Análise de Senhas Numéricas: Pesquisando Padrões de Comportamento

Wesley Rodrigues
wesley.it@gmail.com
http://lattes.cnpq.br/8077750057671460
Instituto de Pesquisas Tecnológicas do Estado de São Paulo
São Paulo, SP, Brasil

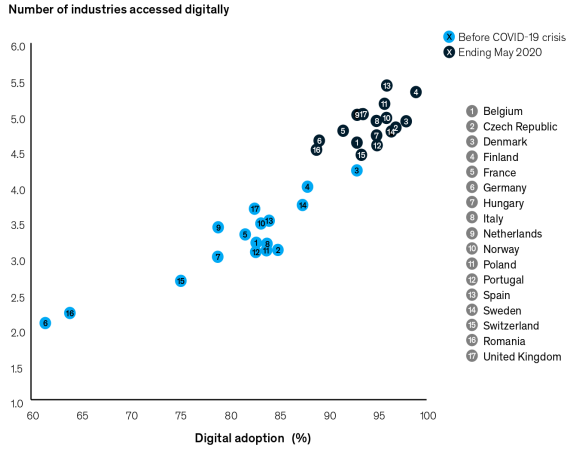
RESUMO

Senhas ainda são o mecanismo mais utilizado para identificar pessoas em sistemas on-line. Este artigo demonstra que, em alguns cenários, por mais que a complexidade matemática de adivinhar uma senha possa ser alta, a análise estatística de conjuntos de dados contendo senhas vazadas pode revelar padrões comportamentais dos usuários que influenciam na escolha da senha, diminuindo assim a complexidade da busca aumentando as probabilidades de adivinhar a senha alvo. Este trabalho utilizou como conjunto de dados a base COMB, contendo aproximadamente 100GB de senhas reais. As senhas numéricas de quatro à oito caracteres de comprimento foram selecionadas e analisadas estatisticamente, principalmente em relação à distribuição de cada um dos 10 algarismos nas posições dos caracteres da senha. A análise mostrou que para as senhas com 6 e 8 caracteres, cerca de 75% dos usuários utilizam uma data no formado DDMMYY ou DDMMYYYY. Finalmente, o artigo propõe formas de expandir esta análise e de mitigar estes problemas por meio da adoção de MFA ou passwordless.

PALAVRAS-CHAVE

Senhas, Padrões, Força-bruta, Análise

Figura 1: A pandemia reduziu a disparidade digital entre os países europeus.



Fonte: McKinsey (2020)

1 INTRODUÇÃO

A sociedade contemporânea tem ampliado sua dependência de recursos digitais para realizar atividades cotidianas. Setores como serviços bancários, cartorários, financeiros, de entretenimento, trabalho e estudo emergiram como categorias de negócios que experimentaram crescimento online ao longo da última década. Esse fenômeno foi especialmente impulsionado após a pandemia do COVID-19, levando as empresas a acelerarem sua transição para sistemas digitais e de computação em nuvem. Uma investigação conduzida pela consultoria Mckinsey [5], abrangeu uma amostra representativa de 20.000 consumidores distribuídos em 17 países europeus. Os resultados indicam uma transformação significativa nos hábitos digitais, com um aumento médio de 13% na adoção de soluções digitais no primeiro semestre de 2020. Esse aumento expressivo destaca a rápida adaptação da sociedade às demandas impostas pelo cenário global, consolidando a influência crescente das plataformas digitais em suas interações cotidianas.

A autenticação de indivíduos em sistemas digitais normalmente se realiza por meio de um identificador único associado a uma senha. Senhas de natureza simplificada, quando empregadas, podem facilitar que atores maliciosos executem ataques, possibilitando a manipulação fraudulenta dos sistemas ao se fazerem passar pela vítima. Tais incidentes podem resultar em danos financeiros, comprometimento de reputação, violação de confidencialidade, entre outros impactos adversos. No contexto brasileiro, no ano de 2023, foi conduzido o julgamento do hacker Walter Delgatti[1], responsável por invadir e modificar sistemas do Conselho Nacional de Justiça. Durante o processo judicial, o hacker admitiu ter explorado fragilidades nas senhas para obter acesso aos sistemas. Credenciais dotadas de privilégios administrativos apresentavam senhas como **123mudar**, **cnj123**, **p123456**, desprovidas de um segundo fator de autenticação.

