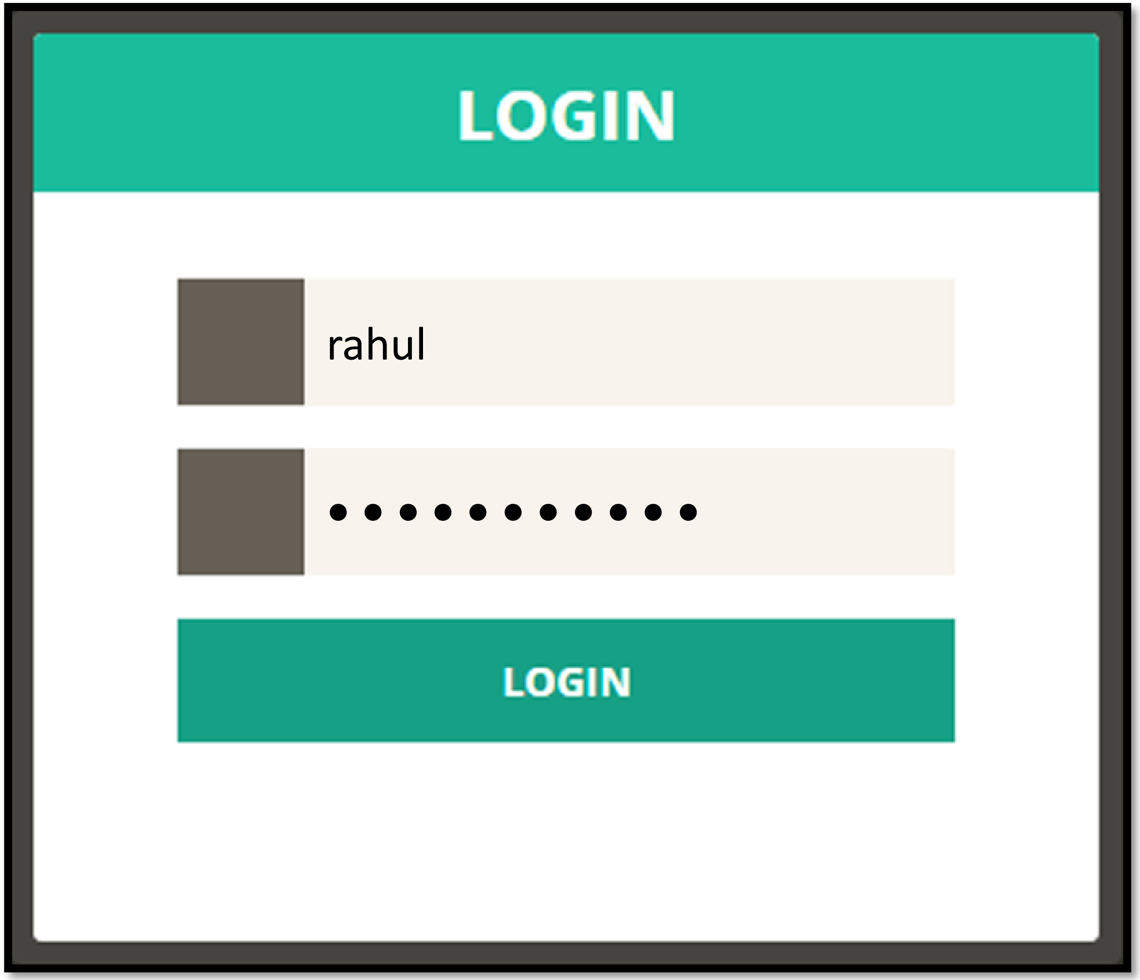
pASSWORD tYPOS
and How to Correct Them Securely

