

username

admin

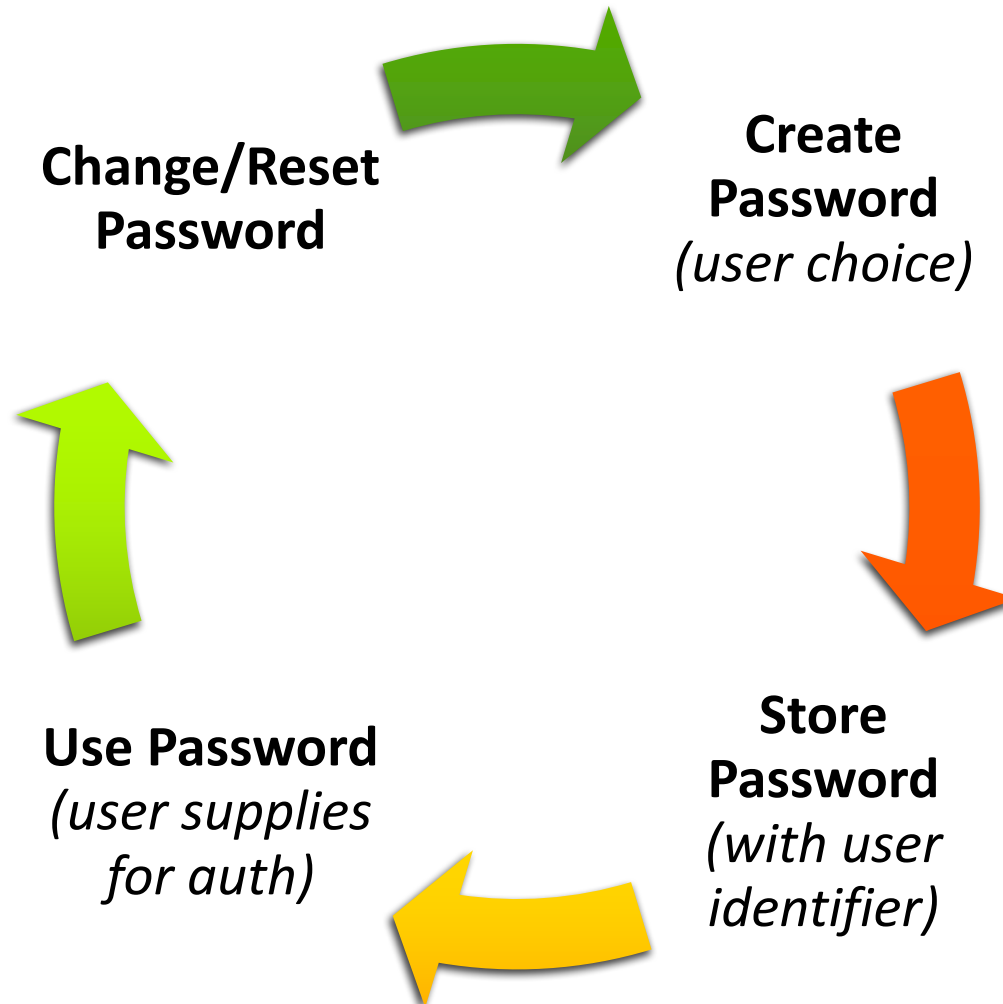
password

* * * * *

Passwords



Password Lifecycle



Password Creation



Password Creation

- **Who creates passwords?**
- **User:** typically guessable passwords
- **System:** can produce hard-to-guess passwords (e.g., random ASCII character strings)
 - But users can't remember them
- **Administrators:** Same as above

User Passwords

➤ Top-10 Most Common Passwords of 2016

- 123456
- 123456789
- qwerty
- 12345678
- 111111
- 1234567890
- 1234567
- Password
- 123123
- 987654321

➤ Users pick terrible passwords!

➤ (duh)

<https://blog.keepersecurity.com/2017/01/13/most-common-passwords-of-2016-research-study/>

Password Strength

- Strength = Resistance to Brute Force
 - High entropy = high resistance
 - If 2^X guesses are required, entropy is X
- Example: Password of length L from alphabet of N characters
 - N^L possible passwords
 - $2^X = N^L \rightarrow X = L \log_2 N$
- NIST recommendations (2006)
 - 14 bits minimum entropy, 30 bits better...

