Salesforce Data Security Model

Salesforce Data Security refers to the set of mechanisms and best practices that control how data is accessed, shared, and protected within the Salesforce platform. It ensures that users have the right level of access to data while preventing unauthorized access or modifications.

Types

- Organization-Level Security
- Object-Level Security (Profiles & Permission Sets)
- Field-Level Security
- Record-Level Security (Sharing Settings)

1. Organization-Level Security

- **IP Restrictions**: Restrict user access to Salesforce from specific IP ranges.
- **Login Hours**: Define login hours for users to restrict access outside business hours.
- **Multi-Factor Authentication (MFA)**: Adds an extra layer of security by requiring a second authentication factor.

2. Object-Level Security (Profiles & Permission Sets)

- **Profiles**: Define the base level of access for users (e.g., which objects they can read, create, edit, or delete).
- **Permission Sets**: Grant additional permissions to users without modifying their profiles.

3. Field-Level Security

- Restricts access to specific fields within an object.
- Can be controlled via **Profiles** and **Permission Sets**.

4. Record-Level Security (Sharing Settings)

Record-Level Security controls access to individual records within an object, ensuring users can only view or modify records they are authorized to acc ess.

Types of Record-Level Security

- OWD
- Role Hierarchy
- Sharing Rules
    - Owner Based Sharing Rule
    - Criteria Based Sharing Rule
- Manual Sharing
- Apex Sharing

OWD (Organization-Wide Defaults)

- Sets the baseline access for records (e.g., **Private, Read-Only, or Read/Write**).
- Example: If OWD for **Accounts** is set to **Private**, users can only see records they own unless additional access is granted.

Role Hierarchy:

- Allows users higher in the hierarchy to access records owned by subordinates.
- Example: A Sales Manager can access records owned by their Sales Reps.

Sharing Rules:

Sharing Rules provide additional record access to users, groups, or roles when **Organization-Wide Defaults (OWD)** are too restrictive. They automatically grant **Read-Only** or **Read/Write** access to records based on ownership or specific criteria.

Types of Sharing Rules

1. **Owner-Based Sharing Rules**
    a. Share records **owned by specific users, roles, or public groups** with other users.
    b. Example: Share all **Opportunities owned by Sales Reps** with the Finance team.
    c. **Use Case**: If OWD is **Private**, but Sales Managers need to see their team's records, an **Owner-Based Sharing Rule** can grant them access.
2. **Criteria-Based Sharing Rules**
    a. Share records that meet specific field conditions (e.g., Industry = "Healthcare").
    b. Example: Share all **Accounts where Type = "Premium Customer"** with the VIP Support team.