Diversos métodos de aquisição ilícita de credenciais estão em vigor, indo desde a manipulação do usuário até a espionagem das comunicações. Estratégias como phishing, engenharia social e impersonificação são conhecidas por ludibriar os usuários, levando-os a fornecer suas credenciais aos atacantes. Em outro cenário, as comunicações do usuário podem ser monitoradas clandestinamente, e senhas obtidas sem o conhecimento do titular. Isso pode ocorrer por meio da instalação de aplicativos maliciosos ou comprometimento da rede, permitindo ao atacante espiar pacotes não criptografados. O enfoque deste artigo aborda uma terceira modalidade, onde a obtenção de senhas ocorre por força bruta, listas conhecidas ou análise de probabilidades.

O processo de força bruta consiste em percorrer todas as possibilidades dentro de um espaço finito de caracteres. O processo pode

[1]CNN Brasil, Disponível em: https://bit.ly/cnn-brasil-hacker-2023, acesso em: 08/12/2023

ser uma atividade bastante demorada, dependendo do tempo que cada tentativa exige, da quantidade de caracteres e do conjunto de caracteres possíveis para cada dígito da senha. Dado que TQ é o tempo total para percorrer todo o espaço de senhas possíveis no caso pessimista, t é o tempo levado para uma única tentativa, n é a quantidade de caracteres que podem ser utilizados, e k é o comprimento da senha em caracteres, temos duas fórmulas distintas para encontrar TQ, a depender das políticas. Caso a senha exija caracteres diferentes em todas as posições (como 1234 ou 8246, por exemplo):

$$TQ(n, k, t) = t \cdot \frac{n!}{k! \cdot (n-k)!}$$

Há sistemas que aplicam políticas de complexidade de senhas, impedindo que caracteres sejam utilizados mais de uma vez. Entretanto, há casos onde tais políticas não são aplicadas, tornando o espaço de busca maior, mas tornando as senhas mais previsíveis. Para senhas que podem ter valores repetidos (como 2222, por exemplo), o cálculo é denotado por:

$$TQ(n, k, t) = t \cdot n^k$$

Por exemplo, uma senha totalmente numérica (contendo apenas caracteres de 0 a 9) com tamanho de 4 dígitos (de 0000 a 9999) que demore 300 milissegundos por tentativa, demoraria 50 minutos (3.000 segundos ou 3.000.000 milissegundos) para ser encontrada via métodos de força bruta:

$$TQ(10, 4, 300) = 300 \cdot 10^4 = 3.000.000ms$$

Para este caso, definindo que CD é o conjunto que contém os caracteres disponíveis para serem utilizados, e x_n é o caractere na enésima posição da senha, temos que para cada x, o tamanho de $CD = 10$. Aumentando o tamanho de CD, a complexidade aumenta exponencialmente. Façamos um exemplo pensando em uma senha com os mesmos parâmetros da anterior, ou seja, uma senha com o tamanho de 4 caracteres, que demore 300 milissegundos por tentativa, mas, desta vez, x_n pode ser repetido, e pode pertencer aos conjuntos $NUM = \{0, 1, 2, ..., 9\}$, $MIN = \{a, b, c, ..., z\}$ e $MAI = \{A, B, C, ..., Z\}$. Agora, o espaço de busca de x_n aumentou de 10 para 62 (10 + 26 + 26). O impacto na performance de força bruta é severo:

$$TQ(62, 4, 300) = 300 \cdot 62^4 = 4.432.900.800ms$$

O tempo aumentou de 50 minutos para mais de 73 mil minutos, que equivalem a mais de 1200 horas ou mais de 50 dias. Isto demonstra que a execução de ataques de força bruta, dependendo do espaço de busca, do tamanho da senha e do tempo necessário para cada tentativa pode tornar o método inviável. Uma senha com 8 caracteres, seguindo os mesmos parâmetros do cálculo anterior, por exemplo:

$$TQ(62, 8, 300) = 300 \cdot 62^8 = 65.502.031.675.468.800ms$$

Para percorrer todo o espaço de busca dos conjuntos e encontrar x_1 até x_8 seriam necessários 758.125.366 dias, que correspondem a 2.077.055 anos, ou mais de 20.000 séculos. Para muitos sistemas o tempo de verificação é maior que 300ms, principalmente os que dependem de redes.

