

GatewayMetadata

August 17, 2026

GatewayMetadata

Kind: Interface

Source: `packages/websockets/interfaces/gateway-metadata.interface.ts`

Part of: [Websockets](#)

External interface

`GatewayMetadata` defines the configuration used to initialize and operate a WebSocket gateway. It captures the gateway namespace, HTTP path, client-serving behavior, transport adapters/parsers, and connection or heartbeat timeout settings.

Properties

Property	Type
<code>namespace</code>	<code>`string</code>
<code>path</code>	<code>string</code>
<code>serveClient</code>	<code>boolean</code>
<code>adapter</code>	<code>any</code>
<code>parser</code>	<code>any</code>
<code>connectTimeout</code>	<code>number</code>
<code>pingTimeout</code>	<code>number</code>
<code>pingInterval</code>	<code>number</code>
<code>upgradeTimeout</code>	<code>number</code>
<code>maxHttpBufferSize</code>	<code>number</code>
<code>allowRequest</code>	<code>`(req: any, fn: (err: string</code>
<code>transports</code>	<code>`Array<'polling'</code>

Property	Type
allowUpgrades	boolean
perMessageDeflate	`boolean
httpCompression	`boolean
wsEngine	string
initialPacket	any
cookie	any
cors	CorsOptions
allowEIO3	boolean
destroyUpgrade	boolean
destroyUpgradeTimeout	number

Diagram

```
graph LR
  GatewayMetadata --> Namespace["namespace: string | RegExp"]
  GatewayMetadata --> Path["path: string"]
  GatewayMetadata --> Client["serveClient: boolean"]
  GatewayMetadata --> Transport["adapter / parser"]
  GatewayMetadata --> Timeouts["Connection and upgrade timeouts"]
  GatewayMetadata --> Heartbeat["Ping interval and timeout"]
  GatewayMetadata --> Buffer["maxHttpBufferSize"]

  Timeouts --> Connect["connectTimeout"]
  Timeouts --> Upgrade["upgradeTimeout"]
  Heartbeat --> PingInterval["pingInterval"]
  Heartbeat --> PingTimeout["pingTimeout"]
```

Usage

```
import type { GatewayMetadata } from './interfaces/gateway-metadata.interface';

const chatGatewayMetadata: GatewayMetadata = {
  namespace: '/chat',
  path: '/socket.io',
  serveClient: true,
  adapter: undefined,
  parser: undefined,
  connectTimeout: 10_000,
  pingTimeout: 20_000,
```

```
pingInterval: 25_000,  
upgradeTimeout: 10_000,  
maxHttpBufferSize: 1e6,  
};  
  
// Pass this metadata to the gateway/WebSocket server initialization layer.  
initializeGateway(chatGatewayMetadata);
```

AI Coding Instructions

- Use `namespace` to isolate gateway endpoints; provide a `RegExp` only when dynamic namespace matching is required.
- Keep `path` aligned with the client connection configuration, or clients will fail to establish a WebSocket connection.
- Configure `pingInterval` and `pingTimeout` together; ensure the timeout allows for expected network latency.
- Treat `adapter` and `parser` as integration points for the underlying WebSocket implementation and provide compatible instances.
- Set `maxHttpBufferSize` conservatively to limit oversized payloads and reduce memory-pressure risks.

Relationships

- IMPORTS → `CorsOptions`