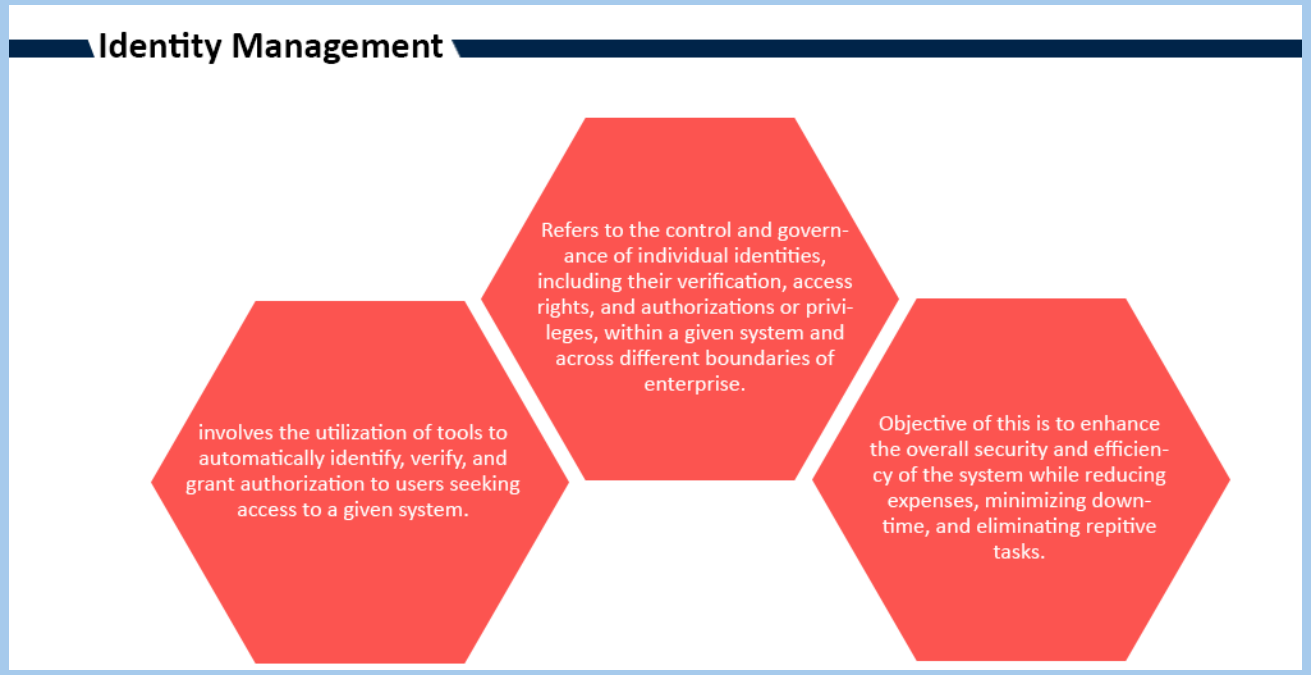
Domain 5

Identity & Access Management

Identity & Access Controls



Identity & Access Controls

Determine how and what type of access to grant users

Least Privileges

Need to know

Access

- Subject (user) will request access to object (file/folder)

Physical Access Control

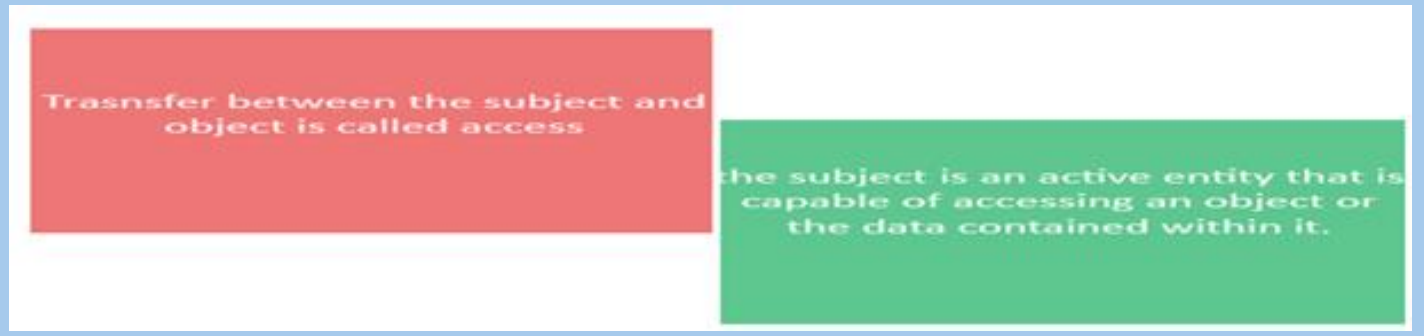
- Limiting physical access of people to access assets
- We have to know – Who is trying to access
- Identity must be know who is doing what