R. Chatterjee, A. Athalye, D. Akhawe, A. Juels, T. Ristenpart

CORNELL TECH Dropbox MIT

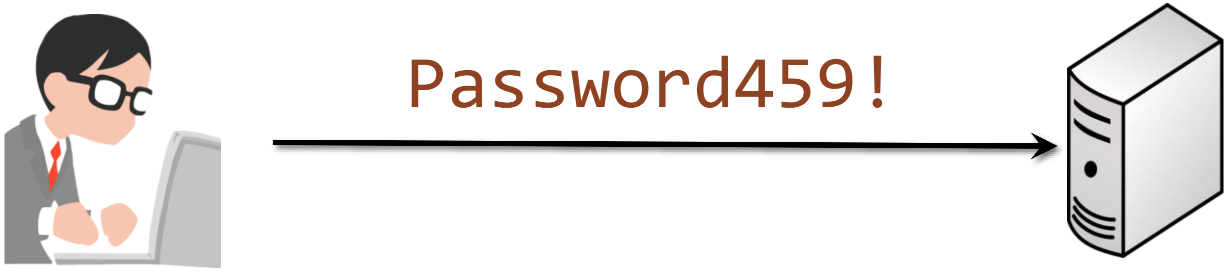
To typo is human; to tolerate, divine.



rahul

●●●●●●●●●●●

Password-based authentication systems



Password459!

Salted, slow cryptographic hash

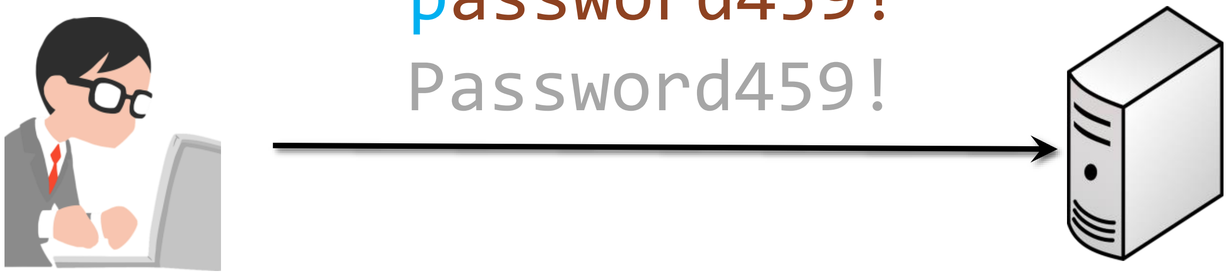
H(Password459!) $\overset{?}{=}$ "a5idoiaU7p.." ✓

Password-based authentication systems

Any typo is rejected

password459!

Password459!



Salted, slow cryptographic hash

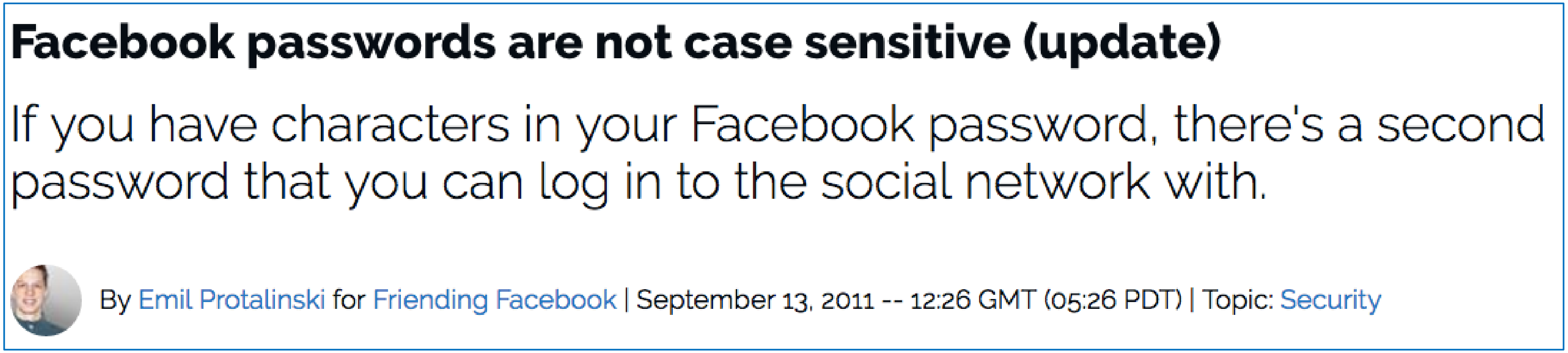
H(Password459!) $\overset{?}{=}$ “a5idoiaU7p..” ✓

H(password459!) $\overset{?}{=}$ “a5idoiaU7p..” ✗

Typo-tolerant password checking

Allow registered password or typos of it

Typo-tolerant password checking in industry



Facebook passwords are not case sensitive (update)

If you have characters in your Facebook password, there's a second password that you can log in to the social network with.

