csrf

August 17, 2026

csrf

Kind: Function

Source: `src/middleware/csrf/index.ts`

Part of: Middleware

CSRF Protection Middleware for Hono.

Protects against Cross-Site Request Forgery attacks by validating request origins and sec-fetch-site headers. The request is allowed if either validation passes.

`csrf` creates Hono middleware that checks request origin and `Sec-Fetch-Site` headers to help prevent Cross-Site Request Forgery attacks. Requests proceed when either validation passes; otherwise, the middleware rejects the request before it reaches route handlers.

Signature

```
function csrf(options: CSRFOptions): MiddlewareHandler
```

Parameters

Name	Type
`options`	`CSRFOptions`

Returns: `MiddlewareHandler`

Diagram

```
graph LR
  Request[Incoming request] --> CSRF[csrf middleware]
  CSRF --> Origin[Validate Origin header]
```

```
  CSRF --> FetchSite[Validate Sec-Fetch-Site header]
  Origin --> Allowed[Continue to route handler]
  FetchSite --> Allowed
  Origin --> Rejected[Reject request]
  FetchSite --> Rejected
```

Usage

```ts
import { Hono } from 'hono'
import { csrf } from 'hono/csrf'

const app = new Hono()

app.use(
  csrf({
    origin: 'https://app.example.com',
  })
)

app.post('/account/email', (c) => {
  return c.json({ updated: true })
})

export default app
```

AI Coding Instructions

- Register `csrf()` before routes that handle state-changing requests.
- Configure `origin` when the application accepts requests from a known external origin.
- Keep browser clients sending standard `Origin` and `Sec-Fetch-Site` headers.
- Do not bypass this middleware for protected routes unless another CSRF validation mechanism is in place.

Relationships

- IMPORTS → `HTTPException`