

PT 스터디 2일차

오늘의 순서



SQL Injection



공격 접근 방법과 예시



잡다한 정보



Kali Linux

DB와 SQL 기본 상식

Table이란?

- Table
 - DB에서 특정한 규칙으로 정리된 정보를 묶어 두는 체계
- 구성 요소
 - Column : 테이블이 담고 있는 한 종류의 정보 요소
 - Record : Column에서 정의한 정보를 담고 있는 하나의 정보

DB, SQL과 관련 있는 단어들

쿼리

- DB에 어떤 것을 처리해 달라고 요청을 보내는 행위

스키마(Schema)

- 테이블 등 DB의 구조

인덱스

- DB를 빨리 찾기 위한 정보를 저장한 테이블

SQL 기초 구문

SELECT

- DB에서 정보를 가져올 때 사용하는 구문

INSERT

- DB에 정보를 집어넣을 때 사용하는 구문

WHERE

- SQL 명령을 수행할 때 조건을 추가하는 구문

DELETE

- DB에 있는 정보를 지울 때 사용하는 구문

UPDATE

- DB에 있는 정보를 수정할 때 사용하는 구문

SQL - SELECT

- DB에서 정보를 가져올 때 사용하는 구문

- 예시

- SELECT password FROM user;

 ↙ ↖
 Column 이름 Table 이름

- SELECT * FROM user;
 - SELECT name, password FROM user;

SQL - WHERE

- SQL 명령을 수행할 때 조건을 추가하는 구문
- 예시
 - `SELECT password FROM user WHERE id = 3;`

 조건

SQL - WHERE

Operator	Description
=	Equal
<>	Not equal. Note: In some versions of SQL this operator may be written as !=
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
BETWEEN	Between an inclusive range
LIKE	Search for a pattern
IN	To specify multiple possible values for a column

https://www.w3schools.com/sql/sql_where.asp

SQL – WHERE (예시)

- `SELECT * FROM user WHERE name = "홍길동" AND id < 100;`
- ... `WHERE (name = "홍길동" AND address LIKE "서울%") or name="성춘향";`
→ 이름이 홍길동이고 주소가 서울로 시작하는 경우 또는 이름이 성춘향인 경우
- ... `WHERE name LIKE "%길동";`
→ 이름이 길동으로 끝나는 경우
- ... `WHERE NOT name LIKE "%길동%";`
→ 이름에 길동이 포함되지 않은 경우

SQL과 주석

- DB 종류에 따라 방법은 다르지만 SQL문에 주석을 넣을 수 있음
- 가능한 문자열
 - #
 - --
- 예시
 - `SELECT name, grade FROM student # 학생 테이블에서 이름과 학점을 가져온다`

SQL Injection

SQL Injection

정의

- 서버에 특이한 패턴의 SQL 명령을 보내서 해커가 원하는 일을 하는 기법

응용

- 개인정보 등 DB 탈취
- 서버 파괴
- 의도되지 않은 동작 수행

SQL Injection의 원리

```
SELECT * FROM user WHERE id = ( );
```




```
SELECT * FROM user WHERE id = (1 or 1=1);
```

왜 생기는가

- 필터를 통하지 않고 그냥 쿼리에 받은 문자를 집어넣기 때문

```
def entry(self, id):  
    '''  
    id를 주면 정보를 얻어옴. (제목, 사용자명, 태그, url, ) 튜플 반환.  
    '''  
    c = self.db.cursor()  
    c.execute(u"SELECT title, username, tag, url, text FROM entries WHERE id = %s" % (id, ) )
```



```
def entry(self, id):  
    '''  
    id를 주면 정보를 얻어옴. (제목, 사용자명, 태그, url, 본문) 튜플 반환.  
    '''  
    c = self.db.cursor()  
    c.execute(u"SELECT title, username, tag, url, text FROM entries WHERE id = ?", (id, ) )
```

