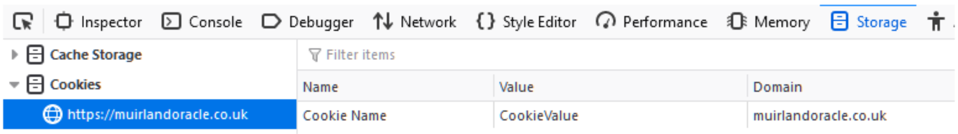
DAY1_advent_of_the_cyber

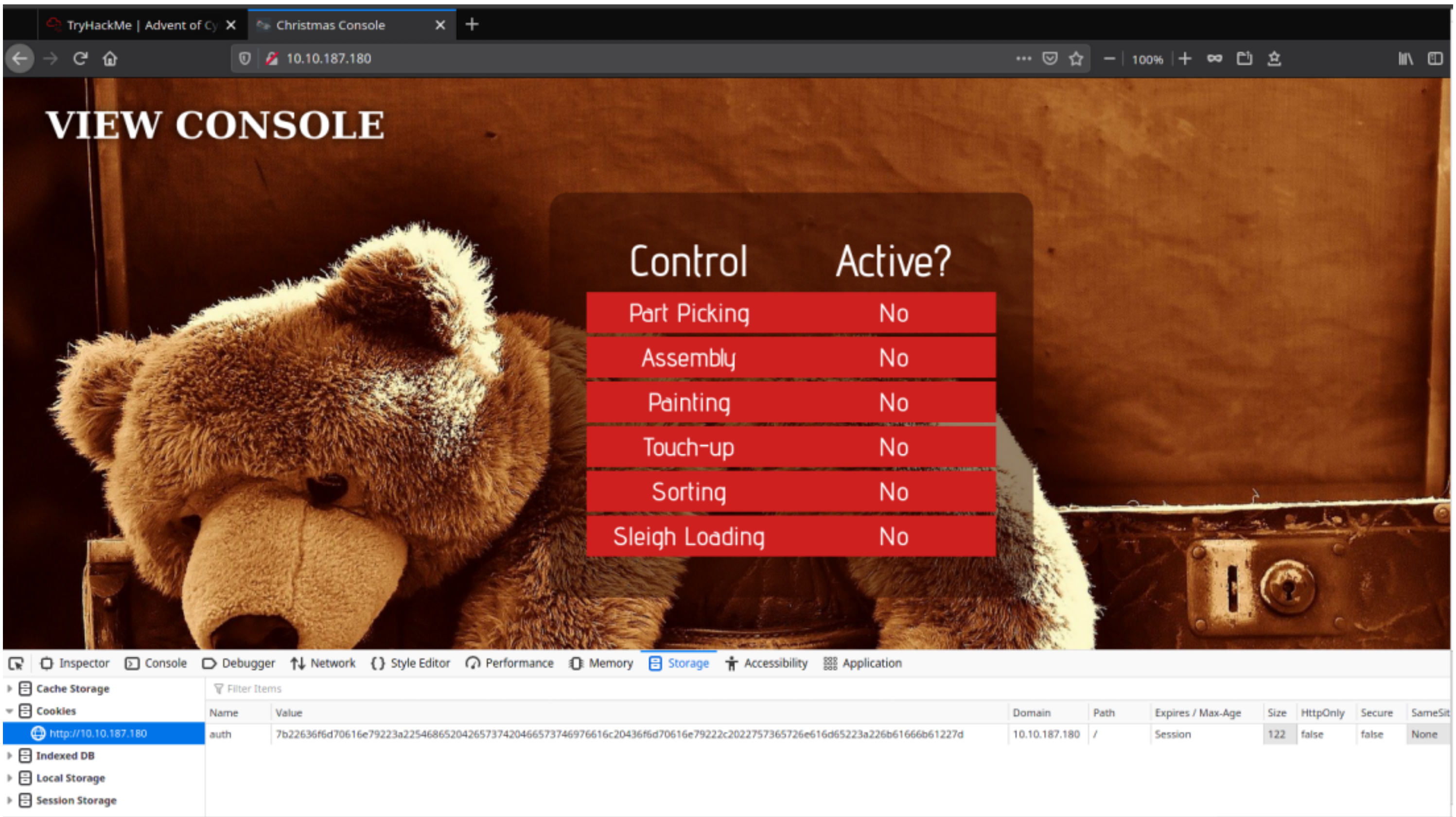
Cookies:

HTTP is an inherently *stateless* protocol. This means that no data persists between connections; your computer could make two requests immediately after each other, and, without relying on separate software, the web server would have no way to know that it was you making both the requests. This begs the important question: if HTTP is stateless, then how do login systems work? The web server must have a way to identify that you have the right level of access, and it can hardly ask you to enter your password every time you request a new page!

