Tighter Analysis of Password Guessing Curves with Applications to PINs

Anonymous Submission

Abstract

A fundamental challenge in password security is to understand and characterize the resilience of user chosen passwords to brute-force guessing attacks. However, it can be challenging to characterize an attacker's guessing curve because we don't know which guessing strategy that the attacker will follow and the user password distribution is unknown to us. Following Kerckhoff's principle Blocki and Liu [8] introduced several statistical techniques to obtain high-confidence upper and lower bounds on the guessing curve of an optimal attacker who knows the password distribution. Even if the password distribution is unknown these statistical techniques can be used to upper/lower bound the attacker's guessing curve as long as we can obtain iid samples from the unknown password distribution. While these statistical bounds yield reasonably close bounds on the attacker's guessing curve in certain settings, their empirical analysis also highlighted the limitations these statistical techniques. In particular, the upper and lower bounds diverge rapidly when the sample size is too small, and the upper/lower bounds for small guessing budgets are sub-optimal due to an small additive error term. In this paper, we propose two new statistical techniques providing upper and lower bounds on password guessing curve under the optimal attacker setting. We apply our bounds to analyze eight password datasets and two PIN datasets with different sample sizes. Our empirical analysis shows that our new statistical techniques often yield tighter upper/lower bounds especially in settings where the number of samples is small or where the attacker's guessing budget is small. We then give a theoretical analysis characterizing when we are guaranteed to have close upper/lower bounds as a function of the number of password samples N. Our results imply that as long as the guessing budget is $o(N/\log N)$ the additive gap between upper/lower bounds will be small with high probability. This theoretical analysis provides the first rigorous justification of the heuristic use of the empirical distribution to analyze the attacker's guessing curve for smaller guessing budgets $o(N/\log N)$. We also apply our statistical techniques to rigorously quantify the impact of blocklists on PIN distributions.

1 Introduction

In password security a fundamental challenge is to characterize the resilience of user chosen passwords to brute-force guessing attacks. Understanding and characterizing the attacker's guessing curve is crucial before we can make informed decisions about password policies involving tradeoffs between usability and security. How many incorrect guesses do we allow before locking an account? How restrictive should our password composition policy[1] be? How expensive does the password hash function need to be to mitigate the threat of offline brute-force attacks? To address these questions we would like to characterize the attacker's guessing curve. In particular, what is the probability that an attacker will crack a random user's password within the first G guessing attempts as the number of guesses G ranges from small (online attacks) to large (offline attacks). However, it can be challenging to characterize the attacker's guessing curve because we don't know the exact guessing strategy that the attacker will follow nor do we have an exact description of the user password distribution.

One approach to characterize the attacker's guessing curve is to consider an empirical distribution derived from samples taken from our user password distribution e.g., breached password datasets. While this heuristic approach has been utilized by multiple papers [2,4–6,10,15,16], it can lead to overly pessimistic security estimates. For example, the support of the empirical distribution is upper bounded by the number of samples N so, if we follow the empirical distribution, we will make the pessimistic (and likely inaccurate) prediction that the attacker will crack 100% of passwords within G guesses. Empirical analysis of Blocki and Liu [8] rigorously demonstrated that the empirical distribution overestimates the true guessing curve for larger guessing budgets. On the positive side the empirical analysis also indicated that empirical distribution provides a reasonable approximation

[1] An example one password composition policy might be to require passwords that are *at least* 8 characters long and have at least 1 number and 1 letter.

of the attacker's guessing curve when the guessing budget is "sufficiently small." However, this was merely an empirical observation and there was no theoretical guarantee that this is always true.

Another approach to characterize the attacker's guessing curve is to fix a particular password cracking model and use this model to analyzing breached password datasets — we can view breached password datasets as samples from our unknown password distribution. While this approach can help us to understand the guessing curve for a particular attacker, we do not know a priori what model the attacker will use[2] or how the parameters of the attacker's model were trained/tuned. Furthermore, it not unreasonable to assume that the attacker's password model will outperform publicly available models. Indeed, empirical analysis of [8] indicates that even state of the art password cracking models can significantly underestimate the fraction of passwords cracked by an optimal attacker who understands the password distribution. Thus, it could be risky to base policy decisions off of a guessing curve which may not necessarily reflect the guessing curve of the actual adversary.

Blocki and Liu [8] advocated that one should follow Kerckhoff's principle when setting password policies and assume that the attacker knows the password distribution and can order guesses in descending order of likelihood. Unfortunately, we cannot directly compute the guessing curve for our optimal attacker as the user password distribution is still unknown to us. Thus, they introduced several statistical techniques to obtain high-confidence upper and lower bounds on the guessing curve of an ideal attacker (λ_G) using samples from the unknown password distribution. Empirically analysis showed that the upper/lower bounds were reasonably close as long as the guessing number G was not too large and as long as the number of samples from the unknown password distribution is sufficiently large.

However, prior work provided no theoretical guarantees characterizing when the upper/lower bounds would be tight. Furthermore, their empirical analysis highlighted several limitations of their techniques. In particular, the upper/lower bounds diverge rapidly when the guessing budget G is very large or if the sample size N is too small. We also observe that when the guessing number G is small the upper/lower bounds are not tight e.g., for Brazzers dataset with guessing number $G = 4$ the bounds are $0 \leq \lambda_G \leq 1.723\%$. While an additive gap of 1.723% between the upper/lower bound might be acceptable when analyzing offline attacks where G and λ_G are larger, such a gap can prevent us from drawing meaningful conclusions about the threats posed by online password spraying attacker with a small guessing budget G. Finally, we observe that the upper/lower bounds are sub-optimal in settings where the size of the support of the user password distribution is small (e.g., 4-digit PIN numbers), but the the number of samples from this distribution is also small.

1.1 Our Contributions

First, we propose two new statistical techniques providing upper and lower bounds on the password guessing curve λ_G. Empirical analysis demonstrates that our techniques yield tighter lower/upper bounds than comparable techniques from [8] especially when the guessing number G is small (e.g., online password spraying attacks).

Second, we give a theoretical analysis which characterizes when we are guaranteed to have close upper/lower bounds as a function of the number of samples N. In particular, our results imply that as long as $G = o(N/\log N)$ that the additive gap between our upper/lower bound will be small (whp). This result also implies that (with high probability) the empirical distribution defined by these N samples yields a guessing curve that is very close to that of an attacker who knows the password distribution as long as the guessing number $G = o(N/\log N)$ is not too large. This provides the first rigorous justification of the use of the empirical distribution to analyze the attacker's guessing curve with the caveat that the empirical guessing curve may be inaccurate once the guessing number G approaches or exceeds $N/\log N$.

Third, we apply our bounds to analyze eight password datasets and two PIN datasets, and compare our bounds with prior bounds [8]. In empirical analysis, we show that our new techniques significantly improve prior work [8] e.g., for Battlefield Heroes dataset [44] with guessing budge $G = 128$ our new bounds proves $3.319\% \leq \lambda_G \leq 3.648\%$ in comparison to prior work $2.360 \leq \lambda_G \leq 3.882\%$, and for a small user study dataset [32] with guessing budge $G = 2$ our new bounds proves $1.396\% \leq \lambda_G \leq 4.438\%$ in comparison to prior work $0 \leq \lambda_G \leq 8.022\%$ obtained from Blocki and Liu [8]. Under the normalized probability model setting [7], we can apply our techniques to quantify the impact of applying blocklists to PIN datasets. The experiment results show that blocklists with size of 2740 and 4000 can better strengthen the PIN distribution than smaller blocklists and no blocklist, while prior bounds are not tight enough to verify this observation. We also apply our statistical techniques to discuss the similarity between PIN distributions and the validity of the normalized probability assumption for PIN distributions. See analysis in Section 5.3.

1.2 Related Work

Bounding Password Distribution. The empirical password distribution and offline password cracking models have been used to estimate password guessing curves to evaluate many research ideas, such as tuning cost parameters for password hashing and distribution-aware password throttling mecha-

[2] There are many different password cracking models including Neural Network Models [36], Markov Models [14, 21, 31, 38], Probabilistic Context-Free Grammars [42, 46] and heuristic rule-based tools such as John the Ripper [26] and Hashcat [25].

nisms [2,4], evaluating distribution-aware password throttling mechanisms [9,40], and quantifying an attacker's advantage of knowing the password length [24]. Blocki and Liu [8] prove that these estimates can significantly overestimate or underestimate the actual guessing curve of a perfect knowledge attacker by introducing several provable upper and lower bounds that hold with high confidence. However, their techniques rely on a large password sample set to reduce the additive error of the bounds and provide no theoretical guarantee understanding when the upper/lower bounds would be tight. Another prior work [6] also propose a lower bound on guessing curves but the bound is less tight than Blocki and Liu [8].

Password Cracking Models. Offline attacks have been studied for decades. Many probabilistic password models have been proposed such as Markov models [13, 14, 31, 41], neural networks [36], and Probabilistic Context-Free Grammars [27, 42, 46]. Password cracking algorithms based on probabilistic password models are often prohibitively expensive to count the accurate guessing number. Thus, Monte-Carlo strength estimation [20] is proposed as a tool for defenders to efficiently approximate the guessing number of a given password. Confident Monte Carlo [30] gives several counterexamples where the Monte Carlo estimation is inaccurate, and tightly bounds the guessing numbers and guessing curves with high confidence. Heuristic (rule-based) software tools Hashcat [25] and John the Ripper [26] are used more oftehn by real world attackers. Liu et al. [29] develop tools to estimate guessing numbers for such software tools without simulating the full attack.

Applying Blocklists to Strengthen PIN Distribution. Personal Identification Number (PIN) is one of the common authentication methods that are used in many areas such as ecrypting mobile devices and banking cards. Bonneau et al. [11] studies the user choice of 4-digit PINs in the chip-and-PIN systems for the use of credit cards and ATMs. They find birthday is a popular selection of PINs which an attacker can leverage to improve the guessing strategy. Wang et al. [45] identify the difference between PINs created by English and Chinese users, and also find that 6-digits PINs are resistant to online attacks than 4-digit PINs. Munyendo et al. [37] also conclude from their user study and analysis that upgrading from 4-digit to 6-digit PINs provides limited security benefit while reducing usability. There have been several proposals on improving PIN security [3, 12, 28, 33, 39, 43]. In particular, applying blocklists that block common PINs has shown promise. Bonneau et al. [11] suggest a blocklist of 100 most common PINs which is optimal in their study. Kim and Huh [28] demonstrate that restricting 200 commonly used PINs is beneficial. Markert et al. [32] conduct a comprehensive user study on 4-digit and 6-digit PINs and suggest that a blocklist that contains about 10% of the PIN space may best balance usability and security.

2 Background And Notation

Given a password distribution $\mathcal{P}$ we define p_{pwd} be the probability of password pwd in $\mathcal{P}$. We denote $s \leftarrow \mathcal{P}$ to be a random sample from the distribution, and let $D = (s_1, \ldots, s_N) \leftarrow \mathcal{P}^N$ denote a dataset of N iid password samples from $\mathcal{P}$. Fixing a dataset D we define $f_i^{D,freq}$ to be the frequency of the ith most *frequent* password in D and let $F_i^{D,freq} = \sum_{j=1}^{i} f_j^{D,freq}$ denote the cumulative frequency of the top i passwords in D. By definition $f_i^{D,freq} \geq f_j^{D,freq}$ for any $i \leq j$. Define pwd_i to be the ith most *probable* password in the distribution $\mathcal{P}$, i.e., $p_{pwd_i} \geq p_{pwd_j}$ for any $i > j$. Then let $f_i^{D,prob}$ be the frequency of the ith most *probable* password (i.e., pwd_i) in D and $F_i^{D,prob} = \sum_{j=1}^{i} f_j^{D,prob}$. Note that the ith most *probable* password in distribution $\mathcal{P}$ is not necessary the ith most *frequent* password in the sample set. By definition we have $f_i^{D,prob} \leq f_i^{D,freq}$ and $F_i^{D,freq} \geq F_i^{D,prob}$ for any $i > 0$. We will use $\texttt{bpdf}(i,N,p) := \binom{N}{i} p^i (1-p)^{N-i}$ to denote the binomial probability density function, i.e. the probability that we draw N samples from distribution $\mathcal{P}$ and the password with probability p is sampled exactly i times. We will also use $\texttt{bcdf}(i,N,p) := \sum_{j \leq i} \texttt{bpdf}(j,N,p)$ to denote the binomial cumulative cumulative distribution function, i.e. the probability that the password with probability p is sampled at most i times among N samples we draw.

