Access Control: Authentication (Part I)

> It used to be expensive to make things public and cheap to make them private. Now it's expensive to make things private and cheap to make them public.
>
> — Clay Shirky, Internet scholar and professor at N.Y.U.

Tamer ABUHMED
School of Computer and Information Engineering
INHA University

Outline

- Introduction
- **Authentication:** Something You Know (Passwords)
  - Trouble with Passwords
  - Why Passwords?
  - Password construction techniques (Good & Bad)
  - Attacks on Passwords
- **Authentication:** Something You Are (Biometrics)
  - Fingerprint
  - Iris
- **Authentication:** Something You Have (Hardware device)

Access Control

- Two parts to access control
- **Authentication:** Are you who you say you are?
  - Determine whether access is allowed
  - Authenticate human to machine
  - Or authenticate machine to machine
- **Authorization:** Are you allowed to do that?
  - Once you have access, what can you do?
  - Enforces limits on actions

- Note: "access control" often used as synonym for authorization

Are You Who You Say You Are?

- How to authenticate human a machine?
- Can be based on…
  - Something you **know**
    - For example, a password
  - Something you **have**
    - For example, a smartcard
  - Something you **are**
    - For example, your fingerprint

Something You Know

- **Passwords**
- Lots of things act as passwords!
  - PIN
  - Social security number
  - Mother's maiden name
  - Date of birth
  - Name of your pet, etc.

Trouble with Passwords

- "Passwords are one of the *biggest practical problems* facing *security engineers* today."
- "Humans are incapable of securely storing high-quality cryptographic keys, and they have unacceptable speed and accuracy when performing cryptographic operations. (They are also large, expensive to maintain, difficult to manage, and they pollute the environment. It is astonishing that these devices continue to be manufactured and deployed.)"

Why Passwords?

- Why is “something you know” more popular than “something you have” and “something you are”?
- **Cost**: passwords are free
- **Convenience**: easier for admin to reset password than to issue a new thumb (hardware device)

Keys vs Passwords (Comparison)

Crypto Keys

- Size of key is 64 bits
- Then, the key space = 2^{64} keys
- Choose key at random…
- …then attacker must try about 2^{63} keys

Passwords

- Size of passwords are 8 characters, and 256 different characters
- Then $256^8 = 2^{64}$ passwords
- Users do not select passwords at random
- Then, attacker has far less than 2^{63} passwords to try

  (**dictionary attack**)

Good and Bad Passwords

- Bad passwords
  - frank
  - Fido
  - password
  - 4444
  - Pikachu
  - 102560
  - AustinStamp
- Good Passwords?
  - jfIej,43j-EmmL+y
  - 09864376537263
  - P0kem0N
  - FSa7Yago
  - 0nceuP0nAt1m8
  - PokeGCTall150

Password Experiment

- Three groups of users — each group advised to select passwords as follows
  - **Group A:** At least 6 chars, 1 non-letter
  - **Group B:** Password based on passphrase ← winner
  - **Group C:** 8 random characters
- Results
  - **Group A:** About 30% of pwds easy to crack
  - **Group B:** About 10% cracked
    - Passwords easy to remember
  - **Group C:** About 10% cracked
    - Passwords hard to remember

Password Experiment

- User compliance hard to achieve
- In each case, 1/3rd did not comply
  - And about 1/3rd of those easy to crack!
- Assigned passwords sometimes best
- If passwords not assigned, best advice is…
  - Choose passwords based on passphrase
  - Use password cracking tool to test for weak passwords
- Require periodic password changes?

Attacks on Passwords

- Attacker could…
  - Target one particular account
  - Target any account on system
  - Target any account on any system
  - Attempt denial of service (DoS) attack
- Common attack path
  - Outsider → normal user → administrator
  - May only require **one** weak password!

Password Retry

- Suppose system locks after 3 bad passwords. How long should it lock?
  - 5 seconds
  - 5 minutes
  - Until SA restores service
- What are +'s and -'s of each?

Password File?

- Bad idea to store passwords in a file
- But we need to verify passwords
- Cryptographic solution: **hash** the password
  - Store y = h(password)
  - Can verify entered password by hashing
  - If Trudy obtains "password file," she does not obtain passwords
- But Trudy can try a *forward search*
  - Guess x and check whether y = h(x)

Dictionary Attack

- Trudy pre-computes h(x) for all x in a **dictionary** of common passwords
- Suppose Trudy gets access to password file containing hashed passwords
  - She only needs to compare hashes to her pre-computed dictionary
  - After one-time work, actual attack is trivial
- Can we prevent this attack? Or at least make attacker's job more difficult?

