7ZC-03

MQTT ブローカに対する SlowDoS 攻撃対策の検討

砂田　拳人†　　鳥居　直哉‡

創価大学　理工学研究科情報システム工学専攻†

1. はじめに

近年，IoT 機器を用いたデータ監視などのアプリケーションに軽量な通信プロトコルである MQTT[1]がよく用いられている．MQTT プロトコルは，パブリッシュ・サブスクライブ・モデルを採用しており，パブリッシャはブローカへ，ブローカはパブリッシャからのデータを集約しサブスクライバに送信する．

本研究では，MQTT ブローカに対する各種攻撃を行い，対策を検討した．MQTT プロトコルの実装にはオープンソースである Mosquitto[2]を Raspberry Pi4[3]に実装して使用した．　特に，最近注目されている Dos 攻撃の一種である SlowDos 攻撃[4]とその対策を実装し評価した．その結果，クライアントのタイムアウトの値である KeepAlive の設定時間を短くとることで有効な対策となることを示した．

2. MQTT への攻撃

MQTT プロトコルへの攻撃は，各種提案されている．まず、MQTT プロトコルは，TCP/IP スタックをベースに作成されているため，IP スプーフィングによるパスワードやメッセージなどの情報を盗聴する MitM(Man in the Middle)攻撃が可能である．次に、リソースが限られた IoT 機器を用いてブローカを実装する場合，ブローカに過剰なアクセスやデータを送信し，負荷をかける DoS 攻撃が有効である。

最新の攻撃として SlowDos 攻撃がある．これは，攻撃検知しにくい低速のリクエストで接続可能な全てのスレッドを拘束することで，正規のユーザがサービスにアクセスできないようにする攻撃である．

本研究では，評価環境にてこれらの攻撃を行い Mosquitto で実装したブローカの耐性を評価した．

3. 評価環境

本研究の評価環境を図 1 に示す．本研究では，ブローカは Rasbwrry Pi4 に MQTT v3.1.1 の Mosquitto v1.5.7 を実装し，パブリッシャ、及びサブスクライバの認証は、パスワード認証で行う．パブリッシャは 2 台の Raspberry Pi4 に実装し，サブスクライバは 1 台のノート PC に実装した．

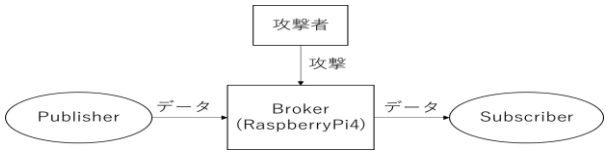

図 1　評価環境

4. 攻撃評価

4.1 MitM 攻撃

MitM 攻撃は、Bettercap[5]を使用して IP スプーフィングを行ない，ブローカとサブスクライバの間で盗聴を行い，サブスクライバの認証に使用している ID/Password を盗聴した．盗聴したパケットを図 2 に示す．図 2 には使用している ID/Password である steve/password が表示されている．

MitM 攻撃の対策としては，通信の暗号化が考えられるが，Mosquitto はデフォルトでは暗号化されておらず，TLS を使用できるが，IoT 機器ではリソースを大きく使用する TLS による暗号化は用いられない場合も多い．MQTT v5[6]で可能となった，チャレンジレスポンス認証が有効となると考えられる．

4.2 DoS 攻撃

DoS 攻撃は hping3[7]を使用して評価を行った．正常通信として 1 秒間に一回パブリッシャからデータを送信し，10 個のデータをサブスクライバが受け取る時間を計測する．システムが正常ならば約 10 秒の結果がでる．

図 3 に横軸を送信するパケット数、縦軸をサブスクライバが 10 個のデータを受信する時間と

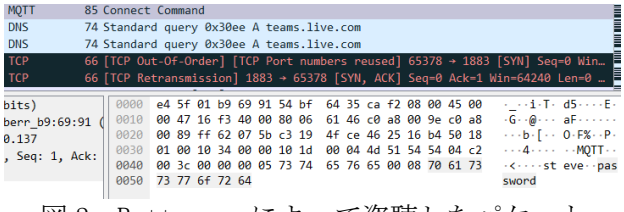

図 2　Bettercap によって盗聴したパケット

した場合の攻撃結果を示す．図 3 より，1 秒間に

Consideration of SlowDoS attack countermeasures against MQTT brokers
†Kento SUNADA, SOKA University
‡Naoya TORII, SOKA University

