동국대학교
dongguk university

제출일	2023.06.06	학과	컴퓨터공학전공
과목	컴퓨터보안	학번	2018112007
담당교수	김영부 교수님	이름	이승현

1) 실습 환경

(1)
운영 체제: Microsoft Windows 11 Home 64bit
프로세서 : Intel(R) Core(TM) i7-10510U @ 1.80GHz (8 CPUs), ~ 2.3GHz
메모리 : DDR4 16GB 2,667MHz
그래픽 카드 : Intel UHD Graphics

(2)
운영 체제: Microsoft Windows 10 Home 64bit
프로세서 : Intel(R) Core(TM) i7-7700HQ @ 2.80GHz (8 CPUs), ~ 2.8GHz
메모리 : DDR4 8GB 2,133MHz
그래픽 카드 : Intel HD Graphics 630, NVIDIA GeForce GTX 1050

2) 실습 진행

1. 문제 분석

i. SQL 삽입

- SQL 삽입 보안 약점은 입력된 데이터에 대한 유효성 검사를 하지 않을 경우, 공격자가 입력 폼 및 URL 입력창에 SQL 문장을 삽입하여 DB로부터 정보를 열람하거나 조작할 수 있는 보안 약점이다.
- 해당 보안 약점이 내재한 웹 응용프로그램에서는 사용자로부터 입력된 값을 검증 없이 넘겨받아 SQL 문장을 생성하기 때문에 개발자 강의도 하지 않은 조작된 SQL 문장이 실행되어 정보 유출의 위험성이 있다.
- 공격자는 SQL 삽입을 악용하여 데이터베이스 중요 데이터 읽기 및 쓰기, 데이터베이스 관리 작업 실행, 데이터베이스 종료 및 감시, DBMS 테이블 및 로그 삭제, DBMS 사용자 추가, DBMS 파일의 내용 복구가 가능하다.
- SQL 삽입은 공격자에게 신분 위조, 기본 데이터 변조, 시스템의 모든 데이터 외부 유출, 데이터 삭제 및 사용 불가, 데이터베이스 서버 관리자 권한 취득을 가능하게 한다.

ii. 크로스 사이트 스크립트

- 크로스 사이트 스크립트(XSS)는 검증되지 않은 입력값으로 인해 희생자의 웹 브라우저에서 의도하지 않은 악성 스크립트가 실행되는 보안 약점이다.
- 공격자는 크로스 사이트 스크립트 보안 약점을 통해 사용자의 개인정보 및 쿠키정보 탈취, 악성코드 감염, 웹 페이지 변조 등을 진행할 수 있고 희생자에게 비정상적인 기능을 수행하게 할 수 있다.
- 크로스 사이트 보안 약점은 공격 유형에 따라 Reflected XSS, Stored XSS, DOM 기반 XSS 세 가지 유형으로 분류할 수 있다.
- 공격자는 크로스 사이트 스크립트 보안 약점을 악용하여 쿠키 정보 및 세션 획득, 시스템 관리자 권한 획득, 기밀 정보 도용, 거짓 요청 생성, 인증 정보 수집을 위한 거짓 필드 생성, 강제 다운로드, 자격 증명 수집을 수행할 수 있다.
- 크로스 사이트 스크립트 보안 약점은 공격자에게 피싱 사이트 리다이렉션, 악성코드 실행 및 다운로드, 부적절한 콘텐츠 삽입, 크로스 사이트 요청 위조 보호 우회를 가능하게 한다.

iii. 경로 순회

- 검증되지 않은 외부 입력값을 통해 파일 및 서버 등 시스템 자원에 대한 접근 혹은 식별을 허용하는 경우, 입력값 조작으로 시스템이 보호하는 자원에 임의로 접근할 수 있는 보안 약점이다.
- 공격자는 경로 조작 및 자원삽입을 통해 허용되지 않은 권한을 획득하고 설정에 관계된 파일을 변경 및 실행시킬 수 있으며 서비스 장애 등을 유발할 수 있음
- 공격자는 경로 순회를 악용하여 자원의 수정 및 삭제, 시스템 정보 노출, 시스템 자원 간 충돌을 수행할 수 있다.

