باسمه تعالی

امنیت در اینترنت اشیاء

دکتر احمدی

تمرین دوم

MQTT

پوریا دادخواه

401201381

1. راه‌اندازی اولیه

این تمرین را در سیستم عامل Linux (Ubuntu 22.04) پیاده‌سازی کردیم. در ابتدا بروکر mosquitto را روی سیستم با استفاده از دستور زیر نصب می‌کنیم:

```
sudo apt-get install mosquitto mosquitto-clients -y
```

mosquito که فایل اصلی مربوط به سرور بروکر و mosquito_clients مربوط به پیاده‌سازی client اسن که بتوان با استفاده از mosquito_pub و mosquito_sub کاربران مشخص را تعریف و در تاپیک‌های مدنظر publish و subscribe کنیم.

برای راه‌اندازی بروکر کافیست با کانفیگ مناسب آن را اجرا کنیم: در حالت کلی تمامی فایل‌های کانفیگی که می‌نویسیم را در ادرس /etc/mosquito/conf.d قرار می‌دهیم.

Mosquito –c sampl.conf

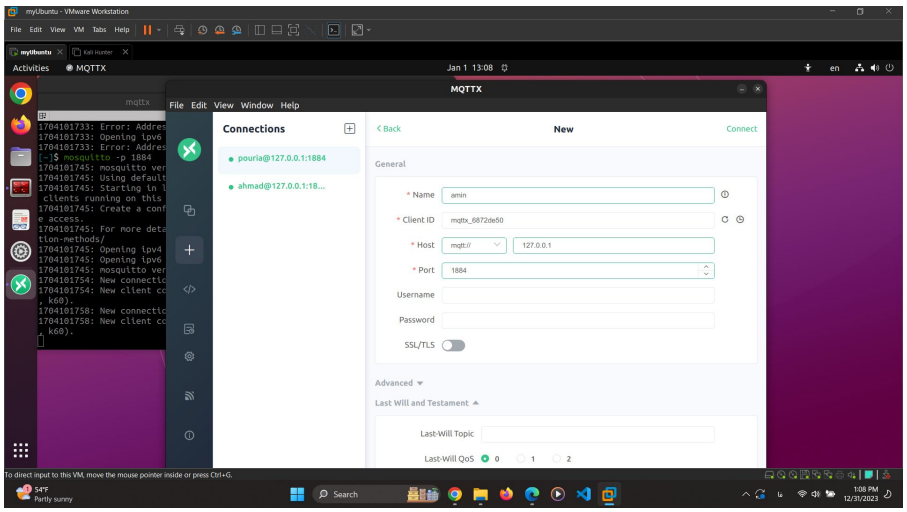
و یا می‌توان بدون هیچ کانفیگ خاصی به صورت پیش فرض اجرا کرد.

در این قسمت یک فایل کانفیگ ساده نوشته که صرفا پورت اجرا را از پیش فرض 1883 به 1884 تغییر می‌دهیم. فایل کانفیگ در پوشه تمرین ضمیمه شده‌است. همچنین allow_anonymous نیز در حالت عادی true می‌باشد ولی ما نیز آن را در فایل کانفیگ ذکر می‌کنیم.

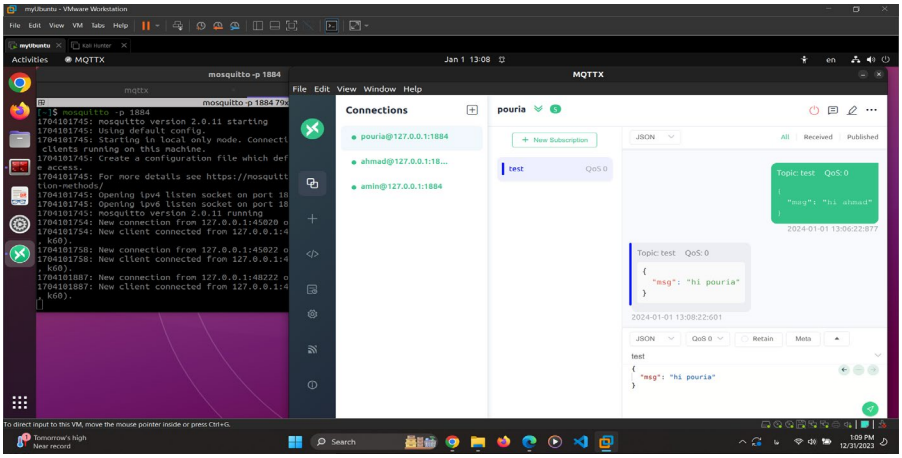
پس از راه‌اندازی بروکر باید client ها را تعریف و متصل کنیم. در این قسمت از mqttx استفاده کردیم و3 کاربر با نام Pouria و و amin ,ahmad تعریف کرده که به بروکر روی ip پیش‌فرض داخلی 127.0.0.1 متصل می‌شوند و هیچ نام‌کاربری و پسوردی برای اتصال نیاز ندارند:

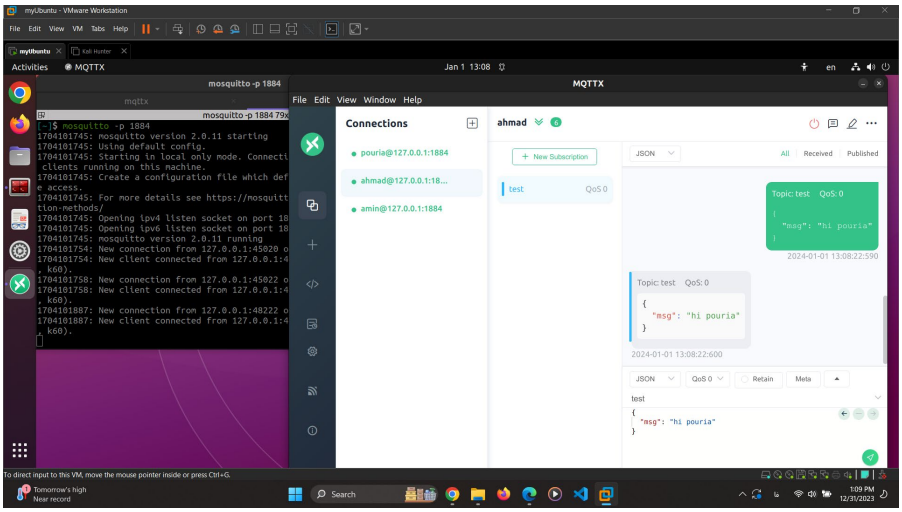
```
[/etc/mosquitto/conf.d]$ mosquitto -c port.conf
1704214673: The 'port' option is now deprecated and will be removed in a future
 version. Please use 'listener' instead.
1704214673: mosquitto version 2.0.11 starting
1704214673: Config loaded from port.conf.
1704214673: Opening ipv4 listen socket on port 1884.
1704214673: Opening ipv6 listen socket on port 1884.
1704214673: mosquitto version 2.0.11 running
```

