

FLERT [MISC]

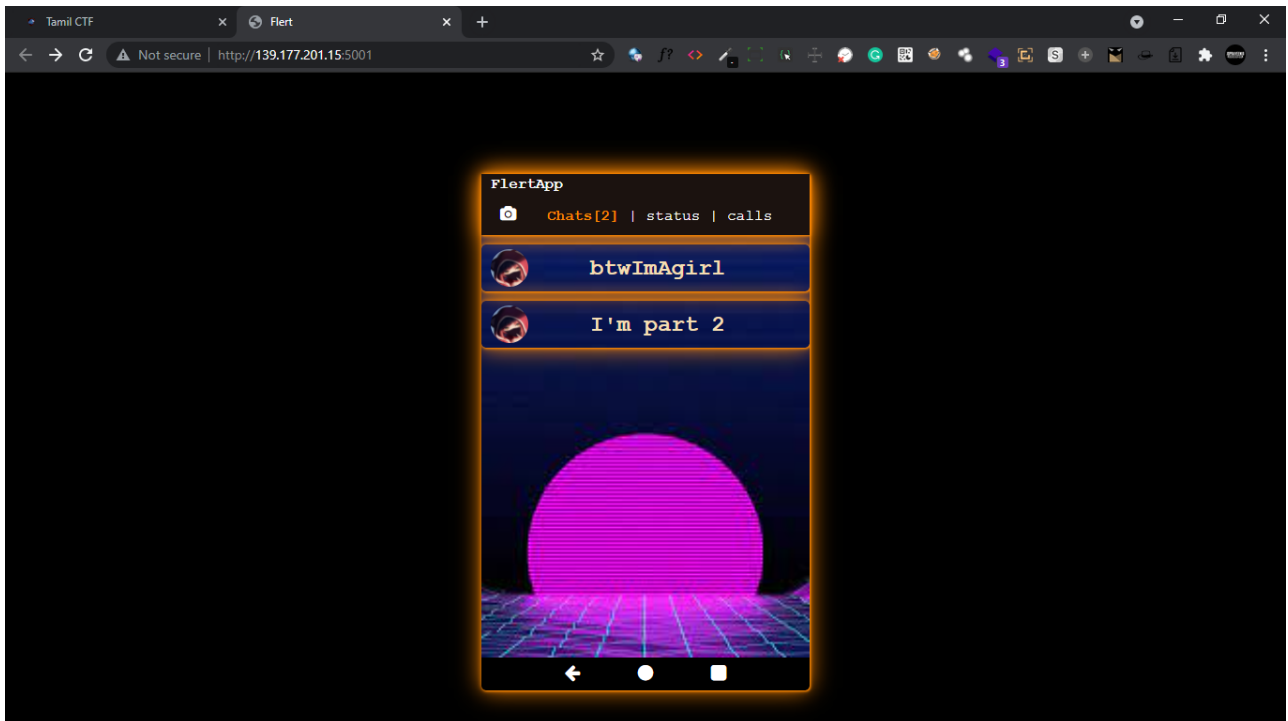
Description

- ▶ This website has cool colours 😊
- ▶ Get first part of flag by chatting with [btwImAgirl](#)
- ▶ Get second part of the flag by chatting with [I'm part 2](#)

