

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO
ESPÍRITO SANTO – CAMPUS SERRA

ISABELLE PEREIRA SOUZA DIAS
WELLERSON OLIVEIRA

PROJETO REDES
TRABALHO 2

SERRA

2021

ISABELLE PEREIRA SOUZA DIAS
WELLERSON OLIVEIRA

PROJETO REDES

TRABALHO 2

Projeto apresentado à disciplina de Redes Industriais de Computadores, do curso de Engenharia de Controle e Automação, do Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo, como requisito parcial para avaliação na referida disciplina.

Professor: Prof. Rafael Emerick

SERRA

2022

SUMÁRIO

1	APRESENTAÇÃO.....	4
2	PLANTA INDUSTRIAL	5
3	SISTEMA SUPERVISÓRIO.....	10
4	TELA IHM DO SUPERVISÓRIO	14

1 APRESENTAÇÃO

Utilizando-se da infraestrutura disponível da AWS Educate, criamos duas máquinas virtuais (MV) para criação do ambiente de testes de supervisão Modbus TCP/IP. Uma MV fará a função de supervisor a outra de CLP controlando uma planta hipotética simulada.

Na máquina de supervisão foi instalado o sistema supervisor ScadaBR, onde foi cadastrado o elemento de controle, e onde disponibilizamos também uma IHM para melhor visualização dos dados enviados pela planta. Também nessa IHM temos um botão de atuação da supervisão sobre o controle.

Na máquina que representa a simulação do controle instalamos o Node-RED. O Node-RED é um ambiente de programação baseado em eventos que disponibiliza uma interface de blocos e permite o desenvolvimento de aplicações orientadas à evento sobre o Node.js. Com isso, uma planta foi simulada por meio de uma aplicação baseada em fluxos.

Para a integração via Modbus-TCP, após a instalação do Node-Red foi instalado o módulo *'node-red-contrib-modbus'*. Para que fosse possível instanciar um Servidor Modbus TCP, com suporte a atualização e leitura das tabelas Modbus. Uma vez criado o servidor, foi desenvolvido um emulador de planta industrial para atualizar variáveis discretas e analógicas simulando uma planta real, por meio dos *'injections'* e das *'functions'*. Para simulação de valores arbitrários, foi instalado também a biblioteca *'node-red-node-random'* que permitiu a geração de valores aleatórios que representavam sensores em nossa planta industrial.

Foi então desenvolvido um simulador de forma a permitir o monitoramento e atuação das variáveis geradas artificialmente no node-red por meio do Supervisor via Modbus-TCP.

Link da apresentação: <https://www.youtube.com/watch?v=8tHapLZrnW4>

2 PLANTA INDUSTRIAL

A planta industrial foi representada em uma VM utilizando o Node-RED, para criação dessa VM foi criada uma instância na AWS.

Resumo da instância para i-0cb736e0fb9428b9c [Informações](#)

Atualizado há less than a minute

↻

Conectar

Estado da instância ▼

Ações ▼

ID de instância i-0cb736e0fb9428b9c	Endereço IPv4 público 3.93.183.43 endereço aberto	Endereços IPv4 privados 172.31.95.54
Endereço IPv6 -	Estado da instância Executando	DNS IPv4 público ec2-3-93-183-43.compute-1.amazonaws.com endereço aberto
Tipo de nome do host Nome do IP: ip-172-31-95-54.ec2.internal	Nome do DNS de IP privado (somente IPv4) ip-172-31-95-54.ec2.internal	Nome do DNS do recurso privado de resposta -
Tipo de instância t2.micro	Endereços IP elásticos -	ID da VPC vpc-02e0e94078956e792
Descoberta do AWS Compute Optimizer Opte por participar do AWS Compute Optimizer para obter recomendações. Saiba mais	Função do IAM -	ID da sub-rede subnet-09a5361e03ef4304b

Com as seguintes regras de segurança:

▼ Regras de entrada

Q Regras de filtro

ID da regra do grupo de ...	Intervalo de p...	Protocolo	Origem	Grupos de segurança
sgr-06b1824aabf286a90	1880	TCP	::/0	launch-wizard-11
sgr-067a5e23a468563a6	10502	TCP	::/0	launch-wizard-11
sgr-0435b37476acfc63c	22	TCP	0.0.0.0/0	launch-wizard-11
sgr-0c615ccf534d9e7ee	10502	TCP	0.0.0.0/0	launch-wizard-11
sgr-058c55618f7142e60	1880	TCP	0.0.0.0/0	launch-wizard-11

▼ Regras de saída

Q Regras de filtro

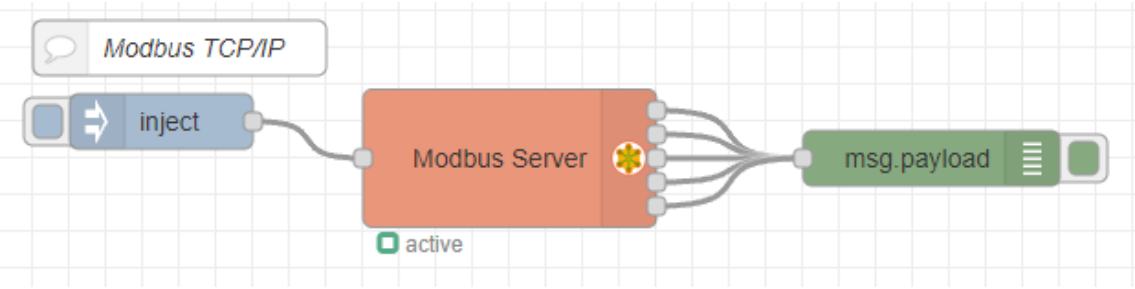
ID da regra do grupo de ...	Intervalo de p...	Protocolo	Destino	Grupos de segurança
sgr-032a0aba8d8c4203b	Todos	Todos	0.0.0.0/0	launch-wizard-11

Foi liberado o acesso de origem de qualquer IP para a porta 1880 pois essa era a porta de acesso para o Node-RED e ela precisava ser acessado por todos os integrantes do grupo. Já a porta 10502 foi liberada para comunicar-se com o Modbus IP do Supervisório ScadaBR. Aqui liberamos a origem para qualquer IP já que por estarmos usando uma instância gratuita os endereços de IP associados à nossas VMs mudavam de tempo em tempo. Para facilitar a configuração e testes da comunicação entre o supervisório e a planta, foi liberado o acesso à qualquer IP, porém em um caso real essa porta estaria liberada somente para IPs específicos.

Configurações no Node-RED

Servidor

Foi criado um servidor Modbus, sendo configurado a porta 10502 para comunicação, habilitando a comunicação modbus entre o Scada e os hosts modbus.



Gerador de entradas para o servidor

Edit inject node

Delete

⚙ Properties

♥ Name

Name

≡ msg. payload

=

▼

≡ msg. topic

=

▼

⚙ add

☐ Inject once after 0.1 seconds, then

🔄 Repeat

none

▼

Configuração do servidor

Sendo configurado para porta 10502 e de origem de qualquer IP.

