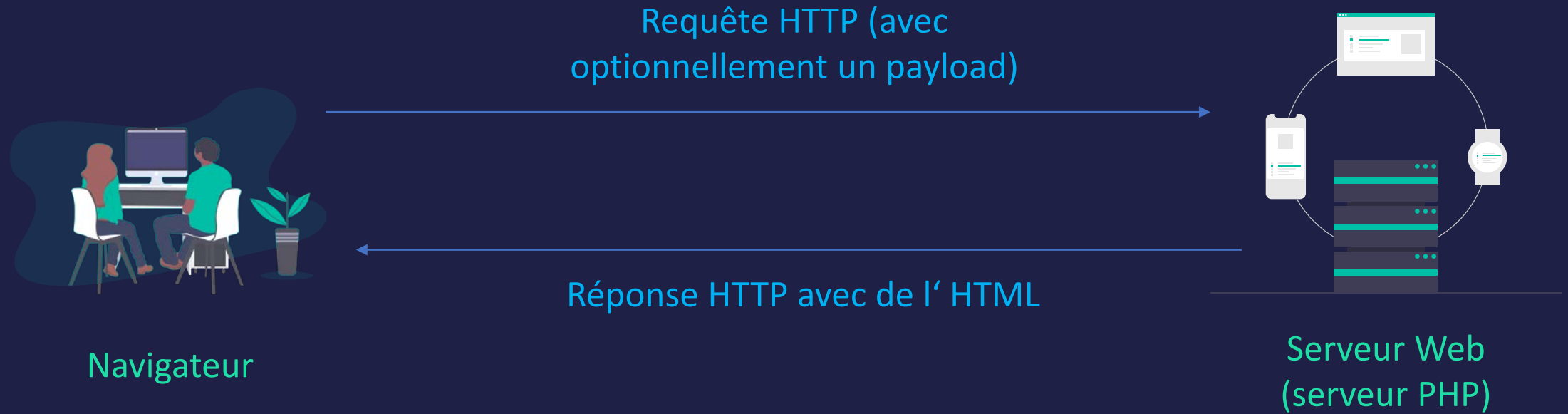


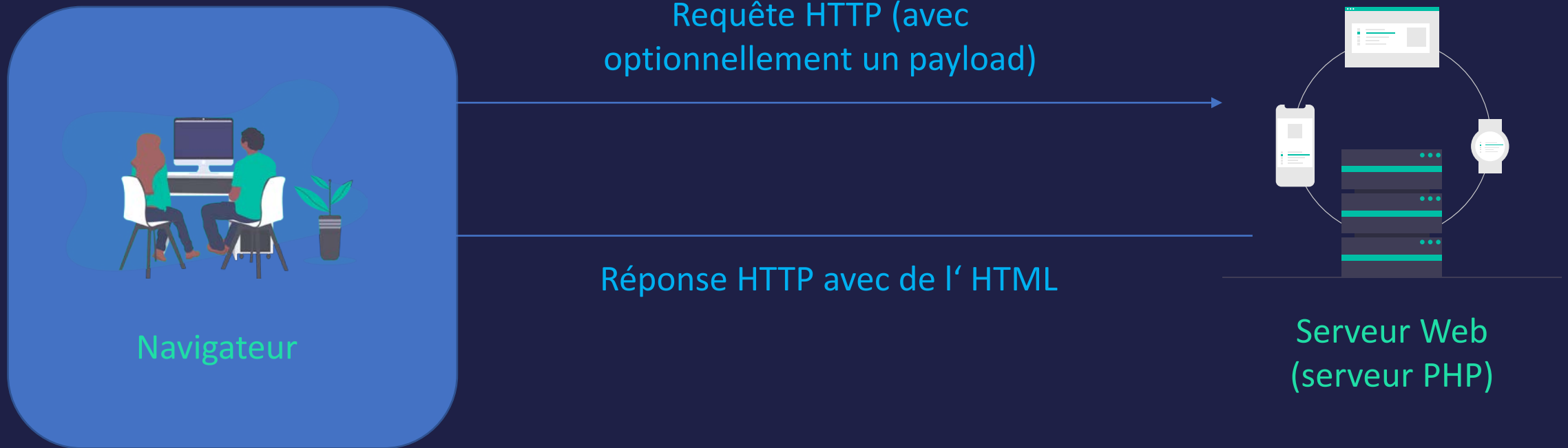
1. Les attaques MITM
2. Le phishing
3. Les attaques CSRF
4. Les modifications du DOM
5. Les injections
6. Le brute force

