# AVALIAÇÃO DA INTEGRAÇÃO DO PROTOCOLO MQTT AO INTERSCITY

Mechanisms To Improve Security in IoT Platforms

Bruno Carneiro da Cunha

Orientador: Daniel Batista

## INTRODUÇÃO

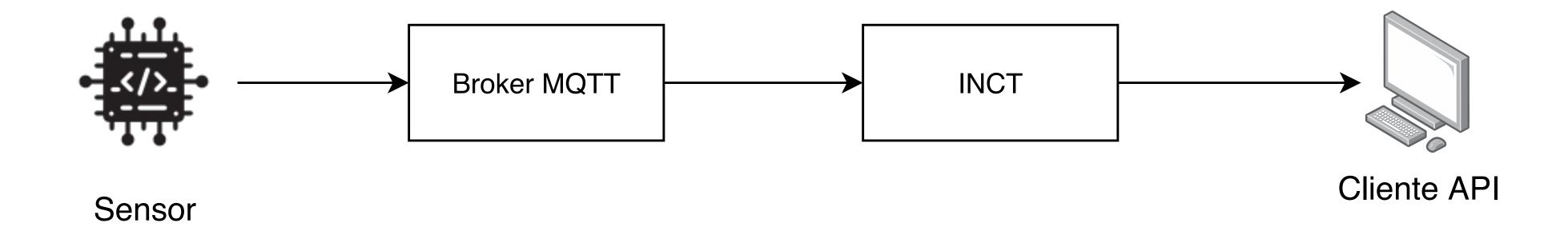
- Apresentação anterior no grupo
- MQTT
- Objetivo da integração

Client A Broker Client B CONNECT **PUBLISH** CONNACK temperature/roof 25 25 °C √ retain SUBSCRIBE temperature/roof **PUBLISH** temperature/roof 25 25 °C

## INTRODUÇÃO

- Apresentação anterior no grupo
- MQTT
- Objetivo da integração

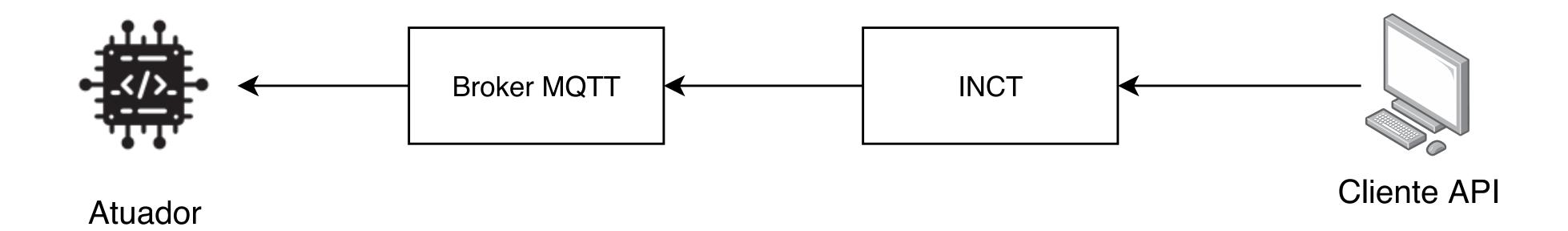
## CENÁRIO - SENSOR



#### SENSOR - EXEMPLO DE MENSAGEM

```
"data":
   "temperature": 10,
   "humidity": 45,
   "pressure": 25,
    "timestamp": "2017-06-14T17:52:25.428Z"
```

### CENÁRIO - ATUADOR



#### ATUADOR - EXEMPLO DE MENSAGEM

```
"data":
   "uuid": 606bd-cc1e3d610652a7c-bbbb,
    "capabilities": {
      "illuminate": "on"
```

#### ARQUITETURA DO INTERSCITY

resource-adaptor data-collector actuator-controller resource-cataloguer resource-discoverer É o principal ponto de interação com os Permite o envio Armazena os Mantém o catálogo Fornece um serviço de comandos aos dados coletados gateways IoT, de pesquisa de de todos os permitindo o dispositivos pelos dispositivos resources resources atuadores cadastro e sensores atualização de novos resources Kong API Gateway Recebe os requests, e os redireciona para o microserviço adequado