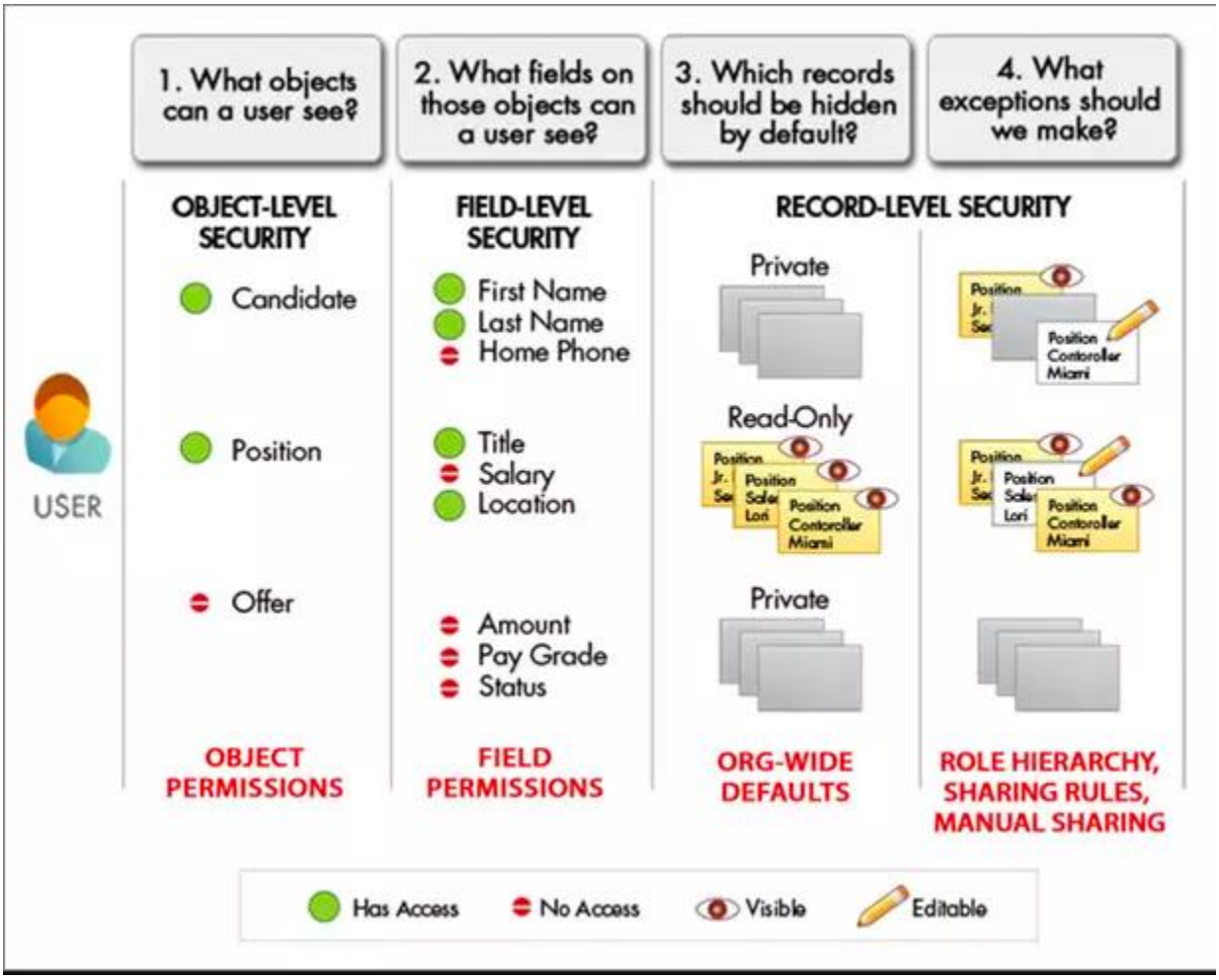
c. **Use Case**: If OWD is **Private**, but certain **high-value deals** need to be visible to executives, a **Criteria-Based Sharing Rule** can be created.

Manual Sharing:

- Users can share individual records with specific users or groups when needed.
- Example: A Sales Rep shares an important Lead with a colleague for collaboration.

Apex Sharing:

- Allows developers to apply custom sharing logic using Apex code.
- Example: If a custom object does not support standard sharing, Apex Sharing can dynamically assign access.



1. **What is Record-Level Security in Salesforce?**

Answer: Record-Level Security ensures that users can only access records they are authorized to view, create, edit, or delete. It is managed through Organization-Wide Defaults (OWD), Role Hierarchy, Sharing Rules, and Manual Sharing.

2. **What are Organization-Wide Defaults (OWD)?**

   Answer: OWD defines the baseline level of access to records for all users in the organization. The options are **Private**, **Public Read Only**, **Public Read/Write**, and **Controlled by Parent**.

3. **How does Role Hierarchy work in Salesforce?**

   Answer: Role Hierarchy allows users higher in the hierarchy to access records owned by users lower in the hierarchy. It enables a user in a senior role (e.g., Sales Manager) to see the records of their subordinates (e.g., Sales Rep).

4. **What is the difference between Role Hierarchy and Sharing Rules?**

   Answer: Role Hierarchy automatically grants access based on the user's role, while Sharing Rules are used to grant additional access to records based on ownership or criteria, independent of the role structure.

5. **Can a Sharing Rule make records more restrictive?**

   Answer: No, Sharing Rules can only make records more accessible but cannot restrict access to records.