MODIFICATION DOM

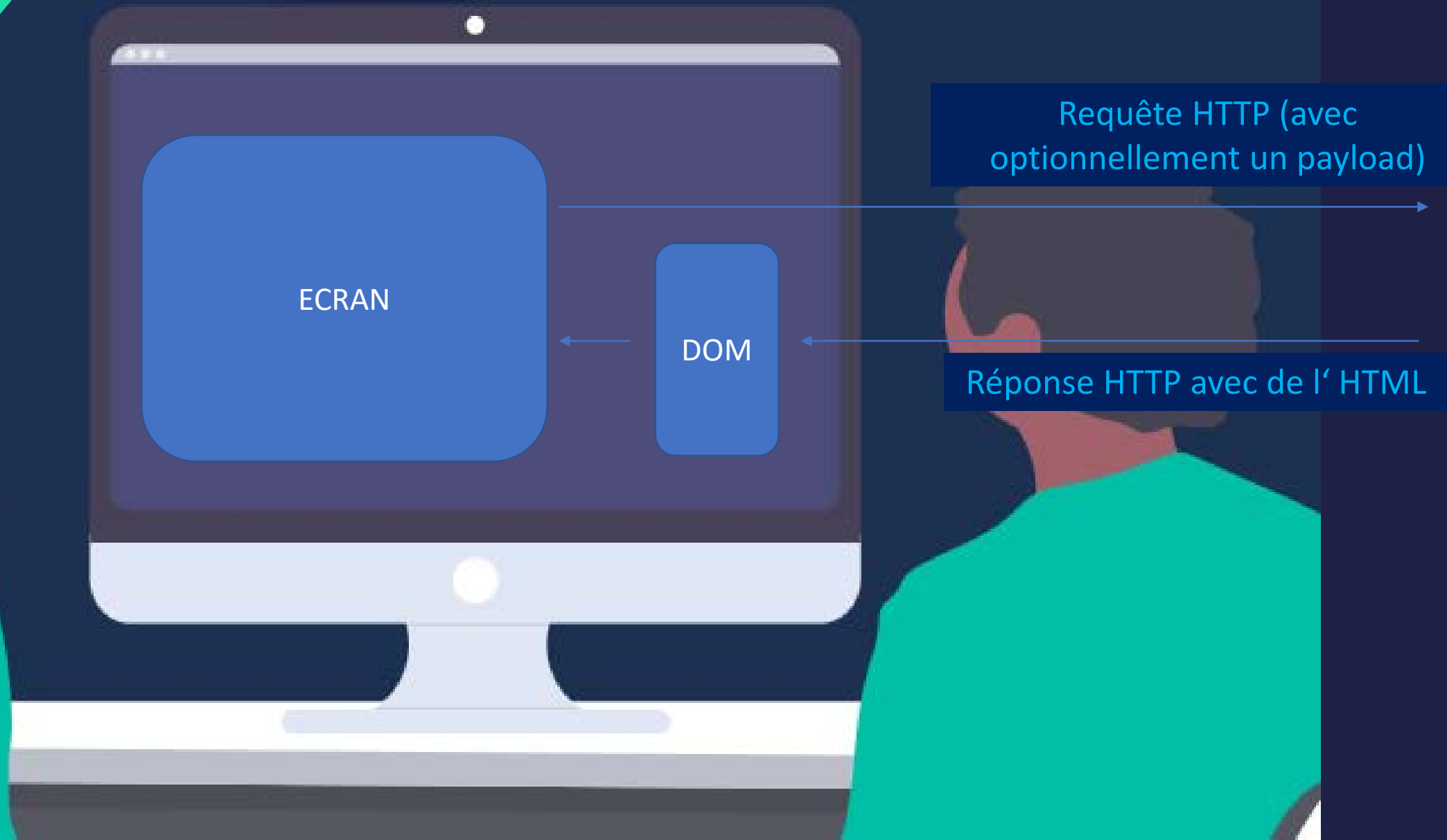
Qu'est-ce que le DOM ?







ZOOMONS DEDANS !



Le DOM est l'interprétation de l'HTML par le navigateur et c'est le DOM qui est affiché à l'écran.

Cela peut être donc faux de dire qu'à l'écran,
on voit le code HTML, car plus précisément,
on voit le DOM sur l'écran (bien que le DOM
vient de l'HTML)

C'est une nouvelle opportunité pour
les hackers !

