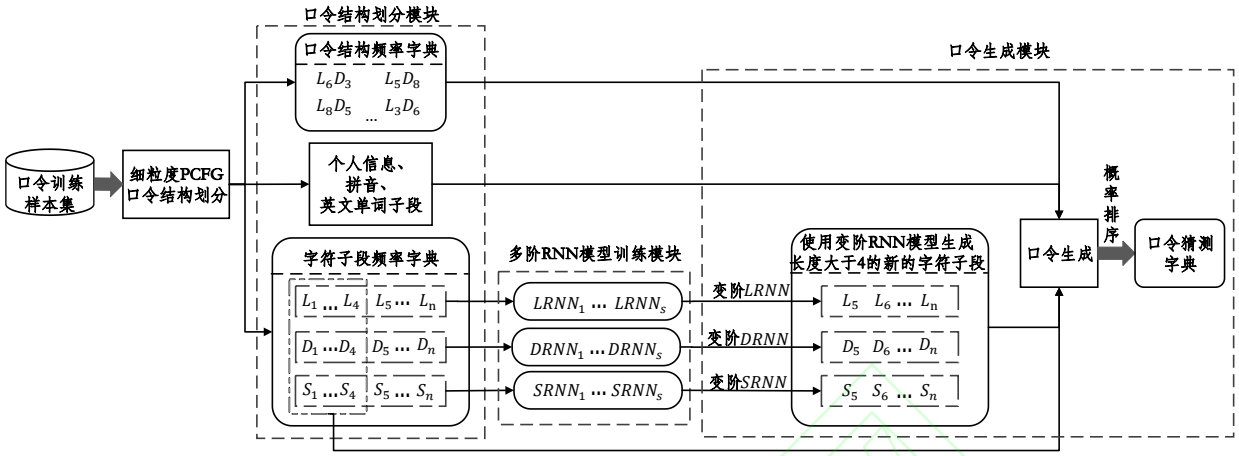

图 2　基于 Backoff-RNN 及概率平衡的 PCFG 口令猜测方法

Fig.2 PCFG password guessing method based on Backoff-RNN and probability balance

3.1 口令结构划分模块

根据第 2 节的分析，本节在传统 PCFG 方法的基础上，实行更细粒度的结构划分策略：将用户口令依次从 PI (个人信息)、W (英文单词)、P (拼音)、L（常规字母子段)、D（常规数字子段)、S（特殊字符子段）这 6 个类别进行口令结构提取。

(1) 个人信息字段。在进行定向口令猜测攻击时，本文借鉴 TarGuess- I [7]中对用户个人信息的划分策略，从姓名、生日、邮箱前缀来提取口令中的个人信息，但与其划分规则稍有不同的是，本文未在“年月日”、“月日年”这类仅调换序列顺序的数据上进行抽取，而是通过将其拆分为“年”、“月日”、“日月”等更细粒度的字段，以容纳用户更多形式的结构变换。

（2）拼音。本文将一个拼音看作一个整体，在统计时只对拼音的个数进行提取，例如“mima”将被抽象为 P_2。

（3）英文单词。本文搜集了常用英文单词、英文名和地名等构成英文词典，用和拼音同样的处理方式对英文单词序列进行提取。

（4）最后，再将余下的字符串分别替换为字符子段、特殊字符子段及数字子段。至此，便完成了口令的结构划分，得到了口令结构以及各子段的频率字典。

3.2 多阶 RNN 模型训练模块

利用上一阶段拆分出的数字子段、字母子段、特殊字符子段分别训练出基于数字的 $DRNN_i$、基于字母的 $LRNN_i$ 及基于特殊字符的 $SRNN_i$ $\left(i \in [1,s]\right)$ (s 为变阶 RNN 模型中的最高阶数)，并提出了基于 Backoff 思想[3]的自适应变阶 RNN 模型来生成口令子段。在生成子段时根据已生成子串，自动选择应使用几阶 RNN 模型去预测下一位生成字符，具体的实现方式将在 3.3.1 节进行论述。

