BASICS WEB SECURITY

1.1 SOP in the DOM

This exercise shows a practical example of how the DOM of another origin cannot be accessed.

Create a HTML page that loads 'https://www.telenet.be' in an iframe, and try to read the content of the iframe. For example:

```
<!DOCTYPE html>
<html>
<body>

<h2>SOP example 2</h2>

<iframe id="telenet" src="https://www.telenet.be"></iframe>

<script>
    window.onload = function() {
        setTimeout(function(){
            document.getElementById('telenet').contentWindow.innerHTML;
        },5000)
    };
</script>

</body>
</html>
```

Deploy this HTML page on Netlify and browse to it.

Open developer tools. Can you read the DOM of the iframe object? Understand what happens.

1.2 SOP in an XMLHTTPRequest and CORS

This exercise shows one practical implication of the Same Origin Policy when using the XMLHttpRequest API..

Create a HTML file on your local file system, containing the following contents:

```
<!DOCTYPE html>
<html>
<body>

<h2>Using the XMLHttpRequest Object</h2>

<div id="demo">
<button type="button" onclick="loadXMLDoc()">Get Content</button>
</div>

<script>
function loadXMLDoc() {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
      document.getElementById("demo").innerHTML =
      this.responseText;
    }
  };
  xhttp.open("GET", "REPLACE_WITH_LINK_TO_YOUR_API", true);
  xhttp.send();
}
</script>

</body>
</html>
```

Make sure to replace "REPLACE_WITH_LINK_TO_YOUR_API" with the link to your API, for example https://api-websec.herokuapp.com/ .

Deploy this HTML page on Netlify and browse to it.

What happens? Can you fix it?