Salt

- Hash password with **salt**
- Choose random salt s and compute

  $$y = h(password, s)$$

  and store (s,y) in the password file
- Note: The salt s is not secret
- Easy to verify salted password
- But Trudy must re-compute dictionary hashes for each user
  - Lots more work for Trudy!

Password Cracking: Do the Math

- Assumptions:
- Passwords are 8 chars, 128 choices per character
  - Then $128^8 = 2^{56}$ possible passwords
- There is a **password file** with 2^{10} passwords
- Attacker has **dictionary** of 2^{20} common passwords
- **Probability** of 1/4 that a password is in dictionary
- Work is measured by number of hashes

Password Cracking: Case I

- Attack 1 password without dictionary
  - Must try $2^{56}/2 = 2^{55}$ on average
  - Like exhaustive key search
- Does **salt** help in this case?

Password Cracking: Case II

- Attack 1 password with dictionary
- With **salt**
  - Expected work: $1/4\ (2^{19}) + 3/4\ (2^{55}) = 2^{54.6}$
  - In practice, try all passwords in dictionary…
  - …then work is at most 2^{20} and probability of success is 1/4
- What if **no salt** is used?
  - One-time work to compute dictionary: 2^{20}
  - Expected work still same order as above
  - But with pre-computed dictionary hashes, the "in practice" attack is free…

Password Cracking: Case III

- Any of 2^{10} =1024 passwords in file, **without** dictionary
  - Assume all 2^{10} passwords are distinct
  - Need 2^{55} **comparisons** before expect to find password
- If **no salt** is used
  - Each computed hash yields 2^{10} comparisons
  - So expected work (hashes) is $2^{55}/2^{10} = 2^{45}$
- If **salt** is used
  - Expected work is 2^{55}
  - Each comparison requires a hash computation

Password Cracking: Case IV

- Any of 1024 passwords in file, **with** dictionary
  - Prob. one or more password in dict.: $1 - (3/4)^{1024} = 1$
  - So, we ignore case where no password is in dictionary
- If **salt** is used, expected work less than 2^{22}
  - See book for details
  - Approximate work: size of dict. / probability
- What if **no salt** is used?
  - If dictionary hashes not pre-computed, work is about $2^{19}/2^{10} = 2^{9}$

Other Password Issues

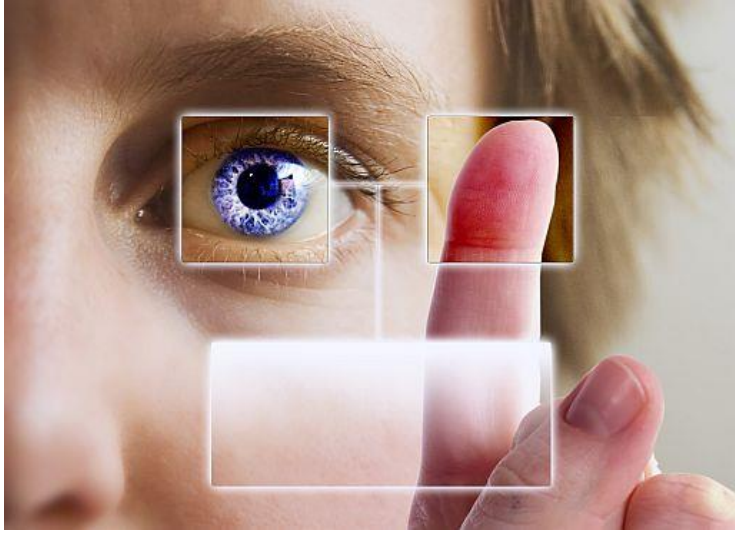
- Too many passwords to remember
  - Results in password reuse
  - Why is this a problem?
- Who suffers from bad password?
  - Login password vs ATM PIN
- Failure to change default passwords
- Social engineering
- Error logs may contain “almost” passwords
- Bugs, keystroke logging, spyware, etc.