模型训练阶段类比于 char-RNN：首先，在预处理阶段，对提取出的每个子段尾部添加结束符 $\langle EOS \rangle$。训练过程中，对于 i 阶 RNN 模型 RNN_i，选择长度不小于 i 的子段（不包含结束符），作为其训练数据，对每一个子段，从第一个字符开始，以滑动窗口的方式，截取窗口大小为 i 的子串作为模型的输入序列，并将当前窗口后的下一个字符作为标签，滑动窗口以 1 为步长不断向后滑动，直到获取到的字符标签为结束符。以训练基于数字的模型 $DRNN_i$ 为例，训练过程如图 3 所示。

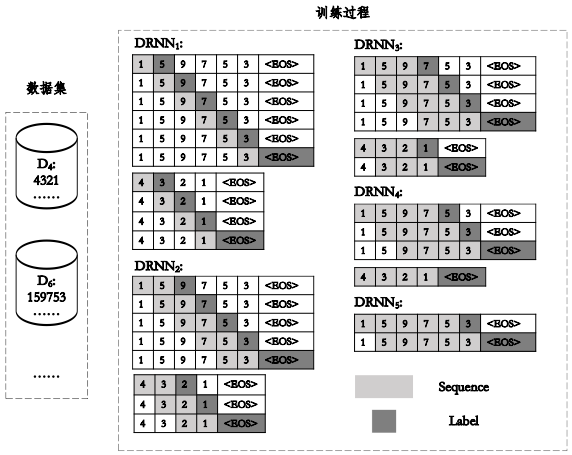

图 3 模型训练示例

Fig.3 Model training example

3.3 口令生成阶段

3.3.1 Backoff-RNN

Katz 提出的 Backoff[3]是一个经典的语言平滑模型，在 n-gram 模型中，高阶数据可以更好地利用历史信息，但同时也会面临数据稀疏的问题，此时如果相对低阶的语法出现的频率更高，那么结果会更加可靠。方法的原理是：首先，为作为输入数据参与模型训练的 n 元语法(对应为 n 阶 RNN 模型的所有输入序列）构建频率字典，在此基础上为 n 元语法的出现频率设置一个阈值 $threshold\_n$ （1 元语法无需设置)，对于子段 $x_1x_2x_3\ldots x_t$，用 $S_{p,q}$ 代表子段从位置 p 到 q 的子串，$p(x_l)$ 表示在 $S_{1,l-1}$ 后出现 x_l 的概率 $(l>2)$，在计算 $p(x_l)$ 时，Backoff 算法会寻找一个最小的 p，保证 $S_{p,l-1}$ 的出现频率大于等于阈值 $threshold\_l-p$，此时，将 $S_{p,l-1}$ 作为 RNN_{l-p} 的输入序列，得到下一位预测字符及其对应的生成概率。通过利用稀疏的高阶语法中的低阶语法对其进行平滑，这样可以利用给定历史信息选择最可靠的模型来提供更好的预测结果。

本文基于上述思想，采用算法 1 来生成长度大于 4 的子段。首先，遍历长度为 i 的待生成子串频率字典 dict_i（键 *item* 为当前待生成子串，值为该子串对应的生成概率)，同时根据训练数据频率字典 dict_n 确定长度为 n 的语法的频率阈值 (dict_n 的键为 n 元语法，值为该 n 元语法在训练集中的出现频率)，从 item[-s:]开始，找到满足阈值的最大子串 *win*，其长度 *len* 便是 RNN 模型的阶数，*win* 作为 RNN_{len} 的输入，得到 *item* 的下一位预测字符字典 dict_pred(字典的键值对分别为预测字符 *pred* 以及对应的生成概率)，当生成字符为结束符时，即生成一个完整的字符串，更新 dict_generated_i，否则更新 dict_i+1，并将其作为长度为 *i+1* 的待生成子串。最后需统一对同一长度的已生成子段的概率进行归一化。

特别地，在初始阶段生成长度为 5 的口令子段时，需要以训练数据频率字典 dict_4 为初始输入数据，得到长度为 5 的待生成子串字典 dict_5 后，再将其作为下一步骤的输入数据。

算法 1 基于 Backoff-RNN 的口令字符子串生成算法（生成子段长度为 i)

输入 训练数据频率字典 dict_n（$n\in[1,s]$)，长度为 i 待生成的口令子串频率字典 dict_i(i>4)