A alternativa muitas vezes empregada é utilizar algum tipo de heurística ao processo de adivinhação da senha, aplicando regras baseadas em estudos estatísticos de probabilidade e distribuição, de características sociais, demográficas, ou mesmo pessoais, incluindo conhecimento sobre as preferências do indivíduo. Para tal, é necessário estudar senhas existentes. Neste contexto, entram os vazamentos de senhas, conhecidos como *password leaks* pela mídia. De 2009 até o momento, bilhões de usuários, e-mails e senhas de sistemas por todo o mundo foram vazados publicamente ou adicionados à listas de senhas e vendidas na DarkWeb. Os casos que ganharam mais notoriedade foram o da RockYou, em 2009, com mais de 30 milhões de credenciais vazadas, o da CSDN em 2011, com mais 6 milhões, o da Yahoo, com 3 bilhões de credenciais em 2017, o LinkedIn, com diversos vazamentos, sendo o último em 2021 com bilhões de credenciais, totalizando 92% de sua base, entre muitos outros [10].

As credenciais vazadas são frequentemente utilizadas por atacantes em tentativas de roubo de identidade digital, dado que muitas pessoas não sabem que suas credenciais foram expostas, e tem o costume de utilizar a mesma senha para diversos serviços online. As credenciais também são utilizadas por alguns times de segurança cibernética para verificar se os usuários de seus sistemas ainda utilizam senhas comprometidas, fazendo com que os usuários troquem suas senhas por opções ainda desconhecidas. As listas compiladas com senhas vazadas mais conhecidas são a RockYou e a COMB, acrônimo para *Compilation (ou Combination) Of Many Breaches* [2].

A lista COMB é a maior base concentrada de dados de logins que se tem notícia até o momento, com 3.2 bilhões de credenciais e senhas, incluindo mais de 200 milhões de credenciais do GMail e mais de 450 milhões de senhas do Yahoo. Estima-se que existam 4.7 bilhões de pessoas online, deste modo, a COMB reúne o equivalente a credenciais de cerca de 70% da população de toda a Internet [2]. O conjunto de senhas COMB possui mais de 99GB, tornando o download bastante difícil diretamente pelo navegador, sendo que a forma mais utilizada para download é por link magnético ou Torrent P2P: https://downloadtorrentfile.com/hash/af2879db0fab2a32ba38d0491aa8fea5e29d3678?name=CompilationOfManyBreaches.7z.

A área de Segurança Digital da IBM publica anualmente um relatório técnico[12] com detalhes sobre os custos das vulnerabilidades de segurança para as empresas e governos. Segundo o relatório, 82% das vulnerabilidades envolveram ambientes expostos na nuvem ou na Internet com abuso de credenciais, e o custo médio global de uma vulnerabilidade de dados é de 4.45 milhões de dólares. As indústrias mais afetadas são as da saúde, de serviços financeiros e de tecnologia.

A existência de conjuntos de dados reais permite uma grande variedade de pesquisas visando melhorar a segurança dos sistemas. Existem abordagens que analisam as senhas do ponto de vista comportamental [7], demonstrando que aspectos locais como o idioma e a cultura de cada país interferem na escolha das senhas.

Outros estudos analisam grandes conjuntos de textos chamados corpus em busca de probabilidades de encontrarem senhas pertencentes ao vocabulário, com apoio de suavização de Laplace e segmentação PCFG [6].

Alguns pesquisadores analisaram a relação das senhas com a religião, gênero e estado civil [9].

Enquanto alguns artigos optam por análises estatísticas, outros seguem com machine learning, empregando técnicas como o uso de processamento de linguagem natural para entender a complexidade das senhas chinesas [11].

Há estudos com métodos que ainda estão se tornando populares, como as gramáticas livres de contexto probabilísticas (GLPCs), utilizadas em conjunto com PCFG [10].

Ferramentas como o John The Ripper[2] e o HashCat[3] oferecem arcabouços para facilitar o processo de conversão de palavras para hashes, de modo a comparar e descobrir a senha a partir de uma lista de dados incompreensíveis. Pesquisadores utilizam técnicas de machine learning para acelerar ainda mais este processo, mas alguns artigos mostram que é possível obter um alto índice de acerto de senhas com análise e combinação de regras de máscaras de senhas [3].

