Insecure Direct Object References

- What types of users do you have in your system?
- Do any users have only partial access to system data?
- The attacker who is an authorized system user, simply changes a parameter value that directly refers to a another object the user isn't authorized for.

Insecure Direct Object References

- Applications frequently use the actual name or key of an object when generating web pages.
- Applications don’t always verify the user is authorized for the target object. This results in an insecure direct object reference flaw.
- Testers can easily manipulate parameter values to detect such flaws. Such flaws can compromise all the data that can be referenced by the parameter.

Am I Vulnerable?

- The best way to find out if an application is vulnerable to insecure direct object references is to verify that all object references have appropriate defenses:
  - For direct references to restricted resources, the application needs to verify the user is authorized to access the exact resource they have requested.
  - If the reference is an indirect reference, the mapping to the direct reference must be limited to values authorized for the current user.

How Do I Prevent This?

- Preventing insecure direct object references requires selecting an approach for protecting each user accessible object:
  - Use per user or session indirect object references. This prevents attackers from directly targeting unauthorized resources.
  - Check access. Each use of a direct object reference from an untrusted source must include an access control check to ensure the user is authorized for the requested object.

Example Attack Scenario

- The application uses unverified data in a SQL call that is accessing account information:

```
String query = "SELECT * FROM accts WHERE account = ?";
PreparedStatement pstmt = connection.prepareStatement(query , … );
pstmt.setString( 1, request.getParameter("acct"));
ResultSet results = pstmt.executeQuery( );
```

- The attacker simply modifies the ‘acct’ parameter in their browser to send whatever account number they want. If not verified, the attacker can access any user’s account, instead of only the intended customer’s account.