输出 已生成的长度为 i 的子段频率字典 dict_generated_i，待生成的长度为 i+1 的子串频率字典 dict_i+1

```
1.  FOR item IN dict_i:
2.      IF dict_s[item[-s:]]⩾threshold_s:
3.          win ← item [-s:]
4.      ELSE IF dict_(s-1)[item[-(s-1):]]⩾threshold_(s-1):
5.          win ← item [-(s-1):]
6.      …如此反复判断，直到得到满足条件的子串
7.      updateSubString(win, item)
8.  END FOR
9.  def updateSubString(win, item):
10.     len = win.length
11.     dict_pred_freq ← RNN_len (win)
12.     FOR preq IN dict_pred:
13.         poss = dict_i[item] * dict_pred[pred]
14.         IF pred == <EOS>:
15.             dict_generated_i[item] = poss
16.         ELSE
17.             dict_i+1[item+pred] = poss
18.     END FOR
```

3.3.2 改进的概率计算方法

PCFG 口令猜测算法是一种典型的基于概率的方法，它使用统计学方法从训练集中学习口令的结构分布以及各长度子段的分布，并通过概率来刻画每一条口令的分布规律，在构造口令猜测字典时，按照概率递减的顺序枚举生成的口令，确保在更少的猜测次数下猜出尽可能多的口令，这也是基于 PCFG 的口令猜测方法优于众多传统方法的原因。

我们认为，最优攻击者生成的猜测集应与测试集完全相同，即按口令概率递减排序后的猜测字典应与按出现频次递减排序后的测试集一致。所以，猜测算法给出的概率值应在最大程度上反映出该口令在口令集中的真实频次，这样才能确保基于概率的方法能够有效地根据口令概率加速口令破解。

如式（2）所示，传统 PCFG 方法的概率计算规则是将口令结构概率和各子段的概率累乘，然而这种计算方法会导致概率计算不平衡的问题：结构越长的口令在概率连乘的情况下概率值必然会越小，例如结构长度为 3 的口令生成概率普遍会比长度为 2 的口令低一个数量级，从暴力破解方法的实现思路来看，这似乎没有什么问题，一条口令的长度或

是结构长度越长，其猜测难度也会相应增大，但根据上述所分析的基于概率的猜测方法的概率生成目的来看，对于一条结构较长的口令 A，只要其在口令集中出现的频次高，那么算法也应为其赋予一个较高的概率估计 $p(A)$，同样地，若口令 B 的结构长度仅为 1，但其出现频次更低，那么攻击算法应为其赋予一个比 $p(A)$ 更低的概率估计。举一个更加直观的例子，以 yahoo 数据集中的 5 个口令为例，它们在数据集中的出现次数和生成概率如表 5 所示，可以看出按照此概率计算方法，口令 1~3 虽然在数据集中出现的次数更多，但由于其口令结构长度均大于口令 4~5，导致其生成概率反而更低。

针对上述问题，本文认为仅使用口令结构概率 $P(struct)$ 足以反映出口令结构分布对整条口令生成概率的影响，在将各子段的概率进行连乘时，则需要按照口令结构长度对其进行标准化处理。

表 5　部分口令在 yahoo 数据集中的出现次数及其生成概率

Table 5 The number of occurrences and their generation probability of partial passwords in the yahoo dataset

序号	口令	出现次数	结构	结构长度	生成概率
1	abc123	250	L_3D_3	2	$P(S=L_3D_3)*P(L_3=abc)*P(D_3=123)=3.01e-5$
2	abcd1234	71	L_4D_4	2	$P(S=L_4D_4)*P(L_4=abcd)*P(D_4=1234)=3.78e-6$
3	qwerty123	51	L_6D_3	2	$P(S=L_6D_3)*P(L_6=qwerty)*P(D_3=123)=1.88e-5$
4	sophie	40	L_6	1	$P(S=L_6)*P(L_6=sophie)=1.02e-4$
5	redsox	32	L_6	1	$P(S=L_6)*P(L_6=redsox)=9.86e-5$

