Lecture 11 – Authentication

University of Illinois

ECE 422/CS 461

Announcements

- Midterm exam is Thursday May 13 in class
  - 75 minutes
  - Mostly multiple choice + a few short answer
  - We will use Scantron; bring pencil & erasers
  - Open note, closed-device

  - All lectures, discussions, and MPs are covered

Goals

- By the end of this lecture you should:
  - Know the three ways of verifying identity
  - Understand the tradeoffs in choosing passwords
  - Follow the best practices in storing passwords
  - Be able to evaluate the pros and cons of biometrics, tokens, and trends in authentication

Authentication Basics

- A central component of many systems, notably OS and websites
- Involve three processes
  - Registration: establish (identity, proof) to system
  - Verification: user submits (identity, proof) and system verifies
  - Recovery: when user loses proof of identity

Authentication Methods

- Something you know (e.g., password)
- Something you have (e.g., credit card)
- Something you are (e.g., fingerprint)

Passwords

Passwords

- User memorizes a secret string
- Advantages?
- Disadvantages?

Attacks: Guess Weak Passwords

- Weak passwords: default, derived from username, real identity, social connection, …
- Dictionary attack: popular passwords

  https://en.wikipedia.org/wiki/Wikipedia:10,000_most_common_passwords

  - 123456, password, 12345678, qwerty, 123456789, 12345, 1234, 111111, ……

- Defense 1: guide/force users to choose strong passwords

Strong Passwords

- Force/guide users to choose strong passwords
  - Must have upper/lower case, special characters?
  - Is Password123! a good password?

- What makes a password strong?
- Passwords ideally should be long and uniformly distributed
  - All characters appear with equal probability

Security vs. Convenience

- Strong passwords are difficult to remember
- Passwords should not be reused
  - An average person in US has 130 accounts
    https://digitalguardian.com/blog/uncovering-password-habits-are-users-password-security-habits-improving-infographic
- Force user to change passwords regularly?

