SonarQube SAST Report

generated by ReflectSonar

Date: 2025-09-29 18:32:14

SonarQube Project Name: juice-shop

Security E **1** Open Issues	**Reliability** C **46** Open Issues	**Maintainability** A **452** Open Issues
Accepted Issues C **0** Valid issues that were not fixed	**Coverage** B **0.0%** on 15106 lines to cover	**Duplications** D **11.4%** on 68620 lines
Security Hotspots **263** Open Issues		

This report is generated by ReflectSonar, an open-source tool to add the report generation mechanism to SonarQube Community and Developer Edition. It is not affiliated with Sonar. The report is generated based on SonarQube instance that its information is provided. All data is fetched from SonarQube API. ReflectSonar just provides a way to generate the report.

Security Issues

Total: 1 issues (Blocker: 1)

Severity	File Path	Rule & Message
B	lib/insecurity.ts **(Line 23)**	**secrets:S6706** Make sure this private key gets revoked, changed, and removed from the code.

■ **Problematic Code:**

```
     20: import * as z85 from 'z85'
     21:
     22: export const publicKey = fs ? fs.readFileSync('encryptionkeys/jwt.pub', 'utf8') :
'placeholder-public-key'
>>>  23: const privateKey = '-----BEGIN RSA PRIVATE KEY-----\r\nMIICXAIBAAKBgQDNwqLEe9wgTXCbC7+
RPdDbBbeqjdbs4kOPOIGzqLpXvJXlxxW8iMz0EaM4BKUqYsIa+ndv3NAn2RxCd5ubVdJJcX43zO6Ko0TFEZx/65gY3BE0O6
syCEmUP4qbSd6exou/F+WTISzbQ5FBVPVmhnYhG/kpwt/cIxK5iUn5hm+4tQIDAQABAoGBAI+8xiPoOrA+KMnG/T4jJsG6T
sHQcDHvJi7o1IKC/hnIXha0atTX5AUkRRce95qSfvKFweXdJXSQ0JMGJyfuXgU6dI0TcseFRfewXAa/ssxAC+iUVR6KUMh1
PE2wXLitfeI6JLvVtrBYswm2I7CtY0q8n5AGimHWVXJPLfGV7m0BAkEA+fqFt2LXbLtyg6wZyxMA/cnmt5Nt3U2dAu77MzF
JvibANUNHE4HPLZxjGNXN+a6m0K6TD4kDdh5HfUYLWWRBYQJBANK3carmulBwqzcDBjsJ0YrIONBpCAsXxk8idXb8jL9aNI
g15Wumm2enqqObahDHB5jnGOLmbasizvSVqypfM9UCQCQl8xIqy+YgURXzXCN+kwUgHinrutZms87Jyi+D8Br8NY0+Nlf+z
HvXAomD2W5CsEK7C+8SLBr3k/TsnRWHJuECQHFE9RA2OP8WoaLPuGCyFXaxzICThSRZYluVnWkZtxsBhW2W8z1b8PvWUE7k
My7TnkzeJS2LSnaNHoyxi7IaPQUCQCwWU4U+v4lD7uYBw00Ga/xt+7+UqFPlPVdz1yyr4q24Zxaw0LgmuEvgU5dycq8N7Jx
jTubX0MIRR+G9fmDBBl8=\r\n-----END RSA PRIVATE KEY-----'
     24:
     25: interface ResponseWithUser {
     26:   status?: string
```

Reliability Issues

Total: 46 issues (Medium: 17, Low: 29)

Severity	File Path	Rule & Message
M	data/datacache.ts **(Line 33)**	**typescript:S6861** Exporting mutable 'let' binding, use 'const' instead.

```
■ Problematic Code:
     30: }
     31: export const notifications: Notification[] = []
     32:
>>>  33: export let retrieveBlueprintChallengeFile: string | null = null
     34: export function setRetrieveBlueprintChallengeFile (retrieveBlueprintChallengeFileArg:
string) {
     35:   retrieveBlueprintChallengeFile = retrieveBlueprintChallengeFileArg
     36: }
```

M	frontend/src/app/score-board/components challenge-card challenge-card.component.scss **(Line 143)**	**css:S4656** Unexpected duplicate "color"

```
■ Problematic Code:
    140:
    141:   border-radius: calc($badge-size / 2);
    142:   color: var(--theme-text) !important;
>>> 143:   color: var(--theme-text);
    144:   cursor: pointer;
    145:
    146:   display: flex;
```

M	routes/chatbot.ts **(Line 25)**	**typescript:S6861** Exporting mutable 'let' binding, use 'const' instead.

```
■ Problematic Code:
     22:
     23: let trainingFile = config.get<string>('application.chatBot.trainingData')
     24: let testCommand: string
>>>  25: export let bot: Bot | null = null
     26:
     27: export async function initializeChatbot () {
     28:   if (utils.isUrl(trainingFile)) {
```

M	test/server/blueprintSpec.ts **(Line 20)**	**typescript:S1848** Either remove this useless object instantiation of "ExifImage" or use it.

■ **Problematic Code:**

```
    17: async function parseExifData (path: string): Promise<any> {
    18:   return await new Promise((resolve, reject) => {
    19:     // eslint-disable-next-line no-new
>>> 20:     new ExifImage({ image: path }, (error: Error | null, exifData: any) => {
    21:       if (error != null) {
    22:         expect.fail(`Could not read EXIF data from ${path}`)
    23:         reject(error)
```

M	frontend/src/app/payment payment.component.html **(Line 84)**	**Web:S6853** A form label must be associated with a control.

■ **Problematic Code:**

```
    81:       <div class="row responsive-row">
    82:         <div class="col col-34 mat-elevation-z0">
    83:           <div class="payment-label">
>>> 84:             <label translate>LABEL_DONATIONS</label>
    85:           </div>
    86:           <small>
    87:             (@if (applicationName === 'OWASP Juice Shop') {
```

M	frontend/src/app/order-completion order-completion.component.html **(Line 76)**	**Web:S5256** Add "" headers to this "".

■ **Problematic Code:**

```
    73:           <mat-header-cell *matHeaderCellDef
translate>LABEL_TOTAL_PRICE</mat-header-cell>
    74:           <mat-cell *matCellDef="let element" class="price-align">{{
(element.total).toFixed(2) }}&curren;</mat-cell>
    75:           <mat-footer-cell *matFooterCellDef>
>>> 76:             <table class="price-align">
    77:               <tr class="mat-row">
    78:                 <td>{{ orderDetails.itemTotal?.toFixed(2) }}&curren;</td>
    79:               </tr>
```

M	frontend/src/app/search-result search-result.component.html **(Line 40)**	**Web:S6842** Non-interactive elements should not be assigned interactive roles.

■ **Problematic Code:**

```
    37:                     }
    38:                     <div class="product" (click)="showDetail(item)" aria-label="Click for
more information about the product"
    39:                       matTooltip="Click for more information" matTooltipPosition="above">
>>> 40:                       <img mat-card-image
[src]="'assets/public/images/products/'+item.image" alt={{item.name}}
    41:                         class="img-responsive img-thumbnail" role="button">
    42:                         <div class="info-box">
    43:                           <div class="item-name">{{item.name}}</div>
```

M	frontend/src/app/order-summary order-summary.component.html **(Line 50)**	**Web:S5256** Add "" headers to this "".

■ **Problematic Code:**

```
     47:       <mat-card appearance="outlined" class="mat-elevation-z0 fill">
     48:         <div class="mdc-card">
     49:           <div class="order-summary">{{"ORDER_SUMMARY" | translate}}</div>
>>>  50:           <table class="mat-table">
     51:             <tr class="mat-row">
     52:               <td class="mat-cell label">{{"ITEMS" | translate}}</td>
     53:               <td class="mat-cell price">{{ itemTotal?.toFixed(2) }}&curren;</td>
```

M	frontend/src/app/order-completion order-completion.component.html **(Line 56)**	**Web:S5256** Add "" headers to this "".

■ **Problematic Code:**

```
     53:             <span>{{element.quantity}}</span>
     54:           </mat-cell>
     55:           <mat-footer-cell *matFooterCellDef>
>>>  56:             <table class="mat-table">
     57:               <tr class="mat-row">
     58:                 <td translate>ITEMS</td>
     59:               </tr>
```

M	frontend/src/app/contact contact.component.html **(Line 38)**	**Web:S6853** A form label must be associated with a control.

■ **Problematic Code:**

```
     35:         </mat-form-field>
     36:
     37:         <div class="rating-container">
>>>  38:           <label style="font-weight:500; margin-right: 8px; float:left;"
translate>LABEL_RATING</label>
     39:           <mat-slider id="rating" min="1" max="5" step="1" showTickMarks discrete
[displayWith]="formatRating" aria-label="Slider for selecting the star rating">
     40:             <input matSliderThumb [(ngModel)]="rating" />
     41:           </mat-slider>
```

M	frontend/src/app/contact contact.component.html **(Line 45)**	**Web:S6853** A form label must be associated with a control.

■ **Problematic Code:**

```
     42:         </div>
     43:
     44:         <div style="margin-bottom: 10px; margin-top: 10px; ">
>>>  45:           <label style="font-weight:500;">CAPTCHA:</label>&nbsp;&nbsp;<span
style="font-size:small;"
     46:         translate>LABEL_WHAT_IS</span>&nbsp;
     47:         <code id="captcha" aria-label="CAPTCHA code which must be
solved">{{captcha}}</code>&nbsp;<label
     48:       style="font-size:small;">?</label>
```

M	frontend/src/app/contact contact.component.html **(Line 47)**	**Web:S6853** A form label must be associated with a control.

■ Problematic Code:

```
     44:         <div style="margin-bottom: 10px; margin-top: 10px; ">
     45:           <label style="font-weight:500;">CAPTCHA:</label>&nbsp;&nbsp;<span
style="font-size:small;"
     46:         translate>LABEL_WHAT_IS</span>&nbsp;
>>>  47:         <code id="captcha" aria-label="CAPTCHA code which must be
solved">{{captcha}}</code>&nbsp;<label
     48:       style="font-size:small;">?</label>
     49:     </div>
     50:     <mat-form-field appearance="outline" color="accent">
```

M	frontend/src/app/payment payment.component.html **(Line 118)**	**Web:S6853** A form label must be associated with a control.

■ Problematic Code:

```
    115:         <span class="fill-remaining-space"></span>
    116:         <div class="col col-65 mat-elevation-z0">
    117:           <div class="payment-label">
>>> 118:             <label translate>LABEL_MERCHANDISE</label>
    119:           </div>
    120:           <small>
    121:             (@if (applicationName === 'OWASP Juice Shop') {
```

M	routes/languages.ts **(Line 69)**	**typescript:S6671** Expected the Promise rejection reason to be an Error.

■ Problematic Code:

```
     66:           }
     67:           resolve((differentStrings / totalStrings) * 100)
     68:         } catch (err) {
>>>  69:           reject(err)
     70:         }
     71:       })
     72:     }
```

M	lib/startup/validatePreconditions.ts **(Line 103)**	**typescript:S6671** Expected the Promise rejection reason to be an Error.

■ Problematic Code:

```
    100:   return await new Promise((resolve, reject) => {
    101:     portscanner.checkPortStatus(portNumber, function (error: unknown, status: string) {
    102:       if (error) {
>>> 103:         reject(error)
    104:       } else {
    105:         if (status === 'open') {
    106:           logger.warn(`Port ${colors.bold(port.toString())} is in use
(${colors.red('NOT OK')})`)
```

M	frontend/src/app/token-sale token-sale.component.html **(Line 96)**	**Web:S6850** Headings must have content and the content must be accessible by a screen reader.

Problematic Code:

```
     93:
     94:       <mat-card appearance="outlined" class="mat-elevation-z6">
     95:         <div class="mdc-card">
>>>  96:           <h5><i class='fas fa-comments fa-2x'></i> <span style="margin-left: 10px;"
[innerHtml]="'ICO_FAQ_QUESTION' | translate:{juicycoin: altcoinName}"></span></h5>
     97:           <small class="text-justify">{{"ICO_FAQ_ANSWER" | translate}}</small>
     98:         </div>
     99:       </mat-card>
```

M

frontend/src/assets/private
threejs-demo.html
(Line 2)

Web:S5254
Add "lang" and/or "xml:lang" attributes to this "" element

Problematic Code:

```
      1: <!DOCTYPE html>
>>>   2: <html>
      3: <head>
      4:     <title>Welcome to Planet Orangeuze</title>
      5:     <script src="/assets/private/three.js"></script>
```

L

frontend/src/app/chatbot
chatbot.component.html
(Line 18)

Web:ImgWithoutAltCheck
Add an "alt" attribute to this image.

Problematic Code:

```
     15:                 @for (message of messages; track message; let index = $index) {
     16:                   <div class="message-container">
     17:                     @if (message.author == 'bot') {
>>>  18:                       <img src="{{juicyImageSrc}}" class="juicy-chat-bot-image"/>
     19:                     }
     20:                     @if (message.author == 'user') {
     21:                       <img src="{{profileImageSrc}}" class="profile-image"/>
```

L

frontend/src/app/chatbot
chatbot.component.html
(Line 21)

Web:ImgWithoutAltCheck
Add an "alt" attribute to this image.

Problematic Code:

```
     18:                       <img src="{{juicyImageSrc}}" class="juicy-chat-bot-image"/>
     19:                     }
     20:                     @if (message.author == 'user') {
>>>  21:                       <img src="{{profileImageSrc}}" class="profile-image"/>
     22:                     }
     23:                     <div class="{{message.author == 'user' ? 'speech-bubble-right' :
'speech-bubble-left'}}">
     24:                       {{message.body}}
```

L

frontend/src/app/code-area
code-area.component.html
(Line 2)

Web:MouseEventWithoutKeyboardEquivalentCheck
Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
      1: <div id="code-area">
>>>   2:   <pre id="code"><div id="emphasize">@for (marker of lineMarkers; track marker) {<div
class="lineMarker" [id]="'line' + marker.lineNumber" (click)="selectLines(marker.lineNumber)">{{
marker.marked ? '■' : '■'}}</div>}
      3: </div><code
      4: [highlight]="code" [lineNumbers]="true"
      5: [languages]="langs"
```

L	frontend/src/app/sidenav sidenav.component.html **(Line 54)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
     51:     }
     52:
     53:     @if (isLoggedIn()) {
>>>  54:       <mat-list-item (click)="showOrdersSubmenu = !showOrdersSubmenu" class="parent"
     55:         aria-label="Show Orders and Payment Menu">
     56:         <mat-icon>
     57:           check_circle_outline
```

L	frontend/src/app/sidenav sidenav.component.html **(Line 129)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
    126:     }
    127:
    128:     @if (isLoggedIn()) {
>>> 129:       <mat-list-item (click)="showPrivacySubmenu = !showPrivacySubmenu" class="parent"
    130:         aria-label="Show Privacy and Security Menu">
    131:         <mat-icon>
    132:           security
```

L	frontend/src/app/address address.component.html **(Line 19)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
     16:         <ng-container matColumnDef="Selection">
     17:           <mat-header-cell *matHeaderCellDef style="display: none;"></mat-header-cell>
     18:           <mat-cell *matCellDef="let element; let row">
>>>  19:             <mat-radio-button
     20:               (click)="emitSelectionToParent(element.id)"
     21:               [checked]="selection.isSelected(row)"
     22:               (change)="$event ? selection.toggle(row) : null">
```

L	frontend/src/app/address address.component.html **(Line 61)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ Problematic Code:

```
     58:           </mat-cell>
     59:         </ng-container>
     60:         <mat-header-row *matHeaderRowDef="displayedColumns"></mat-header-row>
>>>  61:         <mat-row
     62:           *matRowDef="let row; columns: displayedColumns; let element"
     63:           (click)="selection.toggle(row); emitSelectionToParent(element.id)">
     64:         </mat-row>
```

L	frontend/src/app/search-result search-result.component.html **(Line 38)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ Problematic Code:

```
     35:                       <span>{{"LABEL_SOLD_OUT" | translate }}</span>
     36:                     </div>
     37:                   }
>>>  38:                   <div class="product" (click)="showDetail(item)" aria-label="Click for
more information about the product"
     39:                     matTooltip="Click for more information" matTooltipPosition="above">
     40:                     <img mat-card-image
[src]="'assets/public/images/products/'+item.image" alt={{item.name}}
     41:                       class="img-responsive img-thumbnail" role="button">
```

L	frontend/src/app/navbar navbar.component.html **(Line 134)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ Problematic Code:

```
    131:         </mat-form-field>
    132:       </div>
    133:       @for (language of filteredLanguages; track language.key) {
>>> 134:         <mat-radio-button
    135:           (click)="changeLanguage(language.key)"
    136:           [value]="language"
    137:           class="mat-menu-item"
```

L	frontend/src/app/navbar navbar.component.html **(Line 126)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ Problematic Code:

```
    123:     </button>
    124:
    125:     <mat-menu #menu="matMenu" [overlapTrigger]="true">
>>> 126:       <div class="language-search-container" (click)="$event.stopPropagation()">
    127:         <mat-form-field class="language-search-field">
    128:           <mat-icon matPrefix class="search-icon">search</mat-icon>
    129:           <mat-label>{{ 'SEARCH' | translate }}</mat-label>
```

L	frontend/src/app/administration administration.component.html **(Line 59)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
     56:
     57:           <ng-container matColumnDef="comment">
     58:             <mat-header-cell *matHeaderCellDef translate></mat-header-cell>
>>>  59:             <mat-cell *matCellDef="let feedback"
(click)="showFeedbackDetails(feedback.comment, feedback.UserId)">
     60:               <p [innerHTML]="feedback.comment" matTooltip="Click for more information"
matTooltipPosition="above"></p>
     61:             </mat-cell>
     62:           </ng-container>
```

L	frontend/src/app/administration administration.component.html **(Line 119)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
    116:   <mat-row *matRowDef="let row; columns: userColumns; "></mat-row>
    117: </mat-table>
    118:
>>> 119: <img src="assets/public/images/padding/19px.png"/>
```

L	frontend/src/app/mat-search-bar mat-search-bar.component.html **(Line 9)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
      6:
      7:
      8: <span class="mat-search_icons" [class.mat-search_icons--active]="searchVisible">
>>>   9:     <mat-icon class="mat-search_icon-close" (click)="close()" matRipple>
     10:         close
     11:     </mat-icon>
     12:     <mat-icon class="mat-search_icon-search" (click)="open()" matRipple>
```

L	frontend/src/app/mat-search-bar mat-search-bar.component.html **(Line 12)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
      9:     <mat-icon class="mat-search_icon-close" (click)="close()" matRipple>
     10:         close
     11:     </mat-icon>
>>>  12:     <mat-icon class="mat-search_icon-search" (click)="open()" matRipple>
     13:         search
     14:     </mat-icon>
     15: </span>
```

L	frontend/src/app/web3-sandbox web3-sandbox.component.html **(Line 169)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
    166:       }
    167:     </div>
    168:   </div>
>>> 169:   <img src="assets/public/images/padding/11px.png" />
    170: </div>
    171:
```

L	frontend/src/app/faucet faucet.component.html **(Line 6)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
      3:     <div class="mdc-card">
      4:       <h1>{{"TITLE_BEE_HAVEN" | translate}}</h1>
      5:       <div class="faucet-image">
>>>   6:         <img src="assets/public/images/BeeOwner.png" />
      7:       </div>
      8:       <p>{{"BEE_HAVEN_INTRO" | translate}}</p>
      9:       <div class="faucet-balance">
```

L	frontend/src/app/faucet faucet.component.html **(Line 67)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
     64:   <mat-card appearance="outlined" class="mat-elevation-z6 honeypot-container">
     65:     <div class="mdc-card">
     66:       <div class="honeypot-image mat-elevation-z6">
>>>  67:         <img src="assets/public/images/HoneyPot.png" />
     68:       </div>
     69:       <h2>{{"BEE_HONEYPOT_TITLE" | translate}}</h2>
     70:       <p>{{"BEE_HONEYPOT_DESCRIPTION" | translate}}</p>
```

L	frontend/src/app/nft-unlock nft-unlock.component.html **(Line 13)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
     10:
     11:     <div class="detail-container offer-container align-self-center">
     12:       <div>
>>>  13:         <img src="assets/public/images/products/juicy_chatbot.jpg" />
     14:       </div>
     15:
     16:       @if (successResponse) {
```

L	frontend/src/app/delivery-method delivery-method.component.html **(Line 28)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
     25:         <ng-container matColumnDef="Selection">
     26:           <mat-header-cell *matHeaderCellDef></mat-header-cell>
     27:           <mat-cell *matCellDef="let element; let row">
>>>  28:             <mat-radio-button (click)=selectMethod(element.id)
[checked]="selection.isSelected(row)"
     29:             (change)="$event ? selection.toggle(row) : null"></mat-radio-button>
     30:           </mat-cell>
     31:         </ng-container>
```

L	frontend/src/app/delivery-method delivery-method.component.html **(Line 45)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
     42:           <mat-cell *matCellDef="let element">{{ element.eta }} {{"LABEL_DAYS" |
translate}}</mat-cell>
     43:         </ng-container>
     44:         <mat-header-row *matHeaderRowDef="displayedColumns"></mat-header-row>
>>>  45:         <mat-row *matRowDef="let row; columns: displayedColumns; let element"
     46:         (click)="selection.toggle(row); selectMethod(element.id)"></mat-row>
     47:       </mat-table>
     48:     </div>
```

L	frontend/src/app/token-sale token-sale.component.html **(Line 101)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
     98:         </div>
     99:       </mat-card>
    100:     </div>
>>> 101:     <img src="assets/public/images/padding/56px.png"/>
    102:   </div>
    103: </mat-card>
    104:
```

L	frontend/src/app/photo-wall photo-wall.component.html **(Line 14)**	**Web:S6851** Remove redundant word "image" from the "alt" attribute of your "img" tag.

■ **Problematic Code:**

```
     11:         <div class="grid">
     12:           @for (image of slideshowDataSource; track image) {
     13:             <span class="container mat-elevation-z6">
>>>  14:               <img src="{{ image.url }}" alt="{{ image.caption }}" class="image">
     15:               <div class="overlay">
     16:                 <div>{{ image.caption }}</div>
     17:                 @if (twitterHandle) {
```

L	frontend/src/app/privacy-policy privacy-policy.component.html **(Line 172)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
    169:       </ul>
    170:     </section>
    171:
>>> 172:     <img src="assets/public/images/padding/81px.png"/>
    173:   </div>
    174: </mat-card>
    175:
```

L	frontend/src/app/search-result search-result.component.html **(Line 40)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
     37:                   }
     38:                   <div class="product" (click)="showDetail(item)" aria-label="Click for
more information about the product"
     39:                     matTooltip="Click for more information" matTooltipPosition="above">
>>>  40:                     <img mat-card-image
[src]="'assets/public/images/products/'+item.image" alt={{item.name}}
     41:                       class="img-responsive img-thumbnail" role="button">
     42:                       <div class="info-box">
     43:                         <div class="item-name">{{item.name}}</div>
```

L	frontend/src/app/payment-method payment-method.component.html **(Line 14)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
     11:         <ng-container matColumnDef="Selection">
     12:           <mat-header-cell *matHeaderCellDef></mat-header-cell>
     13:           <mat-cell *matCellDef="let element">
>>>  14:             <mat-radio-button
(click)="emitSelectionToParent(element.id)"></mat-radio-button>
     15:           </mat-cell>
     16:         </ng-container>
     17:         <ng-container matColumnDef="Number">
```

L	frontend/src/app/product-details product-details.component.html **(Line 51)**	**Web:MouseEventWithoutKeyboardEquivalentCheck** Add a 'onKeyPress|onKeyDown|onKeyUp' attribute to this tag.

■ **Problematic Code:**

```
     48:           @for (review of reviews$|async; track review) {
     49:             <div class="comment">
     50:               <div class="review-row">
>>>  51:                 <div class="review-text"
     52:                   (click)="review.author !== 'Anonymous' && review.author === author &&
editReview(review)"
     53:                   matTooltipDisabled="{{review.author !== author}}"
matTooltip="{{'LABEL_EDIT_REVIEW' | translate}}"
     54:                   matTooltipPosition="right">
```

L	frontend/src/app/score-board score-board.component.html **(Line 53)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
     50:       </div>
     51:     }
     52:   }
>>>  53:   <img src="assets/public/images/padding/1px.png"/>
     54: }
     55:
     56:
```

L	frontend/src/app/recycle recycle.component.html **(Line 64)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
     61:       <h3 class="responsibility-header">{{"SECTION_PRESS_JUICE_RESPONSIBLY" | translate}}</h3>
     62:
     63:       <mat-card class="mat-elevation-z0 card-row">
>>>  64:         <img mat-card-image [src]="topImage">
     65:         <mat-card-content>
     66:           <div>
     67:             <small>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut
```

L	frontend/src/app/recycle recycle.component.html **(Line 76)**	**Web:ImgWithoutAltCheck** Add an "alt" attribute to this image.

■ **Problematic Code:**

```
     73:       </mat-card>
     74:
     75:       <mat-card class="mat-elevation-z0 card-row">
>>>  76:         <img mat-card-image [src]="bottomImage">
     77:         <mat-card-content>
     78:           <div>
     79:             <small>Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor
```

Maintainability Issues

Total: 452 issues (Blocker: 25, High: 103, Medium: 209, Low: 91, Info: 24)

Severity	File Path	Rule & Message
B	frontend/src/app/change-password change-password.component.ts **(Line 80)**	**typescript:S7655** Lifecycle interface 'OnInit' should be implemented for method 'ngOnInit'. (https://angular.dev/style-guide#style-09-01)

```
■ Problematic Code:
     77:     private readonly translate: TranslateService
     78:   ) {}
     79:
>>>  80:   ngOnInit (): void {
     81:     this.formSubmitService.attachEnterKeyHandler(
     82:       'password-form',
     83:       'changeButton',
```

B	frontend/src/app/last-login-ip last-login-ip.component.ts **(Line 23)**	**typescript:S7655** Lifecycle interface 'OnInit' should be implemented for method 'ngOnInit'. (https://angular.dev/style-guide#style-09-01)

```
■ Problematic Code:
     20:   lastLoginIp: any = '?'
     21:   constructor (private readonly sanitizer: DomSanitizer) {}
     22:
>>>  23:   ngOnInit (): void {
     24:     try {
     25:       this.parseAuthToken()
     26:     } catch (err) {
```

B	frontend/src/app/nft-unlock nft-unlock.component.ts **(Line 32)**	**typescript:S7655** Lifecycle interface 'OnInit' should be implemented for method 'ngOnInit'. (https://angular.dev/style-guide#style-09-01)

```
■ Problematic Code:
     29:
     30:   constructor (private readonly keysService: KeysService) {}
     31:
>>>  32:   ngOnInit (): void {
     33:     this.checkChallengeStatus()
     34:   }
     35:
```

B	frontend/src/app/two-factor-auth two-factor-auth.component.ts **(Line 59)**	**typescript:S7655** Lifecycle interface 'OnInit' should be implemented for method 'ngOnInit'. (https://angular.dev/style-guide#style-09-01)

■ Problematic Code:

```
     56:
     57:   constructor (private readonly twoFactorAuthService: TwoFactorAuthService, private readonly configurationService: ConfigurationService, private readonly snackBar: MatSnackBar, private readonly translateService: TranslateService, private readonly snackBarHelperService: SnackBarHelperService) {}
     58:
>>>  59:   ngOnInit (): void {
     60:     this.updateStatus()
     61:   }
     62:
```

B	frontend/src/app/mat-search-bar mat-search-bar.component.ts **(Line 55)**	**typescript:S7652** Output bindings, including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)

■ Problematic Code:

```
     52:   @Input() matAutocomplete: MatAutocomplete
     53:   @Input() placeholder = ''
     54:   @Input() alwaysOpen: boolean = false
>>>  55:   @Output() onBlur = new EventEmitter<string>()
     56:   @Output() onClose = new EventEmitter<void>()
     57:   @Output() onEnter = new EventEmitter<string>()
     58:   @Output() onFocus = new EventEmitter<string>()
```

B	frontend/src/app/mat-search-bar mat-search-bar.component.ts **(Line 56)**	**typescript:S7652** Output bindings, including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)

■ Problematic Code:

```
     53:   @Input() placeholder = ''
     54:   @Input() alwaysOpen: boolean = false
     55:   @Output() onBlur = new EventEmitter<string>()
>>>  56:   @Output() onClose = new EventEmitter<void>()
     57:   @Output() onEnter = new EventEmitter<string>()
     58:   @Output() onFocus = new EventEmitter<string>()
     59:   @Output() onOpen = new EventEmitter<void>()
```

B	frontend/src/app/mat-search-bar mat-search-bar.component.ts **(Line 57)**	**typescript:S7652** Output bindings, including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)

■ Problematic Code:

```
     54:   @Input() alwaysOpen: boolean = false
     55:   @Output() onBlur = new EventEmitter<string>()
     56:   @Output() onClose = new EventEmitter<void>()
>>>  57:   @Output() onEnter = new EventEmitter<string>()
     58:   @Output() onFocus = new EventEmitter<string>()
     59:   @Output() onOpen = new EventEmitter<void>()
     60:
```

B	frontend/src/app/mat-search-bar mat-search-bar.component.ts **(Line 58)**	**typescript:S7652** Output bindings, including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)

■ Problematic Code:

```
    55:   @Output() onBlur = new EventEmitter<string>()
    56:   @Output() onClose = new EventEmitter<void>()
    57:   @Output() onEnter = new EventEmitter<string>()
>>> 58:   @Output() onFocus = new EventEmitter<string>()
    59:   @Output() onOpen = new EventEmitter<void>()
    60:
    61:   searchVisible = false
```

B	frontend/src/app/mat-search-bar mat-search-bar.component.ts **(Line 59)**	**typescript:S7652** Output bindings, including aliases, should not be named "on", nor prefixed with it (https://angular.dev/guide/components/outputs#choosing-event-names)

■ Problematic Code:

```
    56:   @Output() onClose = new EventEmitter<void>()
    57:   @Output() onEnter = new EventEmitter<string>()
    58:   @Output() onFocus = new EventEmitter<string>()
>>> 59:   @Output() onOpen = new EventEmitter<void>()
    60:
    61:   searchVisible = false
    62:
```

B	frontend/src/app/code-fixes code-fixes.component.ts **(Line 28)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
    25:   @Input('fixes')
    26:   public fixes: string[] = []
    27:
>>> 28:   @Input('selectedFix')
    29:   public selectedFix: number = 0
    30:
    31:   @Input('randomFixes')
```

B	frontend/src/app/code-fixes code-fixes.component.ts **(Line 31)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
    28:   @Input('selectedFix')
    29:   public selectedFix: number = 0
    30:
>>> 31:   @Input('randomFixes')
    32:   public randomFixes: RandomFixes[] = []
    33:
    34:   @Input('format')
```

B	frontend/src/app/address address.component.ts **(Line 31)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
    28: })
    29: export class AddressComponent implements OnInit {
    30:   @Output() emitSelection = new EventEmitter()
>>> 31:   @Input('allowEdit') public allowEdit: boolean = false
    32:   @Input('addNewAddressDiv') public addNewAddressDiv: boolean = true
    33:   @Input('showNextButton') public showNextButton: boolean = false
    34:   public addressId: any = undefined
```

B	frontend/src/app/address address.component.ts **(Line 32)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
    29: export class AddressComponent implements OnInit {
    30:   @Output() emitSelection = new EventEmitter()
    31:   @Input('allowEdit') public allowEdit: boolean = false
>>> 32:   @Input('addNewAddressDiv') public addNewAddressDiv: boolean = true
    33:   @Input('showNextButton') public showNextButton: boolean = false
    34:   public addressId: any = undefined
    35:   public displayedColumns = ['Name', 'Address', 'Country']
```

B	frontend/src/app/web3-sandbox web3-sandbox.component.ts **(Line 54)**	**typescript:S7655** Lifecycle interface 'OnInit' should be implemented for method 'ngOnInit'. (https://angular.dev/style-guide#style-09-01)

■ Problematic Code:

```
    51:     private readonly changeDetectorRef: ChangeDetectorRef
    52:   ) {}
    53:
>>> 54:   ngOnInit (): void {
    55:     this.handleAuth()
    56:     window.ethereum.on('chainChanged', this.handleChainChanged.bind(this))
    57:   }
```

B	frontend/src/app/wallet-web3 wallet-web3.component.ts **(Line 52)**	**typescript:S7655** Lifecycle interface 'OnInit' should be implemented for method 'ngOnInit'. (https://angular.dev/style-guide#style-09-01)

■ Problematic Code:

```
    49:   challengeSolved = false
    50:   errorMessage = ''
    51:   metamaskAddress = ''
>>> 52:   ngOnInit (): void {
    53:     this.handleAuth()
    54:     window.ethereum.on('chainChanged', this.handleChainChanged.bind(this))
    55:   }
```

B	frontend/src/app/faucet faucet.component.ts **(Line 64)**	**typescript:S7655** Lifecycle interface 'OnInit' should be implemented for method 'ngOnInit'. (https://angular.dev/style-guide#style-09-01)

■ Problematic Code:

```
      61:   errorMessage = ''
      62:   metamaskAddress = ''
      63:
>>>   64:   ngOnInit (): void {
      65:     this.translateService.get('NFT_MINT_TEXT_INTRO').subscribe((translatedString:
string) => {
      66:       this.nftMintText = translatedString
      67:     })
```

B	frontend/src/app/code-fixes code-fixes.component.ts **(Line 34)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
      31:   @Input('randomFixes')
      32:   public randomFixes: RandomFixes[] = []
      33:
>>>   34:   @Input('format')
      35:   public format: string = 'SideBySide'
      36:
      37:   @ViewChild('codeComponent', { static: false }) codeComponent: NgxTextDiffComponent
```

B	frontend/src/app/code-fixes code-fixes.component.ts **(Line 25)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
      22:   @Input('snippet')
      23:   public snippet: string = ''
      24:
>>>   25:   @Input('fixes')
      26:   public fixes: string[] = []
      27:
      28:   @Input('selectedFix')
```

B	frontend/src/app/code-fixes code-fixes.component.ts **(Line 22)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
      19:     this.differ = this.differs.find({}).create()
      20:   }
      21:
>>>   22:   @Input('snippet')
      23:   public snippet: string = ''
      24:
      25:   @Input('fixes')
```

B	frontend/src/app/code-area code-area.component.ts **(Line 32)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
     29:     this._code = value || ''
     30:   }
     31:
>>>  32:   @Input('vulnLines')
     33:   public vulnLines: number[]
     34:
     35:   public lineMarkers: LineMarker[]
```

B	frontend/src/app/purchase-basket purchase-basket.component.ts **(Line 30)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
     27: export class PurchaseBasketComponent implements OnInit {
     28:   @Input('allowEdit') public allowEdit: boolean = false
     29:   @Input('displayTotal') public displayTotal: boolean = false
>>>  30:   @Input('totalPrice') public totalPrice: boolean = true
     31:   @Output() emitTotal = new EventEmitter()
     32:   @Output() emitProductCount = new EventEmitter()
     33:   public tableColumns = ['image', 'product', 'quantity', 'price']
```

B	frontend/src/app/address address.component.ts **(Line 33)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
     30:   @Output() emitSelection = new EventEmitter()
     31:   @Input('allowEdit') public allowEdit: boolean = false
     32:   @Input('addNewAddressDiv') public addNewAddressDiv: boolean = true
>>>  33:   @Input('showNextButton') public showNextButton: boolean = false
     34:   public addressId: any = undefined
     35:   public displayedColumns = ['Name', 'Address', 'Country']
     36:   selection = new SelectionModel<Element>(false, [])
```

B	frontend/src/app/purchase-basket purchase-basket.component.ts **(Line 28)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ Problematic Code:

```
     25:   imports: [MatTable, MatColumnDef, MatHeaderCellDef, MatHeaderCell, MatCellDef,
MatCell, MatFooterCellDef, MatFooterCell, MatIconButton, MatHeaderRowDef, MatHeaderRow,
MatRowDef, MatRow, MatFooterRowDef, MatFooterRow, TranslateModule]
     26: })
     27: export class PurchaseBasketComponent implements OnInit {
>>>  28:   @Input('allowEdit') public allowEdit: boolean = false
     29:   @Input('displayTotal') public displayTotal: boolean = false
     30:   @Input('totalPrice') public totalPrice: boolean = true
     31:   @Output() emitTotal = new EventEmitter()
```

B	frontend/src/app/purchase-basket purchase-basket.component.ts **(Line 29)**	**typescript:S7649** Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ **Problematic Code:**

```
    26: })
    27: export class PurchaseBasketComponent implements OnInit {
    28:   @Input('allowEdit') public allowEdit: boolean = false
>>> 29:   @Input('displayTotal') public displayTotal: boolean = false
    30:   @Input('totalPrice') public totalPrice: boolean = true
    31:   @Output() emitTotal = new EventEmitter()
    32:   @Output() emitProductCount = new EventEmitter()
```

B

frontend/src/app/payment-method
payment-method.component.ts
(Line 34)

typescript:S7649
Input bindings should not be aliased (https://angular.dev/style-guide#style-05-13)

■ **Problematic Code:**

```
    31:
    32: export class PaymentMethodComponent implements OnInit {
    33:   @Output() emitSelection = new EventEmitter()
>>> 34:   @Input('allowDelete') public allowDelete: boolean = false
    35:   public displayedColumns = ['Number', 'Name', 'Expiry']
    36:   public nameControl: UntypedFormControl = new UntypedFormControl('',
[Validators.required])
    37:   // eslint-disable-next-line @typescript-eslint/no-loss-of-precision
```

H

frontend/src/app/search-result
search-result.component.ts
(Line 74)

typescript:S3776
Refactor this function to reduce its Cognitive Complexity from 18 to the 15 allowed.

■ **Problematic Code:**

```
    71:     const products = this.productService.search('')
    72:     const quantities = this.quantityService.getAll()
    73:     forkJoin([quantities, products]).subscribe({
>>> 74:       next: ([quantities, products]) => {
    75:         const dataTable: TableEntry[] = []
    76:         this.tableData = products
    77:         this.trustProductDescription(products) // vuln-code-snippet neutral-line
restfulXssChallenge
```

H

frontend/src/app/search-result
search-result.component.ts
(Line 209)

typescript:S2004
Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    206:                this.basketService.put(existingBasketItem.id, { quantity: newQuantity
}).subscribe({
    207:                  next: (updatedBasketItem) => {
    208:                    this.productService.get(updatedBasketItem.ProductId).subscribe({
>>> 209:                      next: (product) => {
    210:                        this.translateService.get('BASKET_ADD_SAME_PRODUCT', {
product: product.name }).subscribe({
    211:                          next: (basketAddSameProduct) => {
    212:                            this.snackBarHelperService.open(basketAddSameProduct,
'confirmBar')
```

H	frontend/src/app/search-result search-result.component.ts **(Line 221)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    218:                           }
    219:                         })
    220:                       },
>>> 221:                      error: (err) => { console.log(err) }
    222:                     })
    223:                   },
    224:                   error: (err) => {
```

H	frontend/src/app/search-result search-result.component.ts **(Line 241)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    238:               this.productService.get(newBasketItem.ProductId).subscribe({
    239:                 next: (product) => {
    240:                   this.translateService.get('BASKET_ADD_PRODUCT', { product: product.name }).subscribe({
>>> 241:                     next: (basketAddProduct) => {
    242:                       this.snackBarHelperService.open(basketAddProduct, 'confirmBar')
    243:                       this.basketService.updateNumberOfCartItems()
    244:                     },
```

H	frontend/src/app/search-result search-result.component.ts **(Line 245)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    242:                       this.snackBarHelperService.open(basketAddProduct, 'confirmBar')
    243:                       this.basketService.updateNumberOfCartItems()
    244:                     },
>>> 245:                     error: (translationId) => {
    246:                       this.snackBarHelperService.open(translationId, 'confirmBar')
    247:                       this.basketService.updateNumberOfCartItems()
    248:                     }
```

H	frontend/src/app server-started-notification server-started-notification.component.ts **(Line 42)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     39:           this.challengeService.restoreProgress(encodeURIComponent(continueCode)).subscribe({
     40:             next: () => {
     41:               this.translate.get('AUTO_RESTORED_PROGRESS').subscribe({
>>>  42:                 next: (notificationServerStarted) => {
     43:                   this.hackingProgress.autoRestoreMessage = notificationServerStarted
     44:                 },
     45:                 error: (translationId) => {
```

H	frontend/src/app server-started-notification server-started-notification.component.ts **(Line 45)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
     42:                 next: (notificationServerStarted) => {
     43:                   this.hackingProgress.autoRestoreMessage = notificationServerStarted
     44:                 },
>>>  45:                 error: (translationId) => {
     46:                   this.hackingProgress.autoRestoreMessage = translationId
     47:                 }
     48:               })
```

H	frontend/src/app server-started-notification server-started-notification.component.ts **(Line 53)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
     50:             error: (error) => {
     51:               console.log(error)
     52:               this.translate.get('AUTO_RESTORE_PROGRESS_FAILED', { error }).subscribe({
>>>  53:                 next: (notificationServerStarted) => {
     54:                   this.hackingProgress.autoRestoreMessage = notificationServerStarted
     55:                 },
     56:                 error: (translationId) => {
```

H	frontend/src/app server-started-notification server-started-notification.component.ts **(Line 56)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
     53:                 next: (notificationServerStarted) => {
     54:                   this.hackingProgress.autoRestoreMessage = notificationServerStarted
     55:                 },
>>>  56:                 error: (translationId) => {
     57:                   this.hackingProgress.autoRestoreMessage = translationId
     58:                 }
     59:               })
```

H	test/server/fileUploadSpec.ts **(Line 33)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     30:         challenges.uploadSizeChallenge = { solved: false, save } as unknown as
Challenge
     31:         req.file.size = size
     32:
>>>  33:         checkUploadSize(req, res, () => {})
     34:
     35:         expect(challenges.uploadSizeChallenge.solved).to.equal(false)
     36:       })
```

H | routes/dataExport.ts **(Line 88)**

typescript:S2004
Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     85:           likedBy: string
     86:         }>) => {
     87:           if (reviews.length > 0) {
>>>  88:             reviews.forEach(review => {
     89:               userData.reviews.push({
     90:                 message: review.message,
     91:                 author: review.author,
```

H | routes/dataExport.ts **(Line 100)**

typescript:S2004
Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     97:           }
     98:           const emailHash = security.hash(email).slice(0, 4)
     99:           for (const order of userData.orders) {
>>> 100:             challengeUtils.solveIf(challenges.dataExportChallenge, () => { return
order.orderId.split('-')[0] !== emailHash })
    101:           }
    102:           res.status(200).send({ userData: JSON.stringify(userData, null, 2),
confirmation: 'Your data export will open in a new Browser window.' })
    103:         },
```

H | routes/languages.ts **(Line 25)**

typescript:S2004
Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     22:         if (err != null) {
     23:           next(new Error(`Unable to read i18n directory: ${err.message}`))
     24:         }
>>>  25:         languageFiles.forEach((fileName) => {
     26:           // eslint-disable-next-line @typescript-eslint/no-misused-promises
     27:           fs.readFile('frontend/dist/frontend/assets/i18n/' + fileName, 'utf-8', async
(err, content) => {
     28:             if (err != null) {
```

H | routes/metrics.ts **(Line 170)**

typescript:S2004
Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    167:       }
    168:
    169:       void retrieveChallengesWithCodeSnippet().then(challenges => {
>>> 170:         ChallengeModel.count({ where: { codingChallengeStatus: { [Op.eq]: 1 } }
}).then((count: number) => {
    171:           codingChallengesProgressMetrics.set({ phase: 'find it' }, count)
    172:         }).catch(() => {
    173:           throw new Error('Unable to retrieve and count such challenges. Please try
again')
```

H	routes/metrics.ts **(Line 172)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    169:       void retrieveChallengesWithCodeSnippet().then(challenges => {
    170:         ChallengeModel.count({ where: { codingChallengeStatus: { [Op.eq]: 1 } }
}).then((count: number) => {
    171:           codingChallengesProgressMetrics.set({ phase: 'find it' }, count)
>>> 172:         }).catch(() => {
    173:           throw new Error('Unable to retrieve and count such challenges. Please try
again')
    174:         })
    175:
```

H	routes/metrics.ts **(Line 176)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    173:           throw new Error('Unable to retrieve and count such challenges. Please try
again')
    174:         })
    175:
>>> 176:         ChallengeModel.count({ where: { codingChallengeStatus: { [Op.eq]: 2 } }
}).then((count: number) => {
    177:           codingChallengesProgressMetrics.set({ phase: 'fix it' }, count)
    178:         }).catch((_: unknown) => {
    179:           throw new Error('Unable to retrieve and count such challenges. Please try
again')
```

H	routes/metrics.ts **(Line 178)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    175:
    176:         ChallengeModel.count({ where: { codingChallengeStatus: { [Op.eq]: 2 } }
}).then((count: number) => {
    177:           codingChallengesProgressMetrics.set({ phase: 'fix it' }, count)
>>> 178:         }).catch((_: unknown) => {
    179:           throw new Error('Unable to retrieve and count such challenges. Please try
again')
    180:         })
    181:
```

H	routes/metrics.ts **(Line 182)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
    179:           throw new Error('Unable to retrieve and count such challenges. Please try
again')
    180:         })
    181:
>>> 182:         ChallengeModel.count({ where: { codingChallengeStatus: { [Op.ne]: 0 } }
}).then((count: number) => {
    183:           codingChallengesProgressMetrics.set({ phase: 'unsolved' }, challenges.length
- count)
    184:         }).catch((_: unknown) => {
    185:           throw new Error('Unable to retrieve and count such challenges. Please try
again')
```

H	routes/metrics.ts **(Line 184)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
    181:
    182:         ChallengeModel.count({ where: { codingChallengeStatus: { [Op.ne]: 0 } }
}).then((count: number) => {
    183:           codingChallengesProgressMetrics.set({ phase: 'unsolved' }, challenges.length
- count)
>>> 184:         }).catch((_: unknown) => {
    185:           throw new Error('Unable to retrieve and count such challenges. Please try
again')
    186:         })
    187:       })
```

H	routes/order.ts **(Line 70)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
     67:           let totalPoints = 0
     68:           basket.Products?.forEach(({ BasketItem, price, deluxePrice, name, id }) => {
     69:             if (BasketItem != null) {
>>>  70:               challengeUtils.solveIf(challenges.christmasSpecialChallenge, () => {
return BasketItem.ProductId === products.christmasSpecial.id })
     71:               QuantityModel.findOne({ where: { ProductId: BasketItem.ProductId }
}).then((product: any) => {
     72:                 const newQuantity = product.quantity - BasketItem.quantity
     73:                 QuantityModel.update({ quantity: newQuantity }, { where: { ProductId:
BasketItem?.ProductId } }).catch((error: unknown) => {
```

H	routes/order.ts **(Line 71)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     68:           basket.Products?.forEach(({ BasketItem, price, deluxePrice, name, id }) => {
     69:             if (BasketItem != null) {
     70:               challengeUtils.solveIf(challenges.christmasSpecialChallenge, () => {
return BasketItem.ProductId === products.christmasSpecial.id })
>>>  71:               QuantityModel.findOne({ where: { ProductId: BasketItem.ProductId }
}).then((product: any) => {
     72:                 const newQuantity = product.quantity - BasketItem.quantity
     73:                 QuantityModel.update({ quantity: newQuantity }, { where: { ProductId:
BasketItem?.ProductId } }).catch((error: unknown) => {
     74:                   next(error)
```

H	routes/order.ts **(Line 76)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     73:                 QuantityModel.update({ quantity: newQuantity }, { where: { ProductId:
BasketItem?.ProductId } }).catch((error: unknown) => {
     74:                   next(error)
     75:                 })
>>>  76:               }).catch((error: unknown) => {
     77:                 next(error)
     78:               })
     79:               let itemPrice: number
```

H	routes/resetPassword.ts **(Line 37)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     34:       }).then((data: SecurityAnswerModel | null) => {
     35:         if ((data != null) && security.hmac(answer) === data.answer) {
     36:           UserModel.findByPk(data.UserId).then((user: UserModel | null) => {
>>>  37:             user?.update({ password: newPassword }).then((user: UserModel) => {
     38:               verifySecurityAnswerChallenges(user, answer)
     39:               res.json({ user })
     40:             }).catch((error: unknown) => {
```

H	routes/resetPassword.ts **(Line 40)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     37:             user?.update({ password: newPassword }).then((user: UserModel) => {
     38:               verifySecurityAnswerChallenges(user, answer)
     39:               res.json({ user })
>>>  40:             }).catch((error: unknown) => {
     41:               next(error)
     42:             })
     43:           }).catch((error: unknown) => {
```

H	routes/vulnCodeSnippet.ts **(Line 70)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 20 to the 15 allowed.

■ Problematic Code:

```
     67:   return notOkLines.length === 0
     68: }
     69:
>>>  70: export const checkVulnLines = () => async (req: Request<Record<string, unknown>,
Record<string, unknown>, VerdictRequestBody>, res: Response, next: NextFunction) => {
     71:   const key = req.body.key
     72:   let snippetData
     73:   try {
```

H	test/server/verifySpec.ts **(Line 216)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
    213:     describe('"changeProductChallenge"', () => {
    214:       beforeEach(() => {
    215:         challenges.changeProductChallenge = { solved: false, save } as unknown as
Challenge
>>> 216:         products.osaft = { reload () { return { then (cb: any) { cb() } } } } as
unknown as Product
    217:       })
    218:
    219:       it(`is solved when the link in the O-Saft product goes to
${config.get<string>('challenges.overwriteUrlForProductTamperingChallenge')}`, () => {
```

H	routes/nftMint.ts **(Line 19)**	**typescript:S3735** Remove this use of the "void" operator.

■ Problematic Code:

```
     16:       const provider = new
WebSocketProvider('wss://eth-sepolia.g.alchemy.com/v2/FZDapFZSs1l6yhHW4VnQqsi18qSd-3GJ')
     17:       const contract = new Contract(nftAddress, nftABI, provider)
     18:       if (!isEventListenerCreated) {
>>>  19:         void contract.on('NFTMinted', (minter: string) => {
     20:           if (!addressesMinted.has(minter)) {
     21:             addressesMinted.add(minter)
     22:           }
```

H	routes/web3Wallet.ts **(Line 21)**	**typescript:S3735** Remove this use of the "void" operator.

■ Problematic Code:

```
     18:       const provider = new
WebSocketProvider('wss://eth-sepolia.g.alchemy.com/v2/FZDapFZSs1l6yhHW4VnQqsi18qSd-3GJ')
     19:       const contract = new Contract(web3WalletAddress, web3WalletABI, provider)
     20:       if (!isEventListenerCreated) {
>>>  21:         void contract.on('ContractExploited', (exploiter: string) => {
     22:           if (walletsConnected.has(exploiter)) {
     23:             walletsConnected.delete(exploiter)
     24:             challengeUtils.solveIf(challenges.web3WalletChallenge, () => true)
```

H	routes/profileImageUrlUpload.ts **(Line 17)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 19 to the 15 allowed.

■ **Problematic Code:**

```
     14: import logger from '../lib/logger'
     15:
     16: export function profileImageUrlUpload () {
>>>  17:   return async (req: Request, res: Response, next: NextFunction) => {
     18:     if (req.body.imageUrl !== undefined) {
     19:       const url = req.body.imageUrl
     20:       if (url.match(/(.)*solve\/challenges\/server-side(.)*/) !== null)
req.app.locals.abused_ssrf_bug = true
```

H	test/server/blueprintSpec.ts **(Line 35)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 23 to the 15 allowed.

■ **Problematic Code:**

```
     32:   let pathToImage: string = 'assets/public/images/products/'
     33:
     34:   describe('checkExifData', () => {
>>>  35:     it('should contain properties from exifForBlueprintChallenge', async () => {
     36:       for (const product of products) {
     37:         if (product.fileForRetrieveBlueprintChallenge && product.image) {
     38:           if (utils.isUrl(product.image)) {
```

H	rsn/rsnUtil.ts **(Line 47)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 31 to the 15 allowed.

■ **Problematic Code:**

```
     44:   }, {})
     45:   for (const val of keys) {
     46:     await retrieveCodeSnippet(val.split('_')[0])
>>>  47:       .then(snippet => {
     48:         if (snippet == null) return
     49:         process.stdout.write(val + ': ')
     50:         const fileData = fs.readFileSync(fixesPath + '/' + val).toString()
```

H	routes/fileUpload.ts **(Line 108)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 18 to the 15 allowed.

■ **Problematic Code:**

```
    105:   next()
    106: }
    107:
>>> 108: function handleYamlUpload ({ file }: Request, res: Response, next: NextFunction) {
    109:   if (utils.endsWith(file?.originalname.toLowerCase(), '.yml') ||
utils.endsWith(file?.originalname.toLowerCase(), '.yaml')) {
    110:     challengeUtils.solveIf(challenges.deprecatedInterfaceChallenge, () => { return true
})
    111:     if (((file?.buffer) != null) &&
utils.isChallengeEnabled(challenges.deprecatedInterfaceChallenge)) {
```

H	test/cypress/e2e/chatbot.spec.ts **(Line 45)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     42:           cy.get('#message-input').type('hi').type('{enter}')
     43:           cy.get('#message-input').type('...').type('{enter}')
     44:
>>>  45:           const genArr = Array.from({ length: 100 }, (v, k) => k + 1)
     46:           cy.wrap(genArr).eachSeries(() => {
     47:             cy.get('#message-input')
     48:               .type(couponIntent.utterances[0])
```

H	frontend/src/app/about about.component.ts **(Line 76)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 17 to the 15 allowed.

■ **Problematic Code:**

```
     73:           console.error(err)
     74:           return EMPTY
     75:         })
>>>  76:       ).subscribe((config) => {
     77:         if (config?.application?.social) {
     78:           if (config.application.social.blueSkyUrl) {
     79:             this.blueSkyUrl = config.application.social.blueSkyUrl
```

H	test/cypress/e2e/geoStalking.spec.ts **(Line 11)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
      8:    it('Should be possible to find the answer to a security question in the meta-data
of a photo on the photo wall', () => {
      9:      cy.task<string>('GetFromMemories', 'geoStalkingMetaSecurityAnswer').then(
     10:        (answer: string) => {
>>>  11:          cy.task<string>('GetFromConfig', 'application.domain').then((appDomain:
string) => {
     12:            cy.get('#email').type(`john@${appDomain}`)
     13:            cy.wait('@securityQuestion')
     14:            cy.get('#securityAnswer').should('not.be.disabled').focus().type(answer)
```

H	test/cypress/e2e/geoStalking.spec.ts **(Line 30)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     27:    it('Should be possible to determine the answer to a security question by looking
closely at an image on the photo wall', () => {
     28:      cy.task<string>('GetFromMemories', 'geoStalkingVisualSecurityAnswer').then(
     29:        (answer: string) => {
>>>  30:          cy.task<string>('GetFromConfig', 'application.domain').then((appDomain:
string) => {
     31:            cy.get('#email').type(`emma@${appDomain}`)
     32:            cy.wait('@securityQuestion')
     33:            cy.get('#securityAnswer').should('not.be.disabled').focus().type(answer)
```

H	test/cypress/e2e/restApi.spec.ts **(Line 51)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     48:   describe('challenge "changeProduct"', () => {
     49:     it('should be possible to change product via PUT request without being logged in',
() => {
     50:       cy.task<number>('GetTamperingProductId').then((tamperingProductId: number) => {
>>>  51:         cy.task<string>('GetOverwriteUrl').then((overwriteUrl: string) => {
     52:           cy.window().then(async () => {
     53:             const response = await fetch(
     54:               `${Cypress.config('baseUrl')}/api/Products/${tamperingProductId}`,
```

H	test/cypress/e2e/b2bOrder.spec.ts **(Line 8)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
      5:         if (!isDocker) {
      6:           cy.login({ email: 'admin', password: 'admin123' })
      7:
>>>   8:           cy.window().then(async () => {
      9:             const response = await fetch(
     10:               `${Cypress.config('baseUrl')}/b2b/v2/orders/`,
     11:               {
```

H	test/cypress/e2e/b2bOrder.spec.ts **(Line 38)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     35:       cy.task('isDocker').then((isDocker) => {
     36:         if (!isDocker) {
     37:           cy.login({ email: 'admin', password: 'admin123' })
>>>  38:           cy.window().then(async () => {
     39:             const response = await fetch(
     40:               `${Cypress.config('baseUrl')}/b2b/v2/orders/`,
     41:               {
```

H	test/cypress/e2e/complain.spec.ts **(Line 158)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    155:           cy.get('#submitButton').click()
    156:           cy.visit('/promotion')
    157:
>>> 158:           cy.on('window:alert', (t) => {
    159:             expect(t).to.equal('xss')
    160:           })
    161:           cy.visit('/')
```

H	test/cypress/e2e/contact.spec.ts **(Line 64)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    61:     cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
    62:
    63:     cy.visit('/#/about')
>>> 64:     cy.on('window:alert', (t) => {
    65:       expect(t).to.equal('xss')
    66:     })
    67:
```

H	test/cypress/e2e/contact.spec.ts **(Line 69)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    66:     })
    67:
    68:     cy.visit('/#/administration')
>>> 69:     cy.on('window:alert', (t) => {
    70:       expect(t).to.equal('xss')
    71:     })
    72:     cy.expectChallengeSolved({ challenge: 'Server-side XSS Protection' })
```

H	test/cypress/e2e/forgedJwt.spec.ts **(Line 19)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    16:  it('should accept a token HMAC-signed with public RSA key with email
rsa_lord@juice-sh.op in the payload ', () => {
    17:   cy.task('isWindows').then((isWindows) => {
    18:    if (!isWindows) {
>>> 19:     cy.window().then(() => {
    20:      localStorage.setItem(
    21:       'token',
    22:       'eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJkYXRhIjp7ImVtYWlsIjoicnNhX2xvcm
RAanVpY2Utc2gub3AifSwiaWF0IjoxNTgzMDM3NzExfQ.gShXDT5TrE5736mpIbfVDEcQbLfteJaQUG7Z0PH8Xc8'
```

H	test/cypress/e2e/noSql.spec.ts **(Line 13)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    10:  it('should be possible to inject a command into the get route', () => {
    11:   cy.task('isDocker').then((isDocker) => {
    12:    if (!isDocker) {
>>> 13:     cy.window().then(() => {
    14:      void fetch(
    15:       `${Cypress.config('baseUrl')}/rest/products/sleep(1000)/reviews`,
    16:       {
```

H	test/cypress/e2e/noSql.spec.ts **(Line 34)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     31:     it('should be possible to inject and get all the orders', () => {
     32:       cy.task('isDocker').then((isDocker) => {
     33:         if (!isDocker) {
>>>  34:           cy.window().then(async () => {
     35:             await fetch(
     36:               `${Cypress.config('baseUrl')}/rest/track-order/%27%20%7C%7C%20true%20%7C
%7C%20%27`,
     37:               {
```

H	test/cypress/e2e/profile.spec.ts **(Line 30)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     27:           cy.get('#username').type('<<a|ascript>alert(`xss`)</script>')
     28:           cy.get('#submit').click()
     29:
>>>  30:           cy.on('window:alert', (t) => {
     31:             expect(t).to.equal('xss')
     32:           })
     33:
```

H	test/cypress/e2e/register.spec.ts **(Line 17)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     14:     it('should be possible to bypass validation by directly using Rest API', async ()
=> {
     15:       cy.task('isDocker').then((isDocker) => {
     16:         if (!isDocker) {
>>>  17:           cy.window().then(async () => {
     18:             const response = await fetch(
     19:               `${Cypress.config('baseUrl')}/api/Users/`,
     20:               {
```

H	test/cypress/e2e/register.spec.ts **(Line 40)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     37:           })
     38:
     39:           cy.visit('/#/administration')
>>>  40:           cy.on('window:alert', (t) => {
     41:             expect(t).to.equal('xss')
     42:           })
     43:           cy.expectChallengeSolved({ challenge: 'Client-side XSS Protection' })
```

H	test/cypress/e2e/restApi.spec.ts **(Line 12)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
      9:     xit('should be possible to create a new product when logged in', () => {
     10:       cy.task('isDocker').then((isDocker) => {
     11:         if (!isDocker) {
>>>  12:           cy.window().then(async () => {
     13:             const response = await fetch(
     14:               `${Cypress.config('baseUrl')}/api/Products`,
     15:               {
```

H	test/cypress/e2e/restApi.spec.ts **(Line 38)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     35:           cy.reload()
     36:           cy.get('img[alt="RestXSS"]').click()
     37:
>>>  38:           cy.on('window:alert', (t) => {
     39:             expect(t).to.equal('xss')
     40:           })
     41:
```

H	test/cypress/e2e/restApi.spec.ts **(Line 89)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     86:     it('should be possible to save log-in IP when logged in', () => {
     87:       cy.task('isDocker').then((isDocker) => {
     88:         if (!isDocker) {
>>>  89:           cy.window().then(async () => {
     90:             const response = await fetch(
     91:               `${Cypress.config('baseUrl')}/rest/saveLoginIp`,
     92:               {
```

H	test/cypress/e2e/trackOrder.spec.ts **(Line 7)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
      4:     xit('Order Id should be susceptible to reflected XSS attacks', () => {
      5:       cy.task('isDocker').then((isDocker) => {
      6:         if (!isDocker) {
>>>   7:           cy.on('uncaught:exception', (_err, _runnable) => {
      8:             return false
      9:           })
     10:
```

H	test/cypress/e2e/trackOrder.spec.ts **(Line 15)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     12:          cy.visit('/#/track-result?id=<iframe src="javascript:alert(`xss`)">')
     13:          cy.reload()
     14:
>>>  15:          cy.on('window:alert', (t) => {
     16:            expect(t).to.equal('xss')
     17:          })
     18:
```

H	routes/fileUpload.ts **(Line 75)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 18 to the 15 allowed.

■ **Problematic Code:**

```
     72:   next()
     73: }
     74:
>>>  75: function handleXmlUpload ({ file }: Request, res: Response, next: NextFunction) {
     76:   if (utils.endsWith(file?.originalname.toLowerCase(), '.xml')) {
     77:     challengeUtils.solveIf(challenges.deprecatedInterfaceChallenge, () => { return true
})
     78:     if (((file?.buffer) != null) &&
utils.isChallengeEnabled(challenges.deprecatedInterfaceChallenge)) { // XXE attacks in
Docker/Heroku containers regularly cause "segfault" crashes
```

H	routes/search.ts **(Line 47)**	**typescript:S3735** Remove this use of the "void" operator.

■ **Problematic Code:**

```
     44:        }
     45:        if (challengeUtils.notSolved(challenges.dbSchemaChallenge)) {
     46:          let solved = true
>>>  47:          void models.sequelize.query('SELECT sql FROM sqlite_master').then(([data]:
any) => {
     48:            const tableDefinitions = utils.queryResultToJson(data)
     49:            if (tableDefinitions.data?.length) {
     50:              for (let i = 0; i < tableDefinitions.data.length; i++) {
```

H	test/cypress/e2e/scoreBoard.spec.ts **(Line 45)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     42:       )
     43:
     44:       cy.get('@showSolvedNotifications').then((showSolvedNotifications) => {
>>>  45:         cy.get('@showFlagsInNotifications').then((showFlagsInNotifications) => {
     46:           if (showSolvedNotifications && showFlagsInNotifications) {
     47:             cy.get('.challenge-solved-toast').then((arrayOfSolvedToasts) => {
     48:               const alertsBefore = Cypress.$(arrayOfSolvedToasts).length
```

H	test/cypress/e2e/basket.spec.ts **(Line 80)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     77:     })
     78:     describe('challenge "manipulateClock"', () => {
     79:       it('should be possible to enter WMNSDY2019 coupon & place order with this expired
coupon', () => {
>>>  80:         cy.window().then(() => {
     81:           window.localStorage.couponPanelExpanded = false
     82:         })
     83:         cy.visit('/#/payment/shop')
```

H	test/cypress/e2e/basket.spec.ts **(Line 85)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     82:         })
     83:         cy.visit('/#/payment/shop')
     84:
>>>  85:         cy.window().then((win) => {
     86:           cy.on('uncaught:exception', (_err, _runnable) => {
     87:             // Introduced to disable the uncaught:exception we get after the eval under
this as TypeError: Date.now is not a function
     88:             return false
```

H	test/cypress/e2e/basket.spec.ts **(Line 141)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    138:
    139:         cy.visit('/#/payment/shop')
    140:         cy.get('#collapseCouponElement').click()
>>> 141:         cy.task<string>('GenerateCoupon', 90).then((coupon: string) => {
    142:           cy.get('#coupon').type(coupon)
    143:           cy.get('#applyCouponButton').click()
    144:         })
```

H	test/cypress/e2e/search.spec.ts **(Line 64)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     61:       cy.request("/rest/products/search?q='))--")
     62:         .its('body')
     63:         .then((sourceContent) => {
>>>  64:           cy.task<Product>('GetPastebinLeakProduct').then((pastebinLeakProduct:
Product) => {
     65:             let foundProduct = false
     66:
     67:             sourceContent.data.forEach((product: Product) => {
```

H	test/cypress/e2e/search.spec.ts **(Line 91)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     88:       cy.request("/rest/products/search?q='))--")
     89:         .its('body')
     90:         .then((sourceContent) => {
>>>  91:           cy.task<Product>('GetChristmasProduct').then((christmasProduct: Product) => {
     92:             let foundProduct = false
     93:
     94:             sourceContent.data.forEach((product: Product) => {
```

H	test/cypress/e2e/search.spec.ts **(Line 109)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    106:       cy.request('/api/products')
    107:         .its('body')
    108:         .then((sourceContent) => {
>>> 109:           cy.task<Product>('GetChristmasProduct').then((christmasProduct: Product) => {
    110:             sourceContent.data.forEach((product: Product) => {
    111:               if (product.name === christmasProduct.name) {
    112:                 cy.window().then(async () => {
```

H	test/cypress/e2e/totpSetup.spec.ts **(Line 33)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     30:           // console.log($val);
     31:           cy.get('#currentPasswordSetup').type('K1f.....................')
     32:
>>>  33:           cy.task<string>('GenerateAuthenticator', $val).then((secret: string) => {
     34:             cy.get('#initialToken').type(secret)
     35:             cy.get('#setupTwoFactorAuth').click()
     36:
```

H	routes/metrics.ts **(Line 209)**	**typescript:S3735** Remove this use of the "void" operator.

■ **Problematic Code:**

```
    206:         if (count) userMetrics.set({ type: 'deluxe' }, count)
    207:       })
    208:
>>> 209:       void UserModel.count().then((count: number) => {
    210:         if (count) userTotalMetrics.set(count)
    211:       })
    212:
```

H	routes/metrics.ts **(Line 213)**	**typescript:S3735** Remove this use of the "void" operator.

■ **Problematic Code:**

```
    210:         if (count) userTotalMetrics.set(count)
    211:       })
    212:
>>> 213:       void WalletModel.sum('balance').then((totalBalance: number) => {
    214:         if (totalBalance) walletMetrics.set(totalBalance)
    215:       })
    216:
```

H	routes/chatbot.ts **(Line 50)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 18 to the 15 allowed.

■ **Problematic Code:**

```
     47:
     48: void initializeChatbot()
     49:
>>>  50: async function processQuery (user: User, req: Request, res: Response, next:
NextFunction) {
     51:   if (bot == null) {
     52:     res.status(503).send()
     53:     return
```

H	routes/order.ts **(Line 37)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 34 to the 15 allowed.

■ **Problematic Code:**

```
     34:   return (req: Request, res: Response, next: NextFunction) => {
     35:     const id = req.params.id
     36:     BasketModel.findOne({ where: { id }, include: [{ model: ProductModel, paranoid:
false, as: 'Products' }] })
>>>  37:       .then(async (basket: BasketModel | null) => {
     38:         if (basket != null) {
     39:           const customer = security.authenticatedUsers.from(req)
     40:           const email = customer ? customer.data ? customer.data.email : '' : ''
```

H	frontend/src/app/web3-sandbox web3-sandbox.component.ts **(Line 190)**	**typescript:S3776** Refactor this function to reduce its Cognitive Complexity from 16 to the 15 allowed.

■ **Problematic Code:**

```
    187:     }
    188:   }
    189:
>>> 190:   async invokeFunction (func) {
    191:     if (!this.session) {
    192:       this.snackBarHelperService.open('PLEASE_CONNECT_WEB3_WALLET', 'errorBar')
    193:       return
```

H	test/cypress/e2e/chatbot.spec.ts **(Line 46)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     43:           cy.get('#message-input').type('...').type('{enter}')
     44:
     45:           const genArr = Array.from({ length: 100 }, (v, k) => k + 1)
>>>  46:           cy.wrap(genArr).eachSeries(() => {
     47:             cy.get('#message-input')
     48:               .type(couponIntent.utterances[0])
     49:               .type('{enter}')
```

H	test/cypress/e2e/basket.spec.ts **(Line 9)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
      6:
      7:     describe('challenge "negativeOrder"', () => {
      8:       it('should be possible to update a basket to a negative quantity via the Rest
API', () => {
>>>   9:         cy.window().then(async () => {
     10:           const response = await fetch(
     11:             `${Cypress.config('baseUrl')}/api/BasketItems/1`,
     12:             {
```

H	test/cypress/e2e/basket.spec.ts **(Line 30)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     27:
     28:         cy.get('mat-cell.mat-column-quantity > span')
     29:           .first()
>>>  30:           .then(($ele) => {
     31:             const quantity = $ele.text()
     32:             expect(quantity).to.match(/-100000/)
     33:           })
```

H	test/cypress/e2e/basket.spec.ts **(Line 45)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     42:
     43:     describe('challenge "basketAccessChallenge"', () => {
     44:       it('should access basket with id from session storage instead of the one
associated to logged-in user', () => {
>>>  45:         cy.window().then(() => {
     46:           window.sessionStorage.bid = 3
     47:         })
     48:
```

H	test/cypress/e2e/basket.spec.ts **(Line 58)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     55:
     56:     describe('challenge "basketManipulateChallenge"', () => {
     57:       it('should manipulate basket of other user instead of the one associated to
logged-in user', () => {
>>>  58:         cy.window().then(async () => {
     59:           await fetch(`${Cypress.config('baseUrl')}/api/BasketItems/`, {
     60:             method: 'POST',
     61:             cache: 'no-cache',
```

H	test/cypress/e2e/basket.spec.ts **(Line 111)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    108:       })
    109:
    110:       it('should be possible to add a product in the basket', () => {
>>> 111:         cy.window().then(async () => {
    112:           const response = await fetch(
    113:             `${Cypress.config('baseUrl')}/api/BasketItems/`,
    114:             {
```

H	test/cypress/e2e/basket.spec.ts **(Line 135)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    132:       })
    133:
    134:       it('should be possible to enter a coupon that gives an 80% discount', () => {
>>> 135:         cy.window().then(() => {
    136:           window.localStorage.couponPanelExpanded = false
    137:         })
    138:
```

H	test/cypress/e2e/contact.spec.ts **(Line 154)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    151:           await sendPostRequest(responseJson)
    152:         }
    153:
>>> 154:         async function sendPostRequest (captcha: {
    155:           captchaId: number
    156:           answer: string
    157:         }) {
```

H	test/cypress/e2e/contact.spec.ts **(Line 202)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
    199:       await sendPostRequest(responseJson)
    200:      }
    201:
>>> 202:      async function sendPostRequest (captcha: {
    203:       captchaId: number
    204:       answer: string
    205:      }) {
```

H	test/cypress/e2e/deluxe.spec.ts **(Line 29)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
     26:      url: '/rest/deluxe-membership',
     27:      method: 'POST',
     28:      headers: { Authorization: `Bearer ${token?.value}` }
>>>  29:     }).then((response) => {
     30:      expect(response.body.status).contains('success')
     31:     })
     32:    })
```

H	test/cypress/e2e/forgotPassword.spec.ts **(Line 54)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ Problematic Code:

```
     51:   describe('for his internal account', () => {
     52:    it('should be able to reset password with his security answer', () => {
     53:     cy.task<string>('GetFromConfig', 'application.domain').then(
>>>  54:      (appDomain: string) => {
     55:       cy.get('#email').type(`bjoern@${appDomain}`)
     56:      }
     57:     )
```

H	test/cypress/e2e/login.spec.ts **(Line 128)**	**typescript:S3735** Remove this use of the "void" operator.

■ Problematic Code:

```
    125:
    126:    cy.task<string>('GenerateAuthenticator',
'IFTXE3SPOEYVURT2MRYGI52TKJ4HC3KH').then(
    127:     (totpToken: string) => {
>>> 128:      void cy.get('#totpToken').type(totpToken)
    129:      void cy.get('#totpSubmitButton').click()
    130:     }
    131:    )
```

H	test/cypress/e2e/login.spec.ts **(Line 129)**	**typescript:S3735** Remove this use of the "void" operator.

■ **Problematic Code:**

```
    126:       cy.task<string>('GenerateAuthenticator', 'IFTXE3SPOEYVURT2MRYGI52TKJ4HC3KH').then(
    127:         (totpToken: string) => {
    128:           void cy.get('#totpToken').type(totpToken)
>>> 129:           void cy.get('#totpSubmitButton').click()
    130:         }
    131:       )
    132:       cy.expectChallengeSolved({ challenge: 'Two Factor Authentication' })
```

H	test/cypress/e2e/noSql.spec.ts **(Line 14)**	**typescript:S3735** Remove this use of the "void" operator.

■ **Problematic Code:**

```
     11:       cy.task('isDocker').then((isDocker) => {
     12:         if (!isDocker) {
     13:           cy.window().then(() => {
>>>  14:             void fetch(
     15:               `${Cypress.config('baseUrl')}/rest/products/sleep(1000)/reviews`,
     16:               {
     17:                 method: 'GET',
```

H	test/cypress/e2e/noSql.spec.ts **(Line 97)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     94:           await editReview(reviewId)
     95:         }
     96:
>>>  97:         async function editReview (reviewId: string) {
     98:           const response = await fetch(
     99:             `${Cypress.config('baseUrl')}/rest/products/reviews`,
    100:             {
```

H	test/cypress/e2e/noSql.spec.ts **(Line 126)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    123:     it('should be possible to like reviews multiple times', () => {
    124:       cy.visit('/')
    125:       cy.window().then(async () => {
>>> 126:         async function sendPostRequest (reviewId: string) {
    127:           const anotherResponse = await fetch(
    128:             `${Cypress.config('baseUrl')}/rest/products/reviews`,
    129:             {
```

H	data/datacreator.ts **(Line 406)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    403:                   product: id,
    404:                   likesCount: 0,
    405:                   likedBy: []
>>> 406:                 }).catch((err: unknown) => {
    407:                   logger.error(`Could not insert Product Review ${text}:
${utils.getErrorMessage(err)}`)
    408:                 })
    409:               )
```

H	routes/metrics.ts **(Line 201)**	**typescript:S3735** Remove this use of the "void" operator.

■ **Problematic Code:**

```
    198:         if (reviewCount) interactionsMetrics.set({ type: 'review' }, reviewCount)
    199:       })
    200:
>>> 201:       void UserModel.count({ where: { role: { [Op.eq]: 'customer' } } }).then((count:
number) => {
    202:         if (count) userMetrics.set({ type: 'standard' }, count)
    203:       })
    204:
```

H	routes/metrics.ts **(Line 205)**	**typescript:S3735** Remove this use of the "void" operator.

■ **Problematic Code:**

```
    202:         if (count) userMetrics.set({ type: 'standard' }, count)
    203:       })
    204:
>>> 205:       void UserModel.count({ where: { role: { [Op.eq]: 'deluxe' } } }).then((count:
number) => {
    206:         if (count) userMetrics.set({ type: 'deluxe' }, count)
    207:       })
    208:
```

H	routes/metrics.ts **(Line 217)**	**typescript:S3735** Remove this use of the "void" operator.

■ **Problematic Code:**

```
    214:         if (totalBalance) walletMetrics.set(totalBalance)
    215:       })
    216:
>>> 217:       void FeedbackModel.count().then((count: number) => {
    218:         if (count) interactionsMetrics.set({ type: 'feedback' }, count)
    219:       })
    220:
```

H	routes/metrics.ts **(Line 221)**	**typescript:S3735** Remove this use of the "void" operator.

■ **Problematic Code:**

```
    218:         if (count) interactionsMetrics.set({ type: 'feedback' }, count)
    219:       })
    220:
>>> 221:       void ComplaintModel.count().then((count: number) => {
    222:         if (count) interactionsMetrics.set({ type: 'complaint' }, count)
    223:       })
    224:     } catch (e: unknown) {
```

H | routes/fileUpload.ts **(Line 49)**

typescript:S2004
Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     46:                 } else {
     47:                   entry.autodrain()
     48:                 }
>>>  49:               }).on('error', function (err: unknown) { next(err) })
     50:           })
     51:         })
     52:       })
```

H | routes/order.ts **(Line 48)**

typescript:S3735
Remove this use of the "void" operator.

■ **Problematic Code:**

```
     45:           const fileWriter = doc.pipe(fs.createWriteStream(path.join('ftp/', pdfFile)))
     46:
     47:           fileWriter.on('finish', async () => {
>>>  48:             void basket.update({ coupon: null })
     49:             await BasketItemModel.destroy({ where: { BasketId: id } })
     50:             res.json({ orderConfirmation: orderId })
     51:           })
```

H | routes/verify.ts **(Line 175)**

typescript:S3735
Remove this use of the "void" operator.

■ **Problematic Code:**

```
    172:
    173: function changeProductChallenge (osaft: Product) {
    174:   let urlForProductTamperingChallenge: string | null = null
>>> 175:   void osaft.reload().then(() => {
    176:     for (const product of config.get<ProductConfig[]>('products')) {
    177:       if (product.urlForProductTamperingChallenge !== undefined) {
    178:         urlForProductTamperingChallenge = product.urlForProductTamperingChallenge
```

H | routes/fileUpload.ts **(Line 40)**

typescript:S2004
Refactor this code to not nest functions more than 4 levels deep.

Problematic Code:

```
     37:           fs.close(fd, function () {
     38:             fs.createReadStream(tempFile)
     39:               .pipe(unzipper.Parse())
>>>  40:               .on('entry', function (entry: any) {
     41:                 const fileName = entry.path
     42:                 const absolutePath = path.resolve('uploads/complaints/' + fileName)
     43:                 challengeUtils.solveIf(challenges.fileWriteChallenge, () => { return
absolutePath === path.resolve('ftp/legal.md') })
```

H	test/server/verifySpec.ts **(Line 168)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

Problematic Code:

```
    165:     describe('is solved when an error occurs on a response with error', () => {
    166:       const httpStatus = [402, 403, 404, 500]
    167:       httpStatus.forEach(statusCode => {
>>> 168:         it(`${statusCode} status code`, () => {
    169:           res.statusCode = statusCode
    170:           err = new Error()
    171:
```

H	test/server/verifySpec.ts **(Line 191)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

Problematic Code:

```
    188:     describe('is not solved when no error occurs on a response with error', () => {
    189:       const httpStatus = [401, 402, 404, 500]
    190:       httpStatus.forEach(statusCode => {
>>> 191:         it(`${statusCode} status code`, () => {
    192:           res.statusCode = statusCode
    193:           err = undefined
    194:
```

H	data/static/codefixes adminSectionChallenge_3.ts **(Line 3)**	**typescript:S3504** Unexpected var, use let or const instead.

Problematic Code:

```
      1: const routes: Routes = [
      2:   {
>>>   3:     path: (function(){var t=Array.prototype.slice.call(arguments),G=t.shift();return
t.reverse().map(function(e,w){return String.fromCharCode(e-G-2-w)}).join('')})(55,167,171,165,1
68,158,154)+(62749278960).toString(36).toLowerCase()+(function(){var
b=Array.prototype.slice.call(arguments),V=b.shift();return b.reverse().map(function(l,S){return
String.fromCharCode(l-V-43-S)}).join('')})(58,211),
      4:     component: AdministrationComponent,
      5:     canActivate: [AdminGuard]
      6:   },
```

H	data/static/codefixes adminSectionChallenge_3.ts **(Line 3)**	**typescript:S3504** Unexpected var, use let or const instead.

■ **Problematic Code:**

```
      1: const routes: Routes = [
      2:   {
>>>   3:     path: (function(){var t=Array.prototype.slice.call(arguments),G=t.shift();return
t.reverse().map(function(e,w){return String.fromCharCode(e-G-2-w)}).join('')})(55,167,171,165,1
68,158,154)+(62749278960).toString(36).toLowerCase()+(function(){var
b=Array.prototype.slice.call(arguments),V=b.shift();return b.reverse().map(function(l,S){return
String.fromCharCode(l-V-43-S)}).join('')})(58,211),
      4:     component: AdministrationComponent,
      5:     canActivate: [AdminGuard]
      6:   },
```

H	test/api/dataExportApiSpec.ts **(Line 218)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    215:                .expect('status', 200)
    216:                .expect('header', 'content-type', /application\/json/)
    217:                .expect('json', 'confirmation', 'Your data export will open in a new
Browser window.')
>>> 218:                .then(({ json }) => {
    219:                  const parsedData = JSON.parse(json.userData)
    220:                  expect(parsedData.username).toBe('')
    221:                  expect(parsedData.email).toBe('jim@' +
config.get<string>('application.domain'))
```

H	test/api/dataExportApiSpec.ts **(Line 352)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    349:              })
    350:                .expect('status', 200)
    351:                .expect('header', 'content-type', /application\/json/)
>>> 352:                .then(({ json: captchaAnswer }) => {
    353:                  return frisby.post(REST_URL + '/user/data-export', {
    354:                    headers: { Authorization: 'Bearer ' + jsonLogin.authentication.token,
'content-type': 'application/json' },
    355:                    body: {
```

H	test/api/dataExportApiSpec.ts **(Line 98)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
     95:                .expect('status', 200)
     96:                .expect('header', 'content-type', /application\/json/)
     97:                .expect('json', 'confirmation', 'Your data export will open in a new
Browser window.')
>>>  98:                .then(({ json }) => {
     99:                  const parsedData = JSON.parse(json.userData)
    100:                  expect(parsedData.username).toBe('bkimminich')
    101:                  expect(parsedData.email).toBe('bjoern.kimminich@gmail.com')
```

H	test/api/dataExportApiSpec.ts **(Line 131)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    128:                .expect('status', 200)
    129:                .expect('header', 'content-type', /application\/json/)
    130:                .expect('json', 'confirmation', 'Your data export will open in a new
Browser window.')
>>> 131:                .then(({ json }) => {
    132:                  const parsedData = JSON.parse(json.userData)
    133:                  expect(parsedData.username).toBe('')
    134:                  expect(parsedData.email).toBe('amy@' +
config.get<string>('application.domain'))
```

H	test/api/dataExportApiSpec.ts **(Line 249)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    246:              })
    247:                .expect('status', 200)
    248:                .expect('header', 'content-type', /application\/json/)
>>> 249:                .then(({ json: captchaAnswer }) => {
    250:                  return frisby.post(REST_URL + '/user/data-export', {
    251:                    headers: { Authorization: 'Bearer ' + jsonLogin.authentication.token,
'content-type': 'application/json' },
    252:                    body: {
```

H	test/api/dataExportApiSpec.ts **(Line 303)**	**typescript:S2004** Refactor this code to not nest functions more than 4 levels deep.

■ **Problematic Code:**

```
    300:                .expect('status', 200)
    301:                .expect('header', 'content-type', /application\/json/)
    302:                .expect('json', 'confirmation', 'Your data export will open in a new
Browser window.')
>>> 303:                .then(({ json }) => {
    304:                  const parsedData = JSON.parse(json.userData)
    305:                  expect(parsedData.username).toBe('')
    306:                  expect(parsedData.email).toBe('jim@' +
config.get<string>('application.domain'))
```

M	frontend/src/app/address address.component.scss **(Line 72)**	**css:S4666** Unexpected duplicate selector ".mat-column-Address", first used at line 63

■ **Problematic Code:**

```
     69:   width: 48px;
     70: }
     71:
>>>  72: .mat-column-Address {
     73:   word-break: break-word;
     74: }
     75:
```

M	frontend/src/app/token-sale token-sale.component.scss **(Line 117)**	**css:S4666** Unexpected duplicate selector ".mdc-card", first used at line 79

■ **Problematic Code:**

```
     114:   width: 10px;
     115: }
     116:
>>> 117: .mdc-card { border: 0; }
     118:
     119: .offer-container { display: flex;
     120:   flex-direction: column;
```

M	frontend/src/app/token-sale token-sale.component.scss **(Line 119)**	**css:S4666** Unexpected duplicate selector ".offer-container", first used at line 94

■ **Problematic Code:**

```
     116:
     117: .mdc-card { border: 0; }
     118:
>>> 119: .offer-container { display: flex;
     120:   flex-direction: column;
     121:   gap: 16px; }
     122:
```

M	frontend/src/app/token-sale token-sale.component.scss **(Line 123)**	**css:S4666** Unexpected duplicate selector ".faq-container", first used at line 46

■ **Problematic Code:**

```
     120:   flex-direction: column;
     121:   gap: 16px; }
     122:
>>> 123: .faq-container { display: flex;
     124:   flex-direction: column;
     125:   gap: 4px;
     126:   margin-top: -7px; }
```

M	Dockerfile **(Line 22)**	**docker:S6596** Use a specific version tag for the image.

■ **Problematic Code:**

```
      19: RUN npm install -g @cyclonedx/cyclonedx-npm@$CYCLONEDX_NPM_VERSION
      20: RUN npm run sbom
      21:
>>>   22: FROM gcr.io/distroless/nodejs22-debian12
      23: ARG BUILD_DATE
      24: ARG VCS_REF
      25: LABEL maintainer="Bjoern Kimminich <bjoern.kimminich@owasp.org>" \
```

M	server.ts **(Line 19)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    16: import colors from 'colors/safe'
    17: import serveIndex from 'serve-index'
    18: import bodyParser from 'body-parser'
>>> 19: // @ts-expect-error FIXME due to non-existing type definitions for finale-rest
    20: import * as finale from 'finale-rest'
    21: import compression from 'compression'
    22: // @ts-expect-error FIXME due to non-existing type definitions for express-robots-txt
```

M	server.ts **(Line 22)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    19: // @ts-expect-error FIXME due to non-existing type definitions for finale-rest
    20: import * as finale from 'finale-rest'
    21: import compression from 'compression'
>>> 22: // @ts-expect-error FIXME due to non-existing type definitions for express-robots-txt
    23: import robots from 'express-robots-txt'
    24: import cookieParser from 'cookie-parser'
    25: import * as Prometheus from 'prom-client'
```

M	server.ts **(Line 29)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    26: import swaggerUi from 'swagger-ui-express'
    27: import featurePolicy from 'feature-policy'
    28: import { IpFilter } from 'express-ipfilter'
>>> 29: // @ts-expect-error FIXME due to non-existing type definitions for express-security.txt
    30: import securityTxt from 'express-security.txt'
    31: import { rateLimit } from 'express-rate-limit'
    32: import { getStream } from 'file-stream-rotator'
```

M	routes/userProfile.ts **(Line 90)**	**typescript:S6397** Replace this character class by the character itself.

■ **Problematic Code:**

```
    87:     const CSP = `img-src 'self' ${user?.profileImage}; script-src 'self' 'unsafe-eval'
https://code.getmdl.io http://ajax.googleapis.com`
    88:
    89:     challengeUtils.solveIf(challenges.usernameXssChallenge, () => {
>>> 90:       return username && user?.profileImage.match(/;[
]*script-src(.)*'unsafe-inline'/g) !== null && utils.contains(username,
'<script>alert(`xss`)</script>')
    91:     })
    92:
    93:     res.set({
```

M	routes/b2bOrder.ts **(Line 8)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
      5:
      6: import vm from 'node:vm'
      7: import { type Request, type Response, type NextFunction } from 'express'
>>>   8: // @ts-expect-error FIXME due to non-existing type definitions for notevil
      9: import { eval as safeEval } from 'notevil'
     10:
     11: import * as challengeUtils from '../lib/challengeUtils'
```

M	data/datacreator.ts **(Line 36)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     33: import { AllHtmlEntities as Entities } from 'html-entities'
     34: import * as datacache from './datacache'
     35: import * as security from '../lib/insecurity'
>>>  36: // @ts-expect-error FIXME due to non-existing type definitions for replace
     37: import replace from 'replace'
     38:
     39: const entities = new Entities()
```

M	test/cypress/e2e/complain.spec.ts **(Line 113)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    110:       })
    111:     })
    112:
>>> 113:     xit('should be solved either through dev/random or Quadratic Blowup attack', () =>
{ // FIXME Unreliable during CI/CD as sometimes the Quadratic Blowup is blocked for entity loops
    114:       cy.task('isDocker').then((isDocker) => {
    115:         if (!isDocker) {
    116:           cy.expectChallengeSolved({ challenge: 'XXE DoS' })
```

M	test/api/fileUploadSpec.ts **(Line 182)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    179:       .expect('status', 204)
    180:   })
    181:
>>> 182:   xit('POST valid file with tampered content length', () => { // FIXME Fails on CI/CD
pipeline
    183:     const file = path.resolve(__dirname, '../files/validSizeAndTypeForClient.pdf')
    184:     const form = frisby.formData()
    185:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
```

M	test/api/profileImageUploadSpec.ts **(Line 155)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    152:       .expect('bodyContains', 'Error: Blocked illegal activity')
    153:   })
    154:
>>> 155:   xit('POST valid image with tampered content length', () => { // FIXME Fails on CI/CD
pipeline
    156:     const file = path.resolve(__dirname, '../files/validProfileImage.jpg')
    157:     const form = frisby.formData()
    158:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
```

M	test/api/fileUploadSpec.ts **(Line 187)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    184:     const form = frisby.formData()
    185:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
    186:
>>> 187:     // @ts-expect-error FIXME form.getHeaders() is not found
    188:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type':
form.getHeaders()['content-type'], 'Content-Length': 42 }, body: form })
    189:       .expect('status', 500)
    190:       .expect('bodyContains', 'Unexpected end of form')
```

M	test/api/profileImageUploadSpec.ts **(Line 172)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    169:         return frisby.post(`${URL}/profile/image/file`, {
    170:           headers: {
    171:             Cookie: `token=${jsonLogin.authentication.token}`,
>>> 172:             // @ts-expect-error FIXME form.getHeaders() is not found
    173:             'Content-Type': form.getHeaders()['content-type'],
    174:             'Content-Length': 42
    175:           },
```

M	test/api/fileUploadSpec.ts **(Line 143)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    140:       form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
    141:
    142:       return frisby.post(URL + '/file-upload', {
>>> 143:         // @ts-expect-error FIXME form.getHeaders() is not found
    144:         headers: { 'Content-Type': form.getHeaders()['content-type'] },
    145:         body: form
    146:       }).then((res) => {
```

M	frontend/src/app/search-result search-result.component.scss **(Line 25)**	**css:S4666** Unexpected duplicate selector ".heading", first used at line 13

■ **Problematic Code:**

```
     22:   width: 60% !important;       /* image ~60%, info-box ~40% */
     23: }
     24:
>>>  25: .heading {
     26:   margin-bottom: 10px;
     27: }
     28:
```

M	frontend/src/app/search-result search-result.component.scss **(Line 151)**	**css:S4666** Unexpected duplicate selector ".product", first used at line 29

■ **Problematic Code:**

```
    148:   margin-top: 10px;
    149: }
    150:
>>> 151: .product {
    152:   margin-left: -17px;
    153:   margin-right: 6px;
    154:   margin-top: -17px;
```

M	frontend/src/app/faucet faucet.component.scss **(Line 139)**	**css:S4666** Unexpected duplicate selector ".withdraw-container button", first used at line 85

■ **Problematic Code:**

```
    136:   width: 100%;
    137: }
    138:
>>> 139: .withdraw-container button {
    140:   height: 68px;
    141: }
    142: @media (max-width: 1100px) {
```

M	frontend/src/app/token-sale token-sale.component.scss **(Line 83)**	**css:S4666** Unexpected duplicate selector ".faq-container", first used at line 46

■ **Problematic Code:**

```
     80:   border: 0;
     81: }
     82:
>>>  83: .faq-container {
     84:   display: flex;
     85:   flex-direction: column;
     86:   gap: 4px;
```

M	frontend/src/app/navbar navbar.component.scss **(Line 43)**	**css:S4666** Unexpected duplicate selector "mat-toolbar", first used at line 6

Problematic Code:

```
     40:   width: 26px;
     41: }
     42:
>>>  43: mat-toolbar {
     44:   mat-icon {
     45:     font-size: 23px;
     46:     height: 24px;
```

M	frontend/src/app/forgot-password forgot-password.component.scss **(Line 30)**	**css:S4666** Unexpected duplicate selector "mat-form-field", first used at line 12

Problematic Code:

```
     27:   width: 60%;
     28: }
     29:
>>>  30: mat-form-field {
     31:   input {
     32:     color: #fff !important;
     33:   }
```

M	test/cypress/e2e/changePassword.spec.ts **(Line 14)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
     11:     it('should be able to change password', () => {
     12:       cy.get('#currentPassword').type('focusOnScienceMorty!focusOnScience')
     13:       cy.get('#newPassword').type('GonorrheaCantSeeUs!')
>>>  14:       cy.get('#newPasswordRepeat').type('GonorrheaCantSeeUs!', { force: true }) //
FIXME Analyze Cypress recordings to properly fix behavior during test
     15:       cy.get('#changeButton').click()
     16:
     17:       cy.get('.confirmation').should('not.be.hidden')
```

M	test/cypress/e2e/contact.spec.ts **(Line 25)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
     22:       cy.get('#userId').clear().type('2')
     23:       cy.get('#rating').type('{rightarrow}{rightarrow}{rightarrow}')
     24:       cy.get('#comment').type('Picard stinks!')
>>>  25:       cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
     26:
     27:       cy.visit('/#/administration')
     28:
```

M	test/cypress/e2e/contact.spec.ts **(Line 61)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     58:           cy.get('#comment').type(
     59:             '<<script>Foo</script>iframe src="javascript:alert(`xss`)">'
     60:           )
>>>  61:           cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
     62:
     63:           cy.visit('/#/about')
     64:           cy.on('window:alert', (t) => {
```

M	test/cypress/e2e/contact.spec.ts **(Line 84)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     81:       cy.get('#comment').type('sanitize-html 1.4.2 is non-recursive.')
     82:       cy.get('#comment').type('express-jwt 0.1.3 has broken crypto.')
     83:
>>>  84:       cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
     85:       cy.expectChallengeSolved({ challenge: 'Vulnerable Library' })
     86:     })
     87:   })
```

M	test/cypress/e2e/contact.spec.ts **(Line 95)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     92:       cy.get('#comment').type(
     93:         'The following libraries are bad for crypto: z85, base85, md5 and hashids'
     94:       )
>>>  95:       cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
     96:       cy.expectChallengeSolved({ challenge: 'Weird Crypto' })
     97:     })
     98:   })
```

M	test/cypress/e2e/contact.spec.ts **(Line 106)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    103:       cy.get('#comment').type(
    104:         'You are a typosquatting victim of this NPM package: epilogue-js'
    105:       )
>>> 106:       cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
    107:       cy.expectChallengeSolved({ challenge: 'Legacy Typosquatting' })
    108:     })
    109:   })
```

M	test/cypress/e2e/contact.spec.ts **(Line 117)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    114:       cy.get('#comment').type(
    115:         'You are a typosquatting victim of this NPM package: ngy-cookie'
    116:       )
>>> 117:       cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
    118:       cy.expectChallengeSolved({ challenge: 'Frontend Typosquatting' })
    119:     })
    120:   })
```

M	test/cypress/e2e/contact.spec.ts **(Line 128)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    125:       cy.get('#comment').type(
    126:         'Pickle Rick is hiding behind one of the support team ladies'
    127:       )
>>> 128:       cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
    129:       cy.expectChallengeSolved({ challenge: 'Steganography' })
    130:     })
    131:   })
```

M	test/cypress/e2e/contact.spec.ts **(Line 232)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    229:       cy.get('#comment').type(
    230:         'Turn on 2FA! Now!!! https://github.com/eslint/eslint-scope/issues/39'
    231:       )
>>> 232:       cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
    233:       cy.expectChallengeSolved({ challenge: 'Supply Chain Attack' })
    234:     })
    235:   })
```

M	test/cypress/e2e/contact.spec.ts **(Line 245)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    242:           pastebinLeakProduct.keywordsForPastebinDataLeakChallenge ?
pastebinLeakProduct.keywordsForPastebinDataLeakChallenge.toString() : '?'
    243:         )
    244:       })
>>> 245:       cy.get('#submitButton').click({ force: true }) // FIXME Analyze Cypress
recordings to properly fix behavior during test
    246:       cy.expectChallengeSolved({ challenge: 'Leaked Unsafe Product' })
    247:     })
    248:   })
```

M	frontend/src/app/password-strength password-strength.component.ts **(Line 63)**	**typescript:S5869** Remove duplicates in this character class.

■ **Problematic Code:**

```
     60:   }
     61:
     62:   get containAtLeastOneSpecialChar (): boolean {
>>>  63:     return /^(?=.*?[" !"#$%&'()*+,-./:;<=>?@[\]^_`{|}~"])/.test(this.password)
     64:   }
     65: }
     66:
```

M	lib/startup/registerWebsocketEvents.ts **(Line 47)**	**typescript:S6397** Replace this character class by the character itself.

■ **Problematic Code:**

```
     44:     })
     45:
     46:     socket.on('verifySvgInjectionChallenge', (data: any) => {
>>>  47:       challengeUtils.solveIf(challenges.svgInjectionChallenge, () => { return data?.ma
tch(/.*\.\.\/\.\.\/\.\.[\w/-]*?\/redirect\?to=https?:\/\/placecats.com\/(g\/)?[\d]+\/[\d]+.*/)
&& security.isRedirectAllowed(data) })
     48:     })
     49:
     50:     socket.on('verifyCloseNotificationsChallenge', (data: any) => {
```

M	lib/startup/registerWebsocketEvents.ts **(Line 47)**	**typescript:S6397** Replace this character class by the character itself.

■ **Problematic Code:**

```
     44:     })
     45:
     46:     socket.on('verifySvgInjectionChallenge', (data: any) => {
>>>  47:       challengeUtils.solveIf(challenges.svgInjectionChallenge, () => { return data?.ma
tch(/.*\.\.\/\.\.\/\.\.[\w/-]*?\/redirect\?to=https?:\/\/placecats.com\/(g\/)?[\d]+\/[\d]+.*/)
&& security.isRedirectAllowed(data) })
     48:     })
     49:
     50:     socket.on('verifyCloseNotificationsChallenge', (data: any) => {
```

M	test/api/memoryApiSpec.ts **(Line 120)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    117:         return frisby.post(REST_URL + '/memories', {
    118:           headers: {
    119:             Authorization: 'Bearer ' + jsonLogin.authentication.token,
>>> 120:             // @ts-expect-error FIXME form.getHeaders() is not found
    121:             'Content-Type': form.getHeaders()['content-type']
    122:           },
    123:           body: form
```

M	test/api/internetResourcesSpec.ts **(Line 8)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
      5:
      6: import * as frisby from 'frisby'
      7:
>>>   8: xdescribe('Required Internet resource', () => { // FIXME Tests fail regularly (but not
always) from ECONNRESET
      9:   describe('PasteBin paste for "Leaked Unsafe Product" challenge available', () => {
     10:     it('for default configuration (https://pastebin.com/90dUgd7s)', () => {
     11:       return frisby.get('https://pastebin.com/90dUgd7s')
```

M	lib/noUpdate.ts **(Line 4)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
      1: // Credit for the implementation in JS:
https://github.com/daviddossantos/sequelize-notupdate-attributes
      2: import { type Model, type ValidationErrorItemType } from 'sequelize/types'
      3: import { type ValidationOptions } from 'sequelize/types/instance-validator'
>>>   4: // @ts-expect-error FIXME due to non-existing type definitions for sequelize/lib/errors
      5: import { ValidationError, ValidationErrorItem } from 'sequelize/lib/errors'
      6:
      7: interface ExtendedValidationOptions extends ValidationOptions {
```

M	lib/antiCheat.ts **(Line 15)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
     12: import logger from './logger'
     13: import { type NextFunction, type Request, type Response } from 'express'
     14: import * as utils from './utils'
>>>  15: // @ts-expect-error FIXME due to non-existing type definitions for median
     16: import median from 'median'
     17: import { type ChallengeKey } from 'models/challenge'
     18:
```

M	lib/startup/customizeApplication.ts **(Line 9)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
      6: import fs from 'node:fs'
      7: import config from 'config'
      8: import * as utils from '../utils'
>>>   9: // @ts-expect-error FIXME due to non-existing type definitions for replace
     10: import replace from 'replace'
     11:
     12: const customizeApplication = async () => {
```

M	lib/startup/customizeEasterEgg.ts **(Line 8)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
      5:
      6: import config from 'config'
      7: import * as utils from '../utils'
>>>   8: // @ts-expect-error FIXME due to non-existing type definitions for replace
      9: import replace from 'replace'
     10:
     11: const customizeEasterEgg = async () => {
```

M	lib/startup/validateConfig.ts **(Line 10)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
      7: import config from 'config'
      8: import process from 'node:process'
      9: import colors from 'colors/safe'
>>>  10: // @ts-expect-error FIXME due to non-existing type definitions for yaml-schema-validator
     11: import validateSchema from 'yaml-schema-validator/src'
     12:
     13: import type { AppConfig, Memory as MemoryConfig, Product as ProductConfig } from '../config.types'
```

M	lib/startup/validateDependencies.ts **(Line 9)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
      6: import colors from 'colors/safe'
      7: import * as utils from '../utils'
      8: import logger from '../logger'
>>>   9: // @ts-expect-error FIXME due to non-existing type definitions for check-dependencies
     10: import dependencyChecker from 'check-dependencies'
     11:
     12: const validateDependencies = async ({ packageDir = '.', exitOnFailure = true } = {}) => {
```

M	lib/startup/validateDependenciesBasic.ts **(Line 10)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
      7: // otherwise this check would be useless as the app would fail on a random import before even reaching this point
      8: const validateIfDependencyCheckerIsInstalled = async () => {
      9:   try {
>>>  10:     // @ts-expect-error FIXME due to non-existing type definitions for check-dependencies
     11:     await import('check-dependencies')
     12:   } catch (err) {
     13:     console.error('Please run "npm install" before starting the application!')
```

M	lib/startup/validatePreconditions.ts **(Line 15)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     12: import process from 'node:process'
     13: import semver from 'semver'
     14: import portscanner from 'portscanner'
>>>  15: // @ts-expect-error FIXME due to non-existing type definitions for
check-internet-connected
     16: import checkInternetConnected from 'check-internet-connected'
     17:
     18: const domainDependencies = {
```

M	data/datacache.ts **(Line 33)**	**typescript:S6861** Exporting mutable 'let' binding, use 'const' instead.

■ **Problematic Code:**

```
     30: }
     31: export const notifications: Notification[] = []
     32:
>>>  33: export let retrieveBlueprintChallengeFile: string | null = null
     34: export function setRetrieveBlueprintChallengeFile (retrieveBlueprintChallengeFileArg:
string) {
     35:   retrieveBlueprintChallengeFile = retrieveBlueprintChallengeFileArg
     36: }
```

M	lib/antiCheat.ts **(Line 77)**	**typescript:S3358** Extract this nested ternary operation into an independent statement.

■ **Problematic Code:**

```
     74:     cheatScore = Math.min(1, cheatScore)
     75:   }
     76:
>>>  77:   logger.info(`Cheat score for ${areCoupled(challenge, previous().challenge) ? 'coupled
' : (isTrivial(challenge) ? 'trivial ' : '')}${challenge.tutorialOrder ? 'tutorial ' :
''}${colors.cyan(challenge.key)} solved in ${Math.round(minutesSincePreviousSolve)}min (expected
~${minutesExpectedToSolve}min) with${config.get('challenges.showHints') ? '' : 'out'} hints
allowed${percentPrecedingInteraction > -1 ? (' and ' + percentPrecedingInteraction * 100 + '%
expected preceding URL interaction') : ''}: ${cheatScore < 0.33 ?
colors.green(cheatScore.toString()) : (cheatScore < 0.66 ? colors.yellow(cheatScore.toString())
: colors.red(cheatScore.toString()))}`)
     78:   solves.push({ challenge, phase: 'hack it', timestamp, cheatScore })
     79:   return cheatScore
     80: }
```

M	lib/antiCheat.ts **(Line 77)**	**typescript:S3358** Extract this nested ternary operation into an independent statement.

■ Problematic Code:

```
     74:   cheatScore = Math.min(1, cheatScore)
     75:  }
     76:
>>>  77:  logger.info(`Cheat score for ${areCoupled(challenge, previous().challenge) ? 'coupled
' : (isTrivial(challenge) ? 'trivial ' : '')}${challenge.tutorialOrder ? 'tutorial ' :
''}${colors.cyan(challenge.key)} solved in ${Math.round(minutesSincePreviousSolve)}min (expected
~${minutesExpectedToSolve}min) with${config.get('challenges.showHints') ? '' : 'out'} hints
allowed${percentPrecedingInteraction > -1 ? (' and ' + percentPrecedingInteraction * 100 + '%
expected preceding URL interaction') : ''}: ${cheatScore < 0.33 ?
colors.green(cheatScore.toString()) : (cheatScore < 0.66 ? colors.yellow(cheatScore.toString())
: colors.red(cheatScore.toString()))}`)
     78:  solves.push({ challenge, phase: 'hack it', timestamp, cheatScore })
     79:  return cheatScore
     80: }
```

M	lib/startup/validatePreconditions.ts **(Line 90)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
     87:   })
     88:   .catch(() => {
     89:    logger.warn(`Domain ${colors.bold(domain)} is not reachable (${colors.yellow('NOT
OK')} in a future major release)`)
>>>  90:    // @ts-expect-error FIXME Type problem by accessing key via variable
     91:    domainDependencies[domain].forEach((dependency: string) => {
     92:     logger.warn(`${colors.italic(dependency)} will not work as intended without
access to ${colors.bold(domain)}`)
     93:    })
```

M	lib/startup/registerWebsocketEvents.ts **(Line 21)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
     18:
     19: const registerWebsocketEvents = (server: any) => {
     20:  const io = new Server(server, { cors: { origin: 'http://localhost:4200' } })
>>>  21:  // @ts-expect-error FIXME Type safety issue when setting global socket-io object
     22:  globalWithSocketIO.io = io
     23:
     24:  io.on('connection', (socket: any) => {
```

M	routes/login.ts **(Line 48)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
     45:       }
     46:      })
     47:     } else if (user.data?.id) {
>>>  48:      // @ts-expect-error FIXME some properties missing in user - vuln-code-snippet
hide-line
     49:      afterLogin(user, res, next)
     50:     } else {
     51:      res.status(401).send(res.__('Invalid email or password.'))
```

M	test/server/b2bOrderSpec.ts **(Line 39)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
      36:     expect(challenges.rceChallenge.solved).to.equal(true)
      37:   })
      38:
>>>   39:   // FIXME Disabled as test started failing on Linux regularly
      40:   xit('timeout after 2 seconds solves "rceOccupyChallenge"', () => {
      41:     req.body.orderLinesData = '/((a+)+)b/.test("aaaaaaaaaaaaaaaaaaaaaaaaaaaaa")'
      42:
```

M	routes/2fa.ts **(Line 49)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
      46:     const [basket] = await BasketModel.findOrCreate({ where: { UserId: userId } })
      47:
      48:     const token = security.authorize(plainUser)
>>>   49:     // @ts-expect-error FIXME set new property for original basket
      50:     plainUser.bid = basket.id // keep track of original basket for challenge solution
check
      51:     security.authenticatedUsers.put(token, plainUser)
      52:
```

M	routes/chatbot.ts **(Line 98)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
      95:   try {
      96:     const response = await bot.respond(req.body.query, `${user.id}`)
      97:     if (response.action === 'function') {
>>>   98:       // @ts-expect-error FIXME unclean usage of any type as index
      99:       if (response.handler && botUtils[response.handler]) {
     100:         // @ts-expect-error FIXME unclean usage of any type as index
     101:         res.status(200).json(await botUtils[response.handler](req.body.query, user))
```

M	routes/chatbot.ts **(Line 100)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
      97:     if (response.action === 'function') {
      98:       // @ts-expect-error FIXME unclean usage of any type as index
      99:       if (response.handler && botUtils[response.handler]) {
>>>  100:         // @ts-expect-error FIXME unclean usage of any type as index
     101:         res.status(200).json(await botUtils[response.handler](req.body.query, user))
     102:       } else {
     103:         res.status(200).json({
```

M	data/mongodb.ts **(Line 6)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
      3:  * SPDX-License-Identifier: MIT
      4:  */
      5:
>>>   6: // @ts-expect-error FIXME due to non-existing type definitions for MarsDB
      7: import * as MarsDB from 'marsdb'
      8:
      9: export const reviewsCollection = new MarsDB.Collection('posts')
```

M	lib/antiCheat.ts **(Line 127)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    124: }
    125:
    126: function areCoupled (challenge: Challenge, previousChallenge: Challenge) {
>>> 127:   // @ts-expect-error FIXME any type issues
    128:   return coupledChallenges[challenge.key]?.indexOf(previousChallenge.key) > -1 ||
coupledChallenges[previousChallenge.key]?.indexOf(challenge.key) > -1
    129: }
    130:
```

M	lib/insecurity.ts **(Line 19)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     16:
     17: /* jslint node: true */
     18: // eslint-disable-next-line @typescript-eslint/prefer-ts-expect-error
>>>  19: // @ts-expect-error FIXME no typescript definitions for z85 :(
     20: import * as z85 from 'z85'
     21:
     22: export const publicKey = fs ? fs.readFileSync('encryptionkeys/jwt.pub', 'utf8') :
'placeholder-public-key'
```

M	lib/startup/validateConfig.ts **(Line 81)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     78: export const checkUnambiguousMandatorySpecialProducts = (products: ProductConfig[]) =>
{
     79:   let success = true
     80:   specialProducts.forEach(({ name, key }) => {
>>>  81:     // @ts-expect-error FIXME Ignoring any type issue on purpose
     82:     const matchingProducts = products.filter((product) => product[key])
     83:     if (matchingProducts.length === 0) {
     84:       logger.warn(`No product is configured as ${colors.italic(name)} but one is
required (${colors.red('NOT OK')})`)
```

M	lib/startup/validateConfig.ts **(Line 97)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     94: export const checkNecessaryExtraKeysOnSpecialProducts = (products: ProductConfig[]) =>
{
     95:   let success = true
     96:   specialProducts.forEach(({ name, key, extra = {} }) => {
>>>  97:     // @ts-expect-error FIXME implicit any type issue
     98:     const matchingProducts = products.filter((product) => product[key])
     99:     // @ts-expect-error FIXME implicit any type issue
    100:     if (extra.key && matchingProducts.length === 1 && !matchingProducts[0][extra.key])
{
```

M	lib/startup/validateConfig.ts **(Line 99)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     96:   specialProducts.forEach(({ name, key, extra = {} }) => {
     97:     // @ts-expect-error FIXME implicit any type issue
     98:     const matchingProducts = products.filter((product) => product[key])
>>>  99:     // @ts-expect-error FIXME implicit any type issue
    100:     if (extra.key && matchingProducts.length === 1 && !matchingProducts[0][extra.key])
{
    101:       logger.warn(`Product ${colors.italic(matchingProducts[0].name)} configured as
${colors.italic(name)} does't contain necessary ${colors.italic(`${extra.name}`)}
(${colors.red('NOT OK')})`)
    102:       success = false
```

M	lib/startup/validateConfig.ts **(Line 111)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    108: export const checkUniqueSpecialOnProducts = (products: ProductConfig[]) => {
    109:   let success = true
    110:   products.forEach((product) => {
>>> 111:     // @ts-expect-error FIXME any type issue
    112:     const appliedSpecials = specialProducts.filter(({ key }) => product[key])
    113:     if (appliedSpecials.length > 1) {
    114:       logger.warn(`Product ${colors.italic(product.name)} is used as
${appliedSpecials.map(({ name }) => `${colors.italic(name)}`).join(' and ')} but can only be
used for one challenge (${colors.red('NOT OK')})`)
```

M	lib/utils.ts **(Line 70)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     67:
     68: export const version = (module?: string) => {
     69:   if (module) {
>>>  70:     // @ts-expect-error FIXME Ignoring any type issue on purpose
     71:     return packageJson.dependencies[module]
     72:   } else {
     73:     return packageJson.version
```

M	models/relations.ts **(Line 44)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
    41:       name: 'BasketId'
    42:     }
    43:   })
>>> 44:   // @ts-expect-error FIXME type mismatch
    45:   makeKeyNonUpdatable(BasketItemModel, 'BasketId')
    46:
    47:   CardModel.belongsTo(UserModel, {
```

M	models/relations.ts **(Line 97)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
    94:       name: 'ProductId'
    95:     }
    96:   })
>>> 97:   // @ts-expect-error FIXME type mismatch
    98:   makeKeyNonUpdatable(BasketItemModel, 'ProductId')
    99:
   100:   QuantityModel.belongsTo(ProductModel, {
```

M	routes/showProductReviews.ts **(Line 16)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
    13: import * as utils from '../lib/utils'
    14:
    15: // Blocking sleep function as in native MongoDB
>>> 16: // @ts-expect-error FIXME Type safety broken for global object
    17: global.sleep = (time: number) => {
    18:   // Ensure that users don't accidentally dos their servers for too long
    19:   if (time > 2000) {
```

M	server.ts **(Line 243)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
   240:   const serveIndexMiddleware = (req: Request, res: Response, next: NextFunction) => {
   241:     // eslint-disable-next-line @typescript-eslint/unbound-method
   242:     const origEnd = res.end
>>> 243:     // @ts-expect-error FIXME assignment broken due to seemingly void return value
   244:     res.end = function () {
   245:       if (arguments.length) {
   246:         const reqPath = req.originalUrl.replace(/\?.*$/, '')
```

M	server.ts **(Line 261)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    258:           return 'a href="' + relativePath + '"'
    259:         })
    260:       }
>>> 261:       // @ts-expect-error FIXME passed argument has wrong type
    262:       origEnd.apply(this, arguments)
    263:     }
    264:     next()
```

M	server.ts **(Line 311)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    308:
    309:   app.use(bodyParser.text({ type: '*/*' }))
    310:   app.use(function jsonParser (req: Request, res: Response, next: NextFunction) {
>>> 311:     // @ts-expect-error FIXME intentionally saving original request in this property
    312:     req.rawBody = req.body
    313:     if (req.headers['content-type']?.includes('application/json')) {
    314:       if (!req.body) {
```

M	test/api/2faSpec.ts **(Line 21)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     18: const jsonHeader = { 'content-type': 'application/json' }
     19:
     20: async function login ({ email, password, totpSecret }: { email: string, password:
string, totpSecret?: string }) {
>>>  21:   // @ts-expect-error FIXME promise return handling broken
     22:   const loginRes = await frisby
     23:     .post(REST_URL + '/user/login', {
     24:       email,
```

M	test/api/2faSpec.ts **(Line 38)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     35:       throw new Error('login with totp required but no totp secret provided to login
function')
     36:     }
     37:
>>>  38:     // @ts-expect-error FIXME promise return handling broken
     39:     const totpRes = await frisby
     40:       .post(REST_URL + '/2fa/verify', {
     41:         tmpToken: loginRes.json.data.tmpToken,
```

M	test/api/2faSpec.ts **(Line 52)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     49: }
     50:
     51: async function register ({ email, password, totpSecret }: { email: string, password: string, totpSecret?: string }) {
>>>  52:   // @ts-expect-error FIXME promise return handling broken
     53:   const res = await frisby
     54:     .post(API_URL + '/Users/', {
     55:       email,
```

M	test/api/2faSpec.ts **(Line 67)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     64:   if (totpSecret) {
     65:     const { token } = await login({ email, password })
     66:
>>>  67:     // @ts-expect-error FIXME promise return handling broken
     68:     await frisby.post(
     69:       REST_URL + '/2fa/setup',
     70:       {
```

M	test/api/2faSpec.ts **(Line 111)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    108:
    109:     const totpToken = otplib.authenticator.generate('IFTXE3SPOEYVURT2MRYGI52TKJ4HC3KH')
    110:
>>> 111:     // @ts-expect-error FIXME promise return handling broken
    112:     await frisby.post(REST_URL + '/2fa/verify', {
    113:       headers: jsonHeader,
    114:       body: {
```

M	test/api/2faSpec.ts **(Line 139)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    136:
    137:     const totpToken = otplib.authenticator.generate('THIS9ISNT8THE8RIGHT8SECRET')
    138:
>>> 139:     // @ts-expect-error FIXME promise return handling broken
    140:     await frisby.post(REST_URL + '/2fa/verify', {
    141:       headers: jsonHeader,
    142:       body: {
```

M	test/api/2faSpec.ts **(Line 158)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    155:
    156:     const totpToken = otplib.authenticator.generate('IFTXE3SPOEYVURT2MRYGI52TKJ4HC3KH')
    157:
>>> 158:     // @ts-expect-error FIXME promise return handling broken
    159:     await frisby.post(REST_URL + '/2fa/verify', {
    160:       headers: jsonHeader,
    161:       body: {
```

M	test/api/2faSpec.ts **(Line 178)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    175:       totpSecret: 'IFTXE3SPOEYVURT2MRYGI52TKJ4HC3KH'
    176:     })
    177:
>>> 178:     // @ts-expect-error FIXME promise return handling broken
    179:     await frisby.get(
    180:       REST_URL + '/2fa/status',
    181:       {
```

M	test/api/2faSpec.ts **(Line 203)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    200:       password: '0Y8rMnww$*9VFYE§59-!Fg1L6t&6lB'
    201:     })
    202:
>>> 203:     // @ts-expect-error FIXME promise return handling broken
    204:     await frisby.get(
    205:       REST_URL + '/2fa/status',
    206:       {
```

M	test/api/2faSpec.ts **(Line 227)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    224:   })
    225:
    226:   it('GET should return 401 when not logged in', async () => {
>>> 227:     // @ts-expect-error FIXME promise return handling broken
    228:     await frisby.get(REST_URL + '/2fa/status')
    229:       .expect('status', 401)
    230:   })
```

M	test/api/2faSpec.ts **(Line 243)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    240:     await register({ email, password })
    241:     const { token } = await login({ email, password })
    242:
>>> 243:     // @ts-expect-error FIXME promise return handling broken
    244:     await frisby.post(
    245:       REST_URL + '/2fa/setup',
    246:       {
```

M	test/api/2faSpec.ts **(Line 262)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    259:       })
    260:       .expect('status', 200)
    261:
>>> 262:     // @ts-expect-error FIXME promise return handling broken
    263:     await frisby.get(
    264:       REST_URL + '/2fa/status',
    265:       {
```

M	test/api/2faSpec.ts **(Line 289)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    286:     await register({ email, password })
    287:     const { token } = await login({ email, password })
    288:
>>> 289:     // @ts-expect-error FIXME promise return handling broken
    290:     await frisby.post(
    291:       REST_URL + '/2fa/setup',
    292:       {
```

M	test/api/2faSpec.ts **(Line 318)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    315:     await register({ email, password })
    316:     const { token } = await login({ email, password })
    317:
>>> 318:     // @ts-expect-error FIXME promise return handling broken
    319:     await frisby.post(
    320:       REST_URL + '/2fa/setup',
    321:       {
```

M	test/api/2faSpec.ts **(Line 347)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    344:     await register({ email, password })
    345:     const { token } = await login({ email, password })
    346:
>>> 347:     // @ts-expect-error FIXME promise return handling broken
    348:     await frisby.post(
    349:       REST_URL + '/2fa/setup',
    350:       {
```

M	test/api/2faSpec.ts **(Line 374)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    371:
    372:     const { token } = await login({ email, password, totpSecret })
    373:
>>> 374:     // @ts-expect-error FIXME promise return handling broken
    375:     await frisby.post(
    376:       REST_URL + '/2fa/setup',
    377:       {
```

M	test/api/2faSpec.ts **(Line 404)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    401:     await register({ email, password, totpSecret })
    402:     const { token } = await login({ email, password, totpSecret })
    403:
>>> 404:     // @ts-expect-error FIXME promise return handling broken
    405:     await getStatus(token)
    406:       .expect('status', 200)
    407:       .expect('json', {
```

M	test/api/2faSpec.ts **(Line 411)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    408:         setup: true
    409:       })
    410:
>>> 411:     // @ts-expect-error FIXME promise return handling broken
    412:     await frisby.post(
    413:       REST_URL + '/2fa/disable',
    414:       {
```

M	test/api/2faSpec.ts **(Line 425)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    422:       }
    423:     ).expect('status', 200)
    424:
>>> 425:     // @ts-expect-error FIXME promise return handling broken
    426:     await getStatus(token)
    427:       .expect('status', 200)
    428:       .expect('json', {
```

M	test/api/2faSpec.ts **(Line 441)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    438:     await register({ email, password, totpSecret })
    439:     const { token } = await login({ email, password, totpSecret })
    440:
>>> 441:     // @ts-expect-error FIXME promise return handling broken
    442:     await getStatus(token)
    443:       .expect('status', 200)
    444:       .expect('json', {
```

M	test/api/2faSpec.ts **(Line 448)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    445:         setup: true
    446:       })
    447:
>>> 448:     // @ts-expect-error FIXME promise return handling broken
    449:     await frisby.post(
    450:       REST_URL + '/2fa/disable',
    451:       {
```

M	test/api/2faSpec.ts **(Line 462)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    459:       }
    460:     ).expect('status', 401)
    461:
>>> 462:     // @ts-expect-error FIXME promise return handling broken
    463:     await getStatus(token)
    464:       .expect('status', 200)
    465:       .expect('json', {
```

M	test/api/chatBotSpec.ts **(Line 19)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     16: let trainingData: { data: any[] }
     17:
     18: async function login ({ email, password }: { email: string, password: string }) {
>>>  19:   // @ts-expect-error FIXME promise return handling broken
     20:   const loginRes = await frisby
     21:     .post(REST_URL + '/user/login', {
     22:       email,
```

M	test/api/dataExportApiSpec.ts **(Line 202)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    199:         return frisby.post(REST_URL + '/memories', {
    200:           headers: {
    201:             Authorization: 'Bearer ' + jsonLogin.authentication.token,
>>> 202:             // @ts-expect-error FIXME form.getHeaders() is not found
    203:             'Content-Type': form.getHeaders()['content-type']
    204:           },
    205:           body: form
```

M	test/api/dataExportApiSpec.ts **(Line 340)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    337:         return frisby.post(REST_URL + '/memories', {
    338:           headers: {
    339:             Authorization: 'Bearer ' + jsonLogin.authentication.token,
>>> 340:             // @ts-expect-error FIXME form.getHeaders() is not found
    341:             'Content-Type': form.getHeaders()['content-type']
    342:           },
    343:           body: form
```

M	test/api/deluxeApiSpec.ts **(Line 14)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     11: const API_URL = 'http://localhost:3000/api'
     12:
     13: async function login ({ email, password }: { email: string, password: string }) {
>>>  14:   // @ts-expect-error FIXME promise return handling broken
     15:   const loginRes = await frisby
     16:     .post(`${REST_URL}/user/login`, {
     17:       email,
```

M	test/api/fileUploadSpec.ts **(Line 21)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     18:     const form = frisby.formData()
     19:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
     20:
>>>  21:     // @ts-expect-error FIXME form.getHeaders() is not found
     22:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type':
form.getHeaders()['content-type'] }, body: form })
     23:       .expect('status', 204)
     24:   })
```

M	test/api/fileUploadSpec.ts **(Line 31)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     28:     const form = frisby.formData()
     29:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
     30:
>>>  31:     // @ts-expect-error FIXME form.getHeaders() is not found
     32:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type':
form.getHeaders()['content-type'] }, body: form })
     33:       .expect('status', 204)
     34:   })
```

M	test/api/fileUploadSpec.ts **(Line 41)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     38:     const form = frisby.formData()
     39:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
     40:
>>>  41:     // @ts-expect-error FIXME form.getHeaders() is not found
     42:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type':
form.getHeaders()['content-type'] }, body: form })
     43:       .expect('status', 204)
     44:   })
```

M	test/api/fileUploadSpec.ts **(Line 51)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     48:     const form = frisby.formData()
     49:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
     50:
>>>  51:     // @ts-expect-error FIXME form.getHeaders() is not found
     52:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type':
form.getHeaders()['content-type'] }, body: form })
     53:       .expect('status', 410)
     54:   })
```

M	test/api/fileUploadSpec.ts **(Line 61)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     58:     const form = frisby.formData()
     59:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
     60:
>>>  61:     // @ts-expect-error FIXME form.getHeaders() is not found
     62:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type':
form.getHeaders()['content-type'] }, body: form })
     63:       .expect('status', 410)
     64:   })
```

M	test/api/fileUploadSpec.ts **(Line 73)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     70:       form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
     71:
     72:       return frisby.post(URL + '/file-upload', {
>>>  73:         // @ts-expect-error FIXME form.getHeaders() is not found
     74:         headers: { 'Content-Type': form.getHeaders()['content-type'] },
     75:         body: form
     76:       })
```

M	test/api/fileUploadSpec.ts **(Line 86)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     83:       form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
     84:
     85:       return frisby.post(URL + '/file-upload', {
>>>  86:         // @ts-expect-error FIXME form.getHeaders() is not found
     87:         headers: { 'Content-Type': form.getHeaders()['content-type'] },
     88:         body: form
     89:       })
```

M	test/api/fileUploadSpec.ts **(Line 99)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     96:       form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
     97:
     98:       return frisby.post(URL + '/file-upload', {
>>>  99:         // @ts-expect-error FIXME form.getHeaders() is not found
    100:         headers: { 'Content-Type': form.getHeaders()['content-type'] },
    101:         body: form
    102:       })
```

M	test/api/fileUploadSpec.ts **(Line 113)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
    110:       form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
    111:
    112:       return frisby.post(URL + '/file-upload', {
>>> 113:         // @ts-expect-error FIXME form.getHeaders() is not found
    114:         headers: { 'Content-Type': form.getHeaders()['content-type'] },
    115:         body: form
    116:       }).then((res) => {
```

M	test/api/fileUploadSpec.ts **(Line 127)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
    124:       form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
    125:
    126:       return frisby.post(URL + '/file-upload', {
>>> 127:         // @ts-expect-error FIXME form.getHeaders() is not found
    128:         headers: { 'Content-Type': form.getHeaders()['content-type'] },
    129:         body: form
    130:       }).then((res) => {
```

M	test/api/fileUploadSpec.ts **(Line 157)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
    154:     const form = frisby.formData()
    155:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
    156:
>>> 157:     // @ts-expect-error FIXME form.getHeaders() is not found
    158:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type':
form.getHeaders()['content-type'] }, body: form })
    159:       .expect('status', 500)
    160:   })
```

M	test/api/fileUploadSpec.ts **(Line 167)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
    164:     const form = frisby.formData()
    165:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
    166:
>>> 167:     // @ts-expect-error FIXME form.getHeaders() is not found
    168:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type':
form.getHeaders()['content-type'] }, body: form })
    169:       .expect('status', 204)
    170:   })
```

M	test/api/fileUploadSpec.ts **(Line 177)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
    174:     const form = frisby.formData()
    175:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
    176:
>>> 177:     // @ts-expect-error FIXME form.getHeaders() is not found
    178:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type':
form.getHeaders()['content-type'] }, body: form })
    179:       .expect('status', 204)
    180:   })
```

M	test/api/memoryApiSpec.ts **(Line 46)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
     43:
     44:     return frisby.post(REST_URL + '/memories', {
     45:       headers: {
>>>  46:         // @ts-expect-error FIXME form.getHeaders() is not found
     47:         'Content-Type': form.getHeaders()['content-type']
     48:       },
     49:       body: form
```

M	test/api/memoryApiSpec.ts **(Line 72)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
     69:         return frisby.post(REST_URL + '/memories', {
     70:           headers: {
     71:             Authorization: 'Bearer ' + jsonLogin.authentication.token,
>>>  72:             // @ts-expect-error FIXME form.getHeaders() is not found
     73:             'Content-Type': form.getHeaders()['content-type']
     74:           },
     75:           body: form
```

M	test/api/memoryApiSpec.ts **(Line 150)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

Problematic Code:

```
    147:         return frisby.post(REST_URL + '/memories', {
    148:           headers: {
    149:             Authorization: 'Bearer ' + jsonLogin.authentication.token,
>>> 150:             // @ts-expect-error FIXME form.getHeaders() is not found
    151:             'Content-Type': form.getHeaders()['content-type']
    152:           },
    153:           body: form
```

M	test/api/metricsApiSpec.ts **(Line 38)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     35:     const form = frisby.formData()
     36:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to blob as the frisby types are wrong and wont accept the fileStream type
     37:
>>>  38:     // @ts-expect-error FIXME form.getHeaders() is not found
     39:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type': form.getHeaders()['content-type'] }, body: form })
     40:       .expect('status', 204)
     41:       .then(() => {
```

M	test/api/metricsApiSpec.ts **(Line 54)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     51:     const form = frisby.formData()
     52:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to blob as the frisby types are wrong and wont accept the fileStream type
     53:
>>>  54:     // @ts-expect-error FIXME form.getHeaders() is not found
     55:     return frisby.post(URL + '/file-upload', { headers: { 'Content-Type': form.getHeaders()['content-type'] }, body: form })
     56:       .expect('status', 500)
     57:       .then(() => {
```

M	test/api/profileImageUploadSpec.ts **(Line 33)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     30:         return frisby.post(`${URL}/profile/image/file`, {
     31:           headers: {
     32:             Cookie: `token=${jsonLogin.authentication.token}`,
>>>  33:             // @ts-expect-error FIXME form.getHeaders() is not found
     34:             'Content-Type': form.getHeaders()['content-type']
     35:           },
     36:           body: form,
```

M	test/api/profileImageUploadSpec.ts **(Line 60)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     57:         return frisby.post(`${URL}/profile/image/file`, {
     58:           headers: {
     59:             Cookie: `token=${jsonLogin.authentication.token}`,
>>>  60:             // @ts-expect-error FIXME form.getHeaders() is not found
     61:             'Content-Type': form.getHeaders()['content-type']
     62:           },
     63:           body: form
```

M	test/api/profileImageUploadSpec.ts **(Line 78)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     75:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to blob as the frisby types are wrong and wont accept the fileStream type
     76:
     77:     return frisby.post(`${URL}/profile/image/file`, {
>>>  78:       // @ts-expect-error FIXME form.getHeaders() is not found
     79:       headers: { 'Content-Type': form.getHeaders()['content-type'] },
     80:       body: form
     81:     })
```

M	test/api/profileImageUploadSpec.ts **(Line 105)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    102:         return frisby.post(`${URL}/profile/image/url`, {
    103:           headers: {
    104:             Cookie: `token=${jsonLogin.authentication.token}`,
>>> 105:             // @ts-expect-error FIXME form.getHeaders() is not found
    106:             'Content-Type': form.getHeaders()['content-type']
    107:           },
    108:           body: form,
```

M	test/api/profileImageUploadSpec.ts **(Line 131)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    128:         return frisby.post(`${URL}/profile/image/url`, {
    129:           headers: {
    130:             Cookie: `token=${jsonLogin.authentication.token}`,
>>> 131:             // @ts-expect-error FIXME form.getHeaders() is not found
    132:             'Content-Type': form.getHeaders()['content-type']
    133:           },
    134:           body: form,
```

M	test/api/profileImageUploadSpec.ts **(Line 146)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    143:     form.append('imageUrl', 'cataas.com/cat')
    144:
    145:     return frisby.post(`${URL}/profile/image/url`, {
>>> 146:       // @ts-expect-error FIXME form.getHeaders() is not found
    147:       headers: { 'Content-Type': form.getHeaders()['content-type'] },
    148:       body: form
    149:     })
```

M	test/api/userProfileSpec.ts **(Line 51)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     48:     form.append('username', 'Localhorst')
     49:
     50:     return frisby.post(`${URL}/profile`, {
>>>  51:       // @ts-expect-error FIXME form.getHeaders() is not found
     52:       headers: { 'Content-Type': form.getHeaders()['content-type'], Cookie:
authHeader.Cookie },
     53:       body: form,
     54:       redirect: 'manual'
```

M	test/api/userProfileSpec.ts **(Line 64)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     61:     form.append('username', 'Localhorst')
     62:
     63:     return frisby.post(`${URL}/profile`, {
>>>  64:       // @ts-expect-error FIXME form.getHeaders() is not found
     65:       headers: { 'Content-Type': form.getHeaders()['content-type'] },
     66:       body: form
     67:     })
```

M	test/server/insecuritySpec.ts **(Line 6)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
      3:  * SPDX-License-Identifier: MIT
      4:  */
      5:
>>>   6: // @ts-expect-error FIXME no typescript definitions for z85 :(
      7: import z85 from 'z85'
      8: import chai from 'chai'
      9: import * as security from '../../lib/insecurity'
```

M	frontend/src/app/score-board/components filter-settings/pipes difficulty-selection-summary.pipe.ts **(Line 34)**	**typescript:S6660** 'If' statement should not be the only statement in 'else' block

■ **Problematic Code:**

```
     31:     if (currentGroup === null) {
     32:       currentGroup = { start: difficulty, end: difficulty }
     33:     } else {
>>>  34:       if (difficulty === currentGroup.end + 1) {
     35:         currentGroup.end = difficulty
     36:       } else {
     37:         difficultyGroups.push(currentGroup)
```

M	frontend/src/app/wallet-web3 wallet-web3.component.ts **(Line 72)**	**typescript:S1854** Remove this useless assignment to variable "txConfirmation".

■ Problematic Code:

```
     69:         value: ethers.utils.parseEther(depositAmount)
     70:       })
     71:       // eslint-disable-next-line @typescript-eslint/no-unused-vars
>>>  72:       const txConfirmation = await transaction.wait()
     73:       this.getUserEthBalance()
     74:     } catch (error) {
     75:       this.errorMessage = error.message
```

M	frontend/src/app/wallet-web3 wallet-web3.component.ts **(Line 90)**	**typescript:S1854** Remove this useless assignment to variable "txConfirmation".

■ Problematic Code:

```
     87:         ethers.utils.parseEther(withdrawalAmount)
     88:       )
     89:       // eslint-disable-next-line @typescript-eslint/no-unused-vars
>>>  90:       const txConfirmation = await transaction.wait()
     91:       this.getUserEthBalance()
     92:     } catch (error) {
     93:       this.errorMessage = error.message
```

M	routes/checkKeys.ts **(Line 21)**	**typescript:S6660** 'If' statement should not be the only statement in 'else' block

■ Problematic Code:

```
     18:       if (req.body.privateKey === privateKey) {
     19:         res.status(200).json({ success: true, message: 'Challenge successfully solved',
status: challenges.nftUnlockChallenge })
     20:       } else {
>>>  21:         if (req.body.privateKey === address) {
     22:           res.status(401).json({ success: false, message: 'Looks like you entered the
public address of my ethereum wallet!', status: challenges.nftUnlockChallenge })
     23:         } else if (req.body.privateKey === publicKey) {
     24:           res.status(401).json({ success: false, message: 'Looks like you entered the
public key of my ethereum wallet!', status: challenges.nftUnlockChallenge })
```

M	Dockerfile **(Line 19)**	**docker:S6570** Surround this variable with double quotes; otherwise, it can lead to unexpected behavior.

■ Problematic Code:

```
     16: RUN rm i18n/*.json || true
     17:
     18: ARG CYCLONEDX_NPM_VERSION=latest
>>>  19: RUN npm install -g @cyclonedx/cyclonedx-npm@$CYCLONEDX_NPM_VERSION
     20: RUN npm run sbom
     21:
     22: FROM gcr.io/distroless/nodejs22-debian12
```

M	data/static/codefixes exposedMetricsChallenge_1.ts **(Line 18)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ **Problematic Code:**

```
     15:   metricsUpdateLoop = Metrics.updateLoop()
     16:
     17:   server.listen(port, () => {
>>>  18:     logger.info(colors.cyan(`Server listening on port ${colors.bold(`${port}`)}`))
     19:     startupGauge.set({ task: 'ready' }, (Date.now() - startTime) / 1000)
     20:     if (process.env.BASE_PATH !== '') {
     21:       logger.info(colors.cyan(`Server using proxy base path
${colors.bold(`${process.env.BASE_PATH}`)} for redirects`))
```

M	data/static/codefixes exposedMetricsChallenge_1.ts **(Line 21)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ **Problematic Code:**

```
     18:     logger.info(colors.cyan(`Server listening on port ${colors.bold(`${port}`)}`))
     19:     startupGauge.set({ task: 'ready' }, (Date.now() - startTime) / 1000)
     20:     if (process.env.BASE_PATH !== '') {
>>>  21:       logger.info(colors.cyan(`Server using proxy base path
${colors.bold(`${process.env.BASE_PATH}`)} for redirects`))
     22:     }
     23:     registerWebsocketEvents(server)
     24:     if (readyCallback) {
```

M	data/static/codefixes exposedMetricsChallenge_2.ts **(Line 15)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ **Problematic Code:**

```
     12:   process.env.BASE_PATH = process.env.BASE_PATH ?? config.get('server.basePath')
     13:
     14:   server.listen(port, () => {
>>>  15:     logger.info(colors.cyan(`Server listening on port ${colors.bold(`${port}`)}`))
     16:     startupGauge.set({ task: 'ready' }, (Date.now() - startTime) / 1000)
     17:     if (process.env.BASE_PATH !== '') {
     18:       logger.info(colors.cyan(`Server using proxy base path
${colors.bold(`${process.env.BASE_PATH}`)} for redirects`))
```

M	data/static/codefixes exposedMetricsChallenge_2.ts **(Line 18)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ **Problematic Code:**

```
     15:     logger.info(colors.cyan(`Server listening on port ${colors.bold(`${port}`)}`))
     16:     startupGauge.set({ task: 'ready' }, (Date.now() - startTime) / 1000)
     17:     if (process.env.BASE_PATH !== '') {
>>>  18:       logger.info(colors.cyan(`Server using proxy base path
${colors.bold(`${process.env.BASE_PATH}`)} for redirects`))
     19:     }
     20:     registerWebsocketEvents(server)
     21:     if (readyCallback) {
```

M	data/static/codefixes exposedMetricsChallenge_3_correct.ts **(Line 18)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ Problematic Code:

```
     15:   metricsUpdateLoop = Metrics.updateLoop()
     16:
     17:   server.listen(port, () => {
>>>  18:     logger.info(colors.cyan(`Server listening on port ${colors.bold(`${port}`)}`))
     19:     startupGauge.set({ task: 'ready' }, (Date.now() - startTime) / 1000)
     20:     if (process.env.BASE_PATH !== '') {
     21:       logger.info(colors.cyan(`Server using proxy base path
${colors.bold(`${process.env.BASE_PATH}`)} for redirects`))
```

M	data/static/codefixes exposedMetricsChallenge_3_correct.ts **(Line 21)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ Problematic Code:

```
     18:     logger.info(colors.cyan(`Server listening on port ${colors.bold(`${port}`)}`))
     19:     startupGauge.set({ task: 'ready' }, (Date.now() - startTime) / 1000)
     20:     if (process.env.BASE_PATH !== '') {
>>>  21:       logger.info(colors.cyan(`Server using proxy base path
${colors.bold(`${process.env.BASE_PATH}`)} for redirects`))
     22:     }
     23:     registerWebsocketEvents(server)
     24:     if (readyCallback) {
```

M	routes/chatbot.ts **(Line 25)**	**typescript:S6861** Exporting mutable 'let' binding, use 'const' instead.

■ Problematic Code:

```
     22:
     23: let trainingFile = config.get<string>('application.chatBot.trainingData')
     24: let testCommand: string
>>>  25: export let bot: Bot | null = null
     26:
     27: export async function initializeChatbot () {
     28:   if (utils.isUrl(trainingFile)) {
```

M	lib/accuracy.ts **(Line 58)**	**typescript:S3358** Extract this nested ternary operation into an independent statement.

■ Problematic Code:

```
     55:   if (solves[challengeKey][phase]) {
     56:     accuracy = 1 / solves[challengeKey].attempts[phase]
     57:   }
>>>  58:   logger.info(`Accuracy for '${phase === 'fix it' ? 'Fix It' : 'Find It'}' phase of
coding challenge ${colors.cyan(challengeKey)}: ${accuracy > 0.5 ?
colors.green(accuracy.toString()) : (accuracy > 0.25 ? colors.yellow(accuracy.toString()) :
colors.red(accuracy.toString()))}`)
     59:   return accuracy
     60: }
     61:
```

M	lib/antiCheat.ts **(Line 102)**	**typescript:S3358** Extract this nested ternary operation into an independent statement.

■ Problematic Code:

```
     99:   const minutesSincePreviousSolve = (timestamp.getTime() -
previous().timestamp.getTime()) / 60000
    100:   cheatScore += Math.max(0, 1 - (minutesSincePreviousSolve / minutesExpectedToSolve))
    101:
>>> 102:   logger.info(`Cheat score for "Find it" phase of ${challenge.key ===
'scoreBoardChallenge' && config.get('hackingInstructor.isEnabled') ? 'tutorial ' :
''}${colors.cyan(challenge.key)} solved in ${Math.round(minutesSincePreviousSolve)}min (expected
~${minutesExpectedToSolve}min): ${cheatScore < 0.33 ? colors.green(cheatScore.toString()) :
(cheatScore < 0.66 ? colors.yellow(cheatScore.toString()) :
colors.red(cheatScore.toString()))}`)
    103:   solves.push({ challenge, phase: 'find it', timestamp, cheatScore })
    104:
    105:   return cheatScore
```

M	lib/antiCheat.ts **(Line 117)**	**typescript:S3358** Extract this nested ternary operation into an independent statement.

■ Problematic Code:

```
    114:   const minutesSincePreviousSolve = (timestamp.getTime() -
previous().timestamp.getTime()) / 60000
    115:   cheatScore += Math.max(0, 1 - (minutesSincePreviousSolve / minutesExpectedToSolve))
    116:
>>> 117:   logger.info(`Cheat score for "Fix it" phase of ${colors.cyan(challenge.key)} solved
in ${Math.round(minutesSincePreviousSolve)}min (expected ~${minutesExpectedToSolve}min):
${cheatScore < 0.33 ? colors.green(cheatScore.toString()) : (cheatScore < 0.66 ?
colors.yellow(cheatScore.toString()) : colors.red(cheatScore.toString()))}`)
    118:   solves.push({ challenge, phase: 'fix it', timestamp, cheatScore })
    119:   return cheatScore
    120: }
```

M	lib/insecurity.ts **(Line 61)**	**typescript:S6397** Replace this character class by the character itself.

■ **Problematic Code:**

```
     58: export const decode = (token: string) => { return jws.decode(token)?.payload }
     59:
     60: export const sanitizeHtml = (html: string) => sanitizeHtmlLib(html)
>>>  61: export const sanitizeLegacy = (input = '') => input.replace(/<(?:\w+)\W+?[\w]/gi, '')
     62: export const sanitizeFilename = (filename: string) => sanitizeFilenameLib(filename)
     63: export const sanitizeSecure = (html: string): string => {
     64:   const sanitized = sanitizeHtml(html)
```

M	lib/startup/validateConfig.ts **(Line 101)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ **Problematic Code:**

```
     98:     const matchingProducts = products.filter((product) => product[key])
     99:     // @ts-expect-error FIXME implicit any type issue
    100:     if (extra.key && matchingProducts.length === 1 && !matchingProducts[0][extra.key])
{
>>> 101:       logger.warn(`Product ${colors.italic(matchingProducts[0].name)} configured as
${colors.italic(name)} does't contain necessary ${colors.italic(`${extra.name}`)}
(${colors.red('NOT OK')})`)
    102:       success = false
    103:     }
    104:   })
```

M	lib/utils.ts **(Line 22)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     19: import isWindows from './is-windows'
     20: export { default as isDocker } from './is-docker'
     21: export { default as isWindows } from './is-windows'
>>>  22: // import isGitpod from 'is-gitpod') // FIXME Roll back to this when
https://github.com/dword-design/is-gitpod/issues/94 is resolve
     23: const isGitpod = () => false
     24:
     25: const months = ['JAN', 'FEB', 'MAR', 'APR', 'MAY', 'JUN', 'JUL', 'AUG', 'SEP', 'OCT',
'NOV', 'DEC']
```

M	server.ts **(Line 727)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ **Problematic Code:**

```
    724:   metricsUpdateLoop = Metrics.updateLoop() // vuln-code-snippet neutral-line
exposedMetricsChallenge
    725:
    726:   server.listen(port, () => {
>>> 727:     logger.info(colors.cyan(`Server listening on port ${colors.bold(`${port}`)}`))
    728:     startupGauge.set({ task: 'ready' }, (Date.now() - startTime) / 1000)
    729:     if (process.env.BASE_PATH !== '') {
    730:       logger.info(colors.cyan(`Server using proxy base path
${colors.bold(`${process.env.BASE_PATH}`)} for redirects`))
```

M	server.ts **(Line 730)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ **Problematic Code:**

```
    727:     logger.info(colors.cyan(`Server listening on port ${colors.bold(`${port}`)}`))
    728:     startupGauge.set({ task: 'ready' }, (Date.now() - startTime) / 1000)
    729:     if (process.env.BASE_PATH !== '') {
>>> 730:       logger.info(colors.cyan(`Server using proxy base path
${colors.bold(`${process.env.BASE_PATH}`)} for redirects`))
    731:     }
    732:     registerWebsocketEvents(server)
    733:     if (readyCallback) {
```

M	test/api/productReviewApiSpec.ts **(Line 42)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     39:       .expect('jsonTypes', reviewResponseSchema)
     40:   })
     41:
>>>  42:   xit('GET product reviews by alphanumeric non-mongoDB-command product id', () => { //
FIXME Turn on when #1960 is resolved
     43:     return frisby.get(`${REST_URL}/products/kaboom/reviews`)
     44:       .expect('status', 400)
     45:   })
```

M	test/cypress/e2e/profile.spec.ts **(Line 107)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    104:           headers: {
    105:             'Content-type': 'application/x-www-form-urlencoded',
    106:             Authorization: `Bearer ${localStorage.getItem('token')}`,
>>> 107:             Origin: 'http://htmledit.squarefree.com', // FIXME Not allowed by browser
due to "unsafe header not permitted"
    108:             Cookie: `token=${localStorage.getItem('token')}` // FIXME Not allowed by
browser due to "unsafe header not permitted"
    109:           },
    110:           body: formData
```

M	test/cypress/e2e/profile.spec.ts **(Line 108)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    105:             'Content-type': 'application/x-www-form-urlencoded',
    106:             Authorization: `Bearer ${localStorage.getItem('token')}`,
    107:             Origin: 'http://htmledit.squarefree.com', // FIXME Not allowed by browser
due to "unsafe header not permitted"
>>> 108:             Cookie: `token=${localStorage.getItem('token')}` // FIXME Not allowed by
browser due to "unsafe header not permitted"
    109:           },
    110:           body: formData
    111:         })
```

M	test/cypress/e2e/complain.spec.ts **(Line 14)**	**typescript:S6666** Use the spread operator instead of '.apply()'.

■ **Problematic Code:**

```
     11:   describe('challenge "uploadSize"', () => {
     12:     it('should be possible to upload files greater 100 KB directly through backend', ()
=> {
     13:       cy.window().then(async () => {
>>>  14:         const over100KB = Array.apply(null, new Array(11000)).map(
     15:           // eslint-disable-next-line @typescript-eslint/unbound-method
     16:           String.prototype.valueOf,
     17:           '1234567890'
```

M	test/api/metricsApiSpec.ts **(Line 14)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     11: const API_URL = 'http://localhost:3000/metrics'
     12:
     13: describe('/metrics', () => {
>>>  14:   xit('GET metrics via public API that are available instantaneously', () => { // FIXME
Flaky on CI/CD on at least Windows
     15:     return frisby.get(API_URL)
     16:       .expect('status', 200)
     17:       .expect('header', 'content-type', /text\/plain/)
```

M	test/api/metricsApiSpec.ts **(Line 33)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     30:       .expect('bodyContains', /^http_requests_count{status_code="[0-9]XX",app=".*"}
[0-9]*$/gm)
     31:   })
     32:
>>>  33:   xit('GET file upload metrics via public API', () => { // FIXME Flaky on CI/CD on at
least Windows
     34:     const file = path.resolve(__dirname, '../files/validSizeAndTypeForClient.pdf')
     35:     const form = frisby.formData()
     36:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
```

M	test/api/metricsApiSpec.ts **(Line 49)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     46:       })
     47:   })
     48:
>>>  49:   xit('GET file upload error metrics via public API', () => { // FIXME Flaky on CI/CD
on at least Windows
     50:     const file = path.resolve(__dirname, '../files/invalidSizeForServer.pdf')
     51:     const form = frisby.formData()
     52:     form.append('file', fs.createReadStream(file) as unknown as Blob) // casting to
blob as the frisby types are wrong and wont accept the fileStream type
```

M	frontend/src/app/last-login-ip last-login-ip.component.spec.ts **(Line 60)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     57:     expect(console.log).toHaveBeenCalled()
     58:   })
     59:
>>>  60:   xit('should set Last-Login IP from JWT as trusted HTML', () => { // FIXME Expected state seems to leak over from previous test case occasionally
     61:     localStorage.setItem('token', 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJkYXRhIjp7Imxhc3RMb2dpbklwIjoiMS4yLjMuNCJ9fQ.RAkmdqwNypuOxv3SDjPO4xMKvd1CddKvDFYDBfUt3bg')
     62:     component.ngOnInit()
     63:     expect(sanitizer.bypassSecurityTrustHtml).toHaveBeenCalledWith('<small>1.2.3.4</small>')
```

M

frontend/src/app/last-login-ip
last-login-ip.component.spec.ts
(Line 66)

typescript:S1134
Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     63:     expect(sanitizer.bypassSecurityTrustHtml).toHaveBeenCalledWith('<small>1.2.3.4</small>')
     64:   })
     65:
>>>  66:   xit('should not set Last-Login IP if none is present in JWT', () => { // FIXME Expected state seems to leak over from previous test case occasionally
     67:     localStorage.setItem('token', 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJkYXRhIjp7fX0.bVBhvll6IaeR3aUdoOeyR8YZe2S2DfhGAxTGfd9enLw')
     68:     component.ngOnInit()
     69:     expect(sanitizer.bypassSecurityTrustHtml).not.toHaveBeenCalled()
```

M

test/cypress/e2e/totpSetup.spec.ts
(Line 30)

typescript:S125
Remove this commented out code.

■ **Problematic Code:**

```
     27:       cy.get('#initialToken')
     28:         .should('have.attr', 'data-test-totp-secret')
     29:         .then(($val) => {
>>>  30:           // console.log($val);
     31:           cy.get('#currentPasswordSetup').type('K1f.....................')
     32:
     33:           cy.task<string>('GenerateAuthenticator', $val).then((secret: string) => {
```

M

test/cypress/e2e/profile.spec.ts
(Line 63)

typescript:S125
Remove this commented out code.

■ **Problematic Code:**

```
     60:             '/solve/challenges/server-side?key=tRy_H4rd3r_n0thIng_iS_Imp0ssibl3'
     61:           )
     62:           cy.visit('/')
>>>  63:           // void browser.driver.sleep(10000);
     64:           // void browser.waitForAngularEnabled(true);
     65:           cy.expectChallengeSolved({ challenge: 'SSTi' })
     66:         }
```

M

test/cypress/e2e/profile.spec.ts
(Line 72)

typescript:S1134
Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     69:   })
     70:
     71:   describe('challenge "csrf"', () => {
>>>  72:     // FIXME Only works on Chrome <80 but Protractor uses latest Chrome version. Test can probably never be turned on again.
     73:     xit('should be possible to perform a CSRF attack against the user profile page', () => {
     74:       cy.visit('http://htmledit.squarefree.com')
     75:       /* The script executed below is equivalent to pasting this string into http://htmledit.squarefree.com: */
```

M	lib/startup/validateChatBot.ts **(Line 36)**	**typescript:S6660** 'If' statement should not be the only statement in 'else' block

■ **Problematic Code:**

```
     33:     logger.warn(`Intent ${colors.italic(intent)} is missing in chatbot training data (${colors.red('NOT OK')})`)
     34:     success = false
     35:   } else {
>>>  36:     if (intentData[0].answers.filter((answer: { action: string, handler: string }) => answer.action === 'function' && answer.handler === handler).length === 0) {
     37:       logger.warn(`Answer with ${colors.italic('function')} action and handler ${colors.italic(handler)} is missing for intent ${colors.italic(intent)} (${colors.red('NOT OK')})`)
     38:       success = false
     39:     }
```

M	rsn/rsnUtil.ts **(Line 129)**	**typescript:S4043** Move this array "sort" operation to a separate statement or replace it with "toSorted".

■ **Problematic Code:**

```
    126: function checkData (data: CacheData, fileData: CacheData) {
    127:   const filesWithDiff = []
    128:   for (const key in data) {
>>> 129:     const fileDataValueAdded = fileData[key].added.sort((a, b) => a - b)
    130:     const dataValueAdded = data[key].added.sort((a, b) => a - b)
    131:     const fileDataValueRemoved = fileData[key].added.sort((a, b) => a - b)
    132:     const dataValueAddedRemoved = data[key].added.sort((a, b) => a - b)
```

M	rsn/rsnUtil.ts **(Line 130)**	**typescript:S4043** Move this array "sort" operation to a separate statement or replace it with "toSorted".

■ **Problematic Code:**

```
    127:   const filesWithDiff = []
    128:   for (const key in data) {
    129:     const fileDataValueAdded = fileData[key].added.sort((a, b) => a - b)
>>> 130:     const dataValueAdded = data[key].added.sort((a, b) => a - b)
    131:     const fileDataValueRemoved = fileData[key].added.sort((a, b) => a - b)
    132:     const dataValueAddedRemoved = data[key].added.sort((a, b) => a - b)
    133:     if (fileDataValueAdded.length === dataValueAdded.length && fileDataValueRemoved.length === dataValueAddedRemoved.length) {
```

M	rsn/rsnUtil.ts **(Line 131)**	**typescript:S4043** Move this array "sort" operation to a separate statement or replace it with "toSorted".

■ Problematic Code:

```
    128:   for (const key in data) {
    129:     const fileDataValueAdded = fileData[key].added.sort((a, b) => a - b)
    130:     const dataValueAdded = data[key].added.sort((a, b) => a - b)
>>> 131:     const fileDataValueRemoved = fileData[key].added.sort((a, b) => a - b)
    132:     const dataValueAddedRemoved = data[key].added.sort((a, b) => a - b)
    133:     if (fileDataValueAdded.length === dataValueAdded.length &&
fileDataValueRemoved.length === dataValueAddedRemoved.length) {
    134:       if (!dataValueAdded.every((val: number, ind: number) => fileDataValueAdded[ind]
=== val)) {
```

M	rsn/rsnUtil.ts **(Line 132)**	**typescript:S4043** Move this array "sort" operation to a separate statement or replace it with "toSorted".

■ Problematic Code:

```
    129:     const fileDataValueAdded = fileData[key].added.sort((a, b) => a - b)
    130:     const dataValueAdded = data[key].added.sort((a, b) => a - b)
    131:     const fileDataValueRemoved = fileData[key].added.sort((a, b) => a - b)
>>> 132:     const dataValueAddedRemoved = data[key].added.sort((a, b) => a - b)
    133:     if (fileDataValueAdded.length === dataValueAdded.length &&
fileDataValueRemoved.length === dataValueAddedRemoved.length) {
    134:       if (!dataValueAdded.every((val: number, ind: number) => fileDataValueAdded[ind]
=== val)) {
    135:         console.log(colors.red(key))
```

M	lib/utils.ts **(Line 41)**	**typescript:S6557** Use 'String#startsWith' method instead.

■ Problematic Code:

```
     38:   return startsWith(url, 'http')
     39: }
     40:
>>>  41: export const startsWith = (str: string, prefix: string) => str ? str.indexOf(prefix)
=== 0 : false
     42:
     43: export const endsWith = (str?: string, suffix?: string) => (str && suffix) ?
str.includes(suffix, str.length - suffix.length) : false
     44:
```

M	frontend/src/environments/environment.ts **(Line 21)**	**typescript:S125** Remove this commented out code.

■ Problematic Code:

```
     18:  * import the following file, but please comment it out in production mode
     19:  * because it will have performance impact when throw error
     20:  */
>>>  21: // import 'zone.js/plugins/zone-error';  // Included with Angular CLI.
     22:
```

M	frontend/src/hacking-instructor challenges/codingChallenges.ts **(Line 20)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     17:       fixture: 'app-navbar',
     18:       fixtureAfter: true,
     19:       unskippable: true,
>>>  20:       resolved: waitForAngularRouteToBeVisited('score-board') // FIXME The tutorial
does not progress automatically. Workaround ^^^^^^^^^^^^^^^^ instruction above should be removed
when fixed.
     21:     },
     22:     {
     23:       text:
```

M	data/static/codefixes dbSchemaChallenge_1.ts **(Line 7)**	**typescript:S1854** Remove this useless assignment to variable "dataString".

■ **Problematic Code:**

```
      4:     criteria = (criteria.length <= 200) ? criteria : criteria.substring(0, 200)
      5:     models.sequelize.query("SELECT * FROM Products WHERE ((name LIKE '%"+criteria+"%'
OR description LIKE '%"+criteria+"%') AND deletedAt IS NULL) ORDER BY name")
      6:       .then(([products]: any) => {
>>>   7:         const dataString = JSON.stringify(products)
      8:         for (let i = 0; i < products.length; i++) {
      9:           products[i].name = req.__(products[i].name)
     10:           products[i].description = req.__(products[i].description)
```

M	data/static/codefixes dbSchemaChallenge_2_correct.ts **(Line 9)**	**typescript:S1854** Remove this useless assignment to variable "dataString".

■ **Problematic Code:**

```
      6:         `SELECT * FROM Products WHERE ((name LIKE '%:criteria%' OR description LIKE
'%:criteria%') AND deletedAt IS NULL) ORDER BY name`,
      7:         { replacements: { criteria } }
      8:       ).then(([products]: any) => {
>>>   9:         const dataString = JSON.stringify(products)
     10:         for (let i = 0; i < products.length; i++) {
     11:           products[i].name = req.__(products[i].name)
     12:           products[i].description = req.__(products[i].description)
```

M	data/static/codefixes dbSchemaChallenge_3.ts **(Line 13)**	**typescript:S1854** Remove this useless assignment to variable "dataString".

■ **Problematic Code:**

```
    10:     }
    11:     models.sequelize.query(`SELECT * FROM Products WHERE ((name LIKE '%${criteria}%' OR description LIKE '%${criteria}%') AND deletedAt IS NULL) ORDER BY name`)
    12:       .then(([products]: any) => {
>>> 13:         const dataString = JSON.stringify(products)
    14:         for (let i = 0; i < products.length; i++) {
    15:           products[i].name = req.__(products[i].name)
    16:           products[i].description = req.__(products[i].description)
```

M	data/static/codefixes unionSqlInjectionChallenge_1.ts **(Line 8)**	**typescript:S1854** Remove this useless assignment to variable "dataString".

■ **Problematic Code:**

```
     5:     criteria.replace(/"|'|;|and|or/i, "")
     6:     models.sequelize.query(`SELECT * FROM Products WHERE ((name LIKE '%${criteria}%' OR description LIKE '%${criteria}%') AND deletedAt IS NULL) ORDER BY name`)
     7:       .then(([products]: any) => {
>>>  8:         const dataString = JSON.stringify(products)
     9:         for (let i = 0; i < products.length; i++) {
    10:           products[i].name = req.__(products[i].name)
    11:           products[i].description = req.__(products[i].description)
```

M	data/static/codefixes unionSqlInjectionChallenge_2_correct.ts **(Line 9)**	**typescript:S1854** Remove this useless assignment to variable "dataString".

■ **Problematic Code:**

```
     6:         `SELECT * FROM Products WHERE ((name LIKE '%:criteria%' OR description LIKE '%:criteria%') AND deletedAt IS NULL) ORDER BY name`,
     7:         { replacements: { criteria } }
     8:       ).then(([products]: any) => {
>>>  9:         const dataString = JSON.stringify(products)
    10:         for (let i = 0; i < products.length; i++) {
    11:           products[i].name = req.__(products[i].name)
    12:           products[i].description = req.__(products[i].description)
```

M	data/static/codefixes unionSqlInjectionChallenge_3.ts **(Line 12)**	**typescript:S1854** Remove this useless assignment to variable "dataString".

■ **Problematic Code:**

```
     9:     }
    10:     models.sequelize.query(`SELECT * FROM Products WHERE ((name LIKE '%${criteria}%' OR description LIKE '%${criteria}%') AND deletedAt IS NULL) ORDER BY name`)
    11:       .then(([products]: any) => {
>>> 12:         const dataString = JSON.stringify(products)
    13:         for (let i = 0; i < products.length; i++) {
    14:           products[i].name = req.__(products[i].name)
    15:           products[i].description = req.__(products[i].description)
```

M	data/static/codefixes noSqlReviewsChallenge_1.ts **(Line 3)**	**typescript:S1854** Remove this useless assignment to variable "user".

■ **Problematic Code:**

```
      1: export function updateProductReviews () {
      2:   return (req: Request, res: Response, next: NextFunction) => {
>>>   3:     const user = security.authenticatedUsers.from(req)
      4:
      5:     if (req.body.id['$ne'] !== undefined) {
      6:       res.status(400).send()
```

M	data/static/codefixes noSqlReviewsChallenge_2.ts **(Line 3)**	**typescript:S1854** Remove this useless assignment to variable "user".

■ **Problematic Code:**

```
      1: export function updateProductReviews () {
      2:   return (req: Request, res: Response, next: NextFunction) => {
>>>   3:     const user = security.authenticatedUsers.from(req)
      4:     db.reviewsCollection.update(
      5:       { _id: req.body.id },
      6:       { $set: { message: req.body.message } }
```

M	data/static/codefixes noSqlReviewsChallenge_3_correct.ts **(Line 3)**	**typescript:S1854** Remove this useless assignment to variable "user".

■ **Problematic Code:**

```
      1: export function updateProductReviews () {
      2:   return (req: Request, res: Response, next: NextFunction) => {
>>>   3:     const user = security.authenticatedUsers.from(req)
      4:
      5:     if (typeof req.body.id !== 'string') {
      6:       res.status(400).send()
```

M	data/static/codefixes forgedReviewChallenge_3.ts **(Line 3)**	**typescript:S1854** Remove this useless assignment to variable "user".

■ **Problematic Code:**

```
      1: export function updateProductReviews () {
      2:   return (req: Request, res: Response, next: NextFunction) => {
>>>   3:     const user = security.authenticatedUsers.from(req)
      4:     db.reviewsCollection.update(
      5:       { _id: req.body.id },
      6:       { $set: { message: req.body.message } }
```

M	test/api/loginApiSpec.ts **(Line 261)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    258:        })
    259:    })
    260:
>>> 261:    xit('GET last login IP will be saved as remote IP when True-Client-IP is not
present', () => { // FIXME Started to fail regularly on CI under Linux
    262:      return frisby.post(REST_URL + '/user/login', {
    263:        headers: jsonHeader,
    264:        body: {
```

M	test/api/productApiSpec.ts **(Line 103)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
    100:        .expect('json', 'data', { description: '<a href="http://kimminich.de"
target="_blank">More...</a>' })
    101:    })
    102:
>>> 103:    xit('PUT update existing product does not filter XSS attacks', () => { // FIXME
Started to fail regularly on CI under Linux
    104:      return frisby.put(API_URL + '/Products/1', {
    105:        header: jsonHeader,
    106:        body: {
```

M	test/server/b2bOrderSpec.ts **(Line 31)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     28:      challenges.rceChallenge = { solved: false, save } as unknown as Challenge
     29:    })
     30:
>>>  31:    xit('infinite loop payload does not succeed but solves "rceChallenge"', () => { //
FIXME Started failing on Linux regularly
     32:      req.body.orderLinesData = '(function dos() { while(true); })()'
     33:
     34:      b2bOrder()(req, res, next)
```

M	frontend/src/assets/public/css dataErasure.css **(Line 104)**	**css:S4666** Unexpected duplicate selector "img", first used at line 86

■ **Problematic Code:**

```
    101: .home-page {
    102:    margin-top: 30px;
    103: }
>>> 104: img {
    105:    padding-top: 5px;
    106:    padding-left: 50px;
    107: }
```

M	test/server/insecuritySpec.ts **(Line 58)**	**typescript:S4623** Remove this redundant "undefined".

■ Problematic Code:

```
     55:
     56:   describe('discountFromCoupon', () => {
     57:     it('returns undefined when not passing in a coupon code', () => {
>>>  58:       expect(security.discountFromCoupon(undefined)).to.equal(undefined)
     59:     })
     60:
     61:     it('returns undefined for malformed coupon code', () => {
```

M	test/server/insecuritySpec.ts **(Line 145)**	**typescript:S4623** Remove this redundant "undefined".

■ Problematic Code:

```
    142:   describe('sanitizeLegacy', () => {
    143:     it('returns empty string for undefined input', () => {
    144:       expect(security.sanitizeLegacy()).to.equal('')
>>> 145:       expect(security.sanitizeLegacy(undefined)).to.equal('')
    146:     })
    147:
    148:     it('returns input unchanged for plain text input', () => {
```

M	frontend/src/app/two-factor-auth-enter two-factor-auth-enter.component.spec.ts **(Line 121)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
    118:     expect(sessionStorage.getItem('bid')).toBe('42')
    119:   })
    120:
>>> 121:   xit('should notify about user login after 2FA verification', () => { // FIXME Spy call is not registered at all
    122:     component.verify()
    123:
    124:     expect(userService.isLoggedIn.next).toHaveBeenCalledWith(true)
```

M	frontend/src/hacking-instructor/helpers helpers.ts **(Line 63)**	**typescript:S1871** This branch's code block is the same as the block for the branch on line 61.

■ Problematic Code:

```
     60:     while (true) {
     61:       if (options.ignoreCase && inputElement.value.toLowerCase() !== value.toLowerCase()) {
     62:         break
>>>  63:       } else if (!options.ignoreCase && inputElement.value !== value) {
     64:         break
     65:       }
     66:       await sleep(100)
```

M	frontend/src/hacking-instructor/helpers helpers.ts **(Line 46)**	**typescript:S1871** This branch's code block is the same as the block for the branch on line 44.

■ Problematic Code:

```
     43:     while (true) {
     44:       if (options.ignoreCase && inputElement.value.toLowerCase() ===
value.toLowerCase()) {
     45:         break
>>>  46:       } else if (!options.ignoreCase && inputElement.value === value) {
     47:         break
     48:       }
     49:       await sleep(100)
```

M	frontend/src/hacking-instructor/helpers helpers.ts **(Line 81)**	**typescript:S1871** This branch's code block is the same as the block for the branch on line 79.

■ Problematic Code:

```
     78:       if (inputElement.value !== '') {
     79:         if (options.ignoreCase && inputElement.value.toLowerCase() !==
value.toLowerCase()) {
     80:           break
>>>  81:         } else if (!options.ignoreCase && inputElement.value !== value) {
     82:           break
     83:         }
     84:       }
```

M	test/api/internetResourcesSpec.ts **(Line 17)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
     14:         .expect('bodyContains', 'this coupled with Eurogium Edule was sometimes found
fatal')
     15:     })
     16:
>>>  17:     xit('for 7MS configuration (https://pastebin.com/8SMbWPxc)', () => { // FIXME Test
would need to confirm/bypass PasteBin SMART filter to retrieve content
     18:       return frisby.get('https://pastebin.com/8SMbWPxc')
     19:         .expect('status', 200)
     20:         .expect('bodyContains', 'TAYLOR SWIFT')
```

M	test/api/internetResourcesSpec.ts **(Line 34)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     31:         .expect('bodyContains', 'Note to self: Option (1) and (3) of the above should
not be combined.')
     32:     })
     33:
>>>  34:     xit('for Mozilla configuration (https://pastebin.com/t8jqE1y7)', () => { // FIXME
Test would need to confirm/bypass PasteBin SMART filter to retrieve content
     35:       return frisby.get('https://pastebin.com/t8jqE1y7')
     36:         .expect('status', 200)
     37:         .expect('bodyContains', 'Fixed a bug that, when this plugin was installed
together with both the')
```

M	test/api/userProfileSpec.ts **(Line 59)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ **Problematic Code:**

```
     56:       .expect('status', 302)
     57:   })
     58:
>>>  59:   xit('POST update username is forbidden for unauthenticated user', () => { // FIXME
runs into "socket hang up"
     60:     const form = frisby.formData()
     61:     form.append('username', 'Localhorst')
     62:
```

M	lib/startup/validateConfig.ts **(Line 114)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ **Problematic Code:**

```
    111:     // @ts-expect-error FIXME any type issue
    112:     const appliedSpecials = specialProducts.filter(({ key }) => product[key])
    113:     if (appliedSpecials.length > 1) {
>>> 114:       logger.warn(`Product ${colors.italic(product.name)} is used as
${appliedSpecials.map(({ name }) => `${colors.italic(name)}`).join(' and ')} but can only be
used for one challenge (${colors.red('NOT OK')})`)
    115:       success = false
    116:     }
    117:   })
```

M	lib/startup/validateConfig.ts **(Line 162)**	**typescript:S4624** Refactor this code to not use nested template literals.

■ **Problematic Code:**

```
    159:   memories.forEach((memory) => {
    160:     const appliedSpecials = specialMemories.filter(({ keys }) => memory[keys[0]] &&
memory[keys[1]])
    161:     if (appliedSpecials.length > 1) {
>>> 162:       logger.warn(`Memory ${colors.italic(memory.caption)} is used as
${appliedSpecials.map(({ name }) => `${colors.italic(name)}`).join(' and ')} but can only be
used for one challenge (${colors.red('NOT OK')})`)
    163:       success = false
    164:     }
    165:   })
```

M	test/api/profileImageUploadSpec.ts **(Line 141)**	**typescript:S1134** Take the required action to fix the issue indicated by this comment.

■ Problematic Code:

```
    138:       })
    139:   })
    140:
>>> 141:   xit('POST profile image URL forbidden for anonymous user', () => { // FIXME runs into "socket hang up"
    142:     const form = frisby.formData()
    143:     form.append('imageUrl', 'cataas.com/cat')
    144:
```

M	test/files/decrypt.py **(Line 9)**	**python:PrintStatementUsage** Replace print statement by built-in function.

■ Problematic Code:

```
      6: d = 8948942500927444436822854592177309391966958606588425744549785445648767483962981839
09349419732628796167979706089172836798754993315741611138540888132754881105882471930775825272784
37906504015680623423550067240042466665654232383502922215493623289472138866445818789127946123407
807725702626644091036502372545139713
      7:
      8: with open('announcement_encrypted.md', 'r') as fl:
>>>   9:     print "".join([chr(pow(int(f[:-1]), d, N)) for f in fl.readlines()])
     10:
```

M	test/files/decrypt_bruteforce.py **(Line 15)**	**python:PrintStatementUsage** Replace print statement by built-in function.

■ Problematic Code:

```
     12:     encrypted_chars[str(c)] = char
     13:
     14: with open('announcement_encrypted.md', 'r') as fl:
>>>  15:     print "".join([encrypted_chars[f[:-1]] for f in fl.readlines()])
     16:
```

M	frontend/src/app/deluxe-user deluxe-user.component.ts **(Line 61)**	**typescript:S6557** Use 'String#startsWith' method instead.

■ Problematic Code:

```
     58:           if (config.application.logo) {
     59:             let logo: string = config.application.logo
     60:
>>>  61:             if (logo.substring(0, 4) === 'http') {
     62:               logo = decodeURIComponent(logo.substring(logo.lastIndexOf('/') + 1))
     63:             }
     64:             this.logoSrc = `assets/public/images/${decalParam || logo}`
```

M	test/smoke/Dockerfile **(Line 1)**	**docker:S6596** Use a specific version tag for the image.

■ **Problematic Code:**

```
>>>    1: FROM alpine
       2:
       3: RUN apk add curl
       4:
```

M	test/smoke/Dockerfile **(Line 3)**	**docker:S6587** Remove cache after installing packages or store it in a cache mount.

■ **Problematic Code:**

```
       1: FROM alpine
       2:
>>>    3: RUN apk add curl
       4:
       5: COPY smoke-test.sh smoke-test.sh
       6:
```

M	frontend/src/app/photo-wall mime-type.validator.ts **(Line 38)**	**typescript:S4165** Review this redundant assignment: "isValid" already holds the assigned value along all execution paths.

■ **Problematic Code:**

```
      35:             isValid = true
      36:             break
      37:           default:
>>>   38:             isValid = false
      39:             break
      40:         }
      41:         if (isValid) {
```

M	frontend/src/app/search-result search-result.component.spec.ts **(Line 264)**	**typescript:S4623** Remove this redundant "undefined".

■ **Problematic Code:**

```
     261:   it('should not add anything to basket on error retrieving basket', fakeAsync(() => {
     262:     basketService.find.and.returnValue(throwError('Error'))
     263:     sessionStorage.setItem('bid', '815')
>>>  264:     component.addToBasket(undefined)
     265:     expect(snackBar.open).not.toHaveBeenCalled()
     266:   }))
     267:
```

M	frontend/src/app/search-result search-result.component.html **(Line 40)**	**Web:S6819** Use or instead of the button role to ensure accessibility across all devices.

■ **Problematic Code:**

```
     37:                   }
     38:                   <div class="product" (click)="showDetail(item)" aria-label="Click for
more information about the product"
     39:                     matTooltip="Click for more information" matTooltipPosition="above">
>>>  40:                     <img mat-card-image
[src]="'assets/public/images/products/'+item.image" alt={{item.name}}
     41:                       class="img-responsive img-thumbnail" role="button">
     42:                       <div class="info-box">
     43:                         <div class="item-name">{{item.name}}</div>
```

M

frontend/src/app/search-result
search-result.component.html
(Line 40)

Web:S6842
Non-interactive elements should not be assigned interactive roles.

■ **Problematic Code:**

```
     37:                   }
     38:                   <div class="product" (click)="showDetail(item)" aria-label="Click for
more information about the product"
     39:                     matTooltip="Click for more information" matTooltipPosition="above">
>>>  40:                     <img mat-card-image
[src]="'assets/public/images/products/'+item.image" alt={{item.name}}
     41:                       class="img-responsive img-thumbnail" role="button">
     42:                       <div class="info-box">
     43:                         <div class="item-name">{{item.name}}</div>
```

M

routes/imageCaptcha.ts
(Line 39)

typescript:S3358
Extract this nested ternary operation into an independent statement.

■ **Problematic Code:**

```
     36:
     37: export const verifyImageCaptcha = () => (req: Request, res: Response, next:
NextFunction) => {
     38:   const user = security.authenticatedUsers.from(req)
>>>  39:   const UserId = user ? user.data ? user.data.id : undefined : undefined
     40:   ImageCaptchaModel.findAll({
     41:     limit: 1,
     42:     where: {
```

M

routes/languages.ts
(Line 41)

typescript:S3358
Extract this nested ternary operation into an independent statement.

■ **Problematic Code:**

```
     38:               icons: locale?.icons,
     39:               shortKey: locale?.shortKey,
     40:               percentage,
>>>  41:               gauge: (percentage > 90 ? 'full' : (percentage > 70 ? 'three-quarters' :
(percentage > 50 ? 'half' : (percentage > 30 ? 'quarter' : 'empty'))))
     42:             }
     43:             if (!(fileName === 'en.json' || fileName === 'tlh_AA.json')) {
     44:               languages.push(lang)
```

M

routes/languages.ts
(Line 41)

typescript:S3358
Extract this nested ternary operation into an independent statement.

■ **Problematic Code:**

```
     38:                 icons: locale?.icons,
     39:                 shortKey: locale?.shortKey,
     40:                 percentage,
>>>  41:                 gauge: (percentage > 90 ? 'full' : (percentage > 70 ? 'three-quarters' :
(percentage > 50 ? 'half' : (percentage > 30 ? 'quarter' : 'empty'))))
     42:               }
     43:               if (!(fileName === 'en.json' || fileName === 'tlh_AA.json')) {
     44:                 languages.push(lang)
```

M	routes/languages.ts **(Line 41)**	**typescript:S3358** Extract this nested ternary operation into an independent statement.

■ **Problematic Code:**

```
     38:                 icons: locale?.icons,
     39:                 shortKey: locale?.shortKey,
     40:                 percentage,
>>>  41:                 gauge: (percentage > 90 ? 'full' : (percentage > 70 ? 'three-quarters' :
(percentage > 50 ? 'half' : (percentage > 30 ? 'quarter' : 'empty'))))
     42:               }
     43:               if (!(fileName === 'en.json' || fileName === 'tlh_AA.json')) {
     44:                 languages.push(lang)
```

M	lib/startup/validatePreconditions.ts **(Line 105)**	**typescript:S6660** 'If' statement should not be the only statement in 'else' block

■ **Problematic Code:**

```
    102:        if (error) {
    103:          reject(error)
    104:        } else {
>>> 105:          if (status === 'open') {
    106:            logger.warn(`Port ${colors.bold(port.toString())} is in use
(${colors.red('NOT OK')})`)
    107:            resolve(false)
    108:          } else {
```

M	frontend/src/app/two-factor-auth-enter two-factor-auth-enter.component.ts **(Line 61)**	**typescript:S125** Remove this commented out code.

■ **Problematic Code:**

```
     58:          this.cookieService.put('token', authentication.token, { expires })
     59:          sessionStorage.setItem('bid', authentication.bid?.toString())
     60:          /* Use userService to notifiy if user has logged in */
>>>  61:          /* this.userService.isLoggedIn = true; */
     62:          this.userService.isLoggedIn.next(true)
     63:          this.ngZone.run(async () => await this.router.navigate(['/search']))
     64:        },
```

M	frontend/src/app/token-sale token-sale.component.html **(Line 96)**	**Web:S6850** Headings must have content and the content must be accessible by a screen reader.

■ Problematic Code:

```
     93:
     94:        <mat-card appearance="outlined" class="mat-elevation-z6">
     95:          <div class="mdc-card">
>>>  96:            <h5><i class='fas fa-comments fa-2x'></i> <span style="margin-left: 10px;"
[innerHtml]="'ICO_FAQ_QUESTION' | translate:{juicycoin: altcoinName}"></span></h5>
     97:            <small class="text-justify">{{"ICO_FAQ_ANSWER" | translate}}</small>
     98:          </div>
     99:        </mat-card>
```

M	frontend/src/app/user-details user-details.component.html **(Line 18)**	**Web:AvoidCommentedOutCodeCheck** Remove this commented out code.

■ Problematic Code:

```
     15:          <strong translate="LABEL_EMAIL"></strong>
     16:          <p>{{user?.email}}</p>
     17:        </div>
>>>  18:        <!--<div>
     19:          <strong translate="LABEL_PASSWORD"></strong>
     20:          <p>{{user?.password}}</p>
     21:        </div>-->
```

M	routes/order.ts **(Line 40)**	**typescript:S3358** Extract this nested ternary operation into an independent statement.

■ Problematic Code:

```
     37:        .then(async (basket: BasketModel | null) => {
     38:          if (basket != null) {
     39:            const customer = security.authenticatedUsers.from(req)
>>>  40:            const email = customer ? customer.data ? customer.data.email : '' : ''
     41:            const orderId = security.hash(email).slice(0, 4) + '-' +
utils.randomHexString(16)
     42:            const pdfFile = `order_${orderId}.pdf`
     43:            const doc = new PDFDocument()
```

M	frontend/src/app/navbar navbar.component.ts **(Line 132)**	**typescript:S6557** Use 'String#startsWith' method instead.

■ Problematic Code:

```
    129:          if (config?.application?.logo) {
    130:            let logo: string = config.application.logo
    131:
>>> 132:            if (logo.substring(0, 4) === 'http') {
    133:              logo = decodeURIComponent(logo.substring(logo.lastIndexOf('/') + 1))
    134:            }
    135:            this.logoSrc = 'assets/public/images/' + logo
```

M	server.ts **(Line 203)**	**typescript:S6397** Replace this character class by the character itself.

■ **Problematic Code:**

```
    200:
    201:   /* Remove duplicate slashes from URL which allowed bypassing subsequent filters */
    202:   app.use((req: Request, res: Response, next: NextFunction) => {
>>> 203:     req.url = req.url.replace(/[/]+/g, '/')
    204:     next()
    205:   })
    206:
```

L

server.ts
(Line 551)

typescript:S4138
Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ **Problematic Code:**

```
    548:     // translate hints on-the-fly
    549:     if (name === 'Hint') {
    550:       resource.list.fetch.after((req: Request, res: Response, context: { instance: 
string | any[], continue: any }) => {
>>> 551:         for (let i = 0; i < context.instance.length; i++) {
    552:           context.instance[i].text = req.__(context.instance[i].text)
    553:         }
    554:         return context.continue
```

L

frontend/src/app/change-password
change-password.component.ts
(Line 92)

typescript:S6594
Use the "RegExp.exec()" method instead.

■ **Problematic Code:**

```
     89:
     90:   changePassword () {
     91:     if (
>>>  92:       localStorage.getItem('email')?.match(/support@.*/) &&
     93:       !this.newPasswordControl.value.match(
     94:         /(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-z\d@$!%*?&]{12,30}/
     95:       )
```

L

frontend/src/app/search-result
search-result.component.ts
(Line 199)

typescript:S4138
Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ **Problematic Code:**

```
    196:       next: (basket) => {
    197:         const productsInBasket: any = basket.Products
    198:         let found = false
>>> 199:         for (let i = 0; i < productsInBasket.length; i++) {
    200:           if (productsInBasket[i].id === id) {
    201:             found = true
    202:             this.basketService.get(productsInBasket[i].BasketItem.id).subscribe({
```

L

routes/deluxe.ts
(Line 51)

typescript:S2486
Handle this exception or don't catch it at all.

■ Problematic Code:

```
     48:         const updatedToken = security.authorize(userWithStatus)
     49:         security.authenticatedUsers.put(updatedToken, userWithStatus)
     50:         res.status(200).json({ status: 'success', data: { confirmation:
'Congratulations! You are now a deluxe member!', token: updatedToken } })
>>>  51:       } catch (error) {
     52:         res.status(400).json({ status: 'error', error: 'Something went wrong. Please
try again!' })
     53:       }
     54:     } catch (err: unknown) {
```

L	test/server/webhookSpec.ts **(Line 23)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ Problematic Code:

```
     20:     it('ignores errors where no webhook URL is provided via environment variable',
async () => {
     21:       try {
     22:         await webhook.notify(challenge)
>>>  23:       } catch (error) {
     24:         chai.assert.fail('webhook.notify should not throw an error when no webhook URL
is provided')
     25:       }
     26:     })
```

L	routes/likeProductReviews.ts **(Line 58)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ Problematic Code:

```
     55:       } catch (err) {
     56:         res.status(500).json(err)
     57:       }
>>>  58:     } catch (err) {
     59:       res.status(400).json({ error: 'Wrong Params' })
     60:     }
     61:   }
```

L	frontend/src/app/web3-sandbox web3-sandbox.component.ts **(Line 312)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ Problematic Code:

```
    309:       }
    310:       console.log('session', this.session)
    311:       this.changeDetectorRef.detectChanges()
>>> 312:     } catch (err) {
    313:       console.log('An error occurred')
    314:     }
    315:   }
```

L	frontend/src/app/password-strength password-strength.component.ts **(Line 59)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     56:   }
     57:
     58:   get containAtLeastOneDigit (): boolean {
>>>  59:     return /^(?=.*?[0-9])/.test(this.password)
     60:   }
     61:
     62:   get containAtLeastOneSpecialChar (): boolean {
```

L	frontend/src/main.ts **(Line 9)**	**typescript:S1128** Remove this unused import of 'AppModule'.

■ **Problematic Code:**

```
      6: import { enableProdMode, importProvidersFrom } from '@angular/core'
      7: import { platformBrowserDynamic } from '@angular/platform-browser-dynamic'
      8:
>>>   9: import { AppModule, HttpLoaderFactory } from './app/app.module'
     10: import { environment } from './environments/environment'
     11: import { AppComponent } from './app/app.component';
     12: import { MatAutocompleteModule } from '@angular/material/autocomplete';
```

L	frontend/src/main.ts **(Line 51)**	**typescript:S1128** Remove this unused import of 'TranslateHttpLoader'.

■ **Problematic Code:**

```
     48: import { provideAnimations } from '@angular/platform-browser/animations';
     49: import { ReactiveFormsModule, FormsModule } from '@angular/forms';
     50:
>>>  51: import { TranslateHttpLoader } from '@ngx-translate/http-loader';
     52: import { TranslateModule, TranslateLoader } from '@ngx-translate/core';
     53: import { Routing } from './app/app.routing';
     54: import { BrowserModule, bootstrapApplication } from '@angular/platform-browser';
```

L	lib/startup/validateDependenciesBasic.ts **(Line 12)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
      9:   try {
     10:     // @ts-expect-error FIXME due to non-existing type definitions for
check-dependencies
     11:     await import('check-dependencies')
>>>  12:   } catch (err) {
     13:     console.error('Please run "npm install" before starting the application!')
     14:     process.exit(1)
     15:   }
```

L	frontend/src/hacking-instructor/index.ts **(Line 201)**	**typescript:S6571** 'unknown' overrides all other types in this union type.

Problematic Code:

```
    198:     }
    199:
    200:     // eslint-disable-next-line @typescript-eslint/no-invalid-void-type
>>> 201:     const continueConditions: Array<Promise<void | unknown>> = [
    202:       hint.resolved()
    203:     ]
    204:
```

L	routes/b2bOrder.ts **(Line 26)**	**typescript:S6594** Use the "RegExp.exec()" method instead.

Problematic Code:

```
     23:         vm.runInContext('safeEval(orderLinesData)', sandbox, { timeout: 2000 })
     24:         res.json({ cid: body.cid, orderNo: uniqueOrderNumber(), paymentDue:
dateTwoWeeksFromNow() })
     25:       } catch (err) {
>>>  26:         if (utils.getErrorMessage(err).match(/Script execution timed out.*/) != null) {
     27:           challengeUtils.solveIf(challenges.rceOccupyChallenge, () => { return true })
     28:           res.status(503)
     29:           next(new Error('Sorry, we are temporarily not available! Please try again
later.'))
```

L	test/cypress/support/commands.ts **(Line 36)**	**typescript:S6594** Use the "RegExp.exec()" method instead.

Problematic Code:

```
     33:   'login',
     34:   (context: { email: string, password: string, totpSecret?: string }) => {
     35:     cy.visit('/#/login')
>>>  36:     if (context.email.match(/\S+@\S+\.\S+/) != null) {
     37:       cy.get('#email').type(context.email)
     38:     } else {
     39:       cy.task<string>('GetFromConfig', 'application.domain').then(
```

L	lib/is-docker.ts **(Line 27)**	**typescript:S6606** Prefer using nullish coalescing operator (`??=`) instead of an assignment expression, as it is simpler to read.

Problematic Code:

```
     24:
     25: export default function isDocker () {
     26:   // TODO: Use `??=` when targeting Node.js 16.
>>>  27:   if (isDockerCached === undefined) {
     28:     isDockerCached = hasDockerEnv() || hasDockerCGroup()
     29:   }
     30:   return isDockerCached
```

L	routes/chatbot.ts **(Line 111)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
    108:     } else {
    109:       res.status(200).json(response)
    110:     }
>>> 111:   } catch (err) {
    112:     try {
    113:       await bot.respond(testCommand, `${user.id}`)
    114:       res.status(200).json({
```

L	test/server/blueprintSpec.ts **(Line 58)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
     55:               for (const property of product.exifForBlueprintChallenge) {
     56:                 expect(properties).to.include(property)
     57:               }
>>>  58:             } catch (error) {
     59:               expect.fail(`Could not read EXIF data from ${pathToImage}`)
     60:             }
     61:           }
```

L	lib/codingChallenges.ts **(Line 37)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
     34:         ) {
     35:           matches.push({ path: currPath, content: code })
     36:         }
>>>  37:       } catch (e) {
     38:         logger.warn(`File ${currPath} could not be read. it might have been moved or
deleted. If coding challenges are contained in the file, they will not be available.`)
     39:       }
     40:     }
```

L	lib/codingChallenges.ts **(Line 59)**	**typescript:S6594** Use the "RegExp.exec()" method instead.

■ **Problematic Code:**

```
     56: }
     57:
     58: function getCodingChallengeFromFileContent (source: string, challengeKey: string) {
>>>  59:   const snippets = source.match(`[/#]{0,2} vuln-code-snippet
start.*${challengeKey}([^])*vuln-code-snippet end.*${challengeKey}`)
     60:   if (snippets == null) {
     61:     throw new BrokenBoundary('Broken code snippet boundaries for: ' + challengeKey)
     62:   }
```

L	routes/chatbot.ts **(Line 71)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
     68:         action: 'response',
     69:         body: bot.greet(`${user.id}`)
     70:       })
>>>  71:     } catch (err) {
     72:       next(new Error('Blocked illegal activity by ' + req.socket.remoteAddress))
     73:     }
     74:     return
```

L	routes/chatbot.ts **(Line 81)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
     78:     bot.addUser(`${user.id}`, username)
     79:     try {
     80:       bot.addUser(`${user.id}`, username)
>>>  81:     } catch (err) {
     82:       next(new Error('Blocked illegal activity by ' + req.socket.remoteAddress))
     83:       return
     84:     }
```

L	routes/chatbot.ts **(Line 199)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
    196:         status: bot.training.state,
    197:         body: bot.training.state ? bot.greet(`${user.id}`) :
`${config.get<string>('application.chatBot.name')} isn't ready at the moment, please wait while
I set things up`
    198:       })
>>> 199:     } catch (err) {
    200:       next(new Error('Blocked illegal activity by ' + req.socket.remoteAddress))
    201:     }
    202:   }
```

L	routes/search.ts **(Line 50)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ **Problematic Code:**

```
     47:           void models.sequelize.query('SELECT sql FROM sqlite_master').then(([data]:
any) => {
     48:             const tableDefinitions = utils.queryResultToJson(data)
     49:             if (tableDefinitions.data?.length) {
>>>  50:               for (let i = 0; i < tableDefinitions.data.length; i++) {
     51:                 if (tableDefinitions.data[i].sql) {
     52:                   solved = solved && utils.containsOrEscaped(dataString,
tableDefinitions.data[i].sql)
     53:                   if (!solved) {
```

L	lib/insecurity.ts **(Line 120)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ Problematic Code:

```
    117: }
    118:
    119: function hasValidFormat (coupon: string) {
>>> 120:   return
coupon.match(/(JAN|FEB|MAR|APR|MAY|JUN|JUL|AUG|SEP|OCT|NOV|DEC)[0-9]{2}-[0-9]{2}/)
    121: }
    122:
    123: // vuln-code-snippet start redirectCryptoCurrencyChallenge redirectChallenge
```

L	lib/insecurity.ts **(Line 120)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ Problematic Code:

```
    117: }
    118:
    119: function hasValidFormat (coupon: string) {
>>> 120:   return
coupon.match(/(JAN|FEB|MAR|APR|MAY|JUN|JUL|AUG|SEP|OCT|NOV|DEC)[0-9]{2}-[0-9]{2}/)
    121: }
    122:
    123: // vuln-code-snippet start redirectCryptoCurrencyChallenge redirectChallenge
```

L	lib/insecurity.ts **(Line 120)**	**typescript:S6594** Use the "RegExp.exec()" method instead.

■ Problematic Code:

```
    117: }
    118:
    119: function hasValidFormat (coupon: string) {
>>> 120:   return
coupon.match(/(JAN|FEB|MAR|APR|MAY|JUN|JUL|AUG|SEP|OCT|NOV|DEC)[0-9]{2}-[0-9]{2}/)
    121: }
    122:
    123: // vuln-code-snippet start redirectCryptoCurrencyChallenge redirectChallenge
```

L	routes/basket.ts **(Line 26)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ Problematic Code:

```
     23:           return user && id && id !== 'undefined' && id !== 'null' && id !== 'NaN' &&
user.bid && user?.bid != parseInt(id, 10) // eslint-disable-line eqeqeq
     24:         })
     25:         if (((basket?.Products) != null) && basket.Products.length > 0) {
>>>  26:           for (let i = 0; i < basket.Products.length; i++) {
     27:             basket.Products[i].name = req.__(basket.Products[i].name)
     28:           }
     29:         }
```

L	routes/payment.ts **(Line 9)**	**typescript:S101** Rename interface "displayCard" to match the regular expression ^[A-Z][a-zA-Z0-9]*$.

■ **Problematic Code:**

```
      6: import { type Request, type Response, type NextFunction } from 'express'
      7: import { CardModel } from '../models/card'
      8:
>>>   9: interface displayCard {
     10:   UserId: number
     11:   id: number
     12:   fullName: string
```

L	server.ts **(Line 218)**	**typescript:S1874** The signature '(from: number, length?: number | undefined): string' of 'locale.key.substr' is deprecated.

■ **Problematic Code:**

```
    215:     contact: config.get('application.securityTxt.contact'),
    216:     encryption: config.get('application.securityTxt.encryption'),
    217:     acknowledgements: config.get('application.securityTxt.acknowledgements'),
>>> 218:     'Preferred-Languages': [...new Set(locales.map((locale: { key: string }) =>
locale.key.substr(0, 2)))].join(', '),
    219:     hiring: config.get('application.securityTxt.hiring'),
    220:     csaf: config.get<string>('server.baseUrl') +
config.get<string>('application.securityTxt.csaf'),
    221:     expires: securityTxtExpiration.toUTCString()
```

L	frontend/src/app/search-result search-result.component.ts **(Line 134)**	**typescript:S4138** Expected a \`for-of\` loop instead of a \`for\` loop with this simple iteration.

■ **Problematic Code:**

```
    131:   }
    132:
    133:   trustProductDescription (tableData: any[]) { // vuln-code-snippet neutral-line
restfulXssChallenge
>>> 134:     for (let i = 0; i < tableData.length; i++) { // vuln-code-snippet neutral-line
restfulXssChallenge
    135:       tableData[i].description =
this.sanitizer.bypassSecurityTrustHtml(tableData[i].description) // vuln-code-snippet vuln-line
restfulXssChallenge
    136:     } // vuln-code-snippet neutral-line restfulXssChallenge
    137:   } // vuln-code-snippet neutral-line restfulXssChallenge
```

L	data/static/codefixes dbSchemaChallenge_1.ts **(Line 8)**	**typescript:S4138** Expected a \`for-of\` loop instead of a \`for\` loop with this simple iteration.

■ **Problematic Code:**

```
      5:     models.sequelize.query("SELECT * FROM Products WHERE ((name LIKE '%"+criteria+"%'
OR description LIKE '%"+criteria+"%') AND deletedAt IS NULL) ORDER BY name")
      6:       .then(([products]: any) => {
      7:         const dataString = JSON.stringify(products)
>>>   8:         for (let i = 0; i < products.length; i++) {
      9:           products[i].name = req.__(products[i].name)
     10:           products[i].description = req.__(products[i].description)
     11:         }
```

L	data/static/codefixes dbSchemaChallenge_2_correct.ts **(Line 10)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ **Problematic Code:**

```
      7:         { replacements: { criteria } }
      8:       ).then(([products]: any) => {
      9:         const dataString = JSON.stringify(products)
>>>  10:         for (let i = 0; i < products.length; i++) {
     11:           products[i].name = req.__(products[i].name)
     12:           products[i].description = req.__(products[i].description)
     13:         }
```

L	data/static/codefixes dbSchemaChallenge_3.ts **(Line 14)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ **Problematic Code:**

```
     11:     models.sequelize.query(`SELECT * FROM Products WHERE ((name LIKE '%${criteria}%' OR
description LIKE '%${criteria}%') AND deletedAt IS NULL) ORDER BY name`)
     12:       .then(([products]: any) => {
     13:         const dataString = JSON.stringify(products)
>>>  14:         for (let i = 0; i < products.length; i++) {
     15:           products[i].name = req.__(products[i].name)
     16:           products[i].description = req.__(products[i].description)
     17:         }
```

L	data/static/codefixes unionSqlInjectionChallenge_1.ts **(Line 9)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ **Problematic Code:**

```
      6:     models.sequelize.query(`SELECT * FROM Products WHERE ((name LIKE '%${criteria}%' OR
description LIKE '%${criteria}%') AND deletedAt IS NULL) ORDER BY name`)
      7:       .then(([products]: any) => {
      8:         const dataString = JSON.stringify(products)
>>>   9:         for (let i = 0; i < products.length; i++) {
     10:           products[i].name = req.__(products[i].name)
     11:           products[i].description = req.__(products[i].description)
     12:         }
```

L	data/static/codefixes unionSqlInjectionChallenge_2_correct.ts **(Line 10)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ **Problematic Code:**

```
      7:         { replacements: { criteria } }
      8:       ).then(([products]: any) => {
      9:         const dataString = JSON.stringify(products)
>>>  10:         for (let i = 0; i < products.length; i++) {
     11:           products[i].name = req.__(products[i].name)
     12:           products[i].description = req.__(products[i].description)
     13:         }
```

L	data/static/codefixes unionSqlInjectionChallenge_3.ts **(Line 13)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ **Problematic Code:**

```
     10:   models.sequelize.query(`SELECT * FROM Products WHERE ((name LIKE '%${criteria}%' OR description LIKE '%${criteria}%') AND deletedAt IS NULL) ORDER BY name`)
     11:    .then(([products]: any) => {
     12:     const dataString = JSON.stringify(products)
>>>  13:     for (let i = 0; i < products.length; i++) {
     14:      products[i].name = req.__(products[i].name)
     15:      products[i].description = req.__(products[i].description)
     16:     }
```

L	routes/vulnCodeFixes.ts **(Line 11)**	**typescript:S101** Rename interface "codeFix" to match the regular expression ^[A-Z][a-zA-Z0-9]*$.

■ **Problematic Code:**

```
      8:
      9: const FixesDir = 'data/static/codefixes'
     10:
>>>  11: interface codeFix {
     12:   fixes: string[]
     13:   correct: number
     14: }
```

L	frontend/src/app/Services code-fixes.service.ts **(Line 7)**	**typescript:S101** Rename interface "result" to match the regular expression ^[A-Z][a-zA-Z0-9]*$.

■ **Problematic Code:**

```
      4: import { catchError, map } from 'rxjs/operators'
      5: import { type Observable } from 'rxjs'
      6:
>>>   7: export interface result {
      8:   verdict: boolean
      9: }
     10:
```

L	frontend/src/app/Services vuln-lines.service.ts **(Line 6)**	**typescript:S101** Rename interface "result" to match the regular expression ^[A-Z][a-zA-Z0-9]*$.

■ **Problematic Code:**

```
      3: import { HttpClient } from '@angular/common/http'
      4: import { catchError, map } from 'rxjs/operators'
      5:
>>>   6: export interface result {
      7:   verdict: boolean
      8:   hint: string
      9: }
```

L	routes/vulnCodeSnippet.ts **(Line 26)**	**typescript:S1301** Replace this "switch" statement by "if" statements to increase readability.

■ Problematic Code:

```
     23: }
     24:
     25: const setStatusCode = (error: any) => {
>>>  26:   switch (error.name) {
     27:     case 'BrokenBoundary':
     28:       return 422
     29:     default:
```

L	routes/verify.ts **(Line 133)**	**typescript:S6594** Use the "RegExp.exec()" method instead.

■ Problematic Code:

```
    130: }
    131:
    132: function hasEmail (token: { data: { email: string } }, email: string | RegExp) {
>>> 133:   return token?.data?.email?.match(email)
    134: }
    135:
    136: export const databaseRelatedChallenges = () => (req: Request, res: Response, next: NextFunction) => {
```

L	lib/startup/validatePreconditions.ts **(Line 118)**	**typescript:S1874** The signature '(from: number, length?: number | undefined): string' of 'pathRelativeToProjectRoot.substr' is deprecated.

■ Problematic Code:

```
    115: }
    116:
    117: export const checkIfRequiredFileExists = async (pathRelativeToProjectRoot: string) => {
>>> 118:   const fileName = pathRelativeToProjectRoot.substr(pathRelativeToProjectRoot.lastIndexOf('/') + 1)
    119:
    120:   return await access(path.resolve(pathRelativeToProjectRoot)).then(() => {
    121:     logger.info(`Required file ${colors.bold(fileName)} is present (${colors.green('OK')})`)
```

L	frontend/src/app/data-export data-export.component.ts **(Line 69)**	**typescript:S1874** '(...text: string[]): void' is deprecated.

■ Problematic Code:

```
     66:         this.error = null
     67:         this.confirmation = data.confirmation
     68:         this.userData = data.userData
>>>  69:         window.open('', '_blank', 'width=500')?.document.write(this.userData)
     70:         this.lastSuccessfulTry = new Date()
     71:         localStorage.setItem('lstdtxprt', JSON.stringify(this.lastSuccessfulTry))
     72:         this.ngOnInit()
```

L	routes/basketItems.ts **(Line 26)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ Problematic Code:

```
     23:     const basketIds = []
     24:     const quantities = []
     25:
>>>  26:     for (let i = 0; i < result.length; i++) {
     27:       if (result[i].key === 'ProductId') {
     28:         productIds.push(result[i].value)
     29:       } else if (result[i].key === 'BasketId') {
```

L	routes/showProductReviews.ts **(Line 40)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ Problematic Code:

```
     37:       const t1 = new Date().getTime()
     38:       challengeUtils.solveIf(challenges.noSqlCommandChallenge, () => { return (t1 - t0)
> 2000 })
     39:       const user = security.authenticatedUsers.from(req)
>>>  40:       for (let i = 0; i < reviews.length; i++) {
     41:         if (user === undefined || reviews[i].likedBy.includes(user.data.email)) {
     42:           reviews[i].liked = true
     43:         }
```

L	frontend/src/app/oauth oauth.component.ts **(Line 69)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ Problematic Code:

```
     66:     const hash = this.route.snapshot.data.params.substr(1)
     67:     const splitted = hash.split('&')
     68:     const params: any = {}
>>>  69:     for (let i = 0; i < splitted.length; i++) {
     70:       const param: string = splitted[i].split('=')
     71:       const key: string = param[0]
     72:       params[key] = param[1]
```

L	routes/resetPassword.ts **(Line 66)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ Problematic Code:

```
     63:   challengeUtils.solveIf(challenges.geoStalkingMetaChallenge, () => {
     64:     const securityAnswer = ((() => {
     65:       const memories = config.get<MemoryConfig[]>('memories')
>>>  66:       for (let i = 0; i < memories.length; i++) {
     67:         if (memories[i].geoStalkingMetaSecurityAnswer) {
     68:           return memories[i].geoStalkingMetaSecurityAnswer
     69:         }
```

L	routes/resetPassword.ts **(Line 77)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

```
■ Problematic Code:
     74:   challengeUtils.solveIf(challenges.geoStalkingVisualChallenge, () => {
     75:     const securityAnswer = ((() => {
     76:       const memories = config.get<MemoryConfig[]>('memories')
>>>  77:       for (let i = 0; i < memories.length; i++) {
     78:         if (memories[i].geoStalkingVisualSecurityAnswer) {
     79:           return memories[i].geoStalkingVisualSecurityAnswer
     80:         }
```

L	test/api/metricsApiSpec.ts **(Line 18)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

```
■ Problematic Code:
     15:     return frisby.get(API_URL)
     16:       .expect('status', 200)
     17:       .expect('header', 'content-type', /text\/plain/)
>>>  18:       .expect('bodyContains', /^.*_version_info{version="[0-9]+.[0-9]+.[0-9]+(-SNAPSHO
T)?",major="[0-9]+",minor="[0-9]+",patch="[0-9]+",app=".*"} 1$/gm)
     19:       .expect('bodyContains',
/^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     20:       .expect('bodyContains',
/^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:       .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 18)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

```
■ Problematic Code:
     15:     return frisby.get(API_URL)
     16:       .expect('status', 200)
     17:       .expect('header', 'content-type', /text\/plain/)
>>>  18:       .expect('bodyContains', /^.*_version_info{version="[0-9]+.[0-9]+.[0-9]+(-SNAPSHO
T)?",major="[0-9]+",minor="[0-9]+",patch="[0-9]+",app=".*"} 1$/gm)
     19:       .expect('bodyContains',
/^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     20:       .expect('bodyContains',
/^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:       .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 18)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ Problematic Code:

```
     15:     return frisby.get(API_URL)
     16:       .expect('status', 200)
     17:       .expect('header', 'content-type', /text\/plain/)
>>>  18:       .expect('bodyContains', /^.*_version_info{version="[0-9]+.[0-9]+.[0-9]+(-SNAPSHO
T)?",major="[0-9]+",minor="[0-9]+",patch="[0-9]+",app=".*"} 1$/gm)
     19:       .expect('bodyContains',
/^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     20:       .expect('bodyContains',
/^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:       .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 18)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ Problematic Code:

```
     15:     return frisby.get(API_URL)
     16:       .expect('status', 200)
     17:       .expect('header', 'content-type', /text\/plain/)
>>>  18:       .expect('bodyContains', /^.*_version_info{version="[0-9]+.[0-9]+.[0-9]+(-SNAPSHO
T)?",major="[0-9]+",minor="[0-9]+",patch="[0-9]+",app=".*"} 1$/gm)
     19:       .expect('bodyContains',
/^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     20:       .expect('bodyContains',
/^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:       .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 18)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ Problematic Code:

```
     15:     return frisby.get(API_URL)
     16:       .expect('status', 200)
     17:       .expect('header', 'content-type', /text\/plain/)
>>>  18:       .expect('bodyContains', /^.*_version_info{version="[0-9]+.[0-9]+.[0-9]+(-SNAPSHO
T)?",major="[0-9]+",minor="[0-9]+",patch="[0-9]+",app=".*"} 1$/gm)
     19:       .expect('bodyContains',
/^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     20:       .expect('bodyContains',
/^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:       .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 18)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     15:     return frisby.get(API_URL)
     16:       .expect('status', 200)
     17:       .expect('header', 'content-type', /text\/plain/)
>>>  18:       .expect('bodyContains', /^.*_version_info{version="[0-9]+.[0-9]+.[0-9]+(-SNAPSHOT)?",major="[0-9]+",minor="[0-9]+",patch="[0-9]+",app=".*"} 1$/gm)
     19:       .expect('bodyContains', /^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     20:       .expect('bodyContains', /^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:       .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 19)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     16:       .expect('status', 200)
     17:       .expect('header', 'content-type', /text\/plain/)
     18:       .expect('bodyContains', /^.*_version_info{version="[0-9]+.[0-9]+.[0-9]+(-SNAPSHOT)?",major="[0-9]+",minor="[0-9]+",patch="[0-9]+",app=".*"} 1$/gm)
>>>  19:       .expect('bodyContains', /^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     20:       .expect('bodyContains', /^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:       .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
     22:       .expect('bodyContains', /^.*_orders_placed_total{app=".*"} [0-9]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 20)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     17:       .expect('header', 'content-type', /text\/plain/)
     18:       .expect('bodyContains', /^.*_version_info{version="[0-9]+.[0-9]+.[0-9]+(-SNAPSHOT)?",major="[0-9]+",minor="[0-9]+",patch="[0-9]+",app=".*"} 1$/gm)
     19:       .expect('bodyContains', /^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
>>>  20:       .expect('bodyContains', /^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:       .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
     22:       .expect('bodyContains', /^.*_orders_placed_total{app=".*"} [0-9]*$/gm)
     23:       .expect('bodyContains', /^.*_users_registered{type="standard",app=".*"} [0-9]*$/gm)
```

L	test/server challengeTutorialSequenceSpec.ts **(Line 34)**	**typescript:S6653** Use 'Object.hasOwn()' instead of 'Object.prototype.hasOwnProperty.call()'.

■ **Problematic Code:**

```
     31:     for (const { tutorial } of challenges) {
     32:       if (tutorial) {
     33:         const order: string = tutorial.order
>>>  34:         if (!Object.prototype.hasOwnProperty.call(tutorialOrderCounts, order)) {
     35:           tutorialOrderCounts[order] = 0
     36:         }
     37:         tutorialOrderCounts[order]++
```

L	test/api/metricsApiSpec.ts **(Line 61)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     58:         return frisby.get(API_URL)
     59:           .expect('status', 200)
     60:           .expect('header', 'content-type', /text\/plain/)
>>>  61:           .expect('bodyContains', /^file_upload_errors{file_type=".*",app=".*"}
[0-9]*$/gm)
     62:       })
     63:   })
     64: })
```

L	test/api/metricsApiSpec.ts **(Line 45)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     42:         return frisby.get(API_URL)
     43:           .expect('status', 200)
     44:           .expect('header', 'content-type', /text\/plain/)
>>>  45:           .expect('bodyContains', /^file_uploads_count{file_type=".*",app=".*"}
[0-9]*$/gm)
     46:       })
     47:   })
     48:
```

L	test/api/metricsApiSpec.ts **(Line 30)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     27:       .expect('bodyContains', /^.*_user_social_interactions{type="review",app=".*"}
[0-9]*$/gm)
     28:       .expect('bodyContains', /^.*_user_social_interactions{type="feedback",app=".*"}
[0-9]*$/gm)
     29:       .expect('bodyContains', /^.*_user_social_interactions{type="complaint",app=".*"}
[0-9]*$/gm)
>>>  30:       .expect('bodyContains', /^http_requests_count{status_code="[0-9]XX",app=".*"}
[0-9]*$/gm)
     31:   })
     32:
     33:   xit('GET file upload metrics via public API', () => { // FIXME Flaky on CI/CD on at
least Windows
```

L	test/api/metricsApiSpec.ts **(Line 30)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     27:       .expect('bodyContains', /^.*_user_social_interactions{type="review",app=".*"} [0-9]*$/gm)
     28:       .expect('bodyContains', /^.*_user_social_interactions{type="feedback",app=".*"} [0-9]*$/gm)
     29:       .expect('bodyContains', /^.*_user_social_interactions{type="complaint",app=".*"} [0-9]*$/gm)
>>>  30:       .expect('bodyContains', /^http_requests_count{status_code="[0-9]XX",app=".*"} [0-9]*$/gm)
     31:   })
     32:
     33:   xit('GET file upload metrics via public API', () => { // FIXME Flaky on CI/CD on at least Windows
```

L	test/api/metricsApiSpec.ts **(Line 27)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     24:       .expect('bodyContains', /^.*_users_registered{type="deluxe",app=".*"} [0-9]*$/gm)
     25:       .expect('bodyContains', /^.*_users_registered_total{app=".*"} [0-9]*$/gm)
     26:       .expect('bodyContains', /^.*_wallet_balance_total{app=".*"} [0-9]*$/gm)
>>>  27:       .expect('bodyContains', /^.*_user_social_interactions{type="review",app=".*"} [0-9]*$/gm)
     28:       .expect('bodyContains', /^.*_user_social_interactions{type="feedback",app=".*"} [0-9]*$/gm)
     29:       .expect('bodyContains', /^.*_user_social_interactions{type="complaint",app=".*"} [0-9]*$/gm)
     30:       .expect('bodyContains', /^http_requests_count{status_code="[0-9]XX",app=".*"} [0-9]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 28)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     25:       .expect('bodyContains', /^.*_users_registered_total{app=".*"} [0-9]*$/gm)
     26:       .expect('bodyContains', /^.*_wallet_balance_total{app=".*"} [0-9]*$/gm)
     27:       .expect('bodyContains', /^.*_user_social_interactions{type="review",app=".*"} [0-9]*$/gm)
>>>  28:       .expect('bodyContains', /^.*_user_social_interactions{type="feedback",app=".*"} [0-9]*$/gm)
     29:       .expect('bodyContains', /^.*_user_social_interactions{type="complaint",app=".*"} [0-9]*$/gm)
     30:       .expect('bodyContains', /^http_requests_count{status_code="[0-9]XX",app=".*"} [0-9]*$/gm)
     31:   })
```

L	test/api/metricsApiSpec.ts **(Line 29)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ Problematic Code:

```
     26:        .expect('bodyContains', /^.*_wallet_balance_total{app=".*"} [0-9]*$/gm)
     27:        .expect('bodyContains', /^.*_user_social_interactions{type="review",app=".*"}
[0-9]*$/gm)
     28:        .expect('bodyContains', /^.*_user_social_interactions{type="feedback",app=".*"}
[0-9]*$/gm)
>>>  29:        .expect('bodyContains', /^.*_user_social_interactions{type="complaint",app=".*"}
[0-9]*$/gm)
     30:        .expect('bodyContains', /^http_requests_count{status_code="[0-9]XX",app=".*"}
[0-9]*$/gm)
     31:    })
     32:
```

L	test/api/metricsApiSpec.ts **(Line 22)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ Problematic Code:

```
     19:        .expect('bodyContains',
/^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     20:        .expect('bodyContains',
/^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:        .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
>>>  22:        .expect('bodyContains', /^.*_orders_placed_total{app=".*"} [0-9]*$/gm)
     23:        .expect('bodyContains', /^.*_users_registered{type="standard",app=".*"}
[0-9]*$/gm)
     24:        .expect('bodyContains', /^.*_users_registered{type="deluxe",app=".*"}
[0-9]*$/gm)
     25:        .expect('bodyContains', /^.*_users_registered_total{app=".*"} [0-9]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 23)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ Problematic Code:

```
     20:        .expect('bodyContains',
/^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:        .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
     22:        .expect('bodyContains', /^.*_orders_placed_total{app=".*"} [0-9]*$/gm)
>>>  23:        .expect('bodyContains', /^.*_users_registered{type="standard",app=".*"}
[0-9]*$/gm)
     24:        .expect('bodyContains', /^.*_users_registered{type="deluxe",app=".*"}
[0-9]*$/gm)
     25:        .expect('bodyContains', /^.*_users_registered_total{app=".*"} [0-9]*$/gm)
     26:        .expect('bodyContains', /^.*_wallet_balance_total{app=".*"} [0-9]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 24)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     21:        .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
     22:        .expect('bodyContains', /^.*_orders_placed_total{app=".*"} [0-9]*$/gm)
     23:        .expect('bodyContains', /^.*_users_registered{type="standard",app=".*"}
[0-9]*$/gm)
>>>  24:        .expect('bodyContains', /^.*_users_registered{type="deluxe",app=".*"}
[0-9]*$/gm)
     25:        .expect('bodyContains', /^.*_users_registered_total{app=".*"} [0-9]*$/gm)
     26:        .expect('bodyContains', /^.*_wallet_balance_total{app=".*"} [0-9]*$/gm)
     27:        .expect('bodyContains', /^.*_user_social_interactions{type="review",app=".*"}
[0-9]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 25)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     22:        .expect('bodyContains', /^.*_orders_placed_total{app=".*"} [0-9]*$/gm)
     23:        .expect('bodyContains', /^.*_users_registered{type="standard",app=".*"}
[0-9]*$/gm)
     24:        .expect('bodyContains', /^.*_users_registered{type="deluxe",app=".*"}
[0-9]*$/gm)
>>>  25:        .expect('bodyContains', /^.*_users_registered_total{app=".*"} [0-9]*$/gm)
     26:        .expect('bodyContains', /^.*_wallet_balance_total{app=".*"} [0-9]*$/gm)
     27:        .expect('bodyContains', /^.*_user_social_interactions{type="review",app=".*"}
[0-9]*$/gm)
     28:        .expect('bodyContains', /^.*_user_social_interactions{type="feedback",app=".*"}
[0-9]*$/gm)
```

L	test/api/metricsApiSpec.ts **(Line 26)**	**typescript:S6353** Use concise character class syntax '\d' instead of '[0-9]'.

■ **Problematic Code:**

```
     23:        .expect('bodyContains', /^.*_users_registered{type="standard",app=".*"}
[0-9]*$/gm)
     24:        .expect('bodyContains', /^.*_users_registered{type="deluxe",app=".*"}
[0-9]*$/gm)
     25:        .expect('bodyContains', /^.*_users_registered_total{app=".*"} [0-9]*$/gm)
>>>  26:        .expect('bodyContains', /^.*_wallet_balance_total{app=".*"} [0-9]*$/gm)
     27:        .expect('bodyContains', /^.*_user_social_interactions{type="review",app=".*"}
[0-9]*$/gm)
     28:        .expect('bodyContains', /^.*_user_social_interactions{type="feedback",app=".*"}
[0-9]*$/gm)
     29:        .expect('bodyContains', /^.*_user_social_interactions{type="complaint",app=".*"}
[0-9]*$/gm)
```

L	routes/search.ts **(Line 31)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ **Problematic Code:**

```
     28:           UserModel.findAll().then(data => {
     29:             const users = utils.queryResultToJson(data)
     30:             if (users.data?.length) {
>>>  31:               for (let i = 0; i < users.data.length; i++) {
     32:                 solved = solved && utils.containsOrEscaped(dataString,
users.data[i].email) && utils.contains(dataString, users.data[i].password)
     33:                 if (!solved) {
     34:                   break
```

L	server.ts **(Line 516)**	**typescript:S4138** Expected a \`for-of\` loop instead of a \`for\` loop with this simple iteration.

■ **Problematic Code:**

```
    513:     // translate challenge descriptions on-the-fly
    514:     if (name === 'Challenge') {
    515:       resource.list.fetch.after((req: Request, res: Response, context: { instance:
string | any[], continue: any }) => {
>>> 516:         for (let i = 0; i < context.instance.length; i++) {
    517:           let description = context.instance[i].description
    518:           if (utils.contains(description, '<em>(This challenge is <strong>')) {
    519:             const warning = description.substring(description.indexOf(' <em>(This
challenge is <strong>'))
```

L	lib/startup/validateConfig.ts **(Line 62)**	**typescript:S1874** The signature '(from: number, length?: number | undefined): string' of 'message.substr' is deprecated.

■ **Problematic Code:**

```
     59:   if (schemaErrors.length !== 0) {
     60:     logger.warn(`Config schema validation failed with ${schemaErrors.length} errors
(${colors.red('NOT OK')})`)
     61:     schemaErrors.forEach(({ path, message }: { path: string, message: string }) => {
>>>  62:       logger.warn(`${path}:${colors.red(message.substr(message.indexOf(path) +
path.length))}`)
     63:     })
     64:     success = false
     65:   }
```

L	routes/search.ts **(Line 64)**	**typescript:S4138** Expected a \`for-of\` loop instead of a \`for\` loop with this simple iteration.

■ **Problematic Code:**

```
     61:             }
     62:           })
     63:         } // vuln-code-snippet hide-end
>>>  64:         for (let i = 0; i < products.length; i++) {
     65:           products[i].name = req.__(products[i].name)
     66:           products[i].description = req.__(products[i].description)
     67:         }
```

L	server.ts **(Line 537)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ Problematic Code:

```
    534:     // translate security questions on-the-fly
    535:     if (name === 'SecurityQuestion') {
    536:       resource.list.fetch.after((req: Request, res: Response, context: { instance: string | any[], continue: any }) => {
>>> 537:         for (let i = 0; i < context.instance.length; i++) {
    538:           context.instance[i].question = req.__(context.instance[i].question)
    539:         }
    540:         return context.continue
```

L	server.ts **(Line 565)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ Problematic Code:

```
    562:     // translate product names and descriptions on-the-fly
    563:     if (name === 'Product') {
    564:       resource.list.fetch.after((req: Request, res: Response, context: { instance: any[], continue: any }) => {
>>> 565:         for (let i = 0; i < context.instance.length; i++) {
    566:           context.instance[i].name = req.__(context.instance[i].name)
    567:           context.instance[i].description = req.__(context.instance[i].description)
    568:         }
```

L	routes/currentUser.ts **(Line 18)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ Problematic Code:

```
     15:       if (security.verify(req.cookies.token)) {
     16:         user = security.authenticatedUsers.get(req.cookies.token)
     17:       }
>>>  18:     } catch (err) {
     19:       user = undefined
     20:     } finally {
     21:       const response = { user: { id: (user?.data ? user.data.id : undefined), email: (user?.data ? user.data.email : undefined), lastLoginIp: (user?.data ? user.data.lastLoginIp : undefined), profileImage: (user?.data ? user.data.profileImage : undefined) } }
```

L	frontend/src/app/photo-wall mime-type.validator.ts **(Line 23)**	**typescript:S4138** Expected a `for-of` loop instead of a `for` loop with this simple iteration.

■ Problematic Code:

```
     20:         const arr = new Uint8Array(fileReader.result as ArrayBuffer).subarray(0, 4)
     21:         let header = ''
     22:         let isValid = false
>>>  23:         for (let i = 0; i < arr.length; i++) {
     24:           header += arr[i].toString(16)
     25:         }
     26:         switch (header) {
```

L	frontend/src/app/photo-wall photo-wall.component.html **(Line 14)**	**Web:S6851** Remove redundant word "image" from the "alt" attribute of your "img" tag.

■ **Problematic Code:**

```
     11:         <div class="grid">
     12:           @for (image of slideshowDataSource; track image) {
     13:             <span class="container mat-elevation-z6">
>>>  14:               <img src="{{ image.url }}" alt="{{ image.caption }}" class="image">
     15:               <div class="overlay">
     16:                 <div>{{ image.caption }}</div>
     17:                 @if (twitterHandle) {
```

L	routes/languages.ts **(Line 63)**	**typescript:S6653** Use 'Object.hasOwn()' instead of 'Object.prototype.hasOwnProperty.call()'.

■ **Problematic Code:**

```
     60:       return await new Promise((resolve, reject) => {
     61:         try {
     62:           for (const key in fileContent) {
>>>  63:             if (Object.prototype.hasOwnProperty.call(fileContent, key) &&
fileContent[key] !== enContent[key]) {
     64:               differentStrings++
     65:             }
     66:           }
```

L	test/server challengeCountryMappingSpec.ts **(Line 41)**	**typescript:S6653** Use 'Object.hasOwn()' instead of 'Object.prototype.hasOwnProperty.call()'.

■ **Problematic Code:**

```
     38:     for (const key of Object.keys(countryMapping)) {
     39:       const { code } = countryMapping[key]
     40:
>>>  41:       if (!Object.prototype.hasOwnProperty.call(countryCodeCounts, code)) {
     42:         countryCodeCounts[code] = 0
     43:       }
     44:       countryCodeCounts[code]++
```

L	routes/countryMapping.ts **(Line 19)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
     16:       } else {
     17:         res.send(countryMapping)
     18:       }
>>>  19:     } catch (err) {
     20:       logger.warn('Country mapping was requested but was not found in the selected
config file. Take a look at the fbctf.yml config file to find out how to configure the country
mappings required by FBCTF.')
     21:       res.status(500).send()
     22:     }
```

L	routes/2fa.ts **(Line 173)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ Problematic Code:

```
    170:   security.authenticatedUsers.updateFrom(req, utils.queryResultToJson(userModel))
    171:
    172:   res.status(200).send()
>>> 173:  } catch (error) {
    174:   res.status(401).send()
    175:  }
    176: }
```

L	routes/2fa.ts **(Line 89)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ Problematic Code:

```
     86:     setup: true
     87:    })
     88:   }
>>>  89:  } catch (error) {
     90:   res.status(401).send()
     91:  }
     92: }
```

L	routes/2fa.ts **(Line 140)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ Problematic Code:

```
    137:   security.authenticatedUsers.updateFrom(req, utils.queryResultToJson(userModel))
    138:
    139:   res.status(200).send()
>>> 140:  } catch (error) {
    141:   res.status(401).send()
    142:  }
    143: }
```

L	lib/utils.ts **(Line 213)**	**typescript:S1874** The signature '(from: number, length?: number | undefined): string' of 'ipv6.substr' is deprecated.

■ Problematic Code:

```
    210:
    211: export const toSimpleIpAddress = (ipv6: string) => {
    212:  if (startsWith(ipv6, '::ffff:')) {
>>> 213:   return ipv6.substr(7)
    214:  } else if (ipv6 === '::1') {
    215:   return '127.0.0.1'
    216:  } else {
```

L	routes/2fa.ts **(Line 54)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
     51:     security.authenticatedUsers.put(token, plainUser)
     52:
     53:     res.json({ authentication: { token, bid: basket.id, umail: user.email } })
>>>  54:   } catch (error) {
     55:     res.status(401).send()
     56:   }
     57: }
```

L	frontend/src/app/Services form-submit.service.ts **(Line 22)**	**typescript:S1874** 'keyCode' is deprecated.

■ **Problematic Code:**

```
     19:     form.addEventListener('keyup', function (event) {
     20:       event.preventDefault()
     21:       // eslint-disable-next-line import/no-deprecated
>>>  22:       if (event.keyCode === 13 && !submitButton.disabled) {
     23:         onSubmit()
     24:       }
     25:     })
```

L	routes/userProfile.ts **(Line 63)**	**typescript:S2486** Handle this exception or don't catch it at all.

■ **Problematic Code:**

```
     60:           throw new Error('Username is null')
     61:         }
     62:         username = eval(code) // eslint-disable-line no-eval
>>>  63:       } catch (err) {
     64:         username = '\\' + username
     65:       }
     66:     } else {
```

L	frontend/src/app/app.routing.ts **(Line 236)**	**typescript:S1874** '(from: number, length?: number): string' is deprecated.

■ **Problematic Code:**

```
    233:   // vuln-code-snippet start tokenSaleChallenge
    234:   {
    235:     matcher: oauthMatcher,
>>> 236:     data: { params: (window.location.href).substr(window.location.href.indexOf('#')) },
    237:     component: OAuthComponent
    238:   },
    239:   { // vuln-code-snippet neutral-line tokenSaleChallenge
```

L	frontend/src/main.ts **(Line 7)**	**typescript:S1128** Remove this unused import of 'platformBrowserDynamic'.

■ **Problematic Code:**

```
      4:  */
      5:
      6: import { enableProdMode, importProvidersFrom } from '@angular/core'
>>>   7: import { platformBrowserDynamic } from '@angular/platform-browser-dynamic'
      8:
      9: import { AppModule, HttpLoaderFactory } from './app/app.module'
     10: import { environment } from './environments/environment'
```

L	lib/utils.ts **(Line 65)**	**typescript:S1874** The signature '(from: number, length?: number | undefined): string' of 'str.substr' is deprecated.

■ **Problematic Code:**

```
     62:
     63: export const trunc = function (str: string, length: number) {
     64:   str = str.replace(/(\r\n|\n|\r)/gm, '')
>>>  65:   return (str.length > length) ? str.substr(0, length - 1) + '...' : str
     66: }
     67:
     68: export const version = (module?: string) => {
```

I	routes/languages.ts **(Line 10)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ **Problematic Code:**

```
      7: import fs from 'node:fs'
      8: import { type Request, type Response, type NextFunction } from 'express'
      9:
>>>  10: export function getLanguageList () { // TODO Refactor and extend to also load backend
translations from /i18n/*json and calculate joint percentage/gauge
     11:   return (req: Request, res: Response, next: NextFunction) => {
     12:     const languages: Array<{ key: string, lang: any, icons: string[], shortKey: string,
percentage: unknown, gauge: string }> = []
     13:     let count = 0
```

I	routes/fileUpload.ts **(Line 119)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ **Problematic Code:**

```
    116:         const yamlString = vm.runInContext('JSON.stringify(yaml.load(data))', sandbox,
{ timeout: 2000 })
    117:         res.status(410)
    118:         next(new Error('B2B customer complaints via file upload have been deprecated
for security reasons: ' + utils.trunc(yamlString, 400) + ' (' + file.originalname + ')'))
>>> 119:       } catch (err: any) { // TODO: Remove any
    120:         if (utils.contains(err.message, 'Invalid string length') ||
utils.contains(err.message, 'Script execution timed out')) {
    121:           if (challengeUtils.notSolved(challenges.yamlBombChallenge)) {
    122:             challengeUtils.solve(challenges.yamlBombChallenge)
```

I	frontend/src/app/password-strength-info password-strength-info.component.spec.ts **(Line 24)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     21:     expect(component).toBeTruthy()
     22:   })
     23: 
>>>  24:   // todo: unit test each conditional | passwordLength message
     25: })
     26: 
```

I	frontend/src/app/app.module.ts **(Line 16)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     13:   return new TranslateHttpLoader(http, './assets/i18n/', '.json')
     14: }
     15: 
>>>  16: @NgModule(/* TODO(standalone-migration): clean up removed NgModule class manually.
     17: {
     18:     declarations: [AppComponent],
     19:     imports: [
```

I	data/datacreator.ts **(Line 76)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     73: 
     74:   await Promise.all(
     75:     challenges.map(async ({ name, category, description, difficulty, hints,
mitigationUrl, key, disabledEnv, tutorial, tags }) => {
>>>  76:       // todo(@J12934) change this to use a proper challenge model or something
     77:       // eslint-disable-next-line @typescript-eslint/consistent-type-assertions
     78:       const { enabled: isChallengeEnabled, disabledBecause } =
utils.getChallengeEnablementStatus({ disabledEnv: disabledEnv?.join(';') ?? '' } as
ChallengeModel)
     79:       description = description.replace('juice-sh.op',
config.get<string>('application.domain'))
```

I	data/datacreator.ts **(Line 93)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     90:           name,
     91:           category,
     92:           tags: (tags != null) ? tags.join(',') : undefined,
>>>  93:           // todo(@J12934) currently missing the 'not available' text. Needs changes to
the model and utils functions
     94:           description: isChallengeEnabled ? description : (description + ' <em>(This
challenge is <strong>potentially harmful</strong> on ' + disabledBecause + '!)</em>'),
     95:           difficulty,
     96:           solved: false,
```

I	lib/config.types.ts **(Line 2)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
      1: // manually typed definitions for the config file
>>>   2: // todo(@jannik.hollenbach) migrate the config schema to something which can
automatically generate the types
      3:
      4: export interface ServerConfig {
      5:   port: number
```

I	test/cypress/e2e/scoreBoard.spec.ts **(Line 30)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     27:   })
     28: })
     29:
>>>  30: xdescribe('/#/score-board-legacy', () => { // TODO Replace with test based on new Score
Board
     31:   describe('repeat notification', () => {
     32:     beforeEach(() => {
     33:       cy.visit('/#/score-board-legacy')
```

I	lib/startup/validatePreconditions.ts **(Line 94)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     91:       domainDependencies[domain].forEach((dependency: string) => {
     92:         logger.warn(`${colors.italic(dependency)} will not work as intended without
access to ${colors.bold(domain)}`)
     93:       })
>>>  94:       return true // TODO Consider switching to "false" with breaking release v16.0.0
or later
     95:     })
     96: }
     97:
```

I	lib/is-docker.ts **(Line 26)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     23: }
     24:
     25: export default function isDocker () {
>>>  26:   // TODO: Use `??=` when targeting Node.js 16.
     27:   if (isDockerCached === undefined) {
     28:     isDockerCached = hasDockerEnv() || hasDockerCGroup()
     29:   }
```

I	lib/codingChallenges.ts **(Line 63)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

```
■ Problematic Code:
     60:   if (snippets == null) {
     61:     throw new BrokenBoundary('Broken code snippet boundaries for: ' + challengeKey)
     62:   }
>>>  63:   let snippet = snippets[0] // TODO Currently only a single code snippet is supported
     64:   snippet = snippet.replace(/\s?[/#]{0,2} vuln-code-snippet start.*[\r\n]{0,2}/g, '')
     65:   snippet = snippet.replace(/\s?[/#]{0,2} vuln-code-snippet end.*/g, '')
     66:   snippet = snippet.replace(/.*[/#]{0,2} vuln-code-snippet hide-line[\r\n]{0,2}/g, '')
```

I	test/cypress/e2e/restApi.spec.ts **(Line 105)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

```
■ Problematic Code:
    102:               console.log('Success')
    103:             }
    104:           })
>>> 105:           cy.expectChallengeSolved({ challenge: 'HTTP-Header XSS' }) // TODO Add
missing check for alert presence
    106:         }
    107:       })
    108:     })
```

I	test/cypress/e2e/basket.spec.ts **(Line 51)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

```
■ Problematic Code:
     48:
     49:         cy.visit('/#/basket')
     50:
>>>  51:         // TODO Verify functionally that it's not the basket of the admin
     52:         cy.expectChallengeSolved({ challenge: 'View Basket' })
     53:       })
     54:     })
```

I	test/api/quantityApiSpec.ts **(Line 185)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

```
■ Problematic Code:
    182:       })
    183:   })
    184:
>>> 185:   xit('GET quantity of all items for accounting users from IP 123.456.789', () => { //
TODO Check if possible to set IP in frisby tests
    186:     return frisby.post(`${REST_URL}/user/login`, {
    187:       headers: jsonHeader,
    188:       body: {
```

I	test/api/quantityApiSpec.ts **(Line 264)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ **Problematic Code:**

```
    261:       })
    262:   })
    263:
>>> 264:   xit('PUT quantity as accounting user from IP 123.456.789', () => { // TODO Check if
possible to set IP in frisby tests
    265:     return frisby.post(`${REST_URL}/user/login`, {
    266:       headers: jsonHeader,
    267:       body: {
```

I	data/datacreator.ts **(Line 391)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ **Problematic Code:**

```
    388:                   persistedProduct)
    389:               })
    390:             }
>>> 391:             if (deletedDate) void deleteProduct(persistedProduct.id) // TODO Rename
into "isDeleted" or "deletedFlag" in config for v14.x release
    392:           } else {
    393:             throw new Error('No persisted product found!')
    394:           }
```

I	routes/fileUpload.ts **(Line 88)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ **Problematic Code:**

```
     85:         challengeUtils.solveIf(challenges.xxeFileDisclosureChallenge, () => { return
(utils.matchesEtcPasswdFile(xmlString) || utils.matchesSystemIniFile(xmlString)) })
     86:         res.status(410)
     87:         next(new Error('B2B customer complaints via file upload have been deprecated
for security reasons: ' + utils.trunc(xmlString, 400) + ' (' + file.originalname + ')'))
>>>  88:       } catch (err: any) { // TODO: Remove any
     89:         if (utils.contains(err.message, 'Script execution timed out')) {
     90:           if (challengeUtils.notSolved(challenges.xxeDosChallenge)) {
     91:             challengeUtils.solve(challenges.xxeDosChallenge)
```

I	lib/utils.ts **(Line 45)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ **Problematic Code:**

```
     42:
     43: export const endsWith = (str?: string, suffix?: string) => (str && suffix) ?
str.includes(suffix, str.length - suffix.length) : false
     44:
>>>  45: export const contains = (str: string, element: string) => str ? str.includes(element) :
false // TODO Inline all usages as this function is not adding any functionality to
String.includes
     46:
     47: export const containsEscaped = function (str: string, element: string) {
     48:   return contains(str, element.replace(/"/g, '\\"'))
```

I	data/static/codefixes adminSectionChallenge_1_correct.ts **(Line 2)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

```
■ Problematic Code:
         1: const routes: Routes = [
>>>      2:   /* TODO: Externalize admin functions into separate application
         3:            that is only accessible inside corporate network.
         4:    */
         5:   // {
```

I	lib/antiCheat.ts **(Line 19)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

```
■ Problematic Code:
        16: import median from 'median'
        17: import { type ChallengeKey } from 'models/challenge'
        18:
>>>     19: const coupledChallenges = { // TODO prevent also near-identical challenges (e.g. all
null byte file access or dom xss + bonus payload etc.) from counting as cheating
        20:   loginAdminChallenge: ['weakPasswordChallenge'],
        21:   nullByteChallenge: ['easterEggLevelOneChallenge', 'forgottenDevBackupChallenge',
'forgottenBackupChallenge', 'misplacedSignatureFileChallenge'],
        22:   deprecatedInterfaceChallenge: ['uploadTypeChallenge', 'xxeFileDisclosureChallenge',
'xxeDosChallenge', 'yamlBombChallenge'],
```

I	frontend/src/hacking-instructor challenges/bonusPayload.ts **(Line 21)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

```
■ Problematic Code:
        18:         'Assuming you did the **DOM XSS** tutorial already, this one just uses a
funnier payload on the _Search_ field.',
        19:       fixture: '#product-search-fixture',
        20:       unskippable: true,
>>>     21:       resolved: waitInMs(10000) // TODO Add check if "DOM XSS" is solved and if not
recommend doing that first
        22:     },
        23:     {
        24:       text: 'Enter or paste this payload into the _Search_ field: <code>&lt;iframe
width=&quot;100%&quot; height=&quot;166&quot; scrolling=&quot;no&quot;
frameborder=&quot;no&quot; allow=&quot;autoplay&quot; src=&quot;https://w.soundcloud.com/player
/?url=https%3A//api.soundcloud.com/tracks/771984076&amp;color=%23ff5500&amp;auto&lowbar;play=tr
ue&amp;hide&lowbar;related=false&amp;show&lowbar;comments=true&amp;show&lowbar;user=true&amp;sh
ow&lowbar;reposts=false&amp;show&lowbar;teaser=true&quot;&gt;&lt;/iframe&gt;</code>.',
```

I	frontend/src/hacking-instructor challenges/passwordStrength.ts **(Line 43)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     40:        text: "Enter the admin's email address into the **email field**.",
     41:        fixture: '#email',
     42:        unskippable: true,
>>>  43:        resolved: waitForInputToHaveValue('#email', 'admin@juice-sh.op') // TODO Use domain from config instead
     44:      },
     45:      {
     46:        text: 'Now for the password. Lucky for us, the admin chose a really, really, **really** stupid one. Just try any that comes to your mind!',
```

I	frontend/src/hacking-instructor challenges/loginBender.ts **(Line 22)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     19:          "To start this challenge, you'll have to log out first.",
     20:        fixture: '#navbarAccount',
     21:        unskippable: true,
>>>  22:        resolved: waitForLogOut() // TODO Add check if "Login Admin" is solved and if not recommend doing that first
     23:      },
     24:      {
     25:        text:
```

I	frontend/src/hacking-instructor challenges/loginJim.ts **(Line 22)**	**typescript:S1135** Complete the task associated to this "TODO" comment.

■ Problematic Code:

```
     19:          "To start this challenge, you'll have to log out first.",
     20:        fixture: '#navbarAccount',
     21:        unskippable: true,
>>>  22:        resolved: waitForLogOut() // TODO Add check if "Login Admin" is solved and if not recommend doing that first
     23:      },
     24:      {
     25:        text:
```

Reflect Sonar

Security Hotspots

Total: 263 hotspots (High: 179, Medium: 26, Low: 58)

Authentication (170 hotspots)

Risk	File Path	Rule & Message
H	test/api/userApiSpec.ts **(Line 302)**	**typescript:S6418** "Authorization" detected here, make sure this is not a hard-coded secret. Category: Authentication

■ **Security Hotspot Code:**

```
    299:   })
    300:
    301:   it('GET who-am-i request returns nothing on expired auth token', () => {
>>> 302:     return frisby.get(`${REST_URL}/user/whoami`, { headers: { Authorization: 'eyJhbGci
OiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLC
JlbWFpbCI6ImFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiOiIwMTkyMDIzYTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwM
CIsInJvbGUiOiJhZG1pbiIsImxhc3RMb2dpbklwIjoiMC4wLjAuMCIsInByb2ZpbGVJbWFnZSI6ImRlZmF1bHQuc3ZnIiwi
dG90cFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDE5LTA4LTE5IDE1OjU2OjE1LjYyOSArMDA
6MDAiLCJ1cGRhdGVkQXQiOiIyMDE5LTA4LTE5IDE1OjU2OjE1LjYyOSArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYX
QiOjE1NjYyMzAyMjQsImV4cCI6MTU2NjI0ODIyNH0.FL0kkcInY5sDMGKeLHfEOYDTQd3BjR6_mK7Tcm_RH6iCLotTSRRoR
xHpLkbtIQKqBFIt14J4BpLapkzG7ppRWcEley5nego-4iFOmXQvCBz5ISS3HdtM0saJnOe0agyVUen3huFp4F2UCth_y2Sc
jMn_4AgW66cz8NSFPRVpC8g' } })
    303:       .expect('status', 200)
    304:       .expect('header', 'content-type', /application\/json/)
    305:       .expect('json', {
```

Risk	File Path	Rule & Message
H	test/server/verifySpec.ts **(Line 263)**	**typescript:S6418** "authorization" detected here, make sure this is not a hard-coded secret. Category: Authentication

■ **Security Hotspot Code:**

```
    260:       Header: { "alg": "none", "typ": "JWT" }
    261:       Payload: { "data": { "email": "jwtn3d@juice-sh.op" }, "iat": 1508639612, "exp":
9999999999 }
    262:        */
>>> 263:       req.headers = { authorization: 'Bearer eyJhbGciOiJub25lIiwidHlwIjoiSldUIn0.eyJkY
XRhIjp7ImVtYWlsIjoiand0bjNkQGp1aWNlLXNoLm9wIn0sImlhdCI6MTUwODYzOTYxMiwiZXhwIjo5OTk5OTk5OTk5fQ.'
}
    264:
    265:       verify.jwtChallenges()(req, res, next)
    266:
```

Risk	File Path	Rule & Message
H	test/server/verifySpec.ts **(Line 275)**	**typescript:S6418** "authorization" detected here, make sure this is not a hard-coded secret. Category: Authentication

■ **Security Hotspot Code:**

```
    272:        Header: { "alg": "none", "typ": "JWT" }
    273:        Payload: { "data": { "email": "jwtn3d@" }, "iat": 1508639612, "exp": 9999999999 }
    274:         */
>>> 275:        req.headers = { authorization: 'Bearer eyJhbGciOiJub25lIiwidHlwIjoiSldUIn0.eyJkY
XRhIjp7ImVtYWlsIjoiand0bjNkQCJ9LCJpYXQiOjE1MDg2Mzk2MTIsImV4cCI6OTk5OTk5OTk5OX0.' }
    276:
    277:        verify.jwtChallenges()(req, res, next)
    278:
```

H	test/server/verifySpec.ts **(Line 297)**	**typescript:S6418** "authorization" detected here, make sure this is not a hard-coded secret. Category: Authentication

■ **Security Hotspot Code:**

```
    294:          Header: { "alg": "HS256", "typ": "JWT" }
    295:          Payload: { "data": { "email": "rsa_lord@juice-sh.op" }, "iat": 1508639612,
"exp": 9999999999 }
    296:           */
>>> 297:          req.headers = { authorization: 'Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.ey
JkYXRhIjp7ImVtYWlsIjoicnNhX2xvcmRAanVpY2Utc2gub3AifSwiaWF0IjoxNTgyMjIxNTc1fQ.ycFwtqh4ht4Pq9K5rh
iPPY256F9YCTIecd4FHFuSEAg' }
    298:
    299:          verify.jwtChallenges()(req, res, next)
    300:
```

H	test/server/verifySpec.ts **(Line 309)**	**typescript:S6418** "authorization" detected here, make sure this is not a hard-coded secret. Category: Authentication

■ **Security Hotspot Code:**

```
    306:          Header: { "alg": "HS256", "typ": "JWT" }
    307:          Payload: { "data": { "email": "rsa_lord@" }, "iat": 1508639612, "exp":
9999999999 }
    308:           */
>>> 309:          req.headers = { authorization: 'Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.ey
JkYXRhIjp7ImVtYWlsIjoicnNhX2xvcmRAIn0sImlhdCI6MTU4MjIyMTY3NX0.50f6VAIQk2Uzpf3sgH-1JVrrTuwudonm2
DKn2ec7Tg8' }
    310:
    311:          verify.jwtChallenges()(req, res, next)
    312:
```

H	frontend/src/app/Services two-factor-auth-service.spec.ts **(Line 65)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     62:        tick()
     63:
     64:        expect(req.request.method).toBe('POST')
>>>  65:        expect(req.request.body).toEqual({ password: 's3cr3t!', initialToken:
'initialToken', setupToken: 'setupToken' })
     66:        expect(res).toBe(undefined)
     67:        httpMock.verify()
     68:      })
```

H	frontend/src/app/Services two-factor-auth-service.spec.ts **(Line 81)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     78:        tick()
     79:
     80:        expect(req.request.method).toBe('POST')
>>>  81:        expect(req.request.body).toEqual({ password: 's3cr3t!' })
     82:        expect(res).toBe(undefined)
     83:        httpMock.verify()
     84:      })
```

H	frontend/src/app/oauth oauth.component.spec.ts **(Line 84)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     81:    it('will create regular user account with base64 encoded reversed email as password',
fakeAsync(() => {
     82:      userService.oauthLogin.and.returnValue(of({ email: 'test@test.com' }))
     83:      component.ngOnInit()
>>>  84:      expect(userService.save).toHaveBeenCalledWith({ email: 'test@test.com', password:
'bW9jLnRzZXRAdHNldA==', passwordRepeat: 'bW9jLnRzZXRAdHNldA==' })
     85:    }))
     86:
     87:    it('logs in user even after failed account creation as account might already have
existed from previous OAuth login', fakeAsync(() => {
```

H	frontend/src/app/oauth oauth.component.spec.ts **(Line 84)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     81:    it('will create regular user account with base64 encoded reversed email as password',
fakeAsync(() => {
     82:      userService.oauthLogin.and.returnValue(of({ email: 'test@test.com' }))
     83:      component.ngOnInit()
>>>  84:      expect(userService.save).toHaveBeenCalledWith({ email: 'test@test.com', password:
'bW9jLnRzZXRAdHNldA==', passwordRepeat: 'bW9jLnRzZXRAdHNldA==' })
     85:    }))
     86:
     87:    it('logs in user even after failed account creation as account might already have
existed from previous OAuth login', fakeAsync(() => {
```

H	frontend/src/app/oauth oauth.component.spec.ts **(Line 91)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     88:     userService.oauthLogin.and.returnValue(of({ email: 'test@test.com' }))
     89:     userService.save.and.returnValue(throwError({ error: 'Account already exists' }))
     90:     component.ngOnInit()
>>>  91:     expect(userService.login).toHaveBeenCalledWith({ email: 'test@test.com', password: 'bW9jLnRzZXRAdHNldA==', oauth: true })
     92:   }))
     93:
     94:   it('removes authentication token and basket id on failed subsequent regular login attempt', fakeAsync(() => {
```

H	frontend/src/app/register register.component.spec.ts **(Line 115)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    112:   it('password should not be more than 20 characters', () => {
    113:     let password: string = ''
    114:     for (let i = 0; i < 41; i++) {
>>> 115:       password += 'a'
    116:     }
    117:     component.passwordControl.setValue(password)
    118:     expect(component.passwordControl.valid).toBeFalsy()
```

H	frontend/src/app/register register.component.spec.ts **(Line 133)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    130:   })
    131:
    132:   it('password and repeat password should be the same', () => {
>>> 133:     const password: string = 'aaaaa'
    134:     const passwordRepeat: string = 'aaaaa'
    135:     component.passwordControl.setValue(password)
    136:     component.repeatPasswordControl.setValue('bbbbb')
```

H	frontend/src/app/register register.component.spec.ts **(Line 134)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    131:
    132:   it('password and repeat password should be the same', () => {
    133:     const password: string = 'aaaaa'
>>> 134:     const passwordRepeat: string = 'aaaaa'
    135:     component.passwordControl.setValue(password)
    136:     component.repeatPasswordControl.setValue('bbbbb')
    137:     expect(component.repeatPasswordControl.valid).toBeFalsy()
```

H	frontend/src/app/register register.component.spec.ts **(Line 151)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    148:     component.repeatPasswordControl.setValue('password')
    149:     component.securityQuestionControl.setValue(1)
    150:     component.securityAnswerControl.setValue('Answer')
>>> 151:     const user = { email: 'x@x.xx', password: 'password', passwordRepeat: 'password',
securityQuestion: { id: 1, question: 'Wat is?' }, securityAnswer: 'Answer' }
    152:     const securityAnswerObject = { UserId: 1, answer: 'Answer', SecurityQuestionId: 1 }
    153:     component.save()
    154:     tick()
```

H	frontend/src/app/register register.component.spec.ts **(Line 151)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    148:     component.repeatPasswordControl.setValue('password')
    149:     component.securityQuestionControl.setValue(1)
    150:     component.securityAnswerControl.setValue('Answer')
>>> 151:     const user = { email: 'x@x.xx', password: 'password', passwordRepeat: 'password',
securityQuestion: { id: 1, question: 'Wat is?' }, securityAnswer: 'Answer' }
    152:     const securityAnswerObject = { UserId: 1, answer: 'Answer', SecurityQuestionId: 1 }
    153:     component.save()
    154:     tick()
```

H	test/api/2faSpec.ts **(Line 174)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    171:   it('GET should indicate 2fa is setup for 2fa enabled users', async () => {
    172:     const { token } = await login({
    173:       email: `wurstbrot@${config.get<string>('application.domain')}`,
>>> 174:       password: 'EinBelegtesBrotMitSchinkenSCHINKEN!',
    175:       totpSecret: 'IFTXE3SPOEYVURT2MRYGI52TKJ4HC3KH'
    176:     })
    177:
```

H	test/api/2faSpec.ts **(Line 200)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    197:   it('GET should indicate 2fa is not setup for users with 2fa disabled', async () => {
    198:     const { token } = await login({
    199:       email: `J12934@${config.get<string>('application.domain')}`,
>>> 200:       password: '0Y8rMnww$*9VFYE§59-!Fg1L6t&6lB'
    201:     })
    202:
    203:     // @ts-expect-error FIXME promise return handling broken
```

H	test/api/2faSpec.ts **(Line 236)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    233: describe('/rest/2fa/setup', () => {
    234:   it('POST should be able to setup 2fa for accounts without 2fa enabled', async () => {
    235:     const email = 'fooooo1@bar.com'
>>> 236:     const password = '123456'
    237:
    238:     const secret = 'ASDVAJSDUASZGDIADBJS'
    239:
```

H	test/api/2faSpec.ts **(Line 282)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    279:
    280:   it('POST should fail if the password doesnt match', async () => {
    281:     const email = 'fooooo2@bar.com'
>>> 282:     const password = '123456'
    283:
    284:     const secret = 'ASDVAJSDUASZGDIADBJS'
    285:
```

H	test/api/2faSpec.ts **(Line 311)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    308:
    309:   it('POST should fail if the initial token is incorrect', async () => {
    310:     const email = 'fooooo3@bar.com'
>>> 311:     const password = '123456'
    312:
    313:     const secret = 'ASDVAJSDUASZGDIADBJS'
    314:
```

H	test/api/2faSpec.ts **(Line 340)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    337:
    338:   it('POST should fail if the token is of the wrong type', async () => {
    339:     const email = 'fooooo4@bar.com'
>>> 340:     const password = '123456'
    341:
    342:     const secret = 'ASDVAJSDUASZGDIADBJS'
    343:
```

H	test/api/2faSpec.ts **(Line 369)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    366:
    367:   it('POST should fail if the account has already set up 2fa', async () => {
    368:     const email = `wurstbrot@${config.get<string>('application.domain')}`
>>> 369:     const password = 'EinBelegtesBrotMitSchinkenSCHINKEN!'
    370:     const totpSecret = 'IFTXE3SPOEYVURT2MRYGI52TKJ4HC3KH'
    371:
    372:     const { token } = await login({ email, password, totpSecret })
```

H	test/api/2faSpec.ts **(Line 398)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    395: describe('/rest/2fa/disable', () => {
    396:   it('POST should be able to disable 2fa for account with 2fa enabled', async () => {
    397:     const email = 'fooooodisable1@bar.com'
>>> 398:     const password = '123456'
    399:     const totpSecret = 'ASDVAJSDUASZGDIADBJS'
    400:
    401:     await register({ email, password, totpSecret })
```

H	test/api/2faSpec.ts **(Line 435)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    432:
    433:   it('POST should not be possible to disable 2fa without the correct password', async
() => {
    434:     const email = 'fooooodisable1@bar.com'
>>> 435:     const password = '123456'
    436:     const totpSecret = 'ASDVAJSDUASZGDIADBJS'
    437:
    438:     await register({ email, password, totpSecret })
```

H	test/api/addressApiSpec.ts **(Line 20)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     17:     headers: jsonHeader,
     18:     body: {
     19:       email: 'jim@juice-sh.op',
>>>  20:       password: 'ncc-1701'
     21:     }
     22:   })
     23:     .expect('status', 200)
```

H	test/api/authenticatedUsersSpec.ts **(Line 21)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     18:     return frisby.get(`${REST_URL}/user/authentication-details`, { headers: authHeader })
     19:       .expect('status', 200)
     20:       .expect('json', 'data.?', {
>>>  21:         password: '********************************'
     22:       })
     23:   })
     24:
```

H	test/api/authenticatedUsersSpec.ts **(Line 30)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     27:       headers: jsonHeader,
     28:       body: {
     29:         email: `jim@${config.get<string>('application.domain')}`,
>>>  30:         password: 'ncc-1701'
     31:       }
     32:     }).promise()
     33:
```

H	test/api/basketApiSpec.ts **(Line 25)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     22:     headers: jsonHeader,
     23:     body: {
     24:       email: 'jim@juice-sh.op',
>>>  25:       password: 'ncc-1701'
     26:     }
     27:   })
     28:     .expect('status', 200)
```

H	test/api/basketApiSpec.ts **(Line 101)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     98:       headers: jsonHeader,
     99:       body: {
    100:         email: 'bjoern.kimminich@gmail.com',
>>> 101:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    102:       }
    103:     })
    104:       .expect('status', 200)
```

H	test/api/basketItemApiSpec.ts **(Line 21)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     18:     headers: jsonHeader,
     19:     body: {
     20:       email: 'jim@' + config.get<string>('application.domain'),
>>>  21:       password: 'ncc-1701'
     22:     }
     23:   })
     24:     .expect('status', 200)
```

H	test/api/chatBotSpec.ts **(Line 56)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     53:     it('GET bot state for authenticated users contains request for username', async ()
=> {
     54:       const { token } = await login({
     55:         email: `J12934@${config.get<string>('application.domain')}`,
>>>  56:         password: '0Y8rMnww$*9VFYE§59-!Fg1L6t&6lB'
     57:       })
     58:
     59:       await frisby.setup({
```

H	test/api/chatBotSpec.ts **(Line 77)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     74:     it('Asks for username if not defined', async () => {
     75:       const { token } = await login({
     76:         email: `J12934@${config.get<string>('application.domain')}`,
>>>  77:         password: '0Y8rMnww$*9VFYE§59-!Fg1L6t&6lB'
     78:       })
     79:
     80:       const testCommand = trainingData.data[0].utterances[0]
```

H	test/api/chatBotSpec.ts **(Line 108)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    105:       }
    106:       const { token } = await login({
    107:         email: 'bjoern.kimminich@gmail.com',
>>> 108:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    109:       })
    110:
    111:       bot.addUser('1337', 'bkimminich')
```

H	test/api/chatBotSpec.ts **(Line 140)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    137:       }
    138:       const { token } = await login({
    139:         email: 'bjoern.kimminich@gmail.com',
>>> 140:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    141:       })
    142:       bot.addUser('12345', 'bkimminich')
    143:       const testCommand = trainingData.data[0].utterances[0]
```

H	test/api/chatBotSpec.ts **(Line 174)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    171:     it('Responds with product price when asked question with product name', async () => {
    172:       const { token } = await login({
    173:         email: 'bjoern.kimminich@gmail.com',
>>> 174:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    175:       })
    176:       const { json } = await frisby.get(API_URL + '/Products/1')
    177:         .expect('status', 200)
```

H	test/api/chatBotSpec.ts **(Line 205)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    202:     it('Greets back registered user after being told username', async () => {
    203:       const { token } = await login({
    204:         email: `stan@${config.get<string>('application.domain')}`,
>>> 205:         password: 'ship coffin krypt cross estate supply insurance asbestos souvenir'
    206:       })
    207:       await frisby.setup({
    208:         request: {
```

H	test/api/chatBotSpec.ts **(Line 250)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    247:       const functionTest = trainingData.data.filter(data => data.intent === 'queries.functionTest')
    248:       const { token } = await login({
    249:         email: 'bjoern.kimminich@gmail.com',
>>> 250:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    251:       })
    252:       const testCommand = functionTest[0].utterances[0]
    253:       const testResponse = '3be2e438b7f3d04c89d7749f727bb3bd'
```

H	test/api/chatBotSpec.ts **(Line 287)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    284:         },
    285:         body: {
    286:           email: `chatbot-testuser@${config.get<string>('application.domain')}`,
>>> 287:           password: 'testtesttest',
    288:           username: '"',
    289:           role: 'admin'
    290:         }
```

H	test/api/chatBotSpec.ts **(Line 295)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    292:
    293:       const { token } = await login({
    294:         email: `chatbot-testuser@${config.get<string>('application.domain')}`,
>>> 295:         password: 'testtesttest'
    296:       })
    297:
    298:       const functionTest = trainingData.data.filter(data => data.intent ===
'queries.functionTest')
```

H	test/api/dataExportApiSpec.ts **(Line 21)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     18:       headers: jsonHeader,
     19:       body: {
     20:         email: 'bjoern.kimminich@gmail.com',
>>>  21:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
     22:       }
     23:     })
     24:       .expect('status', 200)
```

H	test/api/dataExportApiSpec.ts **(Line 48)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     45:       headers: jsonHeader,
     46:       body: {
     47:         email: 'bjoern.kimminich@gmail.com',
>>>  48:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
     49:       }
     50:     })
     51:       .expect('status', 200)
```

H	test/api/dataExportApiSpec.ts **(Line 77)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     74:        headers: jsonHeader,
     75:        body: {
     76:          email: 'bjoern.kimminich@gmail.com',
>>>  77:          password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
     78:        }
     79:      })
     80:        .expect('status', 200)
```

H	test/api/dataExportApiSpec.ts **(Line 112)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    109:        headers: jsonHeader,
    110:        body: {
    111:          email: 'amy@' + config.get<string>('application.domain'),
>>> 112:          password: 'K1f.....................'
    113:        }
    114:      })
    115:        .expect('status', 200)
```

H	test/api/dataExportApiSpec.ts **(Line 152)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    149:        headers: jsonHeader,
    150:        body: {
    151:          email: 'jim@' + config.get<string>('application.domain'),
>>> 152:          password: 'ncc-1701'
    153:        }
    154:      })
    155:        .expect('status', 200)
```

H	test/api/dataExportApiSpec.ts **(Line 194)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    191:        headers: jsonHeader,
    192:        body: {
    193:          email: 'jim@' + config.get<string>('application.domain'),
>>> 194:          password: 'ncc-1701'
    195:        }
    196:      })
    197:        .expect('status', 200)
```

H	test/api/dataExportApiSpec.ts **(Line 234)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     231:       headers: jsonHeader,
     232:       body: {
     233:         email: 'amy@' + config.get<string>('application.domain'),
>>>  234:         password: 'K1f.....................'
     235:       }
     236:     })
     237:       .expect('status', 200)
```

H	test/api/dataExportApiSpec.ts **(Line 282)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     279:       headers: jsonHeader,
     280:       body: {
     281:         email: 'jim@' + config.get<string>('application.domain'),
>>>  282:         password: 'ncc-1701'
     283:       }
     284:     })
     285:       .expect('status', 200)
```

H	test/api/dataExportApiSpec.ts **(Line 332)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     329:       headers: jsonHeader,
     330:       body: {
     331:         email: 'jim@' + config.get<string>('application.domain'),
>>>  332:         password: 'ncc-1701'
     333:       }
     334:     })
     335:       .expect('status', 200)
```

H	test/api/deliveryApiSpec.ts **(Line 23)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
      20:         headers: jsonHeader,
      21:         body: {
      22:           email: 'jim@' + config.get<string>('application.domain'),
>>>   23:           password: 'ncc-1701'
      24:         }
      25:       })
      26:         .expect('status', 200)
```

H	test/api/deliveryApiSpec.ts **(Line 52)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    49:        headers: jsonHeader,
    50:        body: {
    51:          email: 'ciso@' + config.get<string>('application.domain'),
>>> 52:          password: 'mDLx?94T~1CfVfZMzw@sJ9f?s3L6lbMqE70FfI8^54jbNikY5fymx7c!YbJb'
    53:        }
    54:      })
    55:        .expect('status', 200)
```

H	test/api/deliveryApiSpec.ts **(Line 83)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    80:        headers: jsonHeader,
    81:        body: {
    82:          email: 'jim@' + config.get<string>('application.domain'),
>>> 83:          password: 'ncc-1701'
    84:        }
    85:      })
    86:        .expect('status', 200)
```

H	test/api/deliveryApiSpec.ts **(Line 111)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
   108:        headers: jsonHeader,
   109:        body: {
   110:          email: 'ciso@' + config.get<string>('application.domain'),
>>> 111:          password: 'mDLx?94T~1CfVfZMzw@sJ9f?s3L6lbMqE70FfI8^54jbNikY5fymx7c!YbJb'
   112:        }
   113:      })
   114:        .expect('status', 200)
```

H	test/api/deluxeApiSpec.ts **(Line 35)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    32:      headers: jsonHeader,
    33:      body: {
    34:        email: 'bender@' + config.get<string>('application.domain'),
>>> 35:        password: 'OhG0dPlease1nsertLiquor!'
    36:      }
    37:    })
    38:      .expect('status', 200)
```

H	test/api/deluxeApiSpec.ts **(Line 53)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     50:       headers: jsonHeader,
     51:       body: {
     52:         email: 'ciso@' + config.get<string>('application.domain'),
>>>  53:         password: 'mDLx?94T~1CfVfZMzw@sJ9f?s3L6lbMqE70FfI8^54jbNikY5fymx7c!YbJb'
     54:       }
     55:     })
     56:       .expect('status', 200)
```

H	test/api/deluxeApiSpec.ts **(Line 71)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     68:       headers: jsonHeader,
     69:       body: {
     70:         email: 'admin@' + config.get<string>('application.domain'),
>>>  71:         password: 'admin123'
     72:       }
     73:     })
     74:       .expect('status', 200)
```

H	test/api/deluxeApiSpec.ts **(Line 89)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     86:       headers: jsonHeader,
     87:       body: {
     88:         email: 'accountant@' + config.get<string>('application.domain'),
>>>  89:         password: 'i am an awesome accountant'
     90:       }
     91:     })
     92:       .expect('status', 200)
```

H	test/api/deluxeApiSpec.ts **(Line 105)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    102:   it('POST upgrade deluxe membership status for customers', async () => {
    103:     const { token } = await login({
    104:       email: `bender@${config.get<string>('application.domain')}`,
>>> 105:       password: 'OhG0dPlease1nsertLiquor!'
    106:     })
    107:
    108:     const { json } = await frisby.get(API_URL + '/Cards', {
```

H	test/api/deluxeApiSpec.ts **(Line 129)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

Security Hotspot Code:

```
    126:   it('POST deluxe membership status with wrong card id throws error', async () => {
    127:     const { token } = await login({
    128:       email: `jim@${config.get<string>('application.domain')}`,
>>> 129:       password: 'ncc-1701'
    130:     })
    131:
    132:     await frisby.post(REST_URL + '/deluxe-membership', {
```

H	test/api/deluxeApiSpec.ts **(Line 149)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

Security Hotspot Code:

```
    146:       headers: jsonHeader,
    147:       body: {
    148:         email: 'ciso@' + config.get<string>('application.domain'),
>>> 149:         password: 'mDLx?94T~1CfVfZMzw@sJ9f?s3L6lbMqE70FfI8^54jbNikY5fymx7c!YbJb'
    150:       }
    151:     })
    152:       .expect('status', 200)
```

H	test/api/deluxeApiSpec.ts **(Line 170)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

Security Hotspot Code:

```
    167:       headers: jsonHeader,
    168:       body: {
    169:         email: 'admin@' + config.get<string>('application.domain'),
>>> 170:         password: 'admin123'
    171:       }
    172:     })
    173:       .expect('status', 200)
```

H	test/api/deluxeApiSpec.ts **(Line 191)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

Security Hotspot Code:

```
    188:       headers: jsonHeader,
    189:       body: {
    190:         email: 'accountant@' + config.get<string>('application.domain'),
>>> 191:         password: 'i am an awesome accountant'
    192:       }
    193:     })
    194:       .expect('status', 200)
```

H	test/api/erasureRequestApiSpec.ts **(Line 18)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    15:     headers: jsonHeader,
    16:     body: {
    17:       email: 'bjoern@owasp.org',
>>> 18:       password: 'kitten lesser pooch karate buffoon indoors'
    19:     }
    20:   })
    21:     .expect('status', 200)
```

H	test/api/erasureRequestApiSpec.ts **(Line 37)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    34:     headers: jsonHeader,
    35:     body: {
    36:       email: 'bjoern.kimminich@gmail.com',
>>> 37:       password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    38:     }
    39:   })
    40:     .expect('status', 200)
```

H	test/api/erasureRequestApiSpec.ts **(Line 64)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    61:     headers: jsonHeader,
    62:     body: {
    63:       email: 'bjoern.kimminich@gmail.com',
>>> 64:       password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    65:     }
    66:   })
    67:     .expect('status', 200)
```

H	test/api/erasureRequestApiSpec.ts **(Line 80)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    77:         headers: jsonHeader,
    78:         body: {
    79:           email: 'bjoern.kimminich@gmail.com',
>>> 80:           password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    81:         }
    82:       })
    83:         .expect('status', 200)
```

H	test/api/erasureRequestApiSpec.ts **(Line 99)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    96:        headers: jsonHeader,
    97:        body: {
    98:          email: 'bjoern.kimminich@gmail.com',
>>> 99:          password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
   100:        }
   101:      })
   102:        .expect('status', 200)
```

H	test/api/erasureRequestApiSpec.ts **(Line 119)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
   116:        headers: jsonHeader,
   117:        body: {
   118:          email: 'bjoern.kimminich@gmail.com',
>>> 119:          password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
   120:        }
   121:      })
   122:        .expect('status', 200)
```

H	test/api/erasureRequestApiSpec.ts **(Line 140)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
   137:        headers: jsonHeader,
   138:        body: {
   139:          email: 'bjoern.kimminich@gmail.com',
>>> 140:          password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
   141:        }
   142:      })
   143:        .expect('status', 200)
```

H	test/api/feedbackApiSpec.ts **(Line 120)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
   117:        headers: jsonHeader,
   118:        body: {
   119:          email: 'bjoern.kimminich@gmail.com',
>>> 120:          password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
   121:        }
   122:      })
   123:        .expect('status', 200)
```

H	test/api/feedbackApiSpec.ts **(Line 153)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    150:       headers: jsonHeader,
    151:       body: {
    152:         email: 'bjoern.kimminich@gmail.com',
>>> 153:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    154:       }
    155:     })
    156:       .expect('status', 200)
```

H	test/api/loginApiSpec.ts **(Line 21)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     18:       headers: jsonHeader,
     19:       body: {
     20:         email: 'kalli@kasper.le',
>>>  21:         password: 'kallliiii'
     22:       }
     23:     })
     24:       .expect('status', 201)
```

H	test/api/loginApiSpec.ts **(Line 30)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     27:           headers: jsonHeader,
     28:           body: {
     29:             email: 'kalli@kasper.le',
>>>  30:             password: 'kallliiii'
     31:           }
     32:         })
     33:           .expect('status', 200)
```

H	test/api/loginApiSpec.ts **(Line 46)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     43:   it('POST login non-existing user', () => {
     44:     return frisby.post(REST_URL + '/user/login', {
     45:       email: 'otto@mei.er',
>>>  46:       password: 'ooootto'
     47:     }, { json: true })
     48:       .expect('status', 401)
     49:   })
```

H	test/api/loginApiSpec.ts **(Line 64)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     61:       headers: jsonHeader,
     62:       body: {
     63:         email: 'admin@' + config.get<string>('application.domain'),
>>>  64:         password: 'admin123'
     65:       }
     66:     })
     67:       .expect('status', 200)
```

H	test/api/loginApiSpec.ts **(Line 79)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     76:       headers: jsonHeader,
     77:       body: {
     78:         email: 'support@' + config.get<string>('application.domain'),
>>>  79:         password: 'J6aVjTgOpRs@?5l!Zkq2AYnCE@RF$P'
     80:       }
     81:     })
     82:       .expect('status', 200)
```

H	test/api/loginApiSpec.ts **(Line 94)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     91:       headers: jsonHeader,
     92:       body: {
     93:         email: 'mc.safesearch@' + config.get<string>('application.domain'),
>>>  94:         password: 'Mr. N00dles'
     95:       }
     96:     })
     97:       .expect('status', 200)
```

H	test/api/loginApiSpec.ts **(Line 109)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    106:       headers: jsonHeader,
    107:       body: {
    108:         email: 'amy@' + config.get<string>('application.domain'),
>>> 109:         password: 'K1f.....................'
    110:       }
    111:     })
    112:       .expect('status', 200)
```

H	test/api/loginApiSpec.ts **(Line 124)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    121:        headers: jsonHeader,
    122:        body: {
    123:          email: 'wurstbrot@' + config.get<string>('application.domain'),
>>> 124:          password: 'EinBelegtesBrotMitSchinkenSCHINKEN!'
    125:        }
    126:      })
    127:        .expect('status', 401)
```

H	test/api/loginApiSpec.ts **(Line 142)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    139:        headers: jsonHeader,
    140:        body: {
    141:          email: 'bjoern.kimminich@gmail.com',
>>> 142:          password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    143:        }
    144:      })
    145:        .expect('status', 200)
```

H	test/api/loginApiSpec.ts **(Line 245)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    242:        headers: jsonHeader,
    243:        body: {
    244:          email: 'bjoern.kimminich@gmail.com',
>>> 245:          password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    246:        }
    247:      })
    248:        .expect('status', 200)
```

H	test/api/loginApiSpec.ts **(Line 266)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    263:        headers: jsonHeader,
    264:        body: {
    265:          email: 'bjoern.kimminich@gmail.com',
>>> 266:          password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    267:        }
    268:      })
    269:        .expect('status', 200)
```

H	test/api/memoryApiSpec.ts **(Line 26)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     23:       headers: jsonHeader,
     24:       body: {
     25:         email: 'jim@' + config.get<string>('application.domain'),
>>>  26:         password: 'ncc-1701'
     27:       }
     28:     })
     29:       .expect('status', 200)
```

H	test/api/memoryApiSpec.ts **(Line 64)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     61:       headers: jsonHeader,
     62:       body: {
     63:         email: 'jim@' + config.get<string>('application.domain'),
>>>  64:         password: 'ncc-1701'
     65:       }
     66:     })
     67:       .expect('status', 200)
```

H	test/api/memoryApiSpec.ts **(Line 86)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     83:       headers: jsonHeader,
     84:       body: {
     85:         email: 'jim@' + config.get<string>('application.domain'),
>>>  86:         password: 'ncc-1701'
     87:       }
     88:     })
     89:       .expect('status', 200)
```

H	test/api/memoryApiSpec.ts **(Line 112)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    109:       headers: jsonHeader,
    110:       body: {
    111:         email: 'jim@' + config.get<string>('application.domain'),
>>> 112:         password: 'ncc-1701'
    113:       }
    114:     })
    115:       .expect('status', 200)
```

H	test/api/memoryApiSpec.ts **(Line 142)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    139:       headers: jsonHeader,
    140:       body: {
    141:         email: 'jim@' + config.get<string>('application.domain'),
>>> 142:         password: 'ncc-1701'
    143:       }
    144:     })
    145:       .expect('status', 200)
```

H	test/api/orderHistoryApiSpec.ts **(Line 19)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     16:       headers: jsonHeader,
     17:       body: {
     18:         email: 'admin@' + config.get<string>('application.domain'),
>>>  19:         password: 'admin123'
     20:       }
     21:     })
     22:       .expect('status', 200)
```

H	test/api/orderHistoryApiSpec.ts **(Line 56)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     53:       headers: jsonHeader,
     54:       body: {
     55:         email: 'jim@' + config.get<string>('application.domain'),
>>>  56:         password: 'ncc-1701'
     57:       }
     58:     })
     59:       .expect('status', 200)
```

H	test/api/orderHistoryApiSpec.ts **(Line 73)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     70:       headers: jsonHeader,
     71:       body: {
     72:         email: 'admin@' + config.get<string>('application.domain'),
>>>  73:         password: 'admin123'
     74:       }
     75:     })
     76:       .expect('status', 200)
```

H	test/api/orderHistoryApiSpec.ts **(Line 90)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     87:       headers: jsonHeader,
     88:       body: {
     89:         email: 'accountant@' + config.get<string>('application.domain'),
>>>  90:         password: 'i am an awesome accountant'
     91:       }
     92:     })
     93:       .expect('status', 200)
```

H	test/api/orderHistoryApiSpec.ts **(Line 109)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    106:       headers: jsonHeader,
    107:       body: {
    108:         email: 'admin@' + config.get<string>('application.domain'),
>>> 109:         password: 'admin123'
    110:       }
    111:     })
    112:       .expect('status', 200)
```

H	test/api/orderHistoryApiSpec.ts **(Line 129)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    126:       headers: jsonHeader,
    127:       body: {
    128:         email: 'jim@' + config.get<string>('application.domain'),
>>> 129:         password: 'ncc-1701'
    130:       }
    131:     })
    132:       .expect('status', 200)
```

H	test/api/orderHistoryApiSpec.ts **(Line 149)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    146:       headers: jsonHeader,
    147:       body: {
    148:         email: 'accountant@' + config.get<string>('application.domain'),
>>> 149:         password: 'i am an awesome accountant'
    150:       }
    151:     })
    152:       .expect('status', 200)
```

H	test/api/passwordApiSpec.ts **(Line 20)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     17:       headers: jsonHeader,
     18:       body: {
     19:         email: 'kuni@be.rt',
>>>  20:         password: 'kunigunde'
     21:       }
     22:     })
     23:       .expect('status', 201)
```

H	test/api/passwordApiSpec.ts **(Line 29)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     26:           headers: jsonHeader,
     27:           body: {
     28:             email: 'kuni@be.rt',
>>>  29:             password: 'kunigunde'
     30:           }
     31:         })
     32:           .expect('status', 200)
```

H	test/api/passwordApiSpec.ts **(Line 47)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     44:       headers: jsonHeader,
     45:       body: {
     46:         email: 'bjoern@' + config.get<string>('application.domain'),
>>>  47:         password: 'monkey summer birthday are all bad passwords but work just fine in a long passphrase'
     48:       }
     49:     })
     50:       .expect('status', 200)
```

H	test/api/passwordApiSpec.ts **(Line 93)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     90:       headers: jsonHeader,
     91:       body: {
     92:         email: 'bender@' + config.get<string>('application.domain'),
>>>  93:         password: 'OhG0dPlease1nsertLiquor!'
     94:       }
     95:     })
     96:       .expect('status', 200)
```

H	test/api/paymentApiSpec.ts **(Line 20)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     17:     headers: jsonHeader,
     18:     body: {
     19:       email: 'jim@juice-sh.op',
>>>  20:       password: 'ncc-1701'
     21:     }
     22:   })
     23:     .expect('status', 200)
```

H	test/api/productReviewApiSpec.ts **(Line 112)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    109:       headers: jsonHeader,
    110:       body: {
    111:         email: 'bjoern.kimminich@gmail.com',
>>> 112:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    113:       }
    114:     })
    115:       .expect('status', 200)
```

H	test/api/productReviewApiSpec.ts **(Line 132)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    129:       headers: jsonHeader,
    130:       body: {
    131:         email: 'bjoern.kimminich@gmail.com',
>>> 132:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    133:       }
    134:     })
    135:       .expect('status', 200)
```

H	test/api/profileImageUploadSpec.ts **(Line 25)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     22:       headers: jsonHeader,
     23:       body: {
     24:         email: `jim@${config.get<string>('application.domain')}`,
>>>  25:         password: 'ncc-1701'
     26:       }
     27:     })
     28:       .expect('status', 200)
```

H	test/api/profileImageUploadSpec.ts **(Line 52)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     49:       headers: jsonHeader,
     50:       body: {
     51:         email: `jim@${config.get<string>('application.domain')}`,
>>>  52:         password: 'ncc-1701'
     53:       }
     54:     })
     55:       .expect('status', 200)
```

H	test/api/profileImageUploadSpec.ts **(Line 97)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     94:       headers: jsonHeader,
     95:       body: {
     96:         email: `jim@${config.get<string>('application.domain')}`,
>>>  97:         password: 'ncc-1701'
     98:       }
     99:     })
    100:       .expect('status', 200)
```

H	test/api/profileImageUploadSpec.ts **(Line 123)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    120:       headers: jsonHeader,
    121:       body: {
    122:         email: `jim@${config.get<string>('application.domain')}`,
>>> 123:         password: 'ncc-1701'
    124:       }
    125:     })
    126:       .expect('status', 200)
```

H	test/api/profileImageUploadSpec.ts **(Line 164)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    161:       headers: jsonHeader,
    162:       body: {
    163:         email: `jim@${config.get<string>('application.domain')}`,
>>> 164:         password: 'ncc-1701'
    165:       }
    166:     })
    167:       .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 21)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     18:       headers: jsonHeader,
     19:       body: {
     20:         email: `jim@${config.get<string>('application.domain')}`,
>>>  21:         password: 'ncc-1701'
     22:       }
     23:     })
     24:       .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 38)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     35:       headers: jsonHeader,
     36:       body: {
     37:         email: `admin@${config.get<string>('application.domain')}`,
>>>  38:         password: 'admin123'
     39:       }
     40:     })
     41:       .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 55)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     52:       headers: jsonHeader,
     53:       body: {
     54:         email: `accountant@${config.get<string>('application.domain')}`,
>>>  55:         password: 'i am an awesome accountant'
     56:       }
     57:     })
     58:       .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 72)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     69:       headers: jsonHeader,
     70:       body: {
     71:         email: `jim@${config.get<string>('application.domain')}`,
>>>  72:         password: 'ncc-1701'
     73:       }
     74:     })
     75:       .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 93)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     90:       headers: jsonHeader,
     91:       body: {
     92:         email: `admin@${config.get<string>('application.domain')}`,
>>>  93:         password: 'admin123'
     94:       }
     95:     })
     96:       .expect('status', 200)
```

H

test/api/quantityApiSpec.ts
(Line 114)

typescript:S2068
Review this potentially hard-coded password.
Category: Authentication

■ **Security Hotspot Code:**

```
    111:       headers: jsonHeader,
    112:       body: {
    113:         email: `accountant@${config.get<string>('application.domain')}`,
>>> 114:         password: 'i am an awesome accountant'
    115:       }
    116:     })
    117:       .expect('status', 200)
```

H

test/api/quantityApiSpec.ts
(Line 137)

typescript:S2068
Review this potentially hard-coded password.
Category: Authentication

■ **Security Hotspot Code:**

```
    134:       headers: jsonHeader,
    135:       body: {
    136:         email: `jim@${config.get<string>('application.domain')}`,
>>> 137:         password: 'ncc-1701'
    138:       }
    139:     })
    140:       .expect('status', 200)
```

H

test/api/quantityApiSpec.ts
(Line 155)

typescript:S2068
Review this potentially hard-coded password.
Category: Authentication

■ **Security Hotspot Code:**

```
    152:       headers: jsonHeader,
    153:       body: {
    154:         email: `admin@${config.get<string>('application.domain')}`,
>>> 155:         password: 'admin123'
    156:       }
    157:     })
    158:       .expect('status', 200)
```

H

test/api/quantityApiSpec.ts
(Line 173)

typescript:S2068
Review this potentially hard-coded password.
Category: Authentication

■ **Security Hotspot Code:**

```
    170:       headers: jsonHeader,
    171:       body: {
    172:         email: `accountant@${config.get<string>('application.domain')}`,
>>> 173:         password: 'i am an awesome accountant'
    174:       }
    175:     })
    176:       .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 190)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    187:       headers: jsonHeader,
    188:       body: {
    189:         email: `accountant@${config.get<string>('application.domain')}`,
>>> 190:         password: 'i am an awesome accountant'
    191:       }
    192:     })
    193:       .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 207)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    204:       headers: jsonHeader,
    205:       body: {
    206:         email: `jim@${config.get<string>('application.domain')}`,
>>> 207:         password: 'ncc-1701'
    208:       }
    209:     })
    210:       .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 228)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    225:       headers: jsonHeader,
    226:       body: {
    227:         email: `jim@${config.get<string>('application.domain')}`,
>>> 228:         password: 'ncc-1701'
    229:       }
    230:     })
    231:       .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 249)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    246:        headers: jsonHeader,
    247:        body: {
    248:          email: `accountant@${config.get<string>('application.domain')}`,
>>> 249:          password: 'i am an awesome accountant'
    250:        }
    251:      })
    252:        .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 269)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    266:        headers: jsonHeader,
    267:        body: {
    268:          email: `accountant@${config.get<string>('application.domain')}`,
>>> 269:          password: 'i am an awesome accountant'
    270:        }
    271:      })
    272:        .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 292)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    289:        headers: jsonHeader,
    290:        body: {
    291:          email: `accountant@${config.get<string>('application.domain')}`,
>>> 292:          password: 'i am an awesome accountant'
    293:        }
    294:      })
    295:        .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 309)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    306:        headers: jsonHeader,
    307:        body: {
    308:          email: `admin@${config.get<string>('application.domain')}`,
>>> 309:          password: 'admin123'
    310:        }
    311:      })
    312:        .expect('status', 200)
```

H	test/api/quantityApiSpec.ts **(Line 326)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    323:       headers: jsonHeader,
    324:       body: {
    325:         email: `jim@${config.get<string>('application.domain')}`,
>>> 326:         password: 'ncc-1701'
    327:       }
    328:     })
    329:       .expect('status', 200)
```

H	test/api/securityAnswerApiSpec.ts **(Line 44)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     41:   it('POST security answer for a newly registered user', () => {
     42:     return frisby.post(`${API_URL}/Users`, {
     43:       email: 'new.user@te.st',
>>>  44:       password: '12345'
     45:     }, { json: true })
     46:       .expect('status', 201)
     47:       .then(({ json }) => {
```

H	test/api/userApiSpec.ts **(Line 45)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     42:       headers: jsonHeader,
     43:       body: {
     44:         email: 'horst@horstma.nn',
>>>  45:         password: 'hooooorst'
     46:       }
     47:     })
     48:       .expect('status', 201)
```

H	test/api/userApiSpec.ts **(Line 63)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     60:       headers: jsonHeader,
     61:       body: {
     62:         email: 'horst2@horstma.nn',
>>>  63:         password: 'hooooorst',
     64:         role: 'admin'
     65:       }
     66:     })
```

H	test/api/userApiSpec.ts **(Line 85)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     82:         headers: jsonHeader,
     83:         body: {
     84:           email: ' ',
>>>  85:           password: ' '
     86:         }
     87:       })
     88:         .expect('status', 201)
```

H	test/api/userApiSpec.ts **(Line 103)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    100:         headers: jsonHeader,
    101:         body: {
    102:           email: ' ',
>>> 103:           password: ' '
    104:         }
    105:       }).post(`${API_URL}/Users`, {
    106:         headers: jsonHeader,
```

H	test/api/userApiSpec.ts **(Line 109)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    106:         headers: jsonHeader,
    107:         body: {
    108:           email: ' ',
>>> 109:           password: ' '
    110:         }
    111:       })
    112:         .expect('status', 400)
```

H	test/api/userApiSpec.ts **(Line 121)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    118:         headers: jsonHeader,
    119:         body: {
    120:           email: ' test@gmail.com',
>>> 121:           password: ' test'
    122:         }
    123:       })
    124:         .expect('status', 201)
```

H	test/api/userApiSpec.ts **(Line 139)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    136:       headers: jsonHeader,
    137:       body: {
    138:         email: 'horst3@horstma.nn',
>>> 139:         password: 'hooooorst',
    140:         role: 'deluxe'
    141:       }
    142:     })
```

H	test/api/userApiSpec.ts **(Line 161)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    158:       headers: jsonHeader,
    159:       body: {
    160:         email: 'horst4@horstma.nn',
>>> 161:         password: 'hooooorst',
    162:         role: 'accounting'
    163:       }
    164:     })
```

H	test/api/userApiSpec.ts **(Line 183)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    180:       headers: jsonHeader,
    181:       body: {
    182:         email: 'horst5@horstma.nn',
>>> 183:         password: 'hooooorst',
    184:         role: 'accountinguser'
    185:       }
    186:     })
```

H	test/api/userApiSpec.ts **(Line 202)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    199:         headers: jsonHeader,
    200:         body: {
    201:           email: '<iframe src="javascript:alert(`xss`)">',
>>> 202:           password: 'does.not.matter'
    203:         }
    204:       })
    205:         .expect('status', 201)
```

H	test/api/userApiSpec.ts **(Line 256)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    253:       headers: jsonHeader,
    254:       body: {
    255:         email: 'bjoern.kimminich@gmail.com',
>>> 256:         password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
    257:       }
    258:     })
    259:       .expect('status', 200)
```

H	test/api/userProfileSpec.ts **(Line 19)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     16:     headers: jsonHeader,
     17:     body: {
     18:       email: 'jim@juice-sh.op',
>>>  19:       password: 'ncc-1701'
     20:     }
     21:   })
     22:     .expect('status', 200)
```

H	test/api/walletApiSpec.ts **(Line 18)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     15:     headers: jsonHeader,
     16:     body: {
     17:       email: 'demo',
>>>  18:       password: 'demo'
     19:     }
     20:   })
     21:     .expect('status', 200)
```

H	test/cypress/e2e/administration.spec.ts **(Line 5)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
      2:   beforeEach(() => {
      3:     cy.login({
      4:       email: 'admin',
>>>   5:       password: 'admin123'
      6:     })
      7:   })
      8:   describe('challenge "adminSection"', () => {
```

H	test/cypress/e2e/b2bOrder.spec.ts **(Line 6)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
      3:     it('an infinite loop deserialization payload should not bring down the server', ()
=> {
      4:       cy.task('isDocker').then((isDocker) => {
      5:         if (!isDocker) {
>>>   6:           cy.login({ email: 'admin', password: 'admin123' })
      7:
      8:           cy.window().then(async () => {
      9:             const response = await fetch(
```

H	test/cypress/e2e/b2bOrder.spec.ts **(Line 37)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     34:     it('should be possible to cause request timeout using a recursive regular
expression payload', () => {
     35:       cy.task('isDocker').then((isDocker) => {
     36:         if (!isDocker) {
>>>  37:           cy.login({ email: 'admin', password: 'admin123' })
     38:           cy.window().then(async () => {
     39:             const response = await fetch(
     40:               `${Cypress.config('baseUrl')}/b2b/v2/orders/`,
```

H	test/cypress/e2e/basket.spec.ts **(Line 4)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
      1: describe('/#/basket', () => {
      2:   describe('as admin', () => {
      3:     beforeEach(() => {
>>>   4:       cy.login({ email: 'admin', password: 'admin123' })
      5:     })
      6:
      7:     describe('challenge "negativeOrder"', () => {
```

H	test/cypress/e2e/basket.spec.ts **(Line 76)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     73:
     74:   describe('as jim', () => {
     75:     beforeEach(() => {
>>>  76:       cy.login({ email: 'jim', password: 'ncc-1701' })
     77:     })
     78:     describe('challenge "manipulateClock"', () => {
     79:       it('should be possible to enter WMNSDY2019 coupon & place order with this expired
coupon', () => {
```

H	test/cypress/e2e/changePassword.spec.ts **(Line 6)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

Security Hotspot Code:

```
     3:     beforeEach(() => {
     4:       cy.login({
     5:         email: 'morty',
>>>  6:         password: 'focusOnScienceMorty!focusOnScience'
     7:       })
     8:       cy.visit('/#/privacy-security/change-password')
     9:     })
```

H	test/cypress/e2e/changePassword.spec.ts **(Line 25)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

Security Hotspot Code:

```
    22:     it('should be able to change password via XSS-powered attack on password change without passing current password', () => {
    23:       cy.login({
    24:         email: 'bender',
>>> 25:         password: 'OhG0dPlease1nsertLiquor!'
    26:       })
    27:       cy.visit(
    28:         '/#/search?q=%3Ciframe%20src%3D%22javascript%3Axmlhttp%20%3D%20new%20XMLHttpRequest%28%29%3B%20xmlhttp.open%28%27GET%27%2C%20%27http%3A%2F%2Flocalhost%3A3000%2Frest%2Fuser%2Fchange-password%3Fnew%3DslurmCl4ssic%26amp%3Brepeat%3DslurmCl4ssic%27%29%3B%20xmlhttp.setRequestHeader%28%27Authorization%27%2C%60Bearer%3D%24%7BlocalStorage.getItem%28%27token%27%29%7D%60%29%3B%20xmlhttp.send%28%29%3B%22%3E'
```

H	test/cypress/e2e/changePassword.spec.ts **(Line 31)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

Security Hotspot Code:

```
    28:         '/#/search?q=%3Ciframe%20src%3D%22javascript%3Axmlhttp%20%3D%20new%20XMLHttpRequest%28%29%3B%20xmlhttp.open%28%27GET%27%2C%20%27http%3A%2F%2Flocalhost%3A3000%2Frest%2Fuser%2Fchange-password%3Fnew%3DslurmCl4ssic%26amp%3Brepeat%3DslurmCl4ssic%27%29%3B%20xmlhttp.setRequestHeader%28%27Authorization%27%2C%60Bearer%3D%24%7BlocalStorage.getItem%28%27token%27%29%7D%60%29%3B%20xmlhttp.send%28%29%3B%22%3E'
    29:       )
    30:       cy.wait(2000)
>>> 31:       cy.login({ email: 'bender', password: 'slurmCl4ssic' })
    32:       cy.url().should('match', /\/search/)
    33:
    34:       cy.expectChallengeSolved({ challenge: "Change Bender's Password" })
```

H	test/cypress/e2e/chatbot.spec.ts **(Line 3)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

Security Hotspot Code:

```
     1: describe('/chatbot', () => {
     2:   beforeEach(() => {
>>>  3:     cy.login({ email: 'admin', password: 'admin123' })
     4:   })
     5:
     6:   describe('challenge "killChatbot"', () => {
```

H	test/cypress/e2e/complain.spec.ts **(Line 5)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
      2:   beforeEach(() => {
      3:     cy.login({
      4:       email: 'admin',
>>>   5:       password: 'admin123'
      6:     })
      7:
      8:     cy.visit('/#/complain')
```

H	test/cypress/e2e/contact.spec.ts **(Line 11)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
      8:
      9:   describe('challenge "forgedFeedback"', () => {
     10:     beforeEach(() => {
>>>  11:       cy.login({ email: 'admin', password: 'admin123' })
     12:       cy.visit('/#/contact')
     13:       solveNextCaptcha()
     14:     })
```

H	test/cypress/e2e/contact.spec.ts **(Line 47)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     44:
     45:   describe('challenge "persistedXssFeedback"', () => {
     46:     beforeEach(() => {
>>>  47:       cy.login({ email: 'admin', password: 'admin123' })
     48:       cy.visit('/#/contact')
     49:       solveNextCaptcha()
     50:     })
```

H	test/cypress/e2e/dataErasure.spec.ts **(Line 3)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
      1: describe('/dataerasure', () => {
      2:   beforeEach(() => {
>>>   3:     cy.login({ email: 'admin', password: 'admin123' })
      4:   })
      5:
      6:   describe('challenge "lfr"', () => {
```

H	test/cypress/e2e/dataExport.spec.ts **(Line 24)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     21:     })
     22:
     23:     it('should be possible to steal admin user data by causing email clash during
export', () => {
>>>  24:       cy.login({ email: 'admun', password: 'admun123' })
     25:
     26:       cy.visit('/#/privacy-security/data-export')
     27:       cy.get('#formatControl').contains('JSON').click()
```

H	test/cypress/e2e/deluxe.spec.ts **(Line 4)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
      1: describe('/#/deluxe-membership', () => {
      2:   describe('challenge "svgInjection"', () => {
      3:     it('should be possible to pass in a forgotten test parameter abusing the
redirect-endpoint to load an external image', () => {
>>>   4:       cy.login({ email: 'jim', password: 'ncc-1701' })
      5:       cy.location().then((loc) => {
      6:         cy.visit(
      7:           `/#/deluxe-membership?testDecal=${encodeURIComponent(
```

H	test/cypress/e2e/deluxe.spec.ts **(Line 21)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     18:     it('should upgrade to deluxe for free by making a post request to
/rest/deluxe-membership by setting the paymentMode parameter to null', () => {
     19:       cy.login({
     20:         email: 'jim',
>>>  21:         password: 'ncc-1701'
     22:       })
     23:       cy.visit('/#/')
     24:       cy.getCookie('token').then((token) => {
```

H	test/cypress/e2e/noSql.spec.ts **(Line 8)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
      5:
      6:   describe('challenge "NoSQL DoS"', () => {
      7:     beforeEach(() => {
>>>   8:       cy.login({ email: 'admin', password: 'admin123' })
      9:     })
     10:     it('should be possible to inject a command into the get route', () => {
     11:       cy.task('isDocker').then((isDocker) => {
```

H	test/cypress/e2e/noSql.spec.ts **(Line 53)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     50:
     51:  describe('challenge "NoSQL Manipulation"', () => {
     52:    beforeEach(() => {
>>>  53:      cy.login({ email: 'admin', password: 'admin123' })
     54:    })
     55:
     56:    it('should be possible to inject a selector into the update route', () => {
```

H	test/cypress/e2e/noSql.spec.ts **(Line 76)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
     73:
     74:  describe('challenge "Forged Review"', () => {
     75:    beforeEach(() => {
>>>  76:      cy.login({ email: 'mc.safesearch', password: 'Mr. N00dles' })
     77:    })
     78:
     79:    it('should be possible to edit any existing review', () => {
```

H	test/cypress/e2e/noSql.spec.ts **(Line 120)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
    117:
    118:  describe('challenge "Multiple Likes"', () => {
    119:    beforeEach(() => {
>>> 120:      cy.login({ email: 'mc.safesearch', password: 'Mr. N00dles' })
    121:    })
    122:
    123:    it('should be possible to like reviews multiple times', () => {
```

H	test/cypress/e2e/profile.spec.ts **(Line 3)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ **Security Hotspot Code:**

```
      1: describe('/profile', () => {
      2:   beforeEach(() => {
>>>   3:     cy.login({ email: 'admin', password: 'admin123' })
      4:   })
      5:   describe('challenge "ssrf"', () => {
      6:     it('should be possible to request internal resources using image upload URL', () =>
{
```

H	test/cypress/e2e/register.spec.ts **(Line 10)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
      7:     beforeEach(() => {
      8:       cy.login({
      9:         email: 'admin',
>>>  10:         password: 'admin123'
     11:       })
     12:     })
     13:
```

H	test/cypress/e2e/register.spec.ts **(Line 28)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     25:                 },
     26:                 body: JSON.stringify({
     27:                   email: '<iframe src="javascript:alert(`xss`)">',
>>>  28:                   password: 'XSSed',
     29:                   passwordRepeat: 'XSSed',
     30:                   role: 'admin'
     31:                 })
```

H	test/cypress/e2e/register.spec.ts **(Line 29)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     26:                 body: JSON.stringify({
     27:                   email: '<iframe src="javascript:alert(`xss`)">',
     28:                   password: 'XSSed',
>>>  29:                   passwordRepeat: 'XSSed',
     30:                   role: 'admin'
     31:                 })
     32:               }
```

H	test/cypress/e2e/register.spec.ts **(Line 60)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     57:           },
     58:           body: JSON.stringify({
     59:             email: 'testing@test.com',
>>>  60:             password: 'pwned',
     61:             passwordRepeat: 'pwned',
     62:             role: 'admin'
     63:           })
```

H	test/cypress/e2e/register.spec.ts **(Line 61)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     58:           body: JSON.stringify({
     59:             email: 'testing@test.com',
     60:             password: 'pwned',
>>>  61:             passwordRepeat: 'pwned',
     62:             role: 'admin'
     63:           })
     64:         })
```

H	test/cypress/e2e/register.spec.ts **(Line 84)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     81:           },
     82:           body: JSON.stringify({
     83:             email: 'uncle@bob.com',
>>>  84:             password: 'ThereCanBeOnlyOne'
     85:           })
     86:         })
     87:         if (response.status === 201) {
```

H	test/cypress/e2e/restApi.spec.ts **(Line 4)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
      1: describe('/api', () => {
      2:   describe('challenge "restfulXss"', () => {
      3:     beforeEach(() => {
>>>   4:       cy.login({ email: 'admin', password: 'admin123' })
      5:     })
      6:
      7:     // Cypress alert bug
```

H	test/cypress/e2e/restApi.spec.ts **(Line 82)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     79:     beforeEach(() => {
     80:       cy.login({
     81:         email: 'admin',
>>>  82:         password: 'admin123'
     83:       })
     84:     })
     85:
```

H	test/cypress/e2e/search.spec.ts **(Line 56)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    53:     beforeEach(() => {
    54:       cy.login({
    55:         email: 'admin',
>>> 56:         password: 'admin123'
    57:       })
    58:     })
    59:
```

H	test/cypress/e2e/search.spec.ts **(Line 83)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    80:     beforeEach(() => {
    81:       cy.login({
    82:         email: 'admin',
>>> 83:         password: 'admin123'
    84:       })
    85:     })
    86:
```

H	test/cypress/e2e/totpSetup.spec.ts **(Line 6)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
     3:     beforeEach(() => {
     4:       cy.login({
     5:         email: 'wurstbrot',
>>>  6:         password: 'EinBelegtesBrotMitSchinkenSCHINKEN!',
     7:         totpSecret: 'IFTXE3SPOEYVURT2MRYGI52TKJ4HC3KH'
     8:       })
     9:     })
```

H	test/cypress/e2e/totpSetup.spec.ts **(Line 20)**	**typescript:S2068** Review this potentially hard-coded password. Category: Authentication

■ Security Hotspot Code:

```
    17:     beforeEach(() => {
    18:       cy.login({
    19:         email: 'amy',
>>> 20:         password: 'K1f.....................'
    21:       })
    22:     })
    23:
```

H	Dockerfile **(Line 38)**	**docker:S6504** Make sure no write permissions are assigned to the copied resource. Category: Authentication

■ **Security Hotspot Code:**

```
     35:     org.opencontainers.image.revision=$VCS_REF \
     36:     org.opencontainers.image.created=$BUILD_DATE
     37: WORKDIR /juice-shop
>>>  38: COPY --from=installer --chown=65532:0 /juice-shop .
     39: USER 65532
     40: EXPOSE 3000
     41: CMD ["/juice-shop/build/app.js"]
```

Denial of Service (DoS) (10 hotspots)

Risk	File Path	Rule & Message
M	server.ts **(Line 682)**	**typescript:S5693** Make sure the content length limit is safe here. Category: Denial of Service (DoS)

■ **Security Hotspot Code:**

```
    679:   'image/jpeg': 'jpg',
    680:   'image/jpg': 'jpg'
    681: }
>>> 682: const uploadToDisk = multer({
    683:   storage: multer.diskStorage({
    684:     destination: (req: Request, file: any, cb: any) => {
    685:       const isValid = mimeTypeMap[file.mimetype]
```

M	frontend/src/app/change-password change-password.component.ts **(Line 94)**	**typescript:S5852** Make sure the regex used here, which is vulnerable to super-linear runtime due to backtracking, cannot lead to denial of service. Category: Denial of Service (DoS)

■ **Security Hotspot Code:**

```
     91:     if (
     92:       localStorage.getItem('email')?.match(/support@.*/) &&
     93:       !this.newPasswordControl.value.match(
>>>  94:         /(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-z\d@$!%*?&]{12,30}/
     95:       )
     96:     ) {
     97:       console.error(
```

M	lib/codingChallenges.ts **(Line 66)**	**typescript:S5852** Make sure the regex used here, which is vulnerable to super-linear runtime due to backtracking, cannot lead to denial of service. Category: Denial of Service (DoS)

■ Security Hotspot Code:

```
     63:   let snippet = snippets[0] // TODO Currently only a single code snippet is supported
     64:   snippet = snippet.replace(/\s?[/#]{0,2} vuln-code-snippet start.*[\r\n]{0,2}/g, '')
     65:   snippet = snippet.replace(/\s?[/#]{0,2} vuln-code-snippet end.*/g, '')
>>>  66:   snippet = snippet.replace(/.*[/#]{0,2} vuln-code-snippet hide-line[\r\n]{0,2}/g, '')
     67:   snippet = snippet.replace(/.*[/#]{0,2} vuln-code-snippet hide-start([^])*[/#]{0,2}
vuln-code-snippet hide-end[\r\n]{0,2}/g, '')
     68:   snippet = snippet.trim()
     69:
```

M	lib/codingChallenges.ts **(Line 67)**	**typescript:S5852** Make sure the regex used here, which is vulnerable to super-linear runtime due to backtracking, cannot lead to denial of service. Category: Denial of Service (DoS)

■ Security Hotspot Code:

```
     64:   snippet = snippet.replace(/\s?[/#]{0,2} vuln-code-snippet start.*[\r\n]{0,2}/g, '')
     65:   snippet = snippet.replace(/\s?[/#]{0,2} vuln-code-snippet end.*/g, '')
     66:   snippet = snippet.replace(/.*[/#]{0,2} vuln-code-snippet hide-line[\r\n]{0,2}/g, '')
>>>  67:   snippet = snippet.replace(/.*[/#]{0,2} vuln-code-snippet hide-start([^])*[/#]{0,2}
vuln-code-snippet hide-end[\r\n]{0,2}/g, '')
     68:   snippet = snippet.trim()
     69:
     70:   let lines = snippet.split('\r\n')
```

M	lib/startup/registerWebsocketEvents.ts **(Line 47)**	**typescript:S5852** Make sure the regex used here, which is vulnerable to super-linear runtime due to backtracking, cannot lead to denial of service. Category: Denial of Service (DoS)

■ Security Hotspot Code:

```
     44:     })
     45:
     46:     socket.on('verifySvgInjectionChallenge', (data: any) => {
>>>  47:       challengeUtils.solveIf(challenges.svgInjectionChallenge, () => { return data?.ma
tch(/.*\.\.\/\.\.\/\.\.[\w/-]*?\/redirect\?to=https?:\/\/placecats.com\/(g\/)?[\d]+\/[\d]+.*/)
&& security.isRedirectAllowed(data) })
     48:     })
     49:
     50:     socket.on('verifyCloseNotificationsChallenge', (data: any) => {
```

M	lib/utils.ts **(Line 232)**	**typescript:S5852** Make sure the regex used here, which is vulnerable to super-linear runtime due to backtracking, cannot lead to denial of service. Category: Denial of Service (DoS)

■ Security Hotspot Code:

```
    229: }
    230:
    231: export const matchesEtcPasswdFile = (text: string) => {
>>> 232:   const match = text.match(/(\w*:\w*:\d*:\d*:\w*:.*)|(Note that this file is consulted
directly)/gi)
    233:   return match !== null && match.length >= 1
    234: }
    235:
```

M	routes/profileImageUrlUpload.ts **(Line 20)**	**typescript:S5852** Make sure the regex used here, which is vulnerable to super-linear runtime due to backtracking, cannot lead to denial of service. Category: Denial of Service (DoS)

■ Security Hotspot Code:

```
     17:   return async (req: Request, res: Response, next: NextFunction) => {
     18:     if (req.body.imageUrl !== undefined) {
     19:       const url = req.body.imageUrl
>>>  20:       if (url.match(/(.)*solve\/challenges\/server-side(.)*/) !== null)
req.app.locals.abused_ssrf_bug = true
     21:       const loggedInUser = security.authenticatedUsers.get(req.cookies.token)
     22:       if (loggedInUser) {
     23:         try {
```

M	server.ts **(Line 246)**	**typescript:S5852** Make sure the regex used here, which is vulnerable to super-linear runtime due to backtracking, cannot lead to denial of service. Category: Denial of Service (DoS)

■ Security Hotspot Code:

```
    243:     // @ts-expect-error FIXME assignment broken due to seemingly void return value
    244:     res.end = function () {
    245:       if (arguments.length) {
>>> 246:         const reqPath = req.originalUrl.replace(/\?.*$/, '')
    247:         // eslint-disable-next-line @typescript-eslint/no-non-null-assertion
    248:         const currentFolder = reqPath.split('/').pop()!
    249:         arguments[0] = arguments[0].replace(/a href="([^"]+?)"/gi, function
(matchString: string, matchedUrl: string) {
```

M	test/api/metricsApiSpec.ts **(Line 18)**	**typescript:S5852** Make sure the regex used here, which is vulnerable to super-linear runtime due to backtracking, cannot lead to denial of service. Category: Denial of Service (DoS)

■ Security Hotspot Code:

```
     15:     return frisby.get(API_URL)
     16:       .expect('status', 200)
     17:       .expect('header', 'content-type', /text\/plain/)
>>>  18:       .expect('bodyContains', /^.*_version_info{version="[0-9]+.[0-9]+.[0-9]+(-SNAPSHO
T)?",major="[0-9]+",minor="[0-9]+",patch="[0-9]+",app=".*"} 1$/gm)
     19:       .expect('bodyContains',
/^.*_challenges_solved{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     20:       .expect('bodyContains',
/^.*_challenges_total{difficulty="[1-6]",category=".*",app=".*"} [0-9]*$/gm)
     21:       .expect('bodyContains', /^.*_cheat_score{app=".*"} [0-9.]*$/gm)
```

M	test/cypress/support/commands.ts **(Line 36)**	**typescript:S5852** Make sure the regex used here, which is vulnerable to super-linear runtime due to backtracking, cannot lead to denial of service. Category: Denial of Service (DoS)

■ **Security Hotspot Code:**

```
     33:   'login',
     34:   (context: { email: string, password: string, totpSecret?: string }) => {
     35:     cy.visit('/#/login')
>>>  36:     if (context.email.match(/\S+@\S+\.\S+/) != null) {
     37:       cy.get('#email').type(context.email)
     38:     } else {
     39:       cy.task<string>('GetFromConfig', 'application.domain').then(
```

Encrypt Data (42 hotspots)

Risk	File Path	Rule & Message
L	data/static/codefixes redirectChallenge_1.ts **(Line 6)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      3:   'https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
      4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
      5:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
>>>   6:   'http://shop.spreadshirt.com/juiceshop',
      7:   'http://shop.spreadshirt.de/juiceshop',
      8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
      9:   'http://leanpub.com/juice-shop'
```

L	data/static/codefixes redirectChallenge_1.ts **(Line 7)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
      5:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
      6:   'http://shop.spreadshirt.com/juiceshop',
>>>   7:   'http://shop.spreadshirt.de/juiceshop',
      8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
      9:   'http://leanpub.com/juice-shop'
     10: ])
```

L	data/static/codefixes redirectChallenge_1.ts **(Line 9)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      6:   'http://shop.spreadshirt.com/juiceshop',
      7:   'http://shop.spreadshirt.de/juiceshop',
      8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
>>>   9:   'http://leanpub.com/juice-shop'
     10: ])
     11:
     12: export const isRedirectAllowed = (url: string) => {
```

L	data/static/codefixes redirectChallenge_2.ts **(Line 6)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      3:   'https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
      4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
      5:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
>>>   6:   'http://shop.spreadshirt.com/juiceshop',
      7:   'http://shop.spreadshirt.de/juiceshop',
      8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
      9:   'http://leanpub.com/juice-shop'
```

L	data/static/codefixes redirectChallenge_2.ts **(Line 7)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
      5:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
      6:   'http://shop.spreadshirt.com/juiceshop',
>>>   7:   'http://shop.spreadshirt.de/juiceshop',
      8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
      9:   'http://leanpub.com/juice-shop'
     10: ])
```

L	data/static/codefixes redirectChallenge_2.ts **(Line 9)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      6:   'http://shop.spreadshirt.com/juiceshop',
      7:   'http://shop.spreadshirt.de/juiceshop',
      8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
>>>   9:   'http://leanpub.com/juice-shop'
     10: ])
     11:
     12: export const isRedirectAllowed = (url: string) => {
```

L	data/static/codefixes redirectChallenge_3.ts **(Line 6)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      3:   'https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
      4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
      5:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
>>>   6:   'http://shop.spreadshirt.com/juiceshop',
      7:   'http://shop.spreadshirt.de/juiceshop',
      8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
      9:   'http://leanpub.com/juice-shop'
```

L	data/static/codefixes redirectChallenge_3.ts **(Line 7)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
       5:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
       6:   'http://shop.spreadshirt.com/juiceshop',
>>>    7:   'http://shop.spreadshirt.de/juiceshop',
       8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
       9:   'http://leanpub.com/juice-shop'
      10: ])
```

L	data/static/codefixes redirectChallenge_3.ts **(Line 9)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       6:   'http://shop.spreadshirt.com/juiceshop',
       7:   'http://shop.spreadshirt.de/juiceshop',
       8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
>>>    9:   'http://leanpub.com/juice-shop'
      10: ])
      11:
      12: export const isRedirectAllowed = (url: string) => {
```

L	data/static/codefixes redirectChallenge_4_correct.ts **(Line 6)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       3:   'https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
       4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
       5:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
>>>    6:   'http://shop.spreadshirt.com/juiceshop',
       7:   'http://shop.spreadshirt.de/juiceshop',
       8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
       9:   'http://leanpub.com/juice-shop'
```

L	data/static/codefixes redirectChallenge_4_correct.ts **(Line 7)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
       5:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
       6:   'http://shop.spreadshirt.com/juiceshop',
>>>    7:   'http://shop.spreadshirt.de/juiceshop',
       8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
       9:   'http://leanpub.com/juice-shop'
      10: ])
```

L	data/static/codefixes redirectChallenge_4_correct.ts **(Line 9)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       6:   'http://shop.spreadshirt.com/juiceshop',
       7:   'http://shop.spreadshirt.de/juiceshop',
       8:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
>>>    9:   'http://leanpub.com/juice-shop'
      10: ])
      11:
      12: export const isRedirectAllowed = (url: string) => {
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_1.ts **(Line 5)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       2:   'https://github.com/juice-shop/juice-shop',
       3:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
       4:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
>>>    5:   'http://shop.spreadshirt.com/juiceshop',
       6:   'http://shop.spreadshirt.de/juiceshop',
       7:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
       8:   'http://leanpub.com/juice-shop'
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_1.ts **(Line 6)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       3:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
       4:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
       5:   'http://shop.spreadshirt.com/juiceshop',
>>>    6:   'http://shop.spreadshirt.de/juiceshop',
       7:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
       8:   'http://leanpub.com/juice-shop'
       9: ])
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_1.ts **(Line 8)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       5:   'http://shop.spreadshirt.com/juiceshop',
       6:   'http://shop.spreadshirt.de/juiceshop',
       7:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
>>>    8:   'http://leanpub.com/juice-shop'
       9: ])
      10:
      11: export const isRedirectAllowed = (url: string) => {
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_2.ts **(Line 5)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      2:   'https://github.com/juice-shop/juice-shop',
      3:   'https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
      4:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
>>>   5:   'http://shop.spreadshirt.com/juiceshop',
      6:   'http://shop.spreadshirt.de/juiceshop',
      7:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
      8:   'http://leanpub.com/juice-shop'
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_2.ts **(Line 6)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      3:   'https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
      4:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6',
      5:   'http://shop.spreadshirt.com/juiceshop',
>>>   6:   'http://shop.spreadshirt.de/juiceshop',
      7:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
      8:   'http://leanpub.com/juice-shop'
      9: ])
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_2.ts **(Line 8)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      5:   'http://shop.spreadshirt.com/juiceshop',
      6:   'http://shop.spreadshirt.de/juiceshop',
      7:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
>>>   8:   'http://leanpub.com/juice-shop'
      9: ])
     10:
     11: export const isRedirectAllowed = (url: string) => {
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_3_correct.ts **(Line 3)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      1: export const redirectAllowlist = new Set([
      2:   'https://github.com/juice-shop/juice-shop',
>>>   3:   'http://shop.spreadshirt.com/juiceshop',
      4:   'http://shop.spreadshirt.de/juiceshop',
      5:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
      6:   'http://leanpub.com/juice-shop'
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_3_correct.ts **(Line 4)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       1: export const redirectAllowlist = new Set([
       2:   'https://github.com/juice-shop/juice-shop',
       3:   'http://shop.spreadshirt.com/juiceshop',
>>>    4:   'http://shop.spreadshirt.de/juiceshop',
       5:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
       6:   'http://leanpub.com/juice-shop'
       7: ])
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_3_correct.ts **(Line 6)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       3:   'http://shop.spreadshirt.com/juiceshop',
       4:   'http://shop.spreadshirt.de/juiceshop',
       5:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
>>>    6:   'http://leanpub.com/juice-shop'
       7: ])
       8:
       9: export const isRedirectAllowed = (url: string) => {
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_4.ts **(Line 5)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       2:   'https://github.com/juice-shop/juice-shop',
       3:   'https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
       4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
>>>    5:   'http://shop.spreadshirt.com/juiceshop',
       6:   'http://shop.spreadshirt.de/juiceshop',
       7:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
       8:   'http://leanpub.com/juice-shop'
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_4.ts **(Line 6)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
       3:   'https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm',
       4:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW',
       5:   'http://shop.spreadshirt.com/juiceshop',
>>>    6:   'http://shop.spreadshirt.de/juiceshop',
       7:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
       8:   'http://leanpub.com/juice-shop'
       9: ])
```

L	data/static/codefixes redirectCryptoCurrencyChallenge_4.ts **(Line 8)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      5:   'http://shop.spreadshirt.com/juiceshop',
      6:   'http://shop.spreadshirt.de/juiceshop',
      7:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
>>>   8:   'http://leanpub.com/juice-shop'
      9: ])
     10:
     11: export const isRedirectAllowed = (url: string) => {
```

L	frontend/src/app/order-completion order-completion.component.spec.ts **(Line 136)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    133:
    134:   it('should use configured URL as is if it is not a twitter URL', () => {
    135:     trackOrderService.find.and.returnValue(of({ data: [{ products: [] }] }))
>>> 136:     configurationService.getApplicationConfiguration.and.returnValue(of({ application:
{ social: { twitterUrl: 'http://localhorst:42' } } }))
    137:     component.ngOnInit()
    138:     expect(component.tweetText).toBe('I just purchased%0afrom http://localhorst:42')
    139:   })
```

L	lib/insecurity.ts **(Line 129)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    126:   'https://blockchain.info/address/1AbKfgvw9psQ41NbLi8kufDQTezwG8DRZm', //
vuln-code-snippet vuln-line redirectCryptoCurrencyChallenge
    127:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW', //
vuln-code-snippet vuln-line redirectCryptoCurrencyChallenge
    128:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6', //
vuln-code-snippet vuln-line redirectCryptoCurrencyChallenge
>>> 129:   'http://shop.spreadshirt.com/juiceshop',
    130:   'http://shop.spreadshirt.de/juiceshop',
    131:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
    132:   'http://leanpub.com/juice-shop'
```

L	lib/insecurity.ts **(Line 130)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    127:   'https://explorer.dash.org/address/Xr556RzuwX6hg5EGpkybbv5RanJoZN17kW', //
vuln-code-snippet vuln-line redirectCryptoCurrencyChallenge
    128:   'https://etherscan.io/address/0x0f933ab9fcaaa782d0279c300d73750e1311eae6', //
vuln-code-snippet vuln-line redirectCryptoCurrencyChallenge
    129:   'http://shop.spreadshirt.com/juiceshop',
>>> 130:   'http://shop.spreadshirt.de/juiceshop',
    131:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
    132:   'http://leanpub.com/juice-shop'
    133: ])
```

L	lib/insecurity.ts **(Line 132)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    129:   'http://shop.spreadshirt.com/juiceshop',
    130:   'http://shop.spreadshirt.de/juiceshop',
    131:   'https://www.stickeryou.com/products/owasp-juice-shop/794',
>>> 132:   'http://leanpub.com/juice-shop'
    133: ])
    134:
    135: export const isRedirectAllowed = (url: string) => {
```

L	test/cypress/e2e/profile.spec.ts **(Line 74)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
     71:   describe('challenge "csrf"', () => {
     72:     // FIXME Only works on Chrome <80 but Protractor uses latest Chrome version. Test
can probably never be turned on again.
     73:     xit('should be possible to perform a CSRF attack against the user profile page', ()
=> {
>>>  74:       cy.visit('http://htmledit.squarefree.com')
     75:       /* The script executed below is equivalent to pasting this string into
http://htmledit.squarefree.com: */
     76:       /* <form action="http://localhost:3000/profile" method="POST"><input
type="hidden" name="username" value="CSRF"/><input
type="submit"/></form><script>document.forms[0].submit();</script> */
     77:       let document: any
```

L	test/cypress/e2e/profile.spec.ts **(Line 107)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    104:          headers: {
    105:            'Content-type': 'application/x-www-form-urlencoded',
    106:            Authorization: `Bearer ${localStorage.getItem('token')}`,
>>> 107:            Origin: 'http://htmledit.squarefree.com', // FIXME Not allowed by browser
due to "unsafe header not permitted"
    108:            Cookie: `token=${localStorage.getItem('token')}` // FIXME Not allowed by
browser due to "unsafe header not permitted"
    109:          },
    110:          body: formData
```

L	test/server/redirectSpec.ts **(Line 45)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
     42:   })
     43:
     44:   it('should raise error for URL not on allowlist', () => {
>>>  45:     req.query.to = 'http://kimminich.de'
     46:
     47:     performRedirect()(req, res, next)
     48:
```

L	test/server/redirectSpec.ts **(Line 81)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
     78:   })
     79:
     80:   it('tricking the allowlist should solve "redirectChallenge"', () => {
>>>  81:     req.query.to = 'http://kimminich.de?to=https://github.com/juice-shop/juice-shop'
     82:     challenges.redirectChallenge = { solved: false, save } as unknown as Challenge
     83:
     84:     performRedirect()(req, res, next)
```

L	test/server/utilsSpec.ts **(Line 38)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
     35:     })
     36:
     37:     it('returns filename from http:// URL', () => {
>>>  38:       expect(utils.extractFilename('http://bla.blubb/test.exe')).to.equal('test.exe')
     39:     })
     40:
     41:     it('ignores query part of http:// URL', () => {
```

L	test/server/utilsSpec.ts **(Line 42)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
     39:     })
     40:
     41:     it('ignores query part of http:// URL', () => {
>>>  42:       expect(utils.extractFilename('http://bla.blubb/test.exe?bla=blubb&a=b')).to.equa
l('test.exe')
     43:     })
     44:
     45:     it('also works for file:// URLs', () => {
```

L	test/server/verifySpec.ts **(Line 88)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
     85:   describe('accessControlChallenges', () => {
     86:     it('"scoreBoardChallenge" is solved when the 1px.png transpixel is requested', ()
=> {
     87:       challenges.scoreBoardChallenge = { solved: false, save } as unknown as Challenge
>>>  88:       req.url = 'http://juice-sh.op/public/images/padding/1px.png'
     89:
     90:       verify.accessControlChallenges()(req, res, next)
     91:
```

L	test/server/verifySpec.ts **(Line 97)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
     94:
     95:     it('"adminSectionChallenge" is solved when the 19px.png transpixel is requested',
() => {
     96:       challenges.adminSectionChallenge = { solved: false, save } as unknown as
Challenge
>>>  97:       req.url = 'http://juice-sh.op/public/images/padding/19px.png'
     98:
     99:       verify.accessControlChallenges()(req, res, next)
    100:
```

L	test/server/verifySpec.ts **(Line 106)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    103:
    104:     it('"tokenSaleChallenge" is solved when the 56px.png transpixel is requested', ()
=> {
    105:       challenges.tokenSaleChallenge = { solved: false, save } as unknown as Challenge
>>> 106:       req.url = 'http://juice-sh.op/public/images/padding/56px.png'
    107:
    108:       verify.accessControlChallenges()(req, res, next)
    109:
```

L	test/server/verifySpec.ts **(Line 115)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    112:
    113:     it('"extraLanguageChallenge" is solved when the Klingon translation file is
requested', () => {
    114:       challenges.extraLanguageChallenge = { solved: false, save } as unknown as
Challenge
>>> 115:       req.url = 'http://juice-sh.op/public/i18n/tlh_AA.json'
    116:
    117:       verify.accessControlChallenges()(req, res, next)
    118:
```

L	test/server/verifySpec.ts **(Line 125)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    122:     it('"retrieveBlueprintChallenge" is solved when the blueprint file is requested', () => {
    123:       challenges.retrieveBlueprintChallenge = { solved: false, save } as unknown as Challenge
    124:       setRetrieveBlueprintChallengeFile('test.dxf')
>>> 125:       req.url = 'http://juice-sh.op/public/images/products/test.dxf'
    126:
    127:       verify.accessControlChallenges()(req, res, next)
    128:
```

L	test/server/verifySpec.ts **(Line 134)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    131:
    132:     it('"missingEncodingChallenge" is solved when the crazy cat photo is requested', () => {
    133:       challenges.missingEncodingChallenge = { solved: false, save } as unknown as Challenge
>>> 134:       req.url = 'http://juice-sh.op/public/images/uploads/%E1%93%9A%E1%98%8F%E1%97%A2-%23zatschi-%23whoneedsfourlegs-1572600969477.jpg'
    135:
    136:       verify.accessControlChallenges()(req, res, next)
    137:
```

L	test/server/verifySpec.ts **(Line 143)**	**typescript:S5332** Using http protocol is insecure. Use https instead. Category: Encrypt Data

■ **Security Hotspot Code:**

```
    140:
    141:     it('"accessLogDisclosureChallenge" is solved when any server access log file is requested', () => {
    142:       challenges.accessLogDisclosureChallenge = { solved: false, save } as unknown as Challenge
>>> 143:       req.url = 'http://juice-sh.op/support/logs/access.log.2019-01-15'
    144:
    145:       verify.accessControlChallenges()(req, res, next)
    146:
```

L	test/smoke/Dockerfile **(Line 7)**	**docker:S5332** Make sure that using clear-text protocols is safe here. Category: Encrypt Data

■ **Security Hotspot Code:**

```
      4:
      5: COPY smoke-test.sh smoke-test.sh
      6:
>>>   7: CMD ["sh", "smoke-test.sh", "http://app:3000"]
```

Insecure Configuration (2 hotspots)

Risk	File Path	Rule & Message
L	server.ts **(Line 182)**	**typescript:S5122** Make sure that enabling CORS is safe here. Category: Insecure Configuration

■ **Security Hotspot Code:**

```
    179:
    180:   /* Bludgeon solution for possible CORS problems: Allow everything! */
    181:   app.options('*', cors())
>>> 182:   app.use(cors())
    183:
    184:   /* Security middleware */
    185:   app.use(helmet.noSniff())
```

Risk	File Path	Rule & Message
L	server.ts **(Line 671)**	**typescript:S4507** Make sure this debug feature is deactivated before delivering the code in production. Category: Insecure Configuration

■ **Security Hotspot Code:**

```
    668:
    669:   /* Error Handling */
    670:   app.use(verify.errorHandlingChallenge())
>>> 671:   app.use(errorhandler())
    672: }).catch((err) => {
    673:   console.error(err)
    674: })
```

Others (14 hotspots)

Risk	File Path	Rule & Message
L	lib/insecurity.ts **(Line 43)**	**typescript:S4790** Make sure this weak hash algorithm is not used in a sensitive context here. Category: Others

■ **Security Hotspot Code:**

```
     40:   updateFrom: (req: Request, user: ResponseWithUser) => any
     41: }
     42:
>>>  43: export const hash = (data: string) =>
crypto.createHash('md5').update(data).digest('hex')
     44: export const hmac = (data: string) => crypto.createHmac('sha256',
'pa4qacea4VK9t9nGv7yZtwmj').update(data).digest('hex')
     45:
     46: export const cutOffPoisonNullByte = (str: string) => {
```

L	frontend/src/index.html **(Line 15)**	**Web:S5725** Make sure not using resource integrity feature is safe here. Category: Others

■ **Security Hotspot Code:**

```
     12:   <meta name="viewport" content="width=device-width, initial-scale=1">
     13:   <link id="favicon" rel="icon" type="image/x-icon"
href="assets/public/favicon_js.ico">
     14:   <link rel="stylesheet" type="text/css"
href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.css" />
>>>  15:   <script
src="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.js"></script>
     16:   <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2.4/jquery.min.js"></script>
     17:   <script>
     18:     window.addEventListener("load", function(){
```

L	frontend/src/index.html **(Line 16)**	**Web:S5725** Make sure not using resource integrity feature is safe here. Category: Others

■ **Security Hotspot Code:**

```
     13:   <link id="favicon" rel="icon" type="image/x-icon"
href="assets/public/favicon_js.ico">
     14:   <link rel="stylesheet" type="text/css"
href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.css" />
     15:   <script
src="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.js"></script>
>>>  16:   <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2.4/jquery.min.js"></script>
     17:   <script>
     18:     window.addEventListener("load", function(){
     19:       window.cookieconsent.initialise({
```

L	test/api/loginApiSpec.ts **(Line 253)**	**typescript:S1313** Make sure using a hardcoded IP address 1.2.3.4 is safe here. Category: Others

■ **Security Hotspot Code:**

```
    250:         return frisby.get(REST_URL + '/saveLoginIp', {
    251:           headers: {
    252:             Authorization: 'Bearer ' + jsonLogin.authentication.token,
>>> 253:             'true-client-ip': '1.2.3.4'
    254:           }
    255:         })
    256:           .expect('status', 200)
```

L	test/api/loginApiSpec.ts **(Line 257)**	**typescript:S1313** Make sure using a hardcoded IP address 1.2.3.4 is safe here. Category: Others

■ **Security Hotspot Code:**

```
    254:           }
    255:         })
    256:           .expect('status', 200)
>>> 257:           .expect('json', { lastLoginIp: '1.2.3.4' })
    258:       })
    259:   })
    260:
```

L	test/server/utilsSpec.ts **(Line 16)**	**typescript:S1313** Make sure using a hardcoded IP address 2001:0db8:85a3:0000:0000:8a2e:0370:7334 is safe here. Category: Others

■ **Security Hotspot Code:**

```
     13: describe('utils', () => {
     14:   describe('toSimpleIpAddress', () => {
     15:     it('returns ipv6 address unchanged', () => {
>>>  16:       expect(utils.toSimpleIpAddress('2001:0db8:85a3:0000:0000:8a2e:0370:7334')).to.eq
ual('2001:0db8:85a3:0000:0000:8a2e:0370:7334')
     17:     })
     18:
     19:     it('returns ipv4 address fully specified as ipv6 unchanged', () => {
```

L	test/server/utilsSpec.ts **(Line 16)**	**typescript:S1313** Make sure using a hardcoded IP address 2001:0db8:85a3:0000:0000:8a2e:0370:7334 is safe here. Category: Others

■ **Security Hotspot Code:**

```
     13: describe('utils', () => {
     14:   describe('toSimpleIpAddress', () => {
     15:     it('returns ipv6 address unchanged', () => {
>>>  16:       expect(utils.toSimpleIpAddress('2001:0db8:85a3:0000:0000:8a2e:0370:7334')).to.eq
ual('2001:0db8:85a3:0000:0000:8a2e:0370:7334')
     17:     })
     18:
     19:     it('returns ipv4 address fully specified as ipv6 unchanged', () => {
```

L	test/server/utilsSpec.ts **(Line 20)**	**typescript:S1313** Make sure using a hardcoded IP address 0:0:0:0:0:ffff:7f00:1 is safe here. Category: Others

■ **Security Hotspot Code:**

```
     17:     })
     18:
     19:     it('returns ipv4 address fully specified as ipv6 unchanged', () => {
>>>  20:       expect(utils.toSimpleIpAddress('0:0:0:0:0:ffff:7f00:1')).to.equal('0:0:0:0:0:ffff:7f00:1')
     21:     })
     22:
     23:     it('returns ipv6 loopback address as ipv4 address', () => {
```

L	test/server/utilsSpec.ts **(Line 20)**	**typescript:S1313** Make sure using a hardcoded IP address 0:0:0:0:0:ffff:7f00:1 is safe here. Category: Others

■ **Security Hotspot Code:**

```
     17:     })
     18:
     19:     it('returns ipv4 address fully specified as ipv6 unchanged', () => {
>>>  20:       expect(utils.toSimpleIpAddress('0:0:0:0:0:ffff:7f00:1')).to.equal('0:0:0:0:0:ffff:7f00:1')
     21:     })
     22:
     23:     it('returns ipv6 loopback address as ipv4 address', () => {
```

L	test/server/utilsSpec.ts **(Line 28)**	**typescript:S1313** Make sure using a hardcoded IP address ::ffff:192.0.2.128 is safe here. Category: Others

■ **Security Hotspot Code:**

```
     25:     })
     26:
     27:     it('returns ipv4-mapped address as ipv4 address', () => {
>>>  28:       expect(utils.toSimpleIpAddress('::ffff:192.0.2.128')).to.equal('192.0.2.128')
     29:     })
     30:   })
     31:
```

L	Dockerfile **(Line 4)**	**docker:S6505** Omitting "--ignore-scripts" can lead to the execution of shell scripts. Make sure it is safe here. Category: Others

■ **Security Hotspot Code:**

```
      1: FROM node:22 AS installer
      2: COPY . /juice-shop
      3: WORKDIR /juice-shop
>>>   4: RUN npm i -g typescript ts-node
      5: RUN npm install --omit=dev --unsafe-perm
      6: RUN npm dedupe --omit=dev
      7: RUN rm -rf frontend/node_modules
```

L	Dockerfile **(Line 5)**	**docker:S6505** Omitting "--ignore-scripts" can lead to the execution of shell scripts. Make sure it is safe here. Category: Others

■ **Security Hotspot Code:**

```
      2: COPY . /juice-shop
      3: WORKDIR /juice-shop
      4: RUN npm i -g typescript ts-node
>>>   5: RUN npm install --omit=dev --unsafe-perm
      6: RUN npm dedupe --omit=dev
      7: RUN rm -rf frontend/node_modules
      8: RUN rm -rf frontend/.angular
```

L	Dockerfile **(Line 19)**	**docker:S6505** Omitting "--ignore-scripts" can lead to the execution of shell scripts. Make sure it is safe here. Category: Others

■ **Security Hotspot Code:**

```
     16: RUN rm i18n/*.json || true
     17:
     18: ARG CYCLONEDX_NPM_VERSION=latest
>>>  19: RUN npm install -g @cyclonedx/cyclonedx-npm@$CYCLONEDX_NPM_VERSION
     20: RUN npm run sbom
     21:
     22: FROM gcr.io/distroless/nodejs22-debian12
```

L	Gruntfile.js **(Line 77)**	**javascript:S4790** Make sure this weak hash algorithm is not used in a sensitive context here. Category: Others

■ **Security Hotspot Code:**

```
     74:     const crypto = require('node:crypto')
     75:     fs.readdirSync('dist/').forEach(file => {
     76:       const buffer = fs.readFileSync('dist/' + file)
>>>  77:       const md5 = crypto.createHash('md5')
     78:       md5.update(buffer)
     79:       const md5Hash = md5.digest('hex')
     80:       const md5FileName = 'dist/' + file + '.md5'
```

Permission (2 hotspots)

Risk	File Path	Rule & Message
M	Dockerfile **(Line 2)**	**docker:S6470** Copying recursively might inadvertently add sensitive data to the container. Make sure it is safe here. Category: Permission

■ **Security Hotspot Code:**

```
        1: FROM node:22 AS installer
>>>     2: COPY . /juice-shop
        3: WORKDIR /juice-shop
        4: RUN npm i -g typescript ts-node
        5: RUN npm install --omit=dev --unsafe-perm
```

M	test/smoke/Dockerfile **(Line 1)**	**docker:S6471** The "alpine" image runs with "root" as the default user. Make sure it is safe here. Category: Permission

■ **Security Hotspot Code:**

```
>>>     1: FROM alpine
        2:
        3: RUN apk add curl
        4:
```

Remote Code Execution (3 hotspots)

Risk	File Path	Rule & Message
M	routes/captcha.ts **(Line 23)**	**typescript:S1523** Make sure that this dynamic injection or execution of code is safe. Category: Remote Code Execution

■ **Security Hotspot Code:**

```
     20:     const secondOperator = operators[Math.floor((Math.random() * 3))]
     21:
     22:     const expression = firstTerm.toString() + firstOperator + secondTerm.toString() +
secondOperator + thirdTerm.toString()
>>>  23:     const answer = eval(expression).toString() // eslint-disable-line no-eval
     24:
     25:     const captcha = {
     26:       captchaId,
```

M	routes/userProfile.ts **(Line 62)**	**typescript:S1523** Make sure that this dynamic injection or execution of code is safe. Category: Remote Code Execution

■ **Security Hotspot Code:**

```
     59:         if (!code) {
     60:           throw new Error('Username is null')
     61:         }
>>>  62:         username = eval(code) // eslint-disable-line no-eval
     63:       } catch (err) {
     64:         username = '\\' + username
     65:       }
```

M	test/cypress/e2e/contact.spec.ts **(Line 258)**	**typescript:S1523** Make sure that this dynamic injection or execution of code is safe. Category: Remote Code Execution

■ **Security Hotspot Code:**

```
    255:     .then((val) => {
    256:       cy.get('#captchaControl').clear()
    257:       // eslint-disable-next-line no-eval
>>> 258:       const answer = eval(val).toString()
    259:       cy.get('#captchaControl').type(answer)
    260:     })
    261: }
```

Weak Cryptography (11 hotspots)

Risk	File Path	Rule & Message
M	data/datacreator.ts **(Line 247)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
    244:     let text = ''
    245:     const possible = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789'
    246:
>>> 247:     for (let i = 0; i < length; i++) { text += possible.charAt(Math.floor(Math.random()
* possible.length)) }
    248:
    249:     return text
    250:   }
```

Risk	File Path	Rule & Message
M	data/datacreator.ts **(Line 265)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
    262:     config.get<ProductConfig[]>('products').map(async (product, index) => {
    263:       return await QuantityModel.create({
    264:         ProductId: index + 1,
>>> 265:         quantity: product.quantity ?? Math.floor(Math.random() * 70 + 30),
    266:         limitPerUser: product.limitPerUser ?? null
    267:       }).catch((err: unknown) => {
    268:         logger.error(`Could not create quantity: ${utils.getErrorMessage(err)}`)
```

Risk	File Path	Rule & Message
M	data/datacreator.ts **(Line 323)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
    320:
    321: async function createProducts () {
    322:   const products =
structuredClone(config.get<ProductConfig[]>('products')).map((product) => {
>>> 323:     product.price = product.price ?? Math.floor(Math.random() * 9 + 1)
    324:     product.deluxePrice = product.deluxePrice ?? product.price
    325:     product.description = product.description || 'Lorem ipsum dolor sit amet,
consectetuer adipiscing elit.'
    326:
```

M	data/datacreator.ts **(Line 698)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
    695:       totalPrice: basket1Products[0].total + basket1Products[1].total,
    696:       bonus: basket1Products[0].bonus + basket1Products[1].bonus,
    697:       products: basket1Products,
>>> 698:       eta: Math.floor((Math.random() * 5) + 1).toString(),
    699:       delivered: false
    700:     },
    701:     {
```

M	frontend/src/app/code-snippet code-snippet.component.ts **(Line 167)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
    164:
    165:   shuffle () {
    166:     this.randomFixes = this.fixes
>>> 167:       .map((fix, index) => ({ fix, index, sort: Math.random() }))
    168:       .sort((a, b) => a.sort - b.sort)
    169:       .map(({ fix, index }) => ({ fix, index }))
    170:   }
```

M	lib/insecurity.ts **(Line 55)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
     52: }
     53:
     54: export const isAuthorized = () => expressJwt(({ secret: publicKey }) as any)
>>>  55: export const denyAll = () => expressJwt({ secret: '' + Math.random() } as any)
     56: export const authorize = (user = {}) => jwt.sign(user, privateKey, { expiresIn: '6h',
algorithm: 'RS256' })
     57: export const verify = (token: string) => token ? (jws.verify as ((token: string,
secret: string) => boolean))(token, publicKey) : false
     58: export const decode = (token: string) => { return jws.decode(token)?.payload }
```

M	routes/captcha.ts **(Line 15)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
     12:     const captchaId = req.app.locals.captchaId++
     13:     const operators = ['*', '+', '-']
     14:
>>>  15:     const firstTerm = Math.floor((Math.random() * 10) + 1)
     16:     const secondTerm = Math.floor((Math.random() * 10) + 1)
     17:     const thirdTerm = Math.floor((Math.random() * 10) + 1)
     18:
```

M	routes/captcha.ts **(Line 16)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
     13:     const operators = ['*', '+', '-']
     14:
     15:     const firstTerm = Math.floor((Math.random() * 10) + 1)
>>>  16:     const secondTerm = Math.floor((Math.random() * 10) + 1)
     17:     const thirdTerm = Math.floor((Math.random() * 10) + 1)
     18:
     19:     const firstOperator = operators[Math.floor((Math.random() * 3))]
```

M	routes/captcha.ts **(Line 17)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
     14:
     15:     const firstTerm = Math.floor((Math.random() * 10) + 1)
     16:     const secondTerm = Math.floor((Math.random() * 10) + 1)
>>>  17:     const thirdTerm = Math.floor((Math.random() * 10) + 1)
     18:
     19:     const firstOperator = operators[Math.floor((Math.random() * 3))]
     20:     const secondOperator = operators[Math.floor((Math.random() * 3))]
```

M	routes/captcha.ts **(Line 19)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
    16:     const secondTerm = Math.floor((Math.random() * 10) + 1)
    17:     const thirdTerm = Math.floor((Math.random() * 10) + 1)
    18:
>>> 19:     const firstOperator = operators[Math.floor((Math.random() * 3))]
    20:     const secondOperator = operators[Math.floor((Math.random() * 3))]
    21:
    22:     const expression = firstTerm.toString() + firstOperator + secondTerm.toString() + secondOperator + thirdTerm.toString()
```

M	routes/captcha.ts **(Line 20)**	**typescript:S2245** Make sure that using this pseudorandom number generator is safe here. Category: Weak Cryptography

■ **Security Hotspot Code:**

```
    17:     const thirdTerm = Math.floor((Math.random() * 10) + 1)
    18:
    19:     const firstOperator = operators[Math.floor((Math.random() * 3))]
>>> 20:     const secondOperator = operators[Math.floor((Math.random() * 3))]
    21:
    22:     const expression = firstTerm.toString() + firstOperator + secondTerm.toString() + secondOperator + thirdTerm.toString()
    23:     const answer = eval(expression).toString() // eslint-disable-line no-eval
```

Cross-Site Scripting (XSS) (9 hotspots)

Risk	File Path	Rule & Message
H	frontend/src/app/about about.component.ts **(Line 121)**	**typescript:S6268** Make sure disabling Angular built-in sanitization is safe here. Category: Cross-Site Scripting (XSS)

■ **Security Hotspot Code:**

```
   118:           feedbacks[i].comment = `<span style="width: 90%; display:block;">${
   119:             feedbacks[i].comment
   120:           }<br/> (${this.stars[feedbacks[i].rating]})</span>`
>>> 121:           feedbacks[i].comment = this.sanitizer.bypassSecurityTrustHtml(
   122:             feedbacks[i].comment
   123:           )
   124:
```

Risk	File Path	Rule & Message
H	frontend/src/app/administration administration.component.ts **(Line 57)**	**typescript:S6268** Make sure disabling Angular built-in sanitization is safe here. Category: Cross-Site Scripting (XSS)

■ **Security Hotspot Code:**

```
     54:         this.userDataSource = users
     55:         this.userDataSourceHidden = users
     56:         for (const user of this.userDataSource) {
>>>  57:           user.email = this.sanitizer.bypassSecurityTrustHtml(`<span
class="${this.doesUserHaveAnActiveSession(user) ? 'confirmation' :
'error'}">${user.email}</span>`)
     58:         }
     59:         this.userDataSource = new MatTableDataSource(this.userDataSource)
     60:         this.userDataSource.paginator = this.paginatorUsers
```

H	frontend/src/app/administration administration.component.ts **(Line 75)**	**typescript:S6268** Make sure disabling Angular built-in sanitization is safe here. Category: Cross-Site Scripting (XSS)

■ **Security Hotspot Code:**

```
     72:       next: (feedbacks) => {
     73:         this.feedbackDataSource = feedbacks
     74:         for (const feedback of this.feedbackDataSource) {
>>>  75:           feedback.comment = this.sanitizer.bypassSecurityTrustHtml(feedback.comment)
     76:         }
     77:         this.feedbackDataSource = new MatTableDataSource(this.feedbackDataSource)
     78:         this.feedbackDataSource.paginator = this.paginatorFeedb
```

H	frontend/src/app/data-export data-export.component.ts **(Line 55)**	**typescript:S6268** Make sure disabling Angular built-in sanitization is safe here. Category: Cross-Site Scripting (XSS)

■ **Security Hotspot Code:**

```
     52:
     53:   getNewCaptcha () {
     54:     this.imageCaptchaService.getCaptcha().subscribe((data: any) => {
>>>  55:       this.captcha = this.sanitizer.bypassSecurityTrustHtml(data.image)
     56:     })
     57:   }
     58:
```

H	frontend/src/app/last-login-ip last-login-ip.component.ts **(Line 38)**	**typescript:S6268** Make sure disabling Angular built-in sanitization is safe here. Category: Cross-Site Scripting (XSS)

■ **Security Hotspot Code:**

```
     35:       payload = jwtDecode(token)
     36:       if (payload.data.lastLoginIp) {
     37:         // eslint-disable-next-line @typescript-eslint/restrict-template-expressions
>>>  38:         this.lastLoginIp =
this.sanitizer.bypassSecurityTrustHtml(`<small>${payload.data.lastLoginIp}</small>`)
     39:       }
     40:     }
     41:   }
```

H	frontend/src/app/score-board score-board.component.ts **(Line 86)**	**typescript:S6268** Make sure disabling Angular built-in sanitization is safe here. Category: Cross-Site Scripting (XSS)

■ Security Hotspot Code:

```
     83:           hintsAvailable: hints.filter((hint) => hint.ChallengeId === challenge.id).length,
     84:           tagList: challenge.tags ? challenge.tags.split(',').map((tag) => tag.trim()) : [],
     85:           originalDescription: challenge.description as string,
>>>  86:           description: this.sanitizer.bypassSecurityTrustHtml(challenge.description as string)
     87:         }
     88:       })
     89:
```

H	frontend/src/app/search-result search-result.component.ts **(Line 135)**	**typescript:S6268** Make sure disabling Angular built-in sanitization is safe here. Category: Cross-Site Scripting (XSS)

■ Security Hotspot Code:

```
    132:
    133:   trustProductDescription (tableData: any[]) { // vuln-code-snippet neutral-line restfulXssChallenge
    134:     for (let i = 0; i < tableData.length; i++) { // vuln-code-snippet neutral-line restfulXssChallenge
>>> 135:       tableData[i].description = this.sanitizer.bypassSecurityTrustHtml(tableData[i].description) // vuln-code-snippet vuln-line restfulXssChallenge
    136:     } // vuln-code-snippet neutral-line restfulXssChallenge
    137:   } // vuln-code-snippet neutral-line restfulXssChallenge
    138:   // vuln-code-snippet end restfulXssChallenge
```

H	frontend/src/app/search-result search-result.component.ts **(Line 161)**	**typescript:S6268** Make sure disabling Angular built-in sanitization is safe here. Category: Cross-Site Scripting (XSS)

■ Security Hotspot Code:

```
    158:         this.io.socket().emit('verifyLocalXssChallenge', queryParam)
    159:       }) // vuln-code-snippet hide-end
    160:       this.dataSource.filter = queryParam.toLowerCase()
>>> 161:       this.searchValue = this.sanitizer.bypassSecurityTrustHtml(queryParam) // vuln-code-snippet vuln-line localXssChallenge xssBonusChallenge
    162:       this.gridDataSource.subscribe((result: any) => {
    163:         if (result.length === 0) {
    164:           this.emptyState = true
```

H	frontend/src/app/track-result track-result.component.ts **(Line 45)**	**typescript:S6268** Make sure disabling Angular built-in sanitization is safe here. Category: Cross-Site Scripting (XSS)

■ Security Hotspot Code:

```
     42:     this.orderId = this.route.snapshot.queryParams.id
     43:     this.trackOrderService.find(this.orderId).subscribe((results) => {
     44:       // eslint-disable-next-line @typescript-eslint/restrict-template-expressions
>>>  45:       this.results.orderNo =
this.sanitizer.bypassSecurityTrustHtml(`<code>${results.data[0].orderId}</code>`)
     46:       this.results.email = results.data[0].email
     47:       this.results.totalPrice = results.data[0].totalPrice
     48:       this.results.products = results.data[0].products
```