Supposons que l'on ait un formulaire :

```
<form action="https://www.patissor.com/register" method="POST">  
  <input type="email" name="user-email" /><br />  
  <!-- ... -->  
  <button type="submit">Soumettre</button>  
</form>
```



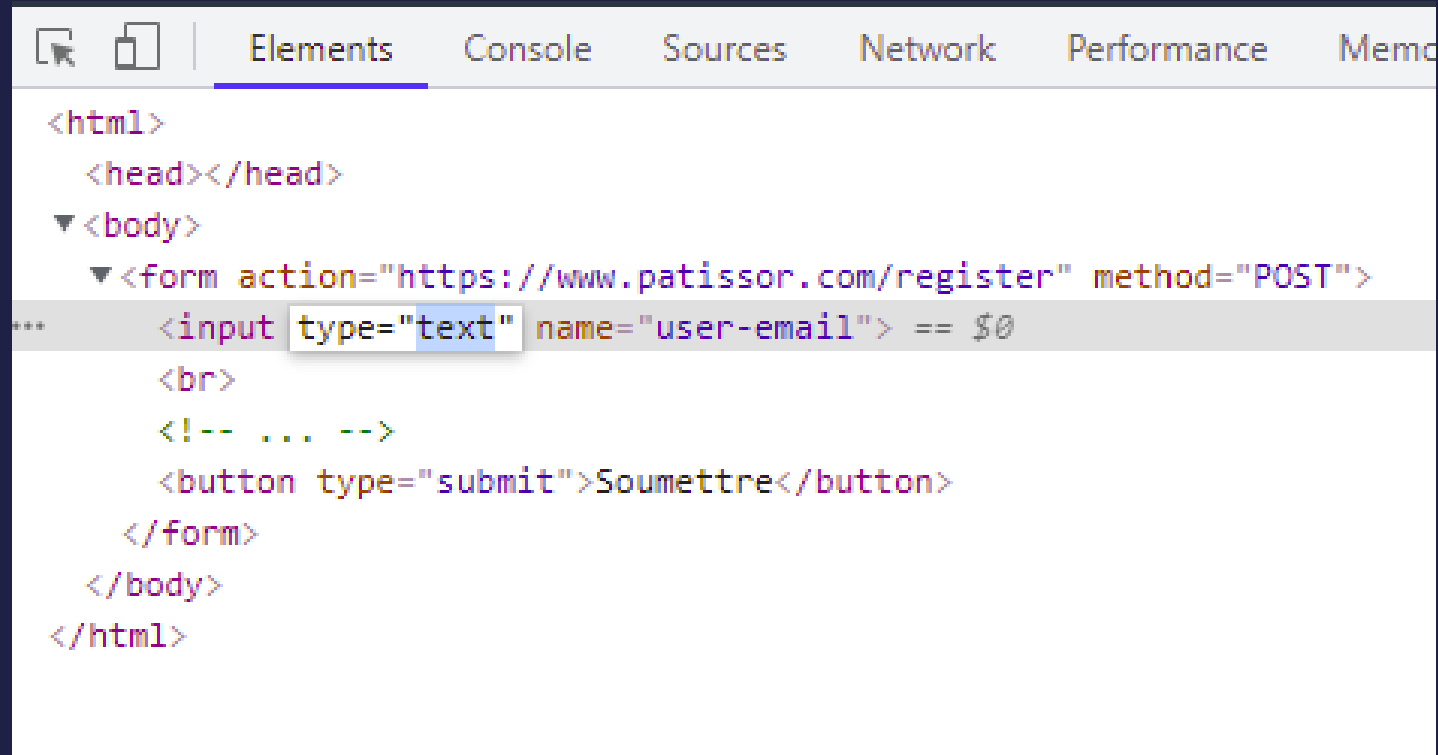
A LA PLACE D'UN HACKER ...



Veuillez inclure "@" dans l'adresse e-mail. Il manque un symbole "@" dans "laurence".

Ici, l'input email force le visiteur à rentrer une adresse email en bonne et due forme.

Modifier le DOM en changeant `type="email"` en `type="text"` permet de faire sauter cette vérification !



The screenshot shows a web browser's developer tools interface with the 'Elements' tab selected. The HTML structure is as follows:

```
<html>
  <head></head>
  <body>
    <form action="https://www.patissor.com/register" method="POST">
      <input type="text" name="user-email" > == $0
      <br>
      <!-- ... -->
      <button type="submit">Soumettre</button>
    </form>
  </body>
</html>
```

The `<input type="text" name="user-email" >` line is highlighted, and a tooltip shows the value `== $0`.

Voila ! **On a enlevé la vérification** sur l'input
facilement en modifiant l'input dans le DOM
par l'inspecteur



A LA PLACE D'UN HACKER ...

Objectifs :

1. Faire sauter les vérifications d'un site internet

Mise en place :

1. Ouvrir l'inspecteur pour avoir l'accès au DOM
2. Modifier l'élément qui nous intéresse