- No clear winning strategy

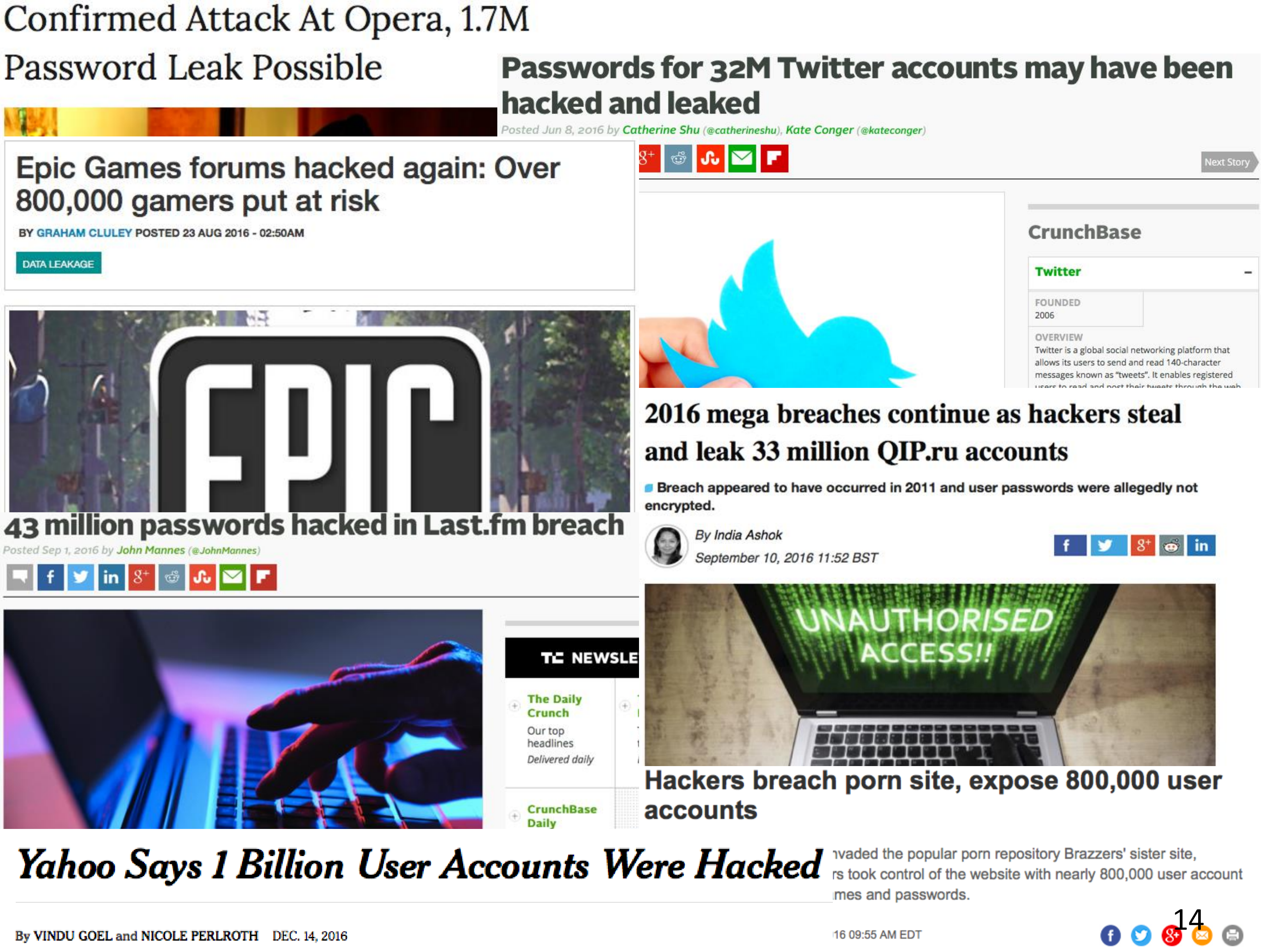
Defenses for Weak Passwords

- Defense 1: guide/force users to choose strong passwords
  - Faced with the fundamental tradeoff between security and convenience

- Defense 2: rate-limit authentication attempts
  - Implemented by most systems and websites

Attack: Steal Passwords

- From user (written down, phishing, keylogger)
- From service (vulnerability, insider)
- From other service (password reuse)



Confirmed Attack At Opera, 1.7M Password Leak Possible

Passwords for 32M Twitter accounts may have been hacked and leaked

Posted Jun 8, 2016 by Catherine Shu (@catherineshu), Kate Conger (@kateconger)

CrunchBase

Twitter –

FOUNDED
2006

OVERVIEW
Twitter is a global social networking platform that allows its users to send and read 140-character messages known as "tweets". It enables registered users to read and post their tweets through the web

Epic Games forums hacked again: Over 800,000 gamers put at risk

BY GRAHAM CLULEY POSTED 23 AUG 2016 - 02:50AM

DATA LEAKAGE

2016 mega breaches continue as hackers steal and leak 33 million QIP.ru accounts

Breach appeared to have occurred in 2011 and user passwords were allegedly not encrypted.

By India Ashok
September 10, 2016 11:52 BST

43 million passwords hacked in Last.fm breach

Posted Sep 1, 2016 by John Mannes (@JohnMannes)

TC NEWSLE

The Daily Crunch
Our top headlines
Delivered daily

CrunchBase Daily

Hackers breach porn site, expose 800,000 user accounts

ıvaded the popular porn repository Brazzers' sister site, rs took control of the website with nearly 800,000 user account ımes and passwords.

16 09:55 AM EDT

Yahoo Says 1 Billion User Accounts Were Hacked

By VINDU GOEL and NICOLE PERLROTH DEC. 14, 2016

NEVER store plaintext passwords!

Store password hashes instead

Cryptographic Hash Functions

- Input – data of an arbitrary length
- Output – fixed length, e.g., 256 bits
- Same input always produces the same output
- Each output looks “random”
- Hard to deduce input from output

- Examples: MD5, SHA1, SHA2, SHA3
- SHA3-256("welcome") = 64db51f8f79ca7ec522a6b4a e5fc7e896daac5318b2e82730d7c7926b66d36eb

Password Hashes

- System stores (username, hash(pw))
- User submits (username, pw)
- System computes hash(pw) and compares

- When system gets compromised, attacker gets (username, hash(pw)), hard to get pw
- ... right?

