به نام خدا

این برنامه یک چت روم با زبان پایتون و بر اساس معماری Peer-to-Peer **است که از پروتکل** TCP **برای اسال پیام های رمز شده به کمک متد** AES **رمز نگاری شده است.**

با توجه به اینکه برنامه حساسیت زیادی به تاخیر های زمانی لحظه ای ندارد و ارسال کامل و صحیح پیام ها برای ما از اهمیت بیشتری برخوردار است پس از پروتکل TCP **برای ارسال پیام ها استفاده شده است که به نسبت** UDP **بسته ها با ضمانت بالا و درصد خطای کمتر در مقصد دریافت میشوند.**

سناریوی کلی

یک کاربر با آدرس آیپی و پورت مشخص وارد برنامه میشود و میتواند اقدام به چت با فرد جدیدی کند یا پیام های ارسالی از سمت دیگر کاربران را مشاهده نماید و به آن ها پاسخ دهد.

هر کاربر که بعنوان یک Peer **شناخته میشود لیستی از دیگر** Peer **ها دارد که با آنها تا کنون مکالمه داشته است و به ازای هر** Peer **یک** thread **وجود دارد که در حال** listening **است و پیام های ارسالی از سوی آن کاربر را دریافت میکند.**

همچنین یک thread **دیگر تمام مدت در حال چک کردن و** accept **کردن** peer **های جدیدی است که درخواست شروع مکالمه را دارند**

در هر شیی Peer **یک آبجکت کلاس** message **وجود دارد که در آن تمامی پیام های رد و بدل شده میان ما و آن** peer **است به همراه تاریخ ارسال و دیده شدن یا نشدن آن پیام توسط طرفین است.**

رمز نگاری

برای انتخاب الگوریتم رمز نگاری ابتدا الگوریتم های سزار و Vigenère **بود که به کمک روش های ریاضی یا آماری (به کمک فرکانس حروف) میتوانستیم به متن رمزشده دستیابی پیدا کنیم.**

گزینه ی بعدی الگوریتم DES **بود که به علت های زیر گزینه ی مناسبی به نظر نمی آمد:**

- وجود چهار کلید ضعیف که استفاده از آن ها میتواند رمزنگاری را ضعیف کند
- وجود 6 کلید نسبتا ضعیف
- همچنین چون مکمل یک کلید متن DES شده ی آن هم مکمل است پس عملا تعداد کلید های قابل استفاده به نصف کاهش پیدا میکند.

گزینه ی بعدی الگوریتم AES است که از دسته الگوریتم هایی است که هر کاربر یک کلید عمومی دارد که همه افراد درون شبکه از آن آگاهند و یک کلید خصوصی که فقط خودش از آن اطلاع دارد.

این الگوریتم ازNP بودن مسئله prime factorization و اینکه الگوریتم کارآمدی برای حل این مسئله وجود ندارد استفاده میکند.برای رمز کردن یک پیام اگر از کلید عمومی فرد مقابل استفاده کنیم میتوانیم اطمینان پیدا کنیم که فقط فردی که کلید خصوصی را دارد میتواند آن را رمز گشایی کند.(با توجه به نحوه درست شدن کلید ها و قضیه ی اویلر)

در ادامه در این الگوریتم میان دو کاربر و به ازای هر جلسه باید یک session key درست شود که میان این دو نفر مشترک است و برای این کار ما نیاز به داشتن public key فرد مقابل داریم.

در این برنامه پس از برقراری ارتباط TCP افراد برای یکدیگر کلید عمومی خود را ارسال میکنند. البته این روش ایمن نیست و سناریو های مختلفی برای سواستفاده وجود دارد برای مثال اگر Alice و Bob بخواهند باهم ارتباط برقرار کنند. اگر فرد سومی کلید عمومی خودش را برای Bob بفرستد میتواند خودش را جای Alice بزند و بجای Alice با Bob مکالمه کند.

برای حل این مشکل دو روش PKI و Web of Trust وجود دارد که به علت محدودیت زمانی انجام نشد.