Attack Model. We consider an attacker who knows the password distribution $\mathcal{P}$ but not the particular user passwords in the dataset D sampled from $\mathcal{P}$. For each password pwd the attacker knows its probability p_{pwd} in $\mathcal{P}$. For each sample s_i in D, the attacker is given G guesses to crack the user' password s_i. The optimal strategy is to check the passwords in decreasing order of the probabilities. Let $\lambda_{D,G} := \sum_{i=1}^{G} f_i^{D,prob}/N$ to be the percentage of password samples in D that would be cracked in G guesses. Then $\lambda_G := \sum_{i=1}^{G} p_i = \mathbb{E}(\lambda_{D,G})$ is the expected value of $\lambda_{D,G}$ where we denote p_i to be the probability of the ith most probable password pwd_i in the distribution (i.e., $p_1 \geq p_2 \geq \cdots$) and the randomness is taken over the sample set D. Blocki and Liu [8] prove that λ_G and $\lambda_{D,G}$ are close with high probability and bounding them are nearly equivalent problems as shown in Theorem 1. In this paper, we will focus on upper and lower bounding λ_G. As a consequence of Theorem 1 we can immediately get the corresponding bound of $\lambda_{D,G}$.

Theorem 1. *[8] For any guessing number $G \geq 0$ and any $0 \leq \varepsilon \leq 1$ we have:*

$$\Pr[\lambda_{D,G} \leq \lambda_G + \varepsilon] \geq 1 - \exp(-2N\varepsilon^2), \textit{ and}$$
$$\Pr[\lambda_{D,G} \geq \lambda_G - \varepsilon] \geq 1 - \exp(-2N\varepsilon^2)$$

where the randomness is taken over the sample set $D \leftarrow \mathcal{P}^N$ of size N.

Password Cracking Models. We consider a password cracking model M that outputs a list of password guesses.

There are lots of password cracking models, such as rule-based password cracking softwares, probabilistic password models, and dictionaries of previous cracked or breached passwords. Given a password cracking model M, let $pwd_{i,M}$ be the ith guess outputted by the model M. We define $f_i^{D,M}$ to denote the frequency of this password in the dataset D and let $F_i^{D,M} = \sum_{j=1}^{i} f_j^{D,M}$. Note that the predicted password distribution outputted by a password cracking model M (i.e., $pwd_{i,M}$ for $i > 0$) may not be the ith most popular password in the real password distribution $\mathcal{P}$. Let $\texttt{Dict}_{M,G}$ be the list of the top G guesses outputted by model M. We define $\lambda_{M,G} := \Pr_{pwd \leftarrow \mathcal{P}}[pwd \in \texttt{Dict}_{M,G}]$ to be the probability that a random password from distribution $\mathcal{P}$ is in the top G guesses outputted by model M. Note that $\lambda_{M,G} = \sum_{j:pwd_j \in \texttt{Dict}_{M,G}} p_j \leq \sum_{i=1}^{G} p_i = \lambda_G$, where pwd_i is the ith most probable password in $\mathcal{P}$.

3 New Techniques for Bounding Password Guessing Curves

In this section, we propose a new statistical techniques upper and lower bound the attackers guessing curve (λ_G) given N samples $D = (s_1, \ldots, s_n) \leftarrow \mathcal{P}^N$ from our unknown password distribution. In contrast to prior statistical techniques [8] our upper/lower bounds directly used the binomial probability density function. Empirical analysis demonstrates that our new techniques generate tighter bounds on λ_G for small to moderate size guessing budgets — the linear programming of [8] still generates the best upper bound when the guessing budget is very large.

For the upper bound we rely on properties of the binomial cumulative distribution function to argue that, except with probability δ, we have $\lambda_G \leq UB_\delta(F_G^{D,freq})$ where $UB_\delta(F) := \min_p\{p : \sum_{j=0}^{F} \texttt{bpdf}(j,N,p) \leq \delta\}$ denotes the minimum possible probability value p such that, when we flip a p-biased coin N times, the probability of observing at most F heads is at most δ. Intuitively, we would like to use $UB_\delta(F_G^{D,prob})$ as our upper bound where $F_G^{D,prob}$ denotes the number of times that a user picked one of the top G most likely passwords in the distribution. While we cannot directly compute $F_G^{D,prob}$ as we don't know for certain which passwords in the distribution are most likely, we do know that $F_G^{D,prob} \leq F_G^{D,freq}$ — a quantity we can determine by finding the *most frequent* passwords in the dataset D. Since the function $UB_\delta(F)$ is (strictly) monotonically increasing with F we can use $UB_\delta(F_G^{D,freq})$ as our upper bound. Similarly, our lower bound relies on the observation that the function $LB_\delta(F) := \max_p\{p : \sum_{j=F}^{N} \texttt{bpdf}(j,N,p) \leq \delta\}$ is (strictly) monotonically increasing in F.

We start by observing that the function $f(p) := \texttt{bcdf}(F,N,p)$ is (strictly) monotonically decreasing in p. Claim 1 is intuitive because if $p_2 > p_1$ then an item with probability p_2 would be sampled *more* frequently than an item with probability p_1 — see Appendix C for a formal proof.

Claim 1. *Given any integers $N > 0$ and $0 \leq F < N$ and any $0 \leq p_1 < p_2 \leq N$ we have $\texttt{bcdf}(F,N,p_1) > \texttt{bcdf}(F,N,p_2)$.*

As an immediate corollary we have the following claim since $\sum_{i=F}^{N} \texttt{bpdf}(j,N,p) = 1 - \texttt{bcdf}(F-1,N,p)$:

Claim 2. *Given any integer $N > 0$ and any $0 < F \leq N$, $f(p) = \sum_{i=F}^{N} \texttt{bpdf}(j,N,p)$ is strictly monotonically increasing for $p \in [0,1]$.*

3.1 Upper Bound

We define $UB_\delta(F) = \min_p\{p : \sum_{j=0}^{F} \texttt{bpdf}(j,N,p) \leq \delta\}$ as the largest probability mass p such that it is likely (with at least δ probability) to sample no more than F times in N total samples. Here we consider $\delta \in [0,1]$ as a constant parameter (e.g. $\delta = 0.01$). The monotonicity property proved in Claim 1 guarantees that the minimum of p satisfying the condition exists for any $0 \leq F < N$ and $\sum_{j=0}^{F} \texttt{bpdf}(j,N,p) > \delta$ for all $p < UB_\delta(F)$. For $F = N$ when such p doesn't exist, we define $UB_\delta(N) = 1$. Claim 3 states that $UB_\delta(F)$ is strictly monotonically increasing — see Appendix C for the proof.

Claim 3. *$UB_\delta(x_1) < UB_\delta(x_2)$ for any $0 \leq x_1 < x_2 \leq N$.*

Recall that λ_G represents the probability mass of the top G most probable passwords in the distribution and these passwords appear $F_G^{D,prob}$ times in total in the sample set D. Intuitively, we can argue that $UB_\delta(F_G^{D,prob})$ is an upper bound of λ_G with high probability. Observe that if $\lambda_G > UB_\delta(F_G^{D,prob})$ then, by definition of UB_δ, the probability of sampling the top G most probable passwords at most $F_G^{D,prob}$ times would have been smaller than δ. Unfortunately, the upper bound $UB_\delta(F_G^{D,prob})$ is not computable as the term $F_G^{D,prob}$ depends on the unknown password distribution i.e., we don't know which G passwords in the support of $\mathcal{P}$ are the most probable. Given the fact $F_G^{D,freq} \geq F_G^{D,prob}$, we can further argue that $UB_\delta(F_G^{D,freq})$ is an easily computable upper bound of λ_G i.e., we can easily find the G most frequent passwords in the sampled set $D \leftarrow \mathcal{P}^N$. Theorem 2 formally proves that $\lambda_G \leq UB_\delta(F_G^{D,freq})$ with probability at least $1 - \delta$.

Theorem 2. *For any $0 \leq \delta \leq 1$, $\Pr[\lambda_G \leq \lambda^{UB}(\delta,D,G)] \geq 1 - \delta$, where $\lambda^{UB}(\delta,D,G) := UB_\delta(F_G^{D,freq})$ and the randomness is taken over the password sample set D with N random samples from a password distribution.*

Proof. For the proof it will be useful to first define $F_\delta^{UB}(p) = \max_F\{F : \sum_{j=0}^{F} \texttt{bpdf}(j,N,p) \leq \delta\}$ for $p^* \leq p \leq 1$ where $p^* = \arg\min_p\{\texttt{bpdf}(0,N,p) \leq \delta\}$. For $0 \leq p < p^*$ where no F satisfies the condition, we define $F_\delta^{UB}(p) = -1$. Denote $F' =$

$F_\delta^{UB}(\lambda_G)$. Notice that F' cannot be N as $\delta < 1$, otherwise the condition $\sum_{j=0}^{F} \texttt{bpdf}(j,N,p) \le \delta$ doesn't hold. Then we have

$$\sum_{j=0}^{F'} \texttt{bpdf}(j,N,\lambda_G) \le \delta \tag{1}$$

whenever $\lambda_G \ge p^*$ and

$$\sum_{j=0}^{F'+1} \texttt{bpdf}(j,N,\lambda_G) > \delta \tag{2}$$

for all $\lambda_G \ge 0$. Equation 2 is true as F' is defined to be the maximum value that satisfies $\sum_{j=0}^{F'} \texttt{bpdf}(j,N,\lambda_G) \le \delta$.

Next we denote $p' = UB_\delta(F'+1)$. Now for $F' < N-1$ we have

$$\sum_{j=0}^{F'+1} \texttt{bpdf}(j,N,p') \le \delta \le \sum_{j=0}^{F'+1} \texttt{bpdf}(j,N,\lambda_G)\ ,$$

where the first inequality follows from the definition of $p' = UB_\delta(F'+1)$ and the second inequality follows from Equation 2. Note that for $F' = N-1$, we have

$$\sum_{j=0}^{F'+1} \texttt{bpdf}(j,N,p') = 1 = \sum_{j=0}^{F'+1} \texttt{bpdf}(j,N,\lambda_G)\ .$$

Recall that Claim 2 proves that function $f(p) = \sum_0^{F'+1} \texttt{bpdf}(j,N,p)$ is strictly monotonically decreasing for $F'+1 < N$ and $p' = 1$ when $F'+1 = N$, so $p' \ge \lambda_G$.

Since Claim 3 proves that $UB_\delta(F)$ is strictly monotonically increasing, we observe that $p' = UB_\delta(F'+1) > UB_\delta(F_G^{D,freq})$ if and only if $F'+1 > F_G^{D,freq}$. Then we have:

$$\begin{aligned}\Pr[\lambda_G > UB_\delta(F_G^{D,freq})] &\le \Pr[p' > UB_\delta(F_G^{D,freq})]\\ = \Pr[F'+1 > F_G^{D,freq}] &\le \Pr[F'+1 > F_G^{D,prob}]\end{aligned}$$

where the second inequality holds due to the fact $F_G^{D,freq} \ge F_G^{D,prob}$. Also note that the definition of F' depends only on the password distribution $\mathcal{P}$, not on the samples in D. Note that if $F' = -1$ we have $\Pr[F'+1 > F_G^{D,prob}] = 0$; if $F' \ge 0$, i.e., by definition $\lambda_G \ge p^*$, then we can apply Equation 1 and have $\Pr[F'+1 > F_G^{D,prob}] = \sum_{j=0}^{F'} \texttt{bpdf}(j,N,\lambda_G) \le \delta$ by the definition of $F' = F_\delta^{UB}(\lambda_G)$. Therefore, $\Pr[\lambda_G > UB_\delta(F_G^{D,freq})] < \delta$.

□

3.2 Lower Bound

Similar to the upper bound of λ_G, we define $LB_\delta(F,N) = \max\limits_p \{p : \sum_{j=F}^{N} \texttt{bpdf}(j,N,p)\} \le \delta\}$ as the smallest probability mass p of a group of passwords such that it is likely to sample these passwords at least F times in total in N samples. We omit N and use $LB_\delta(F)$ for simplicity when it is clear what N is. The monotonicity property proved in Claim 1 guarantees that the maximum of p satisfying the condition exists for any $0 < F \le N$ and $\sum_{j=F}^{N} \texttt{bpdf}(j,N,p) > \delta$ for all $p > LB_\delta(F)$. As an edge case we define $LB_\delta(0) = 0$ since p may not exist when $F = 0$. Claim 4 shows that the function $LB_\delta(F)$ strictly monotonically increasing in F.

