CS 547: Foundation of Computer Security

S. Tripathy
IIT Patna

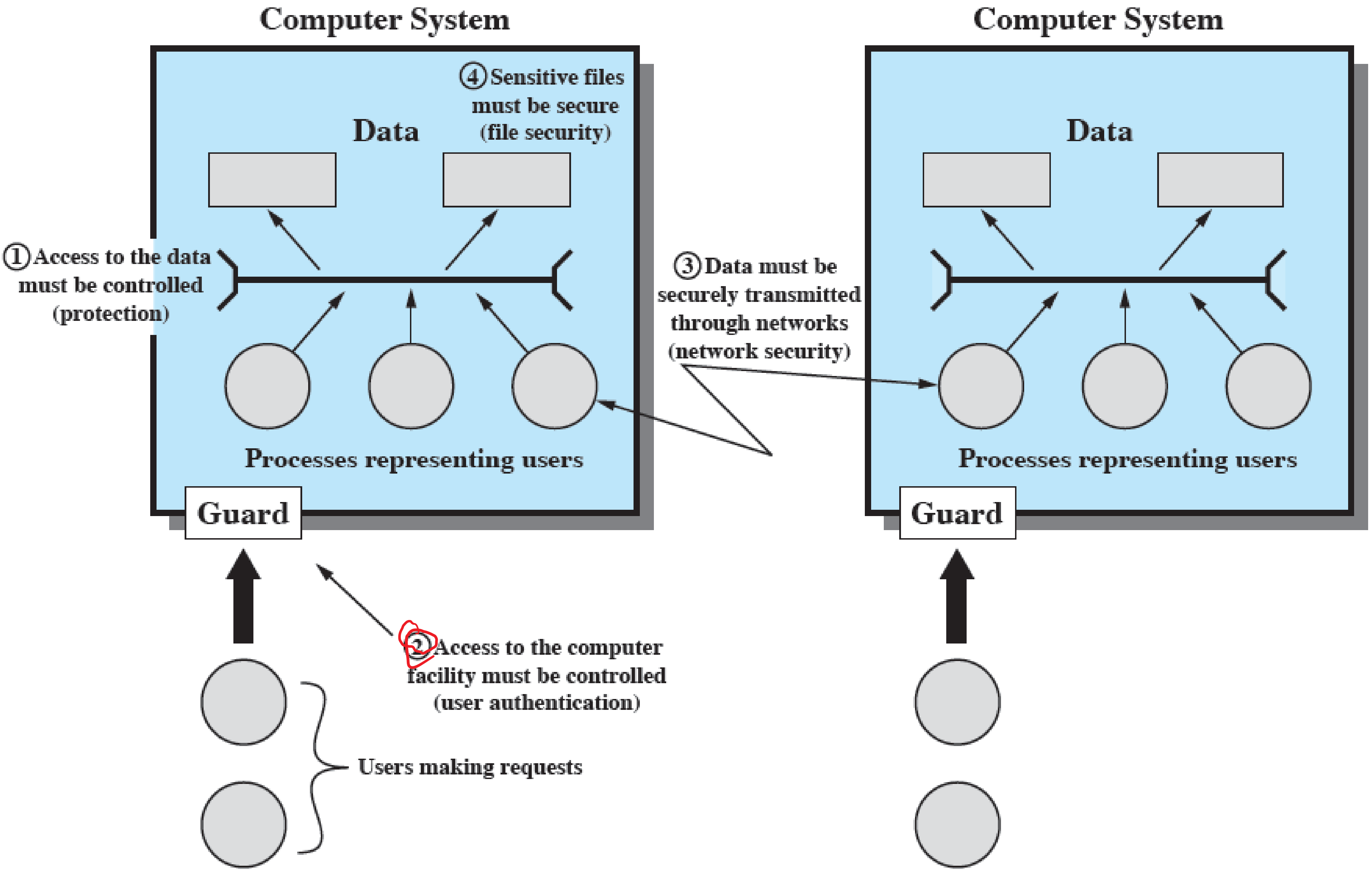
Previous *Class*

- Protection in General-Purpose Operating Systems
  - Segmentation and Paging
  - Dual Mode Protection
  - Potential threat on Virtual memory
    - Windows (page file)
    - Linux (Swap space)