راه‌اندازی اولیه بروکر

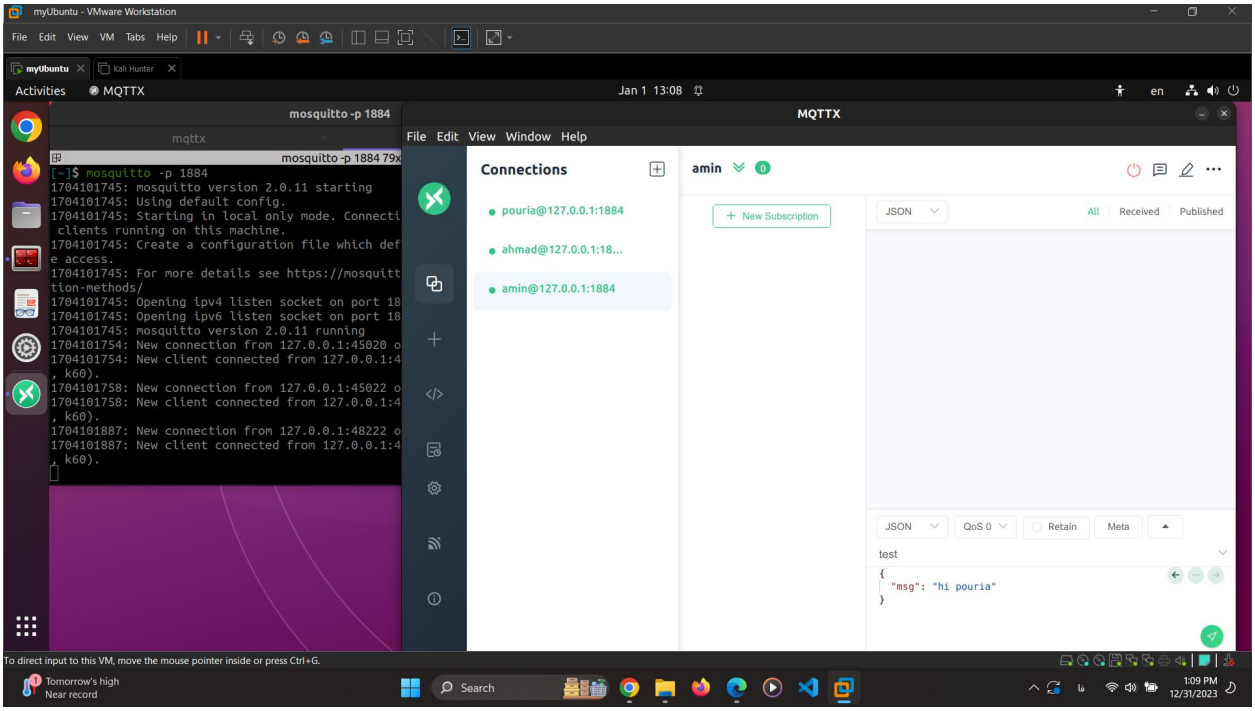


تعریف کاربر جدید و اتصال به بروکر





ارسال پیام در تاپیک test توسط Pouria و دریافت آن توسط Pouria و ahmad که این تاپیک را subscribe می‌کنند.



عدم دریافت پیام Pouria توسط amin چراکه این تاپیک را subscribeنمی‌کند

2. ارتقا امنیتی

Authentication 2.1

در این قسمت ویژگی allow_anonymous را برابر false قرار می‌دهیم و آدرس فایل کاربران و رمز عبورهای هش شده آن‌ها را به آن می‌دهیم. بنابراین فایل کانفیگ که آن را با نام auth.conf ذخیره می‌کنیم به صورت زیر خواهد بود که در پوشه نیز ضمیمه شده است:

```
port 1884

allow_anonymous false

password_file /etc/mosquitto/passwd
```

بنابراین باید تعدادی کاربر با رمز عبور مشخص برای بروکر تعریف کنیم. این کار را می‌توان دستی داخل فایل passwd مانند خط زیر برای هر کاربر اضافه کرد:

```
pouria:$7$101$RsGGaRjh2RKCM3VS$ej5SKF6VUvfIxIzcEUMTZlqKQMY/a8pI2B/U8
nSF9adOXM/HgkNCX0uLmwQofvpzsopiv0f1wCEsTWf+Au33HQ==
```

که Pouria نام کاربری و عبارت مقابل آن هش پسورد 123 است.

و یا میتوان با استفاده از دستور زیر کاربر را ایجاد کرد و پسورد مدنظر را به آن داد:

```
sudo mosquitto_passwd -c /etc/mosquitto/passwd username
```

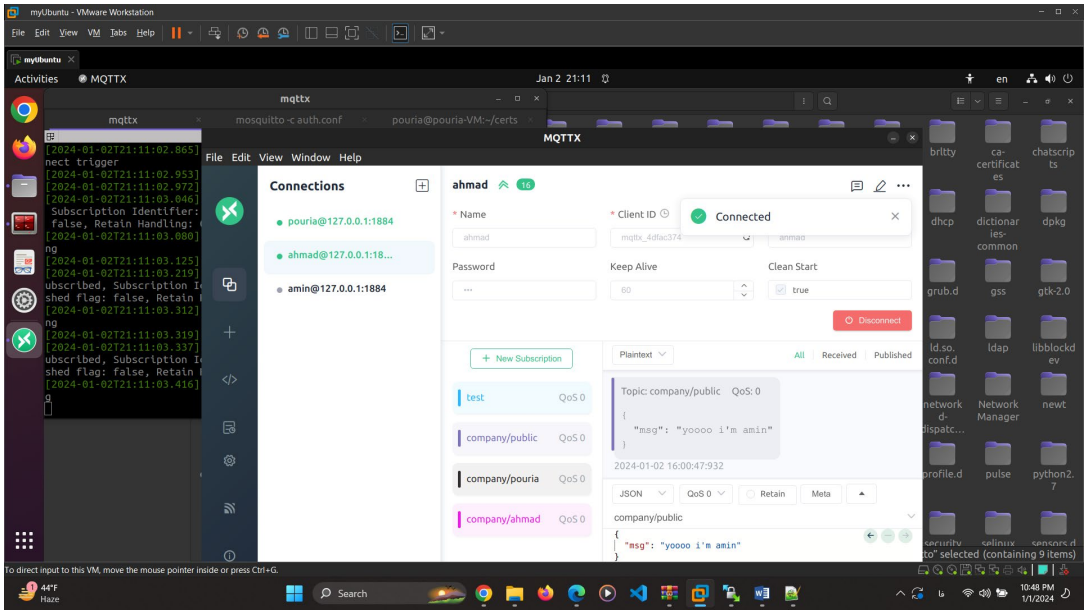
که به جای username ، نام کاربری مدنظر را قرار داده و در ادامه از ما password خواسته و هش آن را به فرمت قبلی گفته شده وارد فایل passwd می‌کند. (که البته این روش فایل فوق را overwrite میکرد و برای کاربرهای بعدی باید خودمان دستی اضافه می‌کردیم)

