Authentication:

User authentication is the fundamental building block and the primary line of defense in computer security. It binds an identity to a subject and uses any of following three parameters to confirm the user identity. Two or more forms can be combined for more solid authentication; for example, a bank card and a PIN combine something the user has with something the user knows.

1. **Something the user *knows***: such as passwords, PIN numbers, or secret information
2. **Something the user *has*:** such as identity badges, physical keys, a driver's license, or smart card
3. **Something the user *is***: physical characteristic of the user, such as a fingerprint or a face.

A. Password based authentication

Passwords based authentication is an authentication mechanism based on what people know where the user supplies a password, and the computer validates it. If the password is the one associated with the user, user's identity is authenticated. If not, the password is rejected and the authentication fails. Password authentication has several vulnerabilities:

- Password may be easy to guess.
- Writing the password down and placing it in a highly visible area.
- Discovering passwords by eavesdropping or even social engineering.

Attacking a Password System

Brute Force Attack

In brute force attack, the attacker tries all possible passwords through some automated fashion. For example, if passwords are words consisting of the 26 characters AZ and can be of any length from 1 to 8 characters, there are 26^1 passwords of 1 character, 26^2 passwords of 2 characters, and 26^8 passwords of 8 characters. Therefore, the system as a whole has $26^1 + 26^2 + ... + 26^8 = 26^9$, possible passwords. This number seems big and if we use a computer to create and try each password at a rate of checking one password per millisecond, it would take on the order of 150 years to test all passwords. But if we can speed up the search to one password per microsecond, the work factor drops to about two months. This amount of time is reasonable.

Dictionary Attack

A dictionary attack is the guessing of a password by repeated trial and error. The name of this attack comes from the list of words (a "dictionary") used for guesses. If an attacker knows the personal information of the user, he can generate some meaningful password using automated software. The system cannot distinguish between the attacker and the legitimate user, and allows access.

B. Token Based Authentication

Definition: Token-based authentication is a security technique that authenticates the users using a security token provided by the server. A token is a data or some physical object (debit card) created by server, and contains information to identify a particular user. An authentication is successful if a user can prove to a server that he or she is a valid user by providing a security token.

Example: A one-time password, where password is generated by server and sent to user using specific mechanism (SMS). This password is invalidated as soon as it is used.

C. Biometric Authentication

Biometric is a biological authentication technique, based on some physical characteristic of the human body. There are devices to recognize the following biometrics: fingerprints, hand geometry (shape and size of fingers), eyes, voice, handwriting, blood vessels in the finger, and face. Authentication with biometrics has advantages over passwords because a biometric cannot be lost, stolen, forgotten, lent, or forged and is always available.

Problems with Biometrics

There are several problems with biometrics:

1. Biometrics are relatively new, and some people find their use intrusive. Hand
2. Biometric recognition devices are costly, although as the devices become more popular.
3. All biometric readers use sampling and establish a threshold for when a match is close enough to accept. There are some problems if, for example, your face is tilted or you press one side of a finger more than another.
4. Biometrics can become a single point of failure. For example, if credit card fails to register, I can always pull out a second card, but if my fingerprint is not recognized, I have only that one finger." Forgetting a password is a user's fault; failing biometric authentication is not.

Access Control Methods

A security policy may use two types of access controls, alone or in combination. In one, access control is left to the owner. In the other, the operating system controls access, and the owner cannot override the controls. The first type is based on user identity and is the most widely known:

1.1 Discretionary Access Control (DAC)

Definition: If an individual user can set an access control policies over an object, that mechanism is a discretionary access control (DAC), also called an identity-based access control (IBAC).

In DAC, access to objects (files, devices, etc.) is permitted based on user identity. Each object is owned by a user and he can specify how to share his objects with other users. For example, **s**uppose a child keeps a diary. The child controls access to the diary. The child allows her mother to read it, but no one else. Very common example of DAC is the Windows file system

1.2 Mandatory Access Control (MAC)

Definition: When a system mechanism controls access to an object and an individual user cannot alter that access, the control is a mandatory access control (MAC), also called a **rule-based access control.**

This access control policy is determined by the system and not the application or information owner. So, the identity is irrelevant. The operating system enforces mandatory access controls. Neither the subject nor the owner of the object can determine whether access is granted.

Example: The law allows a court to access driving records without the owners' permission. This is a mandatory control, because the owner of the record has no control over the court's access.

1.3 Role-Based Access Control (RBAC)

Definition: When a system mechanism controls access to resources based on the role assigned to a user, the control is a Role-Based Access Control (RBAC) model. In this model, an administrator assigns a role to a user that has certain predetermined right and privileges. RBAC is also known as Non-Discretionary Access Control. The roles assigned to users are centrally administered. RBAC is commonly found in government and military where the role definitions are well defined.