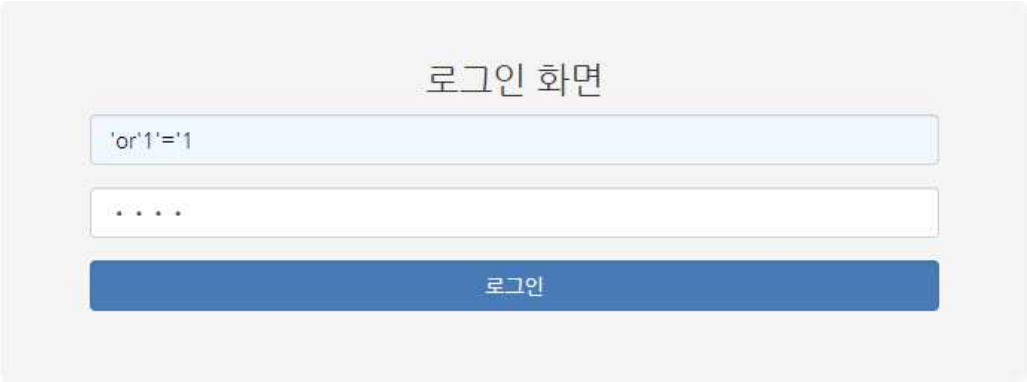
- 경로 순회는 공격자에게 임의 파일 생성, 파일경로 조작, 의도적 파일 삭제, 서비스 장애 유발, 허용되지 않은 권한 획득, 중요 파일 다운로드, 파일 변경 및 실행을 가능하게 한다.

iv. 역 직렬화
- 역 직렬화(Deserialization)는 반대 연산으로 바이너리 파일(Binary file)이나 바이트 스트림(Byte stream)으로부터 객체 구조로 복원할 수 있다
- 송신자가 네트워크를 이용하여 직렬화된 정보를 수신자에게 전달하는 과정에서 공격 자가 전송 또는 저장된 경향을 조작할 수 있는 경우에는 역 직렬화를 이용하여 악의적으로 데이터와 시스템을 침해할 수가 있는 보안 약점이다.
- 공격자는 역 직렬화를 악용하여 바이너리 파일 변조, 바이트 스트림 변조, 무결성 침해, 원격 코드 실행, 서비스 거부 공격을 수행할 수 있다.
- 역 직렬화는 공격자에게 바이너리 파일 객체 복원, 바이트 스트림 객체 복원을 가능하게 한다.

2. 실습
- 보물찾기 : 컴퓨터보안을 수강하는 학생들은 다음 “실습시나리오 예제 구성”을 참고하여 웹사이트에 존재하는 최대한 많은 보물을 찾으시오!!

보안 약점	실습시나리오 예제 구성
SQL 삽입	로그인 창, 주소창, 검색창
크로스 사이트 스크립트	Reflected XSS, Stored XSS, DOM based XSS
경로 순회	파일 다운로드, 파일 내용 출력, 파일 삭제
역 직렬화	데이터 위조, 원격 명령어 실행, 서비스 거부 공격

1. SQL 삽입: 로그인 창



- 아이디 창에 ‘or’1’=’1을, 패스워드 창에 1234를 입력한 모습이다.
- 이미지와 같이 입력하면 userID=’or’1’=’1&userPassword=1234가 전달된다.

- userID, userPassword가 true 되면서 로그인이 성공한다.
- 계정명이 ‘or’1’=’1이라고 출력된다.

2. SQL 삽입: 주소창

- 주소창을 통하여 로그인을 진행하려는 모습이다.
- 로그인 기능을 수행하는 loginAction.jsp에 userID=alice&userPassword=1234를 전달하여 로그인을 진행한다.