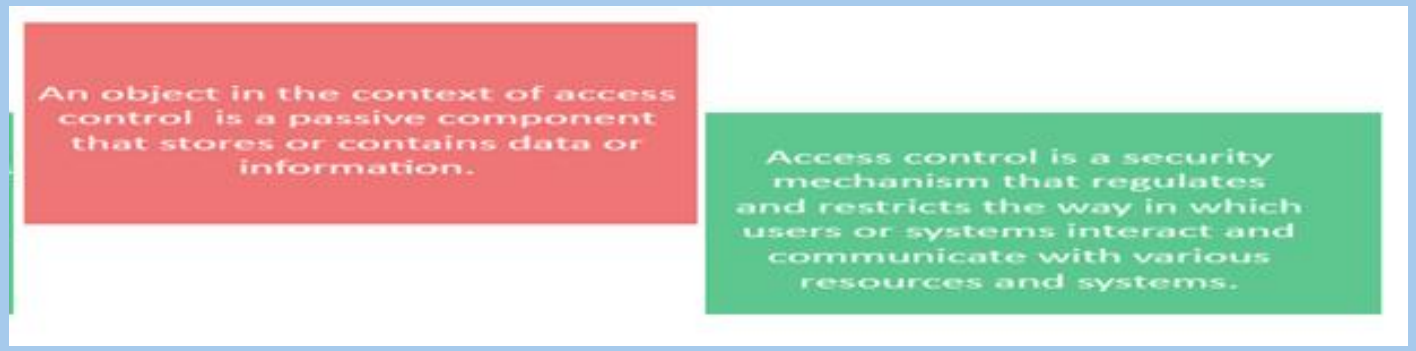
Subject

- Anything request to provide info

Object

- Anyone provide info

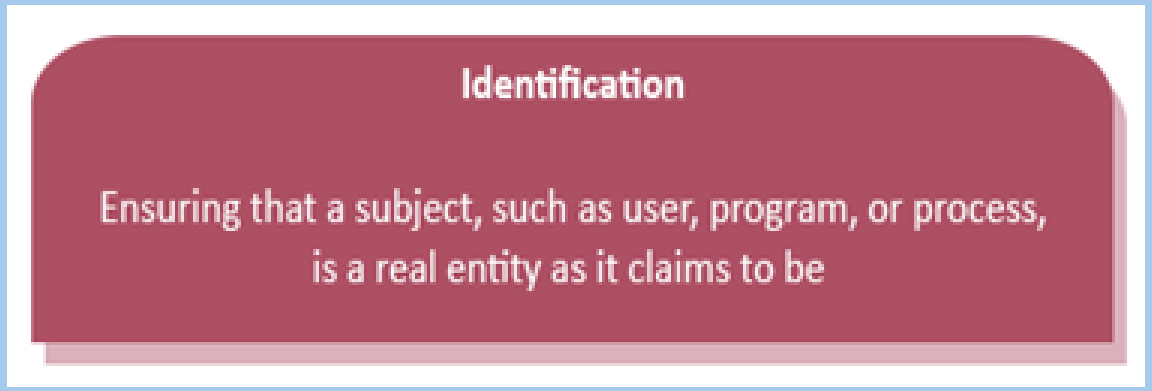


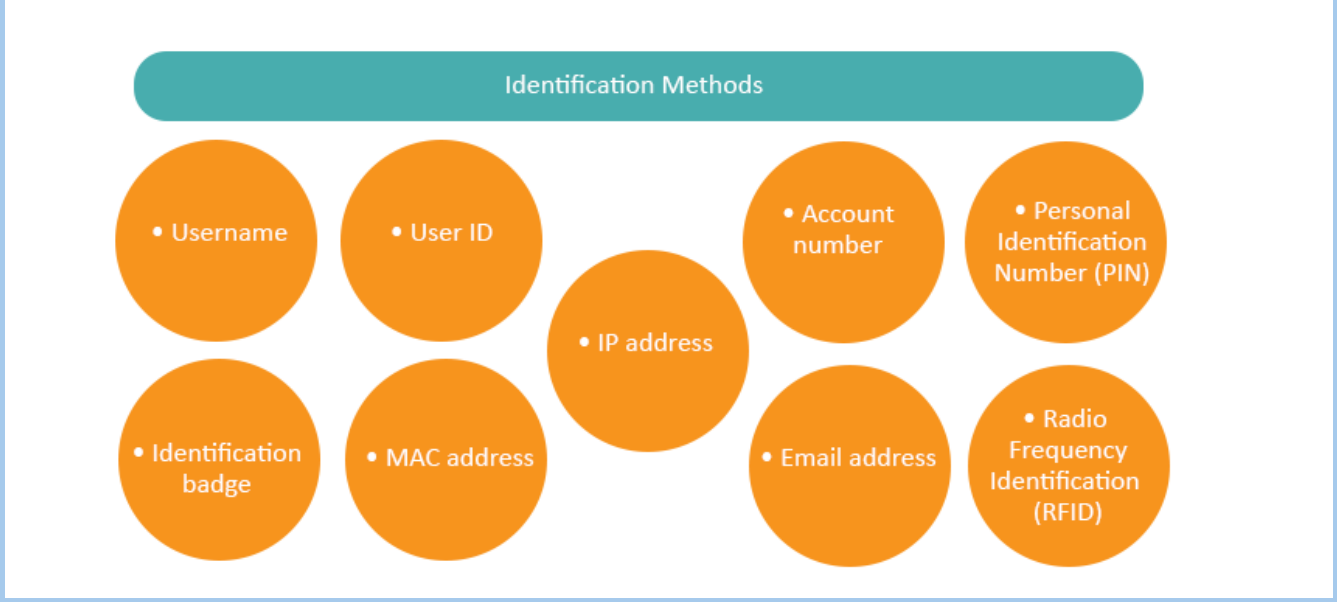


Identification, Authentication, Authorization & Accountability (AAA)

Identification

- Identify
    - Username
    - User ID
    - A/c no
    - MAC Address
    - Email
    - IP





