

Scenáre pre SQL Injekcie

Modifikácia podľa loginu dodávateľa pri neznalosti emailu

Používateľ nepozná cieľový email, ale pozná login. Jeho úlohou je zmeniť login aj email. Systém ale vyžaduje znalosť emailovej adresy pri modifikácii. Cieľom by mohlo byť podhodiť nepravé informácie a odchytiť emaily zo súkromnej emailovej adresy.

Výpis databázy:


Data Output Explain Messages Notifications Query History			
	company_id [PK] integer	company_name character varying (50)	company_email character varying (60)
1	2	admin	admin@company.com

Položku by bolo možné vyhľadať v systéme:

The screenshot shows the pgAdmin 4 interface. On the left, the 'Suppliers' table is selected in the 'Servers' tree. The main window displays the 'Suppliers' table with columns: Company name, Company email, Modify, New company name, New company email, and Delete. The first row contains 'admin' and 'admin@company.com'. Below the table, a search modal is open, allowing search by 'Company name' or 'Company email'. The 'Company name' field contains 'admin'. Search options include 'Search at the beginning', 'Search in the end', and 'Search in the middle'. Navigation buttons like 'Next', 'Previous', 'Next results', 'Prev results', and 'Find' are visible. On the right side of the modal, there is a vertical list of actions: 'Add supplier', 'Add suppliers from file', 'Remove supplier', 'Modify supplier', 'Hide supplier', 'Multiple option', and 'Return'.

Postup exploitácie

1. *Nájdenie formulára, ktorý po aplikovaní znaku ' sa nezatvorí, alebo neinformuje o chybe (zlyháva):*



The image shows a screenshot of a web application window titled "Modify supplier". The window has a close button (X) in the top right corner. The background of the window is a grayscale image of a desk with a fan. The form contains four input fields with the following labels and values:

- Company name**: The input field is empty, with a single quote character (') visible at the end of the text.
- Company email**: The input field contains the text "skip@skip".
- New company name**: The input field contains the text "SQL Inject".
- New company email**: The input field contains the text "inject@inject.com".

At the bottom of the window, there are two buttons: "Confirm" and "Close".

2. *Tvorba injekcie:*

Útočník predpokladá, že SQL príkaz vyzerá nasledovne:

UPDATE supplier SET company_name=.....



The image shows a 'Modify supplier' dialog box with a background image of a desk lamp. It contains four text input fields, each with a label above it and a button below it. The labels are 'Company name', 'Company email', 'New company name', and 'New company email'. The input fields contain the following text: 'admin' or 1=1 --', 'skip', 'SQL Inject', and 'inject@inject.com'. The buttons are 'Confirm' and 'Close'.

Field Label	Input Value
Company name	admin' or 1=1 --'
Company email	skip
New company name	SQL Inject
New company email	inject@inject.com

3. Exploitácia dopadla úspešne:

Data Output			
	company_id [PK] integer	company_name character varying (50)	company_email character varying (60)
1	2	SQL Inject	inject@inject.com

Vysvetlenie:

V databáze nie je ošetrovaný SQL príkaz. Tvorí sa zjednotením reťazcov textu, ktoré sa nekontrolujú a ani parametricky do tohto reťazca nevkladajú. Útočník v prvom kroku zistí, že cieľová stanica neošetroje vstupy zadáním znaku ' do kolónky pre login. Ten spôsobí, že sa reťazce nesprávne uzatvoria. V prípade neošetrenia vstupu, to jednoznačne skončí výnimkou. Útočník v ďalšej fáze môže hádať SQL príkaz. Keďže má prístup k formuláru pre zmenu emailu na základe loginu a ešte neošetrenom heslom, stačí mu vložiť SQL kód do kolónky pre login. Najprv zadá login v systéme a uzavrie ho znakom ', tým povie, že chce zmeniť tohto používateľa, následne ale pridá podmienku 1=1, čím indikuje, že prvá podmienka nemusí platiť, lebo táto platí vždy. Pre úplnosť zakomentuje zvyšok SQL príkazu použitím komentáru pre postgres, pokiaľ vie o tomto type databázy. Inak môže vyskúšať aj ďalšie typy komentárov, napríklad znak #, ktorý je typickým komentárom v Maria DB. Takýto príkaz, ale zmení všetkých používateľov, preto podmienka 1=1, nemusí byť vložená.

Zadaný príkaz vyzerá po dosadení injekcie nasledovne:

```
UPDATE supplier SET company_name='SQL Inject', company_email='inject@inject.com'
WHERE company_name='admin' or 1=1 --' ' AND company_email='skip@skip.com'
```

Podmienka nie je nutná:

```
UPDATE supplier SET company_name='SQL Inject', company_email='inject@inject.com'
WHERE company_name='admin' --' ' AND company_email='skip@skip.com'
```

Chybný kód na strane servera:

```
public void modifySupplier(String companyName, String companyEmail,String
    companyModifiedName, String companyModifiedEmail) throws SQLException
{
    String prepareModifySupplier;
    PreparedStatement preparedStatement;

    prepareModifySupplier =
        "UPDATE supplier
        SET company_name='"+ companyModifiedName +
        "', company_email='"+ companyModifiedEmail +"
        WHERE company_name='" + companyName +"
        AND company_email='" +companyEmail +""";

    preparedStatement = connectionToDatabase.getConnection()
        .prepareStatement(prepareModifySupplier);
    preparedStatement.executeUpdate();
    preparedStatement.close();
}
```

Modifikácia podľa loginu dodávateľa pri neznalosti emailu - riešenie

Pri hľadaní a ošetroení SQL injekcií je potrebné nájsť slabinu v kód umožňujúcu SQL injekciu. V praxi sa odporúčajú používať parametrizované query, prípadne vstup kontrolovať pomocou regexov, alebo iného overenia na možné injekcie. Rovnako ako útočník môžeme pri hľadaní podhodit' znak ' do okna formuláru a pozrieť sa do logov, či niečo pri vyhodnocovaní query nezlyhalo.

Postup ošetrovania

1. *Nájdenie formulára, ktorý po aplikovaní znaku ' sa nezatvorí, alebo neinformuje o chybe (zlyháva):*



2. *Tvorba parametrizovaného query:*

```
public void modifySupplier(String companyName, String companyEmail,String  
    companyModifiedName, String companyModifiedEmail)  
    throws SQLException  
{  
    String prepareModifySupplier;  
    PreparedStatement preparedStatement;
```

```
prepareModifySupplier = "  
    UPDATE supplier  
    SET company_name=?, company_email=?  
    WHERE company_name=? AND company_email=?";  
preparedStatement = connectionToDatabase.getConnection()  
    .prepareStatement(prepareModifySupplier);  
preparedStatement.setString(1,companyModifiedName);  
preparedStatement.setString(2,companyModifiedEmail);  
preparedStatement.setString(3,companyName);  
preparedStatement.setString(4,companyEmail);  
preparedStatement.executeUpdate();  
preparedStatement.close();  
}
```