Passwords

- The bottom line…
- **Password cracking is too easy**
  - One weak password may break security
  - Users choose bad passwords
  - Social engineering attacks, etc.
- Trudy has (almost) all of the advantages
- All of the math favors bad guys
- Passwords are a **BIG** security problem
  - And will continue to be a big problem

Password Cracking Tools

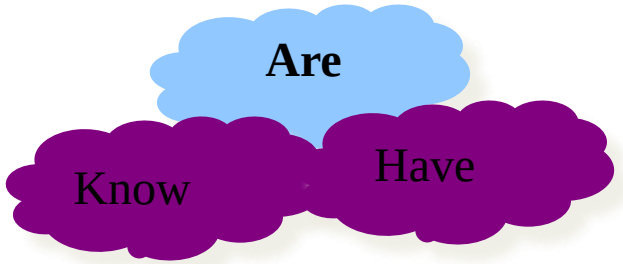
- Popular password cracking tools
  - Password Crackers
  - Password Portal
  - L0phtCrack and LC4 (Windows)
  - John the Ripper (Unix)
- Admins should use these tools to test for weak passwords since attackers will
- Good articles on password cracking
  - Passwords - Conerstone of Computer Security
  - Passwords revealed by sweet deal



Authentication: Biometrics

Something You Are

- Biometric
  - **"You are your key"** — Schneier
  - Examples
    - Fingerprint
    - Handwritten signature
    - Facial recognition
    - Speech recognition
    - Gait (walking) recognition
    - "Digital doggie" (odor recognition)
    - Many more!



Why Biometrics?

- More secure replacement for passwords
- Cheap and reliable biometrics needed
  - Today, an active area of research
- Biometrics **are** used in security today
  - Thumbprint mouse
  - Palm print for secure entry
  - Fingerprint to unlock car door, etc.
- But biometrics not too popular
  - Has not lived up to its promise (yet?)

Ideal Biometric

- **Universal** — applies to (almost) everyone
  - In reality, no biometric applies to everyone
- **Distinguishing** — distinguish with certainty
  - In reality, cannot hope for 100% certainty
- **Permanent** — physical characteristic being measured never changes
  - In reality, OK if it to remains valid for long time
- **Collectable** — easy to collect required data
  - Depends on whether subjects are cooperative
- Also, safe, user-friendly, etc., etc.

Biometric Modes

- **Identification** — Who goes there?
  - Compare **one-to-many**
  - Example: The FBI fingerprint database
- **Authentication** — Are you who you say you are?
  - Compare **one-to-one**
  - Example: Thumbprint mouse
- Identification problem is more difficult
  - More “random” matches since more comparisons
- We are interested in authentication

Enrollment vs Recognition

- Enrollment phase
  - Subject's biometric info put into database
  - Must carefully measure the required info
  - OK if slow and repeated measurement needed
  - Must be very precise
  - May be weak point of many biometric
- Recognition phase
  - Biometric detection, when used in practice
  - Must be quick and simple
  - But must be reasonably accurate

Cooperative Subjects?

- Authentication — cooperative subjects
- Identification — uncooperative subjects
- For example, facial recognition
  - Used in Las Vegas casinos to detect known cheaters (terrorists in airports, etc.)
  - Often do not have ideal enrollment conditions
  - Subject will try to confuse recognition phase
- Cooperative subject makes it much easier
  - We are focused on authentication
  - So, subjects are generally cooperative

Biometric Errors

- **Fraud rate** versus **insult rate**
  - Fraud — Trudy mis-authenticated as Alice
  - Insult — Alice not authenticated as Alice
- For any biometric, can decrease fraud or insult, but other one will increase
- For example
  - 99% voiceprint match ⇒ low fraud, high insult
  - 30% voiceprint match ⇒ high fraud, low insult
- **Equal error rate:** rate where fraud == insult
  - A way to compare different biometrics

Fingerprint

- Examples of **loops**, **whorls**, and **arches**
- Minutia extracted from these features

Loop (double)