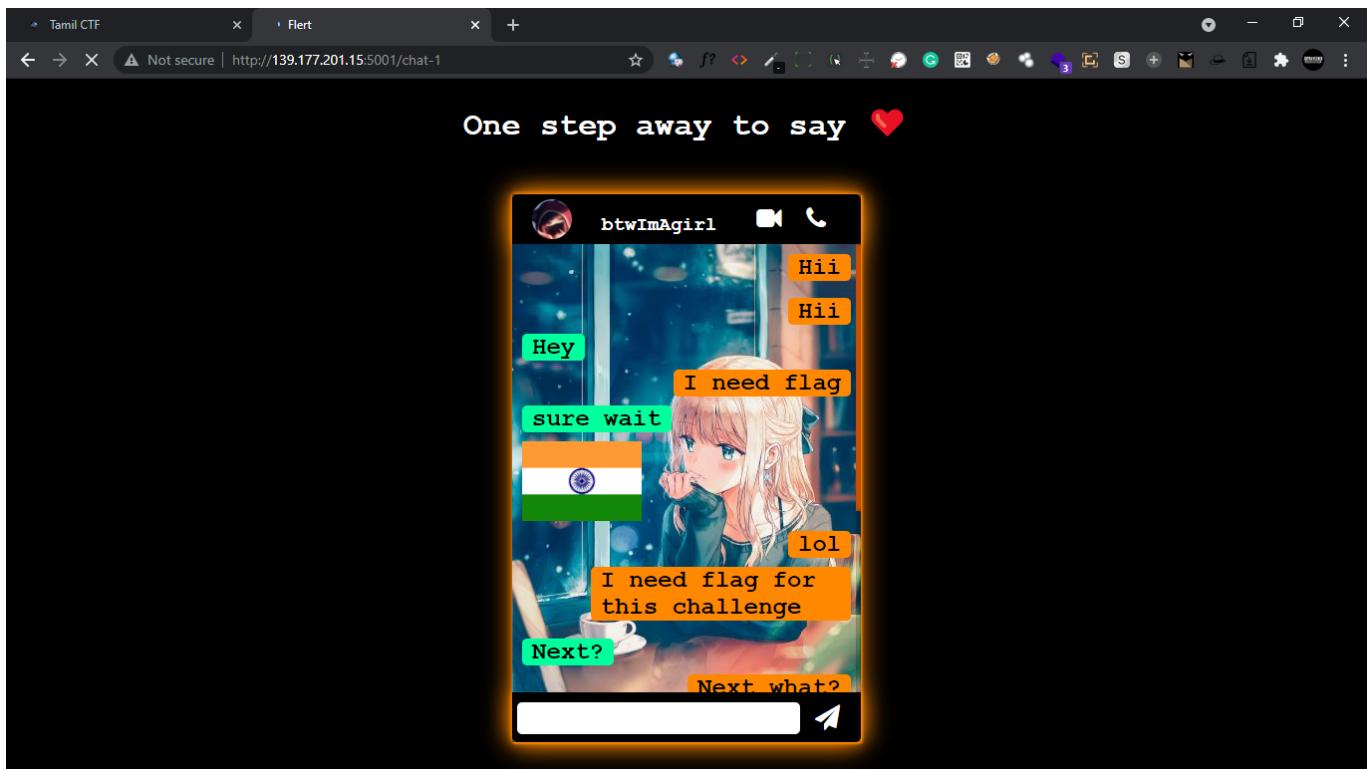
Writeup

First part

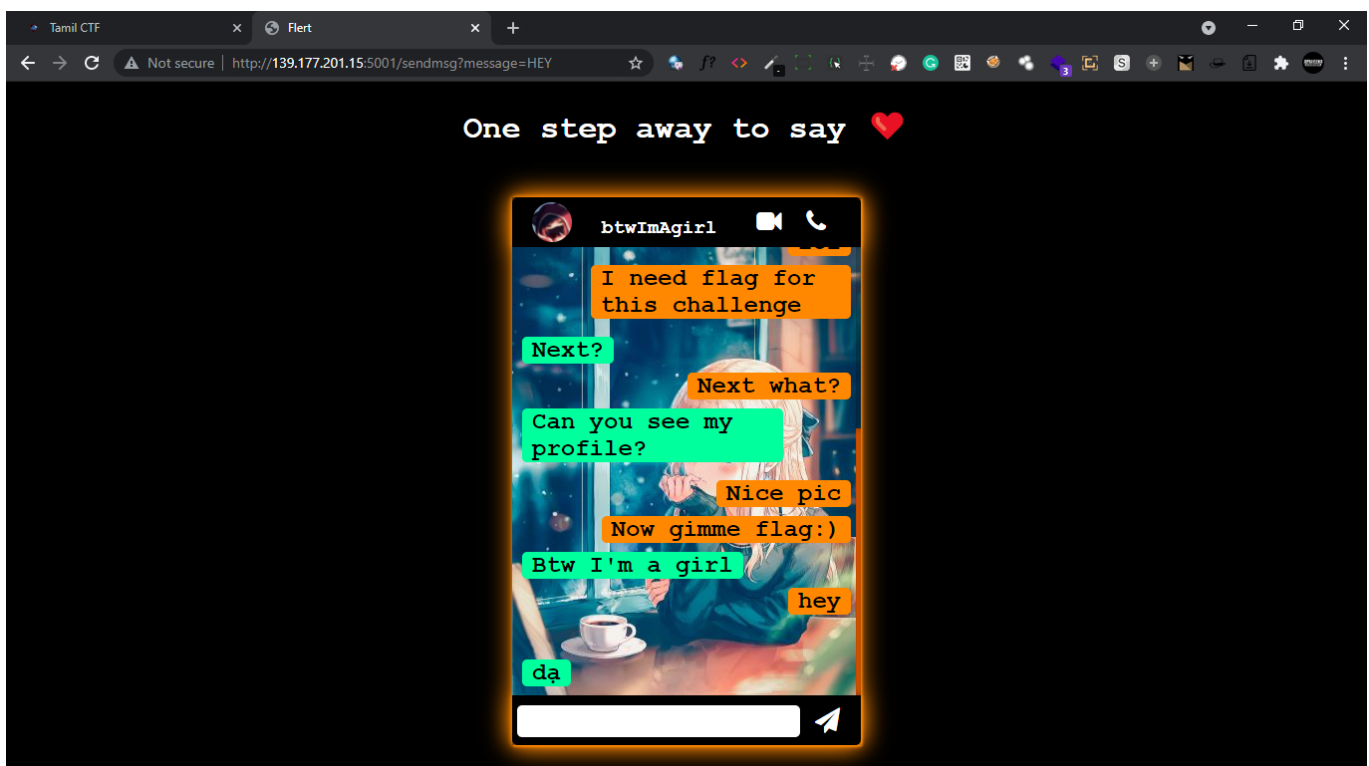
- ▶ First let's see the website
- ▶ Let's click the first chat



