
Documentação de Aplicação

Author :

[wp_objects_pdf]

wp_objects_pdf();

Desenvolvimento Software Open Source

(Em breve)

[Acesso ao Network Server](#)

O *Network Server* é um servidor voltado para Internet das coisas (*IoT – Internet of Things*) e caracteriza-se por gerenciar os dados enviados por um *node* (dispositivo embarcado com transmissor de rádio frequência referente ao protocolo *LoRaWAN*) e disponibilizar estes dados ao usuário final de uma aplicação que se utiliza deste servidor. Este serviço encontra-se em uma plataforma de nuvem hospedada em um *datacenter* localizado no Instituto Mauá de Tecnologia, *campus* São Caetano do Sul. A esta documentação propõe-se permitir que o usuário realize o acesso ao servidor para cadastrar e ativar os *nodes* referentes à respectiva aplicação. Por fim, de maneira objetiva, sugerir meios para que os dados armazenados no banco de dados possam ser recuperados ou transmitidos à aplicação no momento do envio de um pacote de informação pelo *node* ao servidor.

[Acesso ao Network Server para download em pdf.](#)

[Acesso ao MQTT broker \(Network Server\)](#)

O *Network Server* executa um *publish* (publicação) do dado recebido pelo *node* (dispositivo embarcado com transmissor de rádio frequência referente ao protocolo *LoRaWAN*) da aplicação em um *sub-servidor* denominado *MQTT (Message Queueing Telemetry Transport) broker*. O recebimento do dado pelo usuário na aplicação é realizado através de um *subscribe* (inscrição) neste *MQTT broker* de acordo com as características de endereçamento de *IP (Internet Protocol)*, porta, nome de usuário e senha corretamente inscritos em um tipo de endereçamento denominado *tópico*. O *tópico* se trata do *“local”* em que uma mensagem transmitida é disponibilizada. Logo, o acesso ao dado do *node* no momento enviado é restrito somente a quem possui o direito sobre a determinada aplicação. Diferentemente da utilização de *REST API (Representational State Transfer Application Programming Interface)*, não há possibilidade de requisição de dados armazenados no banco de dados. O dado estará disponível apenas no momento em que o *node* enviá-lo ao servidor e para quem realizou um *subscribe* (inscrição) no *tópico* mencionado neste mesmo servidor de maneira autenticada,

através de parâmetros de nome de usuário e senha, como citado anteriormente. Caracterizando-se por um dado volátil, portanto, é necessário que se esteja conectado para recebê-lo. Também é importante lembrar que a conexão é de caráter permanente e autoreconectável. No tipo protocolo *MQTT* não se faz uma requisição de conexão intermitente, isto é, temporária. E, sim permanece conectado até quando se desejar.

[Acesso ao MQTT broker \(Network Server\) para download em pdf.](#)

[Acesso à REST API \(Network Server\)](#)

A recuperação dos dados enviados através dos *nodes* por cada aplicação é importante em relação ao histórico de dados disponível. Além de ser um backup dos dados caso uma conexão *MQTT broker* não seja estabelecida pela aplicação, o acesso à *REST API* permite que o usuário realize um backup de tempos em tempos para o próprio banco de dados. A saber que o banco de dados do *Network Server* apenas garante o armazenamento dos dados referentes aos últimos 7 dias.

Para o processo descrito ocorra com segurança entre as camadas servidor e aplicação, foi desenvolvida uma *REST API* (*Representational State Transfer Application Programming Interface*) em que através de uma requisição *HTTP* (*HyperText Transfer Protocol*) do tipo *GET*, são retornados os últimos valores inseridos no banco de dados de maneira decrescente. Isto é, do mais atual ao mais antigo, de acordo com o parâmetro de quantidade de *logs*, inserido no momento da requisição. Esta documentação especifica os detalhes deste tipo de requisição e métodos de retorno. Para um exemplo de aplicação com este recurso deve-se consultar a documentação EXEMPLO DE APLICAÇÃO REST-API (NODE-RED)

[Acesso à REST API \(Network Server\) para download em pdf.](#)

Criação de Ambiente para Aplicação (Node-RED)

[Instalação ORACLE VM Virtual Box + Ubuntu 16.04.x](#)

Bem como o Centro de Pesquisas desenvolveu esta aplicação em um sistema operacional *Linux*, esta documentação demonstra como instalar um ambiente em máquina virtual para Sistemas operacionais como *Linux* e *Windows*. Também é possível para os Sistemas *MAC* executarem a instalação de uma máquina virtual. Neste caso o ambiente de virtualização trata-se do Oracle Virtual Box.

Instalação Oracle VM Virtual Box + Ubuntu 16.04.x para download em pdf. (Em breve)

[Instalação Node-RED + Segurança](#)

Para que o usuários de aplicação possam reproduzir uma aplicação em Node-RED, este documento aborda como se proceder passo a passo em termos de instalação e segurança necessária para se criar um servidor local de aplicação. A partir daí de qualquer lugar dentro de uma mesma rede é possível acessar esta aplicação.

Instalação *Node-RED* + Segurança para *download* em *pdf*. (Em breve)

[Exemplo de Aplicação MQTT broker \(Node-RED\)](#)

A aplicação demonstrada nesta documentação é baseada em Node-RED. Após instanciada, seja localmente ou em nuvem, é possível que um cliente *MQTT Node-RED* se conecte ao servidor *MQTT* do *Network Server*. Todos os passos para este procedimento até a exibição do valor instantâneamente enviado pelo node da aplicação em um dashboard são demonstrados segundo esta documentação.

Exemplo de Aplicação *MQTT broker (Node-RED)* para *download* em *pdf*. (Em breve)

[Exemplo de Aplicação REST API \(Node-RED\)](#)

A aplicação demonstrada nesta documentação é baseada em Node-RED. Após instanciada, seja localmente ou em nuvem, é possível se utilizar de uma requisição http para que se recupere os dados decriptografados e decodificados do banco de dados exatamente como foi enviado anteriormente pelo *node* ao servidor. Este procedimento é voltado para a recuperação dos dados e não para o monitoramento em tempo real. Para maiores detalhes sobre este tipo de requisição consulte *ACESSO À REST-API (NETWORK SERVER)*. Todos os passos para este procedimento até a exibição de valor recuperado do banco de dados em um dashboard são demonstrados segundo esta documentação.

Exemplo de Aplicação *REST API (Node-RED)* para *download* em *pdf*. (Em breve)