Pesquisas do tipo survey e revisões de literatura [4] também foram encontradas, propondo técnicas estatísticas para calcular a probabilidade de um usuário ser atacado em decorrência de seu comportamento, como o hábito de trocar as senhas ou de usar senhas únicas para cada sistema online.

Este artigo tem como objetivo principal analisar a lista de senhas, delimitando como escopo **apenas as senhas numéricas de 4 a 8 dígitos de tamanho**, e verificar os padrões de comportamento encontrados nesse recorte. A base inteira é grande, e enquanto um computador ou notebook médio em 2023 possui cerca de 8 a 16GB de memória RAM, o dataset completo possui quase 100GB, tornando operações em memória mais trabalhosas, exigindo processamento em lotes.

```
$ du -sh CompilationOfManyBreaches*
99G     CompilationOfManyBreaches
19G     CompilationOfManyBreaches.7z
```

O recorte das senhas numéricas é consideravelmente menor, com cerca de 3GB ou 417.524.472 linhas, o que é razoável para estudo em um computador pessoal moderno.

2 METODOLOGIA

O artigo apresenta um **Estudo Experimental com Dados**. O trabalho foi executado em duas grandes etapas: carga e análise.

2.1 Etapa de Carga

Por mais que um conjunto de dados com 3GB caiba perfeitamente na memória de um notebook moderno, com 32GB de RAM, as operações de filtro, busca, análise e separação começam a apresentar lentidão quando há muitos dados abertos. Desta forma, o processo adotado foi o de utilizar *checkpoints* depois de operações importantes. Um arquivo IPYNB (notebook do iPython) foi criado exclusivamente para as operações de carga, com os seguintes objetivos:

(1) Usando o comando GREP com Expressões Regulares, extrair todas as linhas com senhas numéricas para um arquivo em texto;
(2) Com a biblioteca Pandas, importar o arquivo para um objeto do tipo DataFrame;
(3) Utilizando a biblioteca PickleShare, persistir os dados do DataFrame em um checkpoint;
(4) Sumarizar o DataFrame de senhas criando novas colunas para a quantidade de ocorrências, quantidade de caracteres, quantidade de caracteres únicos e quantidade de ocorrências proporcionais ao total da base;
(5) Salvar o checkpoint do DataFrame sumarizado.

Após este processo, o DataFrame sumarizado diminuiu para 980MB, menos de 33% do tamanho inicial sem perda de informações relevantes para a análise.

Figura 2: O DataFrame sumarizado de senhas numéricas.

TOP 30 senhas numéricas de 4 a 8 dígitos da base analisada:

[6]:

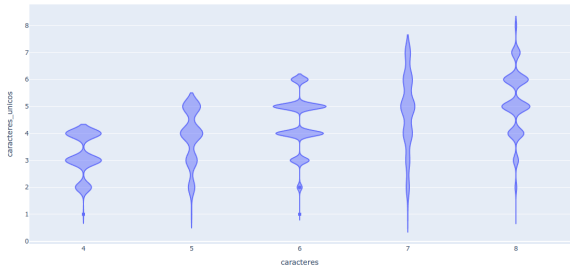
	senha	ocorrencias	caracteres	caracteres_unicos	ocorrencias_percentuais
0	123456	20963035	6	6	0.050208
1	12345	7030768	5	5	0.016839
2	12345678	3521408	8	8	0.008434
3	111111	2906365	6	1	0.006961
4	1234567	2363539	7	7	0.005661
5	123123	2332470	6	3	0.005586
6	000000	1935218	6	1	0.004635
7	123321	1447191	6	3	0.003466
8	654321	1249387	6	6	0.002992
9	666666	1231014	6	1	0.002948
10	1234	1230424	4	4	0.002947
11	7777777	1055009	7	1	0.002527

Fonte: O Autor, 2023

2.2 Etapa de Análise

O DataFrame persistido em formato PickleShare foi carregado no segundo notebook. A distribuição das senhas mostrou uma cauda longa. As 50.000 senhas mais utilizadas foram selecionadas para a análise. Algumas destas senhas, como **123456** ocorreram mais de 20 milhões de vezes. O gráfico de violino mostra o comportamento das senhas de acordo com o comprimento (de 4 a 8 caracteres) versus a quantidade de caracteres únicos:

Figura 3: Gráfico de violino de acordo com o tamanho da senha e os caracteres únicos.



Fonte: O Autor, 2023

As senhas foram separadas em objetos DataFrame diferentes de acordo com o comprimento em caracteres. Um algoritmo foi

[2] https://www.openwall.com/john/
[3] https://hashcat.net/hashcat/

desenvolvido para gerar os gráficos de distribuição de acordo com a posição. Em uma senha que contenha seis caracteres, por exemplo, que variam de 0 a 9, e que não sofre nenhum tipo de influência na escolha dos caracteres, o esperado é que a distribuição seja muito parecida para todos os caracteres em cada uma das posições [1]. Mas, se o critério de escolha não for aleatório, a distribuição terá números com maior frequência em dadas posições em relação a outros. As distribuições foram analisadas e comparadas com padrões conhecidos por meio de expressões regulares para validar comportamentos percebidos nas escolhas das senhas de acordo com o seu comprimento.

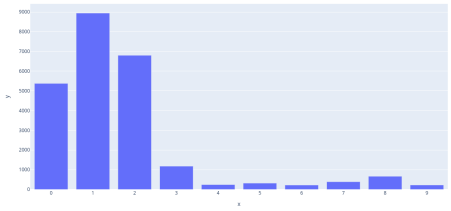
3 RESULTADOS

As senhas foram separadas em grupos de 4, 5, 6, 7 e 8 caracteres. Dentre os grupos, os que apresentaram as maiores discrepâncias na distribuição, indicando que a escolha dos dígitos não foi aleatória, mas seguiu algum tipo de padrão, foram os grupos de 6 e 8 caracteres de comprimento. Os resultados do grupo de 8 caracteres foram similares ao grupo de 6 caracteres, que será demonstrado a seguir.

3.1 Senhas com 6 Caracteres de Comprimento

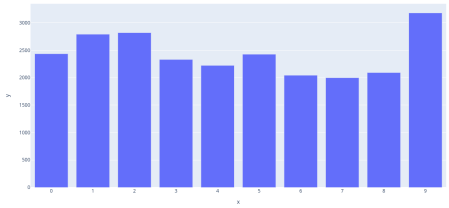
As senhas deste grupo demonstraram um padrão que pode ser percebido claramente nos gráficos de distribuição dos números de acordo com a posição. Para fins de interpretação, fica convencionado no artigo que a posição da senha começa da esquerda para a direita, iniciando em 1.

Figura 4: Distribuição dos números na primeira posição da senha.



Fonte: O Autor, 2023

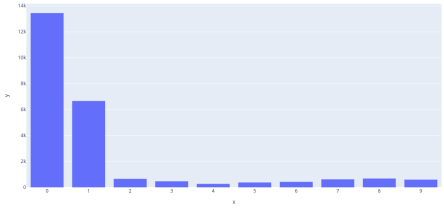
Figura 5: Distribuição dos números na segunda posição da senha.



Fonte: O Autor, 2023

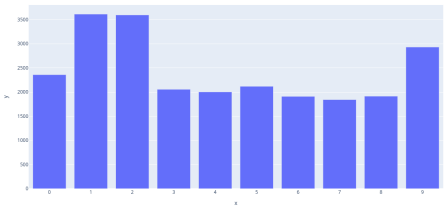
Os gráficos mostram que para cada posição, há um grupo de caracteres que aparece majoritariamente.

Figura 6: Distribuição dos números na terceira posição da senha.



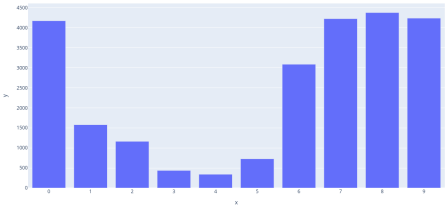
Fonte: O Autor, 2023

Figura 7: Distribuição dos números na quarta posição da senha.