By Emil Protalinski for Friending Facebook | September 13, 2011 -- 12:26 GMT (05:26 PDT) | Topic: Security

Password459! pASSWORD459! password459!

We know little about password typos

Lots of work on usability of passwords…

[Ur et al. 2012], [Shay et al. 2012, 2014], [Mazurek et al. 2013], [Bonneau, Schechter 2014] [Keith et al. 2007, 2009], [Bard 2007], [Jakobsson et al. 2012]

… but nothing on typo-tolerant password checking.

1. How can we build a typo-tolerant systems?

2. How much would tolerating typos help users?

3. Does it endanger security?

Our work

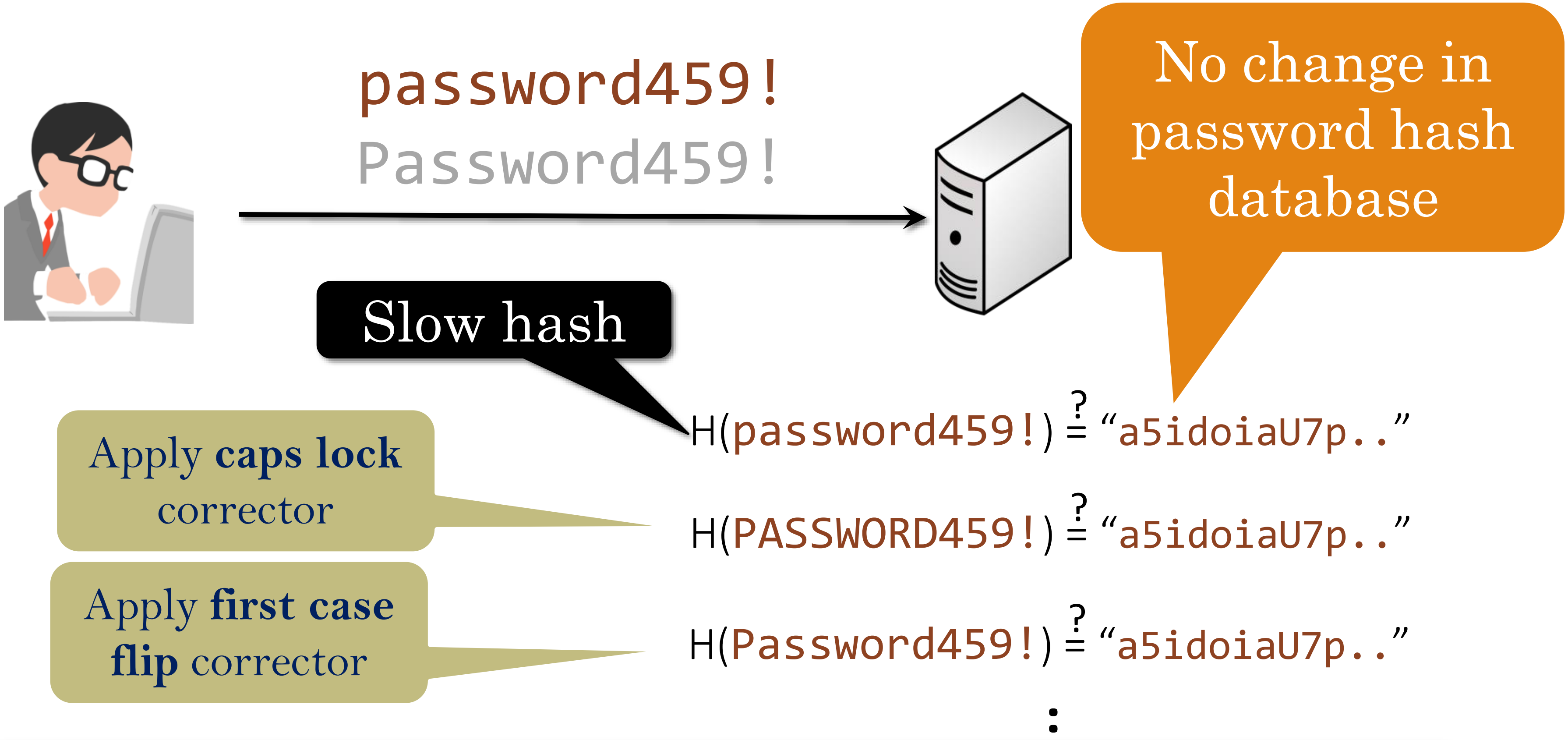
We measure password typos at Dropbox and show they are a huge problem for both users and service providers.

