

توضیح پروتکل:

داده هایی که برای هر دستور باید ارسال شوند به چند بخش تقسیم شده است.

بخش اول که در کد **command** نام گذاری شده است تعیین کننده نوع دستوری هست که از سرور به کلاینت یا برعکس، انتقال یافته است. اندازه **command** مقدار ثابت 64 بایت در نظر گرفته شده است.

بنابراین طول دستورات باید حداکثر 64 کاراکتر باشد و اگر کمتر از 64 کاراکتر باشد در انتهای آن به تعداد مورد نیاز **space** قرار داده شده و بعد ارسال می شود.

بسته به نوع دستور می توان در ادامه **title** و **message** نیز ارسال شوند. اندازه **title** نیز مقدار ثابت 64 بایت در نظر گرفته شده است. و اندازه **message** مقدار متغیری است که قبل از ارسال محاسبه می شود و در قالب 64 بایتی به طرف مقابل ارسال می شود تا طرف مقابل دقیقاً به اندازه **message** بخواند.

یک دیکشنری برای پیغام ها در سرور در نظر گرفته شده است که تمام پیغام ها در قالب {موضوع : متن} ذخیره می شوند. همچنین برای هر موضوع یک لیست از کلاینت های عضو شده در قالب دیکشنری {موضوع : [clients]} در نظر گرفته شده است.

با هر **publish** توسط یک کلاینت، این پیغام ذخیره شده و سپس به تمام کلاینت هایی که در لیست آن موضوع هستند ارسال می شود.

همچنین با هر **subscribe** توسط یک کلاینت، تمام پیغام های ذخیره شده مربوط به آن موضوعات به **client** ارسال می شود.

طبق دستور کار برای هر موضوع که **client** درخواست **subscribe** داده یک **thread** ساخته می شود و این **thread** مسئول دریافت پیام های مربوط به آن موضوع از سرور است.

همچنین مکانیزم های **puback** و **suback** نیز به طور کامل پیاده سازی شده است.

در **publish**، اگر پیغام **publish** نشود بعد از ۱۰ ثانیه پیغام عدم موفقیت توسط کلاینت برای کاربر چاپ می شود.

همچنین در **subscribe**، برای هر موضوع یک **suback** جداگانه در نظر گرفته شده تا به طور دقیق موضوعاتی که کلاینت **subscribe** نشده چاپ شود.

امتیازی:

موارد امتیازی اعم از مکانیزم ۱۰ ثانیه و ۳ بار فرصت برای pong پیاده سازی شدند.

بعد از ایجاد ارتباط با کلاینت، اگر کلاینت به مدت ۱۰ ثانیه با سرور ارتباط برقرار نکند سرور پیغام ping ارسال می کند اگر بعد از ۱۰ ثانیه کلاینت pong ارسال نکند در ترمینال سرور عبارت گفته شده چاپ می شود و بعد از ۳ دوره متوالی عدم پاسخگویی از کلاینت ارتباط سرور با کلاینت بسته شده و از لیست subscriber ها بیرون می آید.

مکانیزم های دیگر پیاده سازی شده:

- برای هر subscriber یک مکانیزم خروج از برنامه در نظر گرفته شده به این صورت که هر ۱ ثانیه به صورت یک وقفه از recv کردن دست بر می دارد و مقدار EXIT را می خواند اگر این مقدار True شود برنامه متوقف می شود.
مقدار EXIT توسط کاربر با وارد کردن exit در ترمینال True می شود.
- یک دستور دیگر به نام Disconnect پیاده سازی شده که قبل از قطع ارتباط توسط کلاینت چه به عنوان publisher یا subscriber به سرور ارسال می شود تا سرور نیز آن کلاینت را از لیست subscriber ها خارج کرده و ارتباط با آن را قطع کند.

در انتها مقادیر مختلفی که کاربر هنگام شروع برنامه به آرگومان ها می دهد تست شده و در صورت نامعتبر بودن هر یک از مقادیر، خطای مناسب برای کاربر چاپ شود.