Edit Modbus-Server node

Delete

⚙ Properties

Name

Name

Hostname

0.0.0.0

Port

10502

Coils

10000

Holdings

10000

Inputs

10000

Discretes

10000

⚙ Show Errors

☐

⚙ Logging

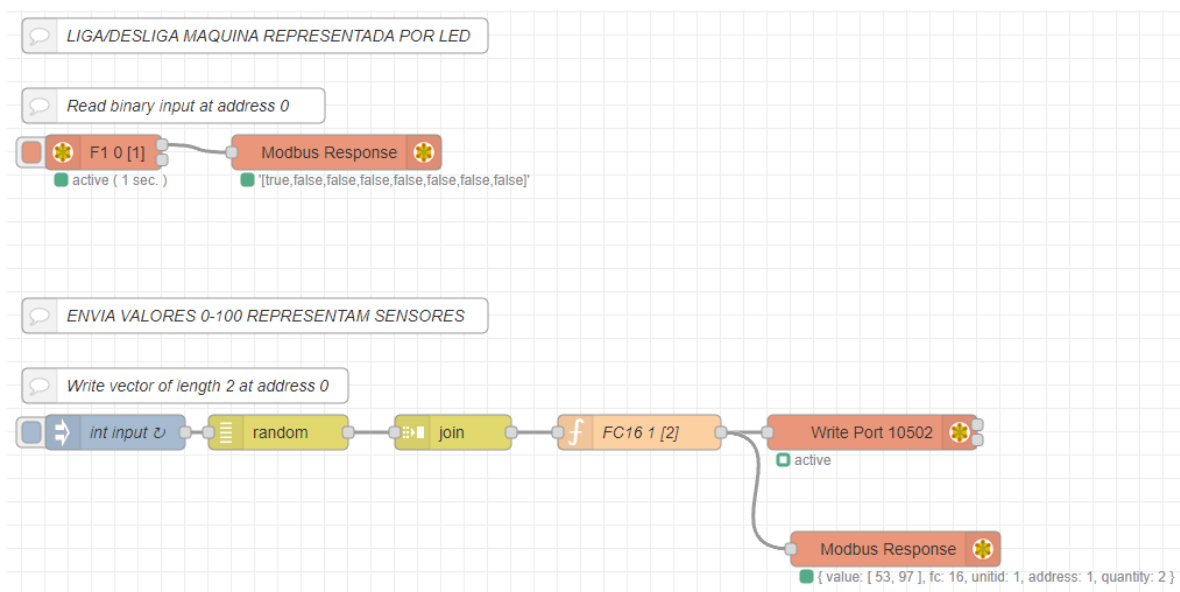
☐

Fluxos implementados

Foram implementados dois fluxos:

O primeiro recebe um sinal de um botão criado no sistema ScadaBR, acionando ou apagando um led. A cada 1 segundo o Node-RED está programado para ler os dados enviados no endereço especificado, e mudar seu estado de acordo com o informado no ScadaBR. Para mostrar a atualização de dados, foi criado um módulo de resposta que recebe os valores do módulo de entrada e indica no seu primeiro bit o estado do led, sendo true para ligado e false para desligado.

O segundo, o Node-RED gera valores aleatórios simulando valores coletados por sensores e os envia para o ScadaBR. Para isto foi criado um módulo de que gera entradas, conectado a um módulo random, que transforma a entrada em um valor aleatório de 0 a 100, o módulo join segura o valor até ser gerado um segundo valor, enviando os dois em forma de vetor para o módulo de função, onde é indicado os parametros para envio dos dados ao scada, como função, endereço inicial, quantidade de elementos, e por ultimo o módulo de escrita envia os dados para o Scada.



Configuração realizada:

Módulo de recebimento do estado do botão:

The screenshot shows the 'Edit Modbus-Read node' configuration window. It has a 'Delete' button at the top. Below is a 'Properties' section with two tabs: 'Settings' (active) and 'Optional'. The 'Settings' tab contains the following fields: 'Name' (F1 0 [1]), 'Topic' (Topic), 'Unit-Id' (1), 'FC' (FC 1: Read Coil Status), 'Address' (0), 'Quantity' (1), 'Poll Rate' (1 second(s)), 'Delay on start' (unchecked), and 'Server' (ScadaBR).

Servidor criado:

Foi utilizado a porta 11502 para comunicação entre o node-red e o scada.

The screenshot shows the 'Properties' configuration for the 'ScadaBR' server. It includes the following fields: 'Name' (ScadaBR), 'Type' (TCP), 'Host' (0.0.0.0), 'Port' (10502), 'TCP Type' (DEFAULT), 'Unit-Id' (1), 'Timeout (ms)' (1000), 'Reconnect on timeout' (checked), and 'Reconnect timeout (ms)' (2000).

Módulo de geração de entradas para geração dos valores aleatórios.

The screenshot shows the 'Edit inject node' configuration window. It has a 'Delete' button at the top. Below is a 'Properties' section with a 'Name' field (int input). There are two message boxes: 'msg. payload' and 'msg. topic', each with a dropdown menu. At the bottom, there is an 'Inject once after' field (0.1 seconds, then) and a 'Repeat' section with a dropdown (interval) and an 'every' field (1 seconds).