Present Class

- User Authentication
- Access Control
  - Linux File System

Scope of Computer Security


Computer System
Computer System
④ Sensitive files must be secure (file security)
Data
Data
① Access to the data must be controlled (protection)
③ Data must be securely transmitted through networks (network security)
Processes representing users
Processes representing users
Guard
Guard
② Access to the computer facility must be controlled (user authentication)
Users making requests

Access Control

- Memory is only one of many objects for which OS has to run access control
- In general, access control :
  - Check every access: Else OS might fail to notice that access has been revoked
  - Enforce least privilege: Grant program access only to smallest number of objects required to perform a task
    - Access to additional objects might be harmless under normal circumstances, but disastrous in special cases
  - In most computer security contexts, user authentication is the fundamental building block and the primary line of defense.

Components of Access control

- Identification, Authentication & Authorization
- User authentication is the basis for most types of access control and for user accountability.
- **identification step**
  - presenting an identifier to the security system
- **verification step**
  - presenting or generating authentication information that corroborates the binding between the entity and the identifier

User Authentication

- Computer systems often have to identify and authenticate users
  - OS: when a user logs in
  - Web server: before handing out confidential information, like your grades/ marks
- Identification and authentication is easy among people that know each other
  - You identify your friends based on their face or voice
- It is difficult for computer to authenticate people sitting in its front
- Even more difficult for computers to authenticate people accessing them remotely

User Authentication

the four means of authenticating user identity are based on:

something the individual knows

- password, PIN, answers to prearranged questions

something the individual possesses (token)

- smartcard, electronic keycard, physical key

something the individual is (static biometrics)

