**1. Defect Leakage**

Defect Leakage refers to the **bugs or defects that were not identified during a particular testing phase**, but are discovered in a **later stage** or **by end-users after release**.

**📌 Use Case:**

A QA team performs system testing on a mobile banking app. After release, users report that **notifications are not showing for failed transactions**.

**💡 Example:**

Let’s say:

* Test case for payment failure flow was tested without enabling the notification feature.
* Notification integration wasn’t verified.
* Defect leaked from **system testing to production**.

**💥 Impact:**

It affects user trust and may lead to financial or reputational damage.

### ****2. Defect Masking****

Defect Masking occurs when **one defect hides another**. This means a defect prevents another defect from being detected during testing.

#### 📌 ****Use Case****:

You are testing a shopping cart. The "Apply Coupon" button crashes the page. Later, you realize even if the crash didn’t happen, the **coupon logic would still have been incorrect** (e.g., wrong discount).

#### 💡 ****Example****:

|  |
| --- |
| **Defect A:** 'Apply Coupon' button throws an exception and crashes.  **Defect B:** Discount is calculated incorrectly even if it didn’t crash.  Defect A masks Defect B. |

#### 💥 ****Impact****:

Only Defect A is reported and fixed initially. After fixing A, Defect B surfaces later—wasting more time.

**3. Defect Cascading**

**🔍 Definition:**

Defect Cascading refers to a situation where **one defect leads to multiple other defects** in different parts of the system, often due to **dependency or propagation**.

**📌 Use Case:**

In a university portal:

* A defect in the login logic assigns the wrong user role.
* This causes wrong access permissions, incorrect dashboards, and misfired notifications.

**💡 Example:**

|  |
| --- |
| **Root Defect:** Incorrect role assignment during login (admin vs student)  **Cascading Defects:**  → Student sees admin dashboard.  → Student gets access to grade modification feature.  → Student gets notified as a teacher. |

#### 💥 ****Impact****:

Multiple test cases fail in different modules, making it hard to trace the root issue.

**Comparison Summary Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Term** | **Found In** | **Root Cause Example** | **Impact Flow** |
| Defect Leakage | Later stages (e.g., prod) | Missed test coverage | Missed → Passed → Released |
| Defect Masking | Testing phase itself | Crash hides deeper bugs | Bug A hides Bug B |
| Defect Cascading | Starts early, spreads | One wrong input propagates | Bug 1 → Bug 2 → Bug 3 |