

Setting up HTTP Server



Content

1. Client Server Architecture
2. HTTP & HTTPS
3. Setting up HTTP server
4. Postman

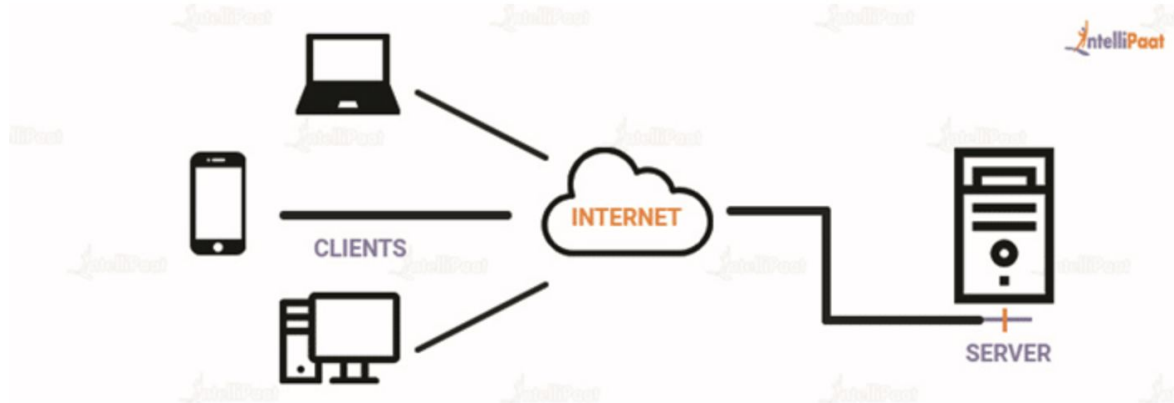
Ever wondered How you are able to view live score on internet ?

The screenshot displays the Cricbuzz website interface. At the top, there's a navigation bar with links for Live Scores, Schedule, Archives, News, Series, Teams, Videos, Rankings, and more. Below this, a 'FEATURED MATCHES' section shows live scores for various matches, including AUS vs ENG, CD vs OTG, CLS vs KDW, GGD vs JKS, and PAK vs WI. The main content area features a 'LATEST NEWS' section with articles about quarantine issues, KL Rahul's appointment, and Gautam Gambhir's role. A large article titled 'Australia dominant after bagging massive lead' is prominently displayed, accompanied by a photo of the Australian team celebrating. To the right, there's a 'FEATURED VIDEOS' section with a video titled 'Cricbuzz Chatter: Australia take a lead of 282 on Day 3! Michael Hussey reviews'. The website also includes various advertisements, such as 'Zero Processing Fee on Personal Loan up to ₹30 lakh' and 'Airtel Broadband @499'.

Backbone of all the interactions on internet is a simple architecture!

CLIENT-SERVER Architecture

Imagine Client as your interface
and Server where the actual
information resides



Client

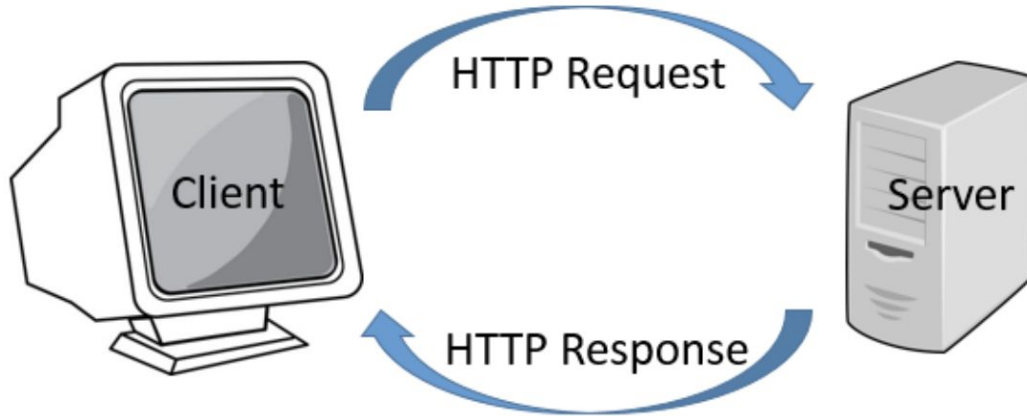
1. Requests Information
2. Depends upon server
3. Holds no additional resources

Server

1. Responsible for Hosting and managing resources which compute results for client
2. Abstracts out the implementation
3. Responds to whatever client asks for

How does the communication between Client and Server actually hap

HTTP to the rescue !!!!



