Authentication

Chapter 2

Basics of Access Control

- Access control is a collection of methods and components
  - Supports confidentiality (protects information from unauthorized disclosure)
  - Supports integrity (protects information from unauthorized modification)
- Goal: to allow only authorized subjects to access objects that they are permitted to access

Access Control

- Two parts to access control
- **Authentication:** Who goes there?
  - Determine whether access is allowed
  - Authenticate human to machine
  - 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

Access Control Basics (continued)

- Subject
  - The entity that requests access to a resource
- Object
  - The resource a subject attempts to access
- Least privilege philosophy
  - A subject is granted permissions needed to accomplish required tasks and nothing more

Controls

- Mechanisms put into place to allow or disallow object access
  - Any potential barrier to unauthorized access
- Controls organized into different categories
- Common categories
  - Administrative (enforce security rules through policies)

    Hiring practice, Usage monitoring and accounting
  - Logical/Technical (implement object access restrictions)

    User identification and authentication, Encryption
  - Physical (limit physical access to hardware)

    Fence, Walls, Locked doors

Access Control Techniques

- Choose techniques that fit the organization's needs
- Considerations include
  - Level of security required
  - User and environmental impact of security measures
- Techniques differ in
  - The way objects and subjects are identified
  - How decisions are made to approve or deny access

Access Control Designs

- Access control designs define rules for users accessing files or devices
- Three common access control designs
  - Mandatory access control
  - Discretionary access control
  - Non-discretionary access control

Mandatory Access Control

- Assigns a security label to each subject and object
- Matches label of subject to label of object to determine when access should be granted
- A common implementation is rule-based access control
  - Often requires a subject to have a need to know in addition to proper security clearance
  - Need to know indicates that a subject requires access to object to complete a particular task

Mandatory Access Control (continued)

- Common military data classifications
  - Unclassified, Sensitive but Unclassified, Confidential, Secret, Top Secret
- Common commercial data classifications
  - Public, Sensitive, Private, Confidential

Discretionary Access Control

- Uses identity of subject to decide when to grant an access request
- All access to an object is defined by the object owner
- Most common design in commercial operating systems
  - Generally less secure than mandatory control
  - Generally easier to implement and more flexible
- Includes
  - Identity-based access control
  - Access control lists (ACLs)

Non-discretionary Access Control

- Uses a subject's role or a task assigned to subject to grant or deny object access
  - Also called role-based or task-based access control
- Works well in environments with high turnover of subjects since access is not tied directly to subject
- Lattice-based control is a variation of non-discretionary control
  - Relationship between subject and object has a set of access boundaries that define rules and conditions for access

Access Control Administration

- Can be implemented as centralized, decentralized, or hybrid
- Centralized access control administration
  - All requests go through a central authority
  - Administration is relatively simple
  - Single point of failure, sometimes performance bottlenecks
  - Common packages include Remote Authentication Dial-In User Service (RADIUS), Challenge Handshake Authentication Protocol (CHAP), Terminal Access Controller Access Control System (TACACS)

Access Control Administration (continued)

- Decentralized access control administration
  - Object access is controlled locally rather than centrally
  - More difficult administration
    - Objects may need to be secured at multiple locations
  - More stable
    - Not a single point of failure
  - Usually implemented using security domains

Accountability

- System auditing used by administrators to monitor
  - Who is using the system
  - What users are doing
- Logs can trace events back to originating users
- Process of auditing can have a negative effect on system performance
  - Must limit data collected in logs
  - Clipping levels set thresholds for when to start collecting data

Access Control Models

- Provide conceptual view of security policies
- Map goals and directives to specific system events
- Provide a formal definition and specification of required security controls
- Many different models and combinations of models are used

State Machine Model

- A collection of defined states and transitions
- Modifications change objects from one state to the next
- A state represents the characteristics of an object at a point in time
- Transitions represent the modifications that can be made to objects to change from one state to another

State Machine Model (continued)