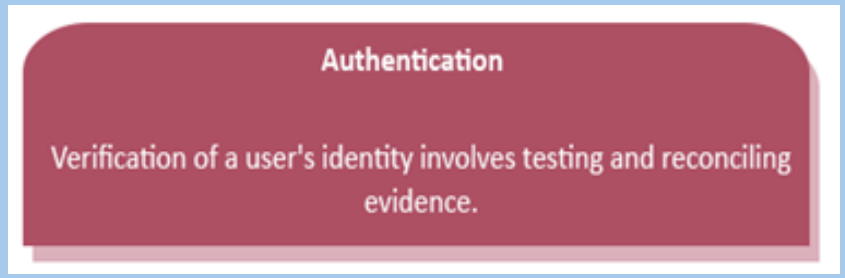
Authentication

Checking if username/password is ok – prove that you are the valid user

Something you know – PIN, password

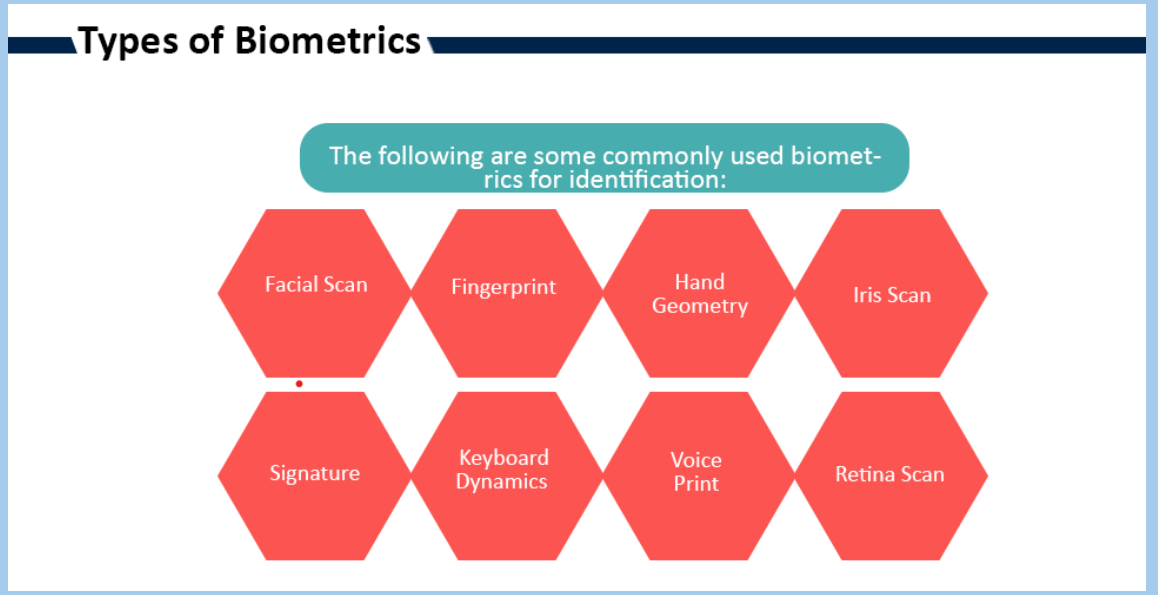
Something you have – Token, ID, smart card

Something you are – Retina scan, finger prints, facial scan



Types of Biometrics

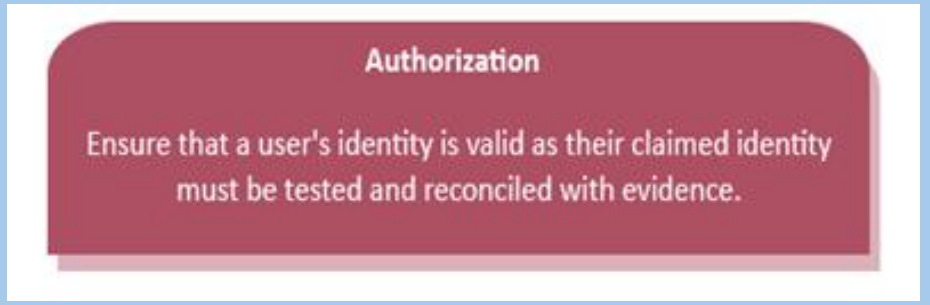
- Facial Scan
- Finger Prints
- Hand Geometry
- Iris scan
- Retina scan
- Voice print
- Keyboard dynamics
- Signature