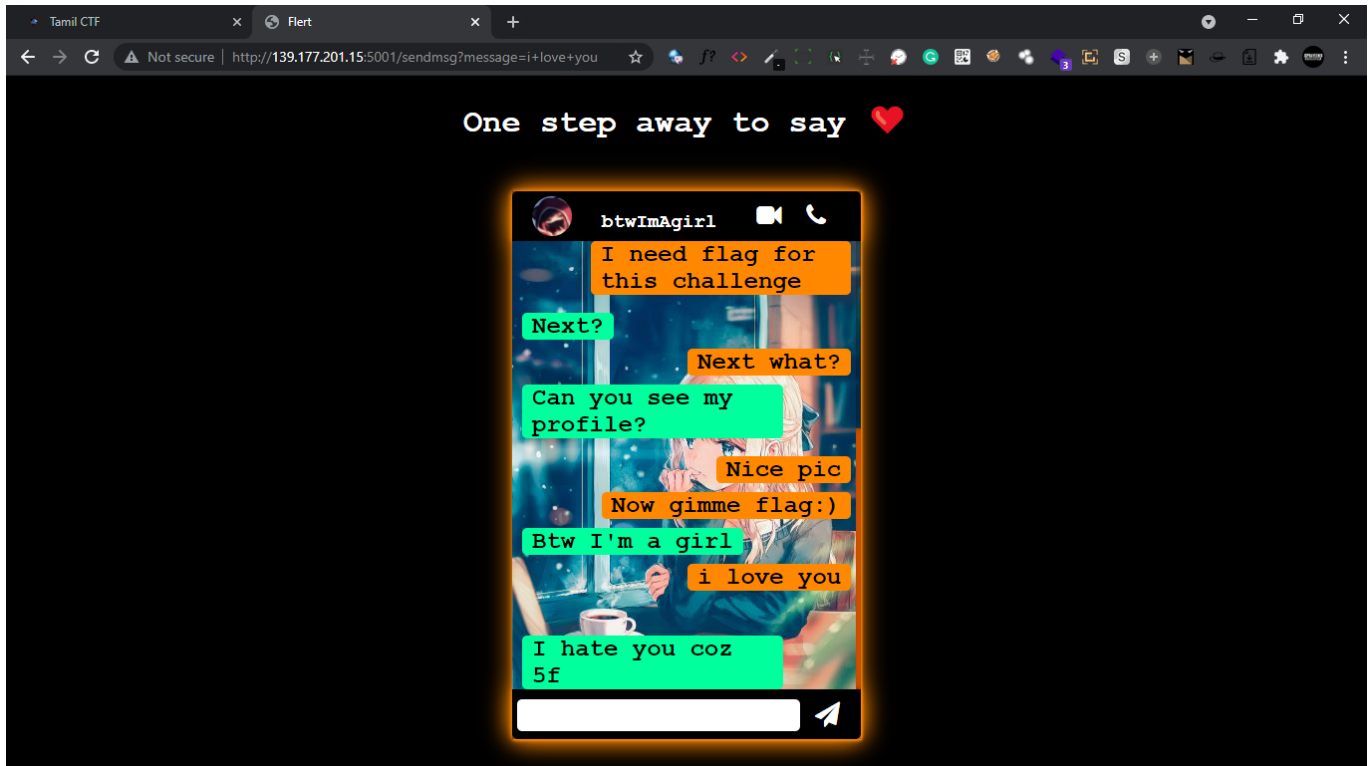
First chat



- ▶ Some chats are here
- ▶ He's asking for flag
- ▶ But she doesn't gave it
- ▶ Now let's send a message



- ▶ Fine it replying us, it looks like a chat bot :)
- ▶ But it doesn't giving us flag :(
- ▶ Look at the title `One step away to say ♥`
- ▶ Fine let's send `i love you`



- ▶ `I hate you coz 5f`
- ▶ Ok what means 5f?
- ▶ Let's send `i love you` again
- ▶ Now it replied `I hate you coz 61`
- ▶ Looks like these are hex values
- ▶ Sending `i love you` again, she replied `I hate you coz 6c`
- ▶ Fine now let's try to collect these values by sending `i love you` multiple times

```
> for i in {1..100}; do curl "http://139.177.201.15:5001/sendmsg?message=i+love+you" | grep "hate" >> out.txt; done
```

- ▶ I'm greping it like this and storing the output in `out.txt`

interesting things in every output

```
color: #4;">I hate you coz 6c
color: #17;">I hate you coz 53
```

- ▶ description says `this website has cool colours` 😊
- ▶ and we getting random hex values in every output
- ▶ So looks like value of the colour is the index
- ▶ ex `color: #4;">I hate you coz 6c`
- ▶ so `6c` is in `4` th index
- ▶ by using this we can arrange the output

Arranging together

- ▶ We don't know how many characters are there
- ▶ So let's put it in a for loop and sort the unique output
- ▶ I'm gonna run this command 1000 times using for loop

```
for i in {1..1000}; do curl "http://139.177.201.15:5001/sendmsg?
message=i+love+you" | grep "hate" >> out.txt; done
```