Password Strength

- Example: 8 character password, 26 character alphabet
 - Entropy = $8 \log_2 26 = 37$ bits
 - **So are we good?**
- Huge problem – *real* humans are not choosing uniformly random characters for their passwords
 - How about imposing some rules on passwords the users can select?



Rules

1. The password must be **exactly** 8 characters long.
2. It must contain **at least** one letter, one number, and one of the following special characters.
 - a. The **only** special characters allowed are: @ # \$
 - b. A special character must **not** be located in the first or last position.
3. Two of the same characters sitting next to each other are considered to be a "set." No "sets" are allowed. **Example:** rr, tt
4. Avoid using names, such as your name, user ID, or the name of your company or employer.
5. Other words that cannot be used are Texas, child, and the months of the year.
6. A new password cannot be too similar to the previous password.
 - a. **Example:** previous password - abc#1234; unacceptable new password - acb\$1243
 - b. Characters in the first, second, and third positions cannot be identical. (abc****)
 - c. Characters in the second, third, and fourth positions cannot be identical. (*bc#****)
 - d. Characters in the sixth, seventh, and eighth positions cannot be identical. (*****234)
7. A password can be changed voluntarily (no Help Desk assistance needed) once in a 15-day period. If needed, the Help Desk can reset the password at any time.
8. The previous 8 passwords cannot be reused.

One way to create a password is creative spelling and substitution. Examples:

1. phuny#2s
2. fish#1ng
3. t0pph@ts
4. run\$4you
5. ba#3ries

[Top of page](#)

Password Recipes

Attorney General of Texas, Child Support Division

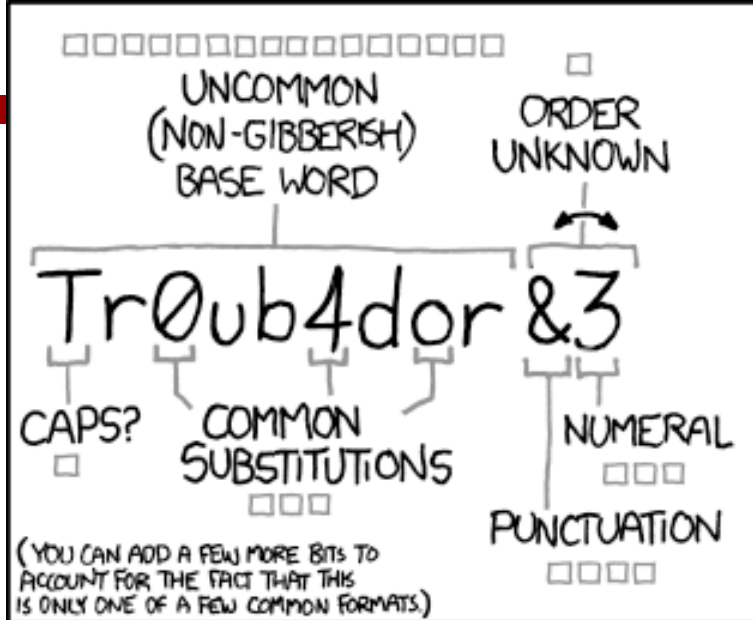
<http://portal.cs.oag.state.tx.us/OAGStaticContent/portal/login/help/listPasswordRules.htm>

Password Recipes

- Will password rules help entropy?
 - Users are annoyed and choose weaker passwords 😞
 - Users pick easy to guess passwords that minimally comply with recipe 😞
- **Warning!** The attackers know all of your clever password tricks, and program their brute force attempts to try these permutations!
 - Only an *idiot* attacker would brute force
aaaaaaaaaaaaaaaaaaaaa to zzzzzzzzzzzzzzzzzzz

Password Creation

- What if the system adds some randomness at the beginning or end of the user password? (and user must remember it all)
 - Users choose weaker base passwords 😞
- Password wallets / Password managers
 - Pro: Have truly random + unique passwords 😊
 - Con: Have to trust password manager 😞
- Passphrases instead of passwords?