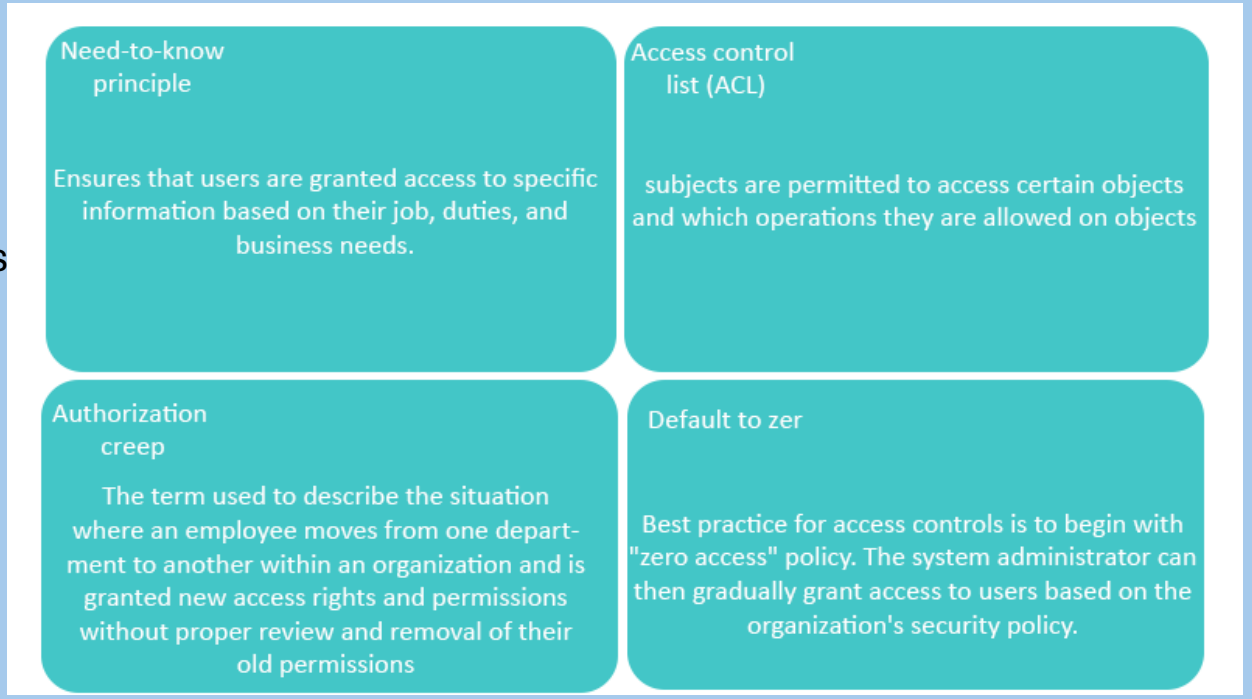
Authorization

If username/password ok, user can access the Data

Once identification and Authentication successfully done, checking permissions and access



- Default to Zero – By default No Access
- Need to Know – If business required, provide access
- Least Privileges
- Scope Creep – if employee switch the dept. old permissions should be removed
- ACL (Access Control List)



Accountability

- Track all the activities

Monitor who performed such activities

Type of audit

Can prove easily which user performed those activities

Non-Repudation

- Cannot denied the activity

How to Implement Accountability

- System logs
- Application logs
- Users logs (log on/log off etc)
- Strong Identification
- Strong Authentication
- Policies to enforce Accountability
- User Awareness Trainings
- Organization behavior
- Independent Audits
- Monitoring

Auditing

- Whatever activities done by users, should be captured in log files (valid/invalid activities)
- Invalid activates – might be threat
- Once Auditing, Logging & Monitoring implemented – Accountability done

FRR & FAR

- **FRR (False Rejection Rate)** – Type 1 error
  - System denied access of valid user

- **FAR (False Acceptance Rate)** – Type 2 error
  - System provide access to invalid user

  - Biometrics devices calibrated on CER (Crossover Error Rate)

- **CER (Crossover Error Rate)**
  - False rejection and False acceptance are same

    Measure the effectiveness of biometrics system

  - **False Positive**
    - Reported Vulnerability but actually not present

  - **False Negative**
    - There is Vulnerability but not reported

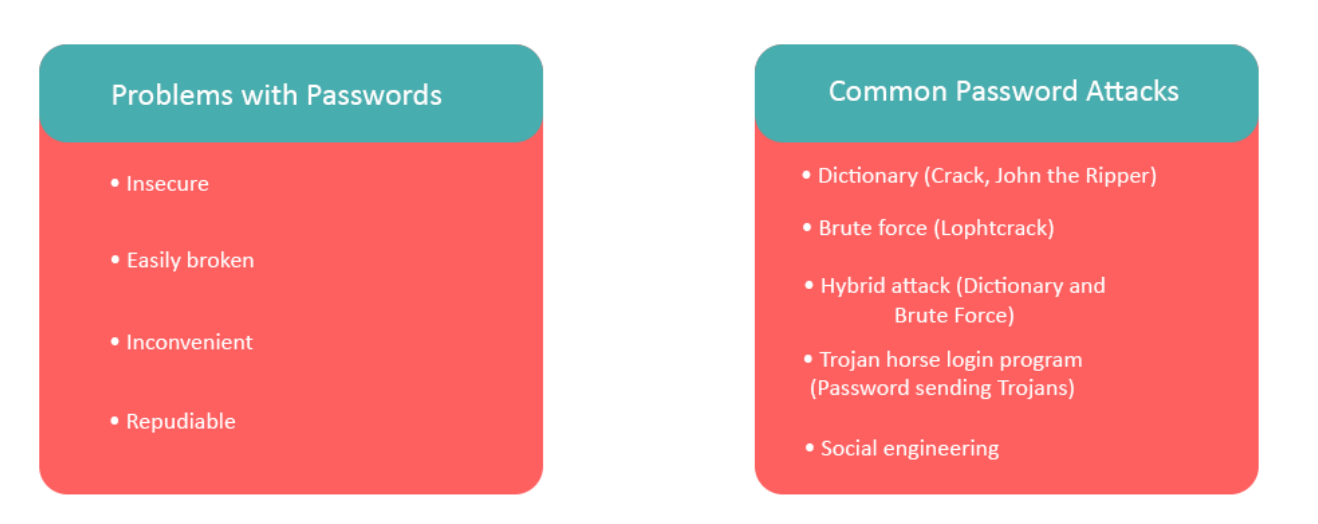
  - Smaller CER rate is more accurate
  - More sensitivity – FRR increases

