

Guards

There is no fundamental difference between microservices guards and [regular HTTP application guards](#). The only difference is that instead of throwing `HttpException`, you should use `RpcException`.

info Hint The `RpcException` class is exposed from `@nestjs/microservices` package.

Binding guards

The following example uses a method-scoped guard. Just as with HTTP based applications, you can also use controller-scoped guards (i.e., prefix the controller class with a `@UseGuards()` decorator).

```
@@filename()
@UseGuards(AuthGuard)
@MessagePattern({ cmd: 'sum' })
accumulate(data: number[]): number {
  return (data || []).reduce((a, b) => a + b);
}

@@switch
@UseGuards(AuthGuard)
@MessagePattern({ cmd: 'sum' })
accumulate(data) {
  return (data || []).reduce((a, b) => a + b);
}
```