**《安全编程》作品报告**

题目： MQTT协议逆向及模拟

班级： 网络空间安全2022级

姓名： 王际翔

教师： 张恒汝

2024年12月17日

目录

**[《安全编程》作品报告 I](#_Toc3899)**

**[1 方案与实现步骤 1](#_Toc13936)**

[1.1 MQTT协议提前了解 1](#_Toc14897)

[1.2 捕获与分析MQTT数据包 2](#_Toc383)

**[2 总体设计 3](#_Toc14291)**

[2.1 方案设计 3](#_Toc9290)

[2.2 协议的模拟与实现 3](#_Toc21502)

[2.3 实现结果 6](#_Toc15173)

**[3 总结 7](#_Toc24887)**

[3.1 项目目标与实现 7](#_Toc24382)

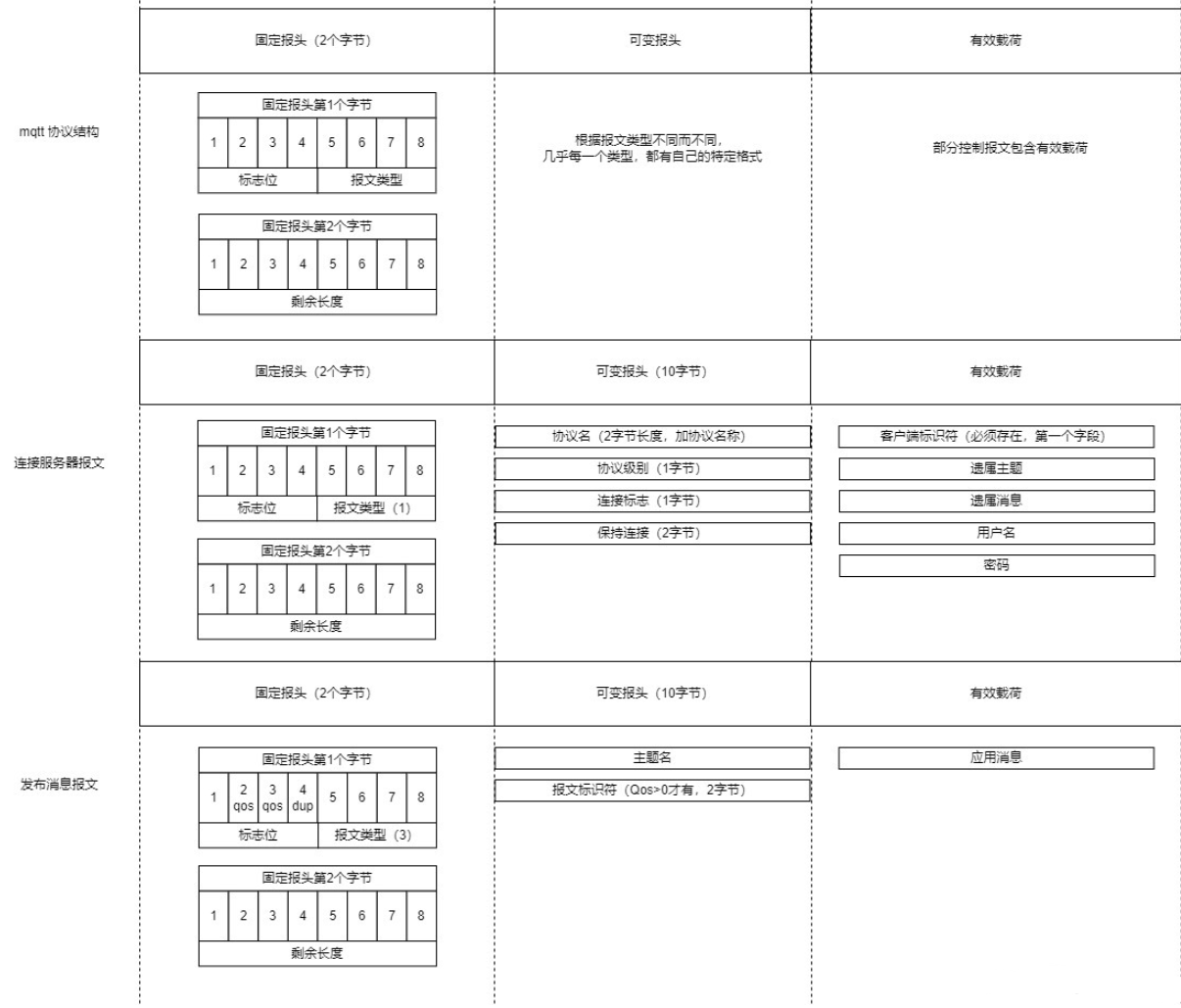
[3.2 开发心得 8](#_Toc19463)