Example 1 – 항상 참인 경우

- 다음 쿼리를 수행해서 값이 있으면 로그인 성공으로 한다
 - `SELECT id FROM user WHERE id = "()" AND pw = "()";`
- id에 로그인 아이디를 넣고 pw에 `1" or "1"="1` 을 넣는다
 - 원래 쿼리에 따옴표가 있어서 맞춰주어야 함
- ... `WHERE id = "1" AND pw = "1" or "1"="1";`
- 두 조건 중 하나만 만족해도 id가 반환된다
 - `id = "1" AND pw = "1" → id가 1이고 pw가 1인 경우`
 - `"1"="1" → 1과 1이 같은 경우 (항상 참)`

Example 2 – 항상 참인 경우

- 예시 쿼리
 - `SELECT * FROM user WHERE id = "()";`
- `SELECT * FROM WHERE id = "(" or "" = ")";`

SQL Injection이 가능할 지도 모르는 징후

- 파라미터 값이 DB에 있을 것 같다
 - 쿼리에 입력한 값이 그대로 들어갈 지도 모름
- 파라미터 값을 지우거나 이상하게 했더니 오류 발생
 - 단서를 얻을 수도 있다
- 게시판인데 좀 허술해 보인다
 - 파라미터를 잘 조작하면 UNION based SQL Injection이 가능할 수도 있다

SQL Injection – 에러 메시지

- Xxx 테이블이 존재하지 않습니다
- Column 개수가 맞지 않습니다
- SELECT 쿼리를 실행할 수 없습니다

SQL injection 공격의 종류

- Boolean-Based blind SQL injection
- Time-Based blind SQL injection
- Error-Based SQL injection
- UNION query-based SQL injection
- Stacked queries SQL injection