困惑度(Perplexity)是衡量句子好坏的指标，对于句子 $S=w_1w_2w_3\cdots w_n$ 在 uni-gram 语言模型下，其计算规则如式（3），当困惑度越小时，生成句子的概率越大，语言模型越好。

$$Perplexity(S)=\prod_{i=1}^{n}P(w_i)^{-\frac{1}{n}} \tag{3}$$

本文在困惑度计算方法的基础上，对基于 PCFG 的口令概率生成方法进行改进。为了让计算结果直接反映出生成口令的概率大小，在计算时不对概率取倒，同时为了加速计算以及避免概率连乘导致数值过小而造成浮点数向下溢出的问题，将原式通过对数的形式进行转化，计算规则如式（4），n 表示分割出的子段个数，$p(w_i)$ 表示第 i 个子段的概率。

$$P(S)=P(struct)*2^{\frac{1}{n}\sum_{i=1}^{n}\log_2 P(w_i)} \tag{4}$$

4 实验

4.1 实验设置

文章所有实验均在 Windows 10 操作系统下执行，处理器为 Intel(R) Core(TM) i5-10400 CPU @ 2.90GHz 2.90 GHz，程序编码使用 Python3.6.10 及 Tensorflow1.14.0。实验中网络模型的层数为 2，优化器为 adam，训练时学习率初始化为 5×10^{-3}，损失函数为 categorical_crossentropy。

4.2 实验场景及评价指标

从 PCFG 猜测方法的实现思路来看，口令结构的命中率、各长度子段的命中率是其攻击效果的重要影响因素。因此，本节分别设计实验验证所提方法在生成口令结构和子段时相较于其他方法的优势，同时设置了以下四种不同的检测场景：

（1）将 12306 数据和 CSDN 数据组成交叉数据 12306_CSDN，随机选择 80%作训练集，20%作测试集，进行漫步口令攻击，验证本文所提方法在漫步口令猜测中针对中文背景口令的攻击能力；

（2）将 yahoo 数据和 rockyou 数据按场景（1）的规则组合，验证本文所提方法在漫步口令猜测中针对英文背景口令的攻击能力；

（3）与 PR 模型[14]实验数据保持一致，使用 CSDN 数据集，将其中 80%的口令作为训练集，10%的口令作为测试集，并在生成 5×10^7 个口令下，将本文所提方法与 2021 年提出的 PR 模型在同场景下进行对比，验证本文所提模型的先进性。

（4）使用 12306 数据集，随机抽取 80%数据作为训练集，剩下的 20%作为测试集，结合个人信息对用户进行定向口令攻击，验证本文所提方法在定向口令猜测中的攻击性能；

实验中将命中率 P 作为评价指标，设 T 为测试集口令集合，G 为猜测方法所生成的候选口令集合，有：

$$P=\frac{|G\cap T|}{|T|} \tag{5}$$

4.3 实验结果分析

4.3.1 口令结构生成方法性能比较

结合本文划分口令结构的方法，因 12306 数据集包含了用户个人信息，可最大程度上反映出口令结构，本节将使用此数据集对口令结构的命中率进行实验。本节使用传统 PCFG、RNN、LSTM 以及

改进生成对抗网络的 seqGAN[15]模型来对比不同方法在生成口令结构时的命中率,之所以使用 seqGAN 代替传统 GAN 模型进行口令生成，是因为 seqGAN 模型通过引入强化学习,使其相较于传统 GAN 模型而言，更适用于文本生成任务。实验结果如图 4，实验中 4 阶 RNN 及 LSTM 的命中率最高，其他阶数则不做展示。

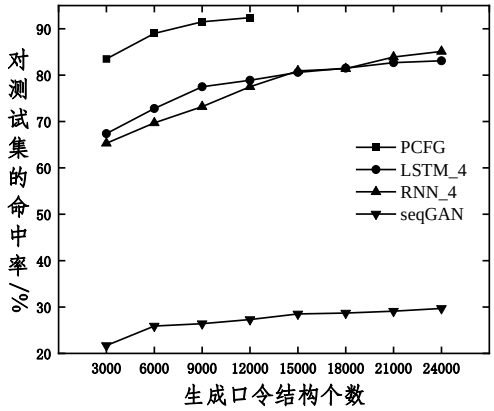

图 4 口令结构生成方法性能比较

Fig.4 Performance of different Password Structure Generation Methods

训练集中不重复的口令结构个数为 11857 个，由实验结果可以看出 PCFG 方法表现优秀，命中率始终高于另外两种方法，对测试集中口令结构的命中率达到了 92.4%，在生成相同个数的口令结构时，LSTM 方法的命中率在 78.9%，RNN 的命中率在 77.5%，而 seqGAN 模型的命中率仅有 27.3%。

