

[HummingbirdCore](#) / [HTTPServerBuilder](#) / plaintextHTTP2(configuration:)

Type Method

# plaintextHTTP2(configuration:)

Build plaintext HTTP2 channel

```
static func plaintextHTTP2(configuration: HTTP2Channel.Configuration = .init  
Builder
```

## Parameters

### **configuration**

HTTP2 channel configuration

## Return Value

HTTPChannelHandler builder

## Discussion

As this is running on a connection without TLS it cannot perform the upgrade negotiation via ALPN. Therefore a client will need to know in advance it is connecting to an HTTP2 server. You can test this with curl as follows: `curl --http2-prior-knowledge http://localhost:8080/`

Use in [Application](#) initialization.

```
let app = Application(  
    router: router,
```

```
server: .plaintextHTTP2()  
)
```

## See Also

### Server

```
static func http2Upgrade(tlsConfiguration: TLSConfiguration,  
configuration: HTTP2UpgradeChannel.Configuration) throws ->  
HTTPServerBuilder
```

Build HTTP channel with HTTP2 upgrade

```
static func http2Upgrade(tlsChannelConfiguration: TLSChannel  
Configuration, configuration: HTTP2UpgradeChannel.Configuration)  
throws -> HTTPServerBuilder
```

Build HTTP channel with HTTP2 upgrade

```
struct HTTP2UpgradeChannel
```

Child channel for processing HTTP1 with the option of upgrading to HTTP2 via ALPN

```
struct HTTP2Channel
```

Child channel for processing HTTP2