

03

Notificação pelo SMS

Transcrição

Com as notificações por e-mail funcionando, podemos configurar as por SMS. No serviço de SNS criamos um novo tópico chamado `temperatura-sms` e com `Display name` de valor `temp-sms`. Deixamos o mesmo selecionado e selecionamos a opção `Subscribe to topic`.

The screenshot shows the AWS SNS Topics page. At the top, there are three buttons: 'Publish to topic', 'Create new topic', and 'Actions ▾'. Below these is a 'Filter' input field. A table lists topics with columns for Name and ARN. The 'temperatura-sms' topic is selected, indicated by a checked checkbox in the first column. A context menu is open over this topic, listing the following options: 'Edit topic display name', 'Subscribe to topic' (which is highlighted in orange), 'Confirm a subscription', 'Edit topic policy', 'Edit topic delivery policy', 'Delivery status', and 'Delete topics'. To the right of the table, there are two lines of text: 'temperatura' and 'temperatura-sms'.

	Name	ARN
<input type="checkbox"/>	temperatura	arn:aws:sns:brasil:temperatura
<input checked="" type="checkbox"/>	temperatura-sms	arn:aws:sns:brasil:temperatura-sms

Quando selecionado o `Protocol` para ser `SMS`, precisamos tomar o cuidado de informar o número completo do telefone, incluindo o código de país que no caso do Brasil é `55`. Então teremos o número `5511xxxxxxxx` para enviar mensagens para algum telefone em São Paulo por exemplo. Configurado isso, podemos seguir para o próximo passo.

Configure um telefone ao qual tenha acesso, pois é preciso confirmar o número com um código que será enviado confirmando o mesmo na própria plataforma.

O tópico criado já será o nosso `SNS target` na integração das ações do AWS IoT. Nós também já temos o `role` de integração dos dois serviços que criamos na aula anterior. O que está faltando?

No menu `Act` do serviço AWS IoT, clicamos na regra `TelemetriaNotify` e dentro dessa regra, configuramos uma nova ação clicando no botão `Add action`. Dessa forma teremos duas ações ligadas a uma mesma regra. Uma delas envia o e-mail e a outra o SMS. Dentro do seletor de serviços da ação selecionamos novamente o serviço de SNS chamado `Send a message as` `SNS push notification` e nos seletores de `Target` e `Role` selecionamos `temperatura-sms` e `iot-sns` respectivamente.



Send a message as an SNS push notification

SNS

*SNS target

temperatura-sms

[Create a new resource](#)

Message format

JSON

Choose or create a role to grant AWS IoT access to the SNS resource to perform this action.

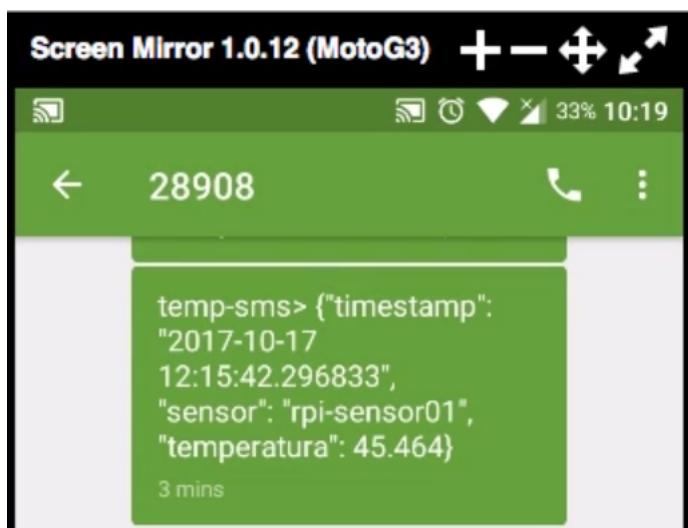
*IAM role name

iot-sns

[Update role](#)[Create a new role](#)

Dessa vez configuramos o formato da mensagem para JSON. Lembre-se de que ao selecionar o role é preciso clicar no botão Update role para confirmar a seleção.

Executamos novamente o `start.sh` e o comando para estressar o processador visto na aula anterior e como resultado temos os e-mails e SMSs sendo enviados.



Pronto! Temos tudo funcionando conforme planejamos. Já capturamos a temperatura, enviamos para os serviços da Amazon e temos o armazenamento em banco de dados e notificações via e-mail e SMS. Nossa próximo passo é ajustar o código em Python para que ao invés de capturar a temperatura do Raspberry PI, capturar a do sensor de temperatura real. Vamos lá?