考虑到传统 PCFG 方法可生成的口令结构数量有限，本文在实验中继续基于 RNN、LSTM 和 seqGAN 模型进行口令结构生成，当生成 24000 条口令结构时,RNN 模型对测试集的命中率为 85.1%，LSTM 方法稍低，seqGAN 的命中率仍仅有 29.7%，虽然命中率会随着生成的个数的增加而不断提高，但口令生成方法还要考虑生成效率的问题，模型应尽可能在更少的猜测数下，击中更多的口令。所以相较于另外三种方法，PCFG 方法在口令结构的生成中无论是效率还是命中率都更有优势。

从本次实验结果来看，seqGAN 的性能较 RNN 或 LSTM 来说差距较大，而 RNN 和 LSTM 相比，两者的命中率极为接近，但前者的训练效率明显高于后者，因此下述实验中，不再将 seqGAN 和 LSTM 模型参与对比。另外，本次实验发现，口令集中其结构长度大多为 1~4，该实验结果也侧面印证了 PCFG 方法在生成短文本序列时的优势。

4.3.2 模型参数选择及口令子段命中率分析

本节同样在 12306 数据集上设计实验验证 Backoff-RNN 模型是否能够提升对字符长度大于 4 的子段的猜测命中率。

首先，本节对 Backoff-RNN 模型中两个重要参数的选择进行了纵向对比实验：①变阶模型的最高阶数 **s**；②子串的出现频率阈值 (阈值的选择依赖于阈值百分比 **PERC**，将 *n* 元语法频率字典按照频率降序排序，若字典大小为 *size*，则将第 PERC**size* 处的频率作为阈值)，这里仍以数字子段为例，实验结果如表 6 所示。综合来看，当最高阶数为 5 且阈值百分比为 80%时，模型的性能最优。

表 6 模型参数选择

Table 6 Model parameter selection

s-PERC	子段长度					
	5	6	7	8	9	10
4-70%	43.8%	42.1%	21.8%	42.4%	**17.3%**	13.2%
4-80%	46.1%	47.4%	22.7%	42.5%	16.3%	14.6%
4-90%	45.4%	46.3%	20.9%	43.7%	15.9%	14.9%
5-70%	46.7%	49.9%	**26.1%**	40.9%	14.3%	13.7%
5-80%	**47.6%**	**54.3%**	24.3%	**45.7%**	16.4%	**14.8%**
5-90%	43.5%	51.3%	24.5%	45.5%	15.4%	12.1%

基于上述所选参数，本节将 Backoff-RNN 模型与传统 PCFG 方法、RNN_4、Markov_4 进行横向对比，同时为突破传统 PCFG 方法生成个数有限带来的瓶颈，对实验中其余三种方法的生成个数，均在 PCFG 所提供的个数的基础之上增加 20%，实验结果如图 5 所示。

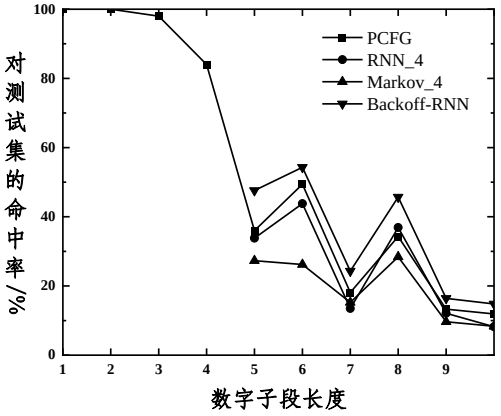

图 5 口令子段生成方法性能比较

Fig.5 Performance of different Password substring Generation Methods

对于长度为 1 至 4 的字符子段，PCFG 方法保持优势，但对于长度大于 4 的字符子段，PCFG 方法普遍性能不足，另外 RNN_4 模型及 Markov_4 模型虽然相较于前者性能稍有提高，但表现不稳定，

而 Backoff-RNN 无论是在命中率还是在稳定性上均优于参与对比的三个模型。另外，图 6 统计了不同长度数字子段在训练数据中出现的次数，数据显示，长度为 6 至 8 的子段在训练集中的占比最高，因此使用 Backoff-RNN 提高长序列口令子段的命中率，对于提高模型整体的攻击效果有重要意义。