Claim 4. *$LB_\delta(x_0) < LB_\delta(x_1)$ for all $0 \le x_0 < x_1 \le N$*

Proof. First of all note that $LB_\delta(F) > 0 = LB_\delta(0)$ for all $F > 0$ and $LB_\delta(0) = 0$. Consider any $0 < x_1 < x_2 \le N$ and any fixed $0 \le p \le 1$, we have $\sum_{j=x_1}^{N} \texttt{bpdf}(j,N,p) > \sum_{j=x_2}^{N} \texttt{bpdf}(j,N,p)$. Let $p_1 = LB_\delta(x_1)$ and $p_2 = LB_\delta(x_2)$. Suppose for contradiction $p_1 \ge p_2$. Then we have $\sum_{j=x_2}^{N} \texttt{bpdf}(j,N,p_1) < \sum_{j=x_1}^{N} \texttt{bpdf}(j,N,p_1) \le \delta$. If $p_1 > p_2$ then this directly contradicts the choice of p_2 as the maximum value satisfying $\sum_{j=x_2}^{N} \texttt{bpdf}(j,N,p_2) \le \delta$. If $p_1 = p_2$ then we have $\sum_{j=x_2}^{N} \texttt{bpdf}(j,N,p_2) < \delta$. Because since the function $f(p) = \sum_{j=x_2}^{N} \texttt{bpdf}(j,N,p)$ is continuous we can find some small $\varepsilon > 0$ such that $\sum_{j=x_2}^{N} \texttt{bpdf}(j,N,p_2+\varepsilon) \le \delta$. Once again this contradicts our choice of p_2 as the maximum value satisfying $\sum_{j=x_2}^{N} \texttt{bpdf}(j,N,p_2) \le \delta$. □

Intuitively, $LB_\delta(F_G^{D,prob})$ is an lower bound of λ_G with high probability. If $\lambda_G < LB_\delta(F_G^{D,prob})$ then, by definition of LB_δ, the probability of sampling the top G most probable passwords at least $F_G^{D,prob}$ times would have been smaller than δ. However, we have the same problem that we cannot compute $F_G^{D,prob}$ since we do not know for certain which passwords in the support of $\mathcal{P}$ are the most probability. However, if we fix a password cracking model M a priori then we can argue that $LB_\delta(F_G^{D,M})$ lower bounds $\lambda_{M,G}$ with high probability — recall that $\lambda_{M,G}$ denotes the probability that an attacker cracks a random password from $\mathcal{P}$ within G guesses. Since the password cracking model M cannot be better than the perfect knowledge attacker we have $\lambda_G \ge \lambda_{M,G}$. Thus, λ_G can also be lower bounded by $LB_\delta(F_G^{D,M})$, as shown in Theorem 3. We defer the proof of Theorem 3 to Appendix C as the proof is similar to Theorem 2.

Theorem 3. *Given a password cracking model M, for any $0 \le \delta \le 1$, $\Pr[\lambda_G \ge \lambda^{LB}(\delta,M,D,G)] \ge 1-\delta$, where $\lambda^{LB}(\delta,M,D,G) := LB_\delta(F_G^{D,M},N)$ and the randomness is taken over the sample set $D \leftarrow \mathcal{P}^N$.*

Given a password sample set D, we can partition the set into training set $D_1 = \{s_1,\ldots,s_{N-d}\}$ and test set $D_2 = \{s_{N-d+1},\ldots,s_N\}$. We can use the training set to train a model M and then apply Theorem 3 with the test set D_2. A straightforward, but powerful, way to use D_1 is to define $\texttt{Dict}_G^{D_1}$ as a dictionary of the top G most frequent passwords in D_1 and let $\texttt{Cracked}(D_2,\texttt{Dict}_G^{D_1})$ be the number of samples in D_2 that are cracked by making guesses in $\texttt{Dict}_G^{D_1}$. Applying Theorem 3 we obtain the following corollary:

Corollary 4. *For any sample set D_1 and any $0 \leq \delta \leq 1$ we have:*

$$\Pr[\lambda_G \geq LB_\delta(\texttt{Cracked}(D_2, \texttt{Dict}_G^{D_1}), |D_2|)] \geq 1 - \delta$$

where the randomness is taken over selection of the second sample set $D_2 \leftarrow \mathcal{P}^N$ of size N.

Note that when applying Corollary 4 the that N denotes the number of samples in the test set D_2 and not the total number of samples in D.

4 Theoretical Analysis of Empirical Password Distribution

The empirical password distribution is a heuristic approach to password security analysis. Blocki and Liu [8] previously used the empirical guessing curve $F_G^{D,freq}/N$ to *upper bound* λ_G. However, the empirical distribution often yields a pessimistic *overestimate* of the fraction of passwords that an attacker can crack within G guesses. For example, it is a guarantee that $F_G^{D,freq} = N$ whenever $G \geq N$ as there are *at most* N passwords in the support of the empirical distribution where N denotes the number of samples in the password dataset D. Empirical analysis from [8] suggests that the approximation $\lambda_G \approx F_G^{D,freq}/N$ is reasonable when the guessing budget in small, but this was simply an empirical observation and it was unclear whether or not the empirical distribution could be used to lower bound λ_G. In this section we show how to use the empirical distribution to *lower bound* λ_G. Our theoretical analysis demonstrates that the lower bound will be tight as long as $G = o(N/\log N)$.

Intuitively, our proof focuses on upper bounding the probability that some password in the dataset is significantly oversampled. We expect that a password *pwd* will be sampled $p_{pwd}N$ times in our dataset, but the actual number of samples may be larger or smaller. Fixing suitable constants $\alpha, \delta > 0$ we say that *pwd* is significantly over-sampled if we sample the password $\alpha p_{pwd}N + \delta \log N$ more times than we expect. We argue that (whp) for all possible passwords *pwd* we have $f_{pwd}^{D,freq} < (1+\alpha)p_{pwd}N + \delta \log N$ i.e., whp no password is significantly over-sampled. The constant $\alpha > 0$ controls the multiplicative error and can be arbitrarily close to 0, but there is a trade-off as the parameter δ which controls additive error increases with α^{-2}. Assuming that $f_{pwd}^{D,freq} < (1+\alpha)p_{pwd}N + \delta \log N$ for all passwords *pwd* it quickly follows that $F_G^{D,freq} \leq (1+\alpha)\lambda_G + \delta G \log N$. Note that as long as the guessing budget is bounded $G = o(N/\log N)$ we can guarantee that the additive error term $\delta G \log N = o(N)$ is small.

4.1 Significant Oversampling is Unlikely

When analysis the probability that any particular password *pwd* is significantly oversampled it is necessary to partition passwords based on their likelihood. Define $B_i = \{pwd : 2^{-i-1} < p_{pwd} \leq 2^{-i}\}$ to be the set of all passwords with probability in the interval $(2^{-i-1}, 2^{-i}]$. Fixing the parameters α and δ let $\texttt{BAD}_i$ be the bad event that there exist a password $pwd \in B_i$ such that this password's frequency $f_{pwd}^{D,freq} > (1+\alpha)p_{pwd}N + \delta \log N$ is unusually large. To upper bound $\Pr[\cup_i \texttt{BAD}_i]$ we consider two cases $p_{pwd} \geq \frac{4\log N}{\alpha^2 N}$ and $p_{pwd} < \frac{4\log N}{\alpha^2 N}$.

Intuitively, if p_{pwd} is large enough then one can use Chernoff's Bounds to argue that (whp) the multiplicative error will be small. This allows us to union bound over all passwords *pwd* in any bucket B_i with $i \leq \log(\frac{\alpha^2 N}{4\log N})$.

Lemma 1. *For any parameters $\alpha > 0$ and $\delta \geq \frac{8e}{\alpha^2}$ we have*

$$\Pr\left[\cup_{i<\log(\frac{\alpha^2 N}{4\log N})} \texttt{BAD}_i\right] \leq \frac{\alpha^2}{4\log N} \cdot N^{\frac{-2}{(1+\frac{\alpha}{3})\ln 2}+1},$$

where the randomness is taken over the selection of $D \leftarrow \mathcal{P}^N$.

Proof. For any parameter $\alpha > 0$ and any password probability $p_{pwd} \geq \frac{4\log N}{\alpha^2 N}$, applying Chernoff's Bound we have:

$$\begin{aligned}
&\Pr[f_{pwd}^{D,freq} \leq (1+\alpha)p_{pwd}N] \\
\geq& 1 - \exp\left(\frac{-\alpha^2}{2(1+\frac{\alpha}{3})} N p_{pwd}\right) \\
\geq& 1 - \exp\left(\frac{-2}{(1+\frac{\alpha}{3})} \log N\right) \\
=& 1 - N^{\frac{-2}{(1+\frac{\alpha}{3})\ln 2}}. \qquad (3)
\end{aligned}$$

We note that there are at most $\frac{\alpha^2 N}{4\log N}$ passwords *pwd* s.t. $p_{pwd} \geq \frac{\alpha^2 N}{4\log N}$ and thus there are at most $\frac{\alpha^2 N}{4\log N}$ passwords in the union $U = \bigcup_{i\leq\log(\frac{\alpha^2 N}{4\log N})} B_i$. Applying Equation 3 for each password $pwd \in U$ we have $\Pr[f_{pwd}^{D,freq} > (1+\alpha)p_{pwd}N + \delta \log N] \leq N^{\frac{-2}{(1+\frac{\alpha}{3})\ln 2}}$. Union bounding over all $\frac{\alpha^2 N}{4\log N}$ passwords we have

$$\begin{aligned}
&\Pr\left[\cup_{i<\log(\frac{\alpha^2 N}{4\log N})} \texttt{BAD}_i\right] \\
=& \Pr\left[\exists pwd \in U \texttt{ s.t. } f_{pwd}^D > (1+\alpha)pN + \delta \log N\right] \\
\leq& \frac{\alpha^2 N}{4\log N} \cdot N^{\frac{-2}{(1+\frac{\alpha}{3})\ln 2}} \\
=& \frac{\alpha^2}{4\log N} \cdot N^{\frac{-2}{(1+\frac{\alpha}{3})\ln 2}+1}.
\end{aligned}$$

□

When p_{pwd} is smaller we cannot use Chernoff Bounds to bound the multiplicative error. Instead we adopt a "Balls and Bins" analysis to upper bound the additive error and show that (whp) $f_{pwd}^{D,freq} \leq \delta \log N$. Lemma 2 deals with passwords whose probability value is small i.e., passwords in buckets B_i with $i \geq \log\left(\frac{\alpha^2 N}{4\log N}\right)$. Intuitively, Lemma 2 follows from the basic observation that $\Pr[f_{pwd}^{D,freq} \geq \delta \log N] \leq \binom{N}{\delta \log N} p_{pwd}^{\delta \log N}$. Union bounding over all passwords $pwd \in B_i$ in bucket i we have $\Pr[\mathtt{BAD}_i] \leq \binom{N}{\delta \log N} 2^{i+1-i\delta\log N}$. Since the probability decays exponentially with i we are able to union bound over all bad events $\mathtt{BAD}_i$ with $i \geq \log\left(\frac{\alpha^2 N}{4\log N}\right)$.

Lemma 2. *For any parameters $\alpha > 0$ and $\delta \geq \frac{8e}{\alpha^2}$ we have* $\Pr[\cup_{i \geq \log(\frac{\alpha^2 N}{4\log N})} \mathtt{BAD}_i] \leq \frac{1}{\sqrt{2\pi\delta N}\log N(1-2^{1-\delta\log N})} \cdot \frac{\alpha^2}{2N^{\delta-1}}$.

The proof of Lemma 2 is in Appendix C. Combining Lemma 1 and 2 we can argue that (whp) there are no significantly over-sampled passwords i.e., the event $\cup_i \mathtt{BAD}_i$ does not occur.

Theorem 5. *For any parameters $\alpha > 0$ and $\delta \geq \frac{8e}{\alpha^2}$,* $\Pr[\cup_i \mathtt{BAD}_i] \leq \frac{\alpha^2}{4\log N} \cdot N^{\frac{-2}{(1+\frac{\alpha}{3})\ln 2}+1} + \frac{1}{\sqrt{2\pi\delta N}\log N(1-2^{1-\delta\log N})} \cdot \frac{\alpha^2}{2N^{\delta-1}}$.

Proof. By Union bounds Lemma 1 and 2 we have

$$\Pr[\cup_i \mathtt{BAD}_i] \leq \Pr[\cup_{i<\log(\frac{\alpha^2 N}{4\log N})} \mathtt{BAD}_i] + \sum_{i \geq \log(\frac{\alpha^2 N}{4\log N})} \Pr[\mathtt{BAD}_i]$$

$$\leq \frac{\alpha^2}{4\log N} \cdot N^{\frac{-2}{(1+\frac{\alpha}{3})\ln 2}+1} + \frac{1}{\sqrt{2\pi\delta N}\log N(1-2^{1-\delta\log N})} \cdot \frac{\alpha^2}{2N^{\delta-1}}$$

Therefore, the theorem is proved. □

4.2 Lower Bounding the Guessing Curve

Let λ_G^{freq} denote the cumulative probability mass of the top G most frequent passwords in D. By definition we have $\lambda_G \geq \lambda_G^{freq}$. Assuming the bad event $\cup_i \mathtt{BAD}_i$ does not occur the upper bound $f_{pwd}^{D,freq} < (1+\alpha)p_{pwd}N + \delta \log N$ holds for all passwords pwd. Thus, summing up the frequencies of the top G most frequent passwords we have $F_G^{D,freq} \leq (1+\alpha)N\lambda_G^{freq} + \delta G \log N \leq (1+\alpha)N\lambda_G + \delta G \log N$ with probability at least $1 - \Pr[\cup_i \mathtt{BAD}_i]$ where the probability of the bad event is upper bounded by Theorem 5. We formally state this observation as Theorem 6 below.