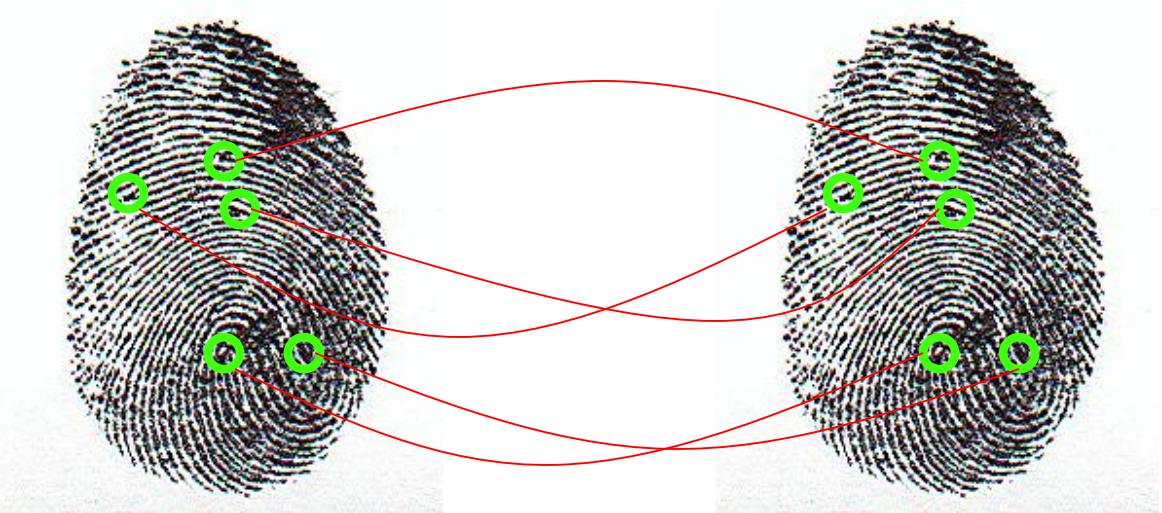
Whorl

Arch

Fingerprint: Enrollment

- Capture image of fingerprint
- Enhance image
- Identify points

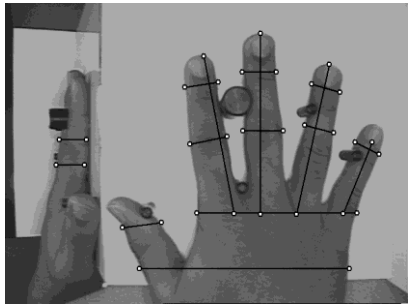
Fingerprint: Recognition



- Extracted points are compared with information stored in a database
- Is it a statistical match?
- Aside: Do identical twins' fingerprints differ?

Hand Geometry

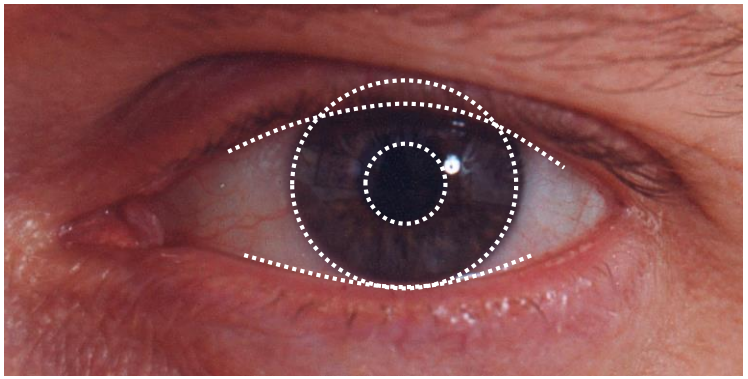
- A popular biometric
- Measures shape of hand
  - Width of hand, fingers
  - Length of fingers, etc.
- Human hands not unique
- Hand geometry sufficient for many situations
- OK for authentication
- Not useful for ID problem



Hand Geometry

- Advantages
  - Quick — 1 minute for enrollment, 5 seconds for recognition
  - Hands are symmetric — so what?
- Disadvantages
  - Cannot use on very young or very old
  - Relatively high equal error rate

Iris Patterns

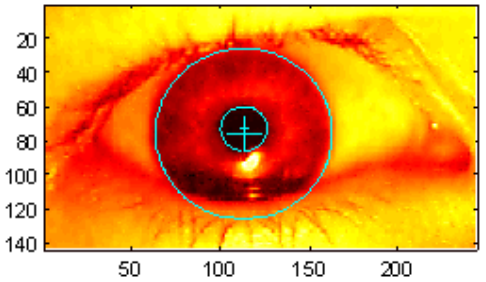


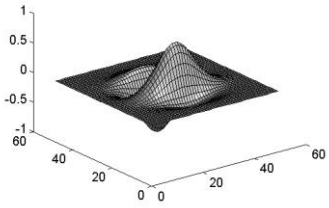
CANPASS Air Iris Recognition System

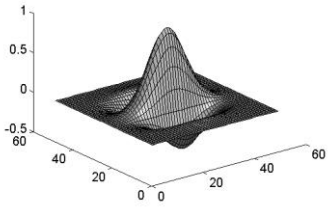
- Iris pattern development is “chaotic”
- Little or no genetic influence
- Different even for identical twins
- Pattern is stable through lifetime