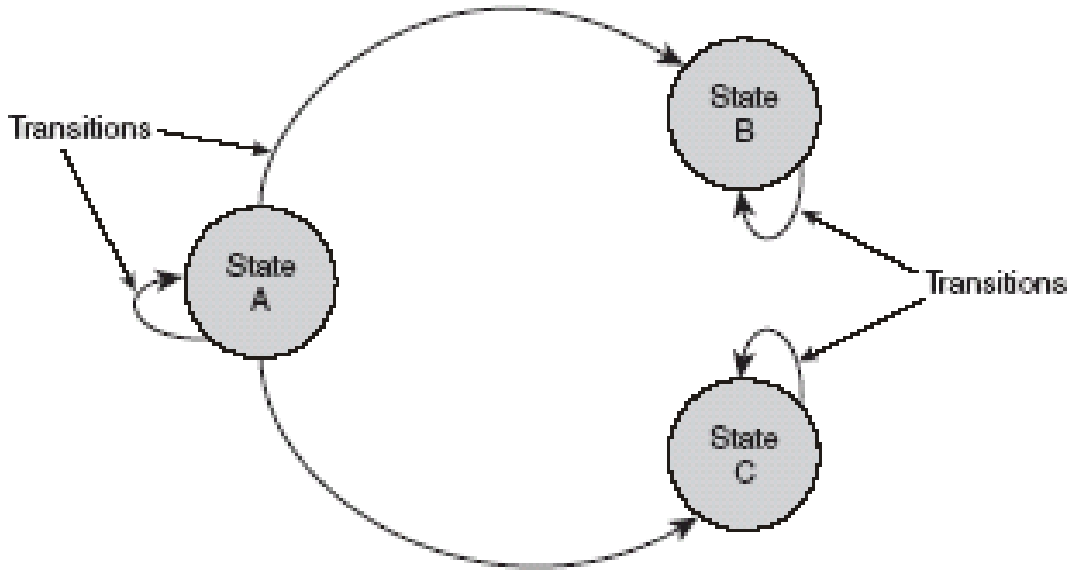
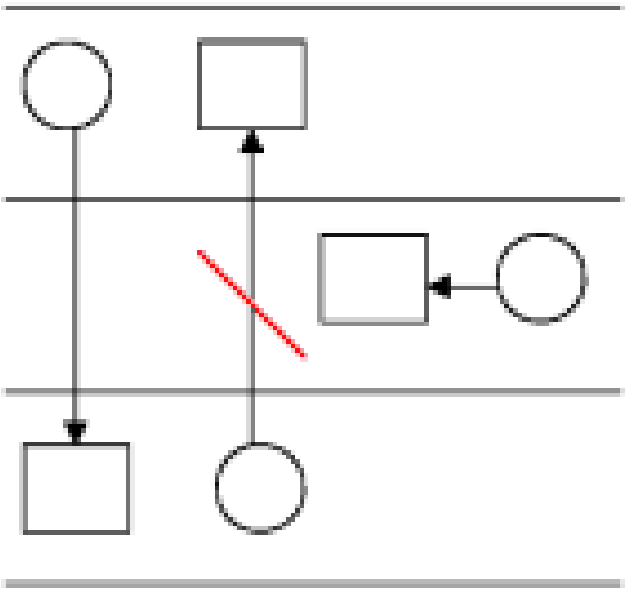


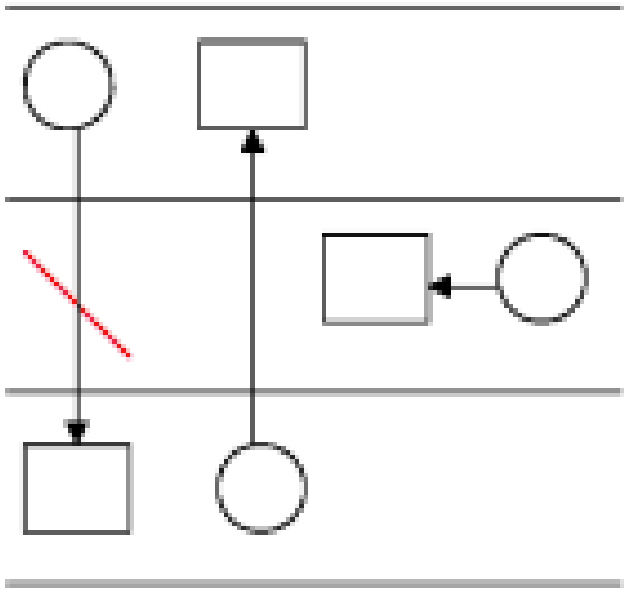
Figure 2.1
Simple state machine

State Machine Model (continued)