Módulo para transformar a entrada em um valor aleatório de 0 a 100

The screenshot shows the 'Edit random node' configuration window. It has a 'Delete' button at the top. Below is a 'Properties' section with the following fields: 'Property' (msg. payload), 'Generate' (a whole number - integer), 'From' (0), 'To' (100), and 'Name' (Name).

Módulo para armazenar os valores gerados até completar 2 valores.

The 'Edit join node' window shows the following configuration:

- Mode:** manual
- Combine each:** msg.payload
- to create:** an Array
- Send the message:**
 - After a number of message parts: 2
 - After a timeout following the first message: seconds
 - After a message with the `msg.complete` property set
- Name:** Name

Módulo com a função que parametriza os dados de envio ao scada

The 'Edit function node' window shows the following configuration:

- Name:** FC16 1 [2]
- Setup:** (selected tab)
- Code:**

```
1- msg.payload = {  
2  value: msg.payload,  
3  'fc': 16,  
4  'unitid': 1,  
5  'address': 1 ,  
6  'quantity': 2  
7- };  
8  return msg;  
9
```









Módulo que realiza o envio dos dados ao scada.

The 'Edit Modbus-Flex-Write node' window shows the following configuration:

- Name:** Write Port 10502
- Server:** ScadaBR
- Options:**
 - Empty msg on fail: ☐
 - Keep Msg Properties: ☐
 - Show Activities: ☐
 - Show Errors: ☐

3 SISTEMA SUPERVISÓRIO

A planta industrial foi representada em uma VM utilizando o ScadaBR instalado nela, para criação dessa VM foi criada uma instância na AWS.

Resumo da instância para i-00e8673b34bc321a7			Informações				
Atualizado há less than a minute					Conectar	Estado da instância	Ações
ID de instância			Endereço IPv4 público	Endereços IPv4 privados			
 i-00e8673b34bc321a7			 54.146.181.206 endereço aberto	 172.31.88.5			
Endereço IPv6			Estado da instância	DNS IPv4 público			
-			 Executando	 ec2-54-146-181-206.compute-1.amazonaws.com endereço aberto			
Tipo de nome do host			Nome do DNS de IP privado (somente IPv4)	Nome do DNS do recurso privado de resposta			
Nome do IP: ip-172-31-88-5.ec2.internal			 ip-172-31-88-5.ec2.internal	IPv4 (A)			
Tipo de instância			Endereços IP elásticos	ID da VPC			
t2.micro			-	 vpc-02e0e94078956e792			
Descoberta do AWS Compute Optimizer			Função do IAM	ID da sub-rede			
 Opte por participar do AWS Compute Optimizer para obter recomendações. Saiba mais			-	 subnet-09a5361e03ef4304b			







Com as seguintes regras de segurança:

▼ Regras de entrada				
<input type="text" value="Regras de filtro"/>				
ID da regra do grupo de ...	Intervalo de p...	Protocolo	Origem	Grupos de segurança
sgr-07428ec95f25328e5	8080	TCP	0.0.0.0/0	launch-wizard-10
sgr-0611c54b91b1d970b	22	TCP	0.0.0.0/0	launch-wizard-10
sgr-094920116c6419d28	8080	TCP	::/0	launch-wizard-10
▼ Regras de saída				
<input type="text" value="Regras de filtro"/>				
ID da regra do grupo de ...	Intervalo de p...	Protocolo	Destino	Grupos de segurança
sgr-09a0ee995eb906abc	Todos	Todos	0.0.0.0/0	launch-wizard-10





Foi liberado o acesso de origem de qualquer IP para a porta 8080 pois essa era a porta de acesso para o ScadaBR e ela precisava ser acessado por todos os integrantes do grupo. Não precisamos configurar uma porta especial para acessar a nossa planta, pois essa configuração entraria nas regras de saída que já estava liberado o acesso para qualquer porta em qualquer destino.

Configurações no SCADA:

Foi criado um data Sourcer do tipo ModBus IP para ser realizada a comunicação com o Node-RED.