Attack: Password Cracking

- Attacker steals hashes of user passwords
- Goal: Find the passwords from the hashes

- Method 1: brute force
  - Guess a password (e.g., from common passwords)
  - Hash it and check if it is one of the pw hashes

Attack: Password Cracking

- Attacker steals hashes of user passwords
- Goal: Find the passwords from the hashes

- Method 2: lookup table
  - Pre-compute hashes of all common passwords and build a lookup table: hash → pw
  - For each pw hash, look up the table
  - Can be reused across attacks

Salting Password

- Generate and store a long and random number (salt) for **each** password
  - System stores (username, salt, hash(pw || salt))
  - User submits (username, pw)
  - System computes hash(pw || salt) and compares

- Effectively a unique hash function for each pw
  - Cannot pre-compute a lookup table now

Salting Password

- Q: Why use a different salt for **each** password?
  - Helps protect common passwords
  - hash(pw || salt1) ≠ hash(pw || salt2)

- Q: Does salt need to be kept secret?
  - No
  - A secret nonce is called *pepper*, not widely used

Passwords in Linux

- /etc/passwd stores public user information, owner = root, permission = -rw-r--r--

```
student:x:1001:1001:,,,:/home/student:bin/bash
username  UID   GID        home directory    shell
```

- /etc/shadow stores password hashes owner = root, permission = -rw-r-----

```
student:$6$ilyOdOy4$fL0UtEM3ToqlTeugk7coSgGHa
dmsh78rmODlTwCntS0hmcPn....:17033:0:99999:7:::
6 = SHA2-512   salt    pw hash         expiry date
```

Password Cracking

- Attacker steals salted hashes of passwords (username, salt, hash(pw || salt))
- Goal: find pw

- Must brute-force each pw individually
  - Guess a pw, hash with salt, compare, repeat
- Is brute-force cracking a concern?

Brute-force Password Cracking

- Still a concern

- Easy to parallelize using arrays of GPUs/FPGAs
- Customized hardware (ASIC) password cracker also conceivable

Brutalis

Highlights

1. World's fastest 8-GPU system -- 14% faster
2. First system to break 330 GH/s on NTLM --
3. First system to break 200 GH/s on MD5!

Base configuration price: 21,169.00 USD

Bitmain
Antminer S17 (56Th)

Model **Antminer S17 (56Th)** from **Bitmain** mining **SHA-256 algorithm** with a maximum hashrate of **56Th/s** for a power consumption of **2520W.**

Password Strength against GPU/ASIC

- 8-character alphanumeric combinations: (26 + 26 + 10) ^ 8 ≈ 200 trillion
- GPU password cracker: 1 TH/s → 200 seconds
- ASIC password cracker: 56 TH/s → 4 seconds
- What can we do about this?

Password Hashing

- Hash functions for passwords should be slow!
  - Opposite to everything else in CS

- First idea: repeated hashing
  - hash(hash(…(hash(.)…))
- crypt (in Linux) loops 5000 times for SHA2
- bcrypt loops a configurable number of times

Password Hashing

- Repeated hashing slow down cracking and authentication equally
  - Cracking: 4s → 20,000s; authentication: 1μs → 5ms

- Can we do better? Can we reduce the advantage of specialized hardware?
- What makes specialized hardware fast?
  - Specialized computation is much cheaper (hence, can put many) and faster than generic computation

Password Hashing

- Memory-hard functions for password hashing
  - Require large memory space and frequent random memory accesses during computation
  - Random memory accesses are no cheaper for specialized hardware

- Notable examples
  - Scrypt: first memory-hard function proposal (2009)
  - Argon2: won Password Hashing Competition (2014)

