[Emitter: Scalable Real-Time Communication Across Devices]

[emitter-io/emitter: High performance, distributed and low latency publish-subscribe platform. (github.com)]

注：我们已向厂商通报此安全问题及修复建议

0x01 攻击场景与测试

考虑IoT应用的共享场景，即智能家居系统使用 MQTT 协议进行物联网设备和用户管理，其中有两个用户角色。管理员，也就是房主
可以授权其他普通用户（例如，Airbnb 客人）访问他的智能家居设备的权利。普通用户的访问权限可能会被撤销和到期。我们
认为管理员和设备是良性的，而客人可能是恶意的，会尽可能地去试图未授权访问设备（越权或是维持被撤销的权限）。

- **攻击场景**

首先，攻击者登记入住，因此目前攻击者拥有主题/通道“test”的 "read" 权限，可以暂时收到来自 "test" 的消息。

1. 攻击者订阅channel "test"

```
    emitter.subscribe({
        key: "<channel key>",
        channel: "test"
    });
```

2. 攻击者权限被撤销(`keyban`)

3. 攻击者保持连接不断开

4. 攻击者能够继续收到来自 "test" 的消息

- **漏洞危害**

攻击者可以在退房后继续监听设备消息或是接收其他敏感信息。

0x02 漏洞测试步骤

- **测试环境**

Emitter：3.0

mqtt client：[https://github.com/emitter-io/python]

访问控制: 内置 `keygen`，`keyban` （在doc中未描述 `keyban`，但在broker中已实现）

[Message Filtering - Emitter Documentation, Samples and Tutorials]

配置测试用户：

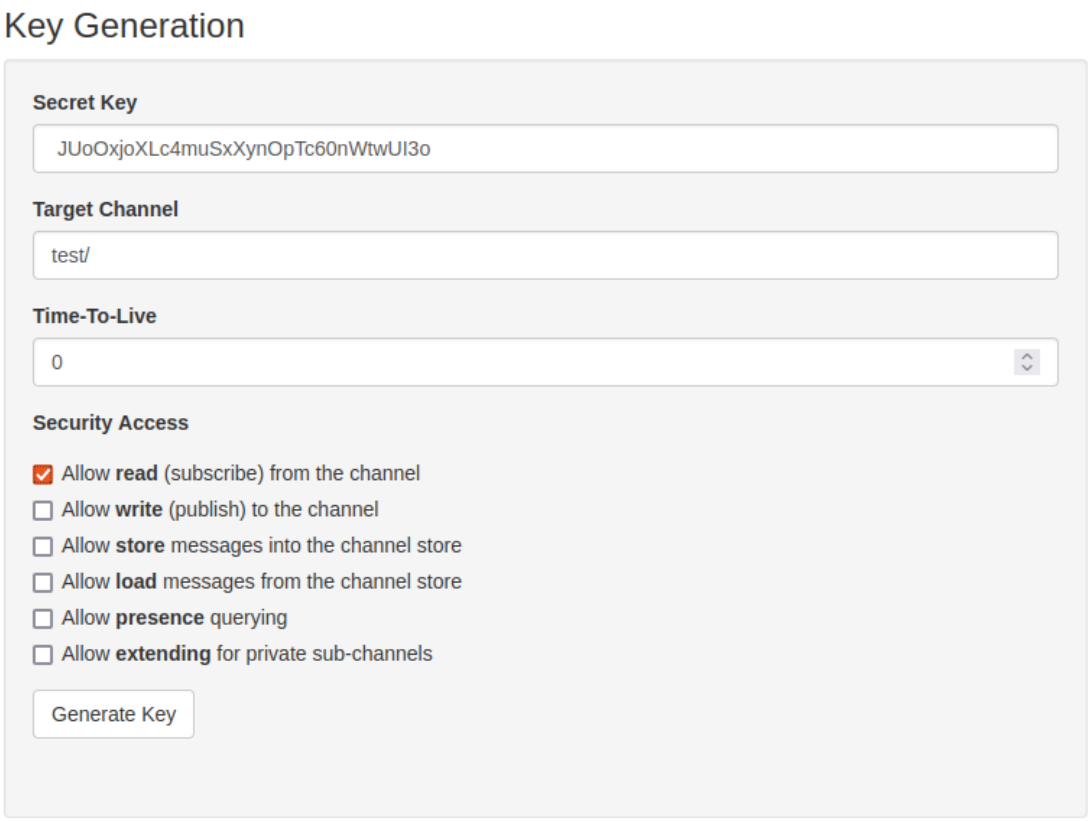
admin: 拥有所有权限

attacker: 拥有read权限

1. 首先按照文档指引，配置license (emitter-io/emitter: High performance, distributed and low latency publish-subscribe platform. (github.com))

```
[service] unable to find a license, make sure 'license' value is set in the config file or EMITTER_LICENSE environment variable
[service] generated new license: uppD0PFIcNK6VY-7PTo7uWH8EobaOGgRAAAAAAAAAAI
[service] generated new secret key: JUoOxjoXLc4muSxXynOpTc60nWtwUI3o
```

2. 重启后创建一个channel key，作为"test" topic的密钥，并分配 "read" 权限，分发给attacker，通过访问 **http://127.0.0.1:8080/keygen**



3. 同理创建"test" 通道另一个admin key，让其拥有所有权限。

4. 如何撤销权限

使用官方python sdk的 keyban 功能可能存在无法成果撤销的情况，可使用下面的脚本（clientID不重要，主要是配置secret key以及target key）

```
def pubKeyBan():
```

```
    client = mqtt.Client(client_id="user1123", protocol=mqtt.MQTTv311)
    #client.username_pw_set(username="user1", password="pass1")
    client.on_connect = connect_callback_v3

    try:
        conProperty = p.Properties(PacketTypes.PacketTypes.CONNECT)
        pubProperty = p.Properties(PacketTypes.PacketTypes.PUBLISH)
        client.connect(host="192.168.179.128", port=8080, keepalive=1000)
        client.loop_start()
        while (input() != "xxx"):
            request = {"secret": "D8qiQ2kOHD5b3xth2_xtOsCs7Mjl2aUZ", "target": "P1-
LbZvpKvYk7Fzu1nSXvT3Y20Rl5W09", "banned": False}
            client.publish(topic="emitter/keyban/", payload=json.dumps(request), qos=0, retain=True)
    except Exception as e:
        print(e)
        client.disconnect()
```

- **测试步骤**

1. 攻击者订阅channel "test"。

```
    emitter.subscribe({
        key: "<channel key>",
        channel: "test"
    });
```

2. 攻击者权限被撤销(keyban)。
3. 攻击者保持连接不断开。
4. 攻击者能够继续收到来自 "test" 的消息。

```
emitter.publish({
    key: "<channel key>",
    channel: "test",
    message: "hello, emitter!"
});
```