Types of HTTP Requests

1. GET
2. POST
3. PUT
4. DELETE
5. HEAD
6. TRACE
7. OPTIONS
8. PATCH

Sample HTTP Request

1. GET /test HTTP/1.1

User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)

Host: www.tutorialspoint.com

Accept-Language: en-us

Accept-Encoding: gzip, deflate

Connection: Keep-Alive

HTTP Response

The screenshot shows the Chrome DevTools Network tab. The top panel displays a timeline of network requests. The selected request is expanded, showing the following details:

- Name:** `?relatedposts=1`
- General:**
 - Request URL: `https://gavilan.blog/wp-admin/admin-ajax.php`
 - Request Method: `POST`
 - Status Code: `200`
 - Remote Address: `192.0.78.25:443`
 - Referrer Policy: `strict-origin-when-cross-origin`
- Response Headers:**
 - `access-control-allow-credentials: true`
 - `access-control-allow-origin: https://gavilan.blog`
 - `cache-control: no-cache, must-revalidate, max-age=0`
 - `content-encoding: br`
 - `content-type: text/html; charset=UTF-8`
 - `date: Thu, 16 Dec 2021 07:56:38 GMT`
 - `expires: Wed, 11 Jan 1984 05:00:00 GMT`
 - `host-header: WordPress.com`
 - `referrer-policy: strict-origin-when-cross-origin`
 - `server: nginx`
 - `strict-transport-security: max-age=31536000`
 - `vary: Accept-Encoding`
 - `vary: Origin`
 - `x-ac: 3.bom_dca`
 - `x-content-type-options: nosniff`
 - `x-frame-options: SAMEORIGIN`
 - `x-hacker: If you're reading this, you should visit automattic.com/jobs and apply to join the fun, mention this header.`
 - `x-robots-tag: noindex`
- Request Headers:**
 - `:authority: gavilan.blog`
 - `:method: POST`
 - `:path: /wp-admin/admin-ajax.php`
 - `:scheme: https`
 - `accept: */*`

At the bottom left, it indicates `4 / 49 requests` and `2.0 kB / 211 kB`.

HTTP Response codes

1. 100's
2. 200's
3. 300's
4. 400's
5. 500's

GET VS POST METHOD

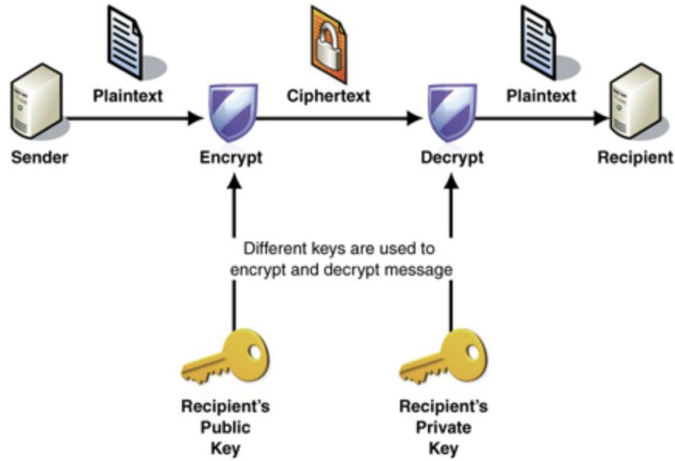
Discuss

1. Reloading
2. Bookmarking
3. Caching
4. Security
5. Size

Is HTTP Secure?

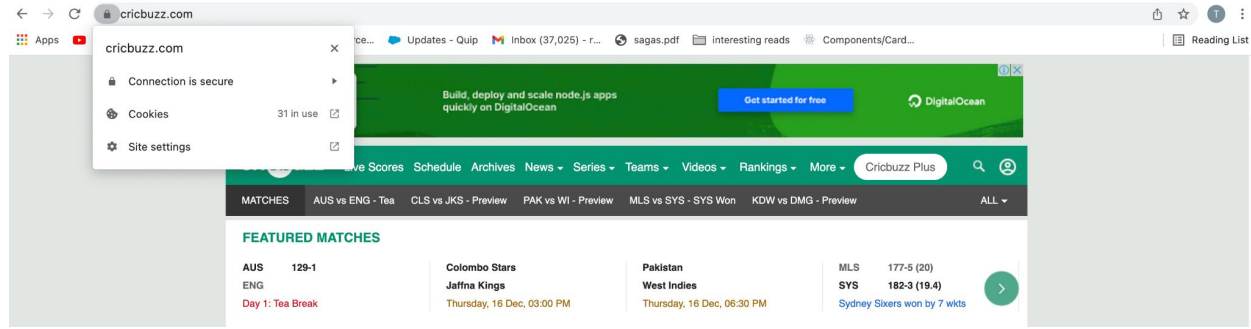
Intro to HTTPS (Hypertext Transfer Protocol Secure)

Encryption at work!!!



Example of asymmetric encryption system

Is your current web page Secure ?



REST (Representational state transfer)

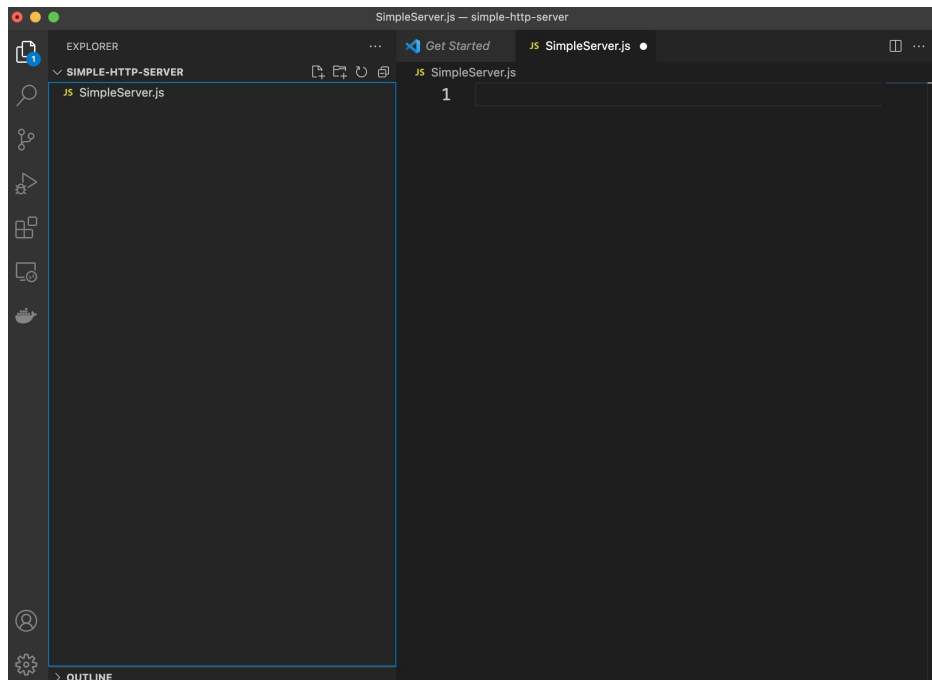
1. Client-Server architecture
2. Cacheability
3. Layered-System
4. Stateless
5. Uniform Interface

Lets Build our own HTTP Server Now!!!

Pre-Req

- Code editor
- Git (Optional)
- Node js installed

Set up Root directory



Create HTTP Server

← → ↻ ⓘ localhost:8080

Our first http server

Create HTTP Server with json response0

← → ↻ ⓘ localhost:8080

```
{"message": "This is a JSON response from our json Server "}
```

Create HTTP Server with CSV response

details

name	rollNumber	school
Tushar raina	34	Thapar

MCQs

1. If we have to update our password, what HTTP method should we use ?

- A. GET
- B. PUT
- C. POST
- D. UPDATE

Answer: B

2. If there is a Null pointer exception in server code, what response should it return?

- A. 404
- B. 302
- C. 500
- D. 201

Answer: C

3. Which of the following methods has no message body?

- A. POST
- B. GET
- C. PUT
- D. DELETE

Answer: B

4. Find incorrect mapping.

- A. 200 OK
- B. 400 Bad Request
- C. 402 Not Found
- D. 301 Moved Permanently

Answer: C

5. If we have to configure a RESTFull url, for searching a book given its id, what will the request look like.

- A. GET /{id}/books/
- B. GET /{Id}
- C. GET /Books/{id}
- D. GET /Books?id={id1}

Answer: C

Homework!

1. Let us change the port to 8448
2. Let us return error 400 error code

Thank You!