پس از اجرای mosquito با کانفیگ auth.conf تنها با استفاده از 3 فرد تعریف شده زیر در passwd می‌توان متصل شد و در غیر این صورت خطای unauthorized می‌دهد: (برای سادگی pass هر سه 123 است.)

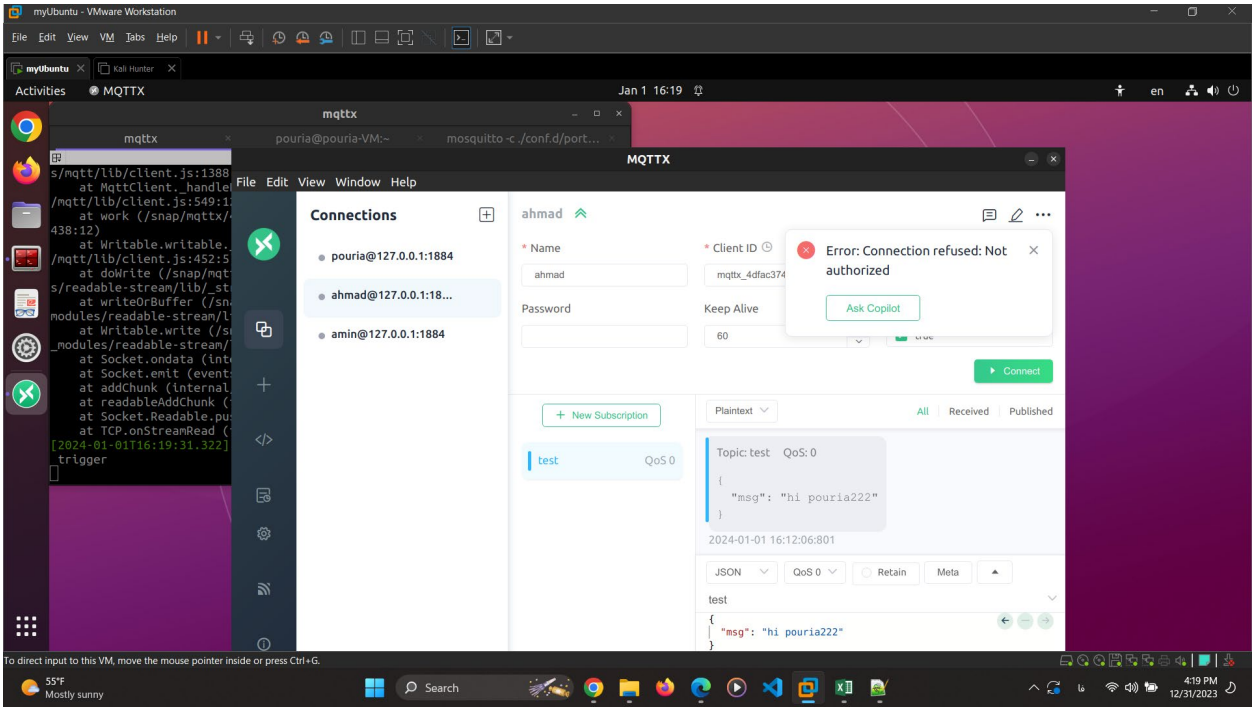
ahmad:$7$101$RsGGaRjh2RKCM3VS$ej5SKF6VUvfIxIzcEUMTZlqKQMY/a8pI2B/U8nSF9adOXM/HgkNCX0uLmwQofvpzsopiv0f1wCEsTWf+Au33HQ==