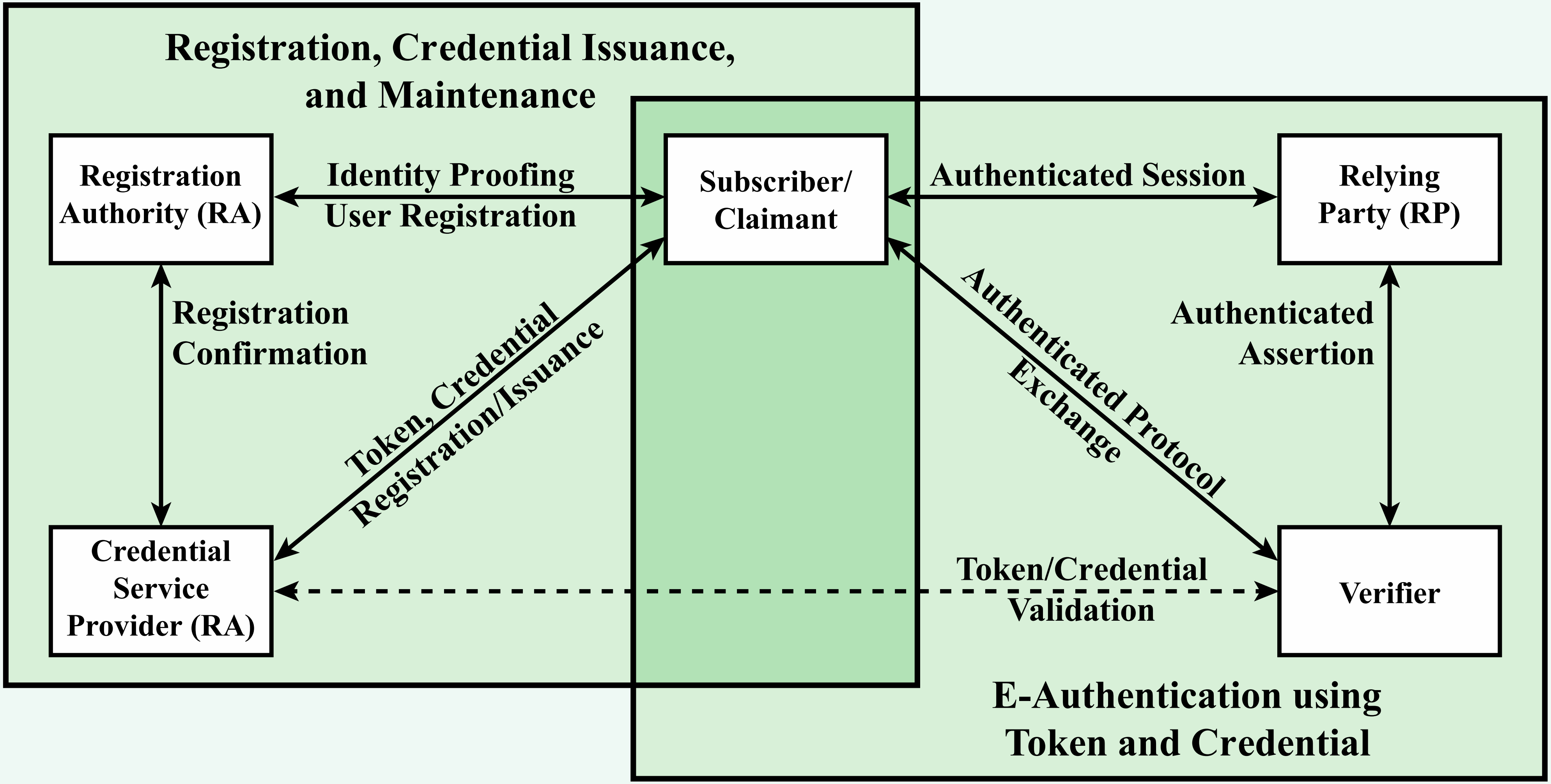
- fingerprint, retina, face

something the individual does (dynamic biometrics)

- voice pattern, handwriting, typing rhythm

Combination of Auth. Factors

- Different classes of authentication factors can be combined for more solid authentication
  - Two- or multi-factor authentication
- Using multiple factors from the same class might not provide better authentication
- "Something you have" can become "something you know"
  - If token can be easily duplicated, such as magnetic strip on ATM card. That's why ATM fraud is so wide spread
  - Some banks distribute small devices displaying numbers that change over time. Current number needs to be input for online banking. However, knowing number does not imply physical possession of device


Registration, Credential Issuance, and Maintenance

Registration Authority (RA)

Identity Proofing
User Registration

Subscriber/ Claimant

Authenticated Session

Relying Party (RP)

Registration Confirmation

Token, Credential Registration/Issuance

Authenticated Protocol Exchange

Authenticated Assertion

Credential Service Provider (RA)