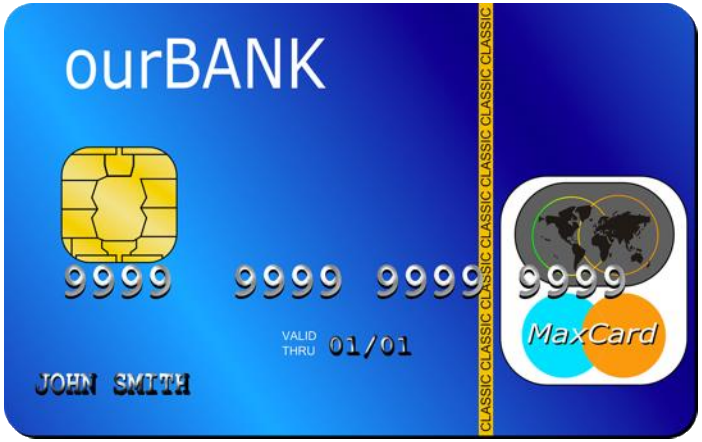
Best Practices in Passwords

- Use strong passwords
- Rate limit authentication attempts
- Store password hashes (never store plaintext)
- Salt the passwords (to defeat lookup tables)
- Use slow or memory-hard hash functions (to slow down cracking)

Other Authentication Methods

Token-based Authentication

- Something the user has
  - ATM card / credit card
  - Smart card
  - Hardware token