We develop approaches to typo-tolerant checking, and show they improve utility with minimal security impact.

“Have your cake and eat it too”

How to do typo-tolerant password checking?

We focus on *relaxed checkers*



password459!

Password459!

No change in password hash database

Slow hash

Apply **caps lock** corrector

Apply **first case flip** corrector

H(password459!) $\overset{?}{=}$ “a5idoiaU7p..”

H(PASSWORD459!) $\overset{?}{=}$ “a5idoiaU7p..”

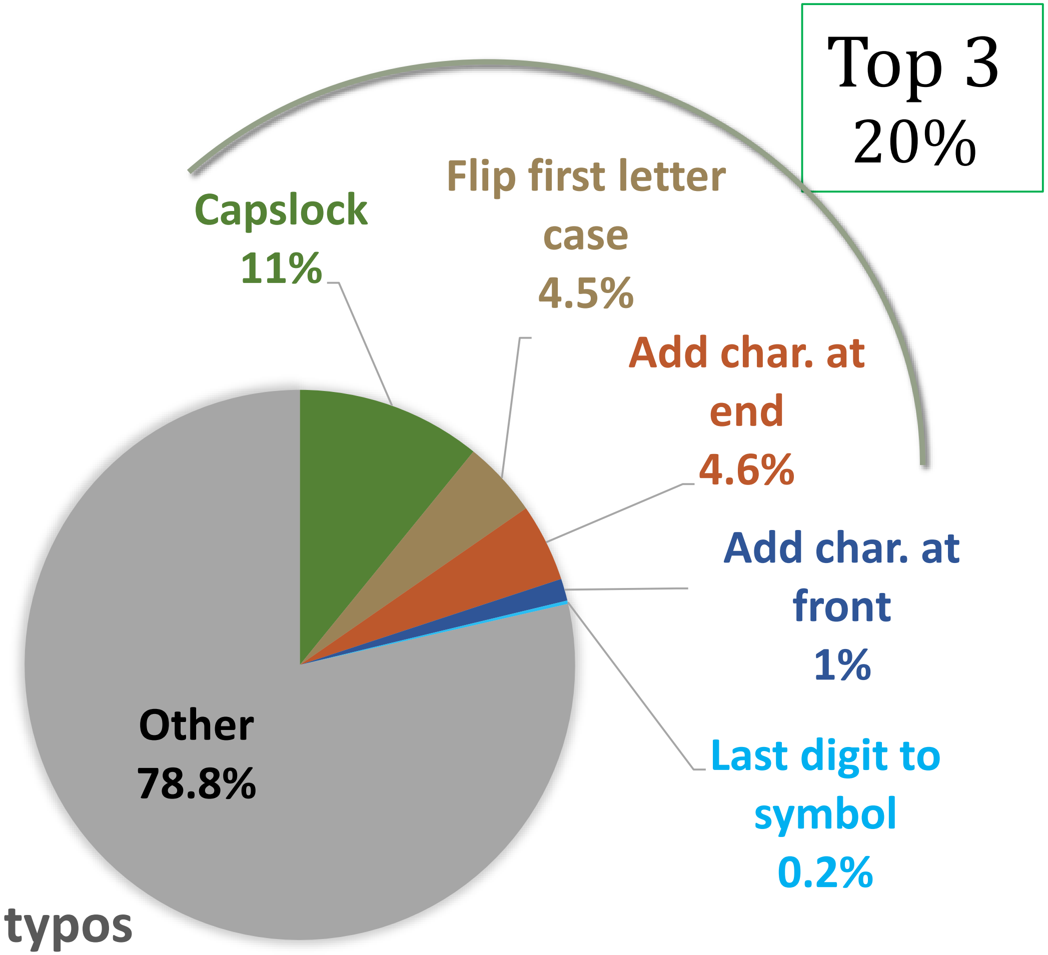
H(Password459!) $\overset{?}{=}$ “a5idoiaU7p..”