# 1 方案与实现步骤

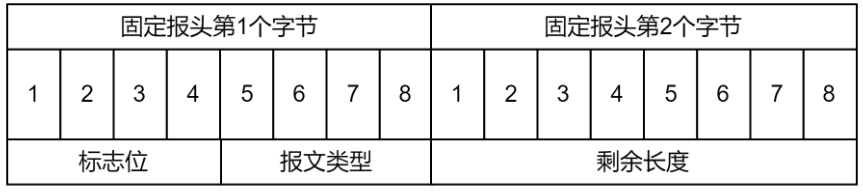
## 1.1 MQTT协议提前了解

MQTT协议在IoT领域是使用的最广泛的通用协议。MQTT是一种基于发布/订阅（publish/subscribe）模式的轻量级通讯协议，该协议构建于TCP/IP协议上，由IBM在1999年发布。MQTT在官方解释中的优点是：用极少的代码和有限的带宽，为连接远程设备提供实时可靠的消息服务；简单理解就是，实现起来简单，并且在传输上无效的数据也很少，并且能够保证数据传输的可靠性。

MQTT协议由三部分组成：固定报头、可变报头、有效载荷。固定报头是所有报文的统一格式；可变报头则根据固定报头中的报文类型不同而改变，每个报文类型基本上都有自己的可变报头格式；有效载荷只是部分报文有。报文内容是根据报文类型的不同而不同。