Password types and Attacks

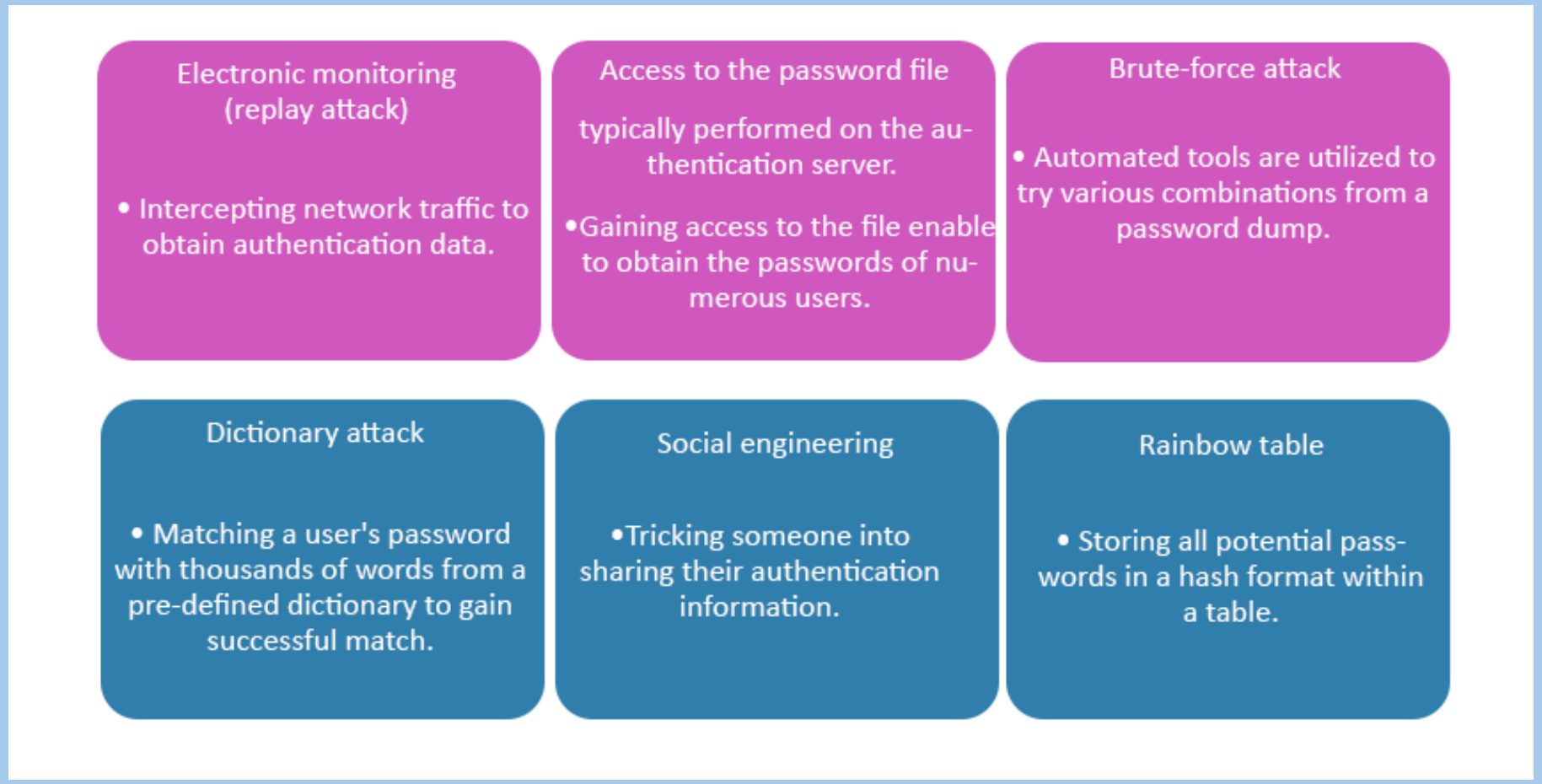
- Passwords are most common and weakest identification Method
- Always implement secure password policy

- **Dictionary Attack**
  - Use all common words in dictionary

- **Brute Force Attack**
  - Try all possible combination of passwords through Brute Force tool



Password types and Attacks



Tokens

- Can be either software or hardware based
- If an attacker gains control of the token, they can impersonate the token owner and gain access to the system or application that the token compromisng authentication protocol.
- Must be protected as they can be vulnerable to theft, damage, or loss

Synchronous Token

User don’t enter any number, it generates automatically

When token change every minute

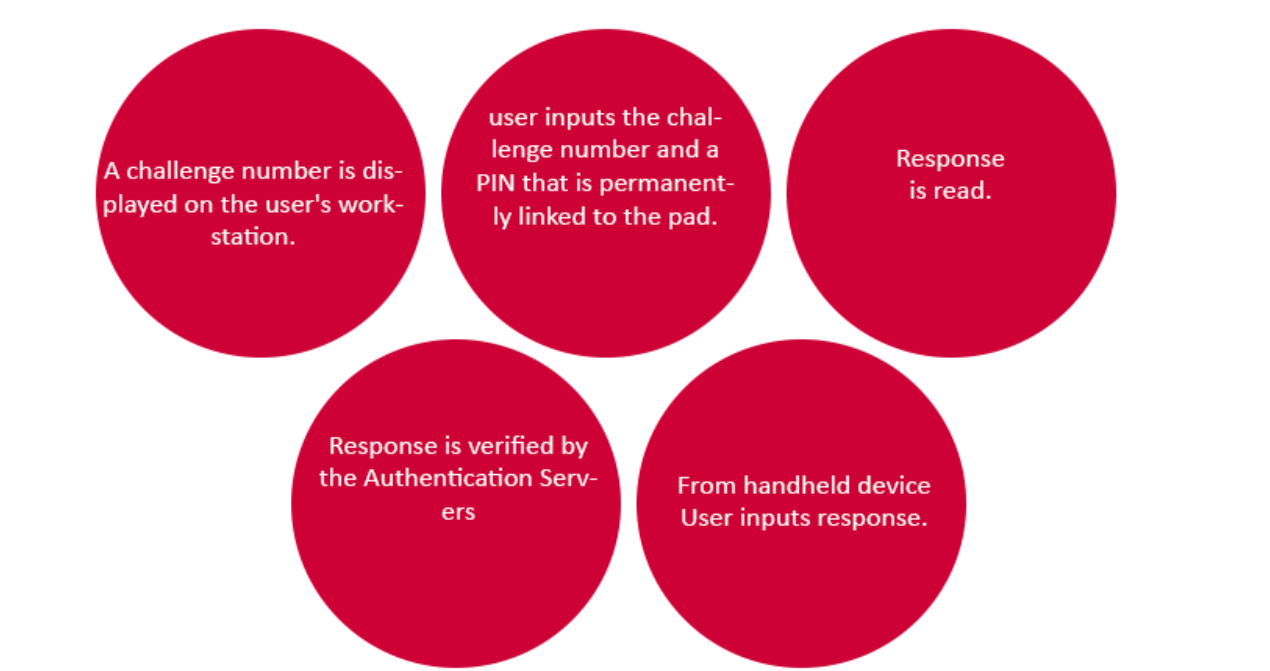
Time based

- it is necessary to maintain synchronizationof internal clocks between the token device and the authentication server
- A one-time password is generated using the secret key and the time on the token.
- One example of a time based synchronous token is the RSA token