Token/Credential Validation

Verifier

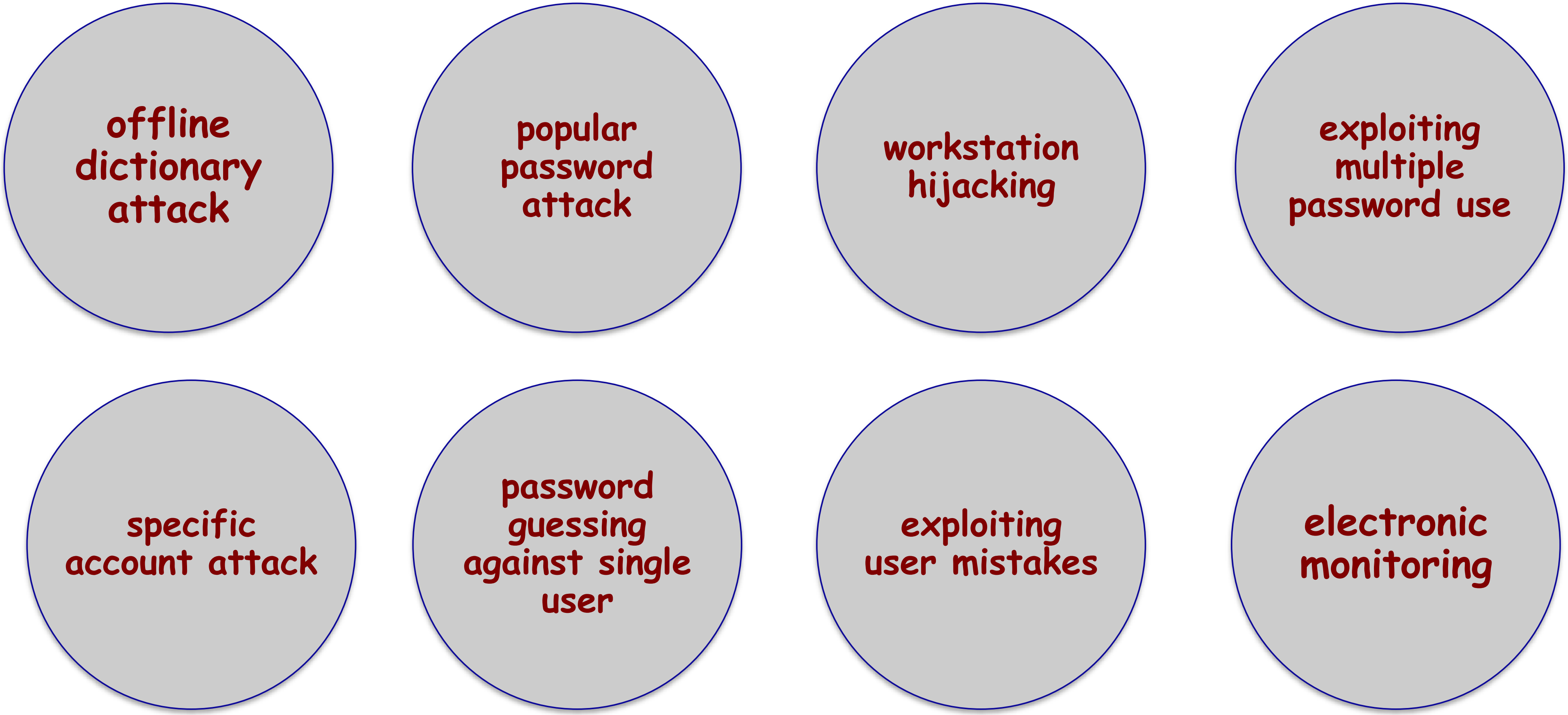
E-Authentication using Token and Credential

Figure 3.1 The NIST SP 800-63-2 E-Authentication Architectural Model

Password Authentication

- widely used line of defense against intruders
  - user provides name/login and password
  - system compares password with the one stored for that specified login
- the user ID:
  - determines that the user is authorized to access the system
  - determines the user's privileges
  - is used in access control