~28 BITS OF ENTROPY

□□□□□□□□
□□□□□□□□
□□□
□□□□


$2^{28} = 3 \text{ DAYS AT}$
1000 GUESSES/SEC

(PLAUSIBLE ATTACK ON A WEAK REMOTE
WEB SERVICE. YES, CRACKING A STOLEN
HASH IS FASTER, BUT IT'S NOT WHAT THE
AVERAGE USER SHOULD WORRY ABOUT.)

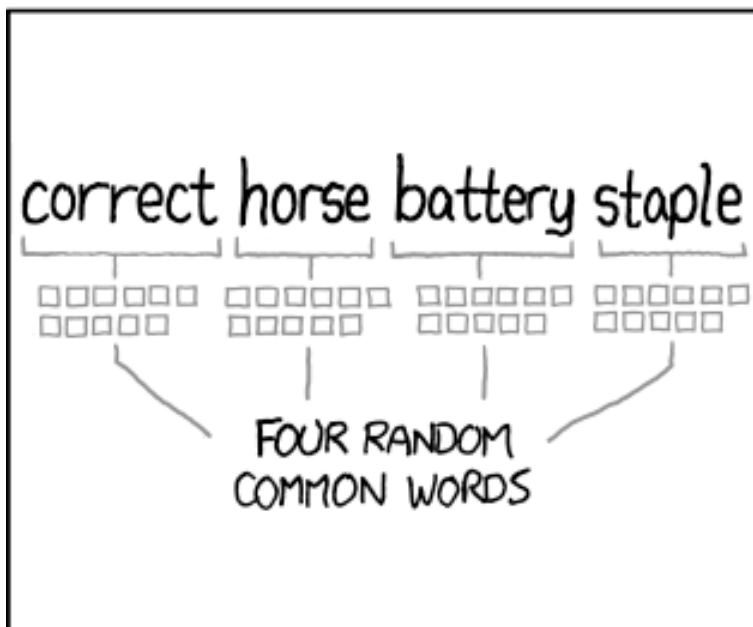
DIFFICULTY TO GUESS:
EASY

WAS IT TROMBONE? NO,
TROUBADOR. AND ONE OF
THE 0s WAS A ZERO?

AND THERE WAS
SOME SYMBOL...



DIFFICULTY TO REMEMBER:
HARD



~44 BITS OF ENTROPY

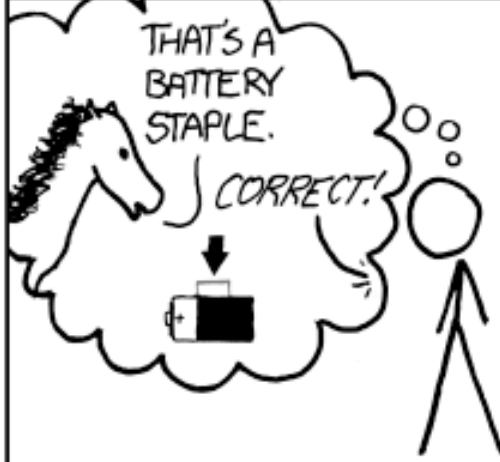
□□□□□□□□□□
□□□□□□□□□□
□□□□□□□□□□
□□□□□□□□□□

$2^{44} = 550 \text{ YEARS AT}$
1000 GUESSES/SEC

DIFFICULTY TO GUESS:
HARD

THAT'S A
BATTERY
STAPLE.

CORRECT!



DIFFICULTY TO REMEMBER:
YOU'VE ALREADY
MEMORIZED IT

THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED
EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS
TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

"XKCD Method"

- Good analysis of XKCD method math for Tr0ub4dor&3
 - <https://blog.agilebits.com/2011/08/10/better-master-passwords-the-geek-edition/>
- Passphrase assumption:
 - Get a dictionary of 2^{11} easy to spell English words
 - Pick 4 of them at **RANDOM**
 - Hence, 2^{44} combinations to brute force (44 bits of entropy)
 - Few days on a GPU via Hashcat? (for non-KDF hashes)
- Is it as good as a truly random 30 character password? No. That would be $30 \log_2(26) = 141$ bits of entropy.
 - But it's much much better than the password your mom usually picks

Kerckhoff's Principle

- Simplified version by Claude Shannon
 - “The enemy knows the system”
- Assume adversary knows everything about your password generation scheme (no secret methods!)
- Only safety is via high entropy and many (many!) brute-force combinations

Password Storage



Storage by Humans

- To keep identities independent, humans should have separate password for every identity
- But humans have scarce memory capacity
- Humans instead
 - Reuse passwords across systems
 - Record passwords (physically, digitally)

Storage by Machines

- **What are the best practices to store user passwords in your system?**

Password Usage



Authentication Fails



The credentials you provided cannot be determined to be authentic.