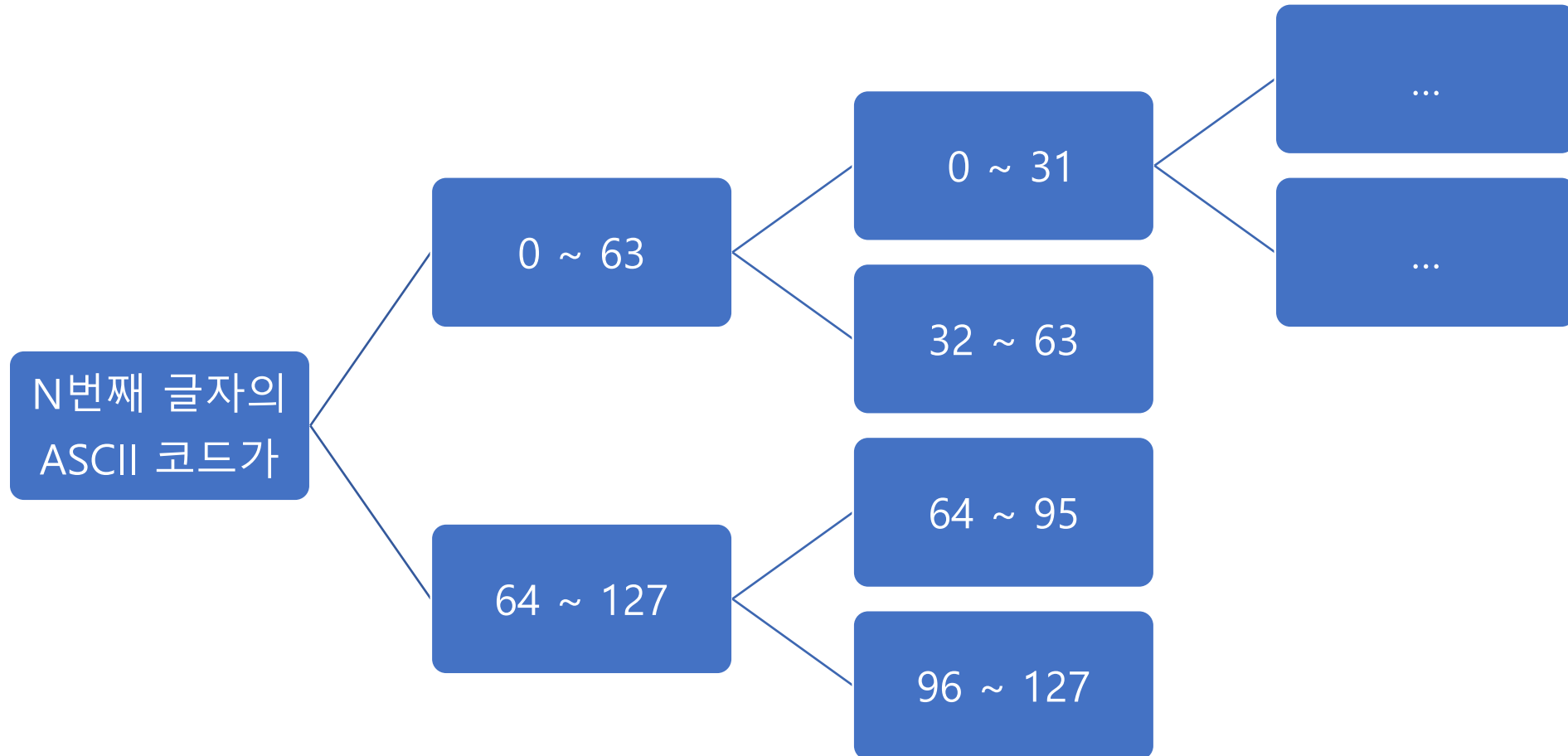
Blind SQL Injection

- 정의
 - 서버에 악의적 쿼리를 날려서 참인 경우와 거짓인 경우의 반응 차이를 이용해 정보를 빼내는 기법
- 이용 가능한 것들
 - 아이디 찾기 할 때 아이디 존재 여부 확인
 - 게시물을 검색할 때 검색 결과가 있음/없음 이용
 - 기타 등등 참, 거짓일 때 반응이 다른 경우 이용
- 방법
 - 얻고 싶은 값의 n번째 byte의 코드를 하나 하나씩 찾는다

SQL에 있는 함수

- substr(string, index, length)
 - 문자열을 자를 수 있는 함수
- ascii(char)
 - 문자를 아스키 코드로 바꿔 주는 함수

Blind SQL Injection 원리 - 이진탐색



Blind SQL Injection

- N번째 글자 비교하는 구문 작성
 - 예) `ascii(substr(<원하는 쿼리>, n, 1)) < 64`
 - 원하는 쿼리는 결과를 하나만 반환해야 함
- 기존 구문과 결합
 - 예) `select * FROM users WHERE id = '(a' and ascii(substr(<원하는 쿼리>, n, 1)) < 64#) and pw = ()`;
 - 앞에서 배운 기초 SQL 인젝션 구문 응용
 - 뒷부분 필요 없는 것은 주석처리 해서 버림
- 이 과정을 계속 반복해서 글자들을 찾아 나간다

웹해킹 Hard 문제 해설

- search 칸에 "a'"를 치면 서버 에러 발생
- "a' or '1'='1"을 검색해도 검색 결과가 나오는 것을 확인.
- a' and ascii('a') < 100 or '2'='1 및 변형 구문으로 취약점 확인
- "a' and ascii(substr((select secret from post where no = 1), 1, 1)) = 98 or '2'='1" 쿼리가 동작하는 것을 확인함
- 이진 탐색 기법을 이용하여 password를 찾기로 함.

UNION based SQL Injection

- UNION
 - 여러 쿼리의 결과값을 하나로 합치는 구문
 - Column 개수가 동일해야 한다
 - 예) `SELECT user, grade FROM student_01 UNION SELECT user, grade FROM student_02`

UNION based SQL Injection

정의

- UNION 구문을 이용해서 원래 나오는 값과 원하는 값을 합쳐서 출력하게 하는 공격 기법

가능한 시나리오

- 게시판 검색에 SQL Injection 취약점이 있어서 게시물 목록 뒤에다가 원하는 값을 덧붙여서 나오도록 할 수 있다

특징

- 원하는 문자열이 한꺼번에 나오기 때문에 속도가 빠르다
- Column 개수가 같아야 함 (SELECT 뒤에 아무 문자열이나 NULL 삽입)