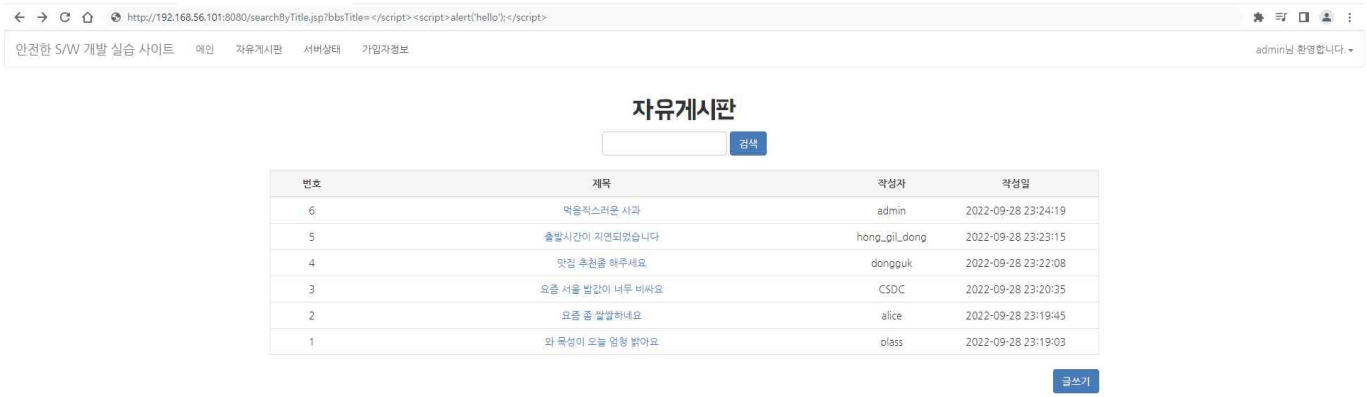
- 성공적으로 alice 계정에 로그인된 모습을 확인할 수 있다.

3. SQL 삽입: 검색창

- 자유게시판 검색창에 'UNION SELECT ALL 1,2,3,4,5,6,7#을 입력한 모습이다.
- 현재 자유게시판 글을 저장하는 데이터베이스의 칼럼 수를 구할 것이다.

- 검색 결과 UNION 연산이 진행된 모습을 볼 수 있다.
- 자유게시판 글을 저장한 데이터베이스의 칼럼 수와 검색창에 입력한 칼럼 수가 같기에 연산이 진행되었다.
- 자유게시판 글을 저장하는 데이터베이스의 칼럼 수는 7개임을 볼 수 있다.

4. 크로스 사이트 스크립트: Reflected XSS



- 검색 기능을 수행하는 파일에서 검색값을 전달하는 부분에 경고 메시지를 출력하는 스크립트를 집어넣는다.
- 검색 기능을 수행하는 searchByTitle.jsp의 경우 bbsTitle을 매개변수로 하여 전달한다.
- bbsTitle에 alert 함수가 포함된 스크립트를 포함하게 된다.

- alert 함수가 포함된 스크립트가 매개변수를 통해 전달되면서 경고 메시지가 출력된다.

5. 크로스 사이트 스크립트: Stored XSS

- 자유게시판의 글쓰기 기능에서 내용에 alert 함수를 포함하는 스크립트를 작성한다.
- 제목은 간단하게 test로 작성한다.

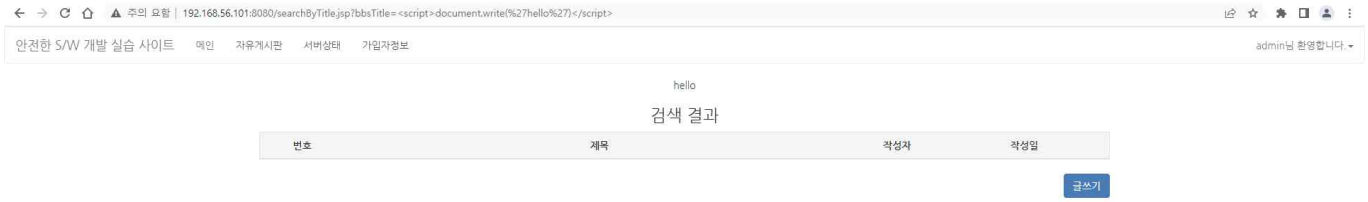
- 정상적으로 자유게시판에 앞서 생성한 글이 표시된다.
- 제목이 test인 글이 새로 생성된 글이다.

- 생성한 글을 클릭하면 경고 메시지가 출력된다.