Enter your PacificNet ID and Password

PacificNet ID:

president_eibeck

Password:

•••••

☐ Warn me before logging me into other sites.

LOGIN clear

[Admitted Students - click here for assistance logging in.](#)

[Parent Information](#)

[Need help logging in?](#)

- **Guiding principle: the system might be under attack, so don't make the attacker's job any easier**
- **Don't leak valid usernames**
 - Prompt for username and password in parallel
 - Don't reveal which was bad

When Authentication Fails

- **Guiding principle: the system might be under attack, so don't make the attacker's job any easier**
- Rate limit, and eventually disable
- Record failed attempts and review [audit]
 - Automate review by administrators?
 - Manually by user at next successful login?

Mutual Authentication

- Before entering their password, the user ought to be authenticating the system itself:
Mutual authentication
- Mechanism - Visual secrets
 - User and system share a secret image
 - User enters username, system retrieves and displays image
 - User authenticates image before entering password
 - Makes phishing attacks harder but not impossible: if users can't or won't discern who is on the other side, man-in-the-middle attack will succeed anyway

Login Spoofing

- What prevents a malicious program that can write to the entire screen from producing a pixel-perfect replica of a login prompt and capturing user credentials?
- **Secure attention key / sequence**
 - Traps directly to OS (bypassing applications)
 - Ctrl+Alt+Del in Windows
 - Alt+SysRq+K in Linux
 - Mutual authentication – Confidence that password prompt is legitimate