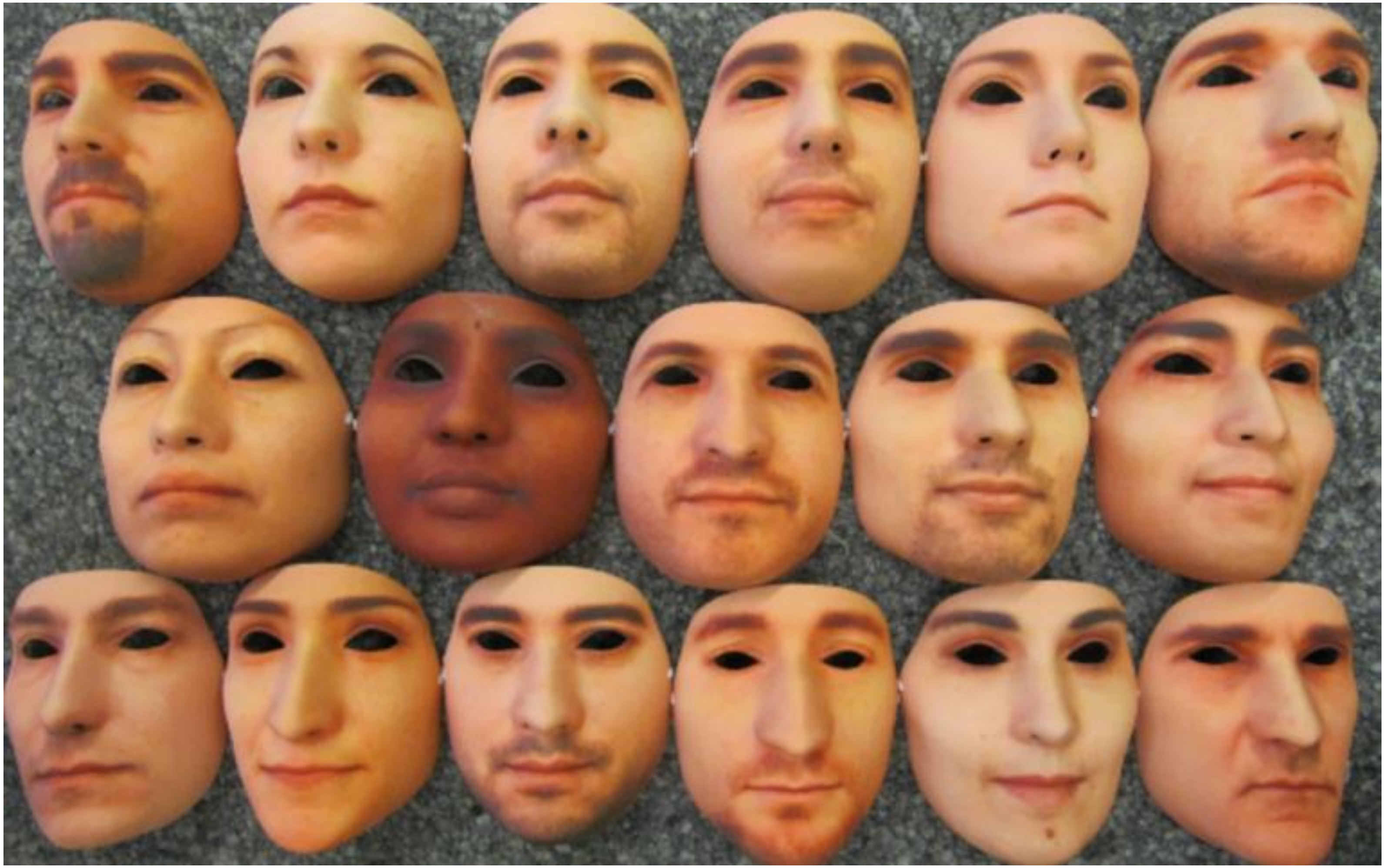
Token-based Authentication

- Something the user has
- Advantages?
  - Each token can carry a strong secret
  - No weak secret, no reused secret
- Disadvantages?
  - Token can be lost, damaged or stolen
  - Less convenient, need to carry token
  - Costs of manufacturing/distributing

Biometric Authentication

- Something the user is
  - Voice, fingerprint, face, iris, …
- Advantages?
  - Convenient
- Disadvantages?
  - Can have false negatives/positives
  - Cannot be replaced
  - Spoofing possible