3000 パケットまでは受信時間が約 10 秒になっており，4000 を超えると正常な通信を行えなくなることが分かる．

対策として Raspberry OS にデフォルトで実装されているファイアウォール ufw を使用して指定したパブリッシャの IP 以外の通信を遮断する．この結果も図 3 に示す．一秒間に 20000 パケットを送信しても受信時間が約 10 秒となっており、DoS 攻撃を遮断できている．ただし，パブリッシャが乗っ取られた場合は，IDS/IPS などを使用して，パケット数での通信遮断が必要となる．今回の実験環境では一秒間に 4000 パケット超えると遮断するのが良いと考えられる．

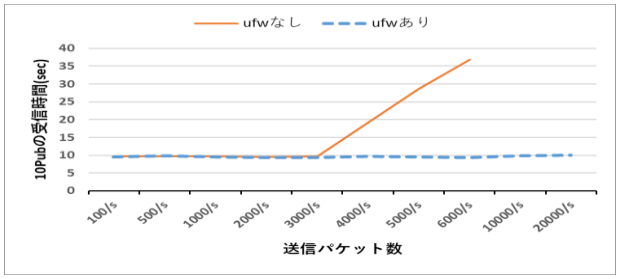

図 3　DoS 攻撃の結果

4.3　SlowDoS 攻撃

Ivan ら[4]は，MQTT に対する SlowDoS 攻撃を行うツールである SlowITe[8]を実装し攻撃に成功している．この SlowITe を使用して攻撃評価を行う．SlowITe はブローカに接続できる全ての接続を低速リクエストで埋めることで正規の通信を遮断している．Mosquitto の場合、最大接続数は 1024 である．また，Mosquitto ではクライアント側からタイムアウトの値である KeepAlive を指定できるため，攻撃時間が変わる．Mosquitto は，KeepAlive で指定された値の 1.5 倍の時間で接続するので，10 秒で指定すると 15 秒間攻撃が持続する．攻撃実験は，4.2 で説明した DoS 攻撃と同様の条件で行う．図 4 に横軸に KeepAlive の時間を取り縦軸にデータの受信時間をとった場合の攻撃結果を示す．受信時間は，正常通信の 10 秒に KeepAlive の 1.5 倍の時間を加えた値になっており，この間の攻撃によりシステムが停止していることが分かる．

この攻撃の対策として，ブローカ側で KeepAlive の上限値を設定することで攻撃によるサービス停止時間を最小限にでき，Mosquitto で KeepAlive の値を設定するにはソースの変更が必要となる．また，ufw を用いて同じ送信元からの同時アクセス数制限することも有効である．図 4 に ufw で同時接続数を 512 に制限した状態での

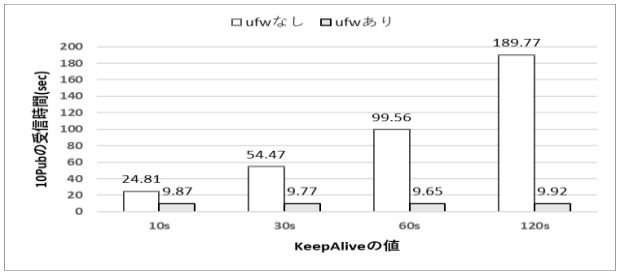

図 4　SlowDoS 攻撃の結果

SlowITe を実行した結果も示す．攻撃による影響がなくなり受信時間は 10 秒となる．ただし，この対策のためには，接続可能な端末の選択や切り替えが必要となる．

5.　まとめ

Mosquitto に対して各種攻撃を行い，評価を行った．MitM 攻撃では，最新の MQTT で採用されたチャレンジアンドレスポンス認証の採用する対策が有効である．DoS 攻撃は ufw によるホワイトリストが有効である．SlowDoS 攻撃は，KeepAlive の上限値をブローカ側で設定できるのであれば攻撃を最小限にでき，ufw で同時アクセス数に制限をかけることでも防ぐ事が可能である．更に，本攻撃の根本的な対策は，パブリッシャの認証の強化である．KeepAlive の時間の統計評価［4］による不正検知やブローカでパスワード認証のチャレンジアンドレスポンス認証が必要となる．

参考文献

[1] MQTT v3.1.1 https://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html
[2] Mosquitto https://github.com/eclipse/mosquito
[3] RaspberriPi4 https://www.raspberrypi.com/products/raspberry-pi-4-model-b/
[4] Ivan Vaccari, Maurizio Aiello and Enrico Cambiaso, “SlowITe, a Novel Denial of Service Attack Affecting MQTT,”*Sensors* **2020**, *20*,2932.,2022
[5] Bettercap https://charlesreid1.com/wiki/Bettercap
[6] MQTT v5.0 https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html
[7] hping3 https://github.com/HiddenShot/Hping3
[8] SlowITe https://github.com/fcarli3/MQTT_DoS_attack