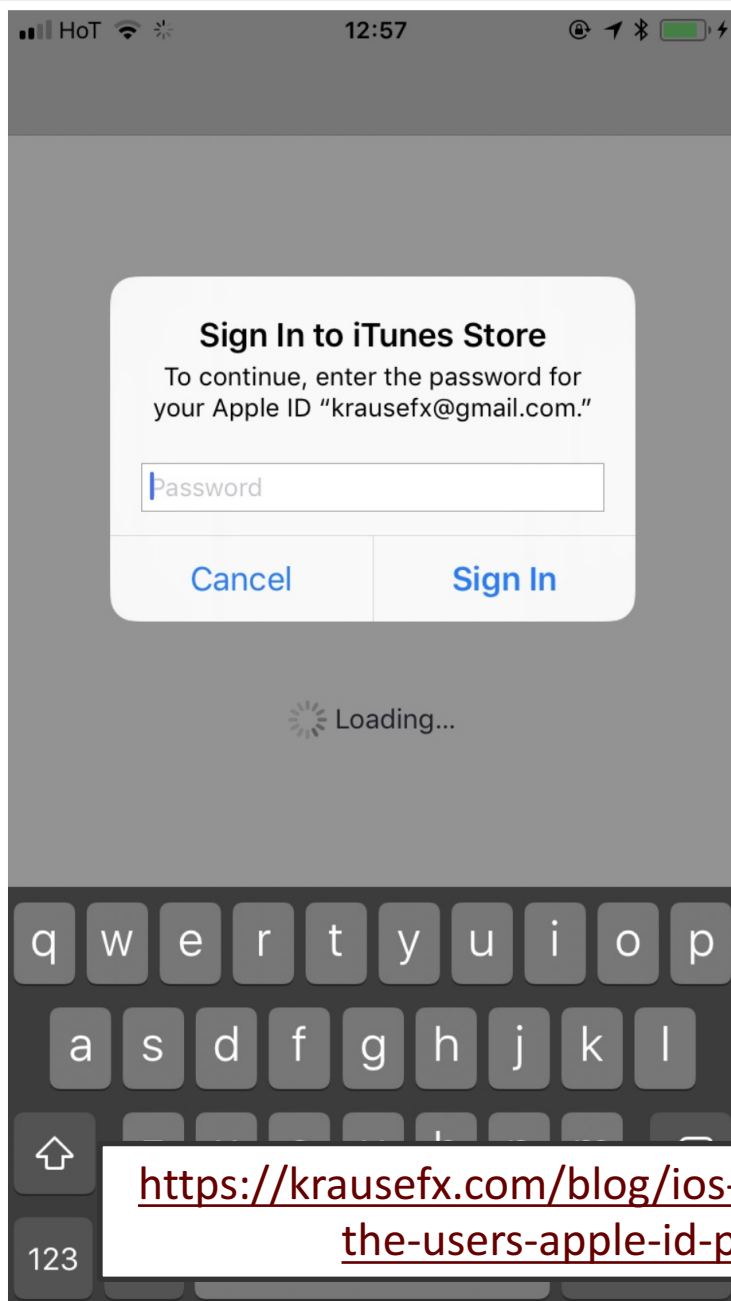
Press CTRL + ALT + DELETE to log on



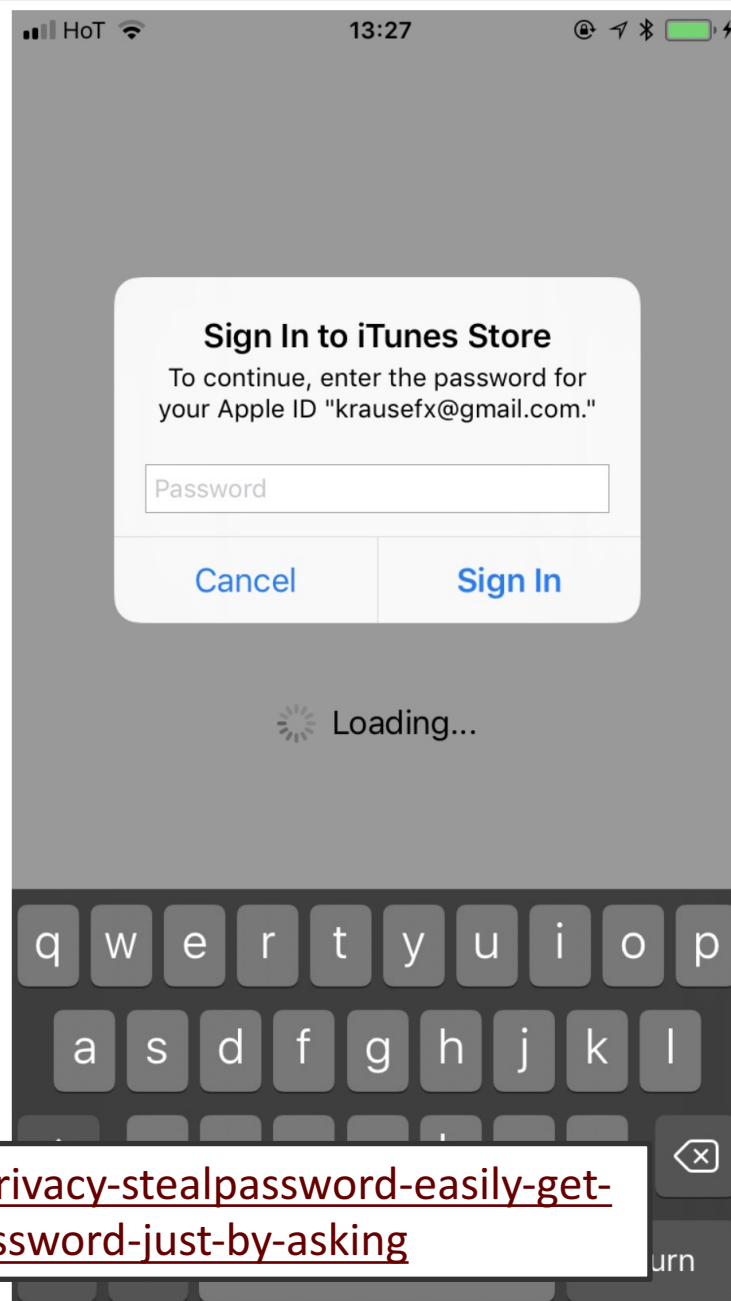
Windows 7 Ultimate

Login Spoofing

Official popup

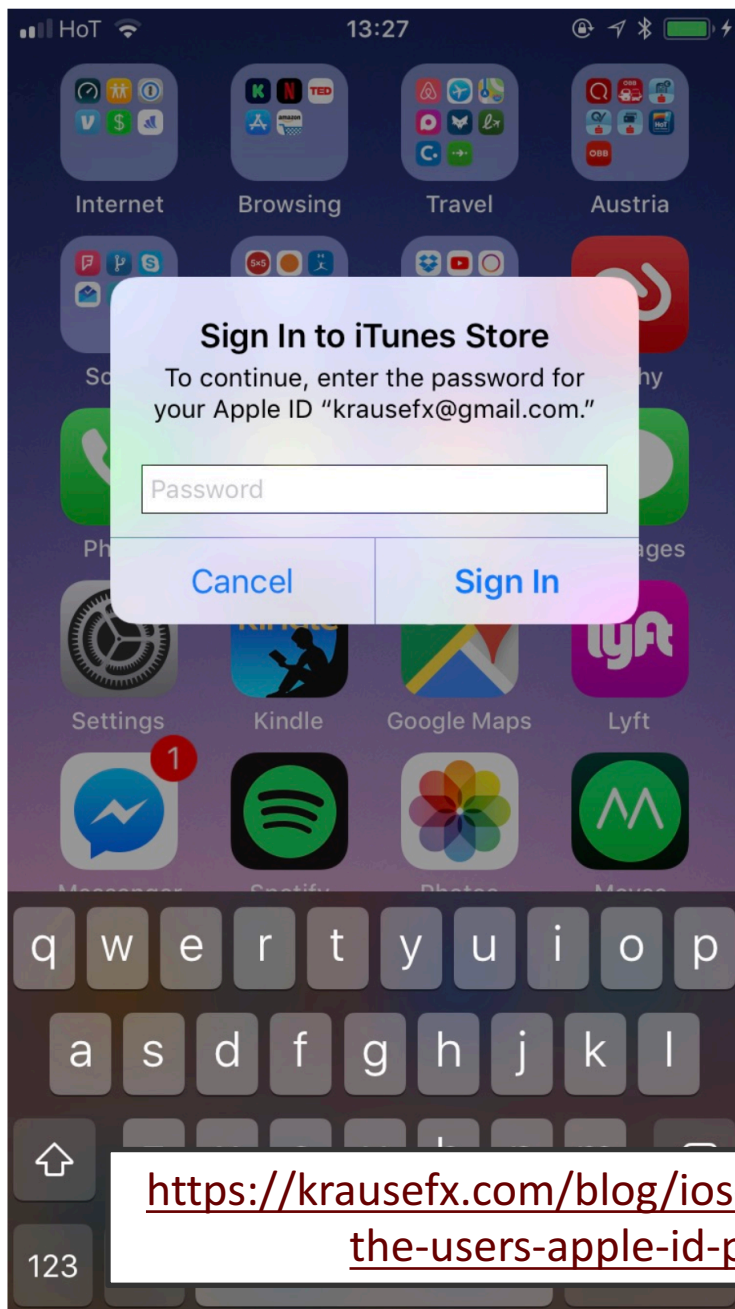


Phishing popup

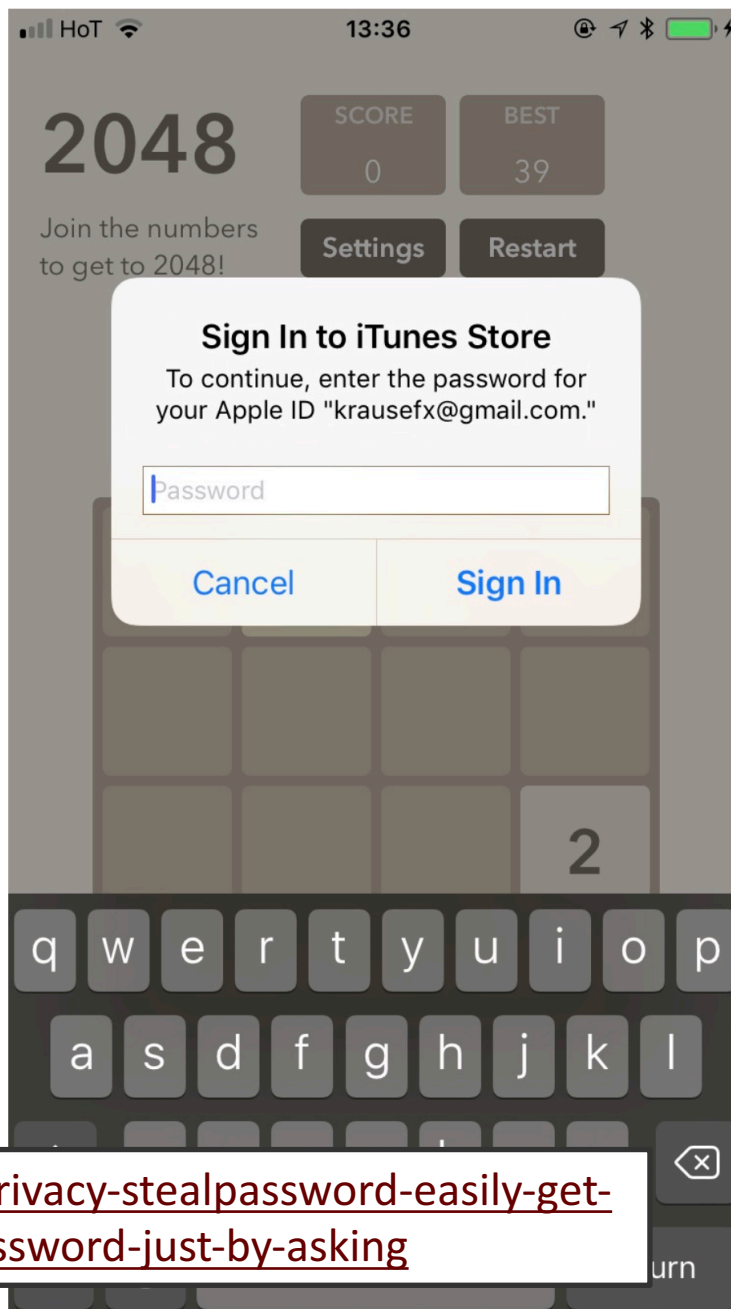


<https://krausefx.com/blog/ios-privacy-stealpassword-easily-get-the-users-apple-id-password-just-by-asking>

Random system popup



Phishing popup in app/game



24

Login
Spoofing

<https://krausefx.com/blog/ios-privacy-stealpassword-easily-get-the-users-apple-id-password-just-by-asking>

Password Change/Recovery



Password Change

- Motivated by...
 - **User** forgets password
 - Just recover password?
 - **System** forces password expiration (e.g. 6 month interval)
 - **Good idea or bad idea?**