6. **What are the different types of Sharing Rules in Salesforce?**

   Answer: There are two main types: **Owner-Based Sharing Rules** (shares records based on owner) and **Criteria-Based Sharing Rules** (shares records based on specific criteria like field values).

7. **Explain the concept of Manual Sharing.**

Answer: Manual Sharing allows users to share individual records with specific users or groups. This is typically used when a user wants to grant access to a record that is not automatically shared by OWD, Role Hierarchy, or Sharing Rules.

8. **What is the difference between Public Groups and Roles?**

   Answer: Public Groups are user-defined groups of users, roles, or other public groups, while Roles represent the hierarchical structure of users and their data access.

9. **How does Apex Sharing work?**

   Answer: Apex Sharing allows developers to programmatically share records based on custom logic using Apex code, offering more flexibility than standard sharing rules.

10. **What is the default sharing model for custom objects?**

    Answer: By default, the sharing model for custom objects is set to **Private** unless specified otherwise, meaning records can only be viewed by the owner and those explicitly granted access.

11. **What is the role of Profiles and Permission Sets in record-level access?**

    Answer: Profiles and Permission Sets control what users can see and do with objects and fields but do not directly control record-level access, which is managed by sharing settings.

12. **What are the limitations of Sharing Rules?**

    Answer: Sharing Rules cannot be applied to **public groups** and cannot **restrict access**—only grant additional access. Also, they only work with objects where OWD is set to **Private** or **Public Read-Only**.

13. **What are the different OWD settings available in Salesforce?**

Answer: OWD settings can be **Private**, **Public Read Only**, **Public Read/Write**, and **Controlled by Parent**.

14. **Can you control access to individual fields within a record?**

    Answer: Yes, field-level security allows you to control access to specific fields within an object using Profiles or Permission Sets.

15. **Explain what “Controlled by Parent” means in the context of OWD.**

    Answer: The "Controlled by Parent" OWD setting means that record-level access is controlled by the parent object. For example, if an Opportunity is linked to an Account, the opportunity's visibility is determined by the Account's OWD settings.

16. **What are the performance considerations when using Apex Sharing?**

    Answer: Apex Sharing should be used judiciously since it involves database operations. It is important to bulkify the code, minimize SOQL queries, and ensure that sharing rules are applied only when absolutely necessary to prevent governor limit violations.

17. **Explain how Criteria-Based Sharing Rules work with dynamic data.**

    Answer: Criteria-Based Sharing Rules dynamically share records when a specified field value or condition changes. For example, if an Opportunity's **Amount** field exceeds a specific value, the rule can automatically share it with a team.

18. **Can you create custom sharing rules for standard objects?**

    Answer: Yes, custom sharing rules can be created for most standard objects, such as **Accounts**, **Opportunities**, and **Contacts**, based on record ownership or criteria.

19. **What is the impact of setting OWD to Private for an object?**

    Answer: When OWD is set to **Private**, users can only see the records they own unless sharing rules or role hierarchy grants additional access.

20. **How would you handle sharing and visibility requirements for a highly sensitive dataset in Salesforce?**

    Answer: For highly sensitive data, I would use **Field-Level Security**, **Record-Level Security**, **Sharing Rules**, and **Platform Encryption**. The use of **MFA** and **IP restrictions** would also be recommended for additional security.

21. **Can you give an example where using Apex Sharing is better than using Sharing Rules?**

    Answer: Apex Sharing is better when custom logic is required that cannot be handled by Sharing Rules, such as sharing records based on complex business rules or dynamically adjusting sharing based on runtime conditions.

22. **How does Salesforce Shield enhance record-level security?**

    Answer: Salesforce Shield provides features like **Platform Encryption** (for encrypting data at rest), **Event Monitoring** (for tracking user activity), and **Field Audit Trail** (for tracking field changes), which enhance security at both the record and field levels.

23. **What is a Sharing Rule Junction Object?**

    Answer: A Sharing Rule Junction Object is used in many-to-many relationships for sharing purposes. It allows records to be shared with more than one group, role, or user.

