

## Autorizando e testando o assistente pessoal

### Transcrição

Para testar o Alexa, não podemos utilizar a linha de comando do Raspberry Pi, via SSH, pois o seu ambiente gráfico é requerido pelo assistente pessoal. E para usar o ambiente gráfico do Raspberry Pi, podemos conectar um monitor e um teclado a ele, como se fosse um computador desktop normal, ou utilizamos o **VNC** (*Virtual Network Computing*), que permite acessar a interface gráfica de um computador remotamente.

*Caso você queira utilizar o VNC, há [este tutorial \(https://cursos.alura.com.br/course/iot-amazon-alexa/task/28796\)](https://cursos.alura.com.br/course/iot-amazon-alexa/task/28796) ensinando a instalá-lo.*

Para o Alexa funcionar, precisamos rodar um *web service* para a parte de autorização, executar um aplicativo para se comunicar com o *Alexa Voice Service* e executar o mecanismo de palavra despertador, que permite iniciar uma interação usando a frase "Alexa".

### Executando o web service

Para executar o *web service*, que fará a parte de autorização, entramos na pasta **samples**, dentro da pasta do projeto, e executamos o seguinte comando:

```
pi@raspberrypi:~/alexa-avs-sample-app/samples $ cd companionService && npm start
```

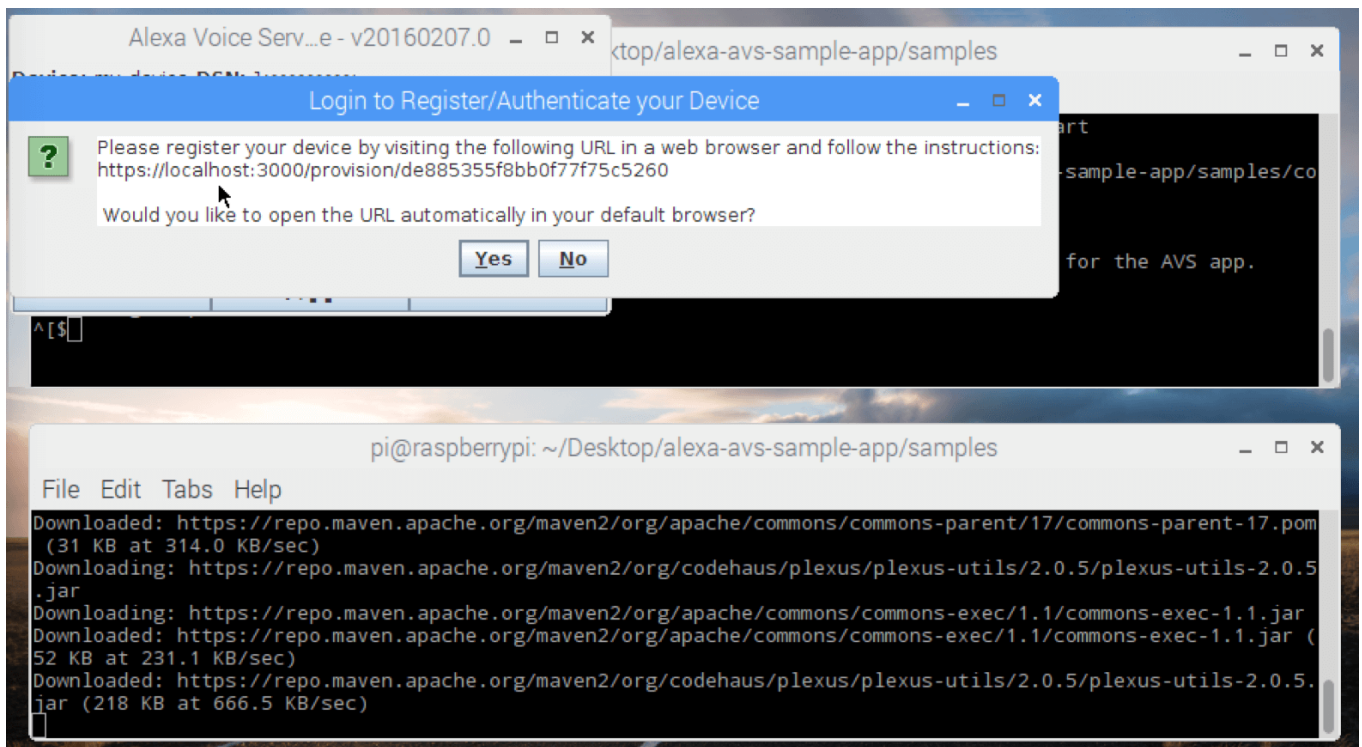
Com isso, deixamos o terminal aberto, com o *web service* ficará rodando na porta 3000, exatamente o endereço que configuramos na nuvem.

