- Filename: eccouncil-ceh31250-v12-15-3-1-blind-based-sqli-attacks.md
- Show Name: CEHv12 (312-50)
- Topic Name: Web Application Hacking: SQL Injection
- Episode Name: Blind-based SQLi Attacks
  ==========

Blind-based SQLi Attacks

Objectives:

- Explain and demonstrate how to use Blind-based SQLi to access sensitive information

---

- Today we're looking into 'Blind' SQL Injection. What does that mean?
    - No visible indicators of a (un)successful injection
- Are there techniques that we can use to verify whether or not an injection is successful?
    - Boolean-based
    - Time-based
- Can you show us an example of a Boolean-Based Injection?
    - Boolean Demo
        - Test for SQLi with single-quote (`'`)
            - Custom error returned, but it looks like special char filtering
        - Try Boolean injection
            - TRUE/FALSE conditions
                - `' OR 1=1 -- -` is TRUE
                - `' OR 1=3 -- -` is FALSE
                - One or both could be useful
            - Now we continue with ORDER BY column enumeration
                - `iron man' order by 1 -- -`
                - Site throws custom error when invalid column is requested
                - `iron man' order by 8 -- -`
                - "*Invalid Syntax Detected!*"
                - Now we know there are 7 columns in the table
                - Then continue DB enumeration with UNION SELECT
- You also mentioned Time-Based Blind injections? How does that work?
    - Lack of feedback from injection tests
        - bWAPP Time-based challenge doesn't return ANY ERRORS!
            - Must find some way of verifying test success/failure
                - Timed responses
- So we force, the app to wait before it responds?
    - Add `-sleep()` to the test
        - `iron man' -sleep(1) -- -`
            - The site should 'sleep' for 10 seconds, then return results
                - If site hangs, then SQLi test is successful

- This becomes our success/failure indicator
    - `iron man' order by 1 -- -` has no indication of success/fail
    - `iron man' order by 8 -- -` has no indication of success/fail
        - Add `-sleep(0.5)` to make it hang 5 seconds
            - `iron man' -sleep(0.5) order by 1 -- -` hangs, SUCCESS!
            - `iron man' -sleep(0.5) order by 8 -- -` no hang, FAILURE!
                - We can then deduce that there are 7 columns in the table