Example: A subject assigned the role of Manager will have access to a different set of objects than someone assigned the role of Analyst.

1.4 Attribute Based Access Control

Attribute-based access control (ABAC), also known as policy-based access control, defines an access control paradigm whereby access rights are granted to users through the use of policies which combine attributes together. The policies can use any type of attributes (user attributes, resource attributes, object, environment attributes etc.). This model supports Boolean logic, in which rules contain "IF, THEN" statements about who is making the request, the resource, and the action. For example: IF the requester is a manager, THEN allow read/write access to sensitive data.

RBAC uses pre-defined roles with set of privileges associated with them and to which subjects are assigned whereas ABAC is the concept of policies that express a complex Boolean rule set that can evaluate many different attributes. Its dynamic capabilities offer greater efficiency, flexibility, scalability and security than traditional access control methods, without burdening administrators or users.

3. Multilevel Security Model

Multilevel security system enforces access control “up and down” where security levels of objects and subjects are ordered in a hierarchy.

3.1 The Bell LaPadula model

It is a security policy model proposed by Bell and LaPadula in 1973. This is a mandatory access-control model for protecting **confidentiality**. It is a multilevel security model derived from traditional military multilevel security environment. The military database can hold information at a number of different levels of classification **e.g. unclassified < confidential < secret < top secret**. All subjects (processes, users, etc) and data objects (files, directories etc) are labeled with security level **e.g. unclassified < confidential < secret < top secret**. This military database system has to ensure that data can only be read by a user whose level is at least as high as the data’s classification level.

Figure: Multilevel security

How the BLP Model Works

The security levels in BLP form a partial order, ≤. Each object, x, is assigned to a security level, L(x). Similarly, each user, u, is assigned to a access level, L(u). Access to objects by users is controlled by the following two rules:

1. **Simple security property.** A user u can read an object x only if $L(x) \leq L(u)$.
2. ***-property.** A user u can write (create, edit or append to) an object x only if $L(u) \leq L(x)$.

The simple security property is also called the "**no read up**" rule, as it prevents users from reading objects with security levels higher than their own. The *-property is also called the "**no write down**" rule. It means no process may write data to a lower level.

For Example: With Bell-LaPadula, secret researchers can create secret or top-secret files but may not create public files; no write-down. Conversely, secret researchers can view public or secret files, but may not view top-secret files; no read-up.

3.2 The Biba Model

The Biba Model or Biba Integrity Model developed by Kenneth J. Biba in 1977. The Biba model is a classic model that describes a set of access control rules to ensure **data integrity.** Data and subjects are grouped into ordered levels of integrity. The model is designed so that subjects may not corrupt objects in a level ranked higher than the subject, or be corrupted by objects from a lower level than the subject. In general the model was developed to overcome a weakness in the Bell–LaPadula model which only addresses data confidentiality.

Data Integrity Goals: In general, preservation of data *integrity* has two goals:

- Prevent data modification by unauthorized parties
- Prevent unauthorized data modification by authorized parties

Similarities and Differences between the Biba model and the BLP model

This security model is directed toward data *integrity* (rather than *confidentiality*) and is characterized by the phrase: "no read down, no write up". This is in contrast to the Bell-LaPadula model which is characterized by the phrase "no write down, no read up". The objects and users are assigned integrity levels that form a partial order, similar to the BLP model.

The Biba Model Rules

• The access-control rules for Biba are the reverse of those for BLP. That is, Biba does not allow reading from lower levels and writing to upper levels.

• If we let I(u) denote the integrity level of a user u and I(x) denote the integrity level for an object, x, we have the following rules in the Biba model:

– A user u can read an object x only if $I(u) < I(x)$.

– A user u can write (create, edit or append to) an object x only if $I(x) < I(u)$.