### Comunicação com o Alexa Voice Service

Executaremos um cliente Java, que fará a comunicação do Raspberry Pi com o *Alexa Voice Service*, na nuvem da Amazon. Para isso, **em outro terminal**, dentro da mesma pasta **alexa-avs-sample-app/samples**, executamos o seguinte comando:

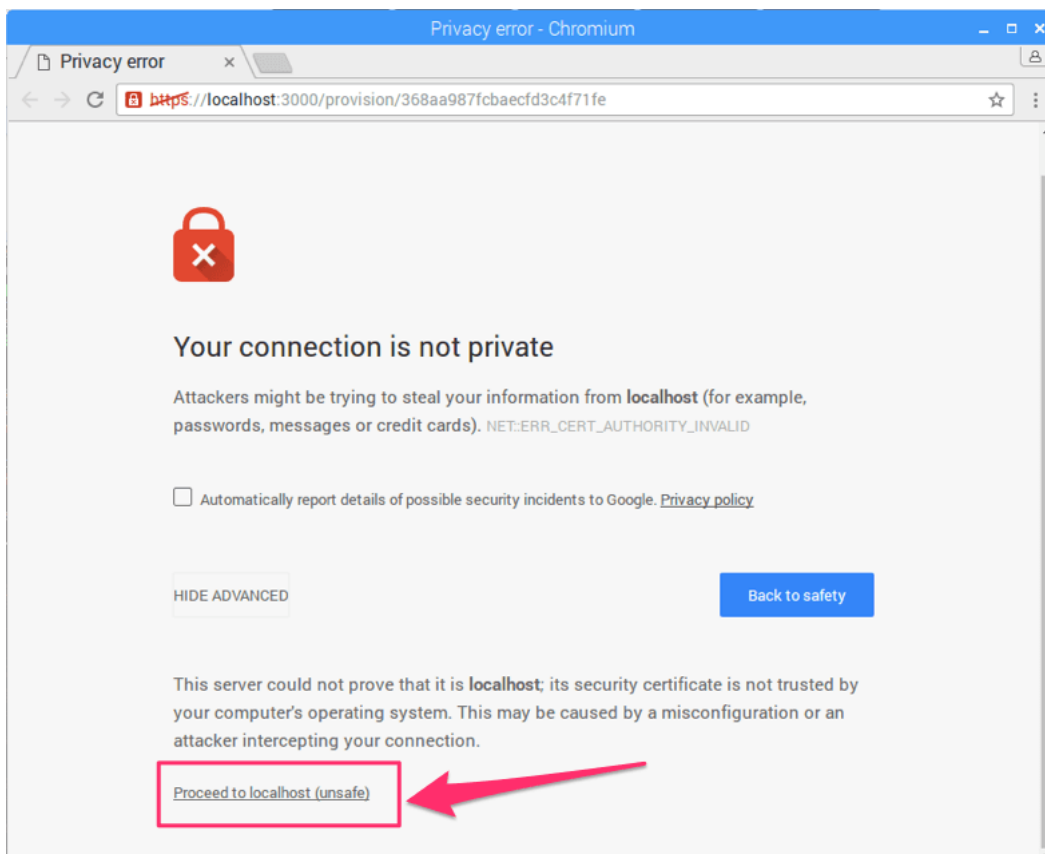
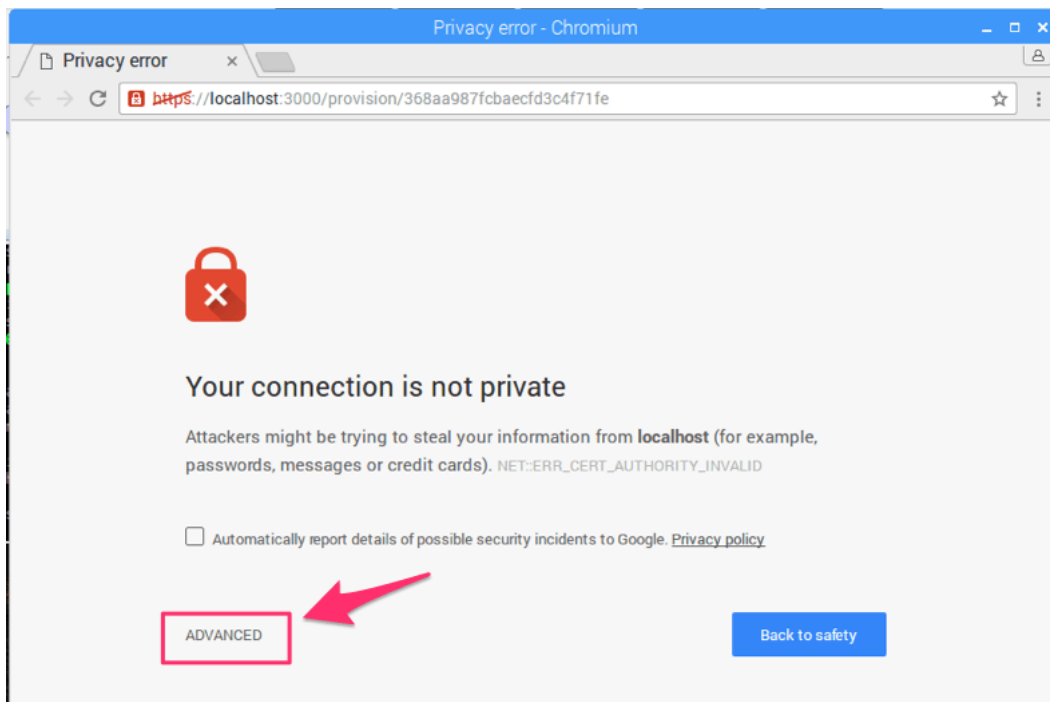
```
pi@raspberrypi:~/alexa-avs-sample-app/samples $ cd javaclient && mvn exec:exec
```

Esse comando demora um pouco mais e terminada a execução, o aplicativo é aberto:



Basta confirmar a inicialização do navegador com o token de autorização, clicando em **Yes**:

Com o navegador aberto, receberemos um alerta. Para prosseguir para o site, clicamos em **Advanced** -> **Proceed to localhost(unsafe)**:



Após isso, a tela de login da Amazon é aberto:



## Sign in to My Alexa Voice Service Sample App Security using your Amazon account

E-mail or mobile number:

What is your password?

☐ Keep me signed in. [Details](#)

[Forgot your password?](#)

[Create an Amazon.com account.](#)

**Login without hassle**  
Use Amazon to log into this site without another password.

**Login safely**  
Amazon does not share your password with this site.

[Learn More](#)

Fazemos o login com a nossa conta de desenvolvedor, confirmamos que o nosso dispositivo acesse o Perfil de Segurança criado anteriormente, clicando em **Okay**.



Hi [redacted] ) Not [redacted] ?

When you click "Okay", we'll provide [redacted] **Alexa Voice Service Sample App Security Profile**:

- Connectivity to Alexa so that you can interact with it using your voice.

By clicking "Okay", you also accept: [All Amazon terms found here](#)

Alexa processes and retains audio and other information in the cloud to provide and improve our

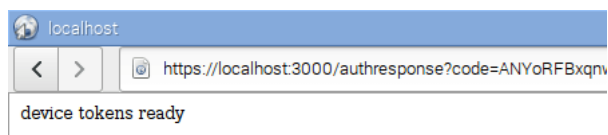
To remove access, visit [Your Account](#) at Amazon.

**Login without hassle**  
Use Amazon to log into this site without another password.

**Login safely**  
Amazon does not share your password with this site.

[Learn More](#)

Seremos redirecionados para uma URL, que começa com `https://localhost:3000/authresponse` com uma mensagem de sucesso:



Fechamos o navegador, voltamos ao cliente Java e clicamos em **OK**: