Evaluating Password Composition Policy and Password Meters of Popular Websites

Kyungchan Lim*, Joshua H. Kang*, Matthew Dixson*, Hyungjoon Koo†, and Doowon Kim*
*_Department of Electrical Engineering and Computer Science, University of Tennessee_, Knoxville, TN, USA
†_Department of Computer Science and Engineering, Sungkyunkwan University_, Suwon, South Korea
*{klim7, jkang17, mdixson2, doowon}@utk.edu † kevin.koo@skku.edu

2023 IEEE Security and Privacy Workshops (SPW) | 979-8-3503-1236-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/SPW59333.2023.00006

***Abstract*—Password-based authentication is one of the most commonly adopted mechanisms for online security. Choosing strong passwords is crucial for protecting ones' digital identities and assets, as weak passwords can be readily guessable, resulting in a compromise such as unauthorized access. To promote the use of strong passwords on the Web, the National Institute of Standards and Technology (NIST) provides website administrators with password composition policy (PCP) guidelines. We manually inspect popular websites to check if their password policies conform to NIST's PCP guidelines by generating passwords that meet each criterion and testing the 100 popular websites. Our findings reveal that a considerable number of websites (on average, 53.5%) do not comply with the guidelines, which could result in password breaches.**

I. Introduction

Authentication is a fundamental aspect of information security that involves verifying the digital identity of an individual or entity seeking access to a system, application, or data. The importance of authentication has become more pronounced with the increasing use of digital technologies and the growing amount of personal and confidential data stored and transmitted online. Of the varying authentication mechanisms, the password-based authentication mechanism has been the most widely adopted in the wild because it is cost-effective for website administrators and does not require extra hardware for end-users.

However, unfortunately, passwords can be easily compromised, leading to breaches of digital identities, especially if end-users use easily-guessable passwords such as previously-breached passwords, dictionary-words-contained passwords, etc. Hence, using strong passwords is essential for end-users to protect their digital identities. To assist end-users in generating strong passwords, the National Institute of Standards and Technology (NIST) provides password composition policy (PCP) guidelines, particularly for the Web ecosystem, for website administrators [1]. The current version of the NIST PCP guidelines (800-63B) specifies technical requirements with five criteria (breached passwords, dictionary words, repetitive characters, sequential characters, and context-specific words) and one recommendation (password-strength meters) for website administrators. Specifically, when end-users sign up for websites, the websites should be able to reject passwords that belong to the five criteria. For example, a website should reject a password "`passw0rd`" from an end-user as it is found in breached password lists.

By enforcing these guidelines, websites can encourage end-users to create stronger passwords and reduce the burden of password management. However, it is still ongoing research and investigation to determine the effectiveness and adoption of these guidelines by both users and website operators. While prior studies focus on improving password strength by recommending stronger password composition policies (PCP) and strategies [2]–[7], they do not examine the current state of websites at an individual level. In this paper, we aim to better understand the current Web ecosystem regarding the proper adoption of the NIST PCP guidelines on popular websites.

To this end, we conduct a measurement study where we manually examine popular websites using weak passwords. Specifically, we choose the top 100 popular websites from Tranco's top domain list [8] for our analysis. Then, we 1) generate our own weak passwords that meet the five criteria, 2) enter the weak passwords into the password fields on the sign-up pages, and 3) record whether the websites allow or reject them. The best practice would be to reject such weak passwords. Our finding is threefold: first, the majority (on average, 53.5%) of websites do not conform to NIST PCP guidelines; second, only 22% of websites use the password-strength meter; and third, the majority (60%) of websites still adopt previously required composition rule – the composition rule have been officially depreciated by NIST.

Our contributions are summarized as follows:

- We conduct a measurement study of how properly websites follow the NIST PCP guidelines (five required criteria, a depreciated criterion, and one recommendation).
- Our study provides insights into the current Web ecosystem pertaining to the NIST PCP policies on popular websites. Particularly, we discover that the majority of the popular websites (on average, 53.5%) do not follow the guidelines provided by NIST.
- We offer recommendations for improving password composition policies on websites.

II. Background

A password is one of the most widely adopted means of end-user authentication. It is the responsibility of users to create strong passwords to safeguard personal, confidential, and sensitive information. To ensure the security of passwords, there is a de facto Password Composition Policies (PCP) [1] developed by the National Institute of Standards and Technology (NIST)

DOI 10.1109/SPW59333.2023.00006

that websites should adhere to within the Web ecosystem. This section provides a brief overview of the requirements and recommendations of the NIST PCP guidelines.

NIST Password Composition Policy (PCP) Guidelines. NIST provides website administrators with (1) *technical requirements* and (2) a *recommendation* to implement PCP guidelines on their websites. For the (1) *requirements*, the current version of the guidelines (800-63B) has five criteria that websites follow to ensure the security of users' passwords. These include checking passwords against breached and easily-guessable passwords, and not allowing users to freely composite passwords such as a length of at least eight characters. The (2) *recommendation* is to use a password-strength meter so that end-users can choose a strong password with real-time feedback (*i.e.*, how easily adversaries can crack their passwords).

- *Five-Criteria Requirement*: The requirements include five criteria [9] that help end-users when composing passwords, avoid the following ones: ① that are obtained from previously breached corpora (*e.g.*, "`passw0rd`" in a leaked password database [10]), ② that contain dictionary words (*e.g.*, "`9a!house32`" where "`house`" is a dictionary word), ③ that contain repetitive characters (*e.g.*, "`aaaa`"), ④ that have sequential characters (*e.g.*, "`abcd`"), and ⑤ that holds context-specific words including a service name or username (*e.g.*, "`facebook1!`" for the Facebook service [11]).
- *Depreciated Requirement*: It is noteworthy mentioning that a composition rule (from the previous NIST guidelines version 800-63-2 [12]) has been excluded, which recommends a mixed form of upper case letters, lower case letters, digits, and symbol characters in a single password. This is based on a prior study [10], which reveals that having a complex password does not increase the entropy of the security, but rather decreases usability. Our measurement study aims to better understand whether websites still follow the compliance of such depreciated composition rule.
- *Password-strength Meter Recommendation*: The NIST recommendation includes a password-strength meter to provide real-time feedback on users' password strengths, which assists end-users in generating stronger passwords. The real-time meter (typically displayed using a horizontal scale next to the password input field) represents the strength of a password with three scales: Strong, Medium, and Weak (the scale name may vary depending on websites). As no standard library or reference implementation is available, each website follows its own logic [13].

III. Motivation and Scope

Motivation. We aim to better understand how popular websites implement PCP in the wild. Despite efforts to encourage strong password creation, a survey conducted by Google [14] revealed that 24% of Americans still use weak and easily-guessable passwords. This indicates that some websites may still allow such passwords to be used. Our primary focus is on evaluating how website administrators comply with the NIST PCP guidelines [9]. Particularly, we seek to answer the following research questions:

- **RQ1)** *How do popular websites comply with the NIST password guidelines by each category?*
- **RQ2)** *Do websites still require their outdated PCP guidelines, such as the composition rule of upper case lower case letters, digits, and symbol characters in a single password?*
- **RQ3)** *How do websites implement password-strength meters?*

Scope. Our study merely focuses on measuring how websites follow the NIST PCP guidelines by manually attempting to insert passwords in the password fields on the sign-up web pages. The lifecycle of passwords (*e.g.*, resetting or recovering passwords when forgotten) and other authentication methods (*e.g.*, multi-factor authenticator) are out of scope.

IV. Dataset and Experiment Design

This section depicts our experimental design for studying PCP implemented by popular websites, focusing on how website administrators comply with the NIST PCP guidelines [1].

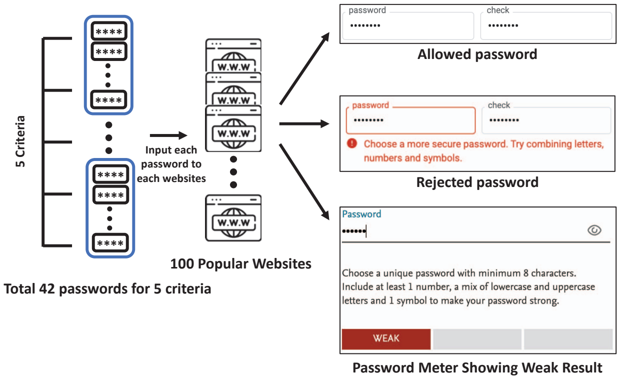
A. Dataset Collection

To obtain our dataset for the compliance with the NIST password guidelines, we utilize the Tranco 1M domains [8]. We selectively target a list of 100 popular websites from the domains by filtering out the followings: ① websites that are written in languages other than English, ② websites that require payments for sign-up via a credit card and/or paperwork, ③ websites that do not offer a sign-up, and ④ websites that are owned and serviced by the same companies (*e.g.*, google.com vs google.co.uk). We acquire the final list of the hundred websites after manually inspecting 248 websites that obey the above conditions.

B. Experimental Design

Design Overview. In this work, we choose NIST PCP that provides a standard in security [1]. For a better understanding of the compliance of the guidelines in the current websites' ecosystem, as shown in Figure 1, we generate our own *weak* passwords from every criterion that follows the guidelines, resulting in 42 passwords (*i.e.*, two for a context-specific criterion, 10 otherwise). Note that all generated passwords are listed Table I in Appendix. Next, we manually confirm whether each website accepts or rejects such weak passwords on its sign-up webpage. It is worth noting that there is no need to interact with a server because a password check has been made at the client side (*e.g.*, JavaScript).

Experiment on Required Criteria. To answer RQ1, we generate a list of *weak* passwords for our experiment by utilizing the "`Secure Password Generator`" tool [15] and carefully design our own standards, ensuring that each generated password belongs to each criterion of the NIST PCP guidelines. For each criterion, we use the above tool to include randomly generated passwords. For example, a password is generated with the options: *e.g.*, "Include Numbers", "Include Lowercase Characters", "Password Length". This tool helps generate



Choose a more secure password. Try combining letters, numbers and symbols.

Choose a unique password with minimum 8 characters. Include at least 1 number, a mix of lowercase and uppercase letters and 1 symbol to make your password strong.

Fig. 1. **Experiment Overview.** We conduct an experiment with a total of 42 passwords using five criteria on a hundred of popular websites of our choice. The results show whether the password has been allowed or rejected. Our experiment includes the proper use of password-strength meters.

a random password with various options that can include symbols, upper characters, lower characters, numbers, and their combinations. We manually double-check each password to avoid any overlap and ensure uniqueness. Section IV-C describes our generated passwords.

Experiment on Composition Rule. Furthermore, we investigate whether the outdated requirement of the "composition rule" is still being applied in the wild. The current version of the NIST PCP guidelines (800-63B) has removed the requirement of the "composition rule", which mandates a mixed form of upper case and lower case letters, digits, and symbol characters in a single password. This criterion was originally recommended under the belief that the composition of characters would provide higher entropy for passwords. However, in reality, end-users tend to create passwords with predictable patterns, such as including the first character as an upper case letter and just adding "1" and "!" at the end of the password to comply with the composition rule. The patterns can make composition passwords easier to breach [16]. We aim to determine if websites still follow the composition rule, even after five years have passed, which can answer RQ2.

Experiment on Strength Meter. In addition to the five required criteria, NIST recommends using a Password-Strength Meter as another measure. For our experiment, we pick one password from each criterion to examine if the websites properly use password-strength meters; having a total of five passwords for this experiment. The meters should provide end-users with real-time feedback, such as "Strong", "Medium", and "Weak" depending on the strength of the password. This experiment is designed to answer RQ3.

C. *Password Criterion for Experiments*

Criterion #1: Breached Passwords. Breached passwords are actual passwords that have been exposed in previous data breaches [17], making them trivial for adversaries to guess and compromise users' digital identities. To generate our list of breached passwords, we adopt two approaches: ① choosing passwords from the list of the breached passwords publicly released by the National Cyber Security Centre (NCSC) [18], and ② randomly generating passwords. Then, we cross-check all of our selected passwords using "Have I Been Pwned?" (HIBP) [19], which identifies exposed passwords. Note that this website offers a widely-used source for breached password research projects [20]–[23].

Criterion #2: Dictionary Words. This criterion aims to prevent end-users from using dictionary words in their passwords because passwords can be easily guessed with a brute-force attack with known word lists [24]. For example, a common dictionary word (*e.g.*, "house") can be in combination with other characters (*e.g.*, "9a!" and "32"), forming "9a!house32". To this end, we randomly generate 10 dictionary passwords and add randomly generated characters, including digits, symbols, upper and lower case letters. We ensure that the resulting passwords are not common dictionary words. It is important to note that a prior study [25] found that 33% of 5-character passwords containing additional restrictions (such as including a symbol) could be easily cracked.

Criterion #3: Repetitive Characters. Simply put, this criterion checks if a character is repeated in a password (*e.g.*, "aaaa", "1111"). Because NIST does not specify how many repeating characters are considered repetitive, nor does it limit repetitive characters to a single character type, we take one step further by incorporating patterns of repetitive characters: our passwords repeat patterns at least twice. For example, the 3^{rd} password from Table I has a pattern of "Zo01" repeated three times, with characters randomly generated with Secure Password Generator [15]. Previous work has shown that even with repetitive patterns, passwords are easily guessed or cracked [26], [27]. While repeating characters may increase the length of a password, thus increasing resistance to brute force attacks, it may be still vulnerable to attacks that identify repeated characters, similar to the Dictionary Words criterion.

Criterion #4: Sequential Characters. In essence, this criterion checks if the characters are in sequence (*e.g.*, "abcd" or "1234"). While such a sequence of letters can appear in a keyboard, NIST PCP does not specify a keyboard pattern (*e.g.*, "asdf" or "qwerty") in its criteria. However, it does not prohibit their inclusion because such a pattern could be fallen into a weak password category [28]. Hence, in this study, we include sequential keyboard patterns with a sequence of characters in our password list as well as non-keyboard sequential passwords to increase the variety. For the keyboard pattern, we randomly choose patterns from the standard QWERTY keyboard as it is most widely used in the world [29]. For the first three passwords in Table I, we display keyboard sequential characters. On the other hand, for the non-keyboard sequential passwords, we generate sequences of characters that are not in sequence on QWERTY. To exemplify, the 4^{th} password in Table I, "qas", has been chosen from the top left corner, and the 6th password, "mnb", from the bottom right corner of the QWERTY keyboard. In this criterion, we set our entropy as three sequential characters (either denotative sequence or keyboard pattern sequence).

Criterion #5: Context-Specific Words. This criterion checks if the user's password contains the name of the service or the username. Since service names and usernames are often publicly available, including them in the password can make it vulnerable to attacks akin to the Breached Passwords and Dictionary Words criteria. To test this criterion, we generate two passwords by appending "1" and "!" at the end of the password for the service name and username, respectively.

Recommendation: Password-Strength Meter. Recall that Password-Strength Meters play a role to provide real-time feedback on the strength of a user's password. The meter typically categorizes a password as Strong, Medium, or Weak; however, some websites may employ different terms like "Fair" instead of "Medium". We select a single password from each of the previous criteria and test it with various websites that employ password-strength meters, followed by comparing the results provided by each password-strength meter. Since our focus is to assess the results accuracy of password-strength meters, we use only one password per criterion.

Depreciated Requirement: Composition Rules. This criterion checks whether a password contains at least one digit, one uppercase letter, and one symbol. The goal behind this rule is to create a more complex password by combining different character types, thereby increasing a search space against brute-force attacks. However, previous findings indicate that users tend to create predictable patterns [10]. The composition rule cannot be predictable on its own; adversaries can use previously breached passwords to collect the patterns of the composition rule. For example, suppose that an end-user uses "`password`" for a password. The user may change the password to "`Password1!`" with a few modification such as replacing the lower case "`p`" with the upper case "`P`" and adding "`1`" and "`!`." As a result, the latest version of NIST PCP guidelines removed the requirement of the composition rule over 5 years ago. We want to better understand whether websites still apply the depreciated composition rule that requires an uppercase letter and a symbol in a password. For this criterion, we create a list of random passwords that are eight characters long and only contain digits and characters, without any additional restrictions typically found in composition passwords (*e.g.*, an uppercase letter or a symbol).

V. Measurement Results

A. *NIST Criteria Results*

From our empirical measurement study, we discover that for all criteria, on average, 53.5% of the websites do not comply with the NIST PCP guidelines. Particularly, in the Context-Specific criteria (that checks if a password contains the name of a service), 84% of websites do not follow the guidelines. As shown in Figure 2, a large number of websites do not reject weak passwords correctly, which means that those websites do not conform to the guidelines.

Breached Passwords. In this criterion, we want to see if the websites properly check end-users' passwords against breached password lists. From our findings, only 4% websites (4 out of 100) do reject all ten breached passwords: these websites are `github.com`, `theguardian.com`, `fandom.com`, and `grammarly.com`. In other words, most websites (96%) allow *at least* one of our breached passwords. More specifically, 34% websites (34 out of 100) allow all ten breached passwords. Out of those that allow all breached passwords, 82.3% of websites (28 out of 34) contain sensitive user information such as payment data. For example, `amazon.com` (one of the largest e-commercial websites where end-users' sensitive information such as payment information is stored) does not reject any passwords in our list. The only requirement they have is a minimum of six characters in length. Unauthorized access to this sensitive information can be obtained through a weak password. To mitigate online attacks such as brute forcing with breached passwords, websites can apply breached password-checking mechanisms. One of the recommendations would be to harness HIBP [19] because it provides an API to help websites check for breached passwords.

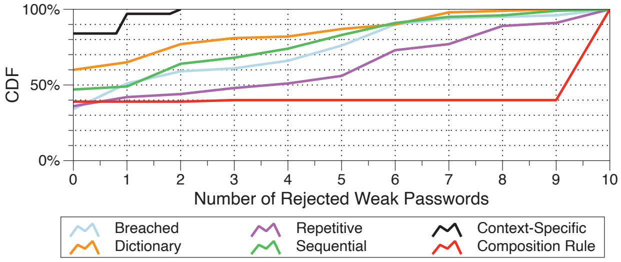


Fig. 2. **CDF of the number of rejected passwords for each criterion.** Note that Context-Specific criteria only have two passwords used, and in the Composition Rule, the number of passwords that are rejected is not complying the guideline. All criteria (on average, 53.5%) do not reject weak passwords properly. For the Context-Specific criteria, more than 80% websites do not reject weak passwords.

Dictionary Words. Remarkably, our finding shows that every website tested permits *at least* one password from our password list. Specifically, 60% of the tested websites (60 out of 100) allow *all* ten weak passwords from the list. The remaining 40 websites employ checking mechanisms to restrict the use of dictionary words in passwords. Of the 60 websites that allow all dictionary words, 86.7% of websites (52 out of 60) contain sensitive user information such as payment data. To address the security issue of dictionary words-based passwords, open-source libraries can be used to check a given password against a dictionary words list. For instance, "Free Dictionary API" [30] is one of the APIs that offers the capability to check if a string is in a dictionary.

Repetitive Characters. Regarding the use of repetitive characters in passwords, our analysis reveals that 91% of the tested websites (91 out of 100) allow at least one password that contains repetitive characters. In particular, 36% of the websites (36 out of 100) permit all ten passwords that are based on the repetition of characters. Of the websites that allow all such passwords, 83.3% websites (30 out of 36) stored sensitive user information, such as payment data. One approach to address the issue of repetitive characters in a password is for websites to implement a mechanism that verifies if a given password contains repetitive characters.

Sequential Characters. Similar to our previous criteria, our finding shows that 99% of the tested websites (99 out of 100) allow at least one password that contains sequential characters. The only website that does not allow all ten passwords is “`paypal.com`”. Our result shows that they block sequential keyboard patterns as well. Specifically, 47% of the tested websites (47 out of 100) allow all ten passwords that are based on sequential characters. Of the websites that allow all such passwords, 85.1% (40 out of 47) store sensitive user information, such as payment data. To address this criterion, websites can implement a mechanism that verifies if a given password contains sequential characters, with a more conservative approach being able to check for sequential keyboard characters.

Context Specific Words. In this criterion, only three websites properly follow the NIST PCP guidelines by rejecting all passwords in our list: `linkedin.com`, `wsj.com`, and `twitch.tv`. From those websites, `twitch.tv` shows a guided text such as “*Try a few random words. No special characters needed.*” to help users choose different passwords. Our findings indicate that 97% of the websites included in our study (97 out of 100) allow at least one context-specific password. Among these, 84% (84 out of 100) allow all context-specific passwords. Of the websites that allow all such passwords, 85.7% of websites (72 out of 84) contain sensitive user information, such as payment data. To mitigate attacks that exploit this criterion, such as brute force attacks, websites can implement a check to identify if a password contains the service name or username. Given that this information is typically not included in other criteria, it appears that websites do not currently check for the context-specific criterion.

Takeaway. The majority (on average, 53.5%) of websites allow weak passwords. We found that only 4% of websites check properly for breached passwords, and every website tested allows at least one password from the dictionary words list. Additionally, 84% of websites allow passwords containing the name of a service and a username.

B. Password-Strength Meter

The use of a password-strength meter is one of the recommendations by NIST to assist end-users in creating stronger passwords. However, our analysis of 100 websites reveals that only 22% (22 out of 100) of them utilize password-strength meters. This indicates that the usage of password-strength meters has not been widespread.

Furthermore, our analysis shows that even password-strength meters across multiple websites provide inconsistent results even with the same password. For instance, the password “`Il0vey0u!`” from the breached password criteria, out of 22 websites that utilize a password-strength meter, five produced a “`Strong`” result, two produced a ”`Medium`” result, and 15 produced a “`Weak`” result. However, the results for the sequential character criteria are more consistent, with only one out of 22 websites producing a “`Medium`” result, and one producing a “`Weak`” result, while the remaining 20 produced a “`Strong`” result.

Inconsistencies in the effectiveness of password-strength meters may be attributed to the lack of a standard guideline for their design and implementation. Website developers are often required to design and implement their own meters, leading to variations in their effectiveness [31]. These inconsistencies can potentially compromise the security of user accounts.

Takeaway. While NIST recommends using password-strength meters, only 22% of our examined websites use them. Moreover, the results of the analysis show that password-strength meters across multiple websites provide inconsistent results.

C. Depreciated Requirement: Composition Rule

In this criterion, our passwords are created to check whether websites require composition rules for end-users’ passwords. Note that passwords allowed mean websites are complying with the NIST PCP guideline, but rejecting our passwords means websites are not complying with the NIST PCP guideline. In the composition rule criterion, a majority of websites, 60% of websites (60 out of 100), do not allow all 10 passwords, while 39% (39 out of 100) do. This suggests that a significant portion of websites still adhere to the depreciated composition rule. Out of those that allow all composition rule passwords, 90% (54 out of 60) contained sensitive user information, such as payment data. However, there is an exception from allowing all 10 passwords (`twitch.tv`), which allows only 7 out of 10 passwords in our list. `twitch.tv` does not require the composition rule but blocks passwords that are randomly generated and do not have composition characters, considering them to be keyboard sequential patterns or not randomly generated. While this criterion may seem to loosen password requirements for users, more than half of websites still use outdated policies. Overall, the majority of websites appear to believe in the efficacy of the outdated composition rules for creating strong passwords, or they may not be following the recommended guidelines for other reasons.

Takeaway. The majority of websites (60%) still enforce composition rules, while 39% do not. Out of those that allowed all composition rule passwords, 90% contain sensitive user information such as payment data.

D. Category Analysis and Other Findings

With the measurement results, we also categorize each website URL using Fortiguard Lab’s Web Filter lookup tool [32], in order to analyze if particular categories of websites are better or more poorly enforced NIST’s criteria. We highlight a few noteworthy examples of our discovery.

Our results show that the tested websites fall into a total of 27 categories. Notably, in the breached password criterion, 80% (4 out of 5) of the websites in the Shopping category allow all breached passwords. Regarding the dictionary word criterion, 100% (7 out of 7) of the websites in the Education category and 83.3% (5 out of 6) of the websites in the Social Networking category allowed all dictionary passwords.

For the repetitive characters criterion, 60% (3 out of 5) of the websites in the Shopping category allow all repetitive passwords. In the sequential characters criterion, 83.3% (5 out of 6) of the websites in the Social Networking category allow all sequential passwords. Furthermore, all websites in the Social Networking, Shopping, File Sharing and Storage, and Education categories allow all context-specific passwords, indicating a lack of adherence to NIST's recommendations.

Unexpected Behavior. There are websites that only follow the NIST PCP guidelines partially. Because when a password is hashed, the size and length of the password are independent, NIST guidelines do not limit the maximum size of passwords. However, our analysis finds that 11 websites limit password length between 15 and 50 characters: `paypal.com`, `tiktok.com`, `alibaba.com`, `cnet.com`, `usatoday.com`, `bbc.co.uk`, `issuu.com`, `wsj.com`, `dailymail.co.uk`, `ibm.com`, and `samsung.com`. While the minimum requirement for password length is eight characters, the following websites are found to allow even less than eight characters in length: `myshopify.com`, `archive.org`, `wsj.com`, `dailymail.co.uk`, `alibaba.com`, `usatoday.com`, and `unsplash.com`.

VI. Discussion and Future Work

Best Practices. Our findings show that the NIST PCP guidelines have not been well applied across different websites. It is worth noting that the latest version of the NIST PCP guidelines is 800-63B, which includes substantial updates. Moreover, a new version is currently in draft form [33], which means there will be even newer guidelines in the future. To ensure the best security practices for password policies, we recommend that website administrators keep themselves informed about the latest updates on the NIST PCP guidelines. This will enable them to implement the latest best practices for creating a secure password policy on their websites.

Dataset Representativeness and Generalizability. Due to our work required to manually visit each website, our number of websites and passwords are limited. We focus on 100 popular websites and we limit our list of passwords to 10 for each criterion (2 for Context-Specific Words). Although it is difficult to generalize our results due to the volume of the dataset, our findings can still be considered impactful as we focus on the Top 100 popular websites that are highly influential. To further explore the impact of a language on password policies, future studies could expand beyond English-based websites.

Breached Password List. Besides, while our study relies on the HIP database, incorporating additional password lists that contain breached password information could enhance the comprehensiveness of our analysis. Note that providing further insights into password policies across various online platforms is part of our future work.

Manual Evaluation. Because our study involves manual analysis, automated tools could expand the scope of the investigation and cover a larger number of websites. Hence, one avenue for future research could involve the development of automated tools for checking compliance with PCP guidelines, broadening the scope of our investigation to encompass a larger number of websites.

Ethics. Our study utilizes a manual approach where we attempt to enter passwords into the designated password fields on the sign-up pages without creating an account. As a result, our findings do not pose ethical concerns for the websites under scrutiny.

VII. Related Work

There has been a number of studies on improving a password entropy with a focus on a policy [2]–[4], meter [31], [34], [35], and composition rule [13], [16], [25], [36], [37].

Password Policy. The Password Policy has been provided for websites to help users create strong passwords. However, websites did not follow the policy [2]. A broader study of Password Policy has been done for each continent [3], [4]. These studies used the top 50 from each country to review Password Policies. Our work on the other hand measured 100 websites manually with the recent version of the NIST guideline whereas previous works were focused on the previous version of the NIST guidelines.

Password Meter. Studies have seen the benefit of having a Password Meter with users updating to stronger passwords based on recommendations provided by Password Meters [34]. This shows the password meter could be helpful for creating a strong password. However, our research shows that only 22% of websites are using the Password Meter. Also, our result is in line with multiple studies [31], [35] which show that Password Meters on different websites give different results.

Composition Rule. Our work showed that the majority of websites do not adopt the current guideline of the Composition Rule. Previous works were mostly focused on checking what is the most reasonable entropy for the complexity and length of the password [13], [16], [25], [36], [37]. Because many studies focus on improving entropy for the complexity and length of the password, they do not seem to consider the removal of the Composition Rule from the guideline.

VIII. Conclusion

Our study has revealed that on average, 53.5% of the examined websites conform to the NIST PCP guidelines in their implementations. Additionally, our findings show that 60% of websites still use outdated PCP guidelines, such as requiring a combination of upper case and lower case letters, digits, and symbol characters in a single password. This practice can jeopardize password security and put users' digital identities at risk. Furthermore, our research discovers that only 22% of websites utilize password-strength meters to assist users in creating stronger passwords, indicating that many websites are not taking appropriate measures to promote password security and usability. These findings highlight the need for more efforts to improve password security on websites. Website operators should adopt the latest password guidelines and best practices recommended by NIST, including the use of password-strength meters.

Acknowledgment

We thank the anonymous reviewers for their constructive feedback. The authors gratefully acknowledge the support of NSF (2210137). This work was supported by Science Alliance's StART program and a gift from Google exploreCSR, and partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (Ministry of Science and ICT) (No. 2022-0-01199; Graduate School of Convergence Security (Sungkyunkwan university)). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsor.

References

[1] "Nist special publication 800-63b," https://pages.nist.gov/800-63-3/sp800-63b.html, (Accessed on 02/06/2023).

[2] D. Florêncio and C. Herley, "Where do security policies come from?" in *Proceedings of the Sixth Symposium on Usable Privacy and Security*, 2010, pp. 1–14.

[3] P. Mayer, J. Kirchner, and M. Volkamer, "A second look at password composition policies in the wild: Comparing samples from 2010 and 2016," in *SOUPS Thirteenth Symposium on Usable Privacy and Security, July 12–14, 2017, Santa Clara, CA, USA*. USENIX-The Advanced Computing Systems Association, 2017, pp. 13–28.

[4] A. Gautam, S. Lalani, and S. Ruoti, "Improving password generation through the design of a password composition policy description language," in *Eighteenth Symposium on Usable Privacy and Security (SOUPS 2022)*. Boston, MA: USENIX Association, Aug. 2022, pp. 541–560. [Online]. Available: https://www.usenix.org/conference/soups2022/presentation/gautam

[5] J. Tan, L. Bauer, N. Christin, and L. F. Cranor, "Practical recommendations for stronger, more usable passwords combining minimum-strength, minimum-length, and blocklist requirements," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020, pp. 1407–1426.

[6] B. Ur, F. Alfieri, M. Aung, L. Bauer, N. Christin, J. Colnago, L. F. Cranor, H. Dixon, P. Emami Naeini, H. Habib *et al.*, "Design and evaluation of a data-driven password meter," in *Proceedings of the 2017 chi conference on human factors in computing systems*, 2017, pp. 3775–3786.

[7] H. Habib, J. Colnago, W. Melicher, B. Ur, S. Segreti, L. Bauer, N. Christin, and L. Cranor, "Password creation in the presence of blacklists," *Proc. USEC*, p. 50, 2017.

[8] V. L. Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczyński, and W. Joosen, "Tranco: A research-oriented top sites ranking hardened against manipulation," *arXiv preprint arXiv:1806.01156*, 2018.

[9] "Nist special publication 800-63b," https://pages.nist.gov/800-63-3/sp800-63b.html#sec5, (Accessed on 02/15/2023).

[10] "Nist special publication 800-63b," https://pages.nist.gov/800-63-3/sp800-63b.html#appA, (Accessed on 02/15/2023).

[11] "Facebook," https://www.facebook.com/, (Accessed on 02/22/2023).

[12] "Electronic authentication guideline," https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-2.pdf, (Accessed on 02/16/2023).

[13] M. Golla and M. Dürmuth, "On the accuracy of password strength meters," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1567–1582. [Online]. Available: https://doi.org/10.1145/3243734.3243769

[14] "Of_google_harrispoll_national_03," https://storage.googleapis.com/gweb-uniblog-publish-prod/documents/PasswordCheckup-HarrisPoll-InfographicFINAL.pdf, (Accessed on 02/06/2023).

[15] "Strong random password generator," https://passwordsgenerator.net/, (Accessed on 02/08/2023).

[16] S. Komanduri, R. Shay, P. G. Kelley, M. L. Mazurek, L. Bauer, N. Christin, L. F. Cranor, and S. Egelman, "Of passwords and people: measuring the effect of password-composition policies," in *Proceedings of the sigchi conference on human factors in computing systems*, 2011, pp. 2595–2604.

[17] "World's biggest data breaches & hacks — information is beautiful," https://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/, (Accessed on 02/15/2023).

[18] "Passwords, passwords everywhere - ncsc.gov.uk," https://www.ncsc.gov.uk/blog-post/passwords-passwords-everywhere, (Accessed on 02/23/2023).

[19] "Have i been pwned: Check if your email has been compromised in a data breach," https://haveibeenpwned.com/, (Accessed on 02/08/2023).

[20] G. Sood and K. Cor, "Pwned: The risk of exposure from data breaches," in *Proceedings of the 10th ACM Conference on Web Science*, ser. WebSci '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 289–292. [Online]. Available: https://doi.org/10.1145/3292522.3326046

[21] K. Cor and G. Sood, "Pwned: How often are americans' online accounts breached?" *arXiv preprint arXiv:1808.01883*, 2018.

[22] D. Pereira, J. F. Ferreira, and A. Mendes, "Evaluating the accuracy of password strength meters using off-the-shelf guessing attacks," in *2020 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*. IEEE, 2020, pp. 237–242.

[23] T. Malderle, M. Wübbeling, S. Knauer, A. Sykosch, and M. Meier, "Gathering and analyzing identity leaks for a proactive warning of affected users," in *Proceedings of the 15th ACM international conference on computing frontiers*, 2018, pp. 208–211.

[24] "What is a dictionary attack? | nordpass," https://nordpass.com/blog/what-is-a-dictionary-attack/, (Accessed on 02/13/2023).

[25] R. W. Proctor, M.-C. Lien, K.-P. L. Vu, E. E. Schultz, and G. Salvendy, "Improving computer security for authentication of users: Influence of proactive password restrictions," *Behavior Research Methods, Instruments, & Computers*, vol. 34, pp. 163–169, 2002.

[26] A. Narayanan and V. Shmatikov, "Fast dictionary attacks on passwords using time-space tradeoff," in *Proceedings of the 12th ACM conference on Computer and communications security*, 2005, pp. 364–372.

[27] S. Houshmand, S. Aggarwal, and R. Flood, "Next gen pcfg password cracking," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 8, pp. 1776–1791, 2015.

[28] K. Yang, X. Hu, Q. Zhang, J. Wei, and W. Liu, "Studies of keyboard patterns in passwords: recognition, characteristics and strength evolution," in *Information and Communications Security: 23rd International Conference, ICICS 2021, Chongqing, China, November 19-21, 2021, Proceedings, Part I 23*. Springer, 2021, pp. 153–168.

[29] J. Noyes, "The qwerty keyboard: A review," *International Journal of Man-Machine Studies*, vol. 18, no. 3, pp. 265–281, 1983.

[30] "Free dictionary api," https://dictionaryapi.dev/, (Accessed on 02/25/2023).

[31] X. de Carné de Carnavalet and M. Mannan, "From very weak to very strong: Analyzing password-strength meters," in *Network and Distributed System Security Symposium (NDSS 2014)*. Internet Society, 2014.