:

Can we find a small but useful set of typo correctors?

MTurk password transcription study

100,000+ passwords typed by 4,300 workers



% of all typos

Impact of top-3 typos in the real world

Instrumented production login of Dropbox to quantify typos
NOTE: We did not change authentication policy.

24 hour period:

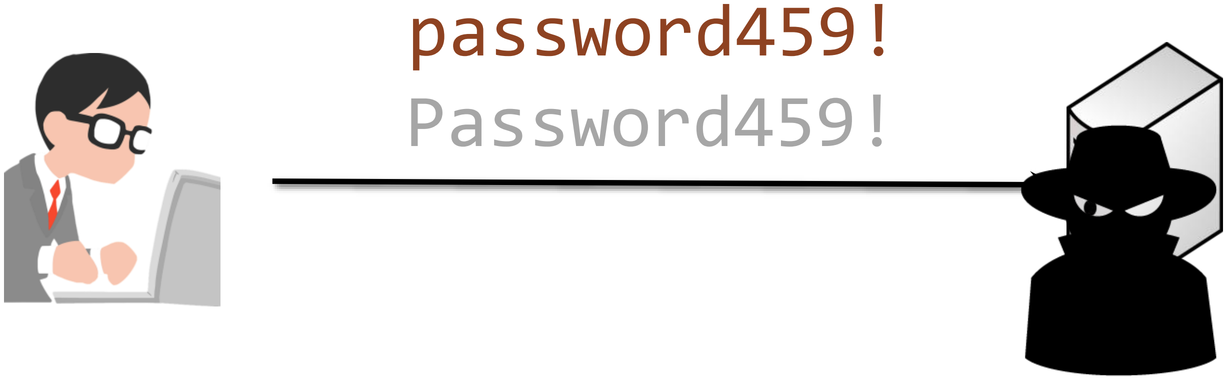
- 3% **of all users** failed to login because one of top 3 typos
- 20% of users who made a typo would have saved at least 1 minute in logging into Dropbox if top 3 typos are corrected.

Allowing typos in password will add several person-months of login time every day.

Typo-tolerance will significantly enhance usability of passwords.

Can it be secure?

Threat #1: Server compromise



password459!

Password459!

No change in password hash database

H(password459!) $\overset{?}{=}$ "a5idoiaU7p.."

H(PASSWORD459!) $\overset{?}{=}$ "a5idoiaU7p.."

H(Password459!) $\overset{?}{=}$ "a5idoiaU7p.."

No change in security in case of server compromise

Threat #2: Remote guessing attack

Web service should lock account after *q* wrong guesses.

Get 3 free checks with every query.

⇒ *q* queries result in 3*q* free password guesses.