- 경고 메시지가 출력된 후 글 정보가 출력된다.
- 내용 부분에서는 아무런 내용이 출력되지 않으나, 실제로는 스크립트가 저장되어 있다.

6. 크로스 사이트 스크립트: DOM based XSS

- 검색 기능을 수행하는 파일에서 검색값을 전달하는 변수에 값을 작성한다.
- 검색 기능을 수행하는 searchByTitle.jsp의 경우 bbsTitle을 매개변수로 하여 전달한다.
- bbsTitle 변수에 'hello'를 document.write() 함수를 이용하여 작성한다.
- Reflected XSS와 큰 차이가 없어 보이지만, DOM based XSS의 경우 jsp 파일을 수정한다는 점에 있어 큰 차이점을 보인다.

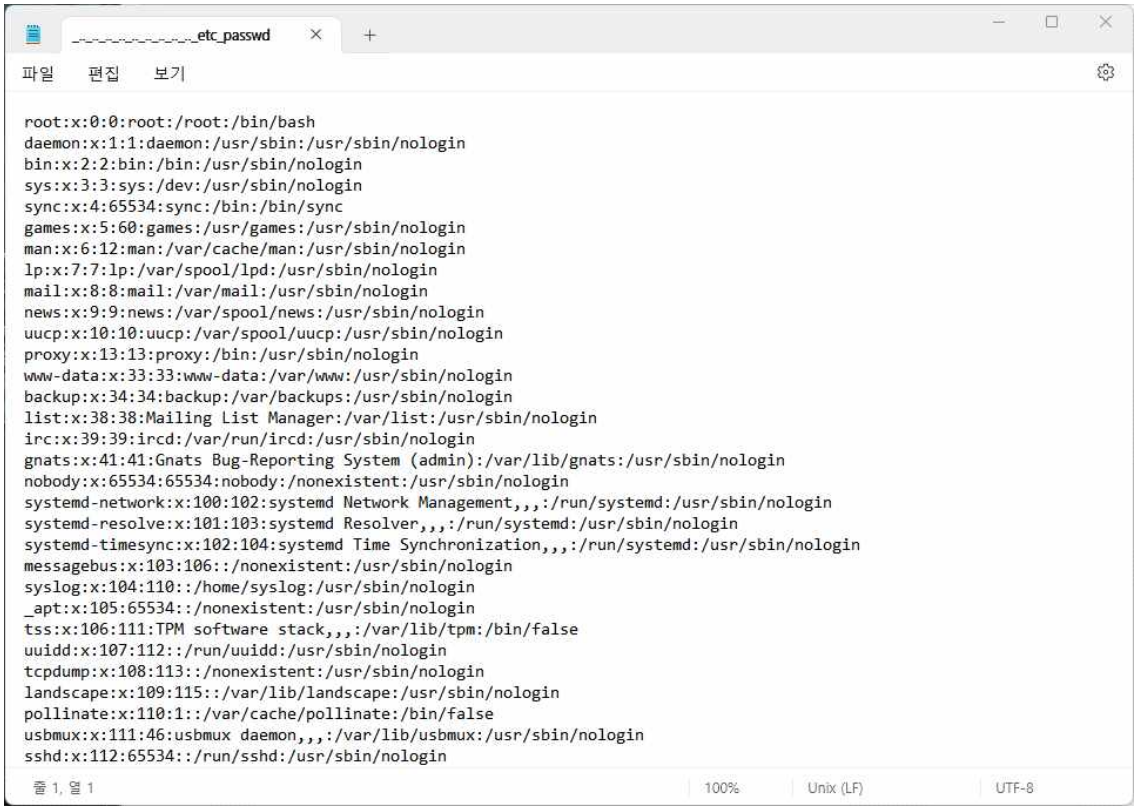


- hello가 검색된 모습을 확인할 수 있다.

7. 경로 순회: 파일 다운로드

- 다운로드 기능을 수행하는 파일의 매개변수로 글의 번호와 다운로드 받을 파일명을 전달한다.
- 현재 첨부된 파일을 클릭하면 매개변스로 현재 글의 번호인 6번과 apple.jpg가 전달된다.
- 이 파일명을 임의로 etc/passwd 파일을 지정한다.