The answer is cookies -- tiny little pieces of information that get stored on your computer and get sent to the server along with every request that you make. Authentication (or session) cookies are used to identify you (these will be *very* important in your mission today!). The server receives your request with the attached cookie, and checks the cookie to see what level of access you are allowed to have. It then returns a response appropriate to that level of access.

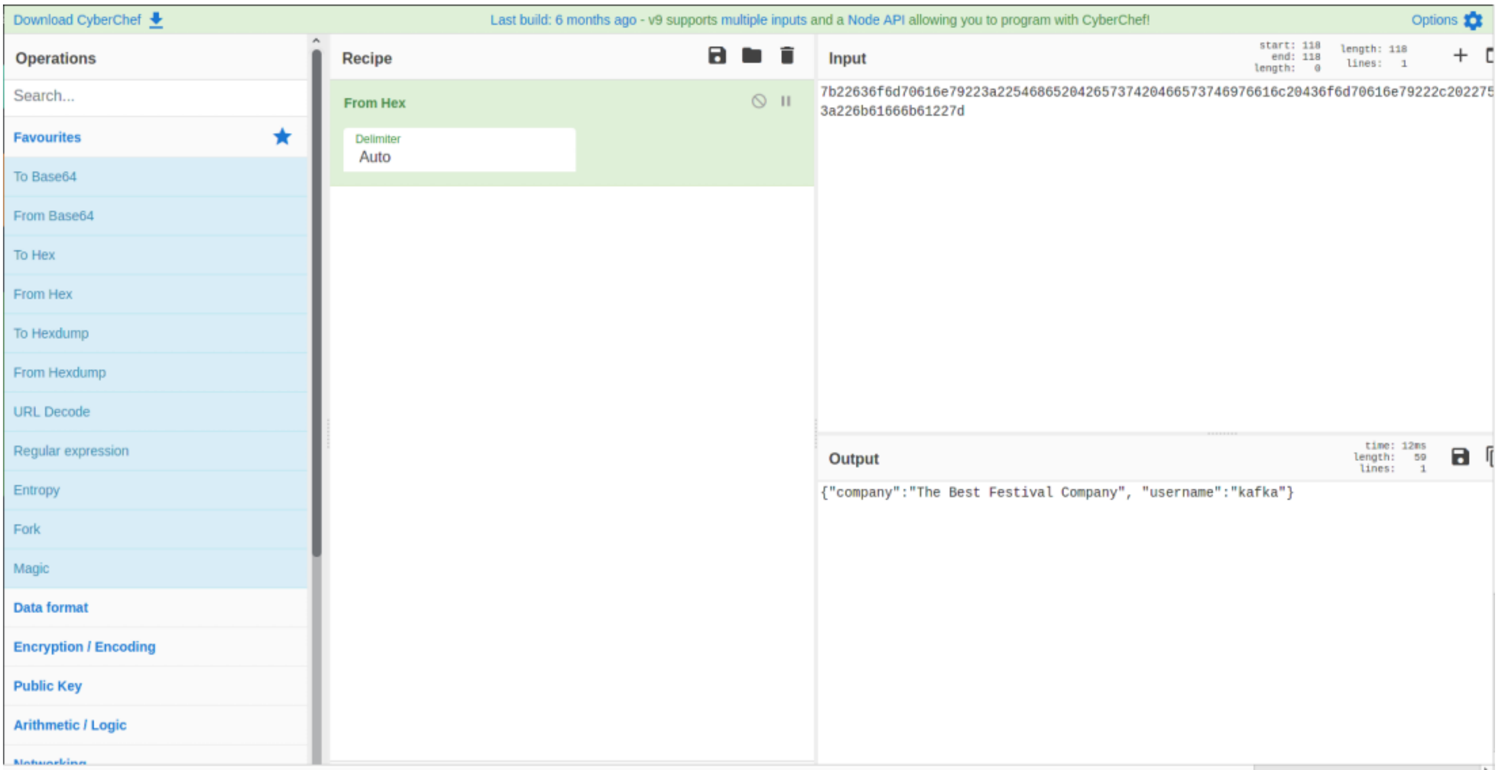
For example, a standard user should be able to see (but not interact with) our control panel; but Santa should be able to access everything! Cookies are also often used for other purposes such as advertising and storing user preferences (light/dark theme, for example); however, this will not be important in your task today. Any site can set cookies with a variety of properties -- the most important of these for today's task are the name and value of the cookies, both of which will always be set. It's worth noting that a site can only access cookies that are associated with its own domain (i.e. google.com can't access any cookies stored by tryhackme.com, and vice versa).

