

serveStatic

August 17, 2026

serveStatic

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

This middleware is not directly used by the user. Create a wrapper specifying `getContent()` by the environment such as Deno or Bun.

`serveStatic` contains the shared static-file middleware flow. Runtime adapters wrap it with an environment-specific `getContent()` implementation, such as one backed by Deno or Bun file APIs.

Signature

```
function serveStatic(options: ServeStaticOptions<E> & { getContent: (path: string, c: Context<E>
```

Parameters

Name	Type
<code>options</code>	<code>`ServeStaticOptions & { getContent: (path: string, c: Context) => Promise<Data</code>

Returns: `MiddlewareHandler`

Diagram

```
graph LR
  Request[Incoming request] --> Middleware[serveStatic middleware]
  Middleware --> Path[Resolve requested path]
  Path --> GetContent[getContent callback]
  GetContent --> Content[File content or response]
  Content --> Response[HTTP response]
```

Usage

```
import { Hono } from 'hono'
import { serveStatic } from 'hono/serve-static'

const app = new Hono()

app.use(
  '*',
  serveStatic({
    root: './public',
    getContent: async (path) => {
      try {
        return await Deno.readFile(path)
      } catch {
        return null
      }
    },
  })
)

export default app
```

AI Coding Instructions

- Keep `serveStatic` runtime-neutral; place `Deno`, `Bun`, or other file-system calls in an adapter through `getContent()`.
- Return `null` when a requested file does not exist so later middleware or route handling can continue.
- Preserve the requested-path and response handling flow when changing static-file behavior.
- Use the runtime-specific static middleware wrapper when one exists instead of calling this shared implementation directly.

Used by

3 references from 3 files. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (3)

- `serveStatic` — `src/adapter/bun/serve-static.ts :8`
- `ServeStaticOptions` — `src/adapter/cloudflare-workers/serve-static.ts :6`

- `serveStatic` — `src/adapters/deno/serve-static.ts` :8