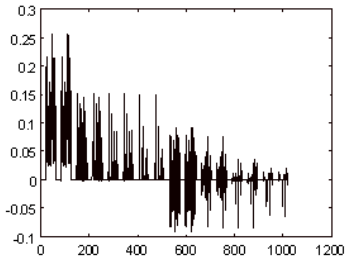
Iris Scan

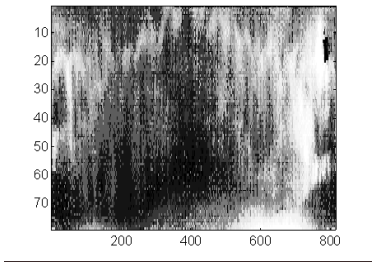
- Scanner locates iris
- Take b/w photo
- Use polar coordinates…
- 2-D wavelet transform
- Get 256 byte iris code











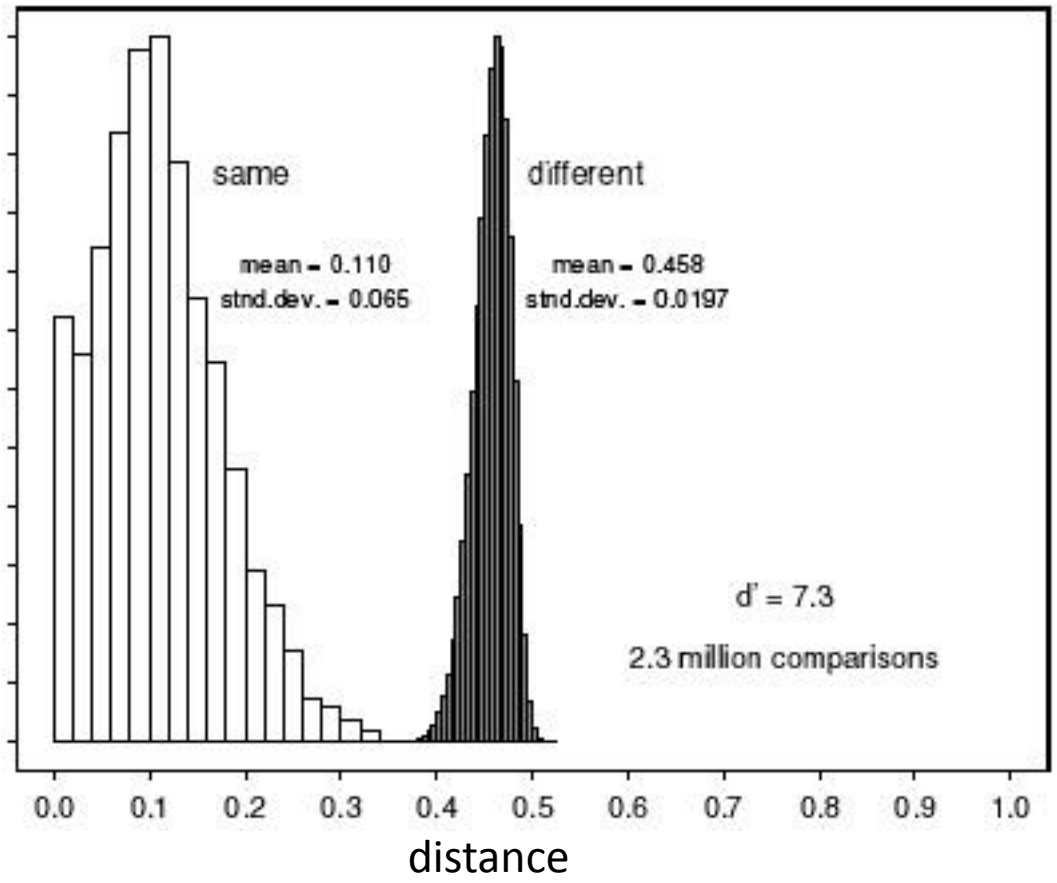
Measuring Iris Similarity

- Based on Hamming distance
- Define d(x,y) to be
  - # of non match bits / # of bits compared
  - d(0010,0101) = 3/4 and d(101111,101001) = 1/3
- Compute d(x,y) on 2048-bit iris code
  - Perfect match is d(x,y) = 0
  - For same iris, expected distance is 0.08
  - At random, expect distance of 0.50
  - Accept iris scan as match if distance < 0.32

Iris Scan Error Rate

distance	Fraud rate
0.29	1 in $1.3*10^{10}$
0.30	1 in $1.5*10^{9}$
0.31	1 in $1.8*10^{8}$
0.32	1 in $2.6*10^{7}$
0.33	1 in $4.0*10^{6}$
0.34	1 in $6.9*10^{5}$
0.35	1 in $1.3*10^{5}$