> It's important to note that cookies are stored locally on *your* computer. This means that they are under your control -- i.e. you can add, edit, or delete them as you wish. There are a few ways to do this, however, it's most commonly done by using your Browser Developer Tools, which can be accessed in most browsers by pressing `F12`, or `Ctrl + Shift + I`. With the developer tools open, navigate to the `Storage` tab in FireFox, or the `Application` tab in Chrome/Edge and select the `Cookies` menu on the left hand side of the console.
>
> 
>
>
> | Name | Value | Domain |
> |---|---|---|
> | Cookie Name | CookieValue | muirlandoracle.co.uk |
>
> In the above image you can see a test cookie for a website. The important attributes "Name" and "Value" are shown. The name of a cookie is used to identify it to the server. The value of the cookie is the data stored by the server. In this example the server would be looking for a cookie called "Cookie Name". It would then retrieve the value "CookieValue" from this cookie.
>
> These values can be edited by double-clicking on them, which is great if you can edit a session or authorisation cookie, as this can lead to an escalation of privileges, assuming you have access to an Administrator's authorisation cookie.



Control	Active?
Part Picking	No
Assembly	No
Painting	No
Touch-up	No
Sorting	No
Sleigh Loading	No

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSit
auth	7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c2022757365726e616d65223a226b61666b61227d	10.10.187.180	/	Session	122	false	false	None



From Hex

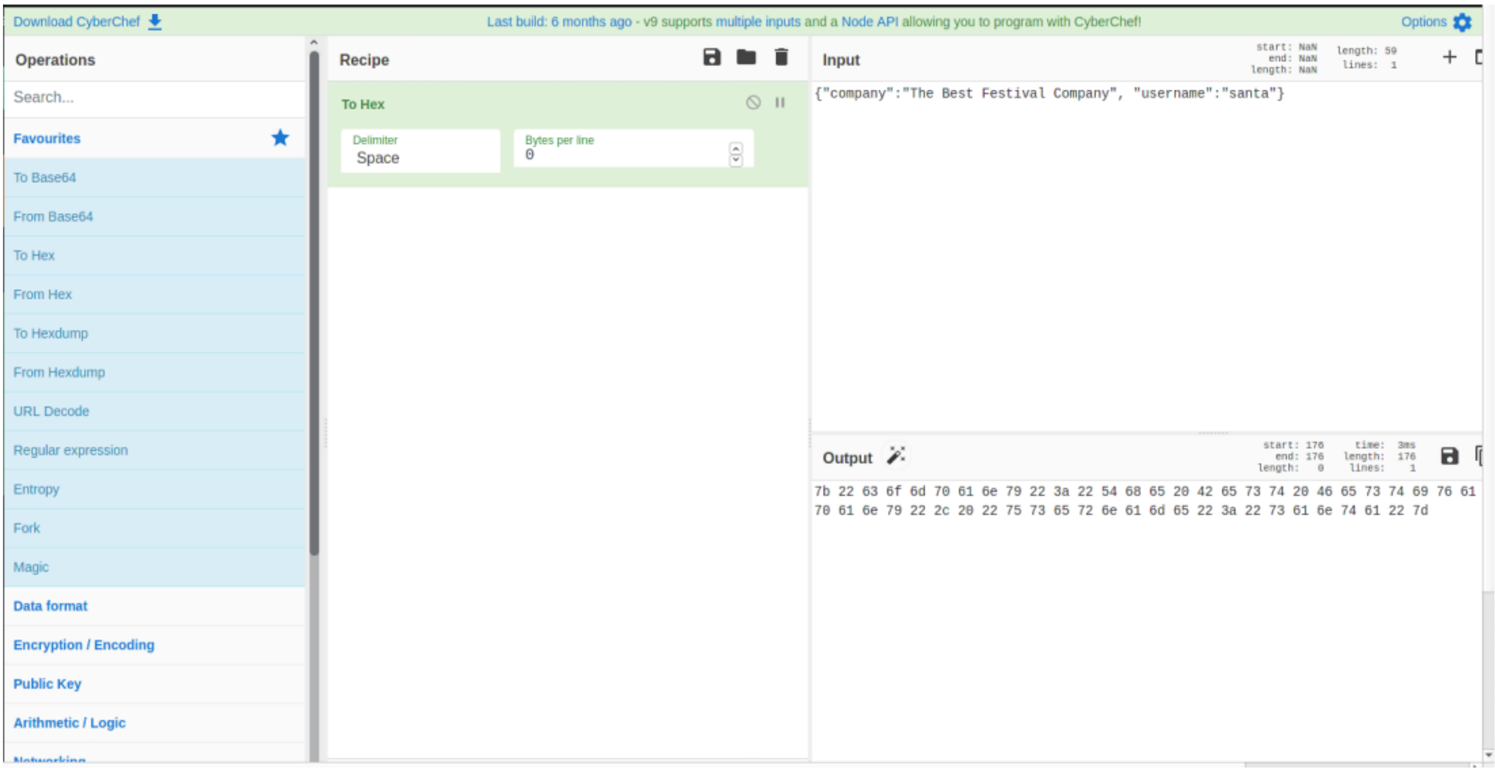
Delimiter: Auto

```
start: 118   length: 118
  end: 118    lines:   1
length:   0
```

```
7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c202275
3a226b61666b61227d
```

```
  time: 12ms
length:   59
 lines:    1
```