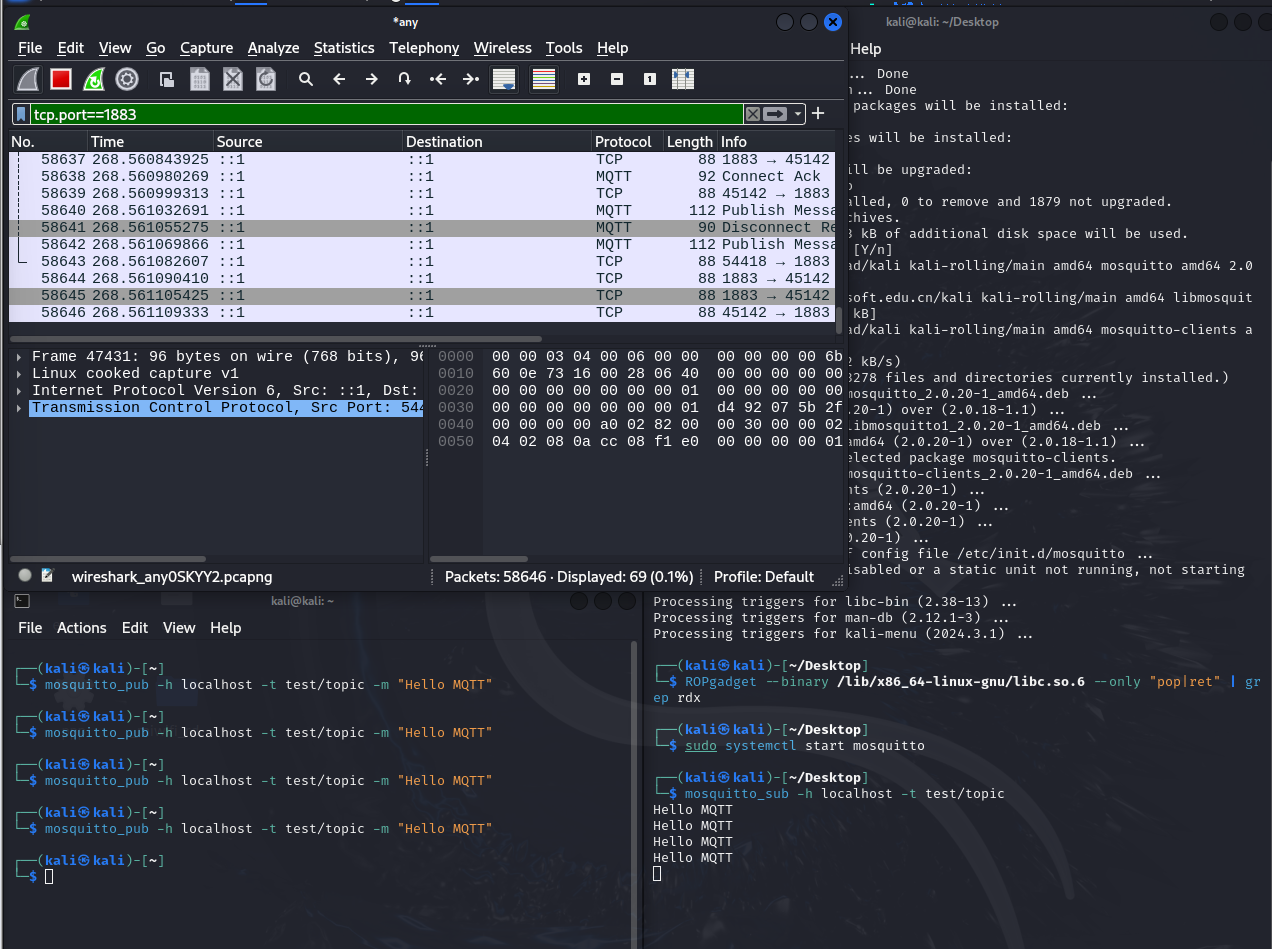


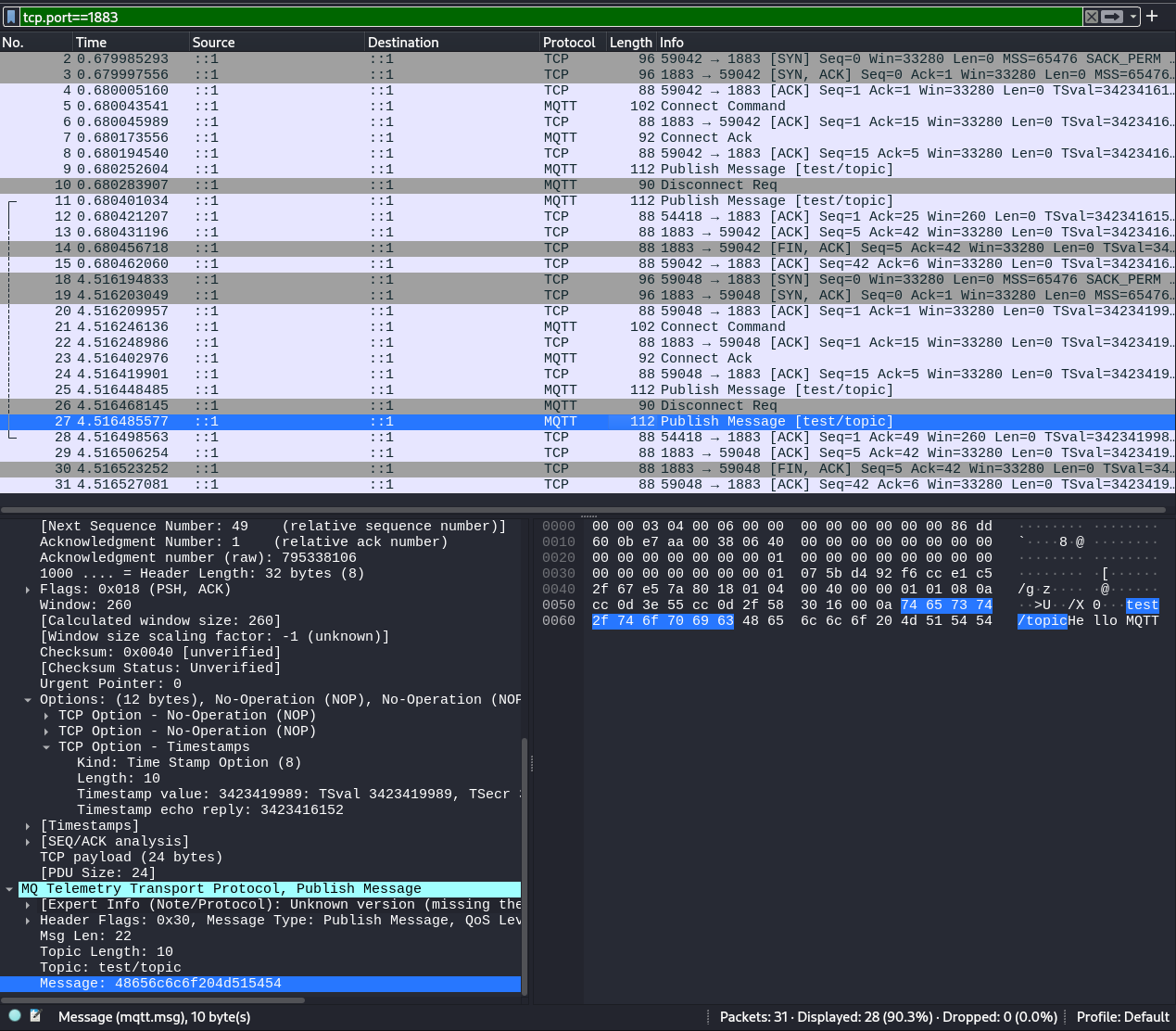
固定报头：一共两个字节



## 1.2 捕获与分析MQTT数据包

首先安装对应包体，通过订阅对应ip的对应主题，再在其他地方向对应ip的对应主题发布消息，便会在订阅端接收到对应的消息。MQTT默认端口为1883，便可以使用wireshark监听对应端口的数据包。





前面提到MQTT是构建于TCP/IP协议上的，前面0x58(88)个字符并非MQTT数据包部分，接下来才是MQTT数据包部分。这里的数据包固定头部为0x30 0x16，其中剩余长度为0x16；接下来是可变头部，主题名长度为0x000a，主题名为test/topic；接下来是负载数据，内容为Hello MQTT。