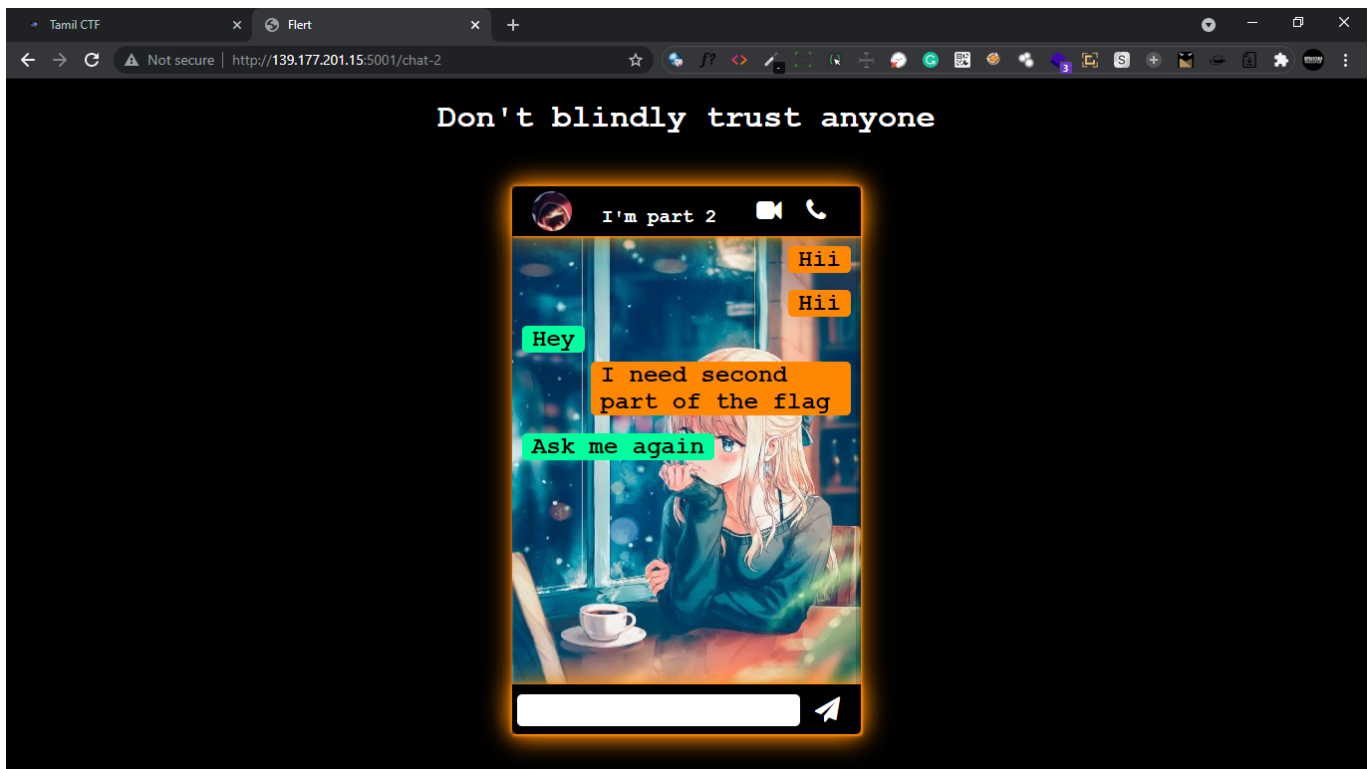
- ▶ And saving this output in `out.txt` so we can uniquely sort things easily
- ▶ sort the unique things
- ▶ and arrange it with those indexes
- ▶ Cool we arranged the hex value
- ▶ Now let's convert this [Hex to Ascii](#)

```
TamilCTF{0.0_w0w_Sh3_
```