Password Vulnerabilities



offline dictionary attack

popular password attack

workstation hijacking

exploiting multiple password use

specific account attack

password guessing against single user

exploiting user mistakes

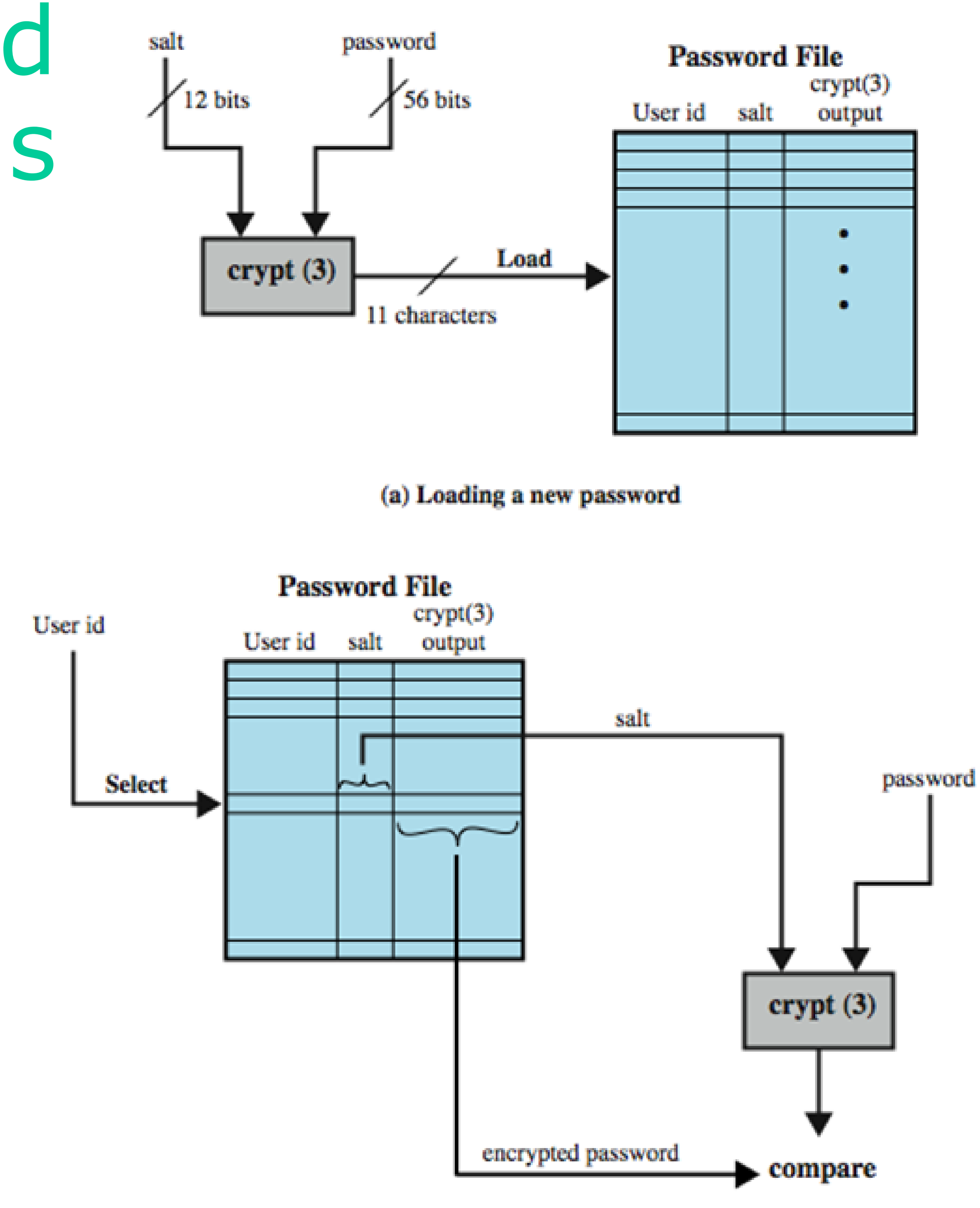
electronic monitoring

Countermeasures

- controls to prevent unauthorized access to password file
- account lockout mechanisms
- policies to inhibit users from selecting common passwords
- training in and enforcement of password policies
- automatic workstation logout
- policies against similar passwords on network devices
- intrusion detection measures

Use of Hashed salt Passwords

- Prevents duplicate passwords
- Increases the difficulty of offline dictionary attacks.
- becomes nearly impossible to find out whether a person with passwords on two or more systems has used the same password on all of them.



(a) Loading a new password

(b) Verifying a password

UNIX Implementation

- original scheme
  - up to eight printable characters in length
  - 12-bit salt used to modify DES encryption into a one-way hash function
  - zero value repeatedly encrypted 25 times
  - Output (64 bits) translated to 11 character sequence

- now regarded as inadequate
  - Dictionary attack using a supercomputer.
  - The attack was able to process over 50 million password guesses in about 80 minutes

Improved Implementations

- much stronger hash/salt schemes available for Unix
- recommended hash function is based on MD5
  - salt of up to 48-bits
  - password length is unlimited
  - produces 128-bit hash
  - uses an inner loop with 1000 iterations to achieve slowdown
- OpenBSD uses Blowfish block cipher based hash algorithm called Bcrypt
  - most secure version of Unix hash/salt scheme
  - uses 128-bit salt to create 192-bit hash value

Password Cracking

- dictionary attacks
  - develop a large dictionary of possible passwords and try each against the password file
  - each password must be hashed using each salt value and then compared to stored hash values