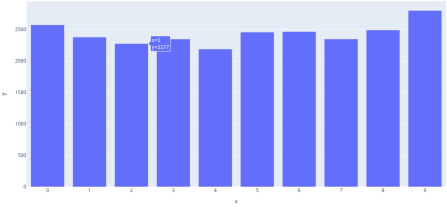
Fonte: O Autor, 2023

Figura 8: Distribuição dos números na quinta posição da senha.



Fonte: O Autor, 2023

Figura 9: Distribuição dos números na sexta posição da senha.



Fonte: O Autor, 2023

- Posição 1: caracteres 0, 1, 2 (mais expressivos), e 3 (menos expressivo);

- Posição 2: caracteres 1, 2 e 9 (mais expressivos) e distribuição aleatória nos outros;
- Posição 3: caracteres 0 (mais expressivo) e 1 (menos expressivo);
- Posição 4: caracteres 1, 2 e 9 (mais expressivos) e distribuição aleatória nos outros;
- Posição 5: caracteres 6, 7, 8, 9 e 0;
- Posição 6: caracteres distribuição aleatória.

O padrão percebido tem correlação com o padrão utilizado em datas, principalmente no formato $DDMMYY$, ou seja, dia com dois dígitos, mês com dois dígitos e ano com dois dígitos. Para confirmar a hipótese, foi executado um teste utilizando todas as senhas com 6 caracteres de comprimento, aplicando uma expressão regular (REGEX) para encontrar datas. Das 24.345 senhas, 18.026 corresponderam ao padrão de data, o equivalente a aproximadamente 74% da base.

4 DISCUSSÕES FINAIS

As implicações deste tipo de heurística são perceptíveis diretamente na redução da complexidade computacional (e no tempo de busca, consequentemente) para efetuar o ataque de força bruta. No caso das senhas de 6 dígitos de comprimento, supondo uma senha completamente aleatória, seguindo o exemplo anterior com 300ms para cada tentativa, teríamos:

$$TQ(10, 6, 300) = 300 \cdot 10^6 = 300.000.000ms$$

Isso equivale a aproximadamente **83 horas** de processamento. Entretanto, aplicando as heurísticas de data, temos que calcular os valores separadamente, já que cada posição pode ter seu conjunto específico de caracteres possíveis:

$$P_1 \in \{0, 1, 2\} \rightarrow L_1 = 3$$
$$P_2 \in \{0, 1, 2, ..., 9\} \rightarrow L_2 = 10$$
$$P_3 \in \{0, 1\} \rightarrow L_3 = 2$$
$$P_4 \in \{0, 1, 2, ..., 9\} \rightarrow L_4 = 10$$
$$P_5 \in \{0, 6, 7, 8, 9\} \rightarrow L_5 = 5$$
$$P_6 \in \{0, 1, 2, ..., 9\} \rightarrow L_6 = 10$$
$$300 * \prod_{i=1}^{6} L_i = 9.000.000$$

Este valor equivale a **menos de 3 horas** de processamento.

4.1 Limitações e Trabalhos Futuros

Este trabalho foi executado levando em considerações aspectos básicos de estatística e análise de dados. A base utilizada conta com dados de diversas regiões, de diversos sistemas, mas pode conter vieses, como por exemplo a política de segurança utilizada pelos sites onde as senhas estavam cadastradas. Isso não afeta a relevância da pesquisa, pois os dados são reais e viu-se que para o tipo de senhas específicamente numéricas com 6 caracteres, é possível obter um índice de sucesso próximo de 74%, diminuindo em dezenas de vezes o tempo necessário para efetuar a força bruta.

Trabalhos futuros podem explorar outros tipos de padrões, como números de telefone, CEP e datas em outros formatos, por exemplo. Como demonstrado em trabalhos das referências [7] [9], fatores culturais implicam diretamente na escolha da senha, e quanto mais informações o atacante possuir sobre a vítima, maior a probabilidade de utilizar heurísticas para melhorar a performance da força bruta.

O National Institute of Standards and Technology (NIST) possui recomendações sobre a adoção segura de senhas, com práticas sobre como criar e gerenciar a segurança das identidades digitais [8]. Abordagens modernas, como a aplicação de biometria e múltiplos fatores de autenticação (MFA) também são técnicas para mitigar as vulnerabilidades das senhas. Organizações como a Microsoft estão promovendo um movimento chamado **passwordless** [13], com a intenção de utilizar dispositivos dotados de capacidades criptográficas como o TPM, para deixar de utilizar as senhas, dificultando o roubo de credenciais.