## MODIFICAÇÕES NO RESOURCE ADAPTOR

- Dois novos workers:
  - mqtt\_subscriber.rb
  - mqtt\_publisher.rb

#### MQTT SUBSCRIBER

API

MQTT

/adaptor/resources/{uuid}/data

/adaptor/resources/{uuid}/data/{capability}

resources/{uuid}

resources/{uuid}/{capability}

#### MQTT PUBLISHER

HTTP
MQTT

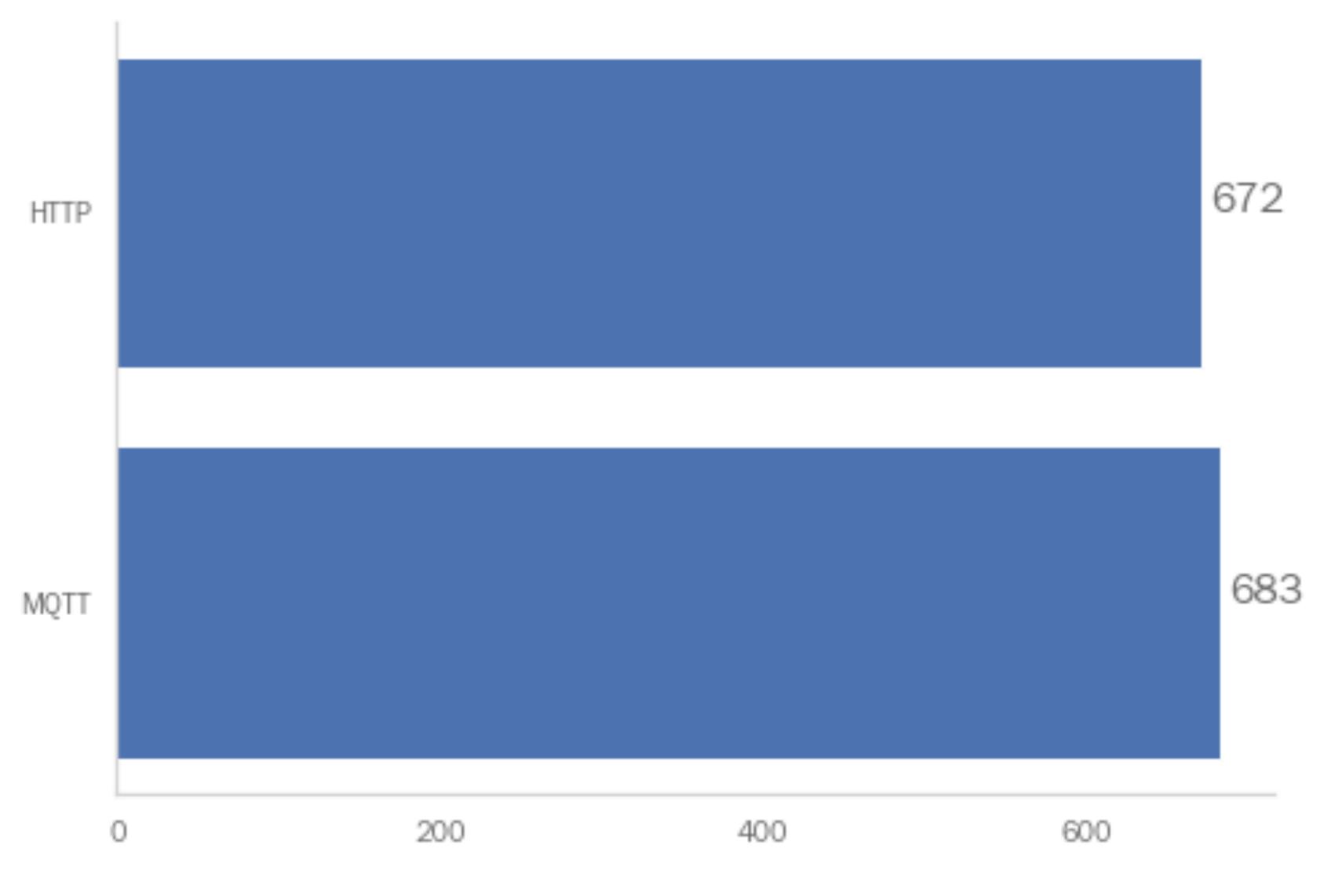
Enviadas diretamente ao webhook cadastrado

