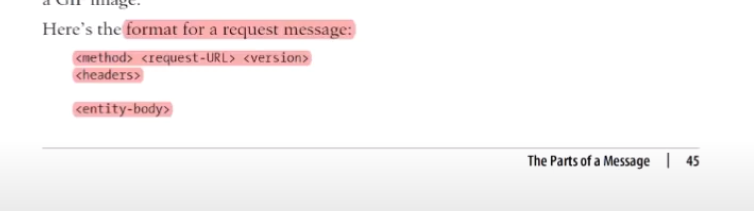
فصل سوم:

Inbound ترافیکی هست که به سمت سرور ارسال میشود و ترافیک outbound ترافیکی هست که به سمت کلاینت یا یوزر ارسال میشود و از سرور خارج میشود. همیشه روند ارسال پیام از up stream به سمت downstream خواهد بود ارسال کننده همیشه upstream و دریافت کننده میشود downstream چه کلاینت باشد چه سرور یعنی هر دو میتوانند upstream یا downstream باشند به جریان کار داریم به flow کار داریم. پیام های http در دو دسته قرار میگیرند یا request هستند یا response هستند خط آغازین درخواست شامل کد های اسکی هست و هر خط به وسیله کاراکتر CRLF از خط دیگری جدا میشود. بخش body اختیاری هست و برخلاف هدر میتواند باینری هم باشد. Start line اول پیام هست بعد هدر بعد بادی.

درخواست مخصوص request:



یعنی اول متد بعد URL که از سرور میخواهیم بعد ورژنی که داریم از HTTP استفاده میکنیم هدر میتواند خالی باشد. در نهایت یک خط خالی CRLF هست بعد به بادی میرسیم و بادی میتواند خالی باشد. متد کاری هست که از سرور میخواهیم انجام دهد مثل GET POST و ...، به همراه درخواست ها یک status code داریم که وضعیت درخواست را نشان میدهد همچنین یک reason-phrase داریم که به زبان انسان ها نوشته شده است و قابل فهم هست برای انسان ها. دقت کنید اگر دستی نسخه http رو تنظیم نکنید خودش به صورت خودکار به نسخه ای که خود صفر پیش فرض دارد تغییر میدهد. بخش هدر باید حتما با یک CRLF خاتمه پیدا کند مگر اینکه در بعضی سرور ها خودشان این را حذف کنند و گرنه درست نیست. برای درخواست مشخص میکنیم چه کاری باید انجام شود و در پاسخ مشخص میشود چه کاری انجام شده است. قبل از تا قبل HTTP 1.0 نیازی نبود که حتما نسخه HTTP قید بشود مثلا برای ورژن 0.8 نیازی نبود درخواست این را داشته باشند. Response line: خط ابتدایی پاسخ هست مثل خط ابتدایی درخواست و همه فیلد ها مثل ورژن و .... با space از همدیگر جدا میشوند. پاسخ ها هم تا قبل از نسخه 1.0 نیازی به نوشتن نسخه نداشتند. Get برای گرفتن سند از سرور هست. متد head برای گرفتن هدر های پاسخ هست که از سمت سرور میاد. دقت شود بین تمام متد های موجود فقط post, put بادی مسیج دارند و حتما باید داشته باشند. یعنی وقتی head میزنیم جای گت دیگر بادی ندارد فقط هدر ها میاد. گت تاخیر و حجم خیلی بیشتری دارد نسبت به head برای مواقع develop یعنی صرفا فقط بخواهیم نتیجه یک درخواست را بفهمیم. متد options را فقط یک سری از سرور ها دارند و وقتی بفرستی سرور میگوید از چه متد هایی پشتیبانی میکند مثلا میگوید من گت و پست را پشتیبانی میکنم. هفت تا متد را همه سرور ها پشتیبانی نمیکنند. متد دیلیت هم یک سندی رو حذف میکند. بعضی از سرور ها ممکن هست یک سری متد دیگر داشته باشند که خودشان تولید کرده اند که به آن extension method گفته میشود. status code نتیجه رسیدگی به درخواست را نشان میدهند مثلا خانواده 100 برای اطلاعات هست 200 برای عملیات های موفقیت 300 برای redirection هست 500 به بعد ارور سرور هست و 400 تا 500 از سمت ما هست. ارور 401: یعنی اجازه دسترسی به اون منبع نداریم و باید یوزرنیم و پسورد بدهیم. اولین نسخه http 0.9 هستش. برای کار با http 1.1 فقط گت و head را باید پیاده سازی بکند و برای سایر متد ها باید محدودیت هایی به دلایل امنیتی قائل شویم. Get, head به عنوان متد های امن شناخته میشوند چون که بر اساس آن هیچ تغییری ایجاد نمیشود بر روی منابع. متد گت برای دسترسی به یک منبع در سمت سرور هست head فقط به هد response دسترسی پیدا میکنه ولی منبع را دریافت نمیکنیم صرفا وجود آن و چک کردن تغییر یا عدم تغییر آن استفاده میکنیم کلا به اطلاعات جواب کار داریم نه منابع. Post برای فرستادن اطلاعات به سمت سرور هست. trace یعنی اینکه میخواهیم یک درخواست بفرستیم ببینیم بعد از فایر وال یا پراکسی درست فرستاده شده است یا نه از این استفاده میکنیم تا متن خود درخواست به ما برسد. متد option میاد از سمت سرور اعلام میشود که چه متد هایی را پشتیبانی میکند. متد lock واسه وقتی هست که وقتی یک منبع دارد تغییر پیدا میکند قفل شود تا دیگران نتوانند آن را تغییر بدهند متد های کپی و move هم داریم. 301 یعنی این منبعی که دنبالشی به صورت دائمی منتقل شده است به لوکیشن جدید که آدرس لوکیشن جدید را هم به ما میدهد. با دستور if-modified since چک میکند که آیا از این تاریخی که کاربر فرستاده اون منبع دچار تغییر شده است یا نه اگر شده بود که دریافت میکند اگر نه دیگر از سرور نمیگیرد تا سرعت سرور بالاتر برود.