REFERÊNCIAS

[1] Peter Bruce and Andrew Bruce. 2019. *Estatística Prática para Cientistas de Dados.* Alta Books, Rio de Janeiro, RJ.

[2] Ron Cresswell. 2023. What you should know about the COMB data leak. https://www.acfeinsights.com/acfe-insights/what-you-should-know-about-comb-data-leak

[3] Alessia Michela Di Campi, Riccardo Focardi, and Flaminia L. Luccio. 2022. The Revenge of Password Crackers: Automated Training of Password Cracking Tools. In *Computer Security - ESORICS 2022 (Lecture Notes in Computer Science)*, Vijayalakshmi Atluri, Roberto Di Pietro, Christian D. Jensen, and Weizhi Meng (Eds.). Springer Nature Switzerland, Cham, 317–336. https://doi.org/10.1007/978-3-031-17146-8_16

[4] Ojonukpe S. Egwuche, Mutiu Ganiyu, and Abiona A. Akeem. 2022. Assessing the Vulnerabilities of Internet Users to Cyber-Attacks using their Password Login Patterns. In *2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON).* IEEE, Lagos, Nigeria, 1–4. https://doi.org/10.1109/NIGERCON54645.2022.9803155

[5] Santiago Fernandez, Benjamin Vieira, and Paul Jenkins. 2020. *Europe's digital migration during COVID-19: Getting past the broad trends and averages.* Technical Report. Mckinsey, Madrid, ES. 10 pages. https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/europes-digital-migration-during-covid-19-getting-past-the-broad-trends-and-averages

[6] Xiaochun Gan, Meng Chen, Dong Li, Zongyan Wu, Weili Han, and Hu Chen. 2021. Corpora-based Password Guessing: An Efficient Approach for Small Training Sets. In *2021 IEEE 4th International Conference on Electronics and Communication Engineering (ICECE).* IEEE, Xi'an, China, 311–319. https://doi.org/10.1109/ICECE54449.2021.9674437

[7] Xiaochun Gan, Dong Li, and Hu Chen. 2022. Analysis of words in passwords from three different countries. In *2022 IEEE 10th Joint International Information Technology and Artificial Intelligence Conference (ITAIC).* IEEE, Chongqing, China, 1775–1781. https://doi.org/10.1109/ITAIC54216.2022.9836812

[8] Paul A Grassi, James L Fenton, Elaine M Newton, Ray A Perlner, Andrew R Regenscheid, William E Burr, Justin P Richer, Naomi B Lefkovitz, Jamie M Danker, Yee-Yin Choong, Kristen K Greene, and Mary F Theofanos. 2017. *Digital identity guidelines: authentication and lifecycle management.* Number NIST SP 800-63b. Gaithersburg, MD. NIST SP 800–63b pages. https://doi.org/10.6028/NIST.SP.800-63b

[9] Daojing He, Hang Yu, Beibei Zhou, Shanshan Zhu, Min Zhang, Sammy Chan, and Mohsen Guizani. 2021. How Does Social Behavior Affect Your Password? *IEEE Network* 35, 5 (Sept. 2021), 284–289. https://doi.org/10.1109/MNET.101.2000762

[10] Xuejing Jiang, Xun Sun, and Qiuming Liu. 2022. Password Guessing Attack Based on Probabilistic Context Free Algorithm. In *2022 IEEE 8th International Conference on Computer and Communications (ICCC).* IEEE, Chengdu, China, 1234–1238. https://doi.org/10.1109/ICCC56324.2022.10065766

[11] Rui Xu, Xiaojun Chen, Yingbing Wang, Jinqiao Shi, and Haitao Mi. 2016. Memory chunking analysis of numerical password for Chinese websites. In *MILCOM 2016 - 2016 IEEE Military Communications Conference.* IEEE, Baltimore, MD, 788–793. https://doi.org/10.1109/MILCOM.2016.7795425

[12] IBM Security. 2023. Relatório o Custo das violações de dados. https://www.ibm.com/br-pt/reports/data-breach

[13] Microsoft Security. 2023. Autenticação sem senha. https://www.microsoft.com/pt-br/security/business/solutions/passwordless-authentication