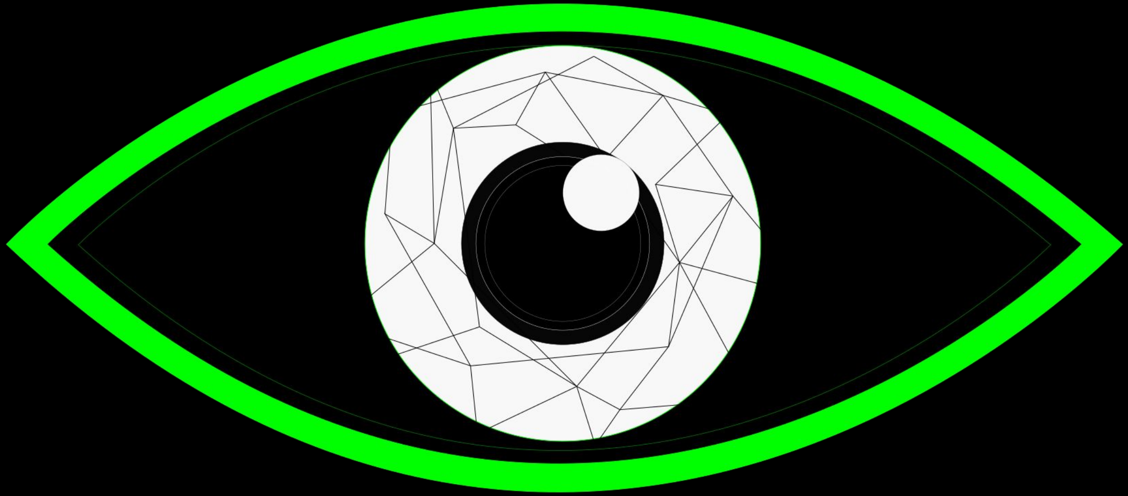


Your Password Sucks

On Passwords, Multi-Factor Authentication, and
the Importance of a Security Mindset

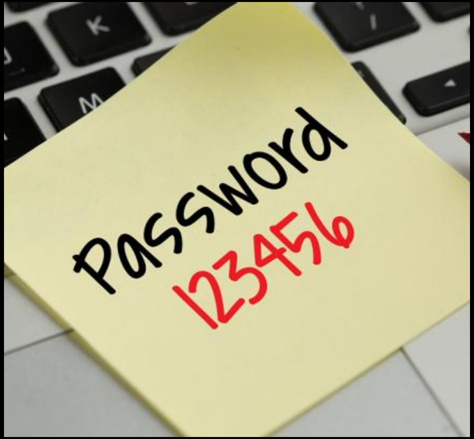
About Me

- Robert Babaev
- 3rd year BCS
- Computer and Internet Security
- VP Events for Carleton Cyber Security Club
- Been in security for 2 years
- Contact will be at the end

Here's the thing.

The Problem With Passwords

- Supposed to be hard to guess, easy to remember
  - Humans really *really* suck at doing that
  - Inadvertently do hard/hard or easy/easy
  - Repeat passwords
  - Bad password constraints
- Hash grabbed = ***bad***
  - Hashes can be cracked in seconds
  - If you only have a password, you're done
- Annoying
  - You're telling me I have to enter a password?
  - I need to *do things!*



The Human Factor

- Hard to guess, easy to remember
    - Easy/easy: Easy to guess, obviously bad
    - Hard/hard: May have to write password down
- Humans tend to prefer short, easy to type things
    - Password crackers *also* prefer short, easy to type things
- Humans tend to like to reuse passwords
    - If one account is compromised, every account is
- Humans not robots
    - Technology (and password-cracking tech) is improving rapidly

Case in point,
human-generated passwords
are out of date.

What the hash?

Hashes

- Way of “securely” storing sensitive data
- One way math function, unlike encryption
- Every* piece of data has a unique hash
- Makes sense to use them for password checking
  - Input text, scramble it, check against existing hash
  - Equivalency comparisons are really easy
  - Have to brute force to guess solutions

*Hashes can’t actually be truly unique, but they can get pretty close!

A hash’s worst enemies

- Phishing
  - Get an email asking you to login
  - Most common way of getting passwords
- Database Leaks
  - Done through multitude of attacks, though the least troublesome is SQL injection
- Password crackers + poorly designed apps
  - John the Ripper, Hydra, etc.
  - Dictionary attacks (wordlists) or Brute-force
  - Database leaks -> wordlists
- With good enough hardware, can brute force (not dictionary attack) complex 14 char password in under a minute
  - Luckily not accessible to the average person
  - If password is good enough, hacker moves on

Solution to making good passwords

- Password managers
  - Keep track of multiple passwords
  - Come with generators, autosave, autofill
  - One master password you remember, enter every now and again
  - Dashlane, Bitwarden
  - Crank up the settings as high as the site allows
- Master password (weakest link)
  - 25 characters or longer
  - Use combinations of words
  - Throw in some random letters/numbers/symbols at low frequency
    - Don’t use 1337speak, be more random
    - (Advanced) Use a script
  - Easier to remember one hard password than multiple

What if that fails?

The Case for 2FA

- If your password is stolen through social engineering
  - Lose your account
  - Password reset, done
  - If you reuse that password, all accounts are compromised
- Second line of defense
  - Multi-factor authentication
    - SMS message
    - Email
    - Hardware
    - Dedicated applications (Twilio Authy)
  - Something you know, something you have, something you are
  - Best line of defense in world of passwords
  - Enable it on important accounts at the VERY least

The hassle

Ways to avoid hassle (and be secure)

- Use a password manager (autofill, able to discern phishing forms from real ones)
- 2FA methods painless in conjunction with a password manager
- Hardware-based methods (Ubikey, key cards)
  - Useful in medical/high risk professions
  - Enter a password, use key, key good for 24 hours

Unfortunately, companies aren’t always secure

- Cost
- Misplaced priorities
- Incompetence
- Tired, underpaid, frustrated employees
- Just do what you can on your end
    - Try to convince people to follow the same route

Security Mindset

- What’s the exploit?
- How can this be broken?
- What can I do to prevent exploits from being effective?
- Never assume honesty
    - Standard rules apply
    - Check your links and messages
    - If it seems too good to be true, it is
    - If someone is asking you to login, double check the link
    - If something seems off, chances are you’re right

Questions?

Thanks for listening!

Resources

https://www.dashlane.com/

https://bitwarden.com/

https://authy.com/guides/googleandgmail/

https://cheapsslsecurity.com/blog/decoded-examples-of-how-hashing-algorithms-work/

https://www.schneier.com/blog/archives/2008/03/the_security_mi_1.html