Defeat Fingerprint Sensor

There is definitely something fishy going on in Kringle Castle. It might be time to explore Santa's office for any information on the culprit.

Objective

> Bypass the Santavator fingerprint sensor. Enter Santa's office without Santa's fingerprint.

`Difficulty: 3/5`

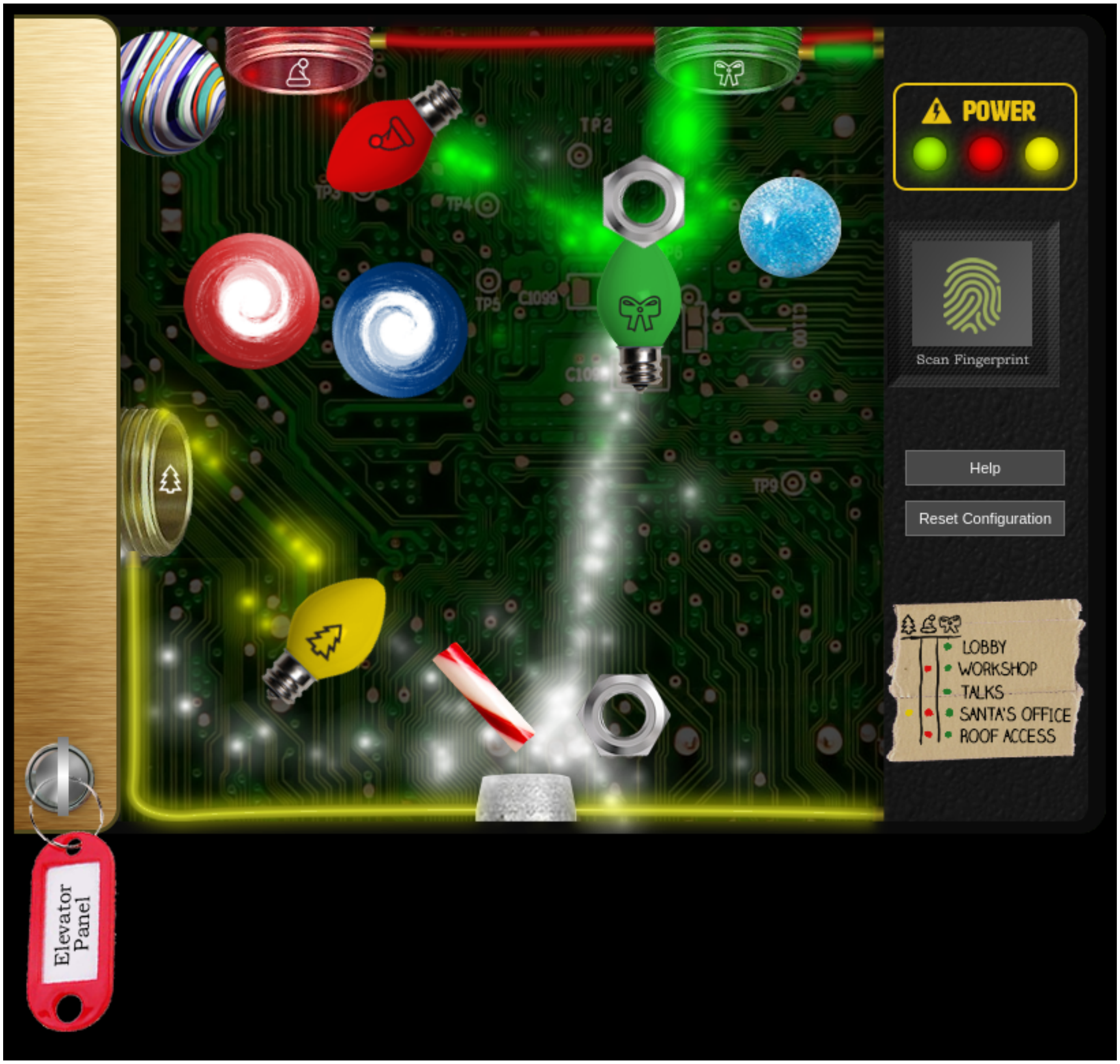
Solution

Looking at the code that runs the elevator, we see that `btn4` (the button for **Santa's Office**) has a different function that handles click() events:

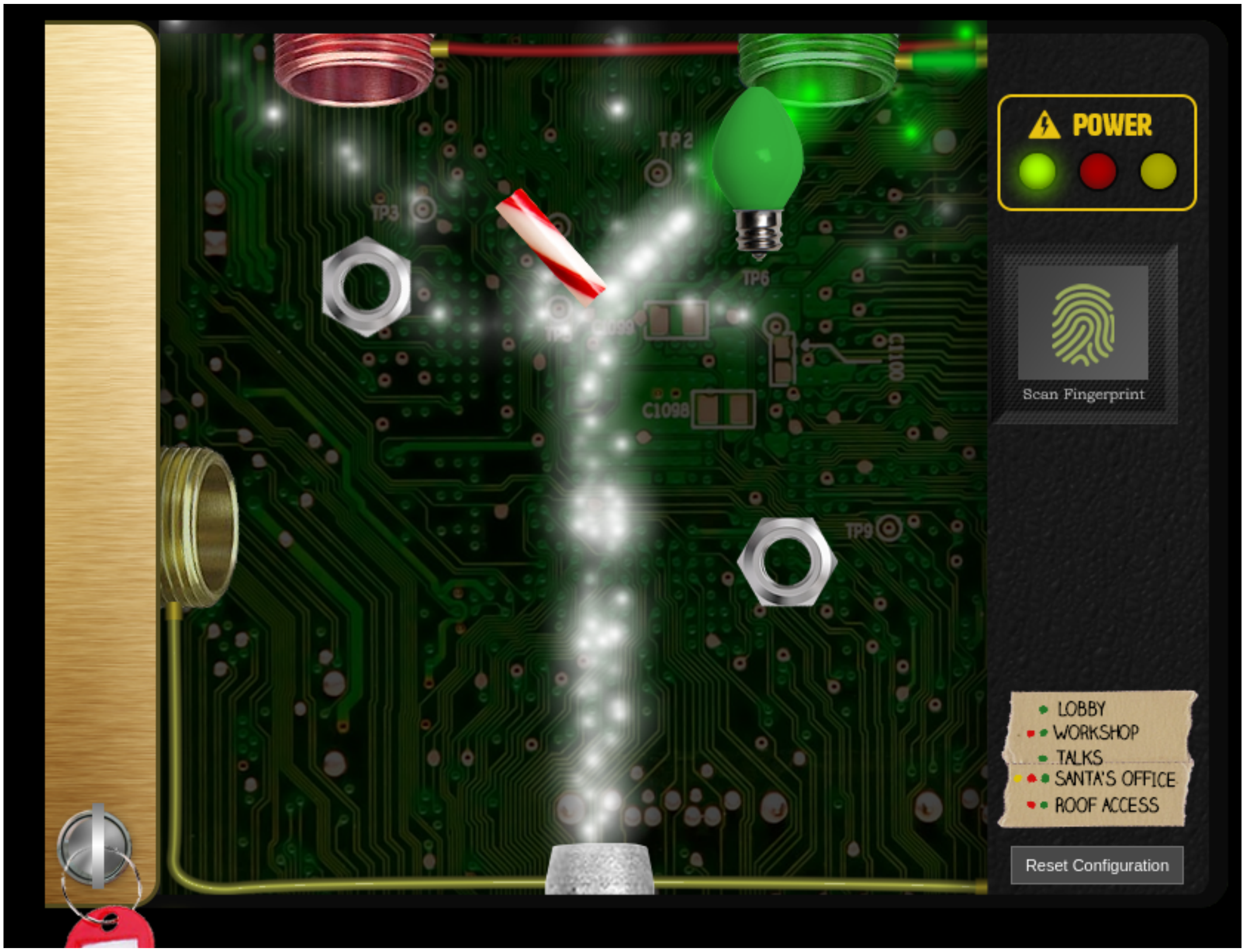
```
const handleBtn4 = () => {
  const cover = document.querySelector('.print-cover');
  cover.classList.add('open');

  cover.addEventListener('click', () => {
    if (btn4.classList.contains('powered') &&
hasToken('besanta')) {
      $.ajax({
        type: 'POST',
        url: POST_URL,
        dataType: 'json',
        contentType: 'application/json',
        data: JSON.stringify({
          targetFloor: '3',
          id: getParams.id,
        }),
        success: (res, status) => {
          if (res.hash) {
            __POST_RESULTS__({
              resourceId: getParams.id || '1111',
              hash: res.hash,
              action: 'goToFloor-3',
            });
          }
        }
      });
    } else {
      __SEND_MSG__({
        type: 'sfx',
        filename: 'error.mp3',
      });
    }
  });
};
```

Of particular note are the checks on line 5: a check to see that the button has a class `powered`, and that the user has a token `besanta`. Solving the `hasToken('besanta')` check is simple: the function `hasToken` checks for the existance of an item in the `tokens` list. In the JavaScript console, we can add `besanta` to `tokens` with `tokens.push('besanta')`.

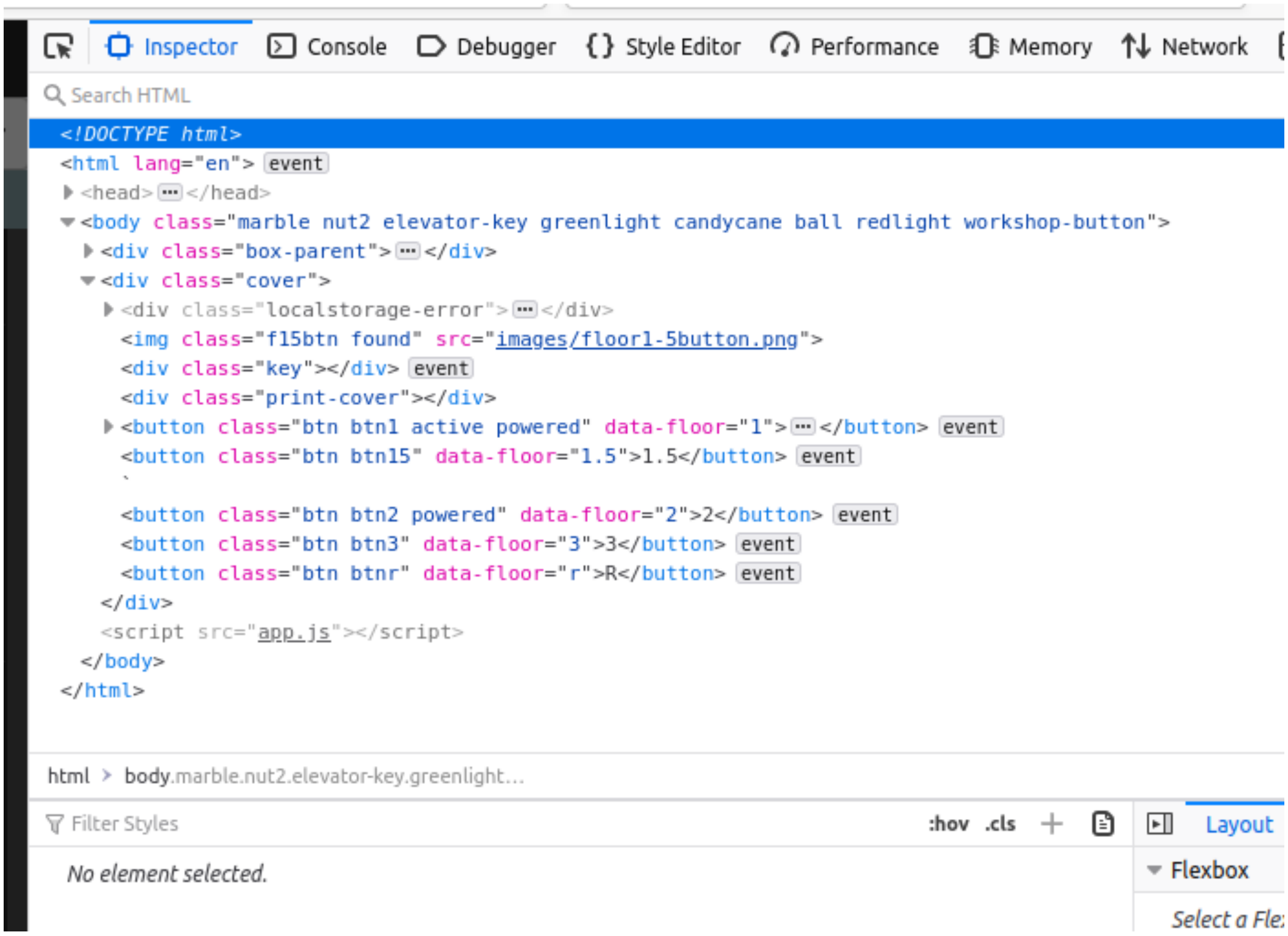
Solving the `powered` is a bit trickier. The `powered` class is added to the button by the function `renderTraps()`, called inside a continually-updating event loop for drawing the Sparkle Stream on the screen. Manually adding `powered` as a class to the button, or modifying the `powered[]` object in the JavaScript console results in the `powered` state being removed. One can build a rather convoluted method to split and color the Sparkle Stream:



But there is a simpler solution: power a single receiver, such as the green one:



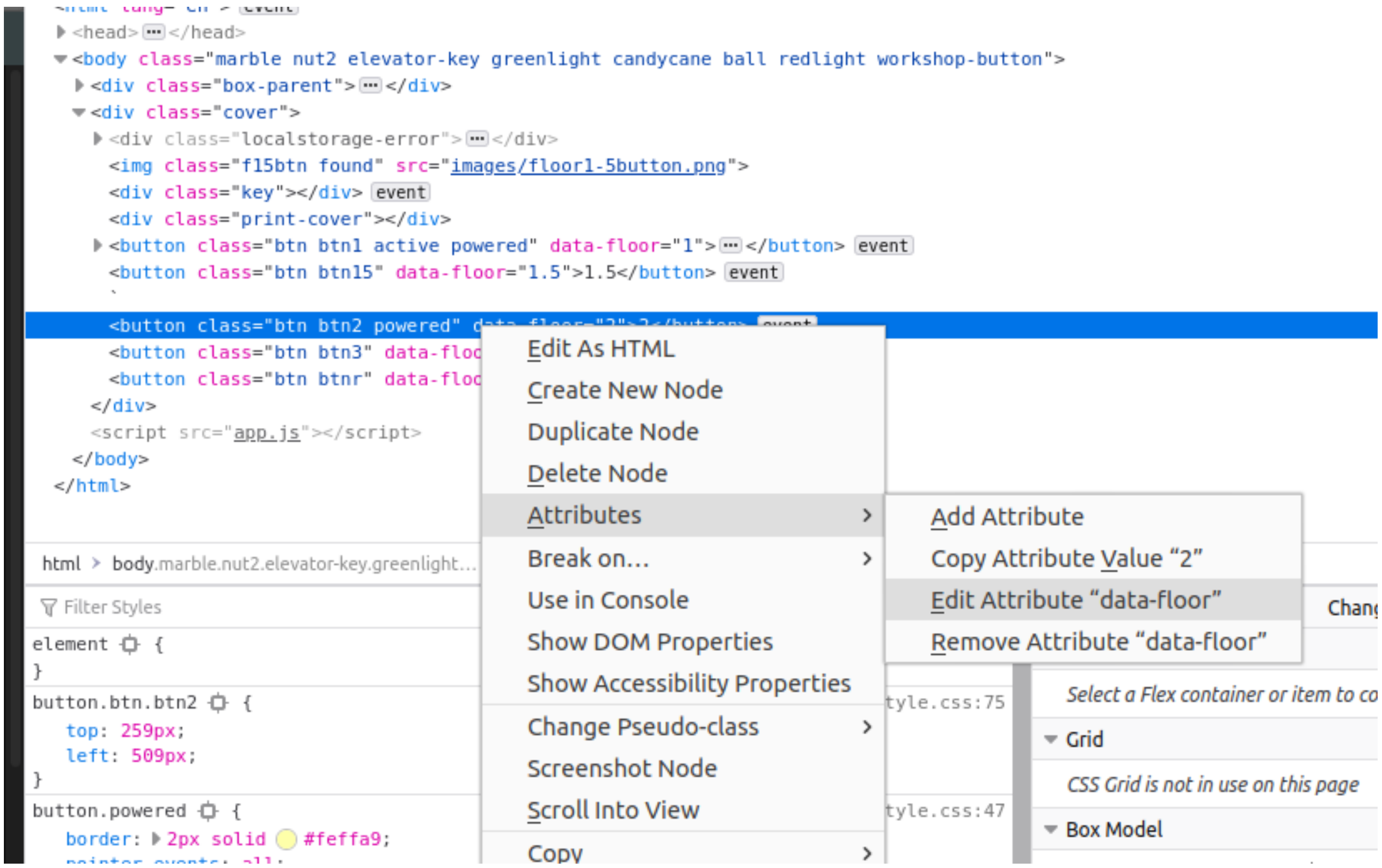
Then change what floor the button sends us to when it is clicked. Open the elevator panel, make sure the green receiver is powered, then open the Developer tools. In the Inspector tab, find the one of the buttons that has the `powered` class:



```
<!DOCTYPE html>
<html lang="en"> event
  <head>…</head>
  <body class="marble nut2 elevator-key greenlight candycane ball redlight workshop-button">
    <div class="box-parent">…</div>
    <div class="cover">
      <div class="localstorage-error">…</div>
      <img class="f15btn found" src="images/floor1-5button.png">
      <div class="key"></div> event
      <div class="print-cover"></div>
      <button class="btn btn1 active powered" data-floor="1">…</button> event
      <button class="btn btn15" data-floor="1.5">1.5</button> event
      `
      <button class="btn btn2 powered" data-floor="2">2</button> event
      <button class="btn btn3" data-floor="3">3</button> event
      <button class="btn btnr" data-floor="r">R</button> event
    </div>
    <script src="app.js"></script>
  </body>
</html>
```

html > body.marble.nut2.elevator-key.greenlight…

Then, edit the `data_floor` attribute to be `3` (the floor number of Santa's Office):


```
 ▸<head>…</head>
 ▾<body class="marble nut2 elevator-key greenlight candycane ball redlight workshop-button">
   ▸<div class="box-parent">…</div>
   ▾<div class="cover">
     ▸<div class="localstorage-error">…</div>
      <img class="f15btn found" src="images/floor1-5button.png">
      <div class="key"></div> event
      <div class="print-cover"></div>
     ▸<button class="btn btn1 active powered" data-floor="1">…</button> event
      <button class="btn btn15" data-floor="1.5">1.5</button> event
      <button class="btn btn2 powered" d
      <button class="btn btn3" data-floo
      <button class="btn btnr" data-floo
    </div>
    <script src="app.js"></script>
  </body>
</html>
```

```
element {
}
button.btn.btn2 {
    top: 259px;
    left: 509px;
}
button.powered {
    border: ▸2px solid #feffa9;
```

Click the modified button, and you'll be taken to Santa's Office.

Answer

Visit Santa's Office.