[32] "Web filter lookup | fortiguard," https://www.fortiguard.com/webfilter, (Accessed on 02/23/2023).

[33] "Nist sp 800-63 digital identity guidelines," https://pages.nist.gov/800-63-3/, (Accessed on 02/24/2023).

[34] B. Ur, P. G. Kelley, S. Komanduri, J. Lee, M. Maass, M. L. Mazurek, T. Passaro, R. Shay, T. Vidas, L. Bauer *et al.*, "How does your password measure up? the effect of strength meters on password creation." in *USENIX Security Symposium*, 2012, pp. 65–80.

[35] D. He, B. Zhou, X. Yang, S. Chan, Y. Cheng, and N. Guiana, "Group password strength meter based on attention mechanism," *IEEE Network*, vol. 34, no. 4, pp. 196–202, 2020.

[36] P. G. Kelley, S. Komanduri, M. L. Mazurek, R. Shay, T. Vidas, L. Bauer, N. Christin, L. F. Cranor, and J. Lopez, "Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms," in *2012 IEEE Symposium on Security and Privacy*, 2012, pp. 523–537.

[37] K.-P. L. Vu, R. W. Proctor, A. Bhargav-Spantzel, B.-L. B. Tai, J. Cook, and E. Eugene Schultz, "Improving password security and memorability to protect personal and organizational information," *International Journal of Human-Computer Studies*, vol. 65, no. 8, pp. 744–757, 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1071581907000560

Appendix

Passwords List for Measure. The Table I shows a list of passwords that we created to examine each of the 100 popular websites. We generate our own *weak* passwords from every criterion that follows the guidelines, resulting in 42 passwords (*i.e.*, two for a context-specific criterion, 10 otherwise). Numbers in table II are the number of passwords that were not allowed for 5 criteria (Breached Passwords, Dictionary Words, Repetitive Characters, Sequential Characters, and Context-Specific Words). For the Composition Rule criterion, numbers are the number of passwords that were allowed.

Measurement Results for 100 Websites. The Table II shows websites of our choice from Tranco's Top 1 million [8].

Measurement Results for Password-Strength Meter. The table III shows the result of the password meter for each criterion.

TABLE I
PASSWORD LISTS FOR EACH CRITERIA EXCEPT FOR CONTEXT-SPECIFIC WORDS CRITERIA, 10 PASSWORDS FOR EACH CRITERION WERE CREATED. PASSWORDS FOR PASSWORD-STRENGTH METERS WERE CHOSEN FROM EACH CRITERION.

Criteria	**Passwords**
Breached Passwords	Pa$$w0rd Oct312010 7/4/1776 Sept171787 Q1w2E3r4T5y6 Il0vey0u! A1d2m3i4n5 Pa$$w0rd1! Pa$$w0rd123! p1a1s1s1w1o1r1d1
Dictionary Words	9a!house32 Laptop!63 DiCtIoNaRy2! Thisisasentencethatisusedasapassword1! ComputerPassword231 4Potatoes4 COOLUser1 SpaceBar123 SoftwareSecurity566 43water_bottle
Repetitive Characters	11aa%%11aa%% bbbb&&&&6666 ZoO1ZoO1ZoO1 AhEq3AhEq3 1!1!1!1!1!1!1!1!1!1!1!1!1!1!1!1!1!1!1! ttttttttttttttttttttttttttttttttttttttttttttttttttt 33333333333333333333333333333333 HhHhHhHhHhHhHhHh 00100010001000010 oo000o0oo0o0oo000
Sequential Characters	abcdefghijklmnopqrstuvwxyz123456789!@#$%^&* ka@dm123456789 Abcdefghij1234567 1qaz@WSX3edc$RFV5tgb^YHN7ujm mju7NHY^bgt5VFR$cde3XSW@zaq1 MnBvCxZaSdFgHjKlPoIuYtRrEwQ1@3$5^7*9 13$5^7*9qWeRtYuIoP lkjhgfdsaLKJHGFDSA mnbvcxzLKJHGFDSA 1qaz0okm2wsx9ijn3edc8uhb4rfv7ygv
Context-Specific Words	Service name + user1! Username + 1!
Composition Rules	jdcm889r hnjn8ybv 3kce5w56 he747wgx bsvek36k ppbq2k9t q7rz2hb3 ket2yuug 9hzqrdes q7dmez6k
Password-Strength Meters	Il0vey0u! 43water_bottle AhEq3AhEq3 1qaz0okm2wsx9ijn3edc8uhb4rfv7ygv Username + 1!

TABLE II

PASSWORD MEASURE RESULTS.

THE NUMBERS INDICATE THE NUMBER OF PASSWORDS THAT WERE NOT ALLOWED.