Theorem 6. *Given a password sample set D with N samples, for any $G > 0$, $\alpha > 0$ and $\delta \geq \frac{8e}{\alpha^2}$, we have:*

$$\Pr[F_G^{D,freq} \leq (1+\alpha)N\lambda_G + G\delta\log N] \geq 1 - \beta$$

where $\beta = \frac{1}{\log N} \cdot N^{\frac{-\alpha^2}{2(1+\frac{\alpha}{3})}\log e+1} + \frac{1}{\sqrt{2\pi\delta N}\log N(1-2^{1-\delta\log N})} \cdot \frac{\alpha^2}{2N^{\delta-1}}$.

Discussion. Intuitively, the above theorem is telling us that $F_G^{D,freq}/N$ is a good approximation of λ_G as long as $G \ll N/\log N$. In particular, if $G = o(N/\log N)$ then the theorem tells us that with high probability we have

$$\lambda_G \geq \frac{F_G^{D,freq}}{(1+\alpha)N} - \frac{G\delta\log N}{(1+\alpha)N} = \frac{F_G^{D,freq}}{(1+\alpha)N} - o(1) \ .$$

We already know from Blocki and Liu [8] that $\lambda_G \leq F_G^{D,freq} + \varepsilon$ with high probability. Thus, with high probability we have

$$\lambda_G - \varepsilon \leq \frac{F_G^{D,freq}}{N} \leq (1+\alpha)\lambda_G + o(1) \ .$$

Intuitively, since we can take α to be an arbitrarily small constant (e.g., $\alpha = 0.01$) this means that (whp) the empirical distribution $F_G^{D,freq}/N$ gives us a good approximation of λ_G as long as our guessing budget $G = o(N/\log N)$ is not too large. The empirical password distribution has frequently been used as a heuristic in password security analysis e.g., see [2, 4–6, 10, 15, 16]. Our observation provides rigorous justification for this heuristic as long as the guessing number $G = o(N/\log N)$ is smaller.

5 Experiments

In this section, we apply the statistical techniques to analyze several password and PIN datasets with both small and large sample sizes.

5.1 Datasets

In the empirical analysis, we use eight empirical password datasets (000webhost, Neopets, Battlefield Heroes, Brazzers, Clixsense, CSDN, Yahoo!, and RockYou) and two 4-digit PIN datasets (Amitay [1], and a dataset collected in a user study of Markert et al. [32]). Table 1 provides the name and the size of each dataset.

For the first six password datasets we use the sanitized versions prepared by Liu et al. [29]. With the exception of Yahoo! all of the password datasets are the result of a data breach. The Yahoo! password dataset [5, 10] is a differentially private frequency list and does not include plaintext passwords. We applied the same differentially private algorithm of Blocki et al. [5] to generate a differentially private version of the Amitay dataset. While these anonymized datasets do not include user passwords, it is still possible to apply the statistical techniques from our paper (and from [8]) to upper and lower bound λ_G. Blocki et al. [5] shows that the additional noise added the preserve differential privacy introduces minimal L1 error $O(1/\sqrt{N})$.

Amitay PIN dataset was collected in 2011 from an iOS application "Big Brother Camera Security" developed by Daniel Amitay [1]. This app mimicked a setup screen and

Table 1: Password/PIN Datasets

Dataset (*D*)	**# Samples** (*N*)
000webhost [22]	15268903
Neopets [17]	68345757
Battlefield Heroes [44]	541016
Brazzers [18]	925614
Clixsense [23]	2222529
CSDN [47]	6428449
Yahoo! [5]	69301337
RockYou [19]	32603388
Amitay (with DP) [1, 5]	204445
Amitay (original) [1]	204432
User Study First Choice [32]	851

a lock screen that are nearly identical to actual iPhone passcode setup/lock, and allowed users to set up 4-digit PINs. In total 204432 4-digit PINs were anonymously collected and released publicly by Amitay. This is a large dataset with realistic PIN data, which we apply in our empirical analysis on PINs.

While the user study dataset gathered by Markert et al. [32] is smaller ($N = 851$ samples) it is particularly useful for analyzing PIN blocklists. The PIN dataset was the result of several different studies. For example, in one study the top 2740 4-digit pins from the Amitay dataset [1] were blocked. If a user selected a blocked PIN number they were simply informed that this particular PIN number was blocked and asked to try again. The dataset includes the list of *all* PIN numbers selected by each user before they succeeded. While Markert et al. [32] conducted experiments with several different blocklists, they combined these smaller PIN datasets into one larger PIN dataset by considering the “first choice" of each user. This yields a dataset with PINS from $N = 851$ different users.

Limitations. Similar to Blocki and Liu [8] we make the assumption that each breached dataset represents N iid samples from some unknown password distribution. We cannot absolutely guarantee that the assumption holds for the datasets that we analyze. However, the linear program from [8] is able to detect blatant violations of this assumption e.g., the dataset contains many duplicate accounts. Other factors such as the age of a particular dataset and the demographics of the particular user base may impact the ecological validity of any particular conclusions that can be drawn from our statistical analysis. However, we stress our statistical bounds can be applied to any password dataset regardless of dataset age or other demographic factors as long as the passwords in the dataset represent independent samples from an unknown distribution. Finally, some of our analysis of PIN/password blocklists utilizes a heuristic assumption called the normalized probabilities assumption. We discuss the ecological validity of this assumption in Section 5.3.4.

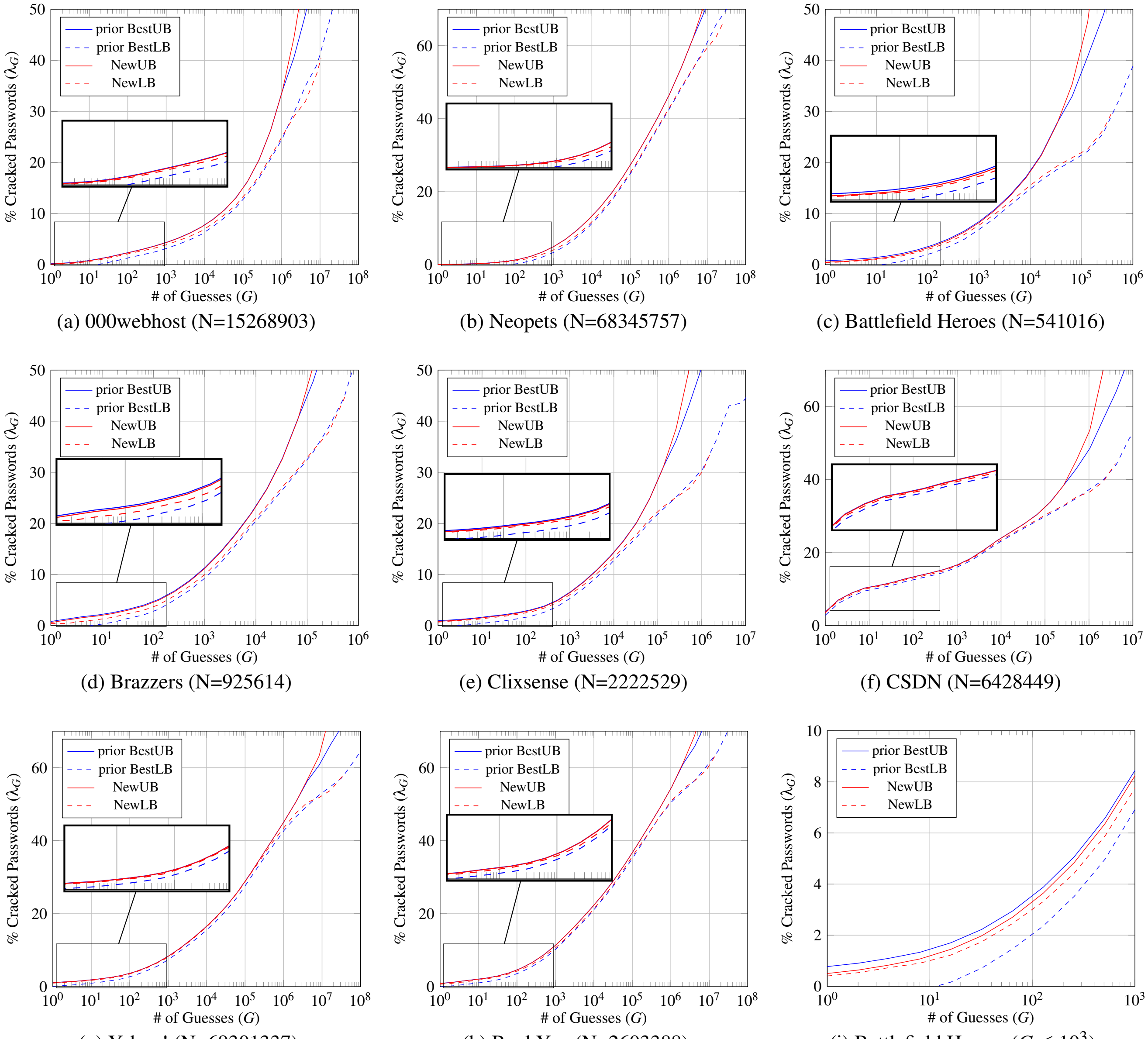
5.2 Comparing Our Bounds with Prior Work

In this section, we evaluate the performance of our statistical upper/lower bounds and compare with previous statistical techniques in Blocki and Liu [8]. In particular, Blocki and Liu presented two upper bounds and three lower bounds that we compare with: an upper bound generated by empirical distribution (denote as `FrequencyUB`), an upper bound generated by linear programming (denoted as `LPUB`), a lower bound generated by the dataset itself (denoted as `SamplingLB`), a lower bound that extends `SamplingLB` using an RNN password cracking model (denoted as `ExtendedLB`), and a lower bound generated by linear programming (denoted as `LPLB`). We then let $\mathtt{priorBestUB} = \min\{\mathtt{FrequencyUB}, \mathtt{LPUB}\}$ be the best upper bound in prior work, and also denote the best lower bound in prior work as $\mathtt{priorBestLB} = \{\mathtt{SamplingLB}, \mathtt{ExtendedLB}, \mathtt{LPLB}\}$. We also denote `newUB` to be our new upper bound in Theorem 2, $\mathtt{newLB}^M$ to be our new lower bound in Theorem 3 using model M, and $\mathtt{newLB}^{samp}$ to be our new lower bound in Corollary 4. In the empirical analysis we guarantee that each individual bound holds with at least 99% confidence. Our goal is to compare the techniques for a wide range of sample sizes N and guessing budgets G.

5.2.1 Password Datasets

We start by evaluating our bounds on password datasets. For each dataset S, we generate upper and lower bounds on λ_G using our results in Section 3 and compare our bounds with existing bounds in Blocki and Liu [8]. For fair comparison we use the same parameter settings (sampling parameter $d = 25000$, at least 99% confidence) in Blocki and Liu [8] for all bounds. We plot our new bounds, the existing best upper bound $\min\{\mathtt{FrequencyUB}(S,G), \mathtt{LPUB}(S,G)\}$ and the existing best lower bound $\max\{\mathtt{LPLB}(S,G), \mathtt{SamplingLB}(S,G), \mathtt{ExtendedLB}(S,G)\}$ on eight password datasets (000webhost, Neopets, Battlefield Heros, Brazzers, Clixsense, and CSDN, Yahoo!, and RockYou) in Figure 1. Upper (resp. lower) bounds are depicted using solid (resp. dashed) lines.

Figure 1 shows that our new techniques significantly tighten the bounds, allowing defenders to estimate the percentage of passwords cracked by an attacker accurately with high confidence when the guessing number is small e.g., see the zoomed in plot in Figure 1i. With the new bounds we can meaningfully characterize the performance of an online password spraying attacker. For example, when the guessing number $G = 8$ existing bounds show that $0 \leq \lambda_G \leq 1.33\%$ while our new bounds are much tighter proving that $0.90\% \leq \lambda_G \leq 1.07\%$ for Battlefield Hero dataset; for the Yahoo! dataset when $G = 1.31 \times 10^5$ our new techniques provide very tight upper/lower bounds as $30.563\% \leq \lambda_G \leq 30.662\%$ while the gap between existing bounds $29.463\% \leq \lambda_G \leq 30.675\%$ is 12.3 times larger. In fact, our new bounds outperforms the existing bounds `FrequencyUB` and `SamplingLB` for all guess-



(a) 000webhost (N=15268903)

(b) Neopets (N=68345757)

(c) Battlefield Heroes (N=541016)

(d) Brazzers (N=925614)

(e) Clixsense (N=2222529)

(f) CSDN (N=6428449)

(g) Yahoo! (N=69301337)

(h) RockYou (N=2603388)

(i) Battlefield Heroes ($G \leq 10^3$)

Figure 1: 000webhost, Neopets, Battlefield Heroes, Brazzers, Clixsense, CSDN, Yahoo!, and RockYou Guessing Curves

ing budgets G on all eight password datasets. When guessing budget G grows very large the linear programming approach of [8] still yields the tightest bounds.

