Bezpieczeństwo Aplikacji Internetowych - laboratorium

Lab 1: REST API + IDOR

1. Zadania do wykonania:

- 1. Stwórz aplikację REST API z operacjami CRUD dla produktów przypisanych do użytkowników.
- 2. Zastosuj generowanie ID produktom w sposób iteracyjny.
- 3. Zaloguj się jako jeden użytkownik, dodaj kilka produktów.
- 4. Następnie zaloguj się jako inny użytkownik i spróbuj pobrać produkt po ID (np. /products/1).
- 5. Zbadaj podatność IDOR przy użyciu Burp Suite.

Lab 2: OTP – metoda Lamporta

1. Zadania do wykonania:

- 1. Zarejestruj użytkownika z hasłem.
- 2. System przechowuje wielokrotnie hashowane hasło ($H^n(P)$).
- 3. Podczas logowania przesyłany jest OTP = $H^{n-1}(P)$, serwer porównuje z poprzednim hashem.
- 4. Udane logowanie aktualizuje stan na nowy hash, zmniejszając licznik.
- 5. Po N użyciach hasła użytkownik musi się zarejestrować ponownie.

Lab 3: Blind-based SQL Injection

1. Zadania do wykonania:

- 1. Przygotować aplikacje z podatnością aby zwracała np liczbę postów wg słowa kluczowego
- 2. Przeprowadź atak typu Blind SQL Injection
- 3. Wydobądź nazwy i hasła użytkowników
- 4. Sporządź raport ze zrzutami ekranu i opisem metodologii.

— skrypt do stworzenia

```
CREATE TABLE blog (
   id INT AUTO_INCREMENT PRIMARY KEY,
   title VARCHAR(255),
   post_content TEXT,
   published TINYINT(1)
);


INSERT INTO blog (title, post_content, published) VALUES
('Pierwszy post', 'To jest pierwszy post na blogu.', 1),
('Drugi post', 'Drugi post zawiera słowo SQL Injection.', 1),
('Nieopublikowany post', 'Ten post nie jest opublikowany.', 0),
('Inny post', 'Wyszukiwanie treści działa w tym przykładzie.', 1);


-- Tworzenie tabeli users
CREATE TABLE s3cret_u5ers (
```

```
    id INT AUTO_INCREMENT PRIMARY KEY,
    l0g1n VARCHAR(50) NOT NULL UNIQUE,
    pa44w0rd_md5 CHAR(32) NOT NULL,
    creation_date DATETIME DEFAULT CURRENT_TIMESTAMP,
    email VARCHAR(100) NOT NULL UNIQUE,
    full_name VARCHAR(100) NOT NULL
);

-- Wstawianie 30 użytkowników z losowymi hasłami i polskimi imionami i nazwiskami
INSERT INTO s3cret_u5ers (l0g1n, pa44w0rd_md5, email, full_name)
VALUES
  ('user01', MD5('A1b2C3d4'), 'user01@example.com', 'Jan Kowalski'),
  ('user02', MD5('E5f6G7h8'), 'user02@example.com', 'Anna Nowak'),
  ('user03', MD5('I9j0K1l2'), 'user03@example.com', 'Krzysztof Wiśniewski'),
  ('user04', MD5('M3n4O5p6'), 'user04@example.com', 'Maria Dąbrowska'),
  ('user05', MD5('Q7r8S9t0'), 'user05@example.com', 'Piotr Zieliński'),
  ('user06', MD5('U1v2W3x4'), 'user06@example.com', 'Katarzyna Wójcik'),
  ('user07', MD5('Y5z6A7b8'), 'user07@example.com', 'Paweł Mazur'),
  ('user08', MD5('C9d0E1f2'), 'user08@example.com', 'Zofia Kwiatkowska'),
  ('user09', MD5('G3h4I5j6'), 'user09@example.com', 'Tomasz Woźniak'),
  ('user10', MD5('K7l8M9n0'), 'user10@example.com', 'Agnieszka Lewandowska'),
  ('user11', MD5('O1p2Q3r4'), 'user11@example.com', 'Grzegorz Szymański'),
  ('user12', MD5('S5t6U7v8'), 'user12@example.com', 'Magdalena Zielińska'),
  ('user13', MD5('W9x0Y1z2'), 'user13@example.com', 'Jakub Kamiński'),
  ('user14', MD5('A3b4C5d6'), 'user14@example.com', 'Joanna Wiśniewska'),
  ('user15', MD5('E7f8G9h0'), 'user15@example.com', 'Michał Kowalczyk'),
  ('user16', MD5('I1j2K3l4'), 'user16@example.com', 'Monika Adamczyk'),
```

('user17', MD5('M5n6O7p8'), 'user17@example.com', 'Rafał Jabłoński'),

('user18', MD5('Q9r0S1t2'), 'user18@example.com', 'Ewa Pawlak'),

('user19', MD5('U3v4W5x6'), 'user19@example.com', 'Bartłomiej Piotrowski'),

('user20', MD5('Y7z8A9b0'), 'user20@example.com', 'Karolina Domańska'),

('user21', MD5('C1d2E3f4'), 'user21@example.com', 'Dawid Kaczmarek'),

('user22', MD5('G5h6I7j8'), 'user22@example.com', 'Sylwia Borkowska'),

('user23', MD5('K9l0M1n2'), 'user23@example.com', 'Aleksander Czarnecki'),

('user24', MD5('O3p4Q5r6'), 'user24@example.com', 'Izabela Kalinowska'),

('user25', MD5('S7t8U9v0'), 'user25@example.com', 'Maciej Jasiński'),

('user26', MD5('W1x2Y3z4'), 'user26@example.com', 'Gabriela Kołodziej'),

('user27', MD5('A5b6C7d8'), 'user27@example.com', 'Sebastian Majewski'),

('user28', MD5('E9f0G1h2'), 'user28@example.com', 'Julia Król'),

('user29', MD5('I3j4K5l6'), 'user29@example.com', 'Tadeusz Malinowski'),

('user30', MD5('M7n8O9p0'), 'user30@example.com', 'Natalia Baranowska');

Lab 4: JWT – tokeny jednorazowe

1. Zadania do wykonania:

- 1. Zaloguj się jako użytkownik i otrzymaj token JWT.
- 2. Użyj tokena do uzyskania dostępu do zasobu chronionego.
- 3. Przy ponownym użyciu tego samego tokena serwer odrzuca dostęp.
- 4. Zbadaj możliwość ataku powtórzeniowego i zabezpieczenia.