Rank	URL	BP	DW	RC	SC	CS	CR	Rank	URL	BP	DW	RC	SC	CS	CR
1	google.com	1	0	0	0	0	0	128	wsj.com	5	3	8	9	2	10
3	facebook.com	0	0	1	0	0	10	129	wix.com	0	0	0	0	0	10
4	netflix.com	0	0	0	0	0	10	133	stackoverflow.com	1	0	6	2	0	0
5	microsoft.com	0	0	3	0	1	0	140	businessinsider.com	2	2	8	5	0	10
6	twitter.com	0	0	0	0	0	0	143	researchgate.net	1	0	4	0	0	10
7	instagram.com	0	0	0	0	0	10	144	imgur.com	1	0	6	2	0	10
9	linkedin.com	4	0	2	0	2	0	145	indeed.com	0	0	0	0	0	0
10	apple.com	5	2	8	5	1	10	148	slideshare.net	0	0	1	1	0	0
14	amazon.com	0	0	0	0	0	10	153	cnbc.com	6	7	10	6	0	10
16	yahoo.com	5	0	5	0	1	0	160	dailymail.co.uk	1	3	6	9	1	10
17	wordpress.org	7	3	6	3	0	0	161	mailchimp.com	6	7	10	6	0	10
22	pinterest.com	1	0	3	0	0	0	162	godaddy.com	2	0	0	0	0	0
23	vimeo.com	5	5	8	4	0	10	163	nature.com	1	0	6	2	0	0
24	adobe.com	3	2	8	5	1	10	167	ibm.com	6	3	10	6	0	10
26	zoom.us	2	2	8	5	0	10	168	tradingview.com	1	0	6	2	0	10
29	amazonaws.com	2	0	6	3	0	10	169	intuit.com	6	7	10	7	0	10
30	github.com	10	1	2	4	0	0	173	alibaba.com	0	1	6	7	0	10
36	wordpress.com	1	0	0	0	0	10	174	cnet.com	6	6	8	6	0	10
37	reddit.com	0	0	0	0	0	0	180	aliyun.com	2	1	6	5	0	10
39	bit.ly	7	0	9	0	0	10	181	fandom.com	10	0	1	0	1	10
50	tumblr.com	8	1	1	1	1	0	182	hp.com	6	7	10	6	0	10
57	nytimes.com	0	0	0	0	0	0	183	usatoday.com	1	0	0	0	0	10
58	flickr.com	7	5	1	0	0	0	184	unsplash.com	0	0	0	0	0	10
63	spotify.com	0	0	0	0	0	0	185	springer.com	1	0	6	2	0	10
65	soundcloud.com	0	0	0	0	0	0	186	booking.com	4	2	0	0	0	0
66	dropbox.com	0	0	0	0	0	10	187	eventbrite.com	0	0	0	0	0	0
69	canva.com	5	0	6	2	0	0	189	shopify.com	0	0	0	0	0	10
71	forbes.com	6	7	10	6	0	10	193	surveymonkey.com	0	0	0	0	0	0
73	cnn.com	1	0	5	2	0	0	195	yelp.com	0	0	0	0	0	10
77	myshopify.com	0	0	0	0	0	10	198	time.com	6	6	8	4	0	10
79	cloudflare.com	5	5	7	4	0	10	200	aol.com	5	0	5	2	1	0
81	archive.org	0	0	0	0	0	10	202	npr.org	0	0	0	0	0	10
82	paypal.com	1	2	7	10	0	0	205	samsung.com	3	4	7	9	0	10
86	twitch.tv	9	8	6	5	2	3	208	ted.com	0	0	0	0	0	0
87	theguardian.com	10	0	1	0	0	0	212	walmart.com	2	2	8	5	0	10
88	ebay.com	1	2	6	2	1	10	217	themeforest.net	4	2	3	2	1	0
89	imdb.com	0	0	0	0	0	0	218	indiatimes.com	7	9	9	7	1	10
90	sourceforge.net	4	2	0	0	0	0	222	wired.com	0	0	0	0	0	10
91	tiktok.com	6	6	8	8	0	10	224	udemy.com	4	0	5	2	0	0
92	bbc.co.uk	1	0	6	2	0	0	225	myspace.com	0	0	4	0	0	10
97	digicert.com	5	2	6	3	0	10	228	grammarly.com	10	0	0	0	0	0
107	issuu.com	0	0	0	0	0	10	232	techcrunch.com	5	0	5	2	0	0
109	weebly.com	1	0	0	0	1	0	234	dailymotion.com	6	5	8	4	0	10
113	etsy.com	0	0	0	0	0	10	235	cpanel.com	5	0	4	2	0	0
117	sciencedirect.com	0	0	0	0	0	0	237	goodreads.com	0	0	0	0	0	10
118	reuters.com	6	7	10	7	0	10	240	huffingtonpost.com	2	0	6	3	0	10
121	washingtonpost.com	6	5	7	4	0	10	242	zillow.com	6	7	10	6	0	10
125	aliexpress.com	0	1	3	5	0	10	244	squarespace.com	0	0	0	0	0	0
126	oracle.com	6	7	10	6	1	10	245	independent.co.uk	2	2	8	5	0	10
127	tinyurl.com	0	0	0	0	0	10	248	espn.com	1	0	6	2	0	10

* BP: Breached Passwords. * DW: Dictionary Words.* RC: Repetitive Characters * SC: Sequential Characters * CS: Context-Specific Words * CR: Composition Rules

TABLE III

PASSWORD-STRENGTH METER RESULTS

PASSWORD-STRENGTH METER MEASURE RESULTS. EACH WEBSITES PROVIDED METERS WITH "STRONG", "MEDIUM", AND "WEAK".

Rank	URL	Breached Passwords	Dictionary Words	Repeated Characters	Sequential Characters	Context-Specific
3	facebook.com	Strong	Strong	Strong	Medium	Strong
10	apple.com	Strong	Strong	Medium	Weak	Weak
17	wordpress.org	Weak	Medium	Medium	Weak	Medium
30	github.com	Weak	Strong	Strong	Strong	Strong
37	reddit.com	Medium	Strong	Strong	Weak	Strong
50	tumblr.com	Weak	Strong	Medium	Strong	Strong
58	flickr.com	Strong	Strong	Weak	Strong	Strong
66	dropbox.com	Medium	Strong	Strong	Weak	Strong
69	canva.com	Medium	Medium	Medium	Weak	Medium
86	twitch.tv	Weak	Weak	Weak	Weak	Weak
113	etsy.com	Strong	Strong	Medium	Medium	Strong
117	sciencedirect.com	Strong	Strong	Medium	Medium	Medium
143	researchgate.net	Strong	Strong	Strong	Medium	Strong
160	dailymail.co.uk	Weak	Strong	Weak	Strong	Weak
162	godaddy.com	Medium	Strong	Strong	Weak	Strong
183	usatoday.com	Strong	Strong	Medium	Medium	Medium
187	eventbrite.com	Medium	Strong	Strong	Weak	Strong
195	yelp.com	Strong	Strong	Medium	Strong	Strong
208	ted.com	Medium	Strong	Strong	Weak	Strong
224	udemy.com	Medium	Strong	Strong	Weak	Strong
235	cpanel.com	Medium	Strong	Medium	Weak	Medium
242	zillow.com	Medium	Strong	Weak	Weak	Weak