обчислюваних можливостей, складнощі використання клітинних автоматів стають все менш актуальними.

Список використаних джерел:

1. Cellular Automata. Stanford Encyclopedia of Philosophy [Електронний ресурс] – Режим доступу до ресурсу: https://plato.stanford.edu/entries/cellular-automata/.

2. Index to Elementary Cellular Automata [Електронний ресурс] – Режим доступу до ресурсу: https://oeis.org/wiki/Index_to_Elementary_Cellular_Automata.

3. Elementary Cellular Automaton [Електронний ресурс] – Режим доступу до ресурсу: https://mathworld.wolfram.com/ElementaryCellularAutomaton.html.

Ключові слова: криптографія, клітинні автомати, шифрування, елементарні клітинні автомати

Ключевые слова: криптография, клеточные автоматы, шифрование, элементарные клеточные автоматы

Keywords: cryptography, cellular automata, encryption, elementary cellular automata.

Науковий керівник: *к.т.н. доцент Бойко В. Д.*

Васильєв Богдан Георгійович
Національний університет «Одеська юридична академія»
студент 4-го курсу факультету кібербезпеки
та інформаційних технологій

SQL-ІН'ЄКЦІЯ ТА ЇЇ НЕБЕЗПЕКА

SQL-ін'єкція-це хакерська техніка, яка була виявлена більше п'ятнадцяти років тому і до сих пір доводить свою руйнівну ефективність сьогодні, залишаючись головним пріоритетом безпеки баз даних.

Є багато негативних наслідків SQL-ін'єкції, наприклад, 21 лютого 2014 року на форумі Організації Об'єднаних Націй з управління інтернетом стався витік даних 3215 облікових записів. У серпні 2014 року в Мілуокі компанія комп'ютерної безпеки Hold Security повідомила, що виявила крадіжку

конфіденційної інформації майже з 420 000 веб-сайтів за допомогою SQL-ін'єкцій. У жовтні 2015 року атака з використанням SQL-ін'єкції була використана для крадіжки особистих даних 156 959 клієнтів з серверів британської телекомунікаційної компанії TalkTalk, використовуючи її вразливість в застарілому веб-порталі [1].

Також вона була використана напередодні президентських виборів 2016 року в США для компрометації особистих даних 200 000 виборців штату Іллінойс, а також в гучних атаках проти таких організацій, як Sony Pictures, PBS, Microsoft, Yahoo, Heartland Payment Systems і навіть ЦРУ [2].

SQL-ін'єкції є одним з найбільш часто використовуваних векторів веб-атак, які використовуються з метою отримання конфіденційних даних з організацій. Вкрадені кредитні картки або списки паролів, часто відбуваються через вразливості SQL-ін'єкцій.

SQL-ін'єкція – це метод, який зловмисники застосовують для вставки SQL-запиту в поля введення, а потім обробляють в базовій базі даних SQL. Ці недоліки потім можуть бути використані, коли форми введення дозволяють згенерувавши користувачем SQL-операторам безпосередньо запитувати базу даних.

Для прикладу беремо типовий сценарій, візьмемо типову форму входу, що складається з поля користувача / електронної пошти і поля пароля. Після відправки реєстраційної інформації вона об'єднується з SQL-запитом на веб-сервері користувача. У PHP команді пишеться наступним чином (див. рис. 1):

```
<?PHP
 $query = " SELECT * FROM users WHERE username = '". $_POST ['имя
пользователя']. "'";
 $запрос .= "И пароль = '". $_POST ['пароль']. "'";

 ?>
```

Рис. 1.

Так, він відправляється на сервер, щоб перевірити, чи було йому дано дійсне ім'я користувача з відповідним паролем. Ім'я користувача «Джеймс» з паролем «1111» призведе до цієї команди (див. рис. 2):

```
Выберите * из пользователей, где имя пользователя= 'Джеймс' и
пароль= '1111'
```

Рис. 2.