5.2.2 PIN datasets

We then evaluate the performances of our bounds as well as existing bounds on PIN datsets with different sample sizes.

Large PIN Datasets. We apply prior bounds (`FrequencyUB` and `SamplingLB`) and our new bounds (`newUB` and $\texttt{newLB}^{samp}$ with sampling parameter $d = 25000$) on Amitay 4-digit PIN dataset and plot the guessing curves in Figure 2a. Interestingly, we get tight upper and lower bounds on guessing curve λ_G over the entire guessing range $1 \leq G \leq 10^4$. This is in contrast to password datasets where the best upper/lower bounds start to diverge as the guessing budget grows large. The reason that it is possible to obtain tight bounds over the entire guessing range is that the number of samples $N = 204,445$ exceeds the support of the distribution since there are at most 10^4 possible 4-digit PIN numbers (98.8% of all possible 4-digit PIN numbers appear *at least* once in the dataset). Our new bounds generally yield a slight improvement over [8], but this improvement is difficult to see on the plot as both upper/lower bounds yield reasonably tight bounds. We compare our upper/lower bounds with the uniform distribution over PIN numbers i.e., each particular PIN is chosen with probability 10^{-4}. The large gap between our lower bound and the uniform distribution guessing curve shows that the Amitay PIN distribution is much more vulnerable to guessing attacks e.g., *at least* 28.66% PINs can be guessed by an attacker making $G = 96$ guesses compared to *at most* 1% of uniformly random PIN numbers.

Small PIN Datasets. We also use compare our new statistical techniques with those of Blocki and Liu [8] the first choice 4-digit PIN dataset collected from the user studies of Markert et al. [32]. The dotted blue line ($\texttt{newLB}^{M}$) applies the lower bound in Theorem 3 by instantiating the model M with a dictionary Amitay PINs (ordered by frequency), while the dotted green line ($\texttt{newLB}^{samp}$) refers to Corollary 4 with sampling parameter $d = N/2$. The dotted orange line (`DictLB`) is derived from the existing technique `ExtendedLB` but using Amitay dataset as a dictionary, referring to Theorem 7 in Appendix A.

Because this dataset is small ($N = 851$) we find that the upper/lower bounds are not particularly tight. However, our new statistical technique significantly reduce the additive gap between the upper and lower bounds. For example, when $G = 1$ (resp. $G = 8$) prior bounds of [8] (LPLB, LPUB, SamplingLB, FrequencyUB) only that the value λ_G lies somewhere in the range $0 \leq \lambda_G \leq 7.2\%$ (resp. $0.175\% \leq \lambda_G \leq 10.725\%$). By contrast, our new bounds imply that $1.13\% \leq \lambda_G \leq 3.42\%$ (resp. $2.783\% \leq \lambda_G \leq 7.6\%$) i.e., the lower bound is no longer completely (resp. nearly) trivial and the additive gap is reduced by a multiplicative factor 3.14 (resp. 2.18). Our new lower bound outperforms existing lower bounds for all guessing numbers G in this four-digit PIN setting. However, the upper bound generated by the linear programming approach `LBUB` is tighter than our new upper bound for larger guessing numbers $G > 193$.

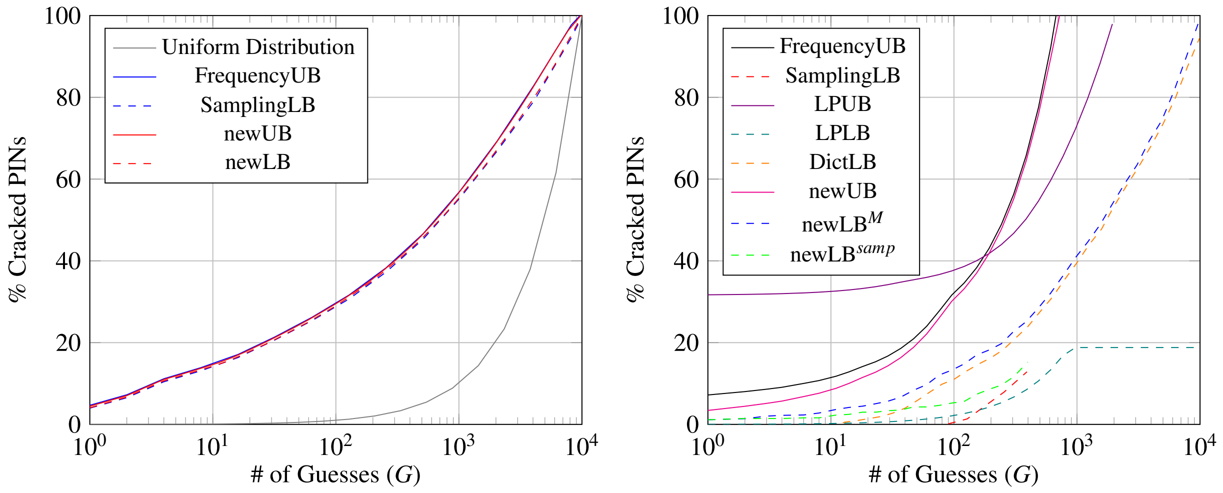
Discussion: In settings where a dataset is collected from a user study one would typically expect that the sample size N will be relatively small. Our new statistical bounds significantly outperform prior bounds of Blocki and Liu [8] in such settings. However, when the sample size too small we are still not able to obtain tight upper/lower bounds even with our improved statistical techniques.

5.2.3 Comparing the PIN Distributions

We now apply our statistical bounds to compare the distributions of Amitary dataset and the first choice user study dataset [32]. To compute the new lower bounds $\texttt{NewLB}^{M}$ for both datasets, we use the first half of Amitay dataset (randomly selected samples) as a guessing dictionary to be model M, and use the remaining half of Amitay dataset as test set. As shown in Figure 3 the PIN distribution of the Amitay dataset is demonstrated to be significantly less secure than the distribution of user study dataset with high confidence for $G \leq 76$. For example, when the guessing number $G = 11$ our confidence bounds indicate that $\lambda_G \geq 14.58\%$ for the Amitay PIN distribution while $\lambda_G \leq 8.83\%$ for the user study distribution. We conjecture that, even for larger guessing budgets $G > 76$, the user study distribution is stronger. However, we cannot make this assertion with confidence as the upper/lower bounds for the user study distribution start to diverge when $G > 76$ due to the smaller sample size.

Explanations for the Gap It is unclear why the user study PIN distribution was significantly strong than the Amitay PIN distribution though this could be a worthwhile topic for follow up research. An optimistic conjecture is that cybersecurity education has positively influenced users towards stronger PIN numbers over the past decade — the Amitay dataset was released in 2011 while the user study dataset was collected from 2019 to 2020. A more pessimistic explanation may be that participants in the user study were more willing to select stronger/less memorable PINs for a user study because they were not worried about forgetting their PIN number. Another possible explanation is that the online user study included a disproportionate percentage of tech savy users who may tend to select stronger PIN numbers. However, only 26% of participants in the user study reported “having a technical background."

Using Amitay as a PIN cracking dictionary? In prior research on PIN numbers the Amitay PIN dataset has been used to generate PIN cracking dictionaries and simulate the



(a) Amitay (with DP) 4-digit PIN (N=204445)

(b) First Choice 4-digit PIN (N=851)

Figure 2: Amitay (with DP) And User Study 4-digit PIN Guessing Curves

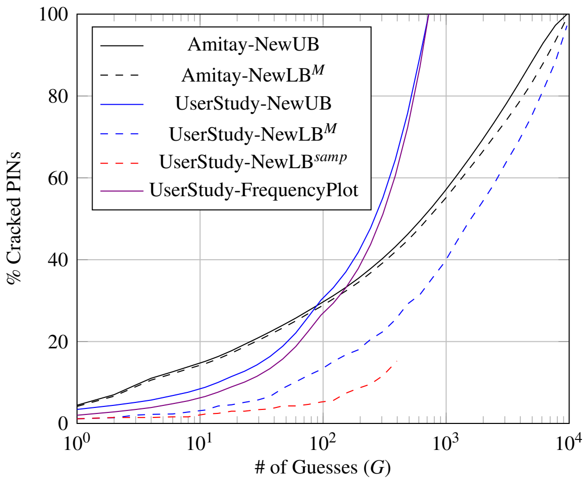


Figure 3: Compare Amitay (without DP) 4-digit PIN (N=204432) And First Choice User Study 4-digit PIN (N=851)

attacker e.g., [32, 34, 37]. It has been claimed that the Amitay dataset offers the “most realistic" simulation and is significantly better than using PINs derived from leaked password datasets [37]. Our finding that the Amitay PIN distribution is less secure raises some questions about the soundness of this methodology — even if it is superior to using PINs derived from leaked password datasets. In particular, it is possible the attacker will have access to superior PIN cracking dictionaries that significantly outperform the Amitay dictionary. To investigate this question we also plot the cumulative distribution function of the empirical distribution defined by the first choice 4-digit PIN dataset [32] — see the purple line in Figure 3. Intuitively, this curve represents the guessing curve of an (unrealistically strong) attacker who knows precisely how many times each particular PIN number appears in the dataset, but does not know a priori which PIN number a particular target user selected. We observe that there is a significant gap between the performance this curve (solid purple) and the lower bound (dashed blue) generated by using an Amitay dictionary to attack the PIN dataset from the user study. The fact that this gap is present even for smaller guessing numbers suggests, but does not prove, that the Amitay dictionary is sub-optimal for cracking PINs from the first choice user study dataset [32].

5.3 The Impact of Blocklists on PIN Security

We previously saw that user PIN distributions are highly non-uniform making them easier for an attacker to guess. One attempt to boost PIN security is to impose a PIN blocklist to prevent users from selecting overly popular PIN numbers. Do blocklists improve security? If so what is the optimal size of a blocklist? In this section we seek to use our statistical bounds to help address this questions.

5.3.1 Blocklist Datasets

Ideally, to analyze the impact of blocklists on PIN security one would like to obtain a large dataset of user PINs selected under *each* different possible blocklist that we are considering. Markert et al. [32] conducted several user studies collecting users’ choices of PINs under various blocklists. Unfortunately, the number of participants under each condition is still too small to draw meaningful conclusions from our rigorous statistical bounds[3].

[3] Each user study successfully collected ≤ 200 participants’ data. For example, in the user study of blocking the top 2740 4-digit PINs in Amitay

5.3.2 Normalized Probabilities Assumption

We address this challenge by making a heuristic assumption about the way that users respond to blocklists following prior work [7, 8]. Let Blocked be a predicate representing the blocklist policy, i.e., $\texttt{Blocked}(pin) = 1$ if and only if the PIN *pin* is on the blocklist that users are not allowed to selected from. If $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) denotes the PIN probability distribution before (resp. after) applying the blocklist then the normalized probabilities model [7] says that $\Pr_{x \leftarrow \mathcal{P}_2}[x = pin] = \Pr_{x \leftarrow \mathcal{P}_1}[x = pin \mid \texttt{Blocked}(x) = 0]$. If a PIN dataset D_1 contains N iid samples from the distribution $\mathcal{P}_1$, we can obtain a filtered dataset $D_2 = \{pin \in D_1 : \texttt{Blocked}(pin) = 0\}$ by removing all PINs that are in the blocklist. Then D_2 can be viewed as $|D_2|$ iid samples from distribution $\mathcal{P}_2$.

We discuss the ecological validity of the normalized probabilities assumption in Section 5.3.4. While the assumption is likely inaccurate for password composition policies, we argue that the assumption is much more plausible in settings the blocklist is simply a list of banned PINs instead of a semantic list of rules governing the password length and the character set. At minimum the heuristic assumption is a useful tool which allows us to quickly identify promising blocklists/blocklist sizes for further empirical evaluation.

5.3.3 Blocklist Analysis

To simulate blocklists of varying sizes we filter the Amitay dataset to remove the top x most frequent passwords for $x \in \{0, 1, 27, 100, 1000, 2740, 4000\}$. For each filtered dataset we apply our new statistical upper/lower bounds (NewUB and $\texttt{NewLB}^{samp}$ with sampling parameter $d = N/2$) to analyzing the attacker's (normalized) guessing curve — see Figure 4b. Each bound holds with at least with probability 99%. Figure 4b shows that blocklist can significantly reduce attackers' guessing efficiency.

Figure 4a is identical to Figure 4b except that we use prior statistical bounds ((FrequencyUB and SamplingLB) from [8] to upper/lower bound the attacker's guessing curve for each normalized distribution. As shown in Figure 4b the prior bounds are not tight enough to draw many meaningful conclusions about the optimal size of a blocklist. We can only conclude that a blocklist of size $x = 1$ is superior to no blocklist ($x = 0$) because the dotted black line (lower bound for $x = 0$) is above the solid blue line (upper bound for $x = 1$). Similarly, we can conclude that blocklists of size $x \geq 27$ are superior to blocklists of size 1. However, the comparison between blocklists of size $x \geq 27$ is uncertain. We can draw sharper comparisons using our new statistical bounds. For example, in Figure 4b it is clear that blocking $x = 100$ passwords improves upon a blocklist of size $x = 27$ and that a blocklists of size $x = 1000$ offer further improvements. However, increasing the size of the blocklist does not always significantly improve the security of the PIN distribution. Compared with blocking 1000 PINs, there are at most a small improvement on security by blocking 2740 or 4000 PINs. Selecting large blocklists can also have a negative impact on usability e.g., when 2740 PINs were blocked in the user study of [32] one participant had to reenter a new PIN eight times before finding a PIN that is not on the blocklist.

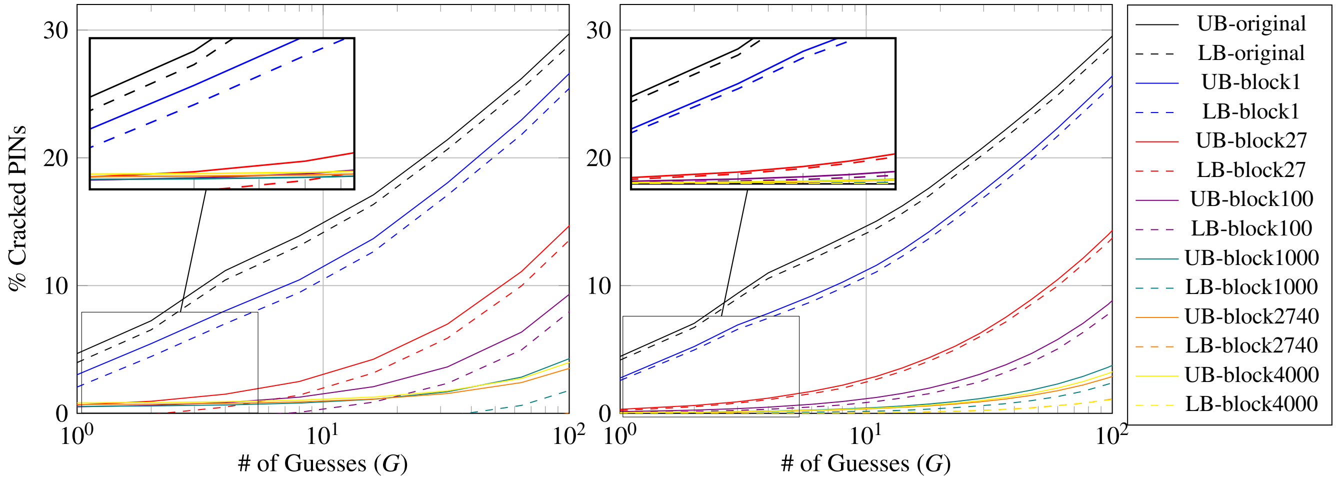
5.3.4 On the Ecological Validity of the Normalized Probabilities Assumption

Recall that the normalized probabilities assumption [7] says that if we block a subset S of PINs/passwords that the updated PIN/password distribution is simply a normalized version of the old distribution i.e., we randomly sample from the original distribution conditioning on the event that the selected PIN/password is not in the blocked set S. This heuristic assumption allows one to quickly analyze different password composition policies or PIN blocklists without conducting additional user studies. In particular, if we are given a dataset of samples from the original distribution then we can simply filter out password that are inconsistent with the policy and the updated dataset can be interpreted as independent samples from the normalized distribution. However, empirical password analysis [27, 35] calls the validity of this assumption into question for password composition policies.

We conjecture that the normalized probabilities assumption may still be reasonable in contexts where the user is not given a simple description of the blocklist. In particular, if the only feedback that a user receives is that the password/PIN they selected was on the blocklist then it seems plausible that the user's behavior would be described by the rejection sampling procedure implicit in the normalized probabilities model i.e., continue sampling fresh random passwords from the underlying distribution until we find one that is allowed. By contrast, if a user picks a password (e.g., "letmein") and then is told that the password must contain a capital letter and a number it seems plausible that the user may try to transform their initial password (e.g. "Letmein1") to comply with these rules instead of resampling a fresh password. Password composition policies tend to be specified in terms of a fixed set of rules (length, character set, capitalization) and so the findings of [27,35] that the normalized probability model does not hold in this context is less surprising. By contrast, PIN policies typically are specified as a list of blocked PIN numbers so it is more plausible that the normalized probabilities assumption would hold. Rigorously testing the validity of of the normalized probabilities assumption for PIN datasets/blocklists is an interesting challenge that is left for future research.

We use the dataset from [32] to investigate the validity of the normalized probability assumption. This dataset includes the sequence of PIN numbers that each user actually selected

dataset, 115 out of 127 (90.5%) PINs before applying the blocklist are unique, and similarly 119 out of 127 (93.7%) PINs after applying the blocklist are unique.



(a) Prior Bounds

(b) New Bounds

Figure 4: Amitay (with DP) 4-digit PIN (N=204445) with Blocklist Size from 1 to 4000

when s/he encountered a particular blocklist. This gives us an alternate way to obtain a PIN dataset for a any blocklist (possibly different than any of the particular blocklists from the actual user study). In particular, for each user we look at the sequence of PIN numbers that each user selected and pick the first PIN number in the sequence that is compatible with the given policy (dropping users who never picked a PIN number consistent with the blocklist we are currently considering). Arguably, this *is* the PIN that the user would have picked under the current blocklist i.e., if this is the ith PIN number in the user's sequence then the user would have received the same feedback (PIN not allowed) after each of the first $i-1$ attempts. We call this dataset the "final choice" dataset. We can then apply our statistical upper/lower bounds to analyze this dataset and compare it with the bounds we get under the normalized probabilities assumption i.e., by starting with the "first choice" dataset and filtering out any password that is not consistent with our blocklist.

The dataset from [32] includes PIN sequences three existing user studies that record the users' choices when applying three different block lists, including all PIN choices that each user made before the user finally picked a PIN that is not in the blocklist. The three user studies contain records of 121, 126, 127 users respectively. To increase the sample size we merge the three datasets and consider the blocklist $B = B_1 \cap B_2 \cap B_3$. The intersection of the three different blocklists (IOS4Digit, Amitay27 and Amitay2740) yields a new blocklist of size 22. Figure 5 in Appendix B compares the resulting upper/lower bounds with those obtained under the normalized probabilities model. The upper (resp. lower) bounds obtained under the normalized probabilities model are very close to the upper/lower bounds obtained from the final choice dataset indicating that the upper/lower bounds generated under the normalized probabilities model are accurate. While Figure 5 provides initial evidence in support of the normalized probabilities model for blocklists, the smaller sample size prevents us from claiming that the two distributions (normalized vs final choice) are close. Additional studies would be required to empirically (in)validate the normalized probabilities assumption for PIN blocklists.

6 Conclusion

We introduced new statistical bounds to analyze the the guessing curve of a optimal password cracking attack given N independent samples from an unknown password distribution. Empirical analysis demonstrates that, in comparison to prior work, we can often obtain tighter upper/lower bounds on the attacker's guessing curve by applying our results. The upper/lower bounds are significantly improved when the guessing budget is small allowing us to draw conclusions about the performance of a password spraying attacker that are both meaningful and statistically rigorous. Our new bounds also allow us to draw sharper conclusions about the impact of password/PIN blocklists on password spraying attacks although further research is needed to (in)validate the normalized probabilities assumption — a heuristic assumption about how password/PIN blocklists impact the unknown password/PIN distribution. Finally, we also provided the first theoretical analysis demonstrating that the one can use the empirical password distribution to obtain asymptotically tight approximations of the actual guessing curve as long as the guessing budget is $G = o(N/\log N)$.

Availability

Our source code is available on an anonymous Github repository at . The Github repository implements our upper bound $\lambda^{UB}(\delta, D, G) := UB_\delta(F_G^{D,freq})$ (see Theorem 2) and our lower bound LB_δ (see Theorem 3). The implementation takes as input a password dataset D as well as relevant parameters like G (guessing number) and δ and outputs an upper/lower bound. For the purpose of comparison our Github repository also included implementations of all of the statistical upper/lower bounds from [8]. The Github repository include two differentially private password frequency datasets: Yahoo! [5, 10] and LinkedIn [24]. Additionally, the Github repository contains several examples showing how to generate upper/lower bounds on λ_G using these differentially private password frequency datasets.

Ethics

We apply our new statistical techniques to analyze breached password datasets and PIN datasets. The usage of breached datasets which contain passwords and PINs selected by actual users raises ethical considerations. We took the following steps to avoid causing additional harm to users: (1) We only utilized password/PIN datasets that are publicly available, and (2) we did not crack any new passwords/PINs as part of our analysis. Our new statistical techniques improve scientific understanding of user password distributions. This could benefit users by helping organizations to adopt more informed password/PIN policies. While we are not introducing new attacks that an adversary could use to crack passwords, we do acknowledge that improving our understanding of password distributions could potentially benefit the attacker as well as the defender. However, we believe that the potential benefits to users will outweigh any harms. In particular, the statistical analysis could benefit users by helping organizations to adopt more informed decisions about lockout policies for passwords/PINs and about blacklist sizes.

References

[1] Daniel Amitay. Most common iphone passcodes. https://www.danielamitay.com/blog/2011/6/13/most-common-iphone-passcodes, June 13, 2011.

[2] Wenjie Bai and Jeremiah Blocki. DAHash: Distribution aware tuning of password hashing costs. In Nikita Borisov and Claudia Díaz, editors, *FC 2021, Part II*, volume 12675 of *LNCS*, pages 382–405. Springer, Heidelberg, March 2021.

[3] Andrea Bianchi, Ian Oakley, and Dong Soo Kwon. Counting clicks and beeps: Exploring numerosity based haptic and audio pin entry. *Interacting with computers*, 24(5):409–422, 2012.

[4] Jeremiah Blocki and Anupam Datta. CASH: A cost asymmetric secure hash algorithm for optimal password protection. In Michael Hicks and Boris Köpf, editors, *CSF 2016 Computer Security Foundations Symposium*, pages 371–386. IEEE Computer Society Press, 2016.

[5] Jeremiah Blocki, Anupam Datta, and Joseph Bonneau. Differentially private password frequency lists. In *NDSS 2016*. The Internet Society, February 2016.

[6] Jeremiah Blocki, Benjamin Harsha, and Samson Zhou. On the economics of offline password cracking. In *2018 IEEE Symposium on Security and Privacy*, pages 853–871. IEEE Computer Society Press, May 2018.

[7] Jeremiah Blocki, Saranga Komanduri, Ariel Procaccia, and Or Sheffet. Optimizing password composition policies. In *Proceedings of the Fourteenth ACM Conference on Electronic Commerce*, EC '13, page 105–122, New York, NY, USA, 2013. Association for Computing Machinery.

[8] Jeremiah Blocki and Peiyuan Liu. Towards a rigorous statistical analysis of empirical password datasets. In *2023 IEEE Symposium on Security and Privacy*, pages 606–625. IEEE Computer Society Press, May 2023.

[9] Jeremiah Blocki and Wuwei Zhang. DALock: Password distribution-aware throttling. *PoPETs*, 2022(3):516–537, July 2022.

[10] Joseph Bonneau. The science of guessing: Analyzing an anonymized corpus of 70 million passwords. In *2012 IEEE Symposium on Security and Privacy*, pages 538–552. IEEE Computer Society Press, May 2012.

[11] Joseph Bonneau, Sören Preibusch, and Ross Anderson. A birthday present every eleven wallets? The security of customer-chosen banking PINs. In Angelos D. Keromytis, editor, *FC 2012*, volume 7397 of *LNCS*, pages 25–40. Springer, Heidelberg, February / March 2012.

[12] Daniel Buschek, Alexander De Luca, and Florian Alt. Improving accuracy, applicability and usability of keystroke biometrics on mobile touchscreen devices. In *proceedings of the 33rd annual ACM conference on human factors in computing systems*, pages 1393–1402, 2015.

[13] Claude Castelluccia, Abdelberi Chaabane, Markus Dürmuth, and Daniele Perito. When privacy meets security: Leveraging personal information for password cracking. *arXiv preprint arXiv:1304.6584*, 2013.