UNION based SQL Injection - Example

- `SELECT title, name, date, username FROM board WHERE id < () AND id >= ();`
- Student table에 있는 이름과 성적을 털고 싶다면
 - ... `id < (0 UNION SELECT name, grade, NULL, '1' FROM student #) AND id >= (0);`
 - Column 개수를 맞추기 위해 임의의 문자열이나 NULL을 넣어 준다

Stacked queries SQL injection

- 정의
 - 세미클론(;)으로 구분된 여러 SQL 쿼리를 실행시키는 기법
- 예시
 - `SELECT * FROM user WHERE id = (1; <원하는 쿼리>;)`

SQL Injection 공격 시나리오

1. DB의 테이블 이름 탈취
 1. MYSQL : INFORMATION_SCHEMA.TABLES 의 TABLE_NAME column
2. 테이블의 column 이름 탈취
 1. MYSQL : INFORMATION_SCHEMA.COLUMNS 의 TABLE_NAME과 COLUMN_NAME 조합
3. 앞에서 배운 기법으로 쿼리 실행

본격적인 SQL Injection

- 앞에서 배운 것들을 조합하면 손으로도 할 수 있음
- 매우 귀찮기 때문에 툴을 사용
 - 스터디 후반부에 다시 설명
- 툴 잘못 쓰면 잡혀가고 동아리 존폐 위기
- 정 써 보고 싶으면 직접 서버 만들어서 쓰세요

추가 참고 자료

- https://www.owasp.org/index.php/SQL_Injection_Prevention_Cheat_Sheet
- 방화벽으로 막힌 경우 우회 방법 :
https://www.owasp.org/index.php/SQL_Injection_Bypassing_WAF

잡다한 정보

Robots.txt

- 정의
 - 웹 크롤러가 크롤링할 수 있는 것과 없는 것을 정의한 텍스트 파일
- 얻을 수 있는 정보
 - 관리자 페이지 경로
 - 파일 업로드 경로
 - 기타 등등...
- 예시
 - Disallow: /myadmin/ → 관리자 화면이 어디에 있는지 확인 가능
 - Disallow: /admin/login → 숨겨진 로그인 링크 확인 가능

구글을 통한 해킹


- Site:example.com
- Inurl:admin
- Intitle:관리자
- Intitle:"index of"
- Intitle:"로그인"
- "Apache/2.0.0"

구글을 통한 해킹

- 다른 해커들이 구글 해킹 쿼리문을 미리 작성해 두었음
- 점검 대상을 찾을 때 이런 쿼리들을 사용하면 유용
- <https://www.exploit-db.com/google-hacking-database/>
- <http://www.hackersforcharity.org/ghdb/>

너무 많이 쓰면..

☐ 로봇이 아닙니다.


reCAPTCHA
개인정보 보호 - 약관

제출

페이지 정보

Google의 시스템이 컴퓨터 네트워크에서 비정상적인 트래픽을 감지했습니다. 이 페이지는 로봇이 아니라 실제 사용자가 요청을 보내고 있는지를 확인하는 페이지입니다. [왜 이런 현상이 발생하는 거죠?](#)

WordPress란?

- 전세계에서 가장 많이 쓰이는 CMS
 - CMS : Content Management System
- 간단하게 웹 사이트, 블로그 같은 것을 만들 수 있음
- 수많은 사이트가 WordPress 기반으로 운영됨
- 예시 사이트 : 학교 교내의 수많은 사이트