Spoofing



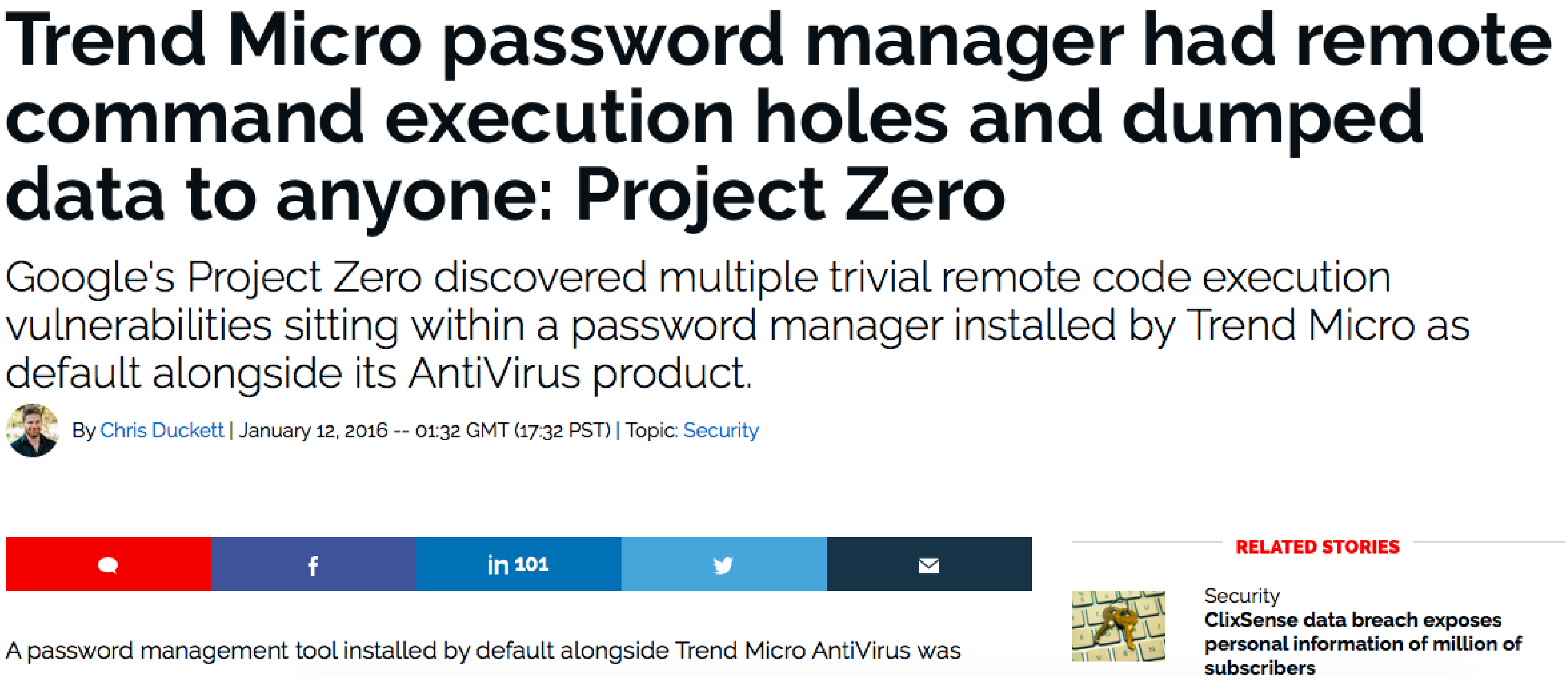
Recent Trends in Authentication

Password Manager

- Application that generates and maintains random and unique passwords for the user
  - Need one strong password for the manager
  - Examples: LastPass, KeePass, DashLane, 1Password

- Advantages and disadvantages?
  - Better security: strong passwords and no reuse
  - Worse security: one point of failure
  - Convenient: memorize only one password
  - Inconvenient: doesn’t work for some sites

One Point of Failure



Trend Micro password manager had remote command execution holes and dumped data to anyone: Project Zero

Google's Project Zero discovered multiple trivial remote code execution vulnerabilities sitting within a password manager installed by Trend Micro as default alongside its AntiVirus product.

By Chris Duckett | January 12, 2016 -- 01:32 GMT (17:32 PST) | Topic: Security

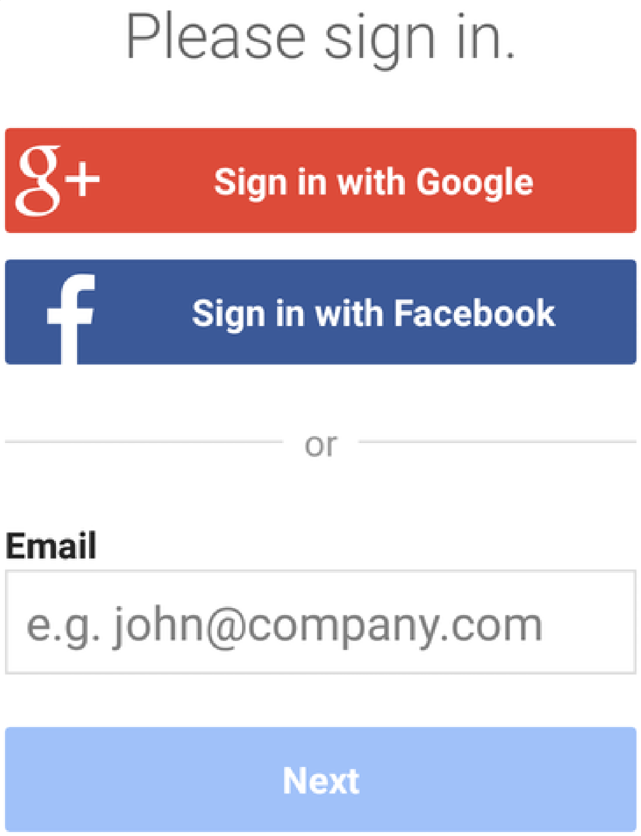
A password management tool installed by default alongside Trend Micro AntiVirus was

Security
ClixSense data breach exposes personal information of million of subscribers

Single Sign-On (SSO)

- Login to trusted 3rd party, who vouches for user identity
  - Examples: Google/Facebook
  - Like a password manager