pouria:$7$101$RsGGaRjh2RKCM3VS$ej5SKF6VUvfIxIzcEUMTZlqKQMY/a8pI2B/U8nSF9adOXM/HgkNCX0uLmwQofvpzsopiv0f1wCEsTWf+Au33HQ==

amin:$7$101$RsGGaRjh2RKCM3VS$ej5SKF6VUvfIxIzcEUMTZlqKQMY/a8pI2B/U8nSF9adOXM/HgkNCX0uLmwQofvpzsopiv0f1wCEsTWf+Au33HQ==



اتصال موفق با ست کردن درست username password



عدم اتصال در صورت یوز و پس تعریف نشده

MQTTS 2.2

در این قسمت با استفاده از تعریف یک CA دستی و اختصاص دادن یک certification و key به سرور (بروکر) و client، ارتباط خود را روی TLS پیاده می‌کنم. الگوریتم رمزی که کلیدهای خود را می‌سازیم rsa انتخاب می‌کنیم.

ابتدا توسط دستورات زیر CA را تعریف می‌کنیم که مرجع certificate خواهد بود و سرور و کلاینت درخواست اعتبار خود را از آن می‌سازند: (در ضمن چون از قبل openssl را نصب داشتیم نیاز به نصب مجدد آن نداشتیم وگرنه بایستی ابتدا این کتابخانه را نصب کنیم)

*در ضمن توجه می‌کنیم که در مراحل ساخت کلیدها پارامترهای دلخواهی باید تنظیم کنیم (برای CA, client , server) و تنها common name آن مهم است که باید common name سرور و کلاینت متفاوت باشد. در ضمن بسته به ورژن ssl ممکن است نیاز باشد که cn مربوط به سرور را برابر ip بروکر تعریف کنیم تا به درستی verify شود.

```
openssl genrsa -out ca.key 2048 → تعریف کلید خصوصی cert
```

```
openssl req -new -x509 -days 3650 -key ca.key -out ca.crt → ایجاد ریشه cert با کلید خصوصی
```

اکنون کلید خصوصی سرور و سپس ایجاد درخواست امضای آن توسط ca را با دستورات زیر اجرا می‌کنیم و در اخر اعتبار سرور را با امضای csr آن ایجاد می‌کنیم.

```
openssl genrsa -out server.key 2048

openssl req -new -key server.key -out server.csr
```

```
openssl x509 -req -in server.csr -CA ca.crt -CAkey ca.key -CAcreateserial -
out server.crt -days 3650
```

چناچه به درستی اجرا شود این خروجی را خواهیم داشت:

```
[~/certs]$ openssl x509 -req -in server.csr -CA ca.crt -CAkey ca.key -
CAcreateserial -out server.crt -days 3650

Certificate request self-signature ok

subject=C = AU, ST = Some-State, O = Internet Widgits Pty Ltd, CN = 127.0.0.1
```

سپس همین مراحل را برای client نیز طی می‌کنیم:

```
openssl genrsa -out client.key 2048
openssl req -new -key client.key -out client.csr
```

```
openssl x509 -req -in client.csr -CA ca.crt -CAkey ca.key -CAcreateserial -
out client.crt -days 3650
```

اکنون که این فایل‌های cert را ایجاد کردیم کافی است در کانفیگ mosquito استفاده از ssl را فعال کرده و ادرس ذخیره فایل‌های اعتبار ca و کلید و امضای سرور را برای بروکر قرار دهیم. در ضمن پورت مورد استفاده در mqtt برای ssl نیز 8883 بوده و آن را از 1884 قبلی خارج می‌کنیم. بنابراین فایل کانفیگ mqqts.conf به صورت درخواهد آمد که در پوشه تمرین نیز ضمیمه شده است:

allow_anonymous false

password_file /etc/mosquitto/passwd

listener 8883

cafile /home/pouria/certs/ca.crt

certfile /home/pouria/certs/server.crt

keyfile /home/pouria/certs/server.key

require_certificate true

پس از اجرای دســـتورات فوق در linux خود، به ارور slef certified خوردیم و با تلاش برای رفع آن به این نتیجه ر سیدیم که ظاهرا ورژن ا ستفاده شده از openssl م شکلی برای اجرای این نوع از cert دارد و به دلیل ذیق وقت از ن صب مجدد و کانفیگ در linux خودداری کردیم و از آن‌جایی که ورژن متنا سبی در windows داشتیم این قسمت را عینا مطابق مراحل ذکر شده در بالا روی windows هم پیاده کردیم. فقط از آنجایی که mqttx روی ویندوز نصب نداشتیم با cmd کلاینت تعریف کردیم و نتیجه sub و pub را در ادامه می‌بینیم:

mosquitto_sub -h localhost -p 8883 -t Test_Topic_3 -u User1 -P 123 --cafile "C:\Program Files\mosquitto\certs\ca\ca.crt" --cert "C:\Program Files\mosquitto\certs\client\client.crt" --key "C:\Program Files\mosquitto\certs\client\client.key" –insecure

mosquitto_pub -h localhost -p 8883 -t Test_Topic_3 -m "This is a secure message!" -u User2 -P 456 --cafile "C:\Program Files\mosquitto\certs\ca\ca.crt" --cert "C:\Program Files\mosquitto\certs\client\client.crt" --key "C:\Program Files\mosquitto\certs\client\client.key" –insecure

```
C:\Program Files\mosquitto>mosquitto_pub -h localhost -p 8883 -t Test_Topic_3 -m "This is a secu
re message!" -u User2 -P 456 --cafile "C:\Program Files\mosquitto\certs\ca\ca.crt" --cert "C:\Pr
ogram Files\mosquitto\certs\client\client.crt" --key "C:\Program Files\mosquitto\certs\client\cl
ient.key" --insecure
```

انتشار پیام رمز شده

```
C:\Program Files\mosquitto>mosquitto_sub -h localhost -p 8883 -t Test_Topic_3 -u User1 -P 123 --
cafile "C:\Program Files\mosquitto\certs\ca\ca.crt" --cert "C:\Program Files\mosquitto\certs\cli
ent\client.crt" --key "C:\Program Files\mosquitto\certs\client\client.key" --insecure
This is a secure message!
```

Subscribe رمز شده

ACL 2.3

در این بخش باید سطح دسترسی‌های خواسته شده را در یک فایل acl.acl ایجاد کنیم که سطح‌های خواسته شده تعریف شده باشد. سپس ادرس این فایل را به acl_file داخل فایل کانفیگ بروکر بدهیم.

با توجه به کاربران تعریف شده قبلی (Pouria, ahmad, amin) اتاق های خصوصیشان و اتاق public را تعریف می کنیم:

acl.acl #

Public topic that all users can subscribe to #

pattern readwrite company/public/#

User "pouria" permissions #

user pouria

topic readwrite company/pouria/#

User "ahmad" permissions #

user ahmad

topic readwrite company/ahmad/#

User "amin" permissions #

user amin

topic readwrite company/amin/#

بنابراین همه امکان خواندن و نوشتن در تاپیک company/public را دارند ولی خواندن و نوشتن در تاپیکهای خصوصی فقط برای صاحب ان است.

فایل کانفیگ acl.conf هم به صورت زیر در می‌آید:

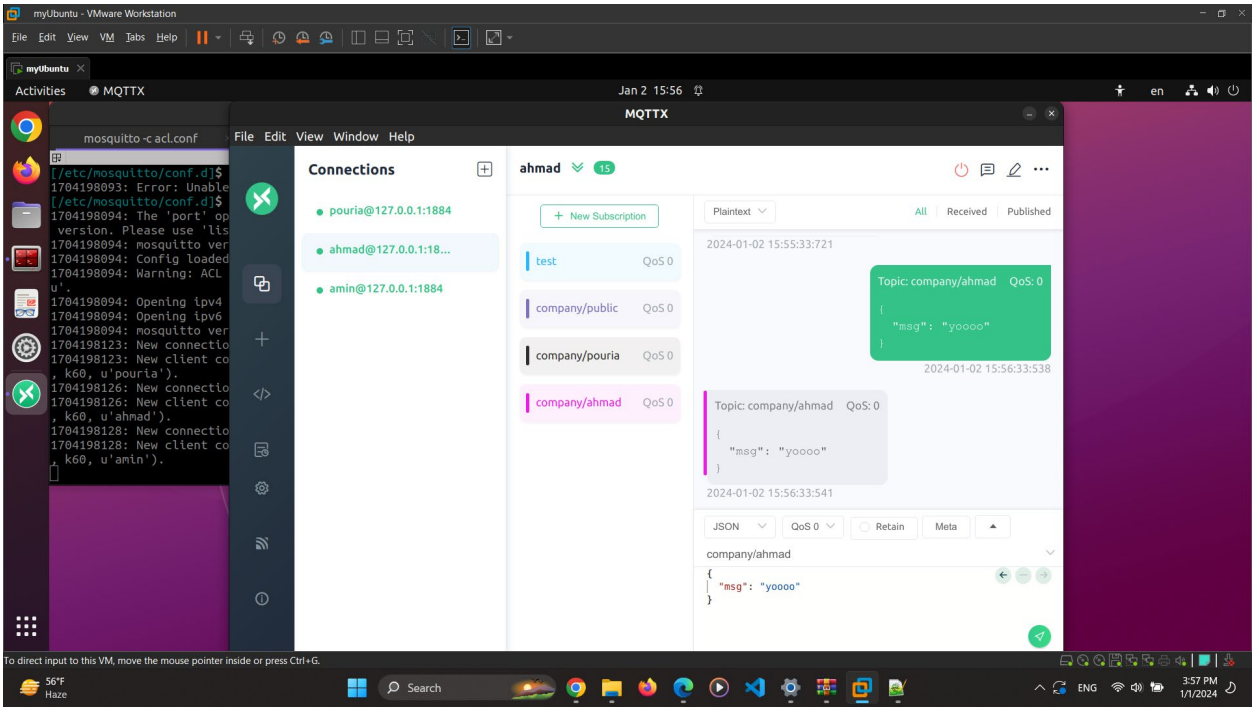
port 1884

allow_anonymous false

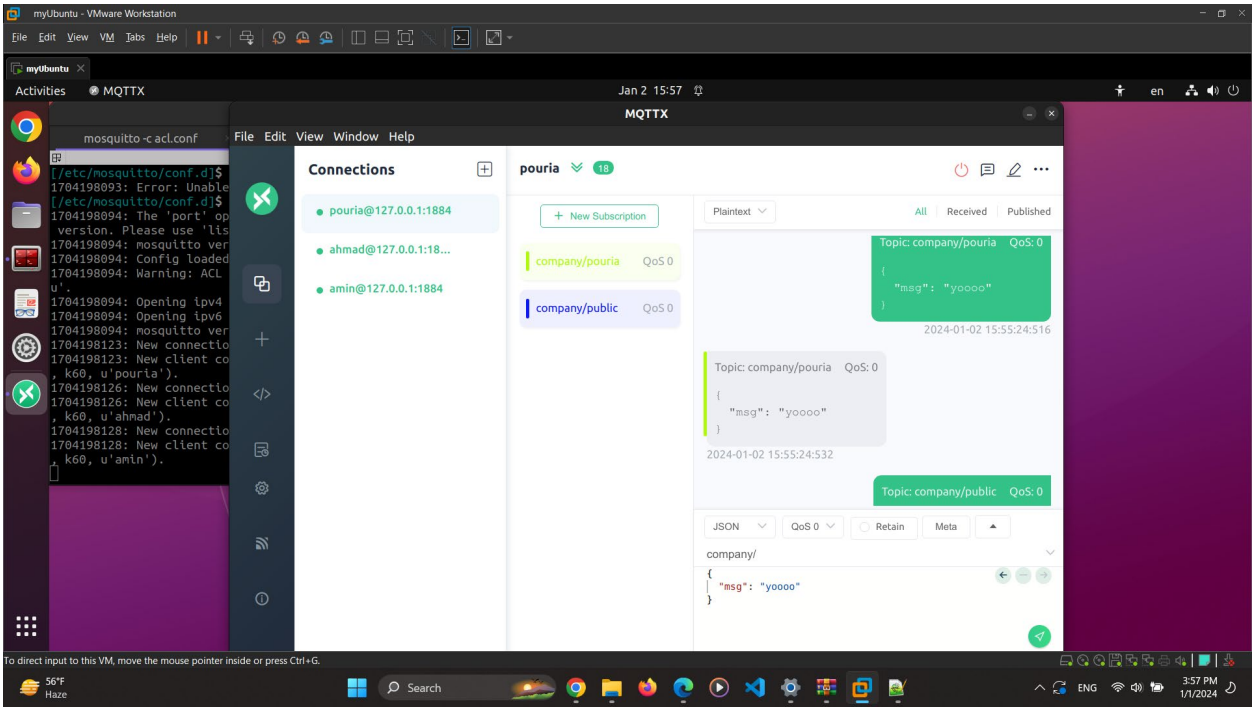
password_file /etc/mosquitto/passwd

acl_file /etc/mosquitto/acl.acl

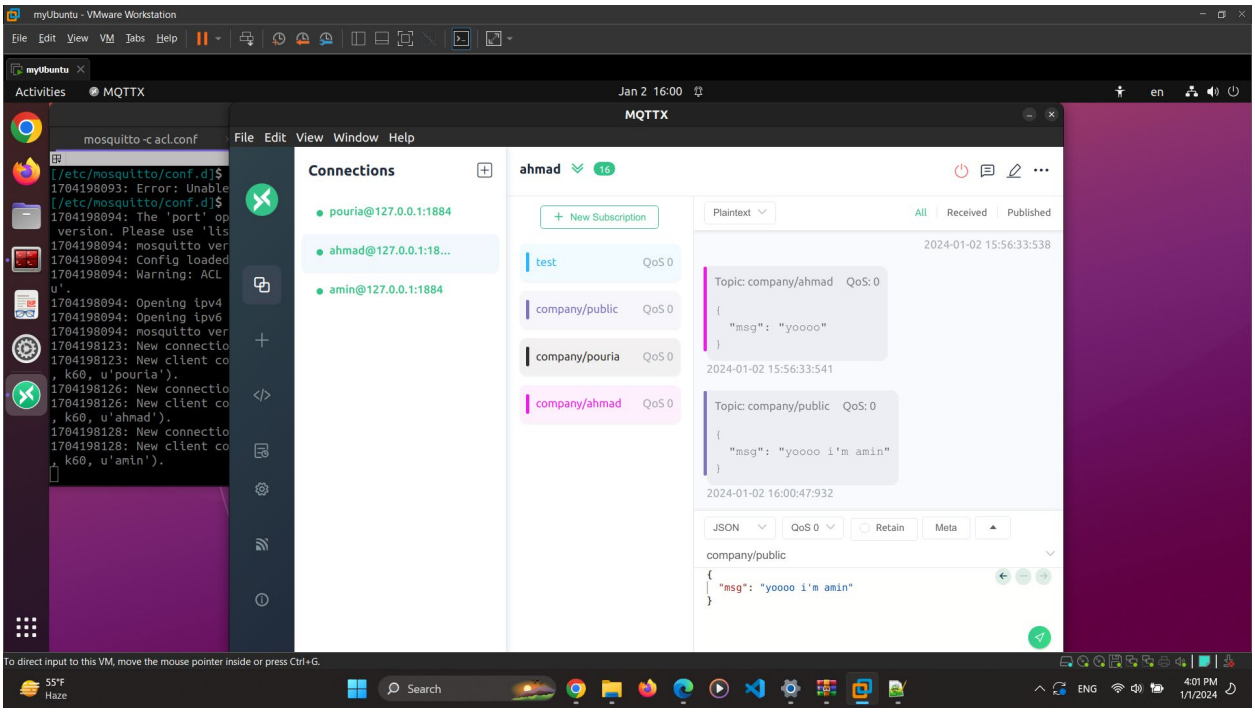
چند نمونه از اجرای این کانفیگ را در ادامه می‌بینیم:



همان‌طور که می‌بینیم، ahmad در تاپیک‌های Pouria , ahmad , company/public سابسکرایب کرده‌ا ست و پیام‌های مربوط به ahmad و public را دریافت کرده ولی پیامی از Pouria نمی‌تواند بخواند یا بنویسد. (در عکس بعدی پیام‌های تاپیک Pouria مشخص است)



پوریا پیام‌های public و pouria را می‌تواند بنویسد و بخواند.



Ahmad پیام منتشر شده توسط amin در تاپیک public را میتواند بخواند.

بنابراین سطح دسترسی خواسته شده مطابق انتظار عمل می‌کند و هرکس در سطح اختیار خود دسترسی به تغییر یا شنود داده‌ها را خواهد داشت.