IDOR - Santa’s Little IDOR

Day 5

Learn about IDOR while helping pentest the TrypresentMe website.

- IDOR on the Shelf

IDOR on the Shelf

Overview

Room URL: https://tryhackme.com/room/idor-aoc2025-zl6MywQid9

Difficulty: Medium

Category: IDOR

Date Completed: 12/5/2025

Objectives

- Understand the concept of authentication and authorization
- Learn how to spot potential opportunities for Insecure Direct Object References (IDORs)
- Exploit IDOR to perform horizontal privilege escalation
- Learn how to turn IDOR into SDOR (Secure Direct Object Reference)

Table of Contents

Introduction

McSkidy's investigation into the elven gift distribution system has uncovered a critical security flaw that could compromise the entire operation. The application managing parent accounts and child records is vulnerable to **Insecure Direct Object Reference (IDOR)** attacks, allowing attackers to bypass authorization checks and access data belonging to other users without permission. This vulnerability demonstrates a fundamental breakdown in access control—the system authenticates

users but fails to verify whether they have permission to view specific data. Through a series of escalating challenges, you'll learn how IDOR vulnerabilities manifest in different forms: from obvious sequential IDs to encoded values and cryptographic hashes. More alarmingly, you'll discover how even seemingly random identifiers like UUIDs can be predictable when generated using deterministic algorithms. Understanding these attack vectors is essential for securing applications against unauthorized data disclosure and horizontal privilege escalation.

Key Information

- **Vulnerability Type:** Insecure Direct Object Reference (IDOR) / Authorization Bypass
- **Impact:** Horizontal privilege escalation allowing access to other users' sensitive data
- **Root Cause:** Missing server-side authorization checks on data access requests
- **Obfuscation Methods Encountered:** Sequential IDs, Base64 encoding, MD5 hashing, UUID v1 generation
- **Critical Lesson:** Security through obscurity (encoding/hashing) is insufficient; server-side authorization verification is mandatory

---

Walk Through

1. What does IDOR stand for?
   - `Insecure Direct Object Reference`
2. What type of privilege escalation are most IDOR cases?
   - `horizontal`
3. Exploiting the IDOR found in the `view_accounts` parameter, what is the `user_id` of the parent that has 10 children?
   1. Spin up attack box and the target machine.
   2. Open the website at `http://10.67.179.187`
   3. Log in using `niels:TryHackMe#2025`
   4. Open developer tools `ctrl+shift+i`
   5. Open the network tab
   6. Click reload to load the network packets
   7. Click the one that says `view_accountinfo`
   8. On the right click responses

9. Right click the packet on the left and select edit and resend
10. Edit the `user_id` and send
    - idor1.png
    - There are ten children on the right `0-9`

4. Bonus Task: If you want to dive even deeper, use either the base64 or md5 child endpoint and try to find the id_number of the child born on 2019-04-17? To make the iteration faster, consider using something like Burp's Intruder.
    1. Used Burp Suite for this
    2. Open BurpSuite and turn intercept off
    3. Enable proxy in firefox and naviagte to the url
    4. View HTTP histroy in Burp Suite Proxy
    5. Send `/api/child/b64/Mg==` to the intruder
    6. Set `MG==` as Payload
    7. Set payload type to numbers
    8. I chose numbers `0-100`
    9. Under payload processessing add encode to base64
    10. Start attack
    11. Click on the first packet
    12. At the bottom click view response
    13. Using the down arrow key I went through each until I found the entry I was looking for.
    14. `19`
        - The data set only has 20 children.
        - IDOR2.png

5. Want to go even further? Using the /parents/vouchers/claim endpoint, find the voucher that is valid on 20 November 2025. Insider information tells you that the voucher was generated exactly on the minute somewhere between 20:00 - 24:00 UTC that day. What is the voucher code?
    1. The vouchers use UUID version 1 which is insecure, however, at this time, I am doing more research as I was unable to complete this one.

Lessons Learned

- **Mastered multiple IDOR exploitation techniques** including direct ID manipulation, Base64-encoded parameter iteration using Burp Suite Intruder, and hash-based access control bypasses, demonstrating that obfuscation alone cannot prevent unauthorized access without proper server-side authorization checks
- **Understood the distinction between authentication and authorization** and recognized that IDOR represents a horizontal privilege escalation vulnerability where authenticated users gain access to data they shouldn't be permitted to view, reinforcing that authorization must be verified on every request regardless of how object references are disguised

---

Resources

TryHackMe

Hash Identifier

UUID Decoder