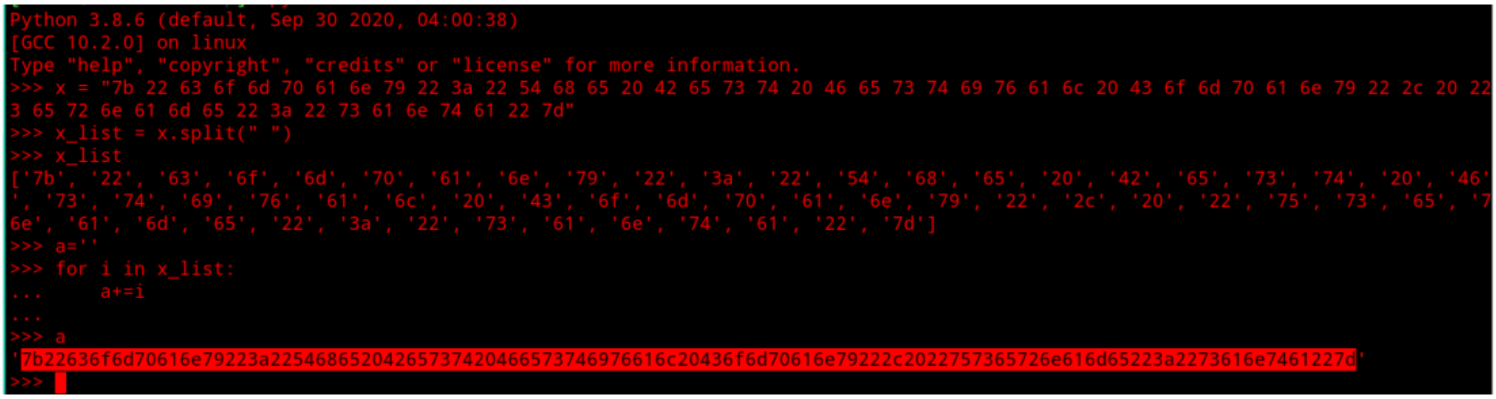
```
{"company":"The Best Festival Company", "username":"kafka"}
```