Asynchronous Token

- More secured
- Not change with time but change when random number entered
- User will input some number to generate token

- **OTP**
  - Created by HMAC Algorithm
  - Generated by software but display on hardware/software



Federated Identity Management

- Once user1 login to App1 then Database will authenticate their credentials

- Org2 has no user Database. How user2 login to organization2 with their user Database
- User2 login to org2, they divert user2 to org1 Database to validate their credentials once verified they can use org2 applications

- Org2 – no need to have db authentication server

- Exp: From organization Domain we can login to insurance company

Single identity created for users and shared with all organizations within federation

Security Assertion Markup Language (SAML)

– Is XML technology use for authentication and authorization
  - For WebApp
  - OpenID
  - Oauth

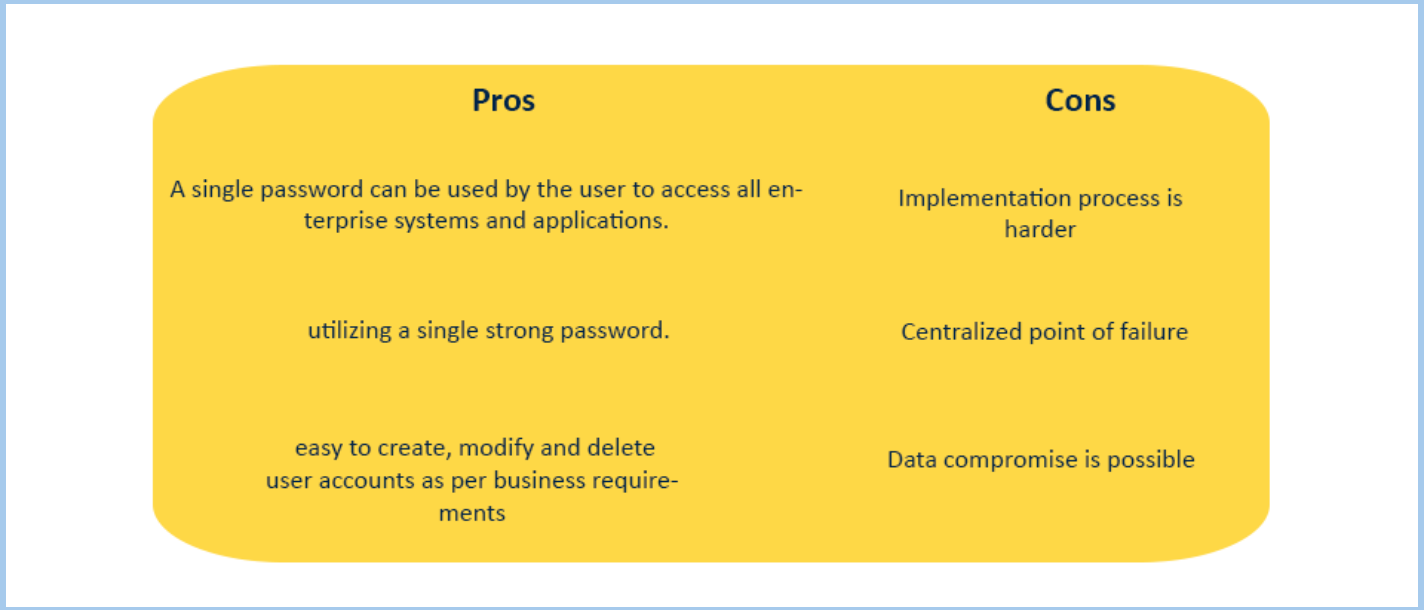
* Is a widely used XML standard that enables the secure exchange of authentication and authorization data between different security domains
* Critical component of federated identity management system it provides the necessary authentication and service mechanisms
* Federated identity systems commonly rely on SAML and (SPML) to facilitate access
* Provides (SSO) capabilities across different browsers and applications
* One important feature of SAML is that it does not have specific security mode of its own. Instead, it relies on use of Transport Layer Security (TLS) to ensure message confidentiality and the use of digital signature

Single Sign ON (SSO)

- Login only once
- By using SSO – Can authenticate multiple systems and applications

In SSO – If hackers compromise password then can enter in all the systems uses SSO

- User ---------- Active Directory
- Active directory verify username/password & issue ticket/Token/Cookies all are same
- Once ticket issued ---- next authentication ticket use for next authentication
- Ticket use for Authentication for this session



Kerberos

- Protocol to implement SSO
- Use Symmetric key Cryptography
- Tickets issued between user and

Authentication protocol generates tickets to allow nodes communicate over non secure network to prove their identity to each other in secure manner

Easy for end users, centralized control

Single point of failure – access everything with single password

Provide 2-way authentication by using tickets

Domain controller can be single point of failure

It is a widely-used authentication protocol that enables network-wide authentication.
Kerberos:

* Relies on symmetric key cryptography
* Designed to provide end-to-end security
* Has Following essential roles.