- Bell-LaPadula model
  - Works well in organizations that focus on confidentiality
  - No read up, no write down



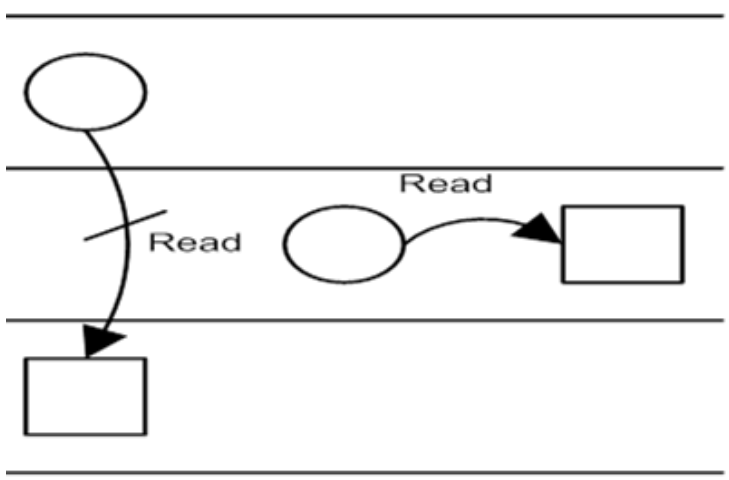
read

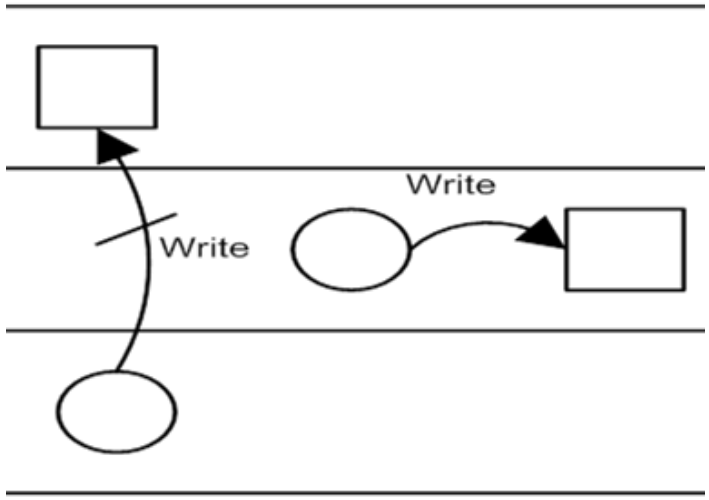


write

State Machine Model (continued)

- Biba model
  - Focuses on integrity controls
  - No read down, no write up





State Machine Model (continued)

- Clark-Wilson Model
  - Not a state machine model
  - Use a different approach to ensure data integrity
  - Restricts access to a small number of tightly controlled access programs
  - CDIs: constrained data items

    Data protected by the model
  - UDIs: unconstrained data items

    Data not protected by the model
  - IVPs: integrity verification procedures

    Procedures that verifies the integrity of a data item
  - TPs: transaction procedures

    Any procedure that makes authorized changes to a data item

State Machine Model (continued)

- Noninterference Model
  - Often an addition to other models
  - Ensures that changes at one security level do not bleed over into other levels

Who Goes There? Authentication

- How to authenticate a human to 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.)"

Why Passwords?

- Why is “something you know” more popular than “something you have” and “something you are”?
- **Cost**: passwords are free
- **Convenience**: easier for SA to reset password than to issue new smartcard

Keys vs Passwords

- **Crypto keys**
- Spse key is 64 bits
- Then 2^{64} keys
- Choose key at random
- Then attacker must try about 2^{63} keys

- **Passwords**
- Spse passwords are 8 characters, and 256 different characters
- Then $256^8 = 2^{64}$ pwds
- Users do not select passwords at random
- Attacker has far less than 2^{63} pwds 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
  - **Group C:** 8 random characters

winner →

- 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 pwd cracking tool to test for weak pwds
  - 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 need a way to verify passwords
- Cryptographic solution: **hash** the passwords
  - Store y = hash(password)
  - Can verify entered password by hashing
  - If attacker obtains password file, he does not obtain passwords
  - But attacker with password file can guess x and check whether y = hash(x)
  - If so, attacker has found password!