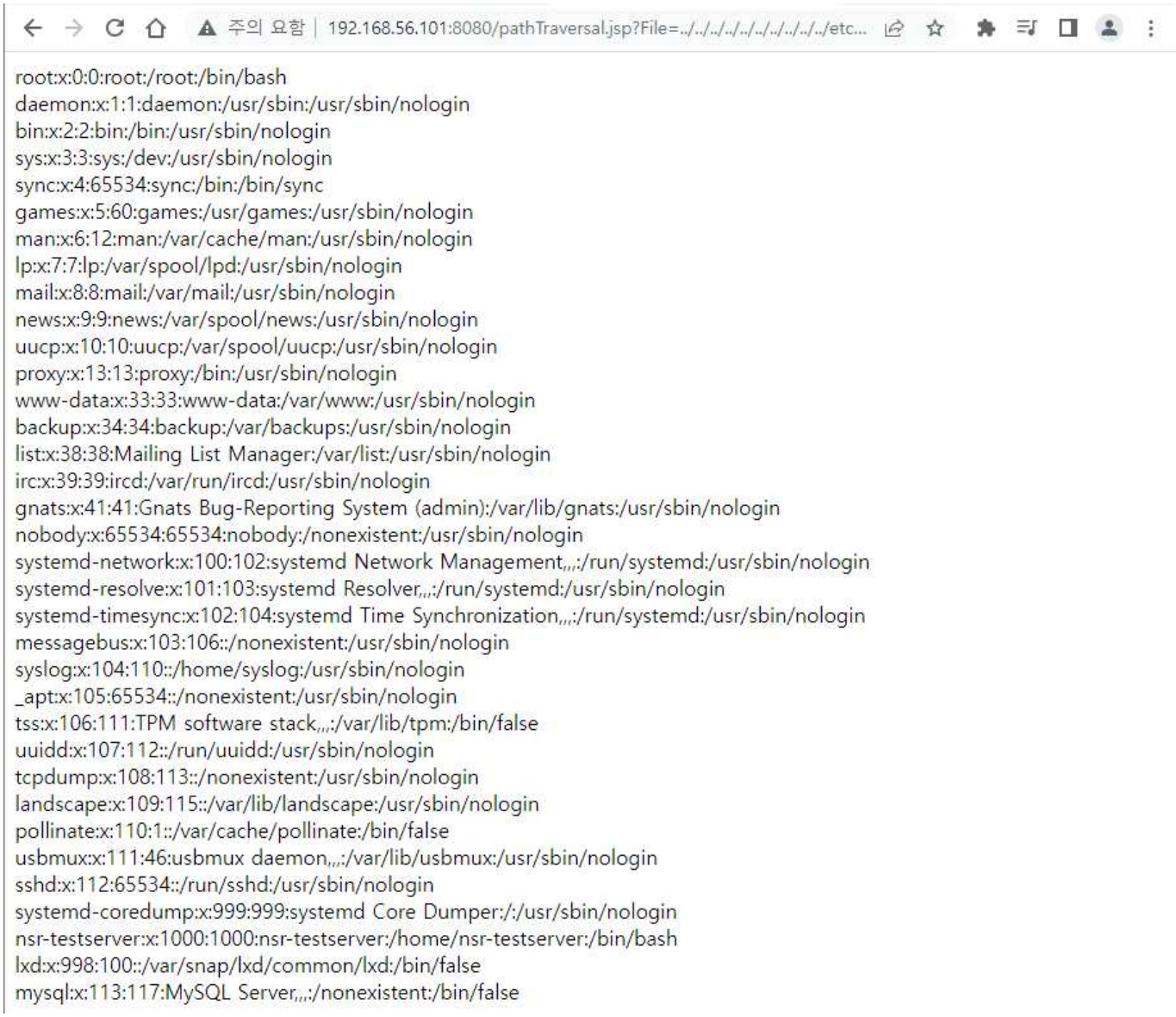
- etc/passwd 파일이 다운로드 된 모습이다.



- 다운로드 된 파일을 살펴보면 사용자 계정의 정보들이 출력된 모습을 확인할 수 있다.

8. 경로 순회: 파일 내용 출력

- 파일 내용을 출력해주는 파일의 매개변수로 파일명을 전달한다.
- 전달하는 파일명은 etc/passwd이다.