⇒ Previously, *q* queries result in no free guesses

⇒ ~~**Attacker's success increases by 300%**~~

flip corrector

Apply **extra char. at end** corrector

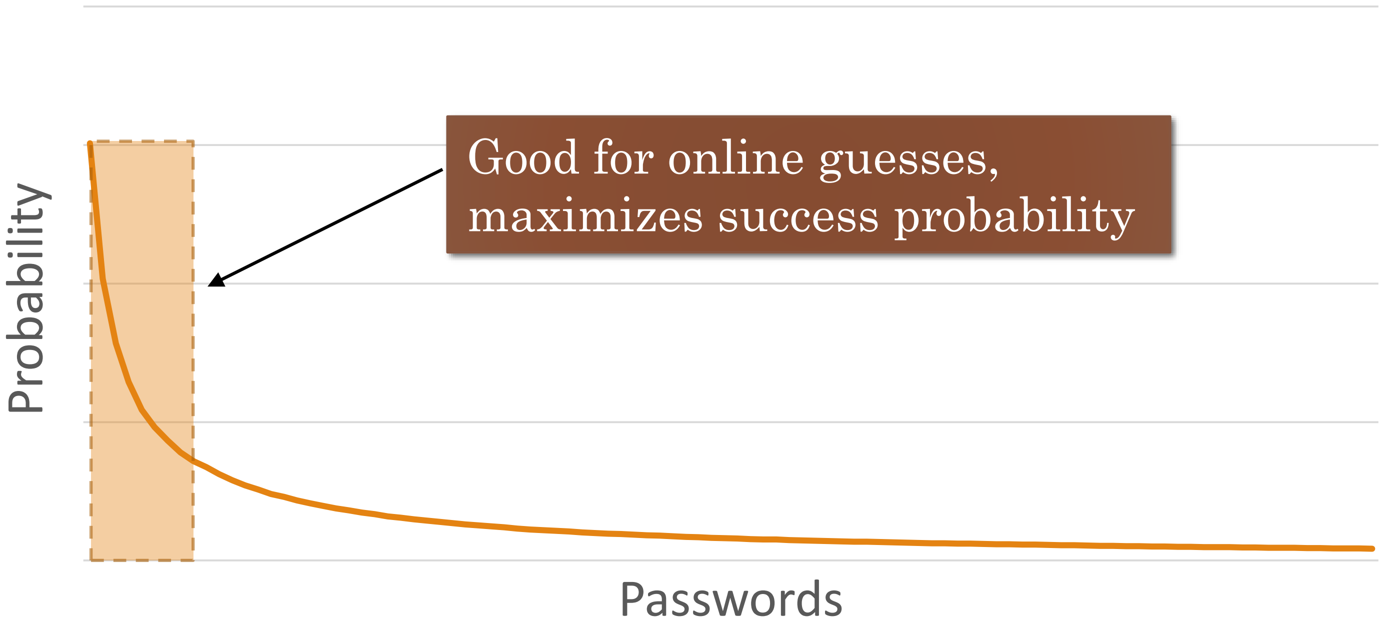
H(Password) = "a5idoiaU7p.."

H(passwor) $\overset{?}{=}$ "a5idoiaU7p.."

Passwords are not uniformly distributed!

300% improvement, only if all checked passwords are equally probable.

BUT, humans do not chose random passwords.



Good for online guesses, maximizes success probability

Attack simulation using password leaks

Adversary knows:

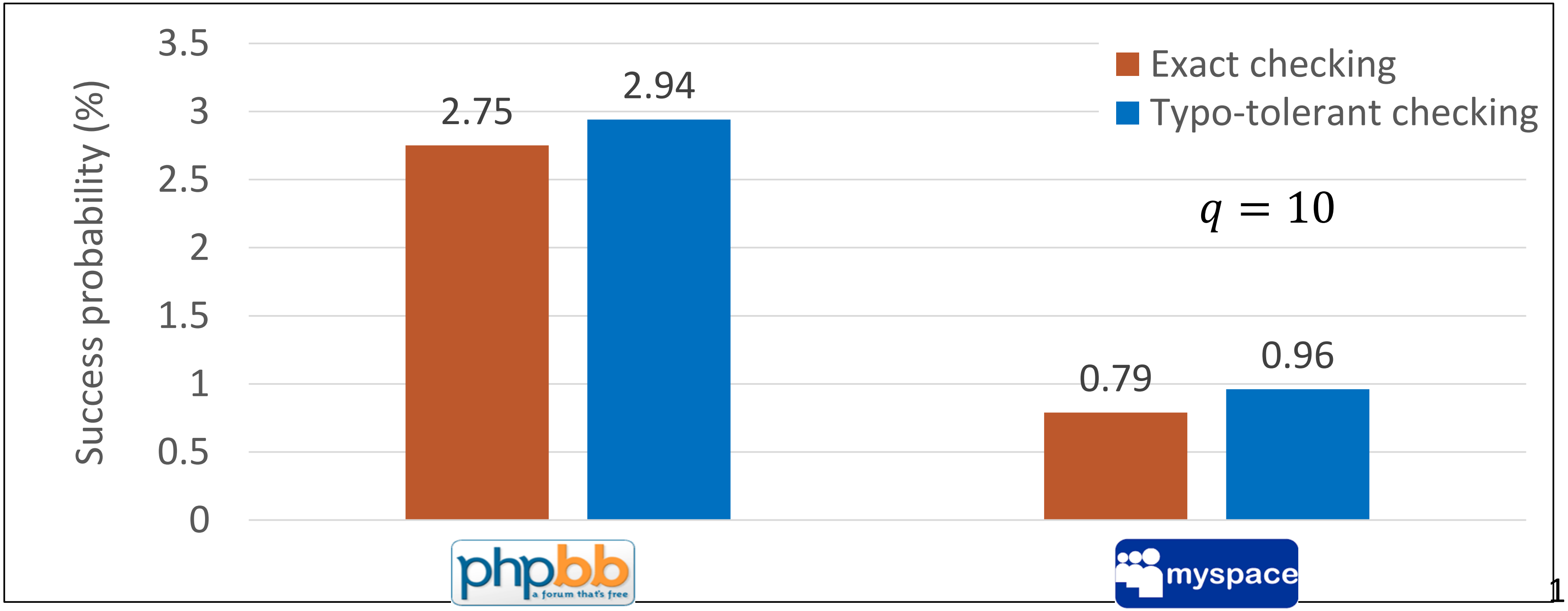
Distribution of passwords, and the set of correctors (Top 3)

Exact checking

Query most probable q passwords

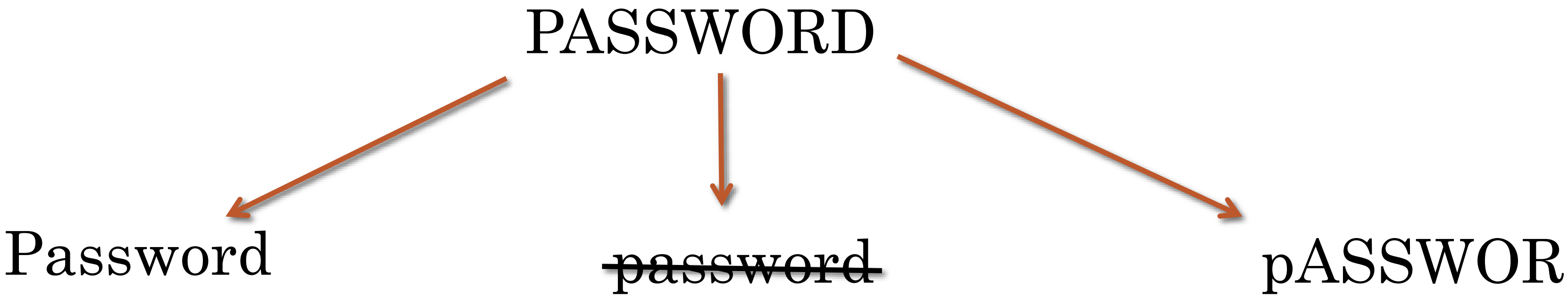
Typo-tolerant checking

Query q passwords that maximizes success. Computed using greedy algo.



Security-sensitive typo correction

Don't check a correction if the resulting password is too popular.