WordPress 사용의 징후

- 소스 코드에 wp-content가 많이 들어 있으면 100% 워드프레스를 사용하는 것임

```
ss' href='http://hurel.hanyang.ac.kr/wp-content/plugins/LayerSlider/static/css/layerslider.css?ver=3.7.4' type='text/css'
:s-css' href='http://fonts.googleapis.com/css?family=Indie+Flower:regular%7C Oswald:300,regular,700&#038;subset=latin%2Clatin-ext' type='text/css'
' href='http://hurel.hanyang.ac.kr/wp-content/plugins/nextgen-gallery/products/photocrati_nextgen/assets/css' href='http://hurel.hanyang.ac.kr/wp-content/themes/Avada/style.css?ver=3.7.4' type='text/css'
des-css' href='http://hurel.hanyang.ac.kr/wp-content/themes/Avada/shortcodes.css?ver=3.7.4' type='text/css'
ss' href='http://hurel.hanyang.ac.kr/wp-content/themes/Avada/fonts/fontawesome/font-awesome.css?ver=3.7.4' type='text/css'
awesome-css' href='http://hurel.hanyang.ac.kr/wp-content/themes/Avada/fonts/fontawesome/font-awesome.css?ver=3.7.4' type='text/css'
ons-css' href='http://hurel.hanyang.ac.kr/wp-content/themes/Avada/css/animations.css?ver=3.7.4' type='text/css'
' href='http://hurel.hanyang.ac.kr/wp-content/themes/Avada/css/ie8.css?ver=3.7.4' type='text/css'
href='http://hurel.hanyang.ac.kr/wp-content/themes/Avada/css/ie.css?ver=3.7.4' type='text/css' me
ss' href='http://hurel.hanyang.ac.kr/wp-content/themes/Avada/css/media.css?ver=3.7.4' type='text/css'
numbarungothic-css' href='http://hurel.hanyang.ac.kr/wp-content/plugins/hangul-font-nanumbarungothic/skin-default-css' href='http://hurel.hanyang.ac.kr/wp-content/plugins/kboard-comments/skin/default-media-css' href='http://hurel.hanyang.ac.kr/wp-content/plugins/kboard/template/css/editor_media.css' href='http://hurel.hanyang.ac.kr/wp-content/plugins/kboard/font-awesome/css/font-awesome.min.css'
```

참고하면 좋은 링크들

- 웹 어플리케이션 보안 점검 체크리스트
 - https://www.owasp.org/index.php/Web_Application_Security_Testing_Cheat_Sheet
- OWASP Cheat Sheet Series
 - https://www.owasp.org/index.php/OWASP_Cheat_Sheet_Series
- WordPress 취약점 리스트
 - https://www.cvedetails.com/product/4096/Wordpress-Wordpress.html?vendor_id=2337

공격 접근 방법과 예시

방법 1 : 직접 사용해보기

- 개념
 - 해킹 대상을 직접 사용해 보면서 '이상한 것'을 찾는다
- 예시
 - 같이 일하는 사람을 추가했더니 리스트에 그 사람의 개인정보가 보임
 - 로그인한 상태가 아닌데 로그인된 상태
- 응용
 - 기능 수준의 접근 통제 누락
 - 보안 설정 오류

방법 2 : 다른 값 넣어 보기

- 개념
 - 파라미터 등에 다른 값을 한번 넣어본다
- 예시
 - 쿠키에 아이디 값이 있으면 바꿔보기
 - 정보를 얻어올 때 파라미터를 조작해서 다른 사람 정보를 가져오기
 - 로그인 아이디를 파라미터로 받아오면 조작하기
- 응용
 - 기능 수준의 접근 통제 누락

방법 3 : 악의적 값 넣어 보기

- 개념
 - 악의적인 값을 넣어서 다른 동작이 수행되는지 확인
- 응용
 - Injection (SQL Injection 등등)
 - XSS (Cross Site Scripting)
 - 파일 업로드 / 다운로드 취약점

방법 4 : 서버와 오가는 값 확인해보기

- 개념
 - 서버와 오가는 통신을 분석해서 어떤 정보가 오가는지 확인한다
- 예시
 - 서버와 통신을 할 때 비밀번호가 평문으로 전송되는 경우
 - SQL 구문 등 실제 실행될 것 같은 값이 전송되는 경우
 - 개인 고유번호 같은 것이 전송되는 경우
- 사용 가능한 툴
 - Proxy Tool
 - Firefox 개발자 도구

방법 5 : 툴 사용

- 개념
 - 툴을 사용해서 취약점 등을 자동으로 찾아낸다
- 예시
 - Scanner를 이용해서 취약점 리스트를 뽑아낸다
 - Fuzzer를 이용해서 이상한 동작을 하는지 확인한다
 - Proxy Tool을 이용해서 HTTP 통신을 변조한다
- 응용
 - 앞에서 언급한 방법을 해 보는 것에 도움을 준다
 - 웹해킹을 보조해 주는 일을 할 뿐 모든 것을 해 주지는 않는다