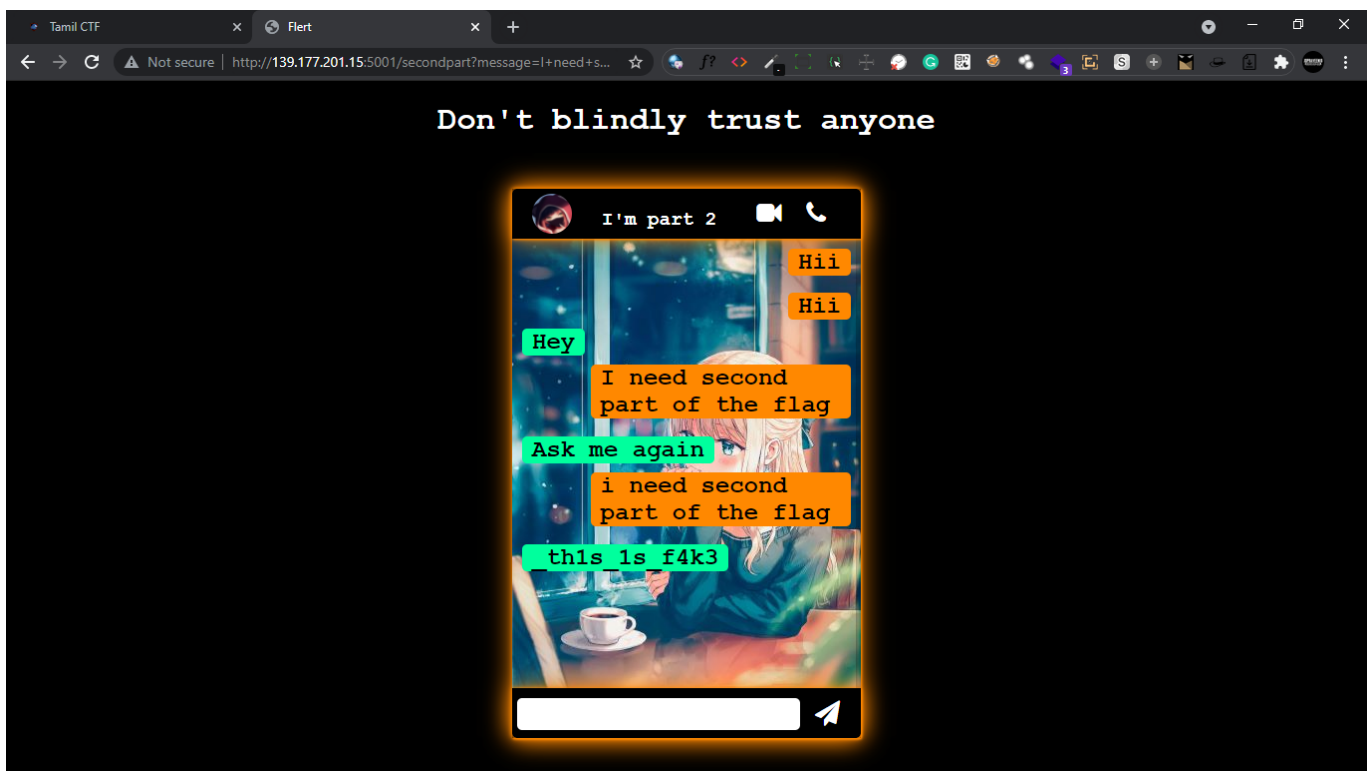
- ▶ Finally we got the first part of the flag
- ▶ Let's go for the second part

Second part

- ▶ Let's go to part 2



- ▶ Some chat's are here, and she said to ask that again
- ▶ Let's send `I need second part of the flag` this



- ▶ We got `_th1s_1s_f4k3` but I think it's fake flag
- ▶ See the title `Don't blindly trust anyone`
- ▶ Flag is not in source

▶ Let's check cookies

The screenshot shows a web browser window with the URL `http://139.177.201.15:5001/secondpart?message=I+need+s...`. The page content displays a chat interface with the text "Don't blindly trust anyone" at the top. Below it, a chat bubble says "I'm part 2". There are two "Hi" messages, a "Hey" message, and a message that says "I need second part of the flag". At the bottom of the chat, it says "Ask me again".

The Application tab in the browser's developer tools is open, showing the cookies for the current domain. The table below represents the data shown in the cookies section:

Name	Value	Domain	Path	Expires / ...	Size	HttpOnly	Secure	SameSite	SameParty	Priority
2nd part	"4cC3P73D_uR_luV_<3"	139.177...	/	Session	29					Medium

Below the table, the "Cookie Value" is shown as `"4cC3P73D_uR_luV_<3"` with a checkbox for "Show URL decoded" which is currently unchecked.

▶ Wow it's here, finally we got second part also

▶ `4cC3P73D_uR_luV_<3}`

▶ Let's join part1 and part2

`TamilCTF{0.0_w0w_Sh3_4cC3P73D_uR_luV_<3}`

▶ That's the flag