Dictionary Attack

- Attacker pre-computes hash(x) for all x in a **dictionary** of common passwords
- Suppose attacker gets access to password file containing hashed passwords
  - Attacker only needs to compare hashes to his pre-computed dictionary
  - Same attack will work each time
- Can we prevent this attack? Or at least make attacker's job more difficult?

Password File

- Store hashed passwords
- Better to hash with **salt**
- Given password, choose random s, compute

  y = hash(password, s)

  and store the pair (s,y) in the password file
- Note: The salt s is **not secret**
- Easy to verify password
- Attacker must recompute dictionary hashes for each user — lots more work!

Password Cracking: Do the Math

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

Password Cracking

- Attack 1 password without dictionary
  - Must try $2^{56}/2 = 2^{55}$ on average
  - Just like exhaustive key search
- Attack 1 password with dictionary
  - Expected work is about

$$1/4\ (2^{19}) + 3/4\ (2^{55}) = 2^{54.6}$$

  - But in practice, try all in dictionary and quit if not found — work is at most 2^{20} and probability of success is 1/4

Password Cracking

- Attack any of 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, each hash computation gives 2^{10} comparisons $\Rightarrow$ the expected work (number of hashes) is $2^{55}/2^{10} = 2^{45}$
  - If salt is used, expected work is 2^{55} since each comparison requires a new hash computation

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.

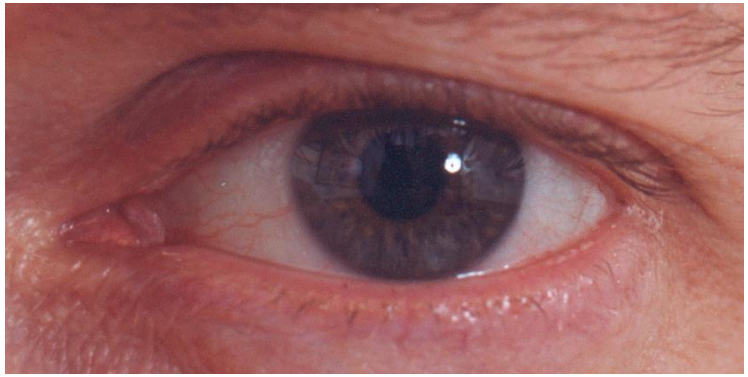
Passwords

- The bottom line
- **Password cracking is too easy!**
  - One weak password may break security
  - Users choose bad passwords
  - Social engineering attacks, etc.
- The bad guy has all of the advantages
- All of the math favors bad guys
- Passwords are a **big** security 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 article on password cracking
  - Passwords - Conerstone of Computer Security

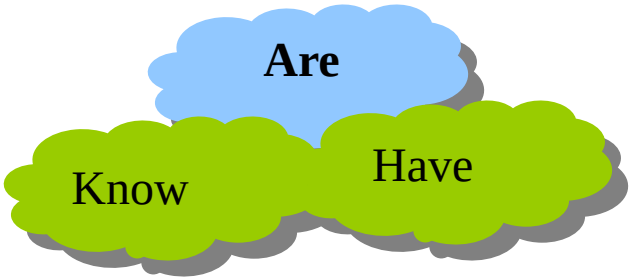
Biometrics



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?

- Biometrics seen as desirable replacement for passwords
- Cheap and reliable biometrics needed
- Today, a very 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, want it to remain valid for a long time
- **Collectable** — easy to collect required data
  - Depends on whether subjects are cooperative
- Safe, easy to use, etc., etc.