== equal error rate



Attack on Iris Scan

- Good **photo** of eye can be scanned
  - Attacker could use photo of eye

- To prevent attack, scanner could use light to be sure it is a "live" iris

Equal Error Rate Comparison

- Equal error rate (EER): fraud == insult rate
- **Fingerprint** biometric has EER of about 5%
- **Hand geometry** has EER of about 10^{-3}
- In theory, **iris scan** has EER of about 10^{-6}
  - But in practice, may be hard to achieve
  - Enrollment phase must be extremely accurate
- Most biometrics much worse than fingerprint!
- Biometrics useful for authentication…
  - …but identification biometrics almost useless today

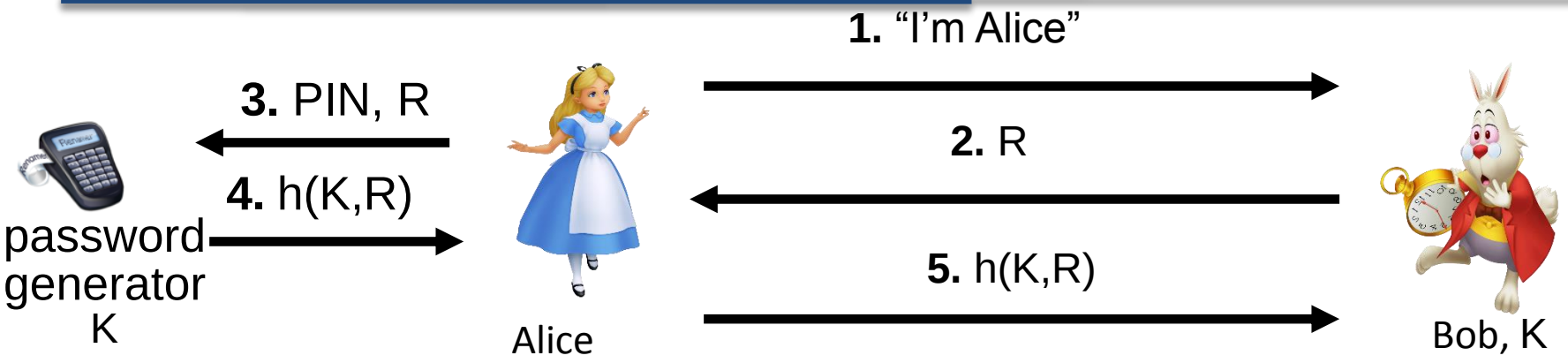
Biometrics: The Bottom Line

- Biometrics are hard to forge
- But attacker could
  - Steal Alice's thumb
  - Photocopy Bob's fingerprint, eye, etc.
  - Subvert software, database, "trusted path" …
- And how to revoke a "broken" biometric?
- **Biometrics are not foolproof**
- Biometric use is limited today
- That should change in the (near?) future

Something You Have

- Something in your possession
- Examples include following…
  - Car key
  - Laptop computer (or MAC address)
  - Password generator (next)
  - ATM card, smartcard, etc.

Password Generator



- Alice receives random "challenge" R from Bob
- Alice enters PIN and R in password generator
- Password generator hashes symmetric key K with R
- Alice sends "response" h(K,R) back to Bob
- Bob verifies response
- Note: Alice **has** pwd generator and **knows** PIN

2-factor Authentication

- Requires any 2 out of 3 of
  - Something you know
  - Something you have
  - Something you are
- Examples
  - ATM: Card and PIN
  - Credit card: Card and signature
  - Password generator: Device and PIN
  - Smartcard with password/PIN

Single Sign-on

- A hassle to enter password(s) repeatedly
  - Alice wants to authenticate only once
  - "Credentials" stay with Alice wherever she goes
  - Subsequent authentications transparent to Alice
- Kerberos --- example single sign-on protocol
- Single sign-on for the Internet?
  - Microsoft: **Passport**
  - Everybody else: **Liberty Alliance**
  - Security Assertion Markup Language (**SAML**)

Web Cookies

- Cookie is provided by a Website and stored on user's machine
- Cookie indexes a database at Website
- Cookies **maintain state** across sessions
  - Web uses a stateless protocol: HTTP
  - Cookies also maintain state within a session
- Sorta like a single sign-on for a website
  - But, a very, very weak form of authentication
- Cookies also create privacy concerns

Authentication

- Authentication — who goes there?
  - Passwords — something you know
  - Biometrics — something you are (you are your key)
  - Something you have