LAB 06: ACCESS CONTROL VULNERABILITIES – IDOR

Part 1: Answer the following questions

1. What is an Insecure Direct Object Reference (IDOR), and how does it present a security risk in web applications?

It's a web application security vulnerability that occurs when an application allows unauthorized access to objects (like database records, files, or other resources) by using their internal identifiers (such as database keys, file paths, or numerical IDs) within URLs or parameters.

IDOR present a security risk in web applications:

- **Unauthorized Access to Sensitive Data:** attackers can exploit IDOR to view confidential data they shouldn't have access to, such as: customer records, financial transactions, personal information, proprietary business data
- **Modification or Deletion of Data**: can potentially alter or delete sensitive data, leading to integrity issues and damage to the application's functionality. ○ **Privilege Escalation**: can leverage IDOR to gain elevated privileges within the application, performing actions they're not authorized to do.

2. How can attackers exploit IDOR vulnerabilities in a website, and what are some common techniques used in such attacks?

Attackers employ several cunning techniques to leverage IDOR vulnerabilities and breach your website's security. Here are some common methods:

- **URL Tampering:**

  This involves directly manipulating the URL parameters containing object identifiers. For example, an attacker might change the user ID in a profile URL to access someone else's information.
- **Predictable ID Generation:**
  If IDs are predictable or follow a pattern, attackers can easily guess them and access unauthorized data.
- **Forced Browsing/Directory Traversal:**
  Attackers force the application to reveal internal file paths or directory structures by manipulating path parameters. This can lead to accessing unauthorized files or exploiting other vulnerabilities.
- **Session Hijacking:**
  In some cases, attackers can steal someone else's session cookie or token and use it to exploit IDOR vulnerabilities on their behalf, gaining access to their data.
- **Cross-Site Scripting (XSS):**
  Attackers might inject malicious scripts into a vulnerable website via XSS and leverage them to exploit IDOR vulnerabilities programmatically, automating data access or manipulation.

- **Brute-Forcing IDs:**
  Attackers might use automated tools to systematically try different ID values until they gain access to unauthorized data.
- **Social Engineering:**
  Tricking users into revealing their own information or clicking on malicious links can expose hidden object identifiers that attackers can exploit.

3. What types of functionality or data in a website can be affected as a result of an IDOR vulnerability being exploited?

User Accounts and Profiles: attackers could access, modify, or delete sensitive user information, including: ○ Personal details (names, addresses, phone numbers, emails) ○ Financial data (credit card numbers, bank accounts) ○ Passwords ○ Private messages ○ Order history ○ Account settings

Financial Transactions and Payment Information: unauthorized access to payment gateways or transaction records could lead to: ○ Fraudulent transactions ○ Identity theft ○ Data breaches

Private Messages and Communications: attackers could intercept or manipulate private messages, emails, or chat conversations.

Sensitive Documents and Files: unauthorized access to confidential documents, such as:

- ○ Business plans
- ○ Contracts
- ○ Legal documents
- ○ Medical records
- ○ Proprietary information

Administrative and Control Panels: attackers could gain access to administrative functions and settings, allowing them to:

- ○ Modify website content
- ○ Create or delete user accounts
- ○ Change system configurations
- ○ Steal sensitive data