24. **What is the relationship between Profiles, Permission Sets, and Record-Level Security?**

    Answer: Profiles and Permission Sets control object and field-level permissions, but **Record-Level Security** (via OWD, Role Hierarchy, Sharing Rules) determines access to individual records.

25. **How do you troubleshoot record-level security issues in Salesforce?**

    Answer: To troubleshoot, first check the OWD settings for the object. Then, verify the role hierarchy, sharing rules, manual sharing settings, and ensure that

users have appropriate permissions via Profiles and Permission Sets. Also, reviewing field-level security for critical fields is essential.

26. What is the importance of the sharing recalculation process in Salesforce?

Answer: The sharing recalculation process ensures that sharing settings are updated when ownership changes, when a sharing rule is modified, or when there are changes to roles or permissions. It guarantees that users have the correct access to records based on the latest criteria.

27. Can you explain how to use Territory Management with Record-Level Sharing?

Answer: Territory Management helps control access to records based on user territories. Users associated with a specific territory can be granted access to accounts within that territory, and the records are shared based on territory rules.

28. How do you grant record-level access to external users on Salesforce Communities?

Answer: Record-level access for external users is typically controlled via **Sharing Sets**, **Public Groups**, and **Profiles** in Salesforce Communities. You can configure record access for external users based on criteria and roles.

29. What is the relationship between OWD and Sharing Rules?

Answer: OWD defines the default access level for all records. Sharing Rules extend that access based on ownership or criteria. Sharing Rules cannot be more restrictive than OWD, but they can make records more accessible.

30. Explain the concept of "Implicit Sharing" in Salesforce.

Answer: Implicit Sharing automatically grants users access to related records (e.g., Account and its related Contacts) based on their access to the parent record. This is done without explicit sharing rules but follows the same record-level security model.

31. **What is the maximum number of sharing rules that can be created for an object in Salesforce?**

    Answer: The maximum number of sharing rules for a single object is **300** (this can vary with the type of Salesforce edition and other factors).

32. **What happens if a user has conflicting sharing rules?**

    Answer: If a user has conflicting sharing rules, Salesforce grants the broader level of access. If one rule grants read/write access and another grants read-only access, the user will have read/write access.

33. **Can Apex Sharing Rules be overridden?**

    Answer: Yes, Apex Sharing Rules can be manually overridden by administrators if required, but they should be used carefully since they can interfere with existing sharing configurations.

34. **What is the "Sharing Calculation" process, and why is it important?**

    Answer: The Sharing Calculation process ensures that sharing settings are recalculated when records are created or updated, making sure that users have the correct access based on ownership, sharing rules, and other security settings.

35. **Explain the role of "Object Manager" in Salesforce with respect to record-level security.**

    Answer: The Object Manager in Salesforce allows admins to configure object-level settings, including **field-level security**, **record types**, and **sharing settings** (such as setting up OWD, creating sharing rules, and configuring role hierarchy).

36. **What is the role of "AppExchange" in securing Salesforce records?**

    Answer: Apps on AppExchange may include custom security models or enhancements that extend or modify Salesforce's native record-level security.

Some apps offer additional encryption or multi-tenant features to enhance security.

37. **How do you handle data security in a multi-tenant Salesforce environment?**

    Answer: Salesforce ensures data isolation in a multi-tenant environment using advanced security mechanisms like **sharing rules**, **field-level security**, **audit trails**, and **encryption** to protect each organization's data.

38. **How does the Salesforce "View All" and "Modify All" permissions work at the record level?**

    Answer: The **View All** and **Modify All** permissions allow users to bypass record-level security. A user with these permissions can view or modify all records for an object, regardless of ownership or sharing settings.

39. **What is the maximum number of manual shares a user can create for a record?**

    Answer: Users can create up to **500 manual shares** for each record.

40. **How does the sharing recalculation process impact large-scale implementations?**

    Answer: The recalculation process can impact performance in large-scale implementations with thousands of records, especially when ownership or sharing rules are modified. Proper planning and execution are needed to minimize system impact during sharing recalculation.