Biometric Modes

- **Identification** — Who goes there?
  - Compare one to many
  - Example: The FBI fingerprint database
- **Authentication** — Is that really you?
  - Compare one to one
  - Example: Thumbprint mouse
- Identification problem 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 for good recognition
  - A weak point of many biometric schemes
- Recognition phase
  - Biometric detection when used in practice
  - Must be quick and simple
  - But must be reasonably accurate

Cooperative Subjects

- We are assuming cooperative subjects
- In identification problem often have uncooperative subjects
- For example, facial recognition
  - Proposed for use in Las Vegas casinos to detect known cheaters
  - Also as way to detect terrorists in airports, etc.
  - Probably do not have ideal enrollment conditions
  - Subject will try to confuse recognition phase
- Cooperative subject makes it much easier!
  - In authentication, subjects are cooperative

Biometric Errors

- **Fraud rate** versus **insult rate**
  - Fraud — user A mis-authenticated as user B
  - Insult — user A not authenticate as user A
- For any biometric, can decrease fraud or insult, but other 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
  - The best measure for comparing biometrics

Fingerprint Comparison

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

Loop (double)

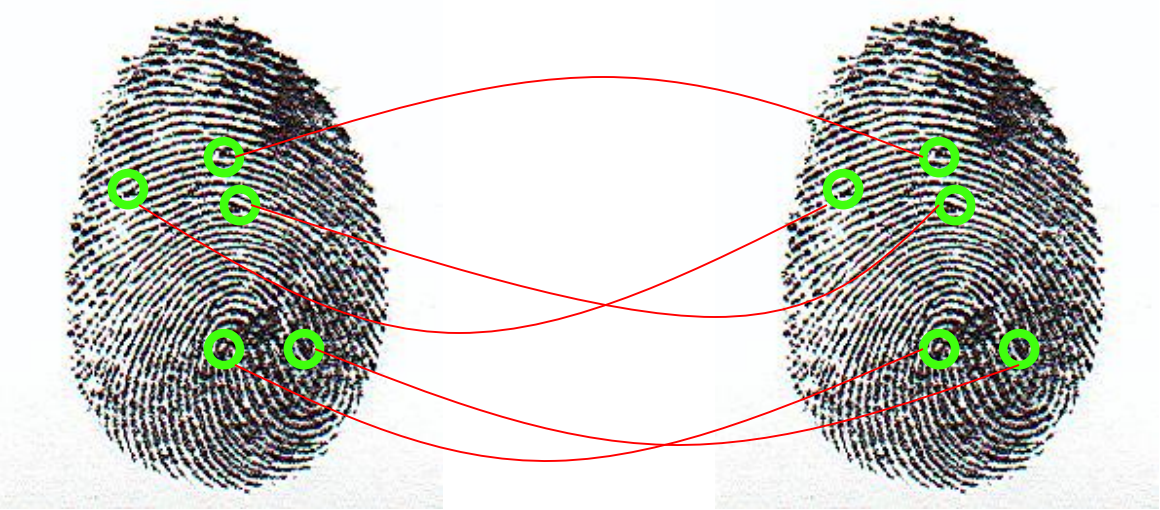
Whorl

Arch

Fingerprint Biometric

- Capture image of fingerprint
- Enhance image
- Identify minutia

Fingerprint Biometric



- Extracted minutia are compared with user's minutia stored in a database
- Is it a statistical match?