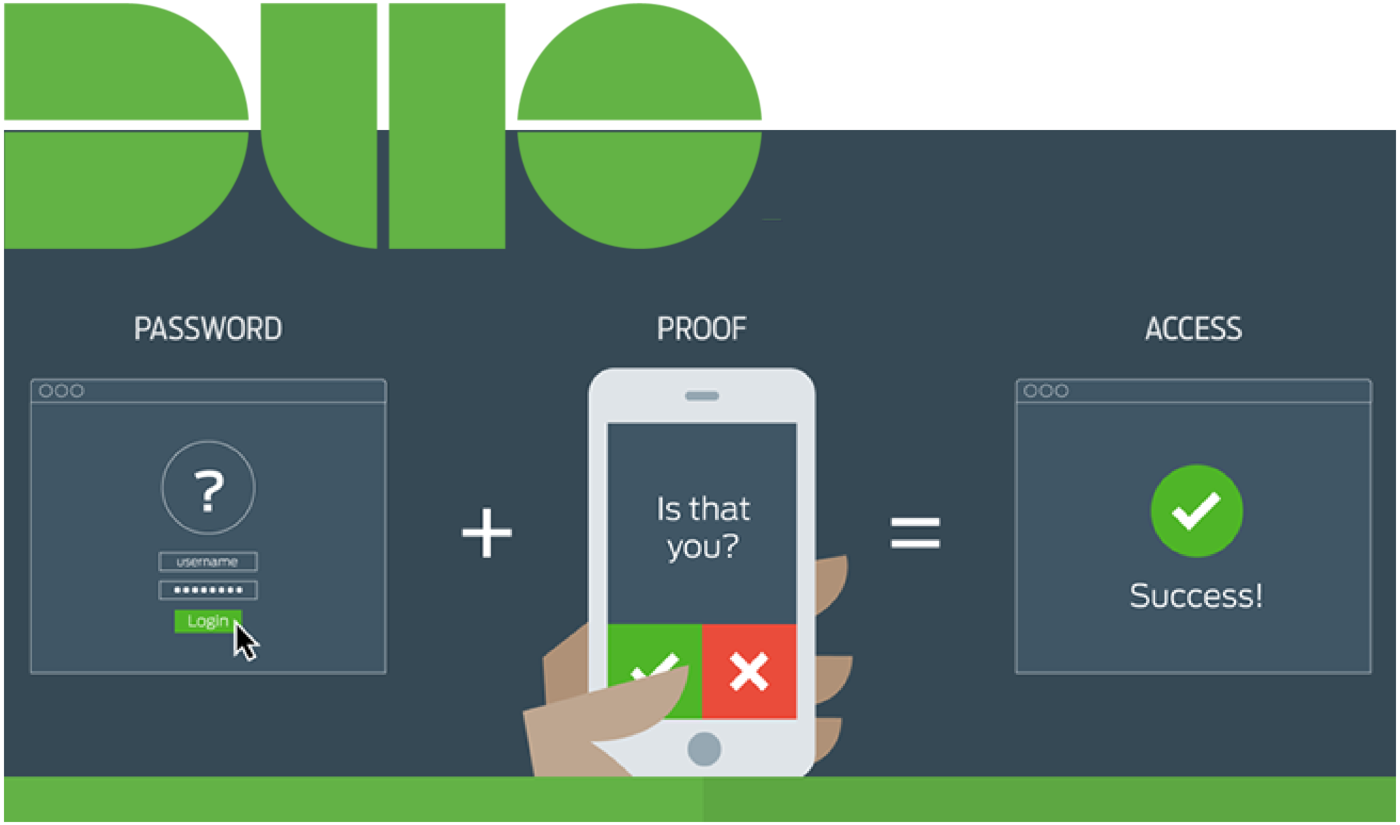
- Pros and cons similar to password managers
  - Third party can track users



Please sign in.

