

SENAI – Desenvolvimento de Sistemas

Relatório: Comunicação entre PHP e HiveMQ via MQTT

**Aluno: Gabriel V. Thiago T. Vinicius T. Rafael S.
DS23M1**

1. Introdução

O protocolo MQTT (Message Queuing Telemetry Transport) é amplamente utilizado na Internet das Coisas (IoT) por sua leveza e eficiência na troca de mensagens entre dispositivos e servidores. O HiveMQ é um dos brokers mais populares, responsável por gerenciar o fluxo dessas mensagens de forma segura e escalável.

2. Objetivo

Analisar como as aplicações PHP podem se comunicar com o HiveMQ, identificando bibliotecas, APIs e soluções híbridas disponíveis para essa integração.

3. Soluções Encontradas

A) Biblioteca PHP Nativa — Bluerhinos/phpMQTT

É uma implementação simples do cliente MQTT em PHP. Permite publicar e assinar tópicos diretamente com brokers como o HiveMQ. Os requisitos incluem PHP 7+, extensão sockets e Composer. É ideal para aplicações que precisam de comunicação em tempo real.

B) API REST do HiveMQ Cloud

Disponível em <https://www.hivemq.com/docs/cloud/api>, essa API permite gerenciar clusters, clientes e métricas do HiveMQ Cloud via requisições HTTP. Pode ser consumida em PHP usando GuzzleHTTP. Contudo, esta API não publica ou consome mensagens MQTT diretamente.

C) Solução Híbrida (Node-RED/Python Intermediário)

Envolve um microserviço externo que gerencia MQTT (usando Node.js ou Python) e expõe uma API HTTP que o PHP consome. Esta solução é ideal para hospedagens que não suportam conexões contínuas. Usa ferramentas como Node-RED, Flask ou Mosquitto.

4. Comparativo das Alternativas

Solução	Vantagens	Limitações
Bluerhinos/phpMQTT	Leve, simples, direto	Não ideal para conexões persistentes
HiveMQ REST API	Ótima para administração	Não envia/recebe mensagens MQTT

Solução Híbrida Escalável e robusta Maior complexidade

5. Aplicação Prática

Em um cenário IoT real, sensores de temperatura poderiam publicar dados em um tópico MQTT, e o PHP usaria phpMQTT para exibir esses dados em tempo real. Para segurança, recomenda-se a adoção de medidas como TLS/SSL, autenticação por usuário/senha e o uso de tokens JWT no HiveMQ Cloud.

6. Conclusão

Entre as opções analisadas, a biblioteca Bluerhinos/phpMQTT é a mais prática e eficiente para projetos PHP reais. Permite integração direta com o HiveMQ, sendo leve, open source e ideal para protótipos e sistemas de IoT.