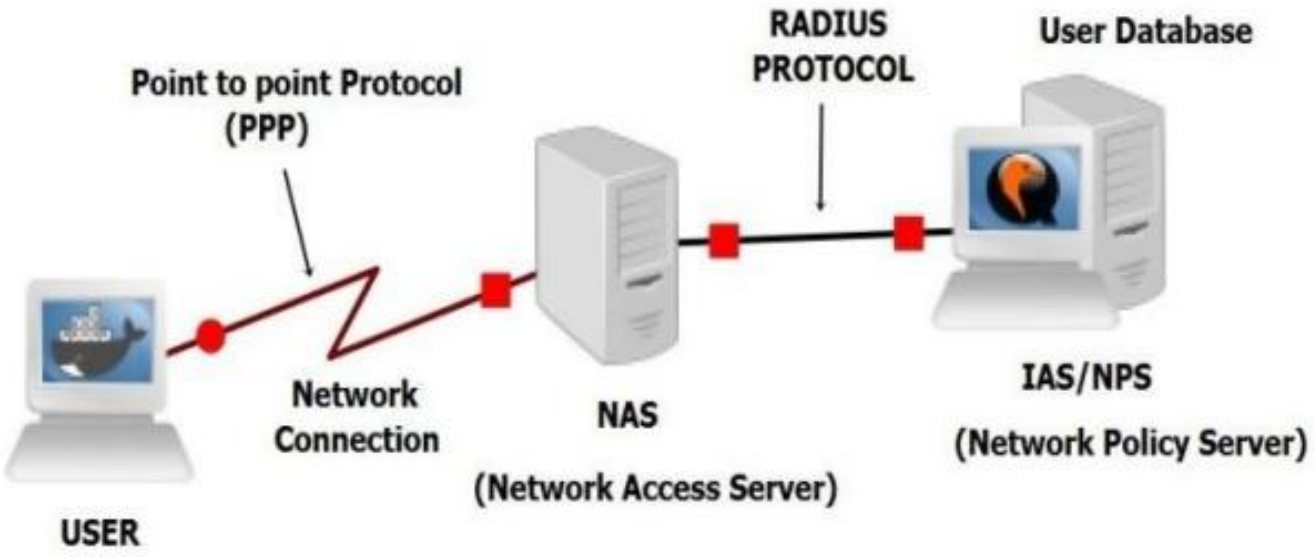
Thus, the Biba rules express the principle that information can only flow down, going from higher integrity levels to lower integrity levels.

RADIUS

Remote Authentication Dial-In User Service (RADIUS) is a networking protocol, operating on port 1812 that provides centralized Authentication, Authorization, and Accounting (AAA) management for users who connect and use a network service. It uses two packet types to manage the full AAA process; Access-Request, which manages authentication and authorization; and Accounting-Request, which manages accounting.

Working of RADIUS protocol

There are three components in the implementation of RADIUS namely: RADIUS client, Network Access Server (NAS) and RADIUS server. The user or machine (RADIUS client) sends a request to a Network Access Server (NAS) to gain access to a particular network resource using access credentials. A Point to Point Protocol (PPP) is used for communication between the end user and NAS. In turn, the NAS sends a RADIUS **Access Request** message to the RADIUS server, requesting authorization to grant access via the RADIUS protocol. This request includes access credentials, typically in the form of username and password or security certificate provided by the user. Additionally, the request may contain other information which the NAS knows about the user, such as its network address or phone number, and information regarding the user's physical point of attachment to the NAS. The RADIUS server checks that the information is correct using authentication schemes. The user's proof of identification is verified, along with, optionally, other information related to the request, such as the user's network address or phone number, account status, and specific network service access privileges. The RADIUS server then returns one of three responses to the NAS: 1) Access Reject, 2) Access Challenge, or 3) Access Accept.



Access Reject: The user is unconditionally denied access to all requested network resources. Reasons may include failure to provide proof of identification or an unknown or inactive user account.

Access Challenge: Requests additional information from the user such as a secondary password, PIN, token, or card. Access Challenge is also used in more complex authentication dialog where a secure tunnel is established between the user machine and the Radius Server in a way that the access credentials are hidden from the NAS.

Access Accept: The user is granted access. Once the user is authenticated, the RADIUS server will often check that the user is authorized to use the network service requested.

TACACS (Terminal Access Controller Access Control System)

TACACS (Terminal Access Controller Access Control System) is a simple UDP-based authentication protocol common to UNIX networks. TACACS allows a client to accept a username and password and send a query to a TACACS authentication server which uses TCP and usually runs on port 49. TACACS authentication server would determine whether to accept or deny the authentication request and send a response back.

TACACS+

It is an entirely new protocol and is not compatible with TACACS. TACACS+ is a CISCO designed extension to TACACS. TACACS+ improves on TACACS by separating the functions of authentication, authorization, and accounting (AAA) and by encrypting all traffic between the client device and the TACACS+ server. It uses TCP (while RADIUS operates over UDP). The difference between the RADIUS and TACACS+ is as follows:

RADIUS	**TACACS+**
It is Open standard protocol	It is Cisco proprietary protocol
It uses UDP as transmission protocol	It uses TCP as transmission protocol
It uses UDP port number 1812 for authentication and authorization and 1813 for accounting.	It uses TCP port number 49.
Here, Authentication and Authorization is combined in RADIUS.	Here, Authentication, Authorization and Accounting is separated in TACACS+.
Here, Only the password is encrypted while the other information such as username, accounting information etc are not encrypted.	Here, All the AAA packets are encrypted.
It does not support multi-protocol	It offers multi-protocol support
It is used for network access	It is used for device administration.
It does have to detect and correct transmission errors like packet loss, timeout etc. since it uses UDP	It does not have to implement transmission control.
It is less secure	It is more secure