# 2 总体设计

在分析了数据包后，我们开始编写程序来模拟MQTT协议的行为。

## 2.1 方案设计

我选择使用Java编写一个跨平台的MQTT客户端，且方便封装成jar包复用。我使用Maven管理依赖，引入了Paho MQTT库来实现MQTT客户端。

我希望实现一个可以反复向对应ip:port的对应主题发布消息或接受消息的客户端

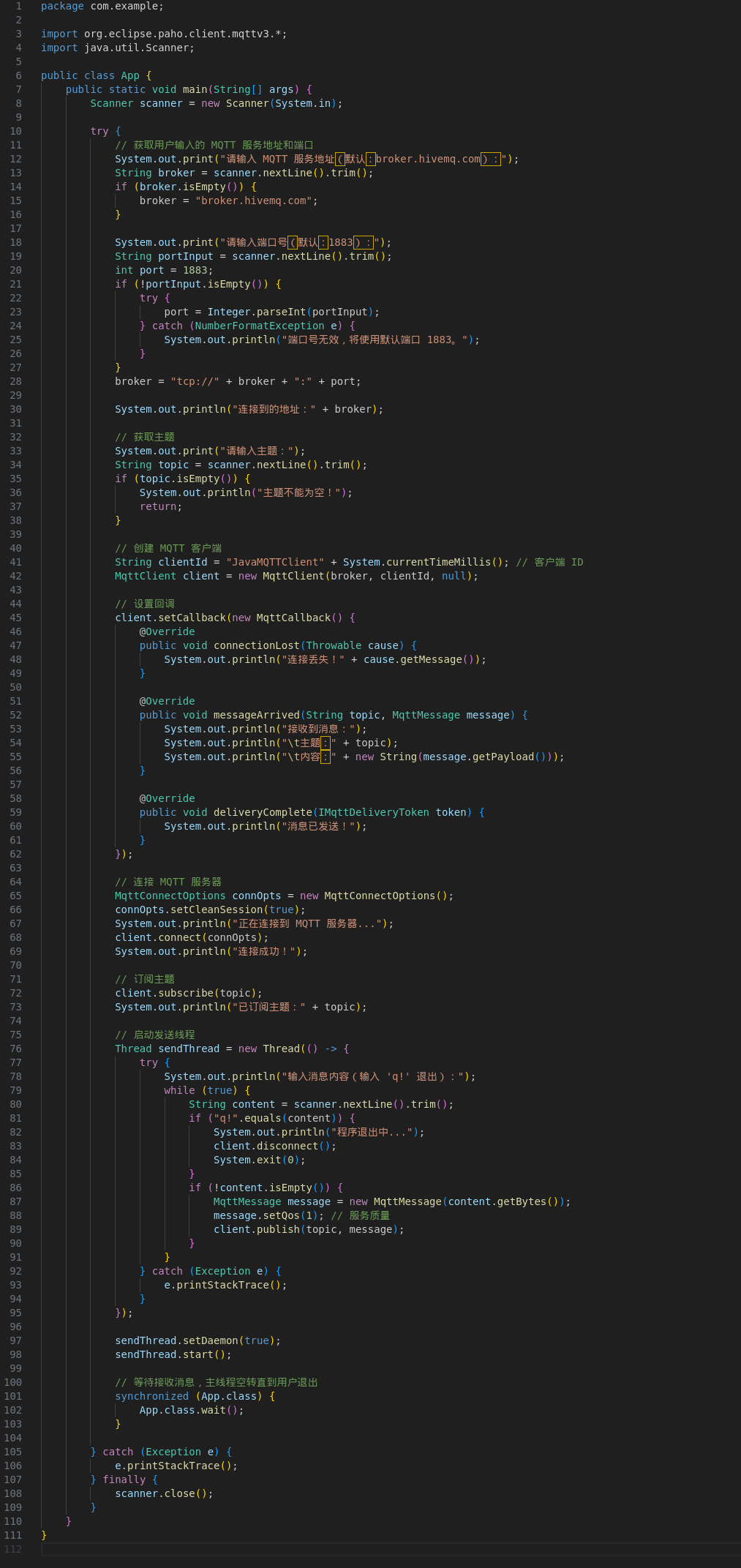
当然执行的前提是你需要一个已有的监听于1883端口的MQTT客户端，例如说mosquitto或者hivemq。

## 2.2 协议的模拟与实现

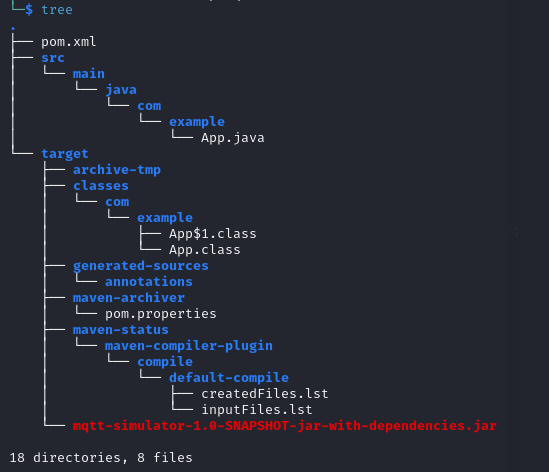
首先是pom.xml:



然后根据需求写出代码：



Tree



### 2.3 实现结果



# 3 总结

## 3.1 项目目标与实现

本项目的核心目标是开发一个 MQTT 客户端模拟器，支持基本的 MQTT 消息发布和接收功能。实现的主要功能包括：

* 支持用户手动输入 MQTT 服务地址和端口，默认为 broker.hivemq.com 和 1883。
* 支持自定义客户端 ID 和消息主题，能够动态发送和接收消息。
* 通过 MQTT 协议与服务器建立连接，进行消息的订阅和发布。
* 通过处理 Ctrl+C 或输入 q! 实现优雅的客户端退出。

项目基于 Java 语言开发，并使用了 org.eclipse.paho.client.mqttv3 库来实现 MQTT 协议的客户端功能。最终，项目成功实现了命令行界面的 MQTT 客户端模拟器，能够连接到指定的 MQTT 服务器，并进行消息通信。

## 3.2 开发心得

* 在项目中，我深入学习并应用了 MQTT 协议的基本原理，理解了消息发布、订阅和客户端连接的机制。
* 用 Java 语言和 MQTT 库，熟悉了如何通过编程实现协议的底层操作。
* 遇到的问题主要集中在客户端连接管理和跨平台兼容性上。通过调试和配置，成功解决了连接丢失、消息过滤等技术难题。