

HTTP

➔ WHAT IS HTTP

- **Hyper Text Transfer Protocol**
- Communication between web servers & Clients
- HTTP Requests /Responses
- Loading pages, from submit, Ajax calls

➔ HTTP IS STATELESS

- Every Request is completely **independent**
- Similar to transactions
- Programming, Local Storage, Cookies, Sessions are used to create enhanced user experiences.

➔ WHAT IS HTTPS

- Hyper Text Transfer Protocol Secure
- Data sent is encrypted
- SSL / TLS
- Install certificate on web host

➔ HTTP METHODS

- **GET**
Retrieves data from the server
- **POST**
Submit data to the server
- **PUT**
Update data already on the Server
- **DELETE**
Deletes data from the server

HTTP HEADER FIELDS

```

method      path      protocol
GET /tutorials/other/top-20-mysql-best-practices/ HTTP/1.1
Host: net.tutaplus.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Cookie: PHPSESSID=r2t5uvjq435r4q7ib3vtdjq120
Pragma: no-cache
Cache-Control: no-cache

```

HTTP headers as Name: Value

General:	Response:	Request:
Request URL	Server	Cookies
Request Method	Set-Cookie	Accept-xxx
Status Code	Content-Type	Content-Type
Remote Address	Content-Length	Content-Length
Referrer Policy	Date	Authorization
		User-Agent
		Referrer

HTTP STATUS CODES

1xx : Informational

Request recieved / processing

2xx: Success

Successfully Recieved, understood and accepted

3xx: Redirect

Further action must be taken / redirect

4xx: Client Error

Request does not have what it needs

5xx: Server Error

Server failed to fulfil an apparent valid request

200 - OK

201 - OK created

301 - Moved to new URL

304 - Not modified (Cached version)

400 - Bad request

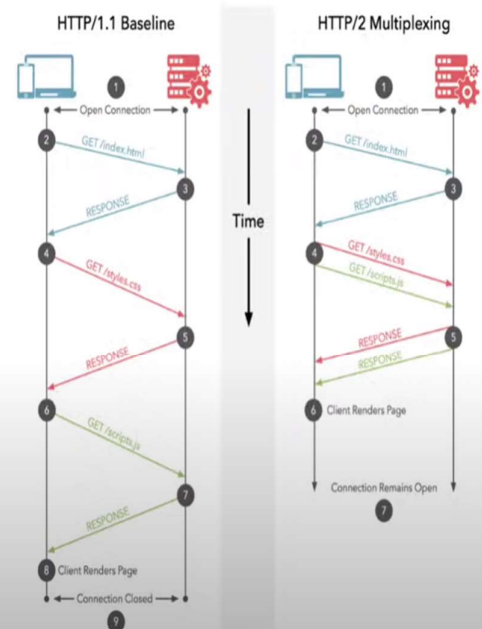
401 - Unauthorized

404 - Not found

500 - Internal server error

HTTP/2

- Major revision of HTTP
- Under the hood changes
- Respond with more data
- Reduce latency by enabling full request and response multiplexing
- Fast, efficient & secure



➔ Here we see all types of document file, CSS file, html file script file if we want to check it then we have a option to open all this file.

Responsive 413 x 634 100%

docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/cost-alloc-tags.html

Apps Webmail Login GIT_Time_Track - G... GitHub Docker Hub Amazon Web Serv... Ultimate AWS Certif... Sign in [Jenkins] GitLab YAMLint - The YAM... Co-WIN Application Reading list

Elements Console Sources Network Performance Memory Application Security Lighthouse

Filter Hide data URLs All XHR JS CSS Img Media Font Doc WS Manifest Other Has blocked cookies Blocked Requests

500 ms 1000 ms 1500 ms 2000 ms 2500 ms 3000 ms 3500 ms

Key Value

i-4a1c2f5d

- aws.createdBy = Root:123456789
- user.Cost Center = 78925
- user.Stack = Test

i-1a2b3c4d

- aws.createdBy = Root:123456789
- user.Cost Center = 78925
- user.Stack = Production

After you or AWS applies tags to your AWS resources (such as Amazon EC2 instances or Amazon S3 buckets) and you activate the tags in the Billing and Cost Management console, AWS generates a cost allocation report as a comma-separated value (CSV file) with your usage and costs grouped by your active tags. You can apply tags that represent business categories (such as cost centers, application names, or owners) to organize your costs across multiple services.

Name	Status	Type	Initiator	Size	Time	Waterfall
cost-alloc-tags.html	200	document	Other	5.3 kB	304 ms	
vendor4.css?version=2020.09.30	200	stylesheet	cost-alloc-tags.html	102 kB	1.35 s	
awsdocs-common.css?version=2020.09.30	200	stylesheet	cost-alloc-tags.html	10.2 kB	284 ms	
awshome_s_code.js	200	script	cost-alloc-tags.html	(disk cache)	8 ms	
vendor3.js?version=2020.09.30	200	script	cost-alloc-tags.html	64.3 kB	1.13 s	
vendor4.js?version=2020.09.30	200	script	cost-alloc-tags.html	68.1 kB	1.14 s	
vendor1.js?version=2020.09.30	200	script	cost-alloc-tags.html	107 kB	1.05 s	
awsdocs-common.js?version=2020.09.30	200	script	cost-alloc-tags.html	39.9 kB	857 ms	
awsdocs-doc-app.js?version=2020.09.30	200	script	cost-alloc-tags.html	17.7 kB	591 ms	
Tag_Example.png	304	png	cost-alloc-tags.html	321 B	295 ms	
CostAllocationPartExampleReport.png	304	png	cost-alloc-tags.html	321 B	292 ms	
data:application/x-...	200	font	vendor4.css?version=2020...	17.7 kB	527 ms	
data:application/x-...	200	font	vendor4.css?version=2020...	18.2 kB	532 ms	
data:application/x-...	200	font	vendor4.css?version=2020...	18.8 kB	535 ms	
data:application/x-...	200	font	vendor4.css?version=2020...	18.1 kB	539 ms	
1x1.png?severity=info&message=checkForCookieCon...	200	xhr	awshome_s_code.js:1	584 B	832 ms	
s758081264385507AQB=1&ndh=1&pf=1&t=4%2F5%...	200	gif	awshome_s_code.js:1	352 B	41 ms	
1x1.png?severity=info&message=checkForCookieCon...	200	xhr	vendor1.js?version=2020.0...	583 B	825 ms	
toc-contents.json	304	xhr	vendor3.js?version=2020.0...	332 B	284 ms	
aws_logo_dark.png	304	png	vendor3.js?version=2020.0...	320 B	286 ms	
feature.txt?code_theme_load=light	304	xhr	vendor3.js?version=2020.0...	328 B	279 ms	
feature.txt?text_theme_load=awsdocs-theme-light	304	xhr	vendor3.js?version=2020.0...	328 B	311 ms	
cost-alloc-taas.html	200	xhr	vendor3.js?version=2020.0...	5.3 kB	289 ms	

25 requests 511 kB transferred 1.9 MB resources Finish: 3.10 s DOMContentLoaded: 2.23 s Load: 2.54 s

Console Issues

Type here to search

33°C Haze 09:33 IN 04-06-2021

➔ If we want to check header response we have a option to check all the things here.

The screenshot displays a web browser window with the URL `docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/cost-alloc-tags.html`. The browser's developer tools are open, showing the Network tab. The selected resource is `cost-alloc-tags.html`, and the 'Headers' sub-tab is active. The 'Response Headers' section is expanded, revealing various headers including `Accept-Ranges: bytes`, `Cache-Control: no-store, no-cache, must-revalidate`, `Content-Encoding: gzip`, `Content-Length: 4730`, `Content-Type: text/html`, `Date: Fri, 04 Jun 2021 04:12:24 GMT`, `ETag: "4e8e-5c3e4ef75fc40-gzip"`, `Expires: Thu, 01 Jan 1970 00:00:00 GMT`, `Last-Modified: Thu, 03 Jun 2021 23:28:41 GMT`, `Permissions-Policy: interest-cohort=()`, `Server: Server`, `Vary: Accept-Encoding, User-Agent, Content-Type, Accept-Encoding, X-Amzn-CDN-Cache, X-Amzn-AX-Treatment, User-Agent`, `x-amz-rid: 271FPG619P0GT78E55AG`, and `X-Frame-Options: SAMEORIGIN`.

On the left side of the browser window, there is a diagram illustrating the structure of AWS Cost Allocation Tags. It shows two EC2 instances, `i-4a1c2f5d` and `i-1a2b3c4d`, each with a set of tags. The tags are organized into a table with 'Key' and 'Value' columns:

Key	Value
<code>aws.createdBy</code>	<code>Root.123456789</code>
<code>user.Cost Center</code>	<code>78925</code>
<code>user.Stack</code>	<code>Test</code>

Below the diagram, a text box explains that after applying tags to AWS resources and activating them in the Billing and Cost Management console, AWS generates a cost allocation report as a comma-separated value (CSV) file. This report groups usage and costs by active tags, allowing for organization of costs across business categories like cost centers, application names, or owners.

➔ Install postman and create new workspace

