

SCP 사이트 취약점 보안

210923 이유경

SCP 1. 사이트 소개 <https://www.teamscp.kro.kr/>



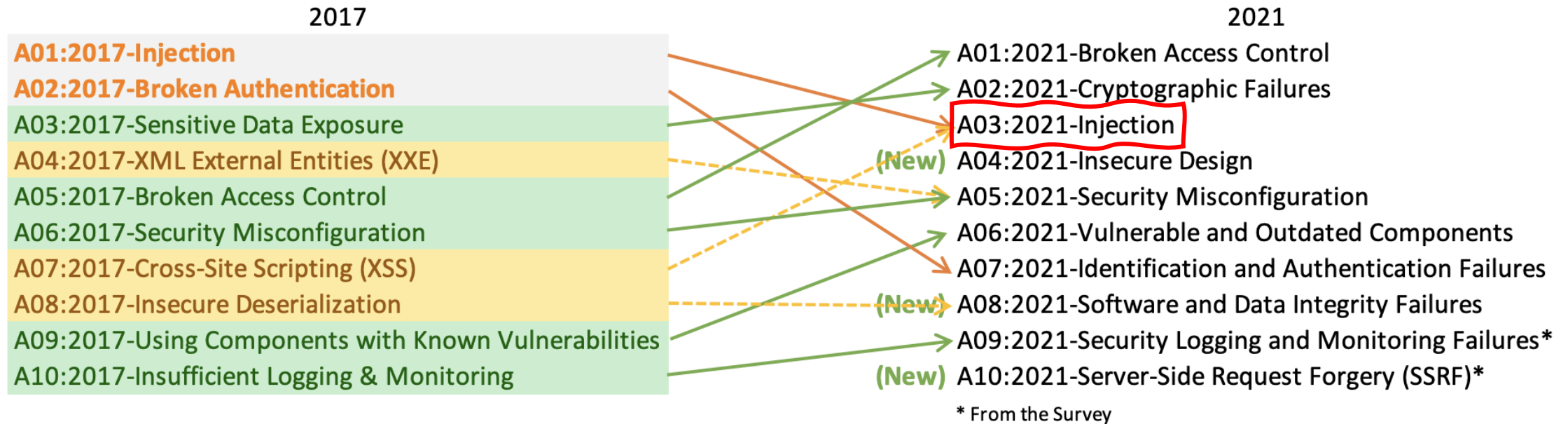
실질적으로 취약점을 찾을 수 있는 곳은 '커뮤니티' 페이지





2. 취약점 점검 목록 - OWASP TOP 10

3

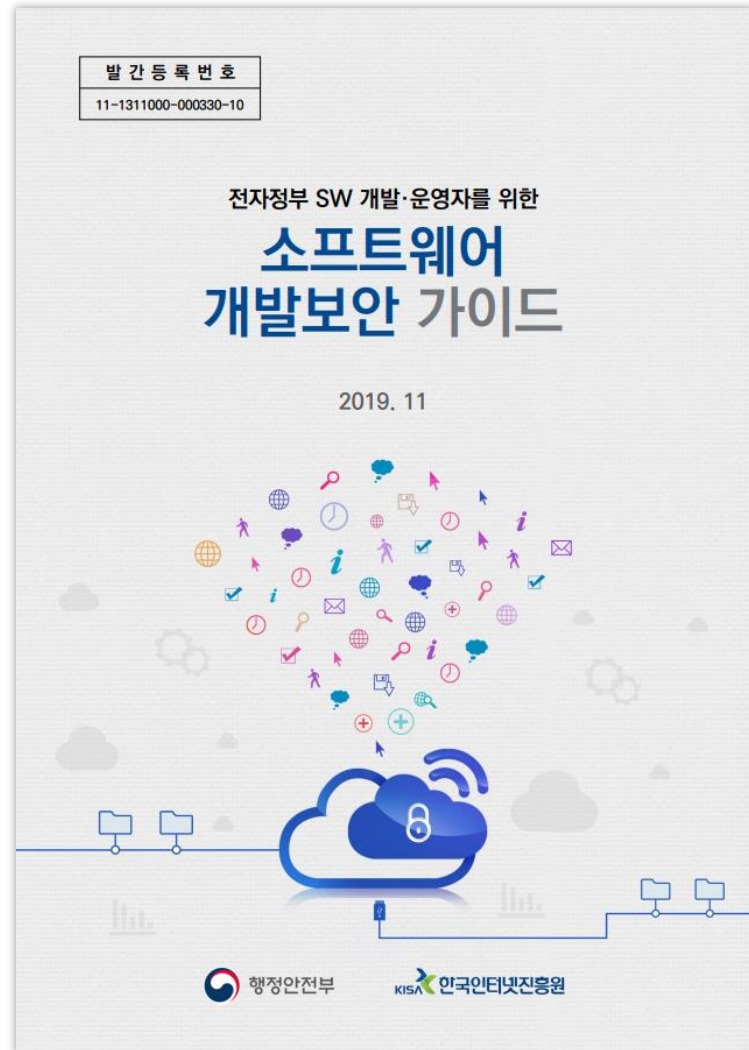


CWE-79: Cross-site Scripting

CWE-89: SQL Injection

CWE-73: External Control of File Name or Path

2. 취약점 점검 목록 - KISA 소프트웨어 개발보안 가이드



PART 제4장

구현단계 시큐어코딩 가이드

제1절 입력데이터 검증 및 표현

1. SQL 삽입
2. 경로 조작 및 자원 삽입
3. 크로스사이트 스크립트
4. 운영체제 명령어 삽입
5. 위험한 형식 파일 업로드
6. 신뢰되지 않는 URL 주소로 자동접속 연결
7. XQuery 삽입
8. XPath 삽입
9. LDAP 삽입
10. 크로스사이트 요청 위조
11. HTTP 응답분할
12. 정수형 오버플로우
13. 보안기능 결정에 사용되는 부적절한 입력값
14. 메모리 버퍼 오버플로우
15. 포맷 스트링 삽입

제2절 보안기능

1. 적절한 인증 없는 중요기능 허용 214
2. 부적절한 인가 218
3. 중요한 자원에 대한 잘못된 권한 설정 222
4. 취약한 암호화 알고리즘 사용 226
5. 중요정보 평문저장 231
6. 중요정보 평문전송 235
7. 하드코딩된 비밀번호 241
8. 충분하지 않은 키 길이 사용 245