[14] Claude Castelluccia, Markus Dürmuth, and Daniele Perito. Adaptive password-strength meters from Markov models. In *NDSS 2012*. The Internet Society, February 2012.

[15] Rahul Chatterjee, Anish Athayle, Devdatta Akhawe, Ari Juels, and Thomas Ristenpart. pASSWORD tYPOS and how to correct them securely. In *2016 IEEE Symposium on Security and Privacy*, pages 799–818. IEEE Computer Society Press, May 2016.

[16] Rahul Chatterjee, Joanne Woodage, Yuval Pnueli, Anusha Chowdhury, and Thomas Ristenpart. The TypTop system: Personalized typo-tolerant password checking. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *ACM CCS 2017*, pages 329–346. ACM Press, October / November 2017.

[17] Joseph Cox. Another day, another hack: Tens of millions of neopets accounts. https://motherboard.vice.com/en_us/article/ezpvw7/neopets-hack-another-day-another-hack-tens-of-millions-of-neopets-accounts, May 5, 2016.

[18] Joseph Cox. Nearly 800,000 brazzers porn site accounts exposed in forum hack. https://motherboard.vice.com/en_us/article/vv7pgd/nearly-800000-brazzers-porn-site-accounts-exposed-in-forum-hack, September 5, 2016.

[19] Nik Cubrilovic. Rockyou hack: From bad to worse. https://techcrunch.com/2009/12/14/rockyou-hack-security-myspace-facebook-passwords/, December 15, 2009.

[20] Matteo Dell'Amico and Maurizio Filippone. Monte Carlo strength evaluation: Fast and reliable password checking. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *ACM CCS 2015*, pages 158–169. ACM Press, October 2015.

[21] Markus Dürmuth, Fabian Angelstorf, Claude Castelluccia, Daniele Perito, and Abdelberi Chaabane. Omen: Faster password guessing using an ordered markov enumerator. In *International Symposium on Engineering Secure Software and Systems*, pages 119–132. Springer, 2015.

[22] Dan Goodin. 13 million plaintext passwords belonging to webhost users leaked online. https://arstechnica.com/information-technology/2015/10/13-million-plaintext-passwords-belonging-to-webhost-users-leaked-online/, October 28, 2015.

[23] Dan Goodin. 6.6 million plaintext passwords exposed as site gets hacked to the bone. https://arstechnica.com/information-technology/2016/09/plaintext-passwords-and-wealth-of-other-data-for-6-6-million-people-go-public/, September 13, 2016.

[24] Benjamin Harsha, Robert Morton, Jeremiah Blocki, John Springer, and Melissa Dark. Bicycle attacks considered harmful: Quantifying the damage of widespread password length leakage. *Computers & Security*, 100:102068, 2021.

[25] Hashcat. https://hashcat.net/hashcat/. Accessed March 15, 2021.

[26] John the ripper. https://www.openwall.com/john/. Accessed March 15, 2021.

[27] Patrick Gage Kelley, Saranga Komanduri, Michelle L. Mazurek, Richard Shay, Timothy Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Julio Lopez. Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms. In *2012 IEEE Symposium on Security and Privacy*, pages 523–537. IEEE Computer Society Press, May 2012.

[28] Hyoungshick Kim and Jun Ho Huh. Pin selection policies: Are they really effective? *computers & security*, 31(4):484–496, 2012.

[29] Enze Liu, Amanda Nakanishi, Maximilian Golla, David Cash, and Blase Ur. Reasoning analytically about password-cracking software. In *2019 IEEE Symposium on Security and Privacy*, pages 380–397. IEEE Computer Society Press, May 2019.

[30] Peiyuan Liu, Jeremiah Blocki, and Wenjie Bai. Confident monte carlo: Rigorous analysis of guessing curves for probabilistic password models. In *2023 IEEE Symposium on Security and Privacy*, pages 626–644. IEEE Computer Society Press, May 2023.

[31] Jerry Ma, Weining Yang, Min Luo, and Ninghui Li. A study of probabilistic password models. In *2014 IEEE Symposium on Security and Privacy*, pages 689–704. IEEE Computer Society Press, May 2014.

[32] Philipp Markert, Daniel V. Bailey, Maximilian Golla, Markus Dürmuth, and Adam J. Aviv. This PIN can be easily guessed: Analyzing the security of smartphone unlock PINs. In *2020 IEEE Symposium on Security and Privacy*, pages 286–303. IEEE Computer Society Press, May 2020.

[33] Philipp Markert, Daniel V Bailey, Maximilian Golla, Markus Dürmuth, and Adam J Aviv. On the security of smartphone unlock pins. *ACM Transactions on Privacy and Security (TOPS)*, 24(4):1–36, 2021.

[34] Philipp Markert, Daniel V. Bailey, Maximilian Golla, Markus Dürmuth, and Adam J. Aviv. On the security of smartphone unlock pins. *ACM Trans. Priv. Secur.*, 24(4), sep 2021.

[35] Michelle L. Mazurek, Saranga Komanduri, Timothy Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Patrick Gage Kelley, Richard Shay, and Blase Ur. Measuring password guessability for an entire university. In Ahmad-Reza Sadeghi, Virgil D. Gligor, and Moti Yung, editors, *ACM CCS 2013*, pages 173–186. ACM Press, November 2013.

[36] William Melicher, Blase Ur, Sean M. Segreti, Saranga Komanduri, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. Fast, lean, and accurate: Modeling password guessability using neural networks. In Thorsten Holz and Stefan Savage, editors, *USENIX Security 2016*, pages 175–191. USENIX Association, August 2016.

[37] Collins W. Munyendo, Philipp Markert, Alexandra Nisenoff, Miles Grant, Elena Korkes, Blase Ur, and Adam J. Aviv. "The same PIN, just longer": On the (in)security of upgrading PINs from 4 to 6 digits. In Kevin R. B. Butler and Kurt Thomas, editors, *USENIX Security 2022*, pages 4023–4040. USENIX Association, August 2022.

[38] Arvind Narayanan and Vitaly Shmatikov. Fast dictionary attacks on passwords using time-space tradeoff. In Vijayalakshmi Atluri, Catherine Meadows, and Ari Juels, editors, *ACM CCS 2005*, pages 364–372. ACM Press, November 2005.

[39] Stuart Schechter and Joseph Bonneau. Learning assigned secrets for unlocking mobile devices. In *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*, pages 277–295, 2015.

[40] Yuan Tian, Cormac Herley, and Stuart Schechter. Stopguessing: Using guessed passwords to thwart online guessing. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 576–589. IEEE, 2019.

[41] Blase Ur, Sean M. Segreti, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Saranga Komanduri, Darya Kurilova, Michelle L. Mazurek, William Melicher, and Richard Shay. Measuring real-world accuracies and biases in modeling password guessability. In Jaeyeon Jung and Thorsten Holz, editors, *USENIX Security 2015*, pages 463–481. USENIX Association, August 2015.

[42] Rafael Veras, Christopher Collins, and Julie Thorpe. On semantic patterns of passwords and their security impact. In *NDSS 2014*. The Internet Society, February 2014.

[43] Emanuel Von Zezschwitz, Alexander De Luca, Bruno Brunkow, and Heinrich Hussmann. Swipin: Fast and secure pin-entry on smartphones. In *Proceedings of the 33rd annual acm conference on human factors in computing systems*, pages 1403–1406, 2015.

[44] John Walker. Lulzsec over, release battlefield heroes data. https://www.rockpapershotgun.com/2011/06/26/lulzsec-over-release-battlefield-heroes-data/, June 26, 2011.

[45] Ding Wang, Qianchen Gu, Xinyi Huang, and Ping Wang. Understanding human-chosen PINs: Characteristics, distribution and security. In Ramesh Karri, Ozgur Sinanoglu, Ahmad-Reza Sadeghi, and Xun Yi, editors, *ASIACCS 17*, pages 372–385. ACM Press, April 2017.

[46] Matt Weir, Sudhir Aggarwal, Breno de Medeiros, and Bill Glodek. Password cracking using probabilistic context-free grammars. In *2009 IEEE Symposium on Security and Privacy*, pages 391–405. IEEE Computer Society Press, May 2009.

[47] Xue Yang. Chinese internet suffers the most serious user data leak in history. https://blogs.forcepoint.com/security-labs/chinese-internet-suffers-most-serious-user-data-leak-history, December 26, 2011.

A Missing Theorem

Theorem 7. *Given a sample set D and a guessing dictionary* $\texttt{Dict}$*, for any guessing number $G > 0$ and any parameter $0 \le \varepsilon \le 1$ we have:*

$$\Pr[\lambda_G \ge \frac{1}{N}\texttt{Cracked}(D, \texttt{Dict}_G) - \varepsilon] \ge 1 - \exp(-2N\varepsilon^2)$$

where the randomness is taken over the sample set $D \leftarrow \mathcal{P}^N$.

Proof. This proof is derived from Theorem 6 in [8] by using an dictionary $\texttt{Dict}$ as model M. Define $\texttt{Dict}_G = \cup_{i=1}^{G}\texttt{Dict}[i]$ to be the top G passwords in $\texttt{Dict}$, and let $\texttt{Cracked}(D, \texttt{Dict}_G) = |\{s : s \in D \wedge s \in \texttt{Dict}_G\}|$ be the number of cracked samples in D by making the top G guesses in $\texttt{Dict}_G$. Given any $\texttt{Dict}$ we note that $\mathbb{E}_D(\texttt{Cracked}(D, \texttt{Dict}_G)) = N \cdot \sum_{pwd \in \texttt{Dict}_G} p_{pwd} \le N \cdot \sum_{i=1}^{G} p_i = N\lambda_G$. Using McDiarmid's inequality we have:

$$\begin{aligned}
&\Pr[\lambda_G \ge \frac{1}{N}\texttt{Cracked}(D, \texttt{Dict}_G) - \varepsilon] \\
\ge &\Pr[\sum_{pwd \in \texttt{Dict}_G} p_{pwd} \ge \frac{1}{N}\texttt{Cracked}(D, \texttt{Dict}_G) - \varepsilon] \\
\ge &1 - \exp(-2N\varepsilon^2)
\end{aligned}$$

□

B Missing Figure

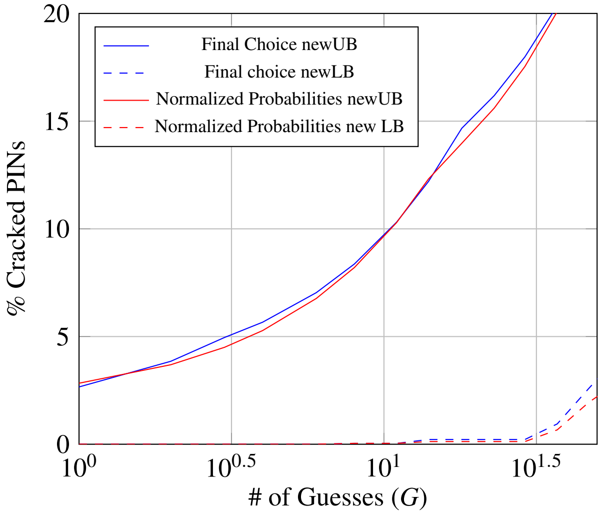


Figure 5: User Selected 4-Digit PINs After Applying The Intersection of Three Blocklists IOS4Digit Amitay27 And Amitay2740 (N=374)

C Missing Proofs

Reminder of Claim 1. *Given any integers $N > 0$ and $0 \le F < N$ and any $0 \le p_1 < p_2 \le N$ we have $\mathtt{bcdf}(F,N,p_1) > \mathtt{bcdf}(F,N,p_2)$.*

Proof of Claim 1. Note that $f(p) = \mathtt{bcdf}(F,N,p) = \sum_{i=0}^{F}\binom{N}{i}p^i(1-p)^{N-i}$. Then the first derivative of $f(p)$ is $f'(p) = -(1-p)^{N-1} + \sum_{i=1}^{F}\binom{N}{i}p^{i-1}(1-p)^{N-i-1}(i-Np)$. For $p \ge F/N$ observe that $f'(p) < 0$ as $(i-Np) \le 0$ for all $i = 0,1,\ldots,F$. By definition of binomial distribution we can also write $f(p)$ as $f(p) = 1 - \sum_{i=F+1}^{N}\binom{N}{i}p^i(1-p)^{N-i}$. Then the first derivative can also be written as $f'(p) = -Np^{N-1} - \sum_{i=F+1}^{N-1}\binom{N}{i}p^{i-1}(1-p)^{N-i-1}(i-Np)$. For $p < F/N$ observe that $f'(p) < 0$ as $(i-Np) \ge 0$ for all $i = F,\ldots,N$. Therefore, $\mathtt{bcdf}(F,N,p)$ is monotonically decreasing.