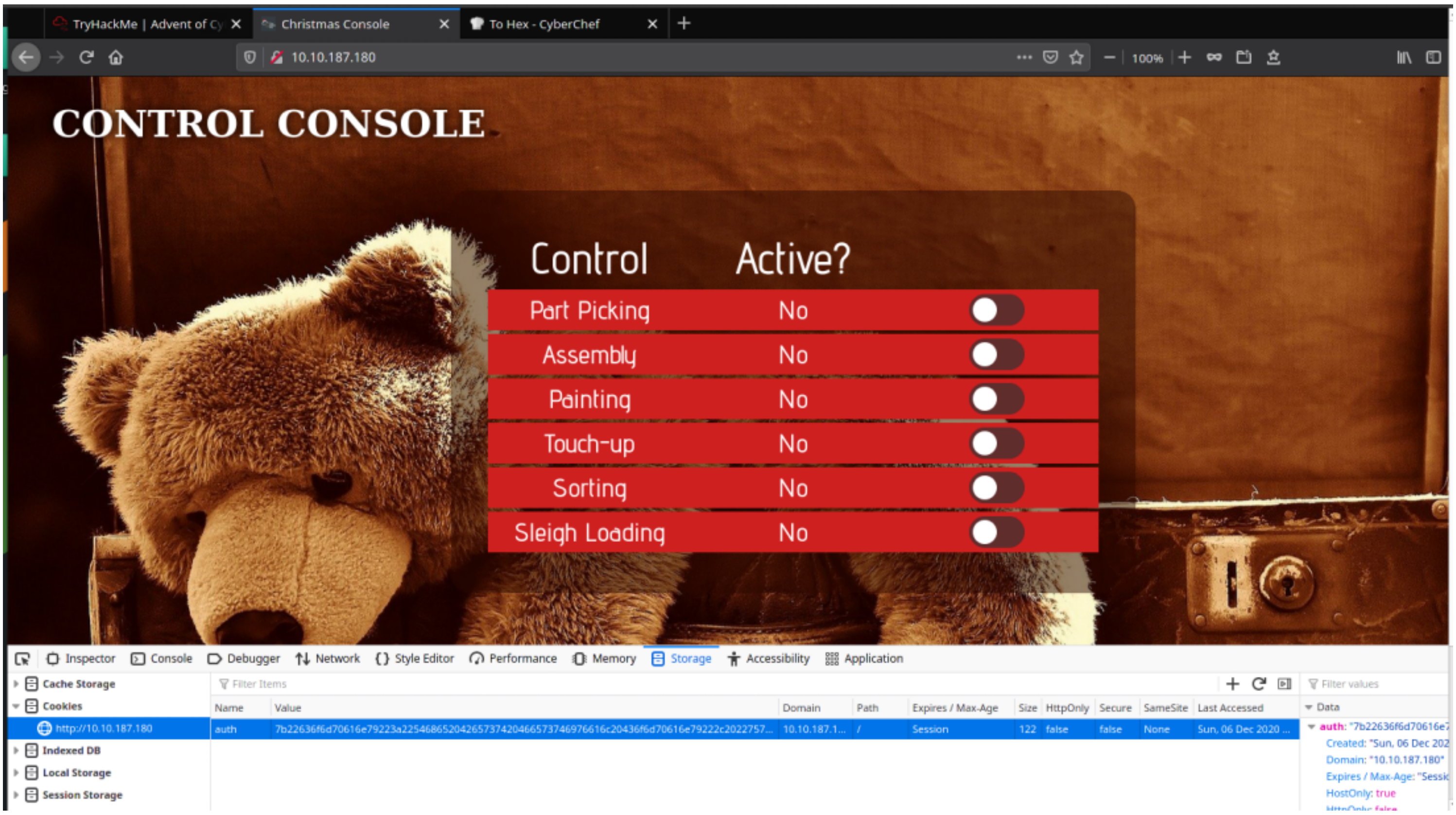
Delimiter: Space
Bytes per line: 0

```
{"company":"The Best Festival Company", "username":"santa"}
```

```
7b 22 63 6f 6d 70 61 6e 79 22 3a 22 54 68 65 20 42 65 73 74 20 46 65 73 74 69 76 61
70 61 6e 79 22 2c 20 22 75 73 65 72 6e 61 6d 65 22 3a 22 73 61 6e 74 61 22 7d
```



```
Python 3.8.6 (default, Sep 30 2020, 04:00:38)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> x = "7b 22 63 6f 6d 70 61 6e 79 22 3a 22 54 68 65 20 42 65 73 74 20 46 65 73 74 69 76 61 6c 20 43 6f 6d 70 61 6e 79 22 2c 20 22
3 65 72 6e 61 6d 65 22 3a 22 73 61 6e 74 61 22 7d"
>>> x_list = x.split(" ")
>>> x_list
['7b', '22', '63', '6f', '6d', '70', '61', '6e', '79', '22', '3a', '22', '54', '68', '65', '20', '42', '65', '73', '74', '20', '46'
', '73', '74', '69', '76', '61', '6c', '20', '43', '6f', '6d', '70', '61', '6e', '79', '22', '2c', '20', '22', '75', '73', '65', '7
6e', '61', '6d', '65', '22', '3a', '22', '73', '61', '6e', '74', '61', '22', '7d']
>>> a=''
>>> for i in x_list:
...     a+=i
...
>>> a
'7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c202275736572e616d65223a2273616e7461227d'
>>> 
```