Data sources 				
1-wire 				
<u>Nome</u> ▲	<u>Tipo</u>	<u>Conexão</u>	<u>Status</u>	
Trab2	Modbus IP	3.93.183.43:10502		   

Dentro do data sourcer foram criados três data points para a leitura e envio dos dados para o node-red.

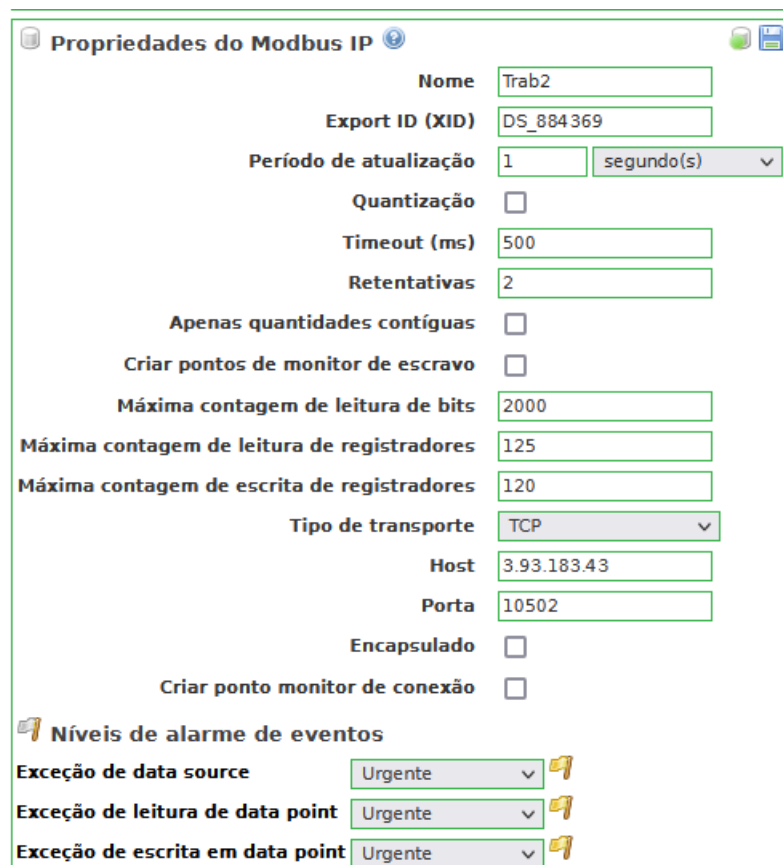
Watch list 	
 Trab2 - LED	0
 Trab2 - Sensor	23.0
 Trab2 - Sensor2	49.0

- LED : utilizado para enviar a mudança de estado do botão e para receber a confirmação de mudança de estado do led.
- Sensor: Recebe o primeiro valor do vetor de número aleatórios gerados no node-red.
- Sensor2: Recebe o primeiro valor do vetor de número aleatórios gerados no node-red.

Configuração do Data Soucer

Primeiro precisamos escolher o tipo de conexão, e como estamos trabalhando com o Modbus esse foi o tipo escolhido. Depois escolhemos as outras configurações específicas de nossa comunicação TCP/IP.

- Tipo de transporte: TCP
- Host: inserido o IP da máquina node-red, nesse caso 3.93.183.43
- Porta: foi utilizada a porta 10502 para comunicação com o node-red.



Propriedades do Modbus IP

Nome: Trab2

Export ID (XID): DS_884369

Período de atualização: 1 segundo(s)

Quantização: ☐

Timeout (ms): 500

Retentativas: 2

Apenas quantidades contíguas: ☐

Criar pontos de monitor de escravo: ☐

Máxima contagem de leitura de bits: 2000

Máxima contagem de leitura de registradores: 125

Máxima contagem de escrita de registradores: 120

Tipo de transporte: TCP

Host: 3.93.183.43

Porta: 10502

Encapsulado: ☐

Criar ponto monitor de conexão: ☐

Níveis de alarme de eventos

Exceção de data source: Urgente

Exceção de leitura de data point: Urgente

Exceção de escrita em data point: Urgente

Configuração dos Data Points

Em seguida precisamos definir os data points, que são os dados que serão supervisionados por nosso supervisor. Aqui definimos uma LED, que representaria o ligar e desligar de alguma máquina ou acionador, e dois sensores.