commands/{uuid}

## AVALIAÇÃO

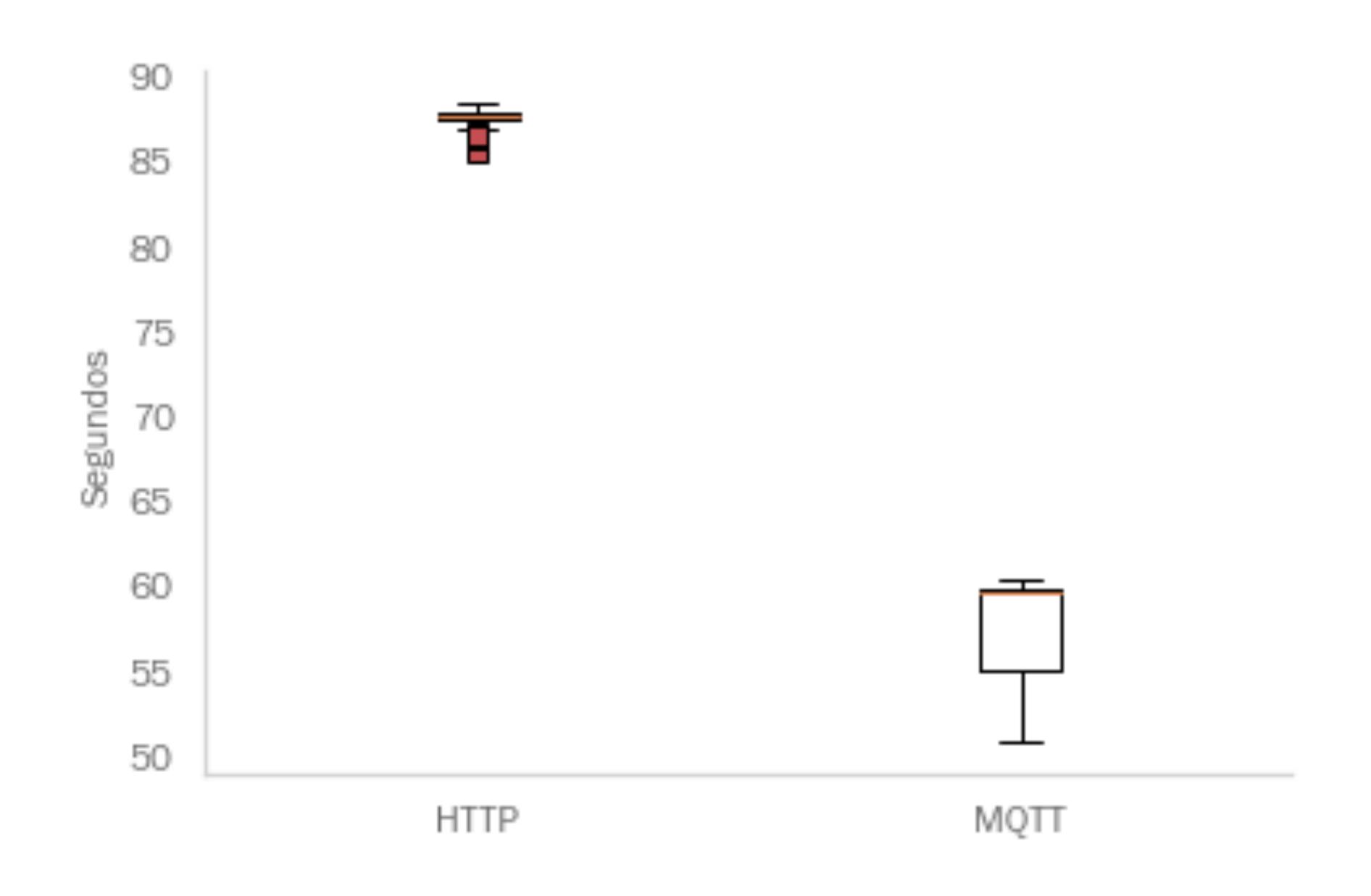
- Tamanho de uma mensagem
- Tempo para envio de um fluxo de dados do sensor
- Latência para envio de um comando a um atuador
- Uso de CPU e memória

#### RESULTADO 1 – TAMANHO DA MENSAGEM

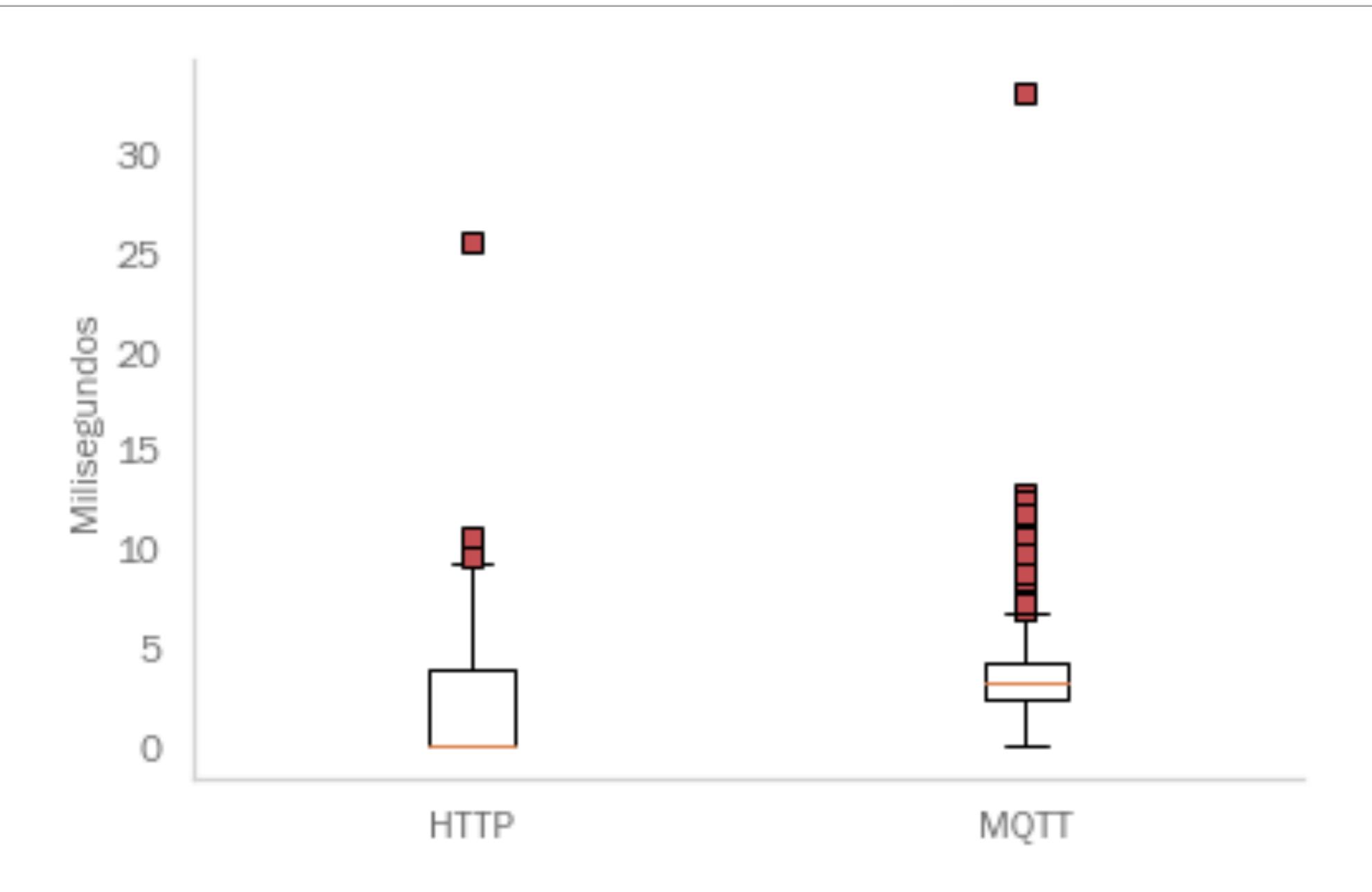


Número total de bytes

#### RESULTADO 2 - TEMPO PARA ENVIO DE UM FLUXO DE DADOS DE SENSOR

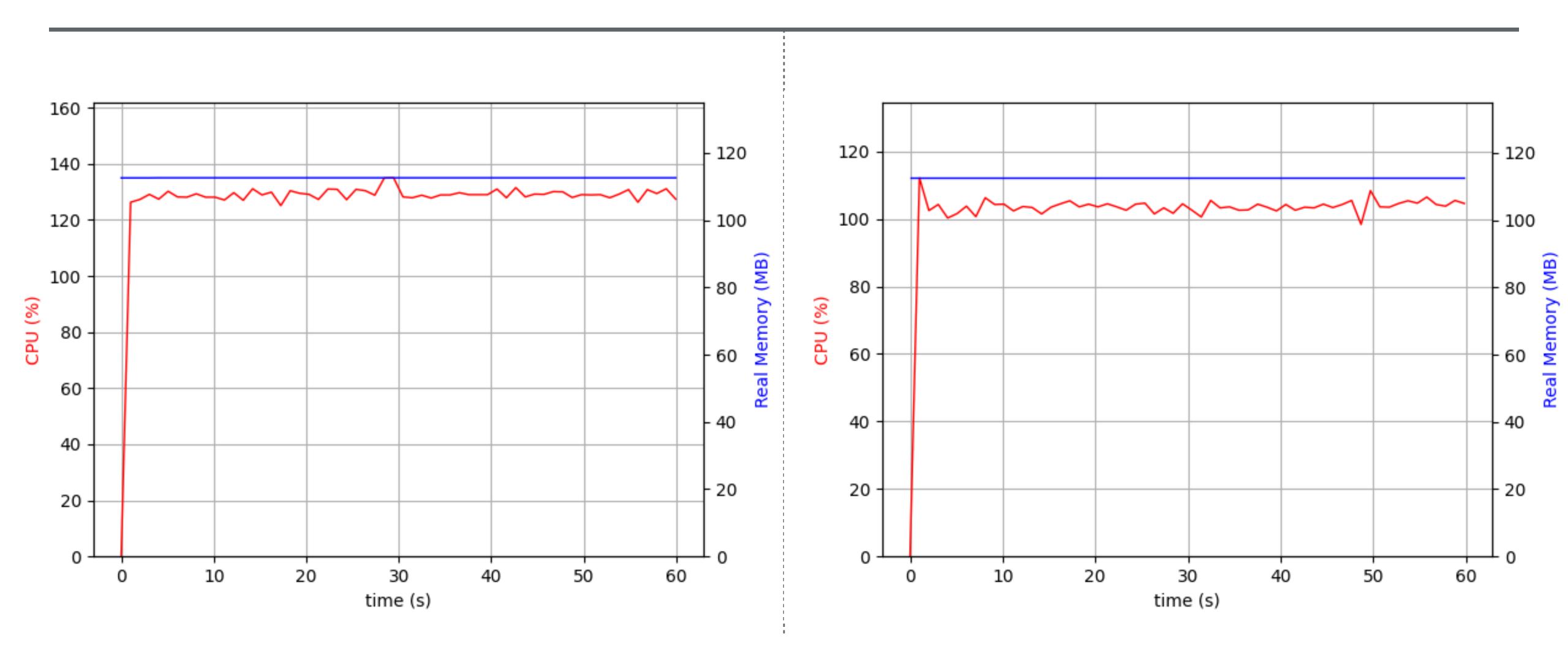