방법 6 : 웹 페이지 분석

- 개념
 - 웹 서버로부터 받은 HTML 문서, 자바스크립트 문서 등을 분석한다
- 예시
 - 숨겨진 메뉴가 주석처리만 되어 있다
 - 모든 정보를 다 받아오고 보면 안 되는 정보는 숨기기만 한다
 - 자바스크립트 함수를 개발자 도구로 실행한다

방법 7 : 소스 코드 분석

- 개념
 - 소스 코드를 분석해서 잘못된 부분을 찾아낸다
- 방법
 - 수작업으로 검토
 - 소스 코드 분석 도구 사용

기대하던 실제 사례!

- 사정상 PPT를 제공하지 않습니다



포털시스템

연구정보 매뉴얼

연구자용

2015. 03.



Kali Linux

Kali Linux

- 모의 해킹을 위한 도구가 완비된 리눅스 배포판
- 웹해킹, 리버싱, 시스템 해킹, 포렌식, 암호학 등등
- 가상머신에 깔아두면 귀찮게 해킹 툴 깔 필요도 없음
- 해킹 툴 설치가 간단
- 홈 페이지 : <https://www.kali.org/>

Kali Linux와 웹해킹 도구

- OWASP ZAP: 프록시 툴 + 간단한 Scanner 등
- Burp Suite: 유명한 프록시 툴
- SQLmap: SQL Injection 툴
- WPScan: 워드프레스 전용 스캐너
- 기타 등등...

Kali Linux의 Tools

- Metasploit : 시스템 해킹의 자동화
 - 미리 준비된 시스템 해킹 스크립트 가동
- Armitage : metasploit을 GUI로 쓸 수 있게 해 주는 툴
 - 마우스만으로 쉽게 쉘을 열 수 있음
- Wireshark : 패킷 분석
 - 패스워드 평문 전송 등을 알 수 있음
- 이외 총 수십GB에 달하는 수많은 툴이 제공됨
- 전체 리스트 : <https://tools.kali.org/tools-listing>

Debian/Ubuntu에 Kali Linux 툴 추가하기

- 주의 : 이 방법을 수행하면 그냥 Kali Linux로 변한다
 1. 칼리 리눅스 레포지토리 추가
 2. Kali Linux Metapackages 설치

Debian에 칼리 리눅스 레포지토리 추가

1. `su / sudo -s`
2. `apt edit-sources`
3. `deb http://http.kali.org/kali kali-rolling main contrib non-free` 추가
4. `gpg --keyserver pgpkeys.mit.edu --recv-key ED444FF07D8D0BF6`
5. `gpg -a --export ED444FF07D8D0BF6 | apt-key add -`
6. `apt update`
7. `apt upgrade`
8. `apt dist-upgrade`

Kali Linux Metapackage

- 해킹 툴을 일정한 기준에 따라 묶은 것
- Kali-linux : kali 커널, ssh 등 Kali의 기본적 기능
- kali-linux-full : kali를 설치할 때 기본으로 깔리는 툴 전부 설치
- kali-linux-all : kali에서 설치할 수 있는 모든 툴 설치
- kali-linux-web : kali가 제공하는 웹해킹 툴 전부 설치
- 설치 방법 : apt install <패키지 이름>
- 자세한 정보 : <https://www.kali.org/news/kali-linux-metapackages/>

Debian 계열 리눅스 기본 명령어

- `cd <디렉토리 이름>` : 디렉토리 이동
- `ls` : 디렉토리 / 파일 리스트 출력
- `mkdir <이름>` : 디렉토리 만들기
- `rm <이름>` : 파일 삭제
- `man <명령어>` : 도움말(영어) 불러오기
- `apt install <프로그램 이름>` 프로그램 설치
- `apt update && apt upgrade` : 시스템 업데이트

실습

- 실습 서버에서 SQL Injection이 되는 곳을 하나 찾는다