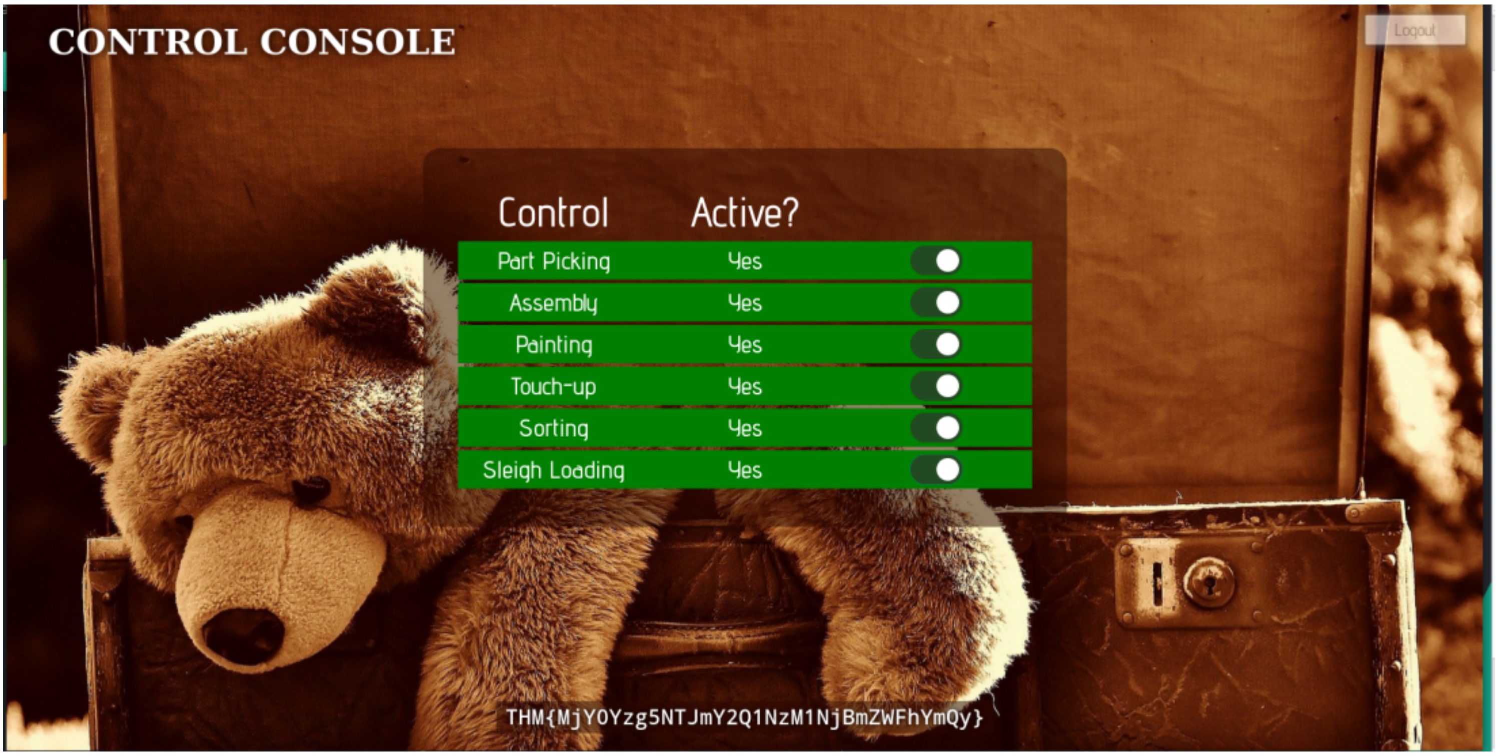


CONTROL CONSOLE

Control	Active?
Part Picking	No
Assembly	No
Painting	No
Touch-up	No
Sorting	No
Sleigh Loading	No

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
auth	7b22636f6d70616e79223a22546865204265737420466573746976616c20436f6d70616e79222c2022757...	10.10.187.1...	/	Session	122	false	false	None	Sun, 06 Dec 2020 ...

Data
auth: "7b22636f6d70616e7
Created: "Sun, 06 Dec 202
Domain: "10.10.187.180"
Expires / Max-Age: "Sessic
HostOnly: true



CONTROL CONSOLE

Control	Active?	
Part Picking	Yes	●
Assembly	Yes	●
Painting	Yes	●
Touch-up	Yes	●
Sorting	Yes	●
Sleigh Loading	Yes	●

THM{MjY0Yzg5NTJmY2Q1NzM1NjBmZWFhYmQy}