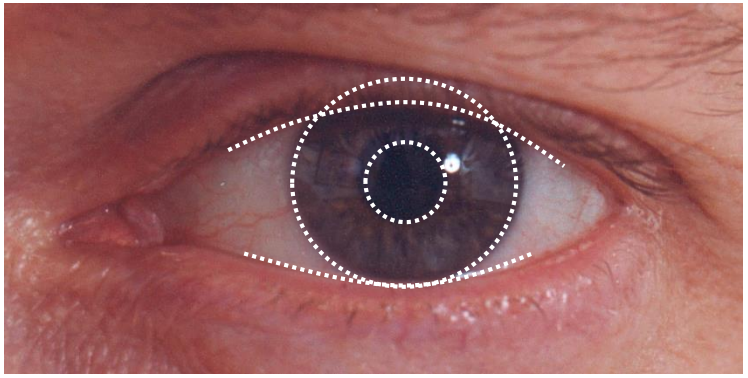
Hand Geometry

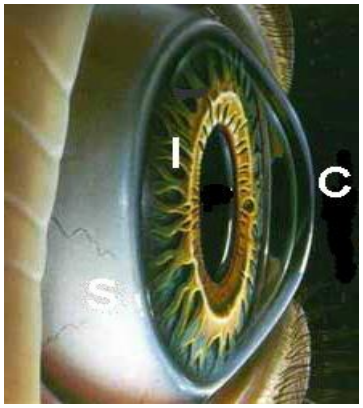
- Popular form of biometric
- Measures shape of hand
  - Width of hand, fingers
  - Length of fingers, etc.
- Human hands not unique
- Hand geometry sufficient for many situations
- Suitable for authentication
- Not useful for ID problem

Hand Geometry

- Advantages
  - Quick
  - 1 minute for enrollment
  - 5 seconds for recognition
  - Hands symmetric (use other hand backwards)
- Disadvantages
  - Cannot use on very young or very old
  - Relatively high equal error rate

Iris Patterns

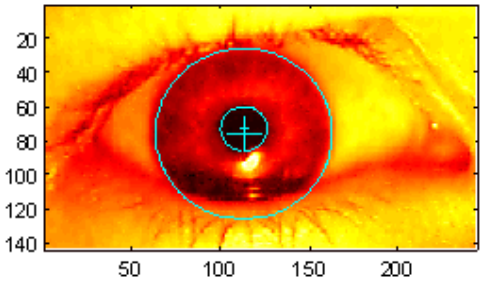


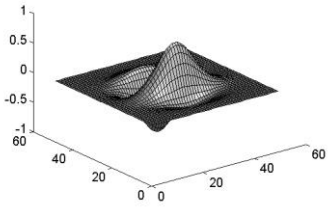


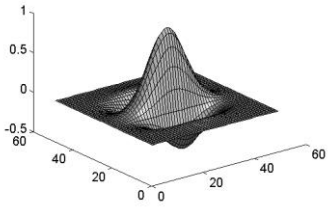
- Iris pattern development is “chaotic”
- Little or no genetic influence
- Different even for identical twins
- Pattern is stable through lifetime

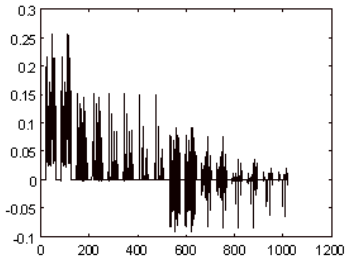
Iris Scan

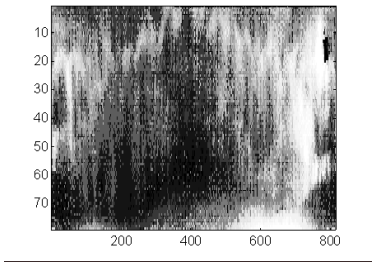
- Scanner locates iris
- Take b/w photo
- Use polar coordinates…
- Find 2-D wavelet trans
- Get 256 byte iris code











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 as match if distance less than 0.32

Attack on Iris Scan

- Good **photo** of eye can be scanned
- And attacker can use photo of eye

- Afghan woman was authenticated by iris scan of old photo
- To prevent photo 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, hard to achieve
  - Enrollment phase must be extremely accurate
- Most biometrics much worse than fingerprint!
- Biometrics useful for authentication…
- But ID biometrics are almost useless today

Something You Have

- Something in your possession
- Examples include
  - Car key
  - Laptop computer
    - Or specific MAC address
  - Password generator
    - We'll look at this next
  - ATM card, smartcard, etc.

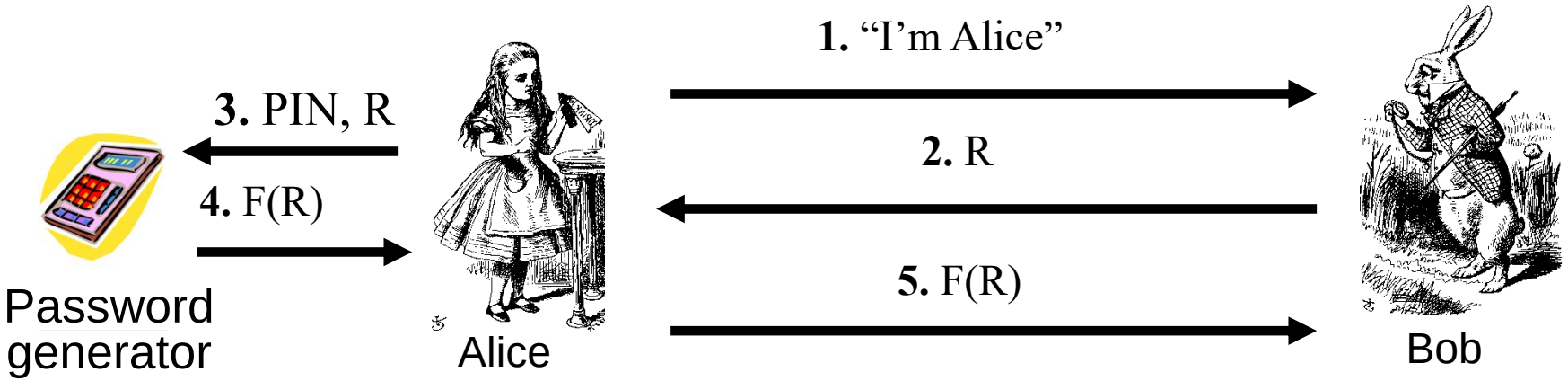
Identification and Authentication Methods

- Security practices often require input from multiple categories of authentication techniques
- Most complex authentication mechanism is biometrics (detection and classification of a subject's physical attributes)

Identification and Authentication Methods

- Two-factor authentication uses two phases
  - Identification
  - Authentication
  - Requires 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

Password Generator



- Alice gets "challenge" R from Bob
- Alice enters R into password generator
- Alice sends "response" back to Bob
- Alice **has** pwd generator and **knows** PIN

Identification and Authentication Methods (continued)

TABLE 2.6 Authentication Types

Authentication Type	Description	Examples
Type 1	What you know	Password, passphrase, PIN, lock combination
Type 2	What you have	Smart card, token device
Type 3	What you are	Biometrics—fingerprint, palm print, retina/iris pattern, voice pattern

Single Sign-On

- Used to avoid multiple logins
- Once a subject is positively identified, authentication information can be used within a trusted group
- Great for users since they can sign on once and use multiple resources
- Requires additional work for administrators
- Several good SSO systems in use, Kerberos is one example

Kerberos

- Uses symmetric key cryptography for messages
- Provides end-to-end security
  - Intermediate machines between the source and target cannot read contents of messages
- Used in distributed environments but implemented with a central server
- Includes a data repository and an authentication process
- Weaknesses include
  - Single point of failure, performance bottleneck
  - Session key lives on client machines for a small amount of time, can be stolen

File and Data Ownership

- Different layers of responsibility for ensuring security of organization's information
- Data owner
  - Bears ultimate responsibility, sets classification levels
- Data custodian
  - Enforces security policies, often a member of IT department
- Data user
  - Accesses data on a day-to-day basis, responsible for following the organization's security policies

Related Methods of Attacks

- Brute force attack
  - Try all possible combinations of characters to satisfy Type 1 authentication (password guessing)
- Dictionary attack
  - Subset of brute force
  - Instead of all possible combinations, uses a list of common passwords
- Spoofing attack
  - Create fake login program, prompt for User ID, password
  - Return login failure message, store captured information

Summary

- Use access control to ensure that only authorized users can view/modify information
- Access control designs define rules for accessing objects
  - Mandatory, discretionary, non-discretionary
- Access control administration defines the mechanisms for access control implementation
  - Centralized, decentralized, hybrid
- Administrators use system logs to monitor access

Summary (continued)

- Access control models
  - Provide a conceptual view of security policies
  - One common example is the state machine model
- Identification and authentication methods
  - Used to identify and validate a user
  - Include passwords, smart cards, and biometrics
  - Single sign-on systems allow trusted groups to share authorizations (e.g., Kerberos)
- Responsibility for information access is shared
  - Data owners, custodians, users
- Attack types related to access controls include
  - Brute force attacks, dictionary attacks, login spoofing