Introdução

Nesta pasta encontrem a máquina virtual que é para correr localmente como dispositivo IoT Edge. Este é um ficheiro (comprimido) que foi criado dentro do Windows, para ser executado com o VMWare Workstation Player, e deve de ser executável também dentro de um sistema operativo Linux (não testado).

Colocar a Máquina Virtual a Correr

Importem a máquina virtual dentro do VMWare Workstation Player, e arranquem-na. Quando estiver a correr, usem estes dados para entrar:

username: **ubuntu**

password: **root**

Podem encontrar o endereço da máquina ao fazerem:

ip addr show

e depois podem usar o terminal SSH da vossa escolha para controlarem a máquina:

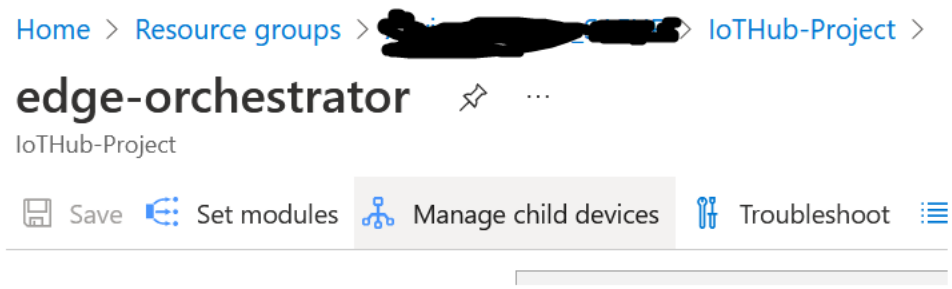
ssh ubuntu@123.456.789.012 (mudar endereço IP para o vosso)

Criem um IoT Hub no Portal Azure, com o registo do vosso dispositivo IoT Edge

Criem um novo recurso "IoT Hub". Quando o fizerem, entrem no recurso, cliquem na aba à esquerda "IoT Edge", e criem um dispositivo edge novo. Dêem-lhe o nome "edge-orchestrator" para que fique igual à configuração da máquina virtual. Configurem autenticação simétrica automaticamente atribuída, e cliquem em criar. Quando estiver criado, entrem no dispositivo, e tomem nota da "Primary Connection String" que lá está.

Criem uma configuração para o vosso dispositivo IoT Edge

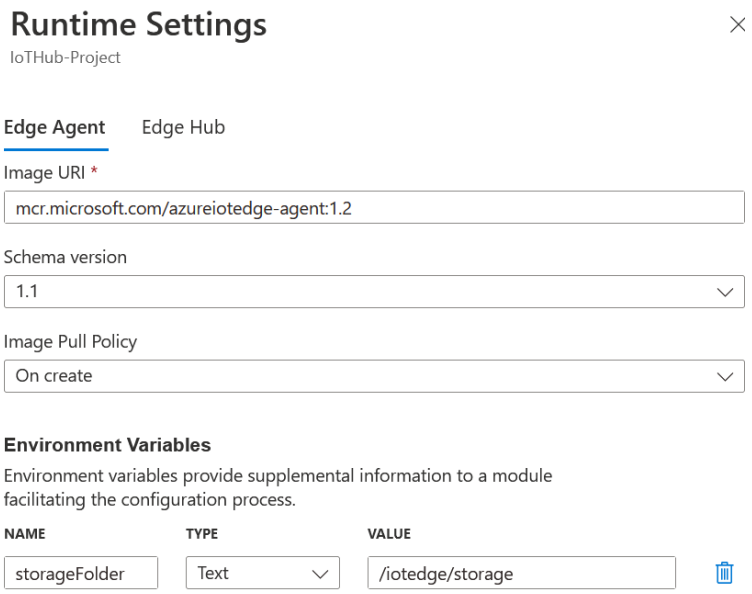
Dentro do vosso hub, selecionem o registo do dispositivo iot edge, e cliquem em set modules:



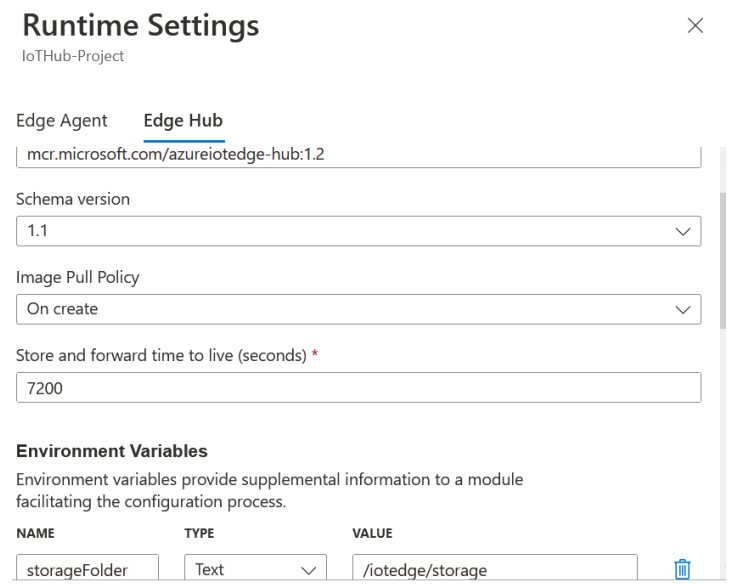
Carreguem em “runtime settings”, e no separador edgeAgent, coloquem este JSON:

```
{
    "HostConfig": {
        "Binds": [
            "/home/ubuntu/storage:/iotedge/storage"
        ],
        "dns": [
            "1.1.1.1"
        ]
    }
}
```

mais as seguintes configurações:



No separador edgeHub, a mesma coisa:



Com o seguinte JSON:

```
{
    "HostConfig": {
        "Binds": [
            "/home/ubuntu/storage:/iotedge/storage"
        ],
        "dns": [
            "1.1.1.1"
        ],
        "PortBindings": {
            "443/tcp": [
                {
                    "HostPort": "443"
                }
            ],
            "5671/tcp": [
                {
                    "HostPort": "5671"
                }
            ],
            "8883/tcp": [
                {
                    "HostPort": "8883"
                }
            ]
        }
    }
}
```

Depois disto, gravem e façam deploy.

Colocar o IoT Edge Runtime a correr

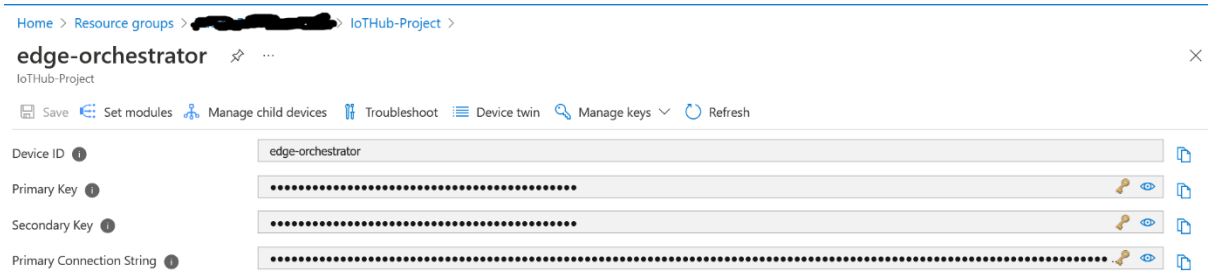
Dentro desta VM (máquina virtual) corre o Ubuntu Server, e espera um registo de dispositivo Edge Enabled no Hub na nuvem. Vocês têm de fazer este registo primeiro, e reconfigurar a string de ligação para o vosso dispositivo, modificando-a no ficheiro "**/etc/aziot/config.toml**".

sudo nano /etc/aziot/config.toml

Procurem a linha

connection_string = "HostName=IoTHub-SAFHE.azure-devices.ne……"

e substituam-na pela primary connection string do vosso dispositivo configurado online:



Para gravar o ficheiro com o editor nano, façam ctrl+X, e gravem o ficheiro. Depois disso, apliquem a nova configuração no IoT Edge Runtime:

sudo iotedge config apply

O ficheiro de configuração mencionado neste ficheiro assume uma configuração de chave simétrica (SAS). Outras configurações têm de ser colocadas neste ficheiro, como o uso de certificados para ambiente de produção. Verifiquem se está tudo bem, e com ligação à Azure online:

sudo iotedge check

O resultado deve de ser:

```
√ aziot-identity-service package is up-to-date - OK
√ host time is close to reference time - OK
√ preloaded certificates are valid - OK
√ keyd is running - OK
√ certd is running - OK
√ identityd is running - OK
√ read all preloaded certificates from the Certificates Service - OK
√ read all preloaded key pairs from the Keys Service - OK
√ ensure all preloaded certificates match preloaded private keys with the same ID - OK

Connectivity checks (aziot-identity-service)
--------------------------------------------
√ host can connect to and perform TLS handshake with iothub AMQP port - OK
√ host can connect to and perform TLS handshake with iothub HTTPS / WebSockets port - OK
√ host can connect to and perform TLS handshake with iothub MQTT port - OK

Configuration checks
--------------------
√ aziot-edged configuration is well-formed - OK
√ configuration up-to-date with config.toml - OK
√ container engine is installed and functional - OK
√ configuration has correct URIs for daemon mgmt endpoint - OK
‼ aziot-edge package is up-to-date - Warning
    Installed IoT Edge daemon has version 1.2.8 but 1.2.9 is the latest stable version available.
    Please see https://aka.ms/iotedge-update-runtime for update instructions.
√ container time is close to host time - OK
√ DNS server - OK
√ production readiness: logs policy - OK
√ production readiness: Edge Agent's storage directory is persisted on the host filesystem - OK
√ production readiness: Edge Hub's storage directory is persisted on the host filesystem - OK
√ Agent image is valid and can be pulled from upstream - OK
√ proxy settings are consistent in aziot-edged, aziot-identityd, moby daemon and config.toml - OK

Connectivity checks
-------------------
√ container on the default network can connect to upstream  AMQP port - OK
√ container on the default network can connect to upstream HTTPS / WebSockets port - OK
√ container on the default network can connect to upstream MQTT port - OK
√ container on the IoT Edge module network can connect to upstream AMQP port - OK
√ container on the IoT Edge module network can connect to upstream HTTPS / WebSockets port - OK
√ container on the IoT Edge module network can connect to upstream MQTT port - OK
35 check(s) succeeded.
1 check(s) raised warnings. Re-run with --verbose for more details.
```

A nível de comunicações, está tudo ok. Se, por outro lado, virem um conjunto de erros a vermelho no connectivity checks, terá a ver com o facto de vocês estarem dentro da rede do PORTIC (SC2, PORTIC, …) ou na rede em que estejam, e que bloqueia as portas dos protocolos testados (MQTT, AMQP). Usem VPN.

Neste ponto, vocês podem executar o seguinte comando para listar os módulos a correr no IoT Edge Runtime:

sudo iotedge list

```
ubuntu@edge-orchestrator: $ sudo iotedge list
NAME             STATUS           DESCRIPTION      CONFIG
edgeAgent        running          Up a minute      mcr.microsoft.com/azureiotedge-agent:1.2
edgeHub          running          Up a minute      mcr.microsoft.com/azureiotedge-hub:1.2
```

Se não virem, façam reset (**sudo iotedge system restart**).

Parabéns, a partir daqui, vocês podem configurar o dispositivo edge da forma que precisarem para o vosso projeto.

[Opcional:] Troubleshooting

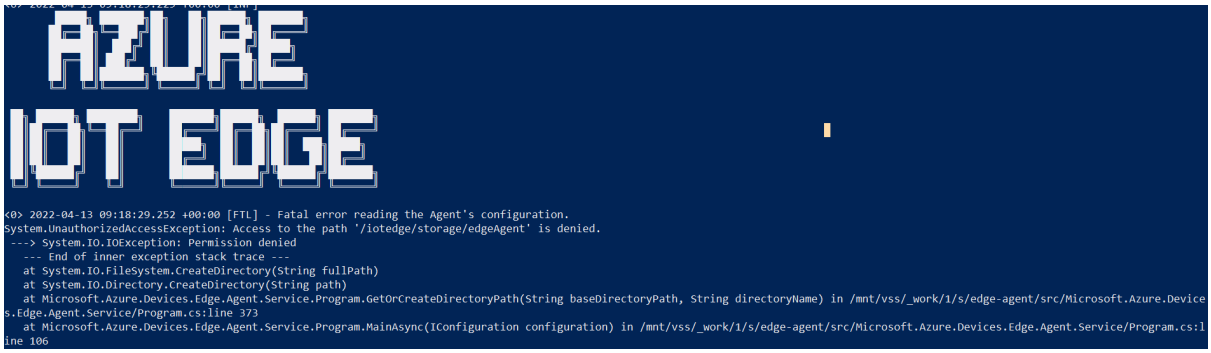
Se vocês obtiveram:

```
ubuntu@edge-orchestrator:~$ sudo iotedge list
NAME             STATUS           DESCRIPTION                CONFIG
edgeAgent        failed           Failed (1) a minute ago    mcr.microsoft.com/azureiotedge-agent:1.2
```

E, ao executarem o comando

sudo iotedge logs edgeAgent

virem:



Executem o seguinte:

sudo chown 1000 /home/ubuntu/storage

sudo chmod 700 /home/ubuntu/storage

sudo iotedge restart edgeAgent

Também pode receber o erro de que a configuração do Digital Twin não foi passada ao dispositivo local:

Isto quer dizer que vocês não têm uma configuração no manifest na nuvem. Têm de ir à nuvem e reconfigurar os vossos módulos. Depois:

sudo iotedge system restart

Ou

sudo iotedge restart edgeAgent