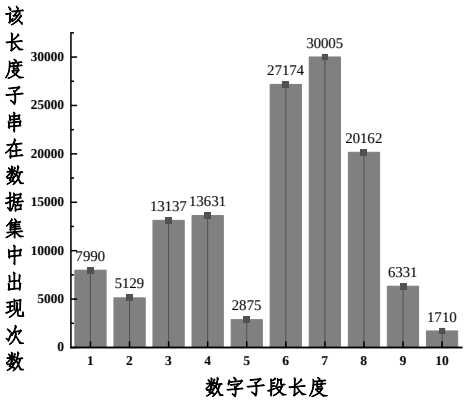

图 6 不同长度数字子段在训练集中的出现次数

Fig.6 Number of occurrences of numeric substrings of different lengths in training set

4.3.3 漫步口令攻击命中率分析

本节设计实验验证所提模型在实验场景 (1) (2) (3) 下的攻击命中率。在前两种场景下，将本文所提方法与传统 PCFG、Markov_4 及 RNN_4 方法的表现性能进行对比。另外，设置消融实验进一步说明本文所提的三个改进点对 PCFG 算法命中率提升的有效性。消融实验中的对比模型及其说明见表 7:

表 7 消融实验中的对比模型

Table 7 Comparative models in ablation experiments

模型名称	模型说明
Model_1	在传统 PCFG 方法上增加改进的口令划分策略
Model_2	在 Model_1 的基础上使用 RNN_4 来生成长度大于 4 的字符子段
Model_3	在 Model_1 的基础上使用 Backoff_RNN 来生成长度大于 4 的字符子段
Model_4	即本文所提模型，在 Model_3 的基础上使用改进的口令概率计算规则

实验结果如图 7 所示，从实验数据来看，Markov 模型的性能相较于其他模型而言综合表现能力较弱，其性能虽然会随着口令生成个数的增加而提升，但仍存在差距。在两种检测场景中，PCFG 方法相较于 Markov-4 模型性能稍有提升，尤其是在生成样本数较少的情况下。另外，RNN_4 模型的综合性能与传统 PCFG 几乎持平，但仍有提升的空间。

Model_1 和 Model_2 相较于传统 PCFG 方法的性能均有所提升，在使用 Model_3 时其综合表现能力相较于前两者更加稳定，且命中率更高，生成 10^7 条口令时，对于中英文两种语言背景的数据集，模型的命中率相较于传统 PCFG 分别提升了 12.8%及 13.6%，相较于 Markov 模型提升了 42.1%及 39.1%，该结果证实 Backoff_RNN 的引入，更加稳定地提升了传统 PCFG 方法的命中率。

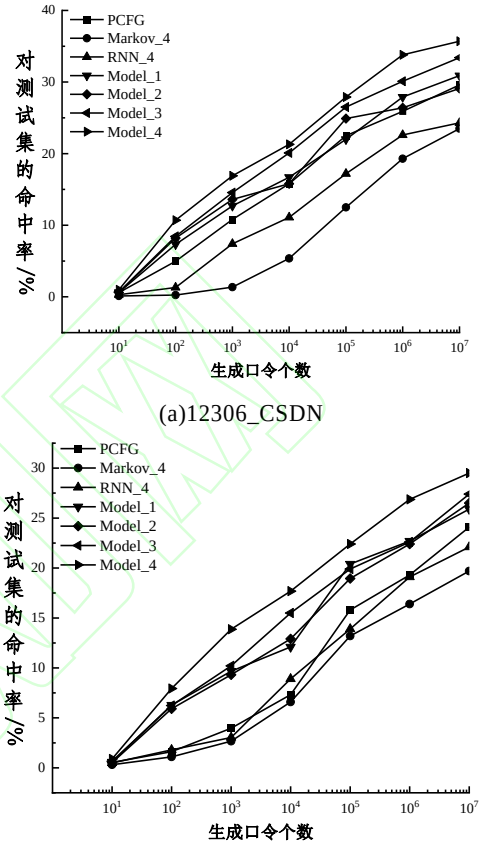

图 7 不同场景下的漫步口令攻击结果

Fig.7 Trawling attack results in different scenarios

Model_4 在初始生成阶段的命中率有明显提升，在生成 10^3 条口令时，对于中英文两种语言背景的数据集，模型的命中率相较于 Model_3 提升了 6.9%及 7.6%，当生成 10^7 条口令时，在中文背景下，模型的命中率为 35.7%，英文背景下命中率为 29.5%，由于在实验中是严格按照口令概率降序的顺序进行猜测，并且随着猜测数量的不断增加，Model_4 的命中率始终高于 Model_3，因此，该结果也可以说明相较于传统概率计算方法而言，改进后的概率计算规则仍能保证概率降序结果，并且对口令分布的刻画能力更优，更能反映出口令在口令集中的真实频次。综合来看，本文所提方法相较于 PCFG 分别提升了 20.6%及 22.4%，相较于 Markov 模型提升了 51.9%及 49.7%。

另外，为了与当前最新研究进展进行对比，本节参照论文[14]中已有数据，在相同实验条件下，当生成 5×10^7 条口令时，其命中率比同场景下的 PR 模型提升了 2.6 个百分点。

综上，相较于传统模型，本文所提模型在口令结构分割、子段生成以及口令生成概率计算上均有一定提升，提高了漫步口令攻击算法的命中率。

4.3.4 定向口令攻击命中率分析

表 8 定向口令攻击结果

Table 8 Targeted attack results

猜测模型	攻击次数	口令破解率
本文模型	10^2	21.4%
TarGuess-Ⅰ	10^2	20.8%
Personal-PCFG	10^2	13.9%

除漫步口令攻击外，本节在场景（4）下验证模型在定向口令攻击中的实验性能，并与 TarGuess-Ⅰ[7]及 Personal-PCFG[6]模型进行对比（实验中所使用的数据集与对比试验一致）。对于定向攻击，本文首先提取用户的个人信息字段，再将其作为原始数据对模型进行训练，同时在生成口令时保留个人信息字段，不对其进行再填充。在生成 100 条口令时，本文所提方法同样提高了在生成小规模数据时展开定向口令攻击的命中率，相较于 Personal-PCFG 提升了 53.9%，相较于 TarGuess-Ⅰ提升了 2.8%。

5 结束语

本文设计了一种基于 Backoff-RNN 及概率平衡的 PCFG 口令猜测方法，该方法对口令进行更细粒度的结构划分，提取出用户口令更深层次的结构信息；使用 Backoff-RNN 模型生成长度大于 4 的字符子段，动态平衡模型拟合问题；对传统方法中生成口令概率的计算方法进行改进，将困惑度融入到口令概率的计算规则中，使得口令概率更能体现出口令在口令集中的真实分布规律。实验结果表明，该方法在漫步攻击和定向攻击中对测试集均有更高的命中率。下一阶段，作者将继续对口令的生成算法进行优化，提高口令生成算法及概率计算的效率。同时在此猜测方法的基础上构建出口令强度评估机制，从而更好地指导用户创建安全可靠的口令。

参考文献

[1] Weir M, Aggarwal S, De Medeiros B, et al. Password cracking using probabilistic context-free grammars[C]// 2009 30th IEEE Symposium on Security and Privacy. IEEE, 2009: 391-405.

[2] Narayanan A, Shmatikov V. Fast Dictionary Attacks on Passwords Using Time-Space Tradeoff[C]//Proceedings of the 12th ACM Conference on Computer and Communications Security, CCS 2005, Alexandria, VA, USA, November 7-11, 2005. ACM, 2005.

[3] Katz S. Estimation of probabilities from sparse data for the language model component of a speech recognizer[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1987, 35(3):400-401.

[4] Veras R, Collins C, Thorpe J. On the Semantic Patterns of Passwords and their Security Impact[C]//Network & Distributed System Security Symposium. 2014.

[5] S. Houshmand, S. Aggarwal and R. Flood, "Next Gen PCFG Password Cracking," in IEEE Transactions on Information Forensics and Security, vol. 10, no. 8, pp. 1776-1791, Aug. 2015, DOI: 10.1109/TIFS.2015.2428671.

[6] Li Y, Wang H, Sun K. A study of personal information in human-chosen passwords and its security implications.[C]//IEEE INFOCOM 2016 - IEEE Conference on Computer Communications. IEEE, 2016.