9. 적절하지 않은 난수값 사용 249
10. 하드코딩된 암호화 키 253
11. 취약한 비밀번호 허용 258
12. 사용자 하드디스크에 저장되는 쿠키를 통한 정보노출 262
13. 주석문 안에 포함된 시스템 주요정보 265
14. 솔트 없이 일방향 해쉬함수 사용 268
15. 무결성 검사 없는 코드 다운로드 272
16. 반복된 인증시도 제한 기능 부재 278

제3절 시간 및 상태

1. 경쟁조건: 검사시점과 사용시점(TOCTOU) 283
2. 종료되지 않는 반복문 또는 재귀함수 290

제4절 에러처리

1. 오류 메시지를 통한 정보노출 292
2. 오류 상황 대응 부재 295
3. 부적절한 예외 처리 298

제5절 코드오류

1. Null Pointer 역참조 301
2. 부적절한 자원 해제 306
3. 해제된 자원 사용 311
4. 초기화되지 않은 변수 사용 314

제6절 캡슐화

1. 잘못된 세션에 의한 데이터 정보노출 316
2. 제거되지 않고 남은 디버그 코드 321
3. 시스템 데이터 정보노출 324
4. Public 메소드부터 반환된 Private 배열 327



SQL Injection

XSS

File Upload

에러 처리

관리자 페이지 노출

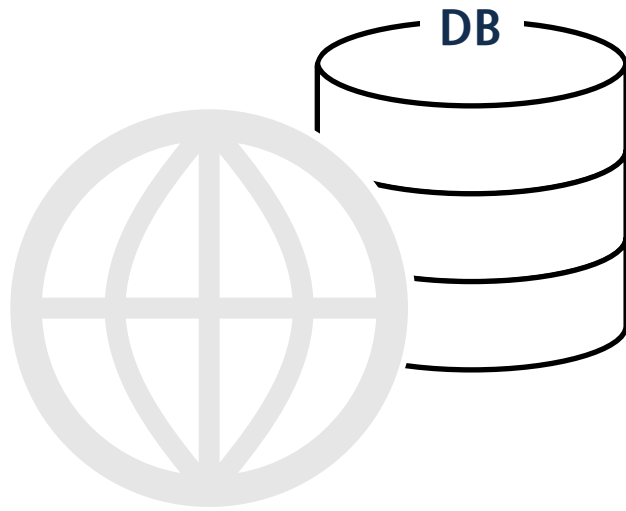
데이터 암호화

Directory Indexing

...

SQL Injection 이란?

웹 어플리케이션의 뒷단에 있는 DB에 질의(쿼리)하는 과정 사이에 일반적인 값 외에 내용을 변조하는 **악의적인 의도를 갖는 구문을 함께 삽입**하여 공격자가 원하는 SQL 쿼리문을 실행한다.