### RESULTADO 3 - LATÊNCIA PARA ENVIO DE UM COMANDO A UM ATUADOR



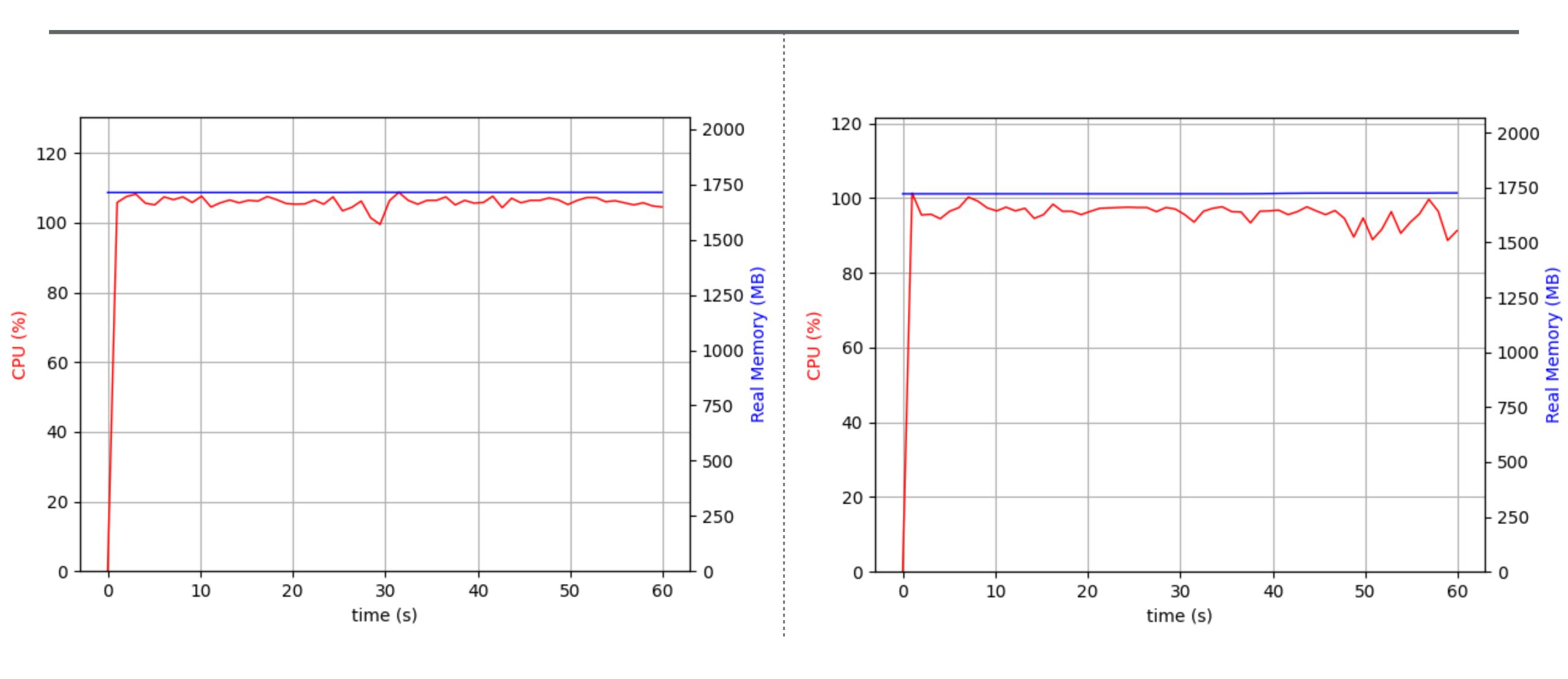
### RESULTADO 4 - USO DE CPU E MEMÓRIA NO SERVIDOR - RESOURCE ADAPTOR

HTTP



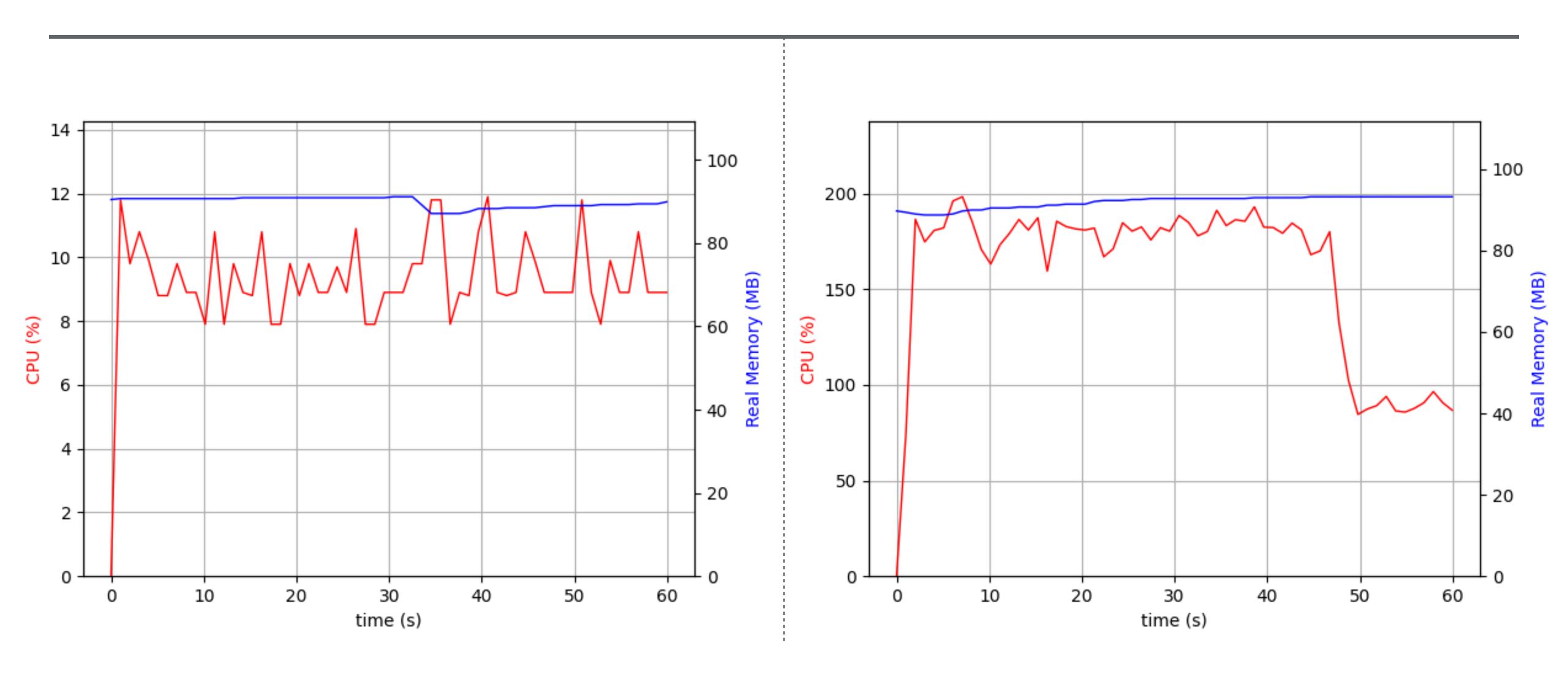
### RESULTADO 4 - USO DE CPU E MEMÓRIA NO SERVIDOR - DATA COLLECTOR

HTTP



### RESULTADO 4 - USO DE CPU E MEMÓRIA NO SERVIDOR - RABBITMQ

HTTP



#### CONCLUSÕES

- Desempenho muito parecido entre as duas formas de interação com a plataforma
- Muito vantajoso para a rede de dispositivos lo T
- Impacto no servidor não deve ser um problema

### PRÓXIMOS PASSOS

- Deploy e Pull Request
- Avaliar o desempenho em um contexto de produção
- Rede de experimentação na USP