[7] Ding W, Zhang Z, Ping W, et al. Targeted Online Password Guessing: An Underestimated Threat[C]// ACM CCS 2016. ACM, 2016.

[8] 章梦礼，张启慧，刘文芬,等. 一种基于结构划分及字符串重组的口令攻击方法[J]. 计算机学报, 2019, 42(4):16.
ZHANG M L, ZHANG Q H, LIU W F, et al. A Method of Password Attack Based on Structure Partition and String Reorganization[J]. Chinese Journal of Computers, 2019, 42(4):16.

[9] Hitaj B, Gasti P, Ateniese G, et al. Passgan: A deep learning approach for password guessing[C]// International Conference on Applied Cryptography and Network Security. Springer, Cham, 2019: 217-237.

[10] Melicher W, Ur B, Segreti S M, et al. Fast, Lean, and Accurate: Modeling Password Guessability Using Neural Networks[C]// ACADEMY PUBLISHER. ACADEMY PUBLISHER, 2013.

[11] 周环，刘奇旭，崔翔，等.基于神经网络的定向口令猜测研究[J].信息安全学报, 2018, 3(5):13.
ZHOU H, LIU Q X, CUI X, et al. Research on targeted password guessing using neural networks. Journal of Cyber Security, 2018, 3(5):13.

[12] Xu L, Ge C, Qiu W, et al. Password Guessing Based on LSTM Recurrent Neural Networks[C]//2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC). IEEE, 2017.

[13] Liu Y, Xia Z, Yi P, et al. GENPass: A General Deep Learning Model for Password Guessing with PCFG Rules and Adversarial Generation[C]// 2018:1-6.

[14] 汪定,邹云开,陶义,等. 基于循环神经网络和生成式对抗网络的口令猜测模型研究[J]. 计算机学报, 2021, 44(8):16.
WANG D, ZOU Y K, TAO Y, et al. Password Guessing Based on Recurrent Neural Networks and Generative Adversarial Networks. Chinese Journal of Computers, 2021, 44(8):16.

[15] Yu L, Zhang W, Wang J, et al. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient[J]. 2016.

[16] Ma J, Yang W, Luo M, et al. A study of probabilistic password models[C]//2014 IEEE Symposium on Security and Privacy. IEEE, 2014: 689-704.

[17] Zhiyang, Xia, Ping, et al. GENPass: A Multi-Source Deep Learning Model for Password Guessing[J]. IEEE Transactions on Multimedia, 2019, 22(5):1323-1332.

[18] Zhang Y, Xian H, Yu A. CSNN: Password guessing method based on Chinese syllables and neural network[J]. Peer-to-Peer Networking and Applications, 2020(4).

[19] Hranický R, Lištiak F, Mikuš D, et al. On practical aspects of PCFG password cracking[C]//IFIP Annual Conference on Data and Applications Security and Privacy. Springer, Cham, 2019: 43- 60.

[20] Nam S, Jeon S, Kim H, et al. Recurrent GANs Password Cracker For IoT Password Security Enhancement[J]. Sensors, 2020, 20(11):3106.

[21] Bonneau J, Herley C, Van Oorschot P C, et al. Passwords and the Evolution of Imperfect Authentication[J]. Communications of the Acm, 2015, 58(7):78-87.

[22] Wang P, Wang D, Huang X. Advances in password security[J]. Computer Research and Development, 2016, 53(10): 2173-2188.

[23] 安亚巍, 罗顺, 朱智慧. 基于马尔可夫链的口令破解算法[J]. 计算机工程, 2018, 44(11):119-122.
AN Yawei, LUO Shun, ZHU Zhihui. Password cracking algorithm based on Markov chain[J]. Computer Engineering, 2018, 44(11):119-122.

[24] 罗敏, 张阳. 一种基于姓名首字母简写结构的口令破解方法[J]. 计算机工程, 2017, 43(1):188-195,200.
LUO Min, Zhang Yang. A password Cracking Method Based on Name Initials Shorthand Structure[J].Computer Engineering, 2017, 43(1):188-195,200.

[25] Pasquini D, Gangwal A, Ateniese G, et al. Improving password guessing via representation learning// Proceedings of the 42nd IEEE Symposium on Security and Privacy. Oakland, USA, 2021:265-282