- rainbow table attacks
  - pre-compute tables of hash values for all salts
  - can be countered by using a sufficiently large salt value and a sufficiently large hash length

Password Cracking

- Password crackers exploit the fact that people choose easily guessable passwords
  - Shorter password lengths are also easier to crack
- John the Ripper
  - Open-source password cracker first developed in 1996
  - Uses a combination of brute-force and dictionary techniques

Modern Approaches

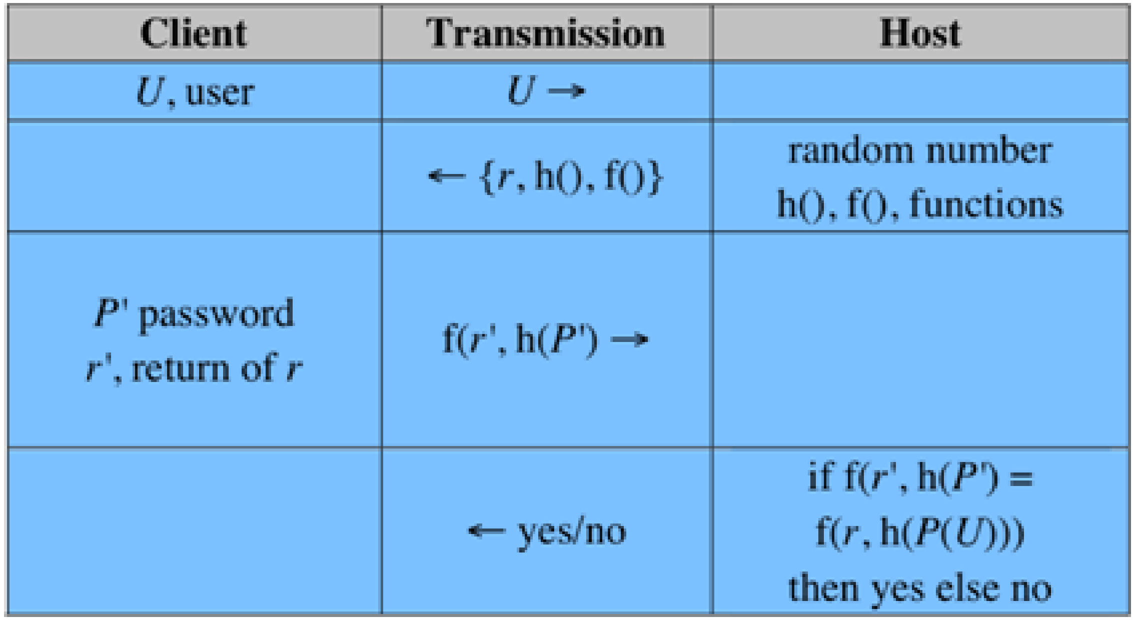
- Complex password policy
  - Forcing users to pick stronger passwords
- However, password-cracking techniques have also improved
  - The processing capacity available for password cracking has increased dramatically
  - The use of sophisticated algorithms to generate potential passwords
  - Studying examples and structures of actual passwords in use

Remote User Authentication

- authentication over a network, the Internet, or a communications link is more complex
  - additional security threats such as:
    - eavesdropping, capturing a password, replaying an authentication sequence that has been observed
- generally rely on some form of a *challenge-response protocol* to counter threats

Password Protocol

- user transmits identity to remote host
- host generates a random number (nonce)
- nonce is returned to the user
- host stores a hash code of the password function in which the password hash is one of the arguments
- use of a random number helps defend against an adversary capturing the user's transmission



Client	Transmission	Host
U, user	$U \rightarrow$	
	$\leftarrow \{r, \mathrm{h}(), \mathrm{f}()\}$	random number h(), f(), functions
P' password r', return of r	$\mathrm{f}(r', \mathrm{h}(P') \rightarrow$	
	$\leftarrow$ yes/no	if $\mathrm{f}(r', \mathrm{h}(P') =$ $\mathrm{f}(r, \mathrm{h}(P(U)))$ then yes else no

(a) Protocol for a password

- Thanks