Але якби поставили щось на кшталт «james '; -», запит виглядав би так (див. рис. 3):

```
Выберите * из пользователей, Где username= 'james'; -- 'и password=
'1111'
```

Рис. 3.

В цьому випадку зловмисник використовує синтаксис коментарів SQL. Що залишився код після послідовності подвійного тире (–) не виконуватиметься. Це означає, що SQL буде (див.рис. 4):

```
Выберите * из пользователей, Где username= 'james';
```

Рис. 4.

Потім він поверне призначені для користувача дані, введені в поле пароля. Цей крок може дозволити обійти екран входу в систему. Зловмисник також може піти далі, додавши ще одну умову вибору, 'або 1 = 1', яке призведе до наступного запиту (див. рис. 5):

```
Выберите * из пользователей, Где username= 'james' или 1=1;
```

Рис. 5.

Запит повертає непорожній набір даних для будь-якого потенційного входу в систему з усією базою даних таблиць «користувачі».

Вищенаведений злом показав значний недолік безпеки для будь-якого сайту, але це лише невеликий приклад того, що він може зробити. Більш просунуті зломи дозволять зловмисникові запускати довільні оператори, завдаючи набагато більший збиток. Так, це може призвести до:

– витоку особистих даних, таких як кредитні картки, паспорти та інш;

– перерахування аутентифікаційних даних користувача, що дозволяють використовувати ці логіни на інших сайтах;

– пошкодження бази даних, виконання команд ОС, віддалені / вставлені дані і зруйновані операції для всього веб-сайту.

Це не повний перелік негативних наслідків.

Успішна атака на базу даних, яка управляє веб-сайтом або веб-додатком, така як атака обходу входу в систему SQL-ін'єкції, потенційно може дати хакеру широкий спектр повноважень, від зміни вмісту веб-сайту («псування») до захоплення конфіденційної інформації, такий як облікові дані облікового запису або внутрішні бізнес-дані. Список команд SQL-ін'єкції по суті такий же, як і список команд бази даних, включаючи потенційно катастрофічні, такі як DROP TABLE.

Отже, зловмисники можуть використовувати уразливості SQL-ін'єкцій для обходу заходів безпеки додатків. Вони можуть обійти перевірку автентичності та авторизацію веб-сторінки або веб-додатки і отримати вміст всієї бази даних SQL. Вони також можуть використовувати SQL-ін'єкцію для додавання, зміни і видалення записів в базі даних.

Уразливість SQL-ін'єкції може вплинути на будь-який веб-сайт або веб-додаток, що використовує базу даних SQL, таку як MySQL, Oracle, SQL Server або інші. Злочинці можуть використовувати його для отримання несанкціонованого доступу до ваших конфіденційних даних: інформації про клієнтів, особистих даних, комерційної таємниці, інтелектуальної власності та ін. Атаки SQL-ін'єкцій є однією з найстаріших, поширених і небезпечних вразливостей веб-додатків.

Але, насправді є багато речей, які власники веб-сайтів можуть зробити, щоб запобігти ін'єкціям SQL. Хоча в мережевої безпеці немає такого поняття, як надійне рішення, але на шляху спроб впровадження SQL можуть бути поставлені величезні перешкоди.

Список використаних джерел:

1. SQL injection [Електронний ресурс] // Wikipediaen – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/SQL_injection.

2. Paul Rubens. How to Prevent SQL Injection Attacks [Електронний ресурс] / Paul Rubens // eSecurityPlanet. – 2018. – Режим доступу до ресурсу: https://www.esecurityplanet.com/threats/how-to-prevent-sql-injection-attacks.html.

Ключові слова: SQL-ін'єкція, веб-атака, база даних, конфіденційних даних, зловмисники.

Ключевые слова: SQL-инъекция, веб-атака, база данных, конфиденциальных данных, злоумышленники.

Keywords: SQL injection, web attack, database, sensitive data, attackers.

***Науковий керівник**: к.т.н., доцент Кухаренко С. В.*