Tags de metadados de **segurança e rastreamento**

Usadas para proteger a página da web, melhorar a privacidade e controlar a forma como os motores de busca e outras ferramentas de rastreamento interagem com o conteúdo.

1. Metadados de Rastreabilidade e Privacidade

`<meta name="robots" content="...">`

- **Descrição**: A tag `robots` instrui os motores de busca e outros rastreadores sobre como devem interagir com a página. Ela pode ser usada para permitir ou bloquear o rastreamento e indexação da página.
- **Uso**: O conteúdo pode incluir valores como:
  - `index`: Permite que a página seja indexada.
  - `noindex`: Impede que a página seja indexada.
  - `follow`: Permite que links na página sejam seguidos.
  - `nofollow`: Impede que links na página sejam seguidos.
  - `noarchive`: Impede que o cache da página seja armazenado.
  - `nosnippet`: Impede que o snippet (resumo) da página seja exibido nos resultados de busca.
- **Exemplo**:

```
<meta name="robots" content="noindex, nofollow">
```

`<meta name="googlebot" content="...">`

- **Descrição**: Semelhante à tag `robots`, mas específica para o **Googlebot**. Ela permite controlar o comportamento do Googlebot, o rastreador do Google, de forma mais específica.
- **Uso**: Pode ter os mesmos valores que a tag `robots`, mas com a vantagem de ser direcionada apenas ao Google.
- **Exemplo**:

```
<meta name="googlebot" content="noindex, nofollow">
```

`<meta name="robots" content="max-snippet:-1, max-image-preview:large, max-video-preview:-1">`

- **Descrição**: Especifica como os motores de busca devem exibir conteúdos de imagens, vídeos e snippets na página.
  - `max-snippet:-1`: Permite exibir o conteúdo completo da página no snippet.

- `max-image-preview:large`: Permite que imagens grandes apareçam nas visualizações.
- `max-video-preview:-1`: Permite que vídeos completos apareçam na visualização.

- **Exemplo**:

```
<meta name="robots" content="max-snippet:-1, max-image-preview:large,
max-video-preview:-1">
```

---

2. Metadados de Segurança

`<meta http-equiv="Content-Security-Policy" content="...">`

- **Descrição**: A **Content Security Policy (CSP)** é uma medida de segurança importante para prevenir ataques de **cross-site scripting (XSS)**, **injeção de conteúdo**, e outros tipos de ataques maliciosos. Através dessa tag, você define quais fontes de conteúdo podem ser carregadas no seu site.
- **Uso**: O valor do `content` define as regras da política de segurança. Por exemplo, você pode restringir quais domínios podem carregar scripts, imagens ou estilos no seu site.
- **Exemplo**:

```
<meta http-equiv="Content-Security-Policy" content="default-src 'self';
script-src 'self' https://trusted-source.com;">
```

Nesse exemplo, a política permite apenas que scripts sejam carregados do mesmo domínio (`'self'`) e de `https://trusted-source.com`.

`<meta http-equiv="X-Content-Type-Options" content="nosniff">`

- **Descrição**: Esta tag ajuda a proteger seu site contra **type sniffing**. Isso significa que ela impede que o navegador tente adivinhar o tipo de conteúdo de arquivos que não são explicitamente declarados.
- **Uso**: Definir `content="nosniff"` impede que o navegador execute arquivos com tipos MIME incorretos.
- **Exemplo**:

```
<meta http-equiv="X-Content-Type-Options" content="nosniff">
```

`<meta http-equiv="Strict-Transport-Security" content="max-age=31536000; includeSubDomains">`

- **Descrição**: Esta tag força o navegador a acessar o site apenas por meio de uma conexão segura HTTPS. Isso ajuda a proteger contra ataques de **man-in-the-middle** (MITM).

- **Uso**: `max-age` define o tempo (em segundos) que o navegador deve se lembrar de usar apenas HTTPS. O valor `includeSubDomains` aplica essa política a todos os subdomínios do site.
- **Exemplo**:

```
<meta http-equiv="Strict-Transport-Security" content="max-age=31536000; includeSubDomains">
```

```
<meta http-equiv="X-Frame-Options" content="DENY">
```

- **Descrição**: Esta tag previne que seu site seja carregado em um **iframe** em outros sites. Isso ajuda a proteger contra ataques de **clickjacking**, onde um site malicioso tenta enganar o usuário para clicar em algo diferente do que ele vê.
- **Uso**: O valor `DENY` impede completamente que a página seja exibida em um iframe. Outra opção é `SAMEORIGIN`, que permite que a página seja exibida em um iframe apenas se o domínio for o mesmo.
- **Exemplo**:

```
<meta http-equiv="X-Frame-Options" content="DENY">
```

```
<meta http-equiv="Referrer-Policy" content="no-referrer">
```

- **Descrição**: A tag `Referrer-Policy` controla o envio do cabeçalho **Referer** em requisições feitas de sua página. O cabeçalho **Referer** é usado para indicar a origem de uma requisição, mas pode expor informações sensíveis.
- **Uso**: `content="no-referrer"` impede que qualquer informação de referer seja enviada com as requisições.
- **Exemplo**:

```
<meta http-equiv="Referrer-Policy" content="no-referrer">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1">
```

- **Descrição**: Embora mais comumente usada para controle de responsividade, a tag `viewport` também pode melhorar a segurança ao limitar o zoom em dispositivos móveis. Definir `maximum-scale=1` impede que o usuário altere a escala da página, o que pode ser útil para evitar certos tipos de ataques baseados em zoom.
- **Uso**: Ajuste da escala do site para uma melhor visualização e prevenção de ataques em dispositivos móveis.
- **Exemplo**:

```

```

```
<meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1">
```

Conclusão

As tags de **segurança** e **rastreamento** ajudam a proteger o conteúdo da sua página, melhorar a privacidade do usuário e controlar como a página é tratada por motores de busca e ferramentas de rastreamento. Usar essas tags de maneira adequada pode aumentar a segurança contra ataques como **cross-site scripting (XSS)**, **clickjacking** e **man-in-the-middle** (MITM), além de otimizar a forma como sua página é rastreada e exibida nas redes sociais e motores de busca.