Data points						
Nome	Tipo de dado	Status	Escravo	Faixa	Offset (baseado em 0)	
LED	Binário		1	Bobina (Coil Status)	0	
Sensor	Numérico		1	Registrador holding	1	
Sensor2	Numérico		1	Registrador holding	2	

Abaixo segue a configurações de cada data point.

LED

Detalhes do data point

Nome

LED

Export ID (XID)

DP_565488

Id do escravo

1

Faixa do registro

Bobina (Coil Status)

Tipo de dados modbus

Binário

Offset (baseado em 0)

0

Bit

0

Número de registradores

0

Codificação de caracteres

ASCII

Configurável

☒

Multiplicador

1

Aditivo

0

Sensor

Detalhes do data point

Nome

Sensor

Export ID (XID)

DP_199975

Id do escravo

1

Faixa do registro

Registrador holding

Tipo de dados modbus

Inteiro de 2 bytes sem sinal

Offset (baseado em 0)

1

Bit

0

Número de registradores

0

Codificação de caracteres

ASCII

Configurável

☒


Multiplicador

1

Aditivo

0

Sensor2

Detalhes do data point 

Nome

Sensor2

Export ID (XID)

DP_376167

Id do escravo

1

Faixa do registro

Registrador holding ▾

Tipo de dados modbus

Inteiro de 2 bytes sem sinal ▾

Offset (baseado em 0)

2

Bit

0

Número de registradores

0

Codificação de caracteres

ASCII

Configurável

☒

Multiplicador

1

Aditivo

0

4 TELA IHM DO SUPERVISÓRIO

A seguinte tela foi desenvolvida para melhor visualização do supervisório:



Componentes:

Botão para acionamento do led

Gráfico e visor para visualização dos valores do primeiro Sensor, e botão para download de um arquivo csv com todos os dados captados pelo ScadaBR.

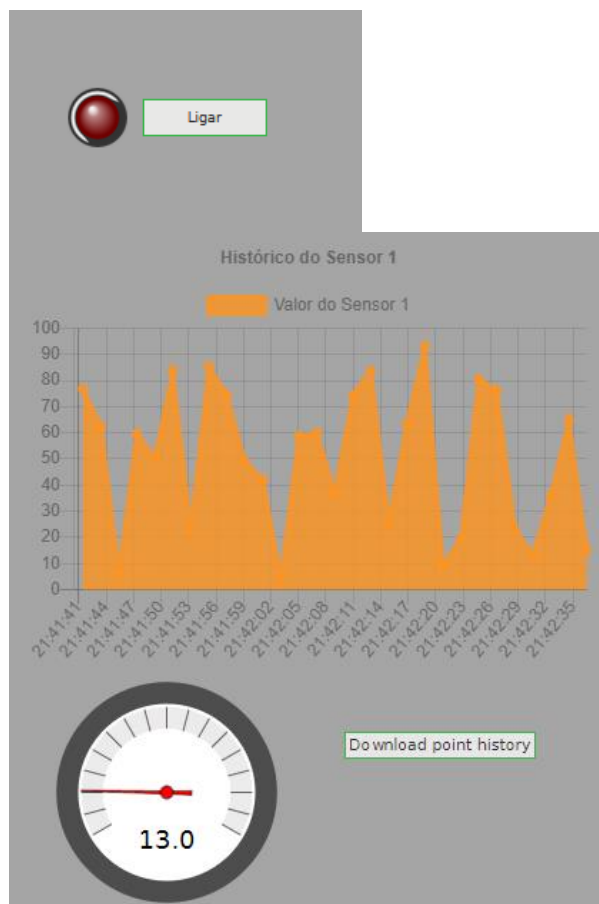


Gráfico e visor para visualização dos valores do primeiro Sensor, e botão para download de um arquivo csv com todos os dados captados pelo ScadaBR.