악의적인 의도의 구문

SQL Injection 보안 방법(PHP)

- PHP 자체 함수

`mysqli_real_escape_string()`

- 사용법

`mysqli_real_escape_string(connection, escapestring);`

MYSQL과 연결하는 connection과 escape형태로 만들어줄 string 입력

* escape string:

ex) *Tom's cat* 이라는 입력을 할 때, ' 는 SQL문에 사용되는 '와 중첩이 될 수 있음

이를 막기 위해 *Wn, Wr, W*” 처럼 문자열로 인식할 수 있게 사용하여 구별해주는 형태로 만들어주는 것

```
$conn = mysqli_connect($호스트명,$아이디,$비밀번호,$dbname);

$catagory = htmlspecialchars($_GET['catgo']);
$encatagory = mysqli_real_escape_string($conn, $catagory);

$search_con = htmlspecialchars($_GET['search']);
$search_con = mysqli_real_escape_string($conn, $search_con);

$conn->query("select * from today where $encatagory like '%$search_con%' order by no desc");
```

SQL Injection 보안 방법(PHP)

- PHP 자체 함수 **미사용**

'test' 검색결과

제목 검색




test
2021-09-23

test

" union select 1,table_name,3,table_schema,5 from information_schema.tables where table_schema='board' #" 검색결과

제목 검색



test
2021-09-23

manager board

notice board

noticeupload board

today board

todayupload board

' union select 1,table_name,3,table_schema,5 from information_schema.tables wheretable_schema='board' #

SQL Injection 보안 방법(PHP)

- PHP 자체 함수 **사용**

'test' 검색결과

제목 ▾ 검색



test
2021-09-23

test

" union select 1,table_name,3,table_schema,5 from information_schema.tables where table_schema='board' #" 검색결과

제목 ▾ 검색

' union select 1,table_name,3,table_schema,5 from information_schema.tables wheretable_schema='board' #



XSS 란?

웹 사이트 관리자가 아닌 사용자가 웹 페이지에 **악성 스크립트를 삽입**함으로써 개발자가 고려하지 않은 기능이 작동하여 발생하는 취약점이다.

- Dom based XSS
서버로 전달되지 않고, 사용자 측에서 브라우저를 통해 발생
- Stored XSS
저장형으로, 웹서버에 스크립트를 저장하여 데이터베이스에 기록
- Reflected XSS
반사형으로, 브라우저에 응답할 때 파라미터에 삽입된 스크립트를 사용자에게 전달

XSS 보안 방법(PHP)

- PHP 자체 함수
`htmlspecialchars()`
- XSS는 입력한 문자가 html 코드로 인식되어 발생한다.
- html 코드로 인식되는 특수문자를 html 엔티티로 변환한다.
html 엔티티: html은 미리 예약된 몇몇 문자가 존재하는데, 이를 기존에 사용하던 의미 그대로 사용하기 위해 별도로 만든 문자셋이다.

```
$conn = mysqli_connect($호스트명,$아이디,$비밀번호,$dbname);  
  
$catagory = htmlspecialchars($_GET['catgo']);  
$search_con = htmlspecialchars($_GET['search']);  
  
$conn->query("select * from today where $encatagory like '%$search_con%' order by no desc");
```



XSS 보안 방법(php)

- PHP 자체 함수 **미사용**

" 검색결과

제목 ▾ <script>alert(1)</script>

검색

www.teamscp.kro.kr 내용:

1

확인



XSS 보안 방법(PHP)

- PHP 자체 함수 **사용**

'<script>alert(1)</script>' 검색결과

제목 ▾

검색



File Upload 취약점이란?

파일 업로드 기능이 존재하는 웹 사이트의 **확장자 필터링이 미흡할 경우**, 공격자가 악성 파일을 업로드하여 시스템을 장악할 수 있는 취약점이다.

악성 스크립트가 업로드 된 후, 서버 상에서 스크립트를 실행하여 **웹을 획득**하는 등의 행위로 시스템 권한을 획득하거나 서버를 변조시키는 등의 방법으로 웹 서버를 장악한다.

웹 쉘: 웹 사이트를 통해 쉘을 여는 공격으로, 이 쉘을 통해 서버에 명령어를 수행하는 파일 업로드 취약점으로 많이 사용됨



File Upload 취약점 보안 방법

- 파일 확장자를 필터링한다.
- Whitelist를 통해 허용하지않은 부분을 필터링하여, 안전한 확장자가 아닐 시, 업로드에 제한을 둔다.
- Whitelist VS Blacklist

Whitelist: 자동 승인 항목

Blacklist: 자동 거부 항목

둘 중 보안에 더 좋은 기법은 Whitelist 다.

Blacklist의 경우, 지정해둔 항목 이외의 새로운 악의적인 파일 업로드가 가능한 확장자가 생길 수 있기 때문이다.

```
$allowed_ext = array('', 'jpg', 'jpeg', 'png', 'gif');
```

File Upload 취약점 보안 방법

- 파일 확장자 필터링 **미사용**

오늘의 활동

test 1

B I | | | | | | | ?

test 1

파일 선택 testfile.png

취소

완료

오늘의 활동

제목 ▾

검색



test 1
2021-09-23

File Upload 취약점 보안 방법

- 파일 확장자 필터링 사용 (png파일을 whitelist에 올리지 않은 상황)

오늘의 활동

test 2

B I | | | : | | | | ?

test 2

파일 선택 testfile.png

취소 완료

www.teamscp.kro.kr 내용:
허용되지 않는 양식입니다.

확인

QnA