PASSWORD

Password

~~password~~

pASSWOR

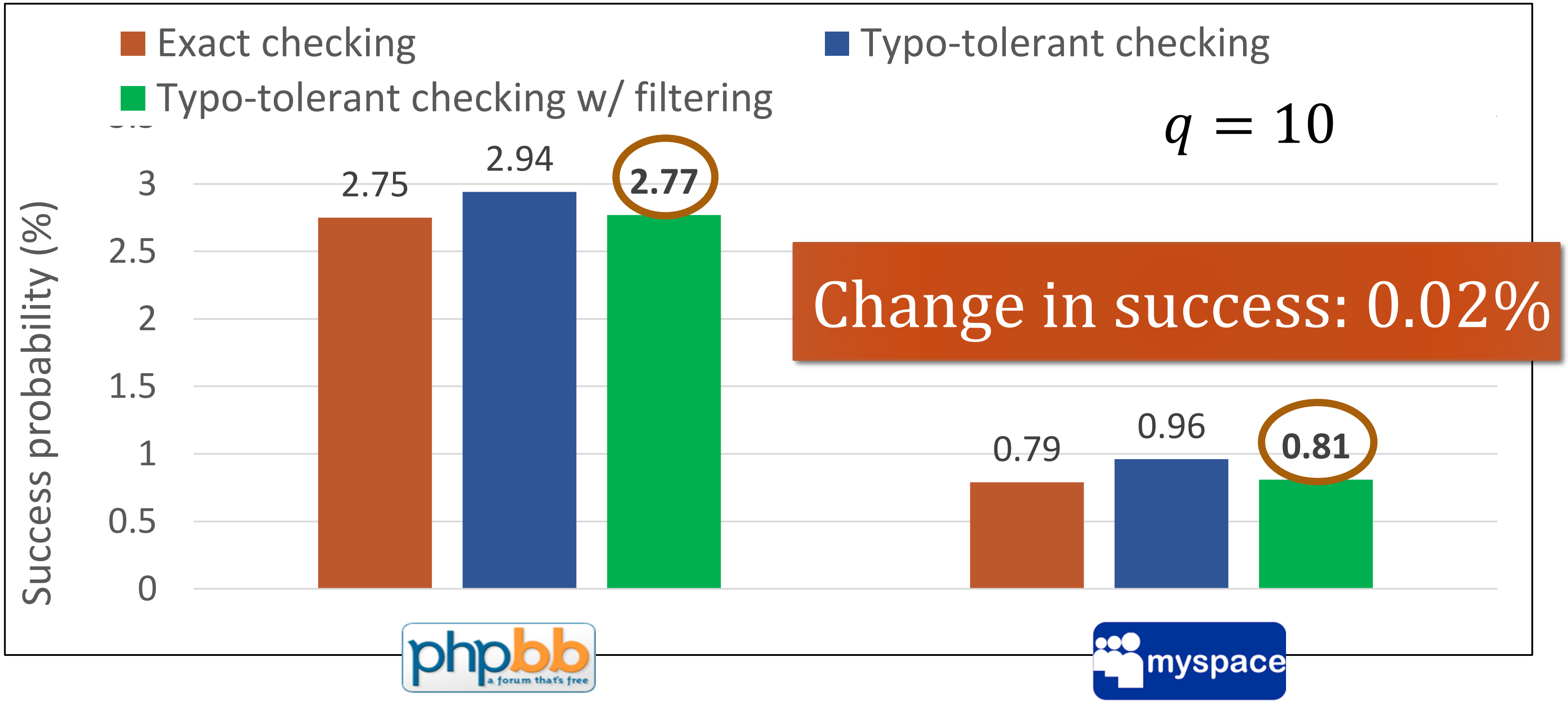
Free Correction Theorem

For any non-uniform password distribution, set of correctors, and adversarial query budget q, there exists a typo correction scheme that corrects typos with no degradation in security.

Security of checkers with filtering

Correct typo ensuring that total probability of all checked password is less than $\Pr[pw_q]$.

Estimated password distribution with rockyou



Change in success: 0.02%

Typo-tolerant checking can enhance users’ experience for essentially no degradation in security.

pASSWORD tYPOS in one slide

1. Introduce typo-tolerant password checkers
   - Compatible with existing password databases, easy to deploy

2. Study password typos empirically
   - 3% of users fail to login due to correctable, top-3 typos

3. Analyze security of typo-tolerant checkers
   - "Free" correction theorem (In theory)
   - With heuristic, works in practice too

GitHub/rchatterjee/mistypography

Thanks!

rahul@cs.cornell.edu