Reminder of Claim 3. *$UB_\delta(x_1) < UB_\delta(x_2)$ for any $0 \le x_1 < x_2 \le N$.*

Proof of Claim 3. First of all note that $UB_\delta(F) < 1$ for all $F < N$ and $UB_\delta(N) = 1$. Then we only need to prove the strictly monotonically increasing property for all $0 \le F < N$. This can be proved by contradiction. For any $0 \le x_1 < x_2 \le N$ and any fixed $0 \le p \le 1$, we have $\sum_{j=0}^{x_1}\mathtt{bpdf}(j,N,p) < \sum_{j=0}^{x_2}\mathtt{bpdf}(j,N,p)$. Let $p_1 = UB_\delta(x_1)$ and $p_2 = UB_\delta(x_2)$. Assume $p_1 \ge p_2$. Then we have $\sum_{j=0}^{x_1}\mathtt{bpdf}(j,N,p_2) < \sum_{j=0}^{x_2}\mathtt{bpdf}(j,N,p_2) \le \delta$, which is contradicted to the fact that p_1 is the minimum value satisfying $\sum_{j=0}^{x_1}\mathtt{bpdf}(j,N,p_1) \le \delta$. Therefore, $UB_\delta(x_1) < UB_\delta(x_2)$ for any $0 \le x_1 < x_2 \le N$.

Reminder of Lemma 2. *For any parameters $\alpha > 0$ and $\delta \ge \frac{8e}{\alpha^2}$ we have $\Pr[\cup_{i \ge \log(\frac{\alpha^2 N}{4\log N})}\mathtt{BAD}_i] \le \frac{1}{\sqrt{2\pi\delta N}\log N(1-2^{1-\delta\log N})} \cdot \frac{\alpha^2}{2N^{\delta-1}}$.*

Proof of Lemma 2. For $i \ge \log(\frac{\alpha^2 N}{4\log N})$ we can upper bound $\Pr[\mathtt{BAD}_i]$ using balls and bins analysis as:

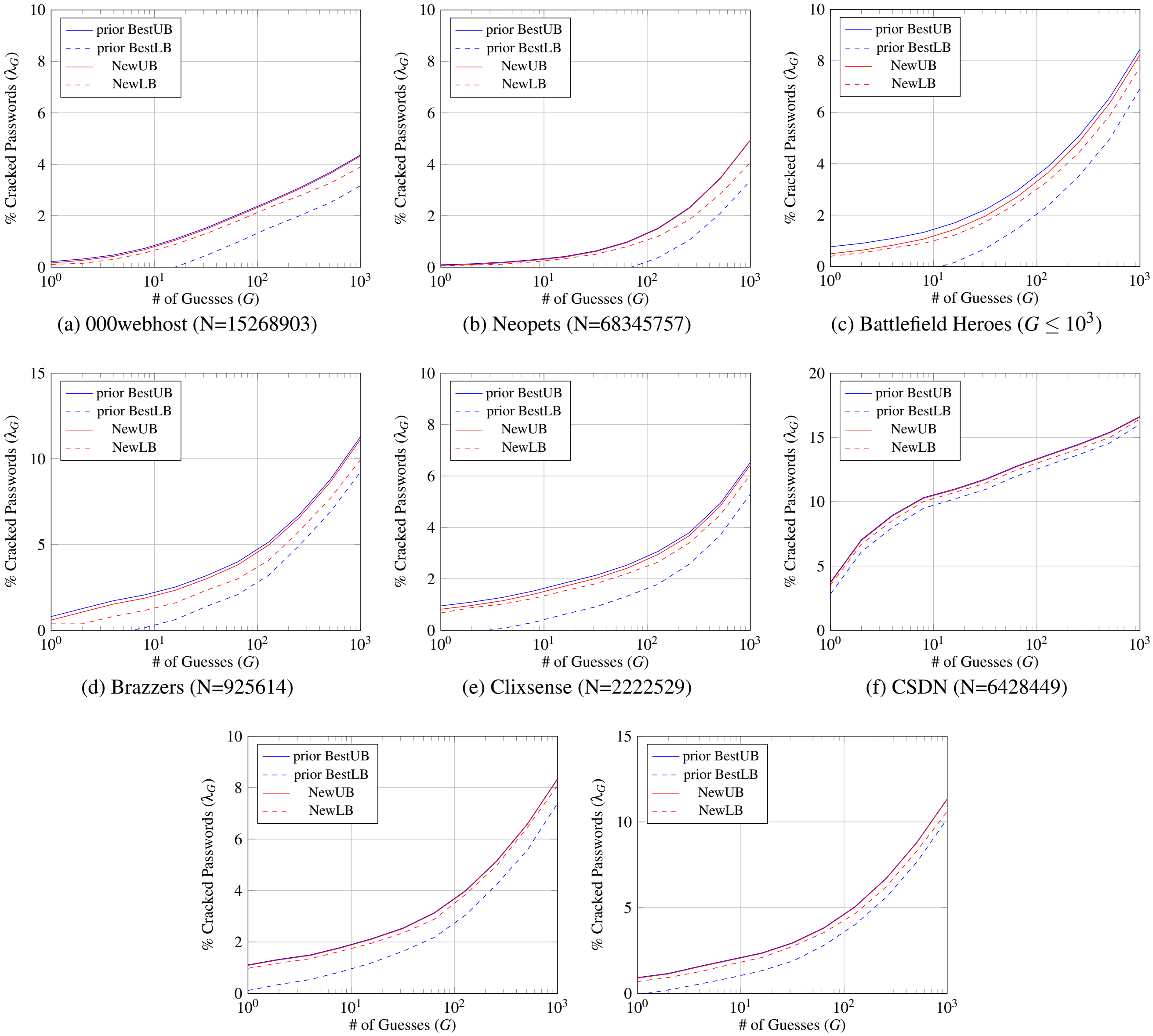
$$\begin{aligned}
\Pr[\mathtt{BAD}_i] &\le \sum_{pwd \in B_i}\binom{N}{\delta\log N}p_{pwd}^{\delta\log N} \\
&\le \sum_{pwd \in B_i}\binom{N}{\delta\log N}\left(2^{-i}\right)^{\delta\log N} \\
&\le \binom{N}{\delta\log N}2^{i+1-i\delta\log N}.
\end{aligned}$$

The first inequality follows by union bounding over all passwords in B_i. The second inequality follows since every password in B_i has probability at least 2^{-i} and the last inequality follows because there are at most 2^{i+1} passwords in the set B_i. It follows that

$$\begin{aligned}
\sum_{i > \log(\frac{\alpha^2 N}{4\log N})}\Pr[\mathtt{BAD}_i] &\le \sum_{i \ge \log(\frac{\alpha^2 N}{4\log N})}\binom{N}{\delta\log N}2^{i+1-i\delta\log N} \\
&= \binom{N}{\delta\log N}\sum_{i \ge \log(\frac{\alpha^2 N}{4\log N})}2^{i+1-i\delta\log N} \\
&\le 2\binom{N}{\delta\log N}\sum_{i > \log(\frac{\alpha^2 N}{4\log N})}2^{(1-\delta\log N)i} \\
&\le 2\binom{N}{\delta\log N}\frac{2^{(1-\delta\log N)\log(\frac{\alpha^2 N}{4\log N})}}{1-2^{(1-\delta\log N)}} \\
&\le 2\frac{N^{\delta\log N}}{(\delta\log N)!}\frac{(\frac{\alpha^2 N}{4\log N})^{(1-\delta\log N)}}{1-2^{(1-\delta\log N)}} \\
&< \frac{1}{\sqrt{2\pi\delta\log N}(\frac{\delta\log N}{e})^{\delta\log N}e^{\frac{1}{12\delta\log N+1}}} \\
&\quad \cdot \frac{2\alpha^{2(1-\delta\log N)}N}{(4\log N)^{(1-\delta\log N)}(1-2^{(1-\delta\log N)})} \\
&< \frac{N}{\sqrt{2\pi\delta N}\log N(1-2^{1-\delta\log N})} \cdot \frac{\alpha^2}{2} \cdot (\frac{4e}{\alpha^2\delta})^{\delta\log N} \\
&\le \frac{1}{\sqrt{2\pi\delta N}\log N(1-2^{1-\delta\log N})} \cdot \frac{\alpha^2}{2N^{\delta-1}}
\end{aligned}$$

where the third last inequality is derived by the lower bound of Stirling's approximation (i.e., $n! > \sqrt{2\pi n}(\frac{n}{e})^n e^{\frac{1}{12n+1}}$ for any $n \ge 1$) and the last inequality is derived by the condition $\delta \ge 8e/\alpha^2$.

Reminder of Theorem 3. *Given a password cracking model M, for any $0 \le \delta \le 1$, $\Pr[\lambda_G \ge \lambda^{LB}(\delta,M,D,G)] \ge 1-\delta$,*



(a) 000webhost (N=15268903)

(b) Neopets (N=68345757)

(c) Battlefield Heroes ($G \leq 10^3$)

(d) Brazzers (N=925614)

(e) Clixsense (N=2222529)

(f) CSDN (N=6428449)

(g) Yahoo! (N=69301337)

(h) RockYou (N=2603388)

Figure 6: 000webhost, Neopets, Battlefield Heroes, Brazzers, Clixsense, CSDN, Yahoo!, and RockYou Guessing Curves with $G \leq 10^3$

where $\lambda^{LB}(\delta, M, D, G) := LB_\delta(F_G^{D,M}, N)$ *and the randomness is taken over the sample set* $D \leftarrow \mathcal{P}^N$.

Proof of Theorem 3. First we define $F_\delta^{LB}(p) = \min_F\{F : \sum_{j=F}^N \texttt{bpdf}(j, N, p) \le \delta\}$ for any $0 \le p \le p^*$ where $p^* = \arg\max_p\{\texttt{bpdf}(N, N, p) \le \delta\}$. For $p^* < p \le 1$ when no F satisfies the condition, define $F_\delta^{LB}(p) = N + 1$. Denote $F' = F_\delta^{LB}(\lambda_{M,G})$. Then we have

$$\sum_{j=F'}^{N} \texttt{bpdf}(j, N, \lambda_{M,G}) \le \delta \tag{4}$$

for $\lambda_{M,G} \le p^*$, and

$$\sum_{j=F'-1}^{N} \texttt{bpdf}(j, N, \lambda_{M,G}) > \delta \tag{5}$$

for all $\lambda_{M,G} \ge 0$. Equation 5 as F' is defined to be the minimum value that satisfies $\sum_{j=F'}^N \texttt{bpdf}(j, N, p) \le \delta$.

Next we denote $p' = LB_\delta(F' - 1)$. Then for $F' > 1$ we have

$$\sum_{j=F'-1}^{N} \texttt{bpdf}(j, N, p') \le \delta \le \sum_{j=F'-1}^{N} \texttt{bpdf}(j, N, \lambda_{M,G})$$

where the first inequality follows from the definition of $p' = LB_\delta(F' - 1)$ and the second inequality follows from Equation 5. Note that for $F' = 1$ we have

$$\sum_{j=F'-1}^{N} \texttt{bpdf}(j, N, p') = 1 = \sum_{j=F'-1}^{N} \texttt{bpdf}(j, N, \lambda_{M,G}) \ .$$

Recall that Claim 2 proves that $f(p) = \sum_{j=F'-1}^N \texttt{bpdf}(j, N, p)\}$ is strictly monotonically increasing for $F' - 1 > 0$ and $p' = 0$ for $F' - 1 = 0$, so $p' \le \lambda_{M,G}$. Note that any password cracking model M cannot be better than the perfect knowledge attacker (i.e., $\lambda_{M,G} \le \lambda_G$). Thus we have $p' \le \lambda_G$.

Since Claim 4 proves that $LB_\delta(F)$ is strictly monotonically increasing, we can observe that $p' = LB_\delta(F' - 1) < LB_\delta(F_G^{D,M})$ if and only if $F' - 1 < F_G^{D,M}$. Therefore, we have:

$$\begin{aligned}\Pr[\lambda_G < LB_\delta(F_G^{D,M})] &\le \Pr[p' < LB_\delta(F_G^{D,M})] \\ &= \Pr[F' - 1 < F_G^{D,M}]\end{aligned}$$

where the inequality holds due to $p' \le \lambda_G$. Here the definition of F' depends only on the password distribution $\mathcal{P}$, not on the samples in D. Note that if $F' = N + 1$ we have $\Pr[F' - 1 < F_G^{D,M}] = 0$; if $F' \le N$ we have $\Pr[F' - 1 < F_G^{D,M}] = \sum_{j=F'}^N \texttt{bpdf}(j, N, \lambda_{M,G}) \le \delta$ by the definition of $F' = F_\delta^{LB}(\lambda_{M,G})$. Therefore, $\Pr[\lambda_G < LB_\delta(F_G^{D,M})] \le \delta$.