



MICROSERVICES

TcpContext

Kind: Class

VERSION
47

UPDATED
7 August 2026

DOCUMENT
[reference-subsystem-packages-microservices/class-tcpcontext](#)

CONTENTS

1	Methods	2	4	AI Coding Instructions	2
2	Diagram	2	5	How it works	2
3	Usage	2			

Kind: Class

Source: `packages/microservices/ctx-host/tcp.context.ts`

Part of: [Microservices](#)

`TcpContext` provides TCP-specific metadata to NestJS microservice message handlers. It exposes the underlying client socket and the pattern associated with the incoming message, allowing handlers to inspect connection details when needed.

Extends: `BaseRpcContext`

1 Methods

Method	Signature	Returns
<code>getSocketRef</code>	<code>getSocketRef()</code>	<code>void</code>
<code>getPattern</code>	<code>getPattern()</code>	<code>void</code>

2 Diagram

```
graph LR
  Client[TCP Client] --> Server[NestJS TCP Server]
  Server --> Handler[Message Handler]
  Handler --> Context[TcpContext]
  Context --> Socket[getSocketRef()]
  Context --> Pattern[getPattern()]
```

3 Usage

```
import { Controller } from '@nestjs/common';
import { Ctx, MessagePattern, TcpContext } from '@nestjs/microservices';

@Controller()
export class MathController {
  @MessagePattern('sum')
  sum(
    data: number[],
    @Ctx() context: TcpContext,
  ): number {
    const socket = context.getSocketRef();
    const pattern = context.getPattern();

    console.log(`Received "${pattern}" from ${socket.remoteAddress}`);
  }
}
```