Key Distribution Center (KDC)
To generate symmetric key to secure session
KDC------AS + TGS

Authentication Center (AS)
Provide authentication

Ticketing Granting Server (TGS)
Generate tickets
TGT - Master
SGT -

Components

KDC – Key Distribution Center
TGS – Ticket Generating Server
AS – Authentication Server

User login to server
User name verify by authentication server -> if user name is available in Active Directory
By using hash value of password -> TGS generates key (only symmetric encryption use)
TGT saved on user machine

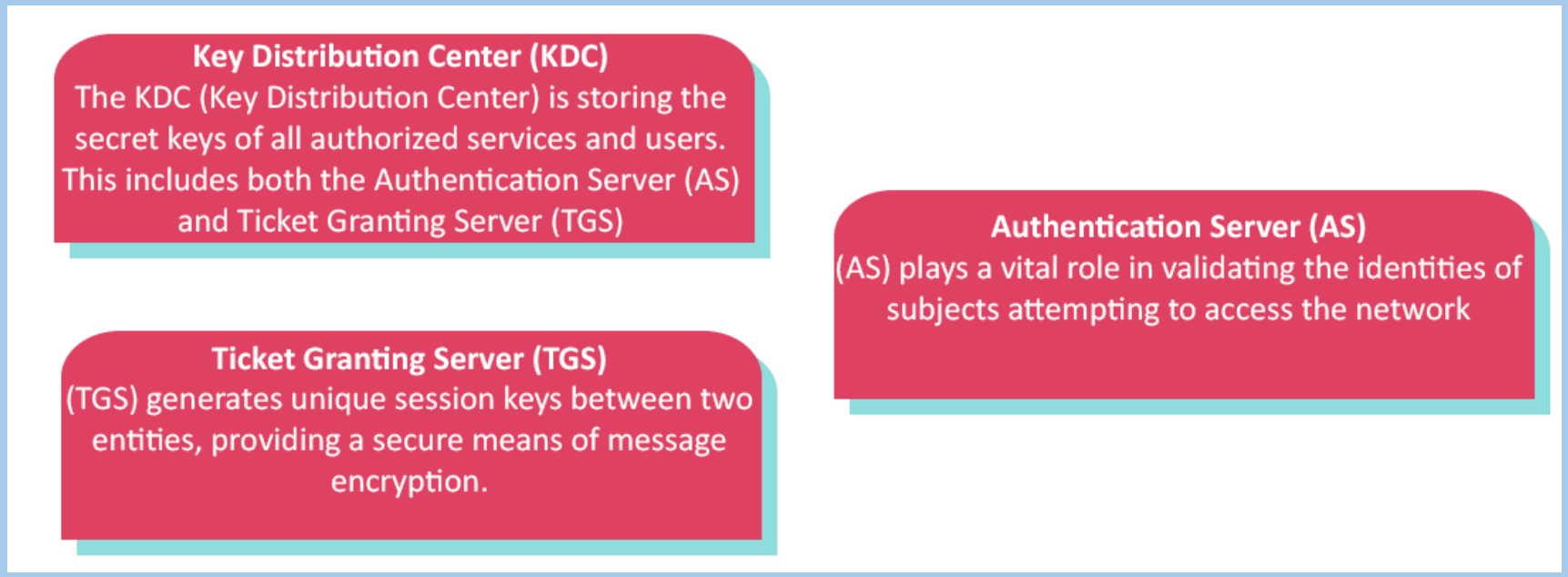
If user needs access on file or folder -> TGT request to TGS -> to issue ticket

Kerberos protocols are used in SSO in only internal environment

KDC--AS—To provide authentication
KDC—TGS—Tickets –TGT (Ticket Granting Ticket)
SGT – Service Granting Ticket

User—AS—Active Directory (Database)
Active Directory provide verify user & hash value of password

KDC - Generate key by using hash value of password
TGS – Generate tickets



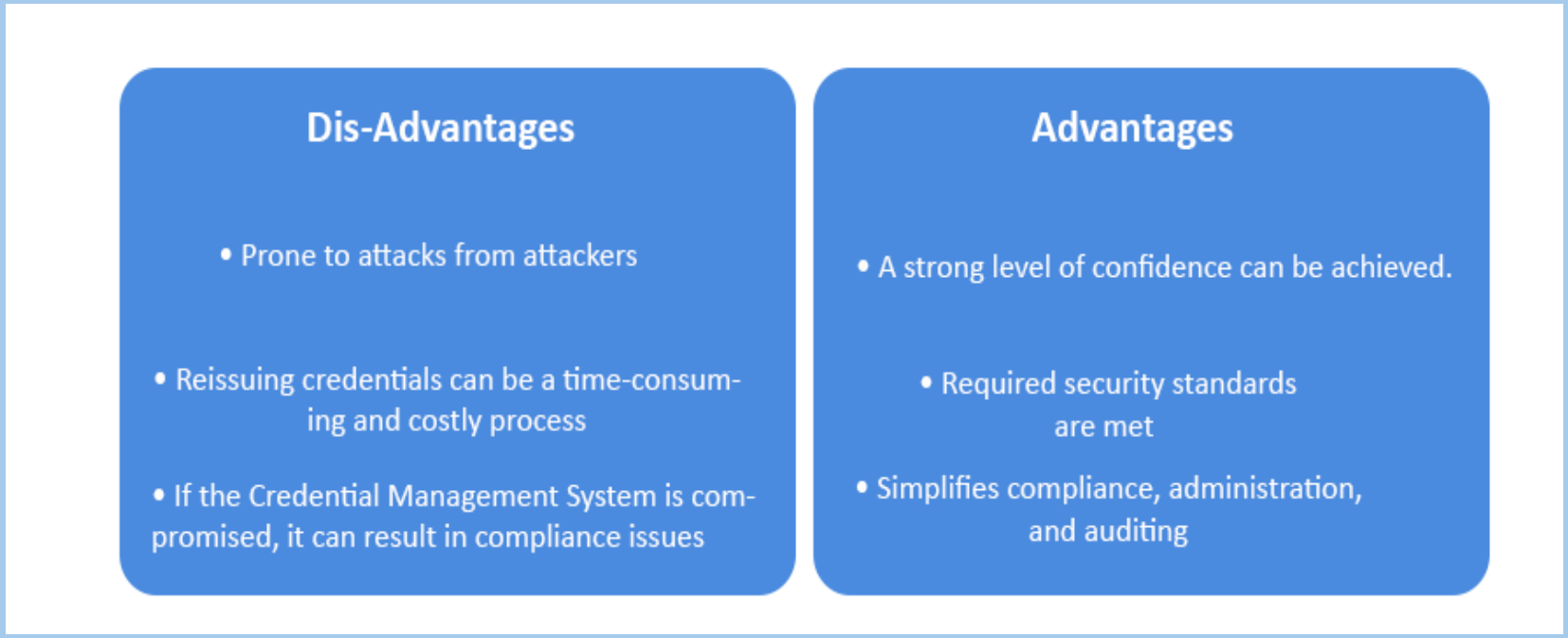
Issues:

KDC is single point of failure (there should be two Domains)
Clock must be synchronized
Security keys are temporarily used
If KDC hacked – all Security destroyed

- **Credential Management System**
  - If we have many credentials (15 apps/3 systems)
    - Help to manage multiple password
    - Can save the passwords for future use

  - Dis-Adv
    - If hackers compromised – All passwords compromised

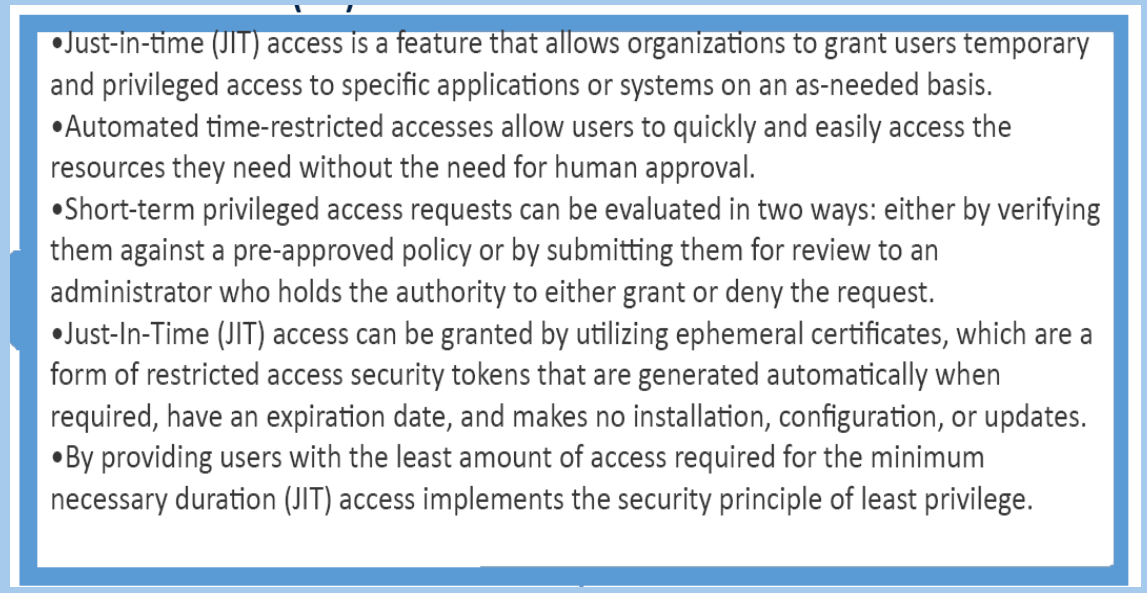
  - Adv
    - Provide more Security
    - Single Point Failure – there should be two servers for account authentication – if one server fails second can take over



Just In Time Access (JIT)

- By default – no access
- Only need access
- Least Privileges

- If user needs real time access → Only provides read access
- JIT – Use for Least Privileges
- Access for particular time period
- Access for definite time period and for one time only



Types of Access Control

DAC – Discretionary Access Control

Access based on owners discretion

*Data owners hold the authority to determine who can access their resources.

Mandatory Access Control (MAC)

Everyone has to follow the rules – no Excuse

Enforced access control

Computer systems determine access control

Need to know

* System's security policy is enforced by the operating system with the use of security labels.

Role Based Access Control (RBAC)

If provide same permission to many users, it saves lot of efforts
Just create group and assign permission

Access granted based on the role

Grant set of permissions instead of single permission

Assign permissions to the groups not directly to single user

Analysts are in one group

Rule Based Access Control

On firewalls create rules for access
If traffic coming from IP ..... denied
If rule 1 not followed then move to rule 2 onwards
Combination of IF & THEN statements

Access granted with if & then statement

Either access granted or denied

If user ID matched then he can gain the access

Access requests are evaluated against a specified list of predefined rules that determine what access should be granted.

Attribute Based Access Control

Access provide based on multiple attributes
IP, user login time, login date etc.

Access based on attributes, name, role, location, clearance

If user is in Sales, gives access to Sales folders only

Rule based access applied on all users but Attribute based is more specific

Only managers can access the tablets

Privileged Account (Admin account)

- More important – because doesn't want to compromise
- Provide special importance for privilege accounts

- **Privileged Escalation**

  - Attackers connected to application
  - If user is under attack and this is normal user then hacker can change the access rights and become admin user and can damage many accounts

When attackers exploit bug in a software to gain access

When user gain rights to another users or administrator