- 다음과 같이 사용자 계정의 정보가 출력된 모습을 확인할 수 있다.

9. 경로 순회: 파일 삭제

- 파일 삭제를 진행하는 파일의 매개변수로 게시판 글의 번호와 파일명을 전달한다.
- 현재 글의 번호인 6번과 파일명 apple.jpg를 전달한다.

- 지정한 글에서 첨부된 파일인 apple.jpg가 삭제된 모습을 확인할 수 있다.

10. 역 직렬화: 원격 명령어 실행

```
root@nsr-testserver:/home/nsr-testserver/share/nsr_content/cli_based# java ServerDeserialize
Server is up and running on port: 5000!
```

- 서버에서 ServerDeserialize 파일을 실행한다.
- 서버에서 클라이언트의 응답을 기다린다.

```
root@nsr-testserver:/home/nsr-testserver/share/nsr_content/cli_based# java ClientSerialize 192.168.5
6.101 ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500        inet 192.168.56.101  netmask 255
.255.255.0  broadcast 192.168.56.255        inet6 fe80::a00:27ff:fe51:2f36  prefixlen 64  scopeid 0x
20<link>        ether 08:00:27:51:2f:36  txqueuelen 1000  (Ethernet)        RX packets 2  bytes 1180
 (1.1 KB)        RX errors 0  dropped 0  overruns 0  frame 0        TX packets 11  bytes 1418 (1.4 K
B)        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0lo: flags=73<UP,LOOPBACK,RUNNING
>  mtu 65536        inet 127.0.0.1  netmask 255.0.0.0        inet6 ::1  prefixlen 128  scopeid 0x10<
host>        loop  txqueuelen 1000  (Local Loopback)        RX packets 223  bytes 16025 (16.0 KB)
     RX errors 0  dropped 0  overruns 0  frame 0        TX packets 223  bytes 16025 (16.0 KB)
 TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
root@nsr-testserver:/home/nsr-testserver/share/nsr_content/cli_based# _
```

- 클라이언트에서 ClientSerialize파일을 실행하는데 매개변수로 서버의 ip address와 서버에서 실행할 명령어를 전달한다.
- 서버의 주소는 1920168.56.101이다.
- 서버에서 실행할 명령어는 ifconfig로 네트워크 정보를 출력한다.
- 성공적으로 서버에서의 네트워크 정보가 출력된 모습을 확인할 수 있다.

3) 느낀 점

- 웹사이트의 보안을 위협하는 SQL 삽입, 크로스 사이트 스크립트, 경로 순회, 역 직렬화에 대한 실습을 이번 시간에 진행해보았다. 확실히 웹사이트에는 사용자 정보가 많이 저장되어 있기에 이러한 보안 위협에 대비해야 하겠다는 생각이 들었다. 물론 최근 웹사이트는 실습처럼 쉽게 보안이 뚫리지야 않겠지만 주의하는 것이

좋을 거라는 생각을 한다. 워낙 최근에 정보 유출 사고가 자주 발생하기에 이러한 위협에 주의해야 하는 것은 당연하다. 특히나 아무 생각 없이 아무런 웹사이트에 접속하는 사람이 많은데 이러한 사람들이 주의해야 할 것이다. 이 실습과 같이 악의적인 기능을 수행하는 스크립트를 집어넣어 정보를 빼내는 경우가 많을 것이다. 나도 어디선가 내 정보가 탈취되었는지 스팸 문자가 많이 오고, 보이스피싱도 당할 뻔했었는데 너무 부주의하게 인터넷을 해서 그런가 싶다. 나도 모르게 웹사이트에 접속하자마자 정보가 탈취되니 주의하더라도 결국 당할 수밖에 없는 거 같다. 아니면 인터넷을 아예 끊고 살아야 하는데 요즘에는 그런 것이 거의 불가능하기에 최대한 주의해서 사는 것밖에 방법이 없을 거 같다. 이번 실습을 통해 아무런 웹사이트에 접속하지 말고 주의해서 확인해봐야겠다는 경각심이 들었다. 내 정보가 다시는 탈취되지 않도록 노력해야겠다.