Two Factor Authentication (2FA)

- Combine two of the three ways to authenticate
  - Debit card AND PIN; password AND phone

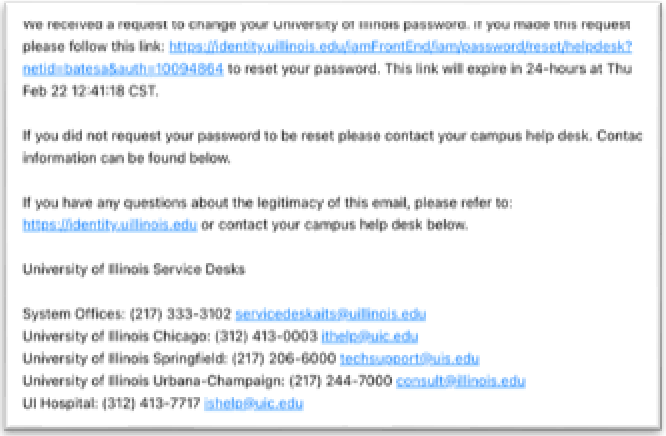


Two Factor Authentication (2FA)

- Combine two of the three ways to authenticate
  - Debit card AND PIN; password AND phone
- Advantages?
  - Harder to compromise account
- Disadvantages?
  - Lost productivity and user experience
  - Loss of availability (lost phone, no service)

Recovery

- Often overlooked
- Can be the weakest link
- Common methods
  - Security questions
  - Recovery email/phone



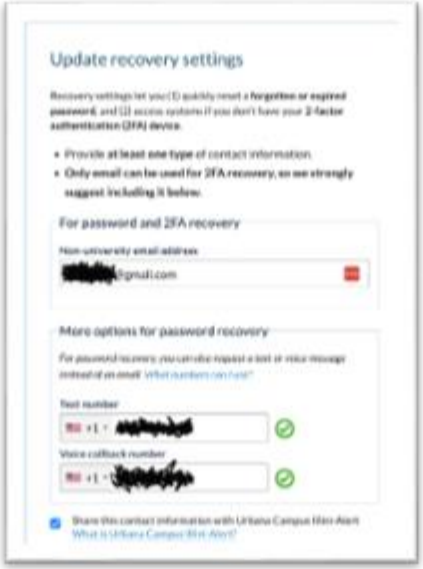
We received a request to change your University of Illinois password. If you made this request please follow this link: https://identity.uillinois.edu/iamFrontEnd/iam/password/reset/helpdesk?netid=batesa&auth=10094864 to reset your password. This link will expire in 24-hours at Thu Feb 22 12:41:18 CST.

If you did not request your password to be reset please contact your campus help desk. Contac information can be found below.

If you have any questions about the legitimacy of this email, please refer to: https://identity.uillinois.edu or contact your campus help desk below.

University of Illinois Service Desks

System Offices: (217) 333-3102 servicedeskaits@uillinois.edu
University of Illinois Chicago: (312) 413-0003 ithelp@uic.edu
University of Illinois Springfield: (217) 206-6000 techsupport@uis.edu
University of Illinois Urbana-Champaign: (217) 244-7000 consult@illinois.edu
UI Hospital: (312) 413-7717 ishelp@uic.edu



Update recovery settings

Recovery settings let you (1) quickly reset a **forgotten or expired password**, and (2) access systems if you don't have your **2-factor authentication (2FA) device**.

- Provide **at least one type** of contact information.
- **Only email can be used for 2FA recovery, so we strongly suggest including it below.**

For password and 2FA recovery

Non-university email address

[redacted]@gmail.com

More options for password recovery

For password recovery, you can also request a text or voice message instead of an email. What numbers can I use?

Text number

+1 · [redacted]

Voice callback number

+1 · [redacted]

☑ Share this contact information with Urbana Campus Illini-Alert
What is Urbana Campus Illini-Alert?

Summary of Authentication

- Three components: registration, verification, recovery
- Three ways: something user knows/has/is
- Token/biometric authentication has limitations: lost token, biometric inaccuracy, etc.
- Password is still by far the most common
- Recent trends: Password manager, SSO, 2FA

Best Practices in Passwords

- Use strong passwords
- Rate limit authentication attempts
- Store password hashes (never store plaintext)
- Salt the passwords (to defeat lookup tables)
- Use slow or memory-hard hash functions (to slow down cracking)