

basicAuth

August 17, 2026

basicAuth

Kind: Function**Source:** `src/middleware/basic-auth/index.ts`**Part of:** [Middleware](#)

Basic Auth Middleware for Hono.

`basicAuth` creates Hono middleware that checks HTTP Basic Authentication credentials before allowing a request to continue. Apply it to a route or route group to return an unauthorized response when the supplied username or password does not match the configured values.

Signature

```
function basicAuth(options: BasicAuthOptions, users: { username: string; password: string }[]):
```

Parameters

Name	Type
<code>options</code>	<code>BasicAuthOptions</code>
<code>users</code>	<code>{ username: string; password: string }[]</code>

Returns: `MiddlewareHandler`

Diagram

```
graph LR
  Request[Incoming request] --> Middleware[basicAuth middleware]
  Middleware --> Credentials[Read Authorization header]
  Credentials --> Valid{Credentials match?}
```

```
Valid -->|Yes| Handler[Route handler]
Valid -->|No| Unauthorized[Unauthorized response]
```

Usage

```
import { Hono } from 'hono'
import { basicAuth } from 'hono/basic-auth'

const app = new Hono()

app.use(
  '/admin/*',
  basicAuth({
    username: 'admin',
    password: 'secret',
  })
)

app.get('/admin/dashboard', (c) => {
  return c.text('Admin dashboard')
})

export default app
```

AI Coding Instructions

- Apply `basicAuth` with `app.use()` before the routes that require authentication.
- Scope the middleware path carefully so public routes do not require credentials.
- Keep usernames and passwords outside source code by reading them from environment configuration.
- Send credentials through the standard `Authorization: Basic ...` request header rather than custom headers.