 - When users change passwords, they change them predictably (Passw@rd01, Passw@rd02, ...)
 - Foreknowledge of password expiration motivates users to choose weaker passwords

Password Change

- Motivated by...
 - **Administrator** forces password change
 - Intrusion or weak password detected?
 - **Attacker** learns password
 - Social engineering: deceitful techniques to manipulate a person into disclosing information
 - Online guessing: attacker uses authentication interface to guess passwords
 - Offline guessing: attacker acquires password database for system and attempts to crack it

Password Reset Mechanisms

- Tend to be more vulnerable than the rest of the authentication system
 - Not designed or tested as well
 - Have to solve the authentication problem without the benefit of a password!
- Two common mechanisms
 - Security questions
 - Emailed passwords

Security Questions

- Something you know: attributes of identity established at enrollment
- Pro: you are unlikely to forget answers
 - *Assumes attacker is unlikely to be able to answer questions*
- Cons:
 - Might not resist targeted attacks
 - Same answers re-used in many systems (one data breach risks logins at other sites)

Emailed Password

- New temporary password valid for single use only
- Something you know: emailed password
- **Assumes:** attacker is unlikely to have compromised your email account
- **Assumes:** email service correctly authenticates you
- Something you <?>: how did you authenticate to your email system?

Later in this unit: Discuss how Google *Advanced Protection Program* attempts to address these problems...