Information Security (CS3002)

Instructor: Dr. Muhammad Usama

Email: usama.khanzada@nu.edu.pk

Book: Computer Security - Principles and Practice (Chapter 3)

User Authentication

RFC 4949

RFC 4949 defines user authentication as:

"The process of verifying an identity claimed by or for a system entity."

Authentication Process

- Fundamental building block and primary line of defense
- Basis for access control and 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

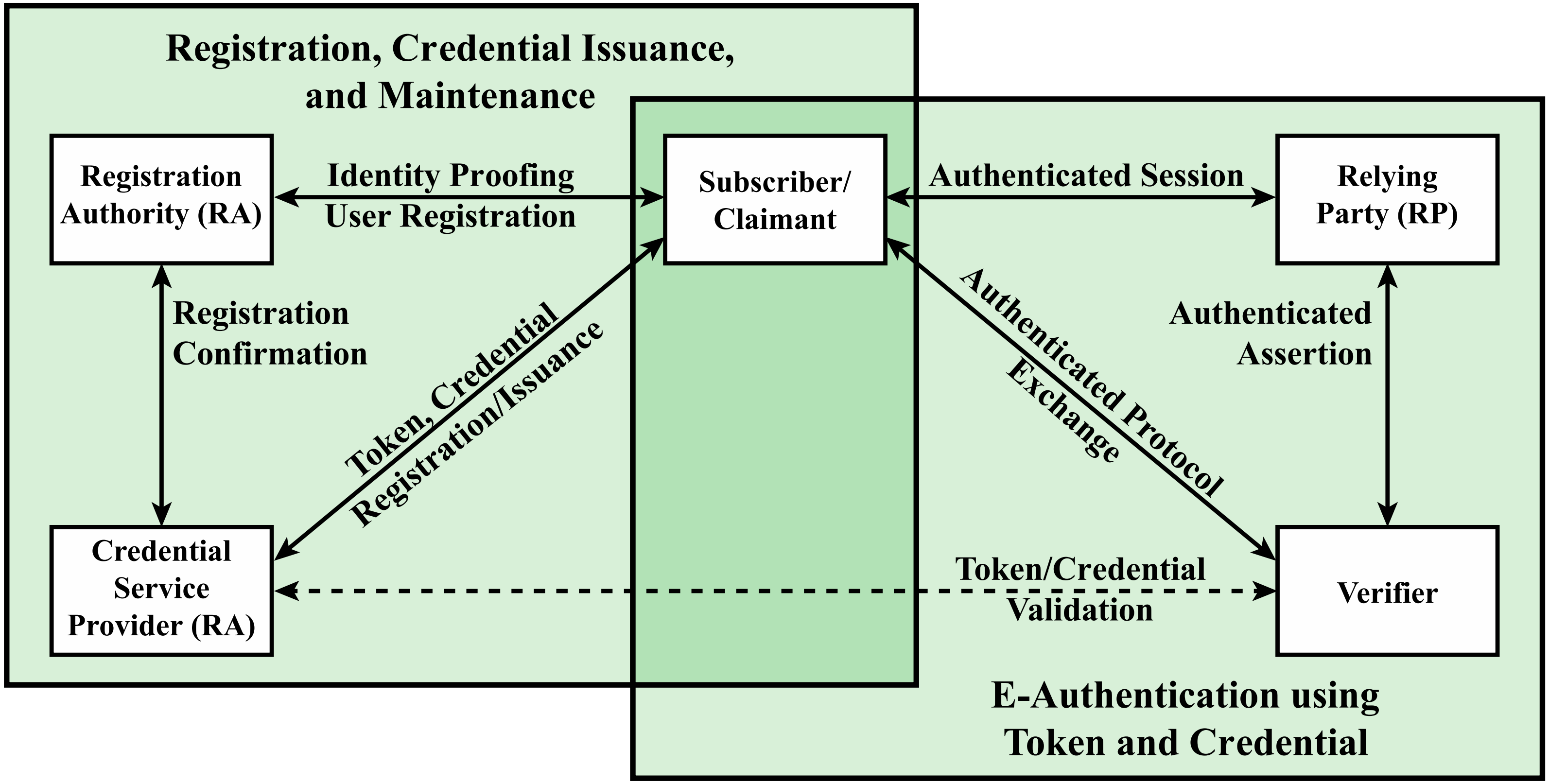


Figure 3.1 The NIST SP 800-63-2 E-Authentication Architectural Model

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)

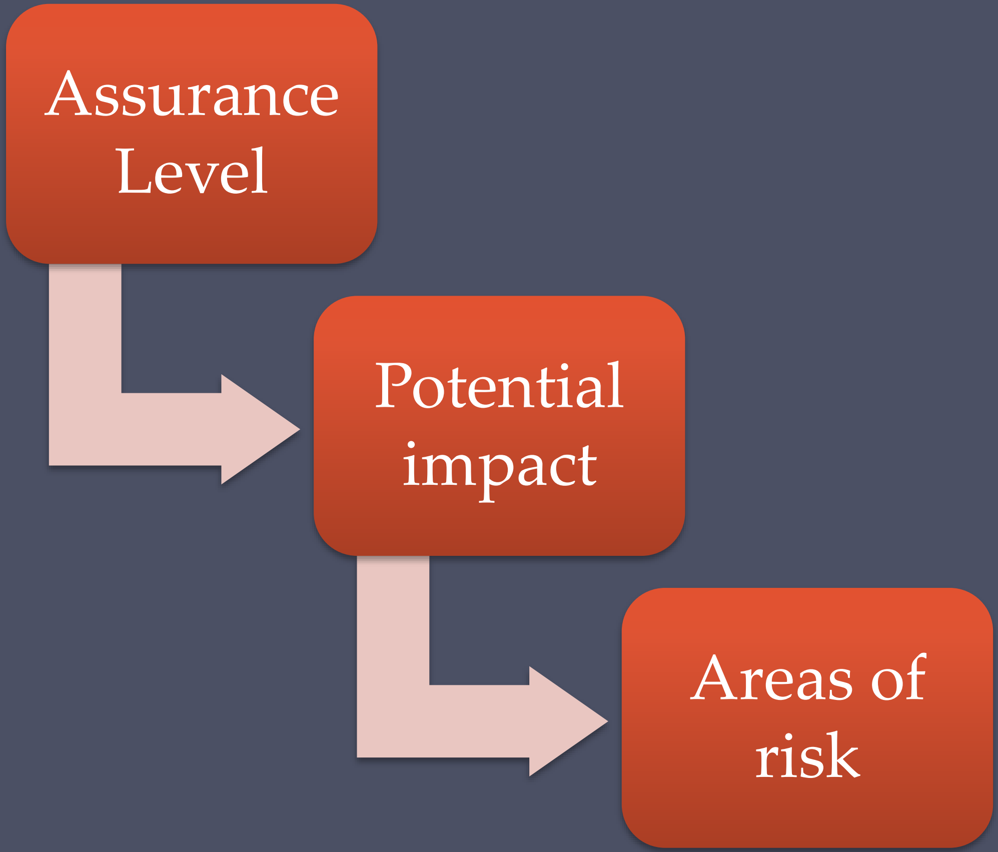
- Fingerprint, retina, face

Something the individual does (dynamic biometrics)

- Voice pattern, handwriting, typing rhythm

Risk Assessment for User Authentication

- There are three separate concepts:



Assurance Level

Describes an organization's degree of certainty that a user has presented a credential that refers to his or her identity

More specifically is defined as:

The degree of confidence in the vetting process used to establish the identity of the individual to whom the credential was issued

The degree of confidence that the individual who uses the credential is the individual to whom the credential was issued

Four levels of assurance

Level 1

- Little or no confidence in the asserted identity's validity

Level 2

- Some confidence in the asserted identity's validity

Level 3

- High confidence in the asserted identity's validity

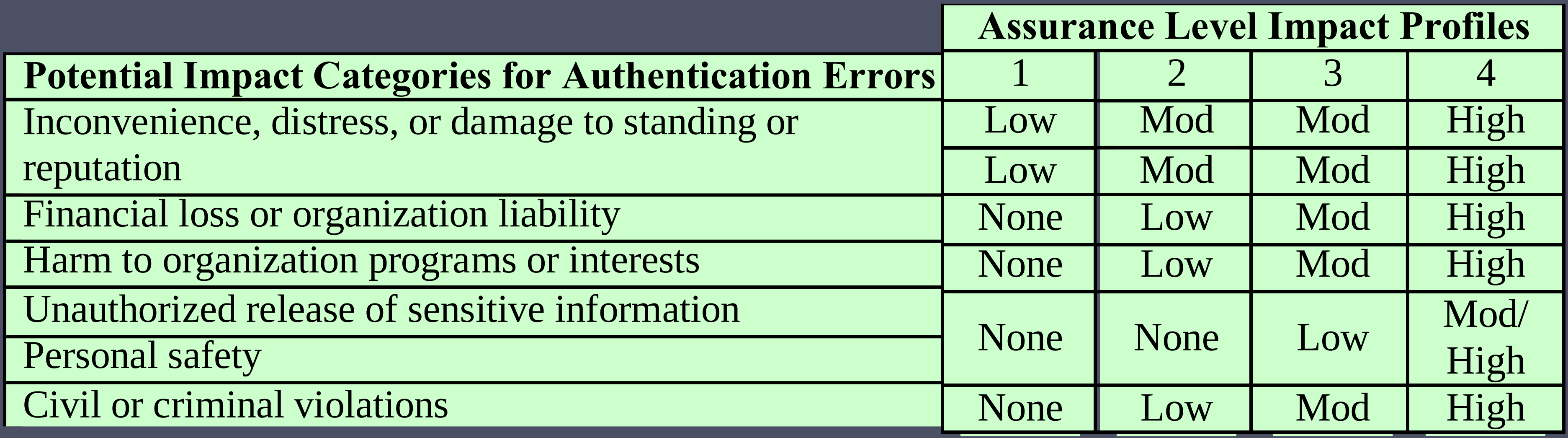
Level 4

- Very high confidence in the asserted identity's validity

Potential Impact

- FIPS 199 defines three levels of potential impact on organizations or individuals should there be a breach of security:
  - Low
    - An authentication error could be expected to have a limited adverse effect on organizational operations, organizational assets, or individuals
  - Moderate
    - An authentication error could be expected to have a serious adverse effect
  - High
    - An authentication error could be expected to have a severe or catastrophic adverse effect

Table 3.1



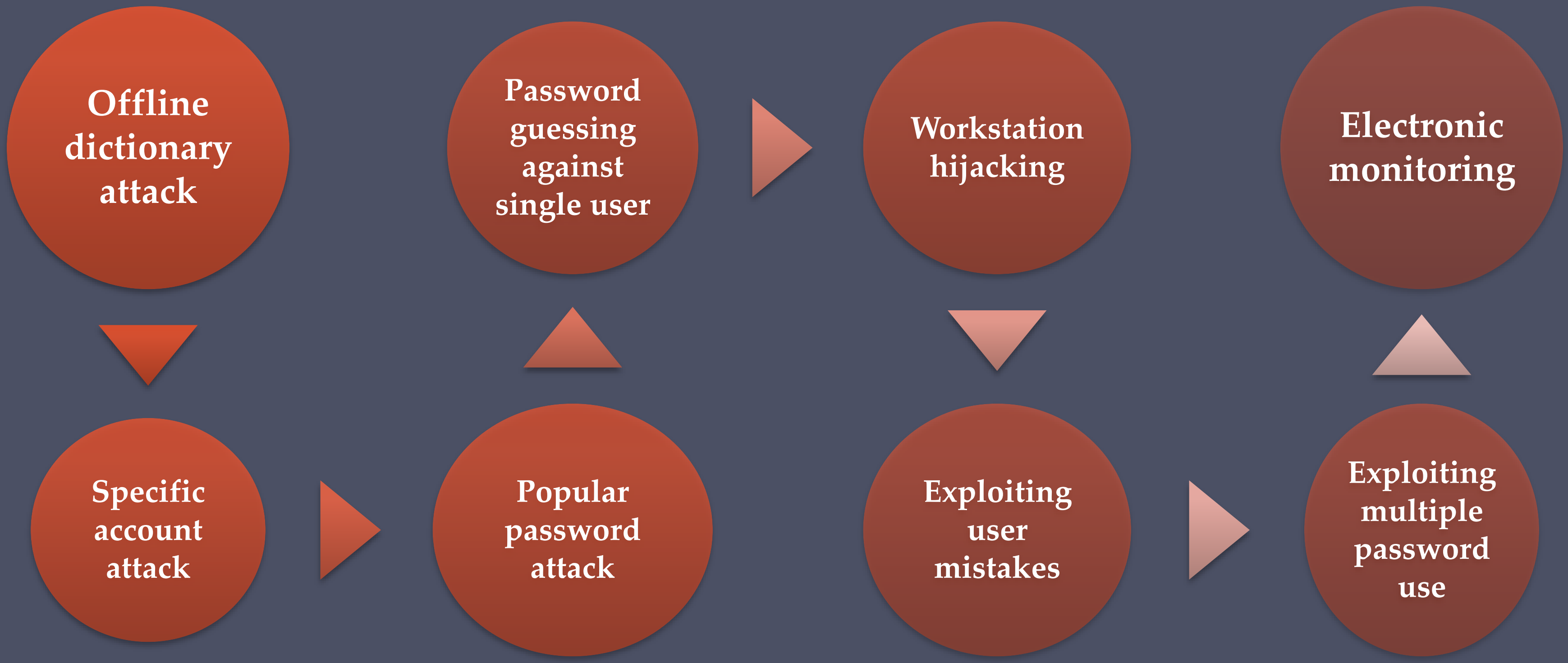
Potential Impact Categories for Authentication Errors	Assurance Level Impact Profiles			
	1	2	3	4
Inconvenience, distress, or damage to standing or reputation	Low	Mod	Mod	High
	Low	Mod	Mod	High
Financial loss or organization liability	None	Low	Mod	High
Harm to organization programs or interests	None	Low	Mod	High
Unauthorized release of sensitive information	None	None	Low	Mod/ High
Personal safety				
Civil or criminal violations	None	Low	Mod	High

Maximum Potential Impacts for Each Assurance Level

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 discretionary access control

Password Vulnerabilities



Offline dictionary attack

Specific account attack

Popular password attack

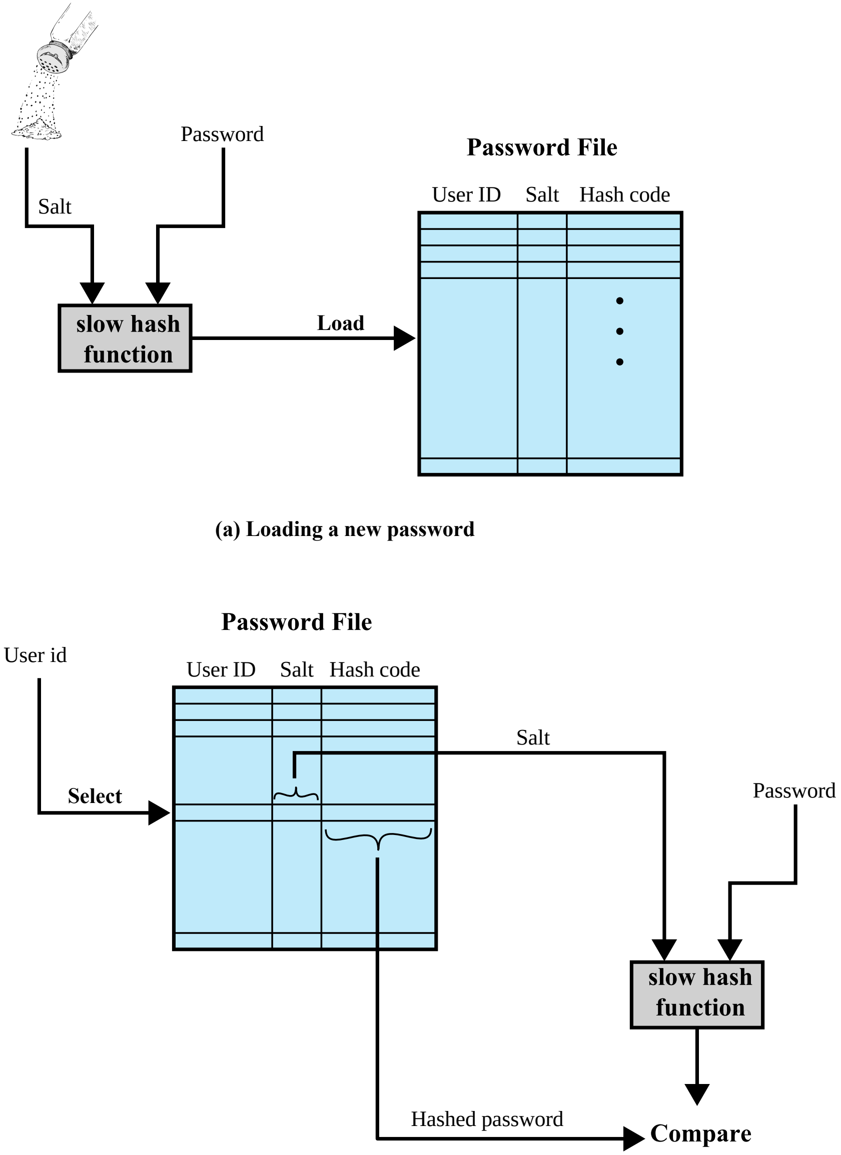
Password guessing against single user

Workstation hijacking

Exploiting user mistakes

Exploiting multiple password use

Electronic monitoring



(a) Loading a new password

(b) Verifying a password

Figure 3.2 UNIX Password Scheme

UNIX Implementation

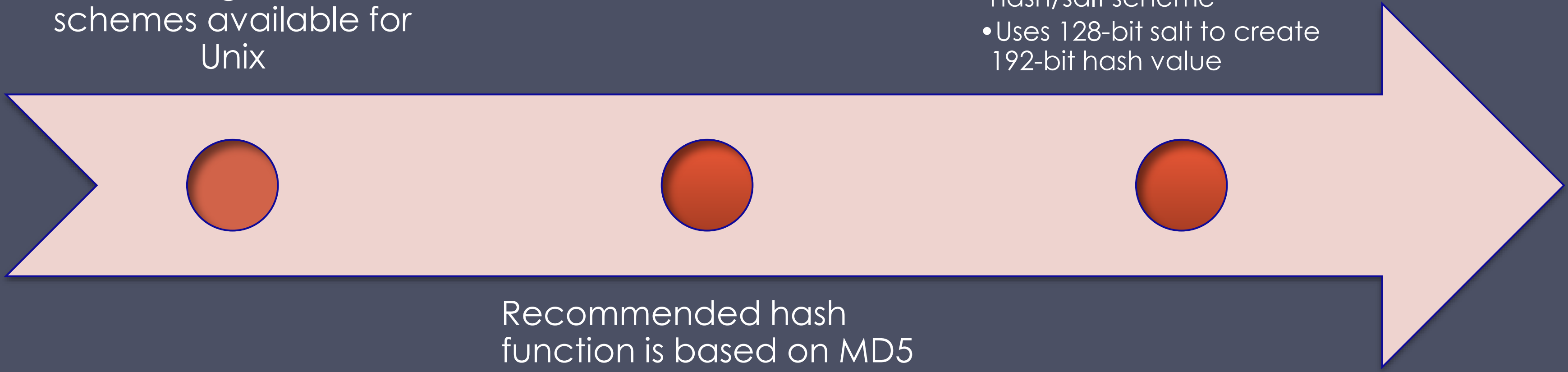
Original scheme

- 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 translated to 11-character sequence

Now regarded as inadequate

- Still often required for compatibility with existing account management software or multivendor environments

Improved Implementations



Much stronger hash/salt schemes available for Unix

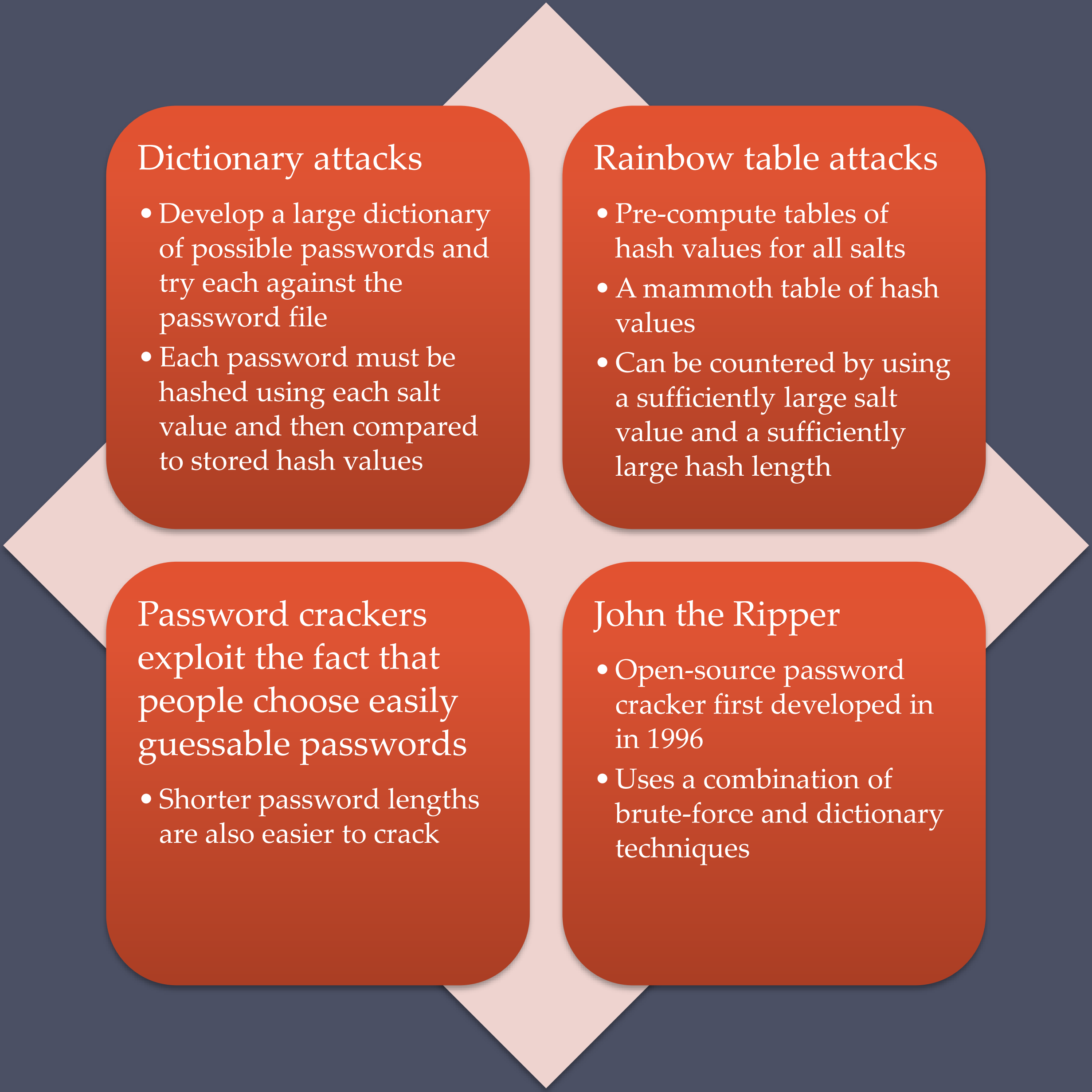
Recommended hash function is based on MD5

- 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
- A mammoth table of hash values
- Can be countered by using a sufficiently large salt value and a sufficiently large hash length

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 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

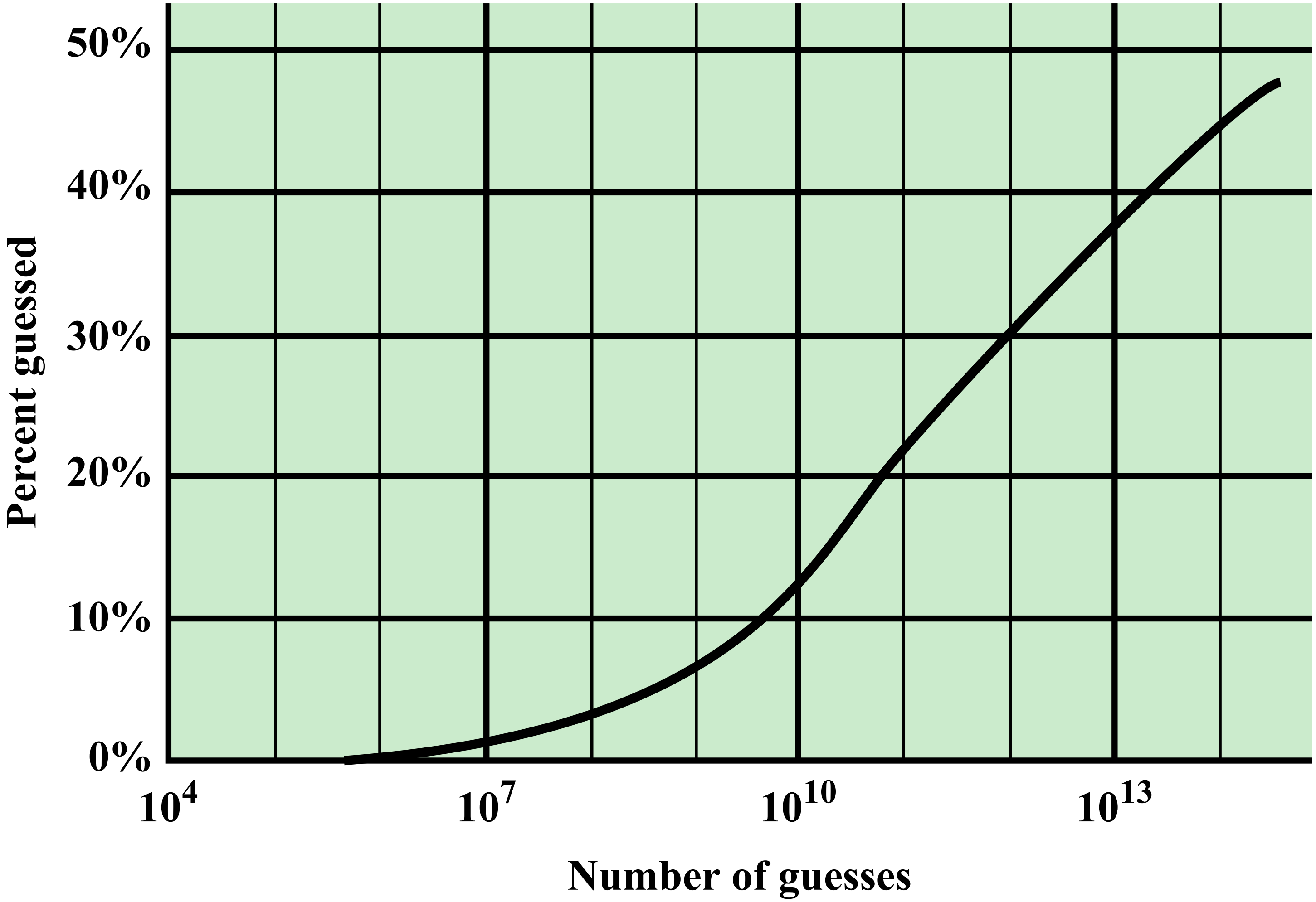


Figure 3.3 The Percentage of Passwords Guessed After a Given Number of Guesses

Password File Access Control

Can block offline guessing attacks by denying access to encrypted passwords

Make available only to privileged users

Shadow password file

Vulnerabilities

Weakness in the OS that allows access to the file

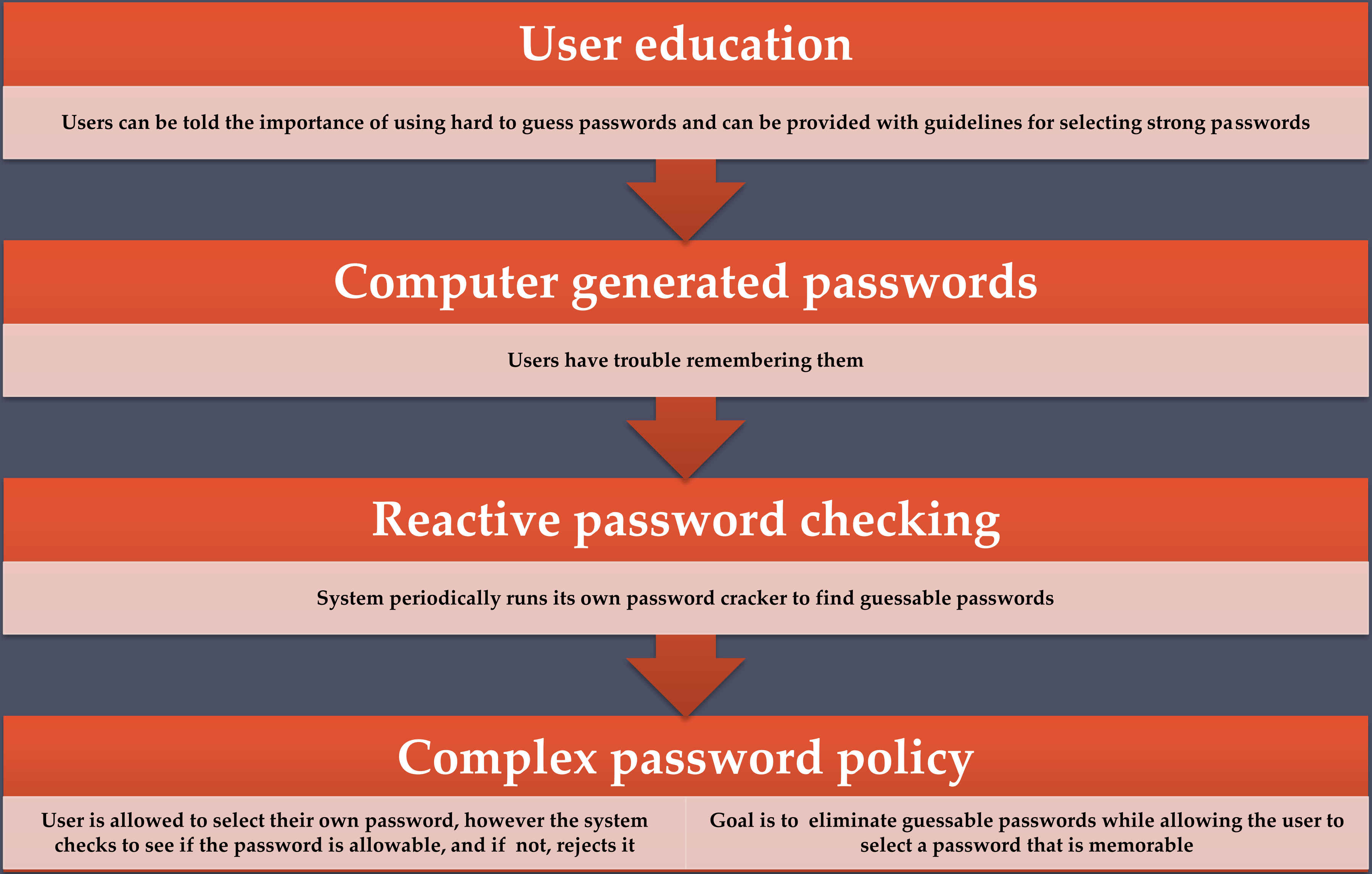
Accident with permissions making it readable

Users with same password on other systems

Access from backup media

Sniff passwords in network traffic

Password Selection Strategies



User education

Users can be told the importance of using hard to guess passwords and can be provided with guidelines for selecting strong passwords

Computer generated passwords

Users have trouble remembering them

Reactive password checking

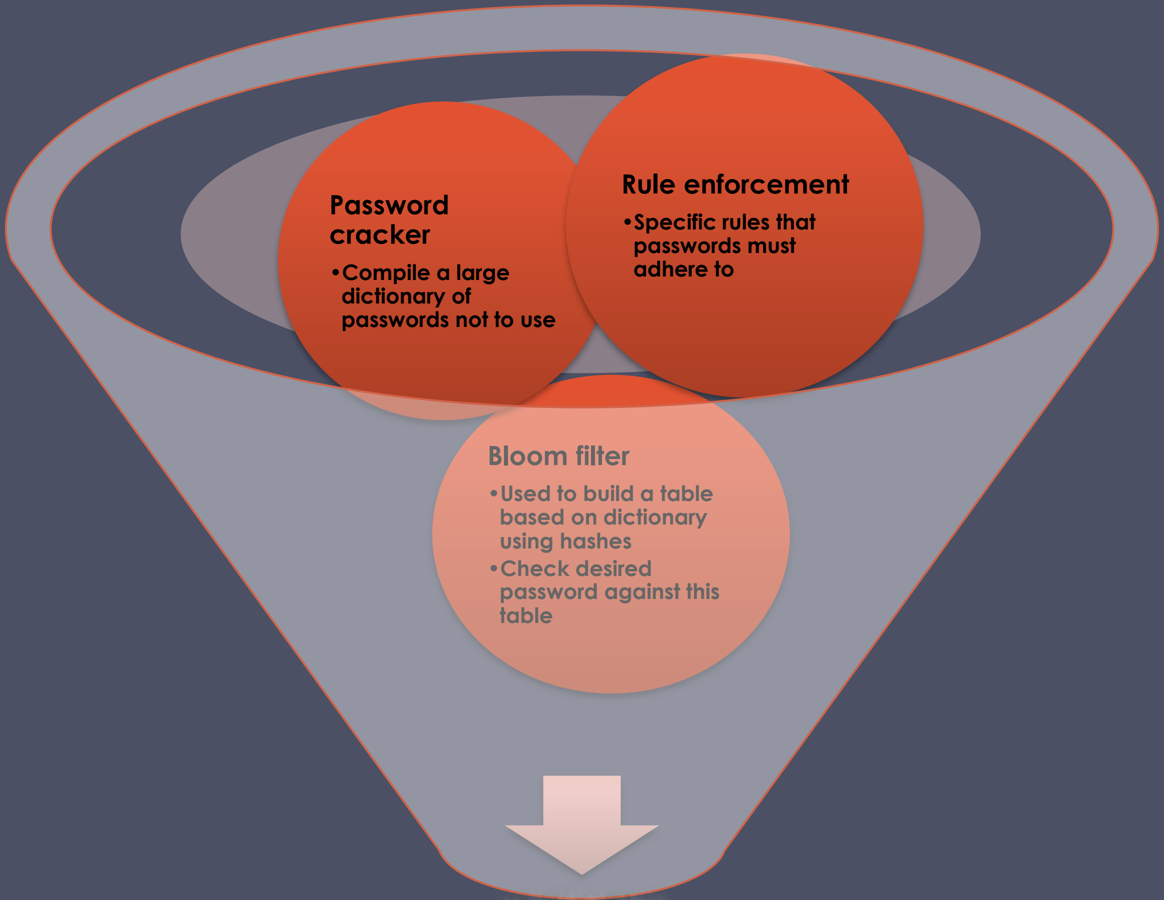
System periodically runs its own password cracker to find guessable passwords

Complex password policy

User is allowed to select their own password, however the system checks to see if the password is allowable, and if not, rejects it

Goal is to eliminate guessable passwords while allowing the user to select a password that is memorable

Proactive Password Checking



Password cracker

- Compile a large dictionary of passwords not to use

Rule enforcement

- Specific rules that passwords must adhere to

